SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
No. 13-300252- -00000-0000
Fecha: 2013-12-26 14:52:55 Dep. 2020 DIR.NUEVASCR
Tra. 11 PATENTEIYII Eve: 378 FASENACIONALI
Act. 411 PRESENTACION Folios: 103

Delegatura de Propiedad Industrial
División de Nuevas Creaciones

SOLICITUD DE

PATENTE DE INVENCION
PCT ENTRADA EN FASE NACIONAL

21. EXPEDIENTE No. _____

54. TITULO: **DISPOSITIVO DE PROCESAMIENTO DE IMAGEN Y METODO DE PROCESAMIENTO DE IMAGEN**

52. CLASIFICACION INTERNACIONAL: **H04N 7/26**

71. SOLICITANTE: **SONY CORPORATION**

DOMICILIO: **1-7-1, Konan, Minato-ku, Tokyo 1080075, JAPON**

74. APODERADO: **Dra. MARGARITA CASTELLANOS A.**

22. BOGOTA, D.C., _____

Industria y Comercio
SUPERINTENDENCIA

SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO

de radicación

No. 13-300252- -00000-0000

Fecha: 2013-12-26 14:52:55 Dep. 2020 DIR.NUEVASCR
Tra. 11 PATENTEIYII Eve: 378 FASENACIONALI
Act. 411 PRESENTACION Folios: 103

DIRECCION DE NUEVAS CREACIONES
SOLICITUD FASE NACIONAL - PCT

1	**(12) TIPO DE SOLICITUD**	X Patente de Invención X Capítulo I	____ Patente de Modelo de Utilidad ____ Capítulo II

2 DATOS SOLICITUD INTERNACIONAL PCT (86)

Solicitud Internacional No.	**PCT/JP2012/063606**	**Fecha**	**28/05/2012**
Publicación Internacional No.	**WO2013/001957 A1**	**Fecha**	**03/01/2013**

3 (54) TITULO DE LA INVENCION (200 caracteres o espacios máximos)

DISPOSITIVO DE PROCESAMIENTO DE IMAGEN Y METODO DE PROCESAMIENTO DE IMAGEN

4 CLASIFICACION INTERNACIONAL (51) CIP

H04N 7/26

5 (71) SOLICITANTE(S) ____ Esta persona también es inventor — Para datos adicionales utilizar hoja de información adicional

APELLIDOS O RAZON SOCIAL	NOMBRE	IDENTIFICACION	TIPO
SONY CORPORATION			

6 DATOS DEL SOLICITANTE

DIRECCION	**1-7-1-, Konan, Minato-ku**	No. TELEFONO	
CIUDAD	**Tokyo 1080075**	E-MAIL	
DEPARTAMENTO/ESTADO		NACIONALIDAD O LUGAR DE CONSTITUCION	**JAPON**
PAIS DE RESIDENCIA	**JAPON**		

7 (72) INVENTOR(ES) — Para datos adicionales utilizar hoja de información adicional

	APELLIDOS	NOMBRES	NACIONALIDAD
1.	**IKEDA**	**MASARU**	**JAPONESA**
2.			
3.			
4.			
5.			

DIRECCION DE CORREO ELECTRONICO:

8 DATOS INVENTOR(ES)

	PAIS RESIDENCIA	DEPARTAMENTO/ESTADO	CIUDAD	DIRECCION
1.	**JAPON**		**TOKIO 1080075**	**C/O SONY CORPORATION, 1-7-1-, KONAN, MINATO-KU**
2.				
3.				
4.				
5				

OTRO(S) SOLICITANTE(S) Y/U OTRO(S) INVENTOR(ES)

____ Los demás solicitantes y/o demás inventores se indican en una hoja a continuación

9 (74) ____ **REPRESENTANTE** **X APODERADO**

APELLIDOS	NOMBRES	IDENTIFICACION
CASTELLANOS ABONDANO	**MARGARITA**	**C.C. 39.779.654 T.P. 70.819**

DIRECCION	**Calle 94A No. 7A-09**	No. TELEFONO	**7560770**
CIUDAD	**BOGOTA**	E-MAIL	castellanos@castellanosyco.co
PAIS	**COLOMBIA**	No. DE PROTOCOLO EN LA OFICINA	

10 **(30) DECLARACIONES DE PRIORIDAD** ___X___ SI _____ NO

	(33) PAIS DE ORIGEN	CODIGO PAIS	(31) NUMERO	(32) FECHA (AAA/MM/DD)
1.	JAPON	JP	2011-143461	28/06/2011
2.	JAPON	JP	2011-240550	01/11/2011
3.	JAPON	JP	2011-243839	07/11/2011
4.	JAPON	JP	2012-009326	19/01/2012

11 **DECLARACION SOBRE USO DE RECURSOS GENETICOS O BIOLOGICOS**

Declaro que el objeto de la presente solicitud de patente fue obtenido a partir de recursos genéticos o biológicos de los que cualquiera de los paises miembros de la Comunidad Andina es país de origen.

_____ SI ___X___ NO

NOTA: En caso afirmativo deberá anexar copia del contrato de acceso de recursos genéticos o productos derivados, o certificado o número de registro expedido por la Autoridad competente.

12 **DECLARACION SOBRE USO DE CONOCIMIENTOS TRADICIONALES**

Declaro que el objeto de la presente solicitud de patente fue obtenido a partir de conocimientos tradicionales de comunidades indígenas, afroamericanas o locales de países miembros de la Comunidad Andina.

_____ SI ___X___ NO

NOTA: En caso afirmativo deberá anexar la licencia o autorización de uso de conocimiento tradicional, o certificado, o número de registro expedido por la Autoridad competente.

13 **REDUCCION DE TASAS**

Declaro que carezco de medios económicos para presentar la solicitud de patente

_____ SI ___X___ NO

NOTA: En caso de ser persona natural y carecer de medios económicos, y por lo tanto, aplique la reducción de tasas a que se refiere la resolución vigente en tarifas, debe firmar la presente solicitud bajo la gravedad de juramento.

Micro, pequeñas y medianas empresas	Universidades públicas o privadas	Entidades sin animo de lucro
_____ Copia simple de la declaración de renta del año inmediatamente anterior, o en su defecto prueba documental idónea. _____ Documento de constancia de cumplimiento con lo establecido en la ley 905 de 2004.	_____ Copia acto de reconocimiento institucional emitido por el Ministerio de Educación. Acreditación Colciencias.	_____ Copia de registro vigente en Cámara de Comercio. _____ Hoja de información complementaria. _____ Otros, especificar.

14 **AUTORIZACION DE NOTIFICACION EN LINEA** _____ SI ___X___ NO

Manifiesto que he leído y entendido perfectamente los términos y condiciones de uso de medios electrónicos para las notificaciones en linea a través de Internet de los actos administrativos proferidos por la Superintendencia de Industria y Comercio que deber ser notificados personalmente y, en consecuencia, autorizo el servicio de notificación a través de internet.

14	COMPROBANTE DE PAGO O PAGO ELECTRONICO	N° 13-120652, 13-130561	Fecha: Noviembre 26, 2013 Diciembre 17, 2013

15 **FIRMA DEL SOLICITANTE, DEL APODERADO O DEL REPRESENTANTE LEGAL**
Junto a cada firma, indicar el nombre del firmante y su calidad (si tal calidad no es obvia al leer el petitorio)

MARGARITA CASTELLANOS ABONDANO - APODERADO

16 **ORDEN Y RELACION DE DOCUMENTOS QUE ACOMPAÑAN LA SOLICITUD**

Documentación Técnica
Solicitud Internacional en Castellano

1. ___X___ Descripción — N° de folios: **115**
2. ___X___ Reivindicaciones — N° de reivindicaciones: **7**
3. ___X___ Dibujos y/o figuras — N° de folios: **35**
4. ___X___ Resumen
5. _____ Certificado de depósito de material biológico si fuera el caso.
6. _____ Listado de secuencias de nucleótidos y/o aminoácidos en forma digital si fuera el caso.
7. ___X___ Arte final 12 x 12
8. ___X___ Anexo formato digital.

CD QUE CONTIENE LOS SIGUIENTES DOCUMENTOS:
* TEXTO EN ESPAÑOL DESCRIPCION, REIVINDICACIONES, RESUMEN Y DIBUJOS
* TEXTO DE LA PUBLICACION INTERNACIONAL EN INGLES

Documentación Jurídica

9. ___X___ Poderes, si fuera el caso.
10. _____ Documento que legalmente pruebe la cesión del inventor al solicitante o a su causante.
11. _____ Copia del contrato de acceso de recursos genéticos o productos derivados, certificado o número de registro, si fuera el caso.
12. _____ Copia de la licencia o autorización de Conocimientos Tradicionales, certificado o número de registro, si fuera el caso.
13. _____ Documento que acredita la existencia y representación legal, cuando el solicitante es persona jurídica.
14. ___X___ Comprobante de pago de la tasa de presentación de la solicitud.
15. _____ Comprobante de pago por reivindicación de prioridad
16. _____ Comprobante de pago de la tasa por concepto de excedente de palabras en la publicación.
17. ___X___ Comprobante de pago por reivindicación adicional a 10.

SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
NIT : 800.176.089-2
- / -

Industria y Comercio
SUPERINTENDENCIA

RECIBO DE CAJA

No. 13 - 120652

Bogotá D.C., Noviembre 26 de 2013 - 08:12:28

RECIBIDO DE : CASTELLANOS Y Y CO NI 800.174.641 RE

*** Soporte del Pago ***

TIPO PAGO	BANCO	CUENTA	No. PAGO	FECHA PAGO	VR PAGO
RECIBO DE CA	999999999	115788	14/11/2013		125.000.00
RECIBO DE CA	999999999	119278	21/11/2013		125.000.00
CONSIGNACION	BANCO DE BOGOTA	062754387	406302105	25/11/2013	3.505.000.00

*** Conceptos Pagados ***

CANT.	RENTISTICO	CONCEPTO	Vr.UNDITARIO	Vr.CONCEPTO
1	50005-01-01 SOLICITUDES	1 TRAMITES DE SOL. DE PATENTE DE INVENCION	500.000.00	500.000.00
				$500.000.00

SON: **QUINIENTOS MIL PESOS MONEDA CORRIENTE***

Responsable: _____
-******-

Recibo de Caja Aplicado al Expediente No. _____

SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
No. 13-300252- -00000-0000
Fecha: 2013-12-26 14:52:55 Dep. 2020 DIR.NUEVASCR
Tra. 11 PATENTEIYII Eve: 378 FASENACIONALI
Act. 411 PRESENTACION Folios: 103

SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
NIT : 800.176.089-2
- / -

Industria y Comercio
SUPERINTENDENCIA

RECIBO DE CAJA

No. 13 - 130561

Bogotá D.C., Diciembre 17 de 2013 - 15:19:38

RECIBIDO DE : CASTELLANOS Y Y CO NI 800.174.641

*** Soporte del Pago ***

TIPO PAGO	BANCO	CUENTA	No. PAGO	FECHA PAGO	VR PAGO
CONSIGNACION	BANCO DE BOGOTA	062754387	542730123	17/12/2013	4.143.000.00

*** Conceptos Pagados ***

CANT.	RENTISTICO	CONCEPTO	Vr.UNDITARIO	Vr.CONCEPTO
10	50005-01-01 SOLICITUDES	1607 REIVINDICACION PATENTE UNITARIA ADICIONAL A LAS 10 INIC	30.000.00	300.000.00
				$300.000.00

SON: **TRESCIENTOS MIL PESOS MONEDA CORRIENTE***

Responsable: _____
-******-

Recibo de Caja Aplicado al Expediente No. _____

SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
No. 13-300252- -00000-0000
Fecha: 2013-12-26 14:52:55 Dep. 2020 DIR.NUEVASCR
Tra. 11 PATENTEIYII Eve: 378 FASENACIONALI
Act. 411 PRESENTACION Folios: 103

4

COLOMBIA

PODER

El (los) **SONY CORPORATION**

Domiciliados en
1-7-1 Konan, Minato-ku, Tokyo 1080075, Japón

Nombran por el presente a **XIMENA CASTELLANOS A. y/o MARGARITA CASTELLANOS A.** abogadas con oficinas en la Calle 94A No. 7A-09, Bogotá-Colombia para

y para obtener la protección de mi (nuestra) propiedad Industrial e intelectual y con este fin autorizamos a dichos apoderados para que nos representen ante cualesquiera autoridades o personas de los órdenes administrativo, contencioso – administrativo y judicial en todo lo relacionado con los siguientes actos y gestiones:

a) Obtención de patentes de invención, modelos y dibujos industriales; registro de marcas de fabrica, de servicio, colectivas o defensivas y de denominaciones de origen, depósito de nombres y enseñas comerciales; su renovación, prórroga, inscripción de traspaso y cambio de nombre o domicilio, cancelación voluntaria y el registro de licencias de uso de esos derechos;

b) Para la obtención de registros sanitarios;

c) Para la obtención de registros de derechos de autor.

c) Acciones de oposición, observación, cancelación, nulidad, medidas cautelares y en general los juicios civiles, penales, administrativos o contencioso - administrativos que promovamos o se nos promuevan.

Pudiendo en todos los actos y gestiones arriba mencionados sustituir éste mandato entre los apoderados, transigir, desistir, recibir, cancelar, renunciar, aceptar notificaciones y nombrar apoderados judiciales y extra-judiciales. .

Finalmente, este poder ratifica la validez de todos los actos y solicitudes que, a nombre de la poderdante hubieren realizado los apoderados en relación con las facultades anteriormente enumeradas, antes de la suscripción del presente poder

Dado y firmado hoy 8 de Junio, 2010

en Tokyo, Japón

(El notario debe suscribir el documento en la pagina siguiente)
(Se requiere legalización consular o mediante apostilla)

POWER OF ATTORNEY

I (we) the undersigned **SONY CORPORATION**

Of 1-7-1 Konan, Minato-ku, Tokyo 1080075, Japan

hereby grant(s) Power of Attorney to **XIMENA CASTELLANOS A. and /or MARGARITA CASTELLANOS** a. attorneys residing at Calle 94A No. 7A-09, Bogotá- Colombia, to

and to obtain the protection of my (our) Intellectual Property, for which purpose we authorize the said attorneys to represent us before any authorities, or persons, specially those, administrative, contentious and judicial, in all matters concerning the following:

a) Obtainment of patents, models and designs, registration of trademarks, service, collective and defensive marks, and of designations of origin; the deposit of trade names and ensigns; its renewal, extension, recordal of assignments, change of name or domicile, voluntary cancellation, and the registration of licenses of use of the above rights;

b) For the obtainment of Sanitary registrations;

c) For the obtainment of copyright registrations

Actions of opposition, cancellation, nullity, infringement, granting of licenses; and in general the civil, criminal and administrative suits related to those rights; actions and suits instituted by us or against us;
In all the above matters they may substitute this power of attorney within the appointed attorneys, compromise, cancel, receive, withdraw, renounce, accept notifications and appoint attorneys in fact in or out of court.

Finally, this power of attorney ratifies the validity of all acts and applications made by the representatives in the name of the signatory of this power of attorney, with regard to the faculties above given, before the execution of the present power of attorney.

Given and signed this 8th day of June of the year 2010
in Tokyo, Japan
by

KOICHI WADA
Deputy General Manager, Planning&Control Department
Intellectual Property Division

(Notary shall execute on the other side)
(Consular legalization or apostille is required)

CASTELLANOS & Co.
INTELLECTUAL PROPERTY LAW FIRM

CALLE 94 A No. 7 A – 09 / P.O. BOX 6349
PHONE (57 1) 756 07 70 / FAX (57 1) 755 01 58
BOGOTÁ D.C. COLOMBIA
info@castellanosyco.com

5

Registered No. 1030

NOTARIAL CERTIFICATE

This is to certify that ***Koichi Wada, Deputy General Manager, Planning & Control Department, Intellectual Property Division of SONY CORPORATION,*** has acknowledged himself in my very presence that the signature on the foregoing document is his own.

August 4, 2010

Akihiko Mitsuda

AKIHIKO MITSUDA

Notary attached to
The Tokyo Legal Affairs Bureau
NO.9-6 3-CHOME AKASAKA
MINATOKU TOKYO JAPAN

TOKYO LEGAL AFFAIRS BUREAU
NOTARY
NO. 9-6, 3-CHOME, AKASAKA
MINATOKU, TOKYO, JAPAN

P A T E N T S

CESION

El (los) suscrito (s)

en consideración al pago de la suma de (.....), que he(mos) recibido, por el presente documento cedo (cedemos) a

una compañía constituida de conformidad con las leyes de

con oficinas en

todos los derechos y títulos sobre la invención denominada

EL CESIONARIO firma también el presente y declara que acepta dicha cesión

Dado y firmado hoy

en

ASSIGNMENT

The undersigned

Masaru IKEDA

in consideration of the sum of (.....), the receipt of which is hereby acknowledged, by these presents do sell, assign and transfer unto
Sony Corporation

a corporation organized under the laws of Japan

having a place of business at
1-7-1 Konan, Minato-Ku, Tokyo, Japan

all rights and title to the invention related to

IMAGE PROCESSING DEVICE AND IMAGE PROCESSING METHOD

THE ASSIGNEE signs these presents and declares that accepts the aforesaid assignment

Given an signed this

in

November 13, 2013

Inventors

Masaru IKEDA

Assignee

Wataru Kato
General Manager, Planning & Control Department,
Intellectual Property Division

CASTELLANOS & Co.
INTELLECTUAL PROPERTY LAW FIRM

CALLE 94 A No. 7 A – 09 / P.O. BOX 6349
PHONE (57 1) 756 07 70 / FAX (57 1) 755 01 58
BOGOTÁ D.C. COLOMBIA
castellanos@castellanosyco.co

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2013 年 1 月 3 日(03.01.2013)

WIPO | PCT

(10) 国際公開番号
WO 2013/001957 A1

(51) 国際特許分類:
H04N 7/26 (2006.01)

(21) 国際出願番号: PCT/JP2012/063606

(22) 国際出願日: 2012 年 5 月 28 日(28.05.2012)

(25) 国際出願の言語: 日本語

(26) 国際公開の言語: 日本語

(30) 優先権データ:
特願 2011-143461 2011 年 6 月 28 日(28.06.2011) JP
特願 2011-240550 2011 年 11 月 1 日(01.11.2011) JP
特願 2011-243839 2011 年 11 月 7 日(07.11.2011) JP
特願 2012-009326 2012 年 1 月 19 日(19.01.2012) JP

(71) 出願人 (米国を除く全ての指定国について): ソニー株式会社(SONY CORPORATION) [JP/JP]; 〒1080075 東京都港区港南 1 丁目 7 番 1 号 Tokyo (JP).

(72) 発明者; および
(75) 発明者/出願人 (米国についてのみ): 池田 優(IKEDA Masaru) [JP/JP]; 〒1080075 東京都港区港南 1 丁目 7 番 1 号 ソニー株式会社内 Tokyo (JP).

(74) 代理人: 稲本 義雄, 外(INAMOTO Yoshio et al.); 〒1600023 東京都新宿区西新宿 7 丁目 5 番 25 号 西新宿木村屋ビルディング 9 階 Tokyo (JP).

(81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

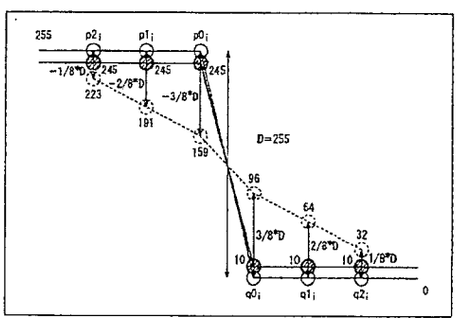
(84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

添付公開書類:
— 国際調査報告 (条約第 21 条(3))

(54) **Title:** IMAGE PROCESSING DEVICE AND IMAGE PROCESSING METHOD

(54) 発明の名称 : 画像処理装置と画像処理方法

図21



(57) **Abstract:** The present technology relates to an image processing device and an image processing method by which filtering can be appropriately performed in a deblocking filtering process. The value of a pixel ($p0_i$) which is 255 (solid line) prior to a deblocking process changes greatly after a conventional deblocking process, becoming 159 (dotted line). Therefore, a clipping process with a clipping value of 10 is performed in strong filtering, and thereby, the value of the pixel ($p0_i$) which is 255 (solid line) prior to the deblocking process becomes 245 (thick line), and the conventional large change in pixel values can be reduced to a minimum. The disclosure can be applied to, for example, an image processing device.

(57) 要約:

[続葉有]

WO 2013/001957 A1

本技術は、デブロッキングフィルタ処理において、適切にフィルタリングがかかるようにする画像処理装置と画像処理方法に関する。デブロッキング処理前には、２５５（実線）である画素ｐ０ｉの値は、従来のデブロッキング処理後に１５９（点線）となり、大きく変化してしまう。そこで、強フィルタリングにおいてクリップ値が１０のクリップ処理を行うことにより、デブロッキング処理前には、２５５（実線）である画素ｐ０ｉの値は、２４５（太線）となり、従来において生じていた画素値の変化を最小限に抑えることができる。本開示は、例えば、画像処理装置に適用することができる。

DESCRIPTION

IMAGE PROCESSING DEVICE AND IMAGE PROCESSING METHOD

TECHNICAL FIELD

[0001]

This technique relates to an image processing device and an image processing method. Specifically, this technique allows a deblocking filtering process to apply filtering appropriately.

BACKGROUND ART

[0002]

In recent years, devices, which handle image information as digital data, which, in such a case, aim to transmit and store information with a high efficiency, and which conform to a scheme, such as MPEG2 (International Organization for Standardization and International Electrotechnical Commission (ISO/IEC) 13818-2), for compressing image information using orthogonal transformation, such as discrete cosine transformation, and using motion compensation by utilizing redundancy that is unique to the image information have become widespread in both information distribution in broadcasting stations and information reception in ordinary homes. In addition, schemes called H.264 and MPEG4 Part10 (Advanced Video Coding (AVC) which require a larger amount of operation for coding and decoding but can realize a higher coding efficiency than MPEG2 or the like have been also used. Further, nowadays, standardization works for high efficiency video coding (HEVC) which is a next-generation image coding scheme are

in progress so that compression, distribution, or the like of high-resolution images of 4000 × 2000 pixels which is four times that of high-vision images can be performed efficiently.

[0003]

In the standardization works for high efficiency video coding (HEVC) which is a next-generation image coding scheme, JCTVC-A119 (see Non-Patent Document 1 below) proposes to apply a deblocking filter to each block having the size of 8 × 8 pixels or larger. In the method proposed in JCTVC-A119, by increasing a minimum unit block size to which a deblocking filter is to be applied, it is possible to execute a filtering process in parallel on a plurality of block boundaries in the same direction within one macroblock.

CITATION LIST

NON-PATENT DOCUMENT

[0004]

Non-Patent Document 1: K. Ugur (Nokia), K. R. Andersson (LM Ericsson), A. Fuldseth (Tandberg Telecom), "JCTVC-A119: Video coding technology proposal by Tandberg, Nokia, and Ericsson," Documents of the first meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Dresden, Germany, 15-23 April, 2010

SUMMARY OF THE INVENTION

PROBLEMS TO BE SOLVED BY THE INVENTION

[0005]

However, in the conventional deblocking filtering process, filtering is not performed appropriately.

to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block;

controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and

encoding the image in units each having a layer structure using an image to which the deblocking filter is applied.

REFERENCE SIGNS LIST

[0483]

10: Image encoding device
11: A/D converter
12, 57: Frame reordering buffer
13: Subtractor
14: Orthogonal transformer
15: Quantizer
16: Lossless encoder
17: Accumulation buffer
18: Rate controller
21, 53: Inverse quantizer
22, 54: Inverse orthogonal transformer
23, 55: Adder
24, 56: Deblocking filtering unit
25, 61: Frame memory
26, 62, 65: Selector
31, 63: Intra predictor
32: Motion estimator/compensator

33: Predicted image/optimal mode selecting unit

50: Image decoding device

51: Accumulation buffer

52: Lossless decoder

61: Frame memory

64: Motion compensation unit

71: Image memory

72, 72-1: Block boundary determining unit

72-2: Line boundary/block boundary determining unit

73, 73-1: Filter strength determining unit

73-2: Line boundary filter strength determining unit

74, 74-1: Filter operation unit

74-2: Line boundary filter operation unit

75: Selector

76: Coefficient memory

77: Controller

77A, 77B: Intra-LCU line determining unit

241: Line memory

242: Line boundary detecting unit

243: Filter strength determining unit

244: Coefficient memory

245: Filter operation unit

246: Filter controller

2451: Data storage unit

2452: Data selecting unit

2453, 2455, 2456: Arithmetic processing unit

2457: Data selecting unit

2461: Line boundary determining unit

CLAIMS

1. An image processing device comprising:

a decoding unit that decodes an encoded stream encoded in units each having a layer structure to generate an image;

a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the image generated by the decoding unit and an adjacent block adjacent to the block; and

a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the image generated by the decoding unit.

2. The image processing device according to claim 1, wherein

the controller applies the clipping process to values of parts that change when the deblocking filter is applied.

3. The image processing device according to claim 2, wherein

the controller applies the clipping process according to the following expressions:

[Mathematical formula 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

4. The image processing device according to claim 3, wherein

the controller sets a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

5. The image processing device according to claim 4, wherein

the controller sets a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

6. The image processing device according to claim 1, further comprising:

a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, wherein

the filtering unit applies the deblocking filter to the block boundary according to the strength determined

by the filter strength determining unit, and

the controller controls the filtering unit so that, when the filter strength determining unit determines that strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the decoding unit.

7. The image processing device according to claim 6, wherein

the filter strength determining unit determines the strength of the deblocking filter using a plurality of lines as processing units.

8. The image processing device according to claim 7, wherein

the filter strength determining unit determines the strength of the deblocking filter using four lines as the processing units.

9. The image processing device according to claim 7, further comprising:

a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, wherein

the filter strength determining unit determines the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

10. An image processing method for allowing an image processing device to execute:

decoding an encoded stream encoded in units each having a layer structure to generate an image;

applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the generated image and an adjacent block adjacent to the block; and

controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the generated image.

11. An image processing device comprising:

a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block;

a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and

an encoding unit that encodes the image in units each having a layer structure using an image to which the deblocking filter is applied.

12. The image processing device according to claim 11,

wherein

the controller applies the clipping process to values of parts that change when the deblocking filter is applied.

13. The image processing device according to claim 12, wherein

the controller applies the clipping process according to the following expressions:

[Mathematical formula 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

14. The image processing device according to claim 13, wherein

the controller sets a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

15. The image processing device according to claim 14, wherein

the controller sets a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

16. The image processing device according to claim 11, further comprising:

a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, wherein

the filtering unit applies the deblocking filter to the block boundary according to the strength determined by the filter strength determining unit, and

the controller controls the filtering unit so that, when the filter strength determining unit determines that strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the decoding unit.

17. The image processing device according to claim 16, wherein

the filter strength determining unit determines the strength of the deblocking filter using a plurality of lines as processing units.

18. The image processing device according to claim 17, wherein

the filter strength determining unit determines the strength of the deblocking filter using four lines as the

processing units.

19. The image processing device according to claim 17, further comprising:

a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, wherein

the filter strength determining unit determines the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

20. An image processing method for allowing an image processing device to execute:

applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block;

controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and

encoding the image in units each having a layer structure using an image to which the deblocking filter is applied.

ABSTRACT

The present technology relates to an image processing device and an image processing method which allow a deblocking filtering process to apply filtering appropriately. A pixel ($p0_i$) of which the value is 255 (solid line) before a deblocking process changes greatly to 159 (dot line) after a conventional deblocking process. Therefore, a clipping process having a clipping value of 10 is performed in strong filtering, whereby the pixel ($p0_i$) of which the value is 255 (solid line) before the deblocking process becomes 245 (bold line). Thus, a change in the pixel value occurring in the conventional technique can be suppressed as much as possible. This disclosure can be applied to an image processing device, for example.

FIG. 1

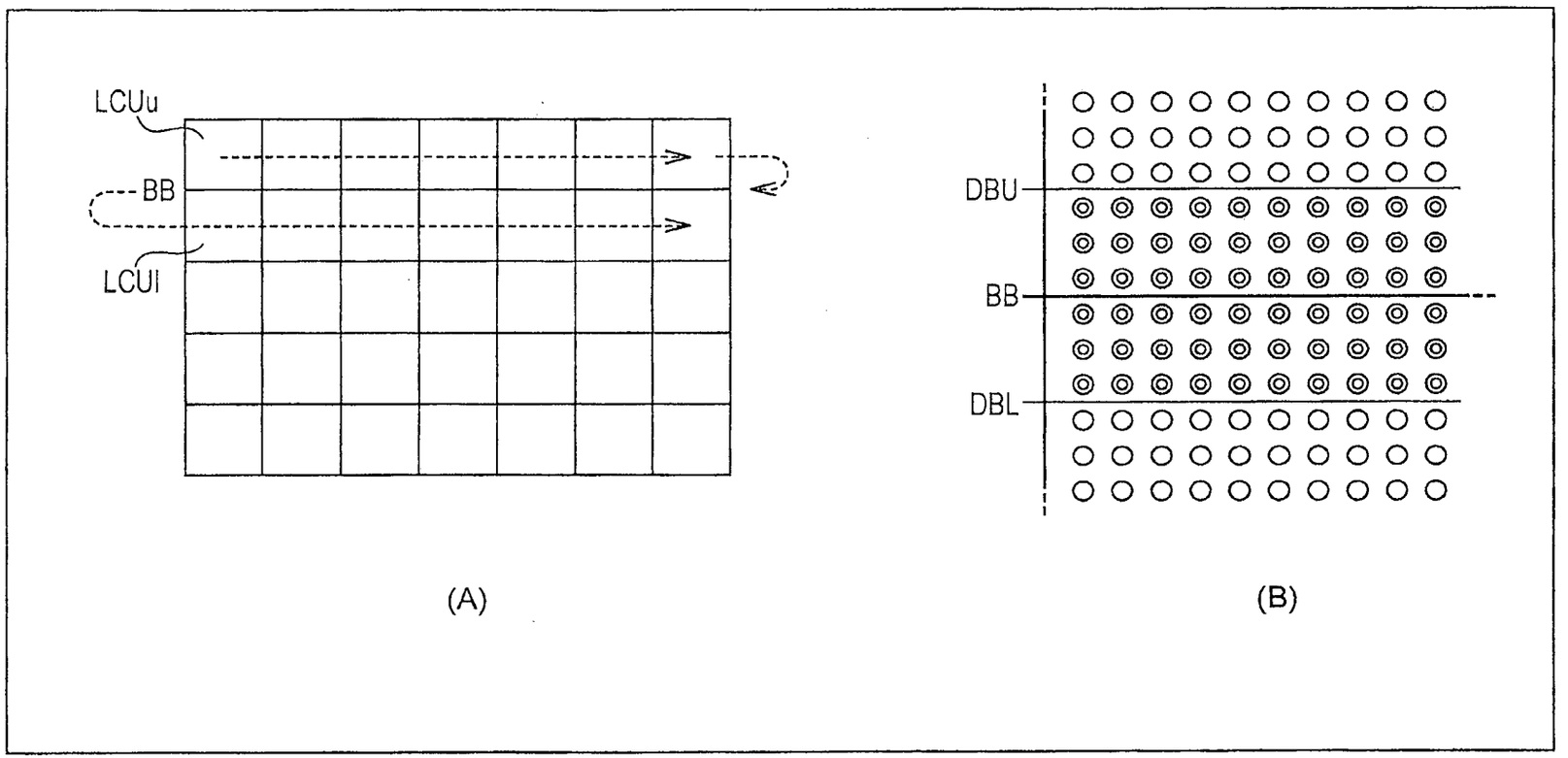


FIG. 2

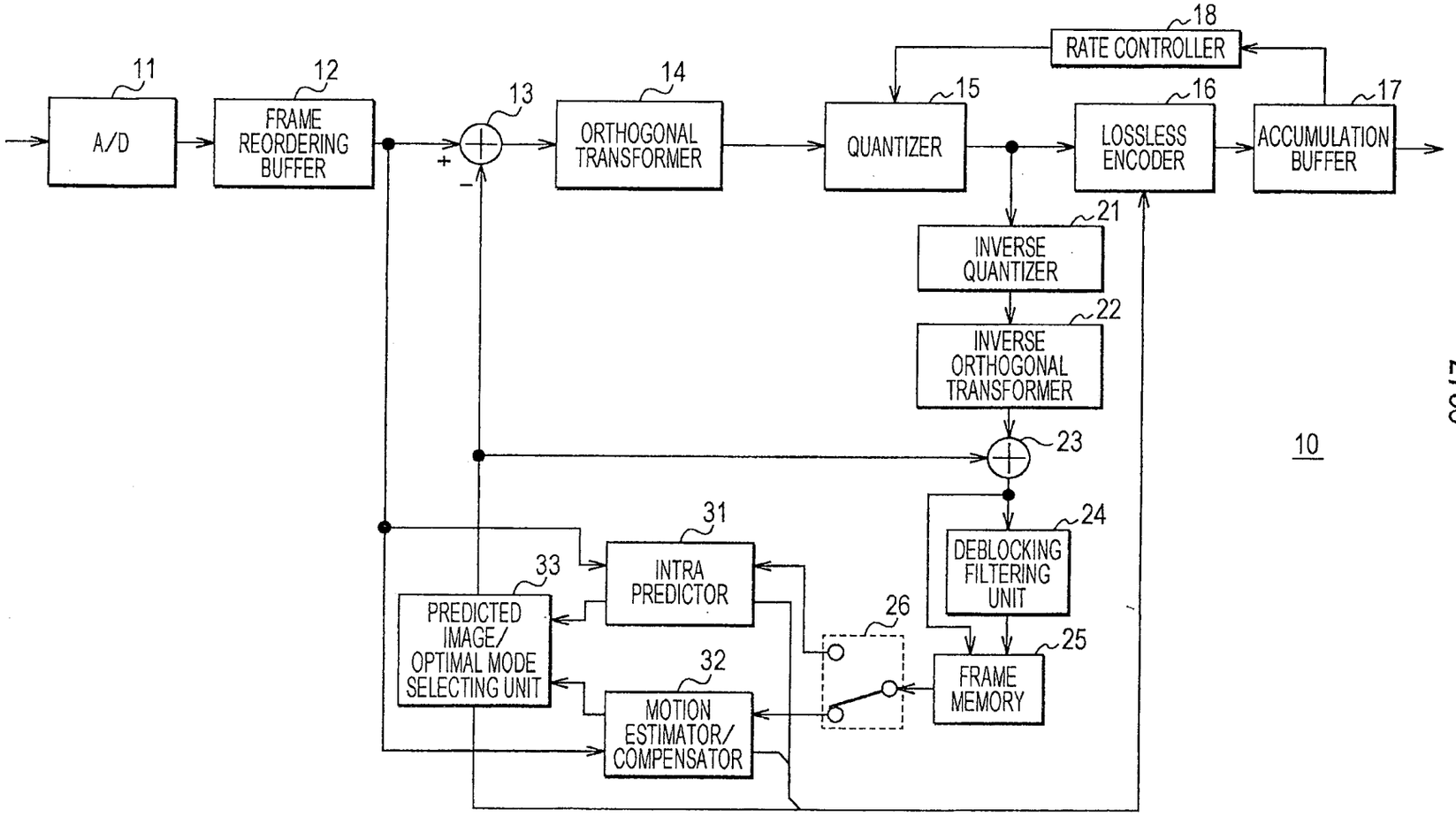

11
A/D
12
FRAME REORDERING BUFFER
13
+
−
14
ORTHOGONAL TRANSFORMER
15
QUANTIZER
18
RATE CONTROLLER
16
LOSSLESS ENCODER
17
ACCUMULATION BUFFER
21
INVERSE QUANTIZER
22
INVERSE ORTHOGONAL TRANSFORMER
23
24
DEBLOCKING FILTERING UNIT
25
FRAME MEMORY
26
31
INTRA PREDICTOR
32
MOTION ESTIMATOR/ COMPENSATOR
33
PREDICTED IMAGE/ OPTIMAL MODE SELECTING UNIT
10

FIG. 3

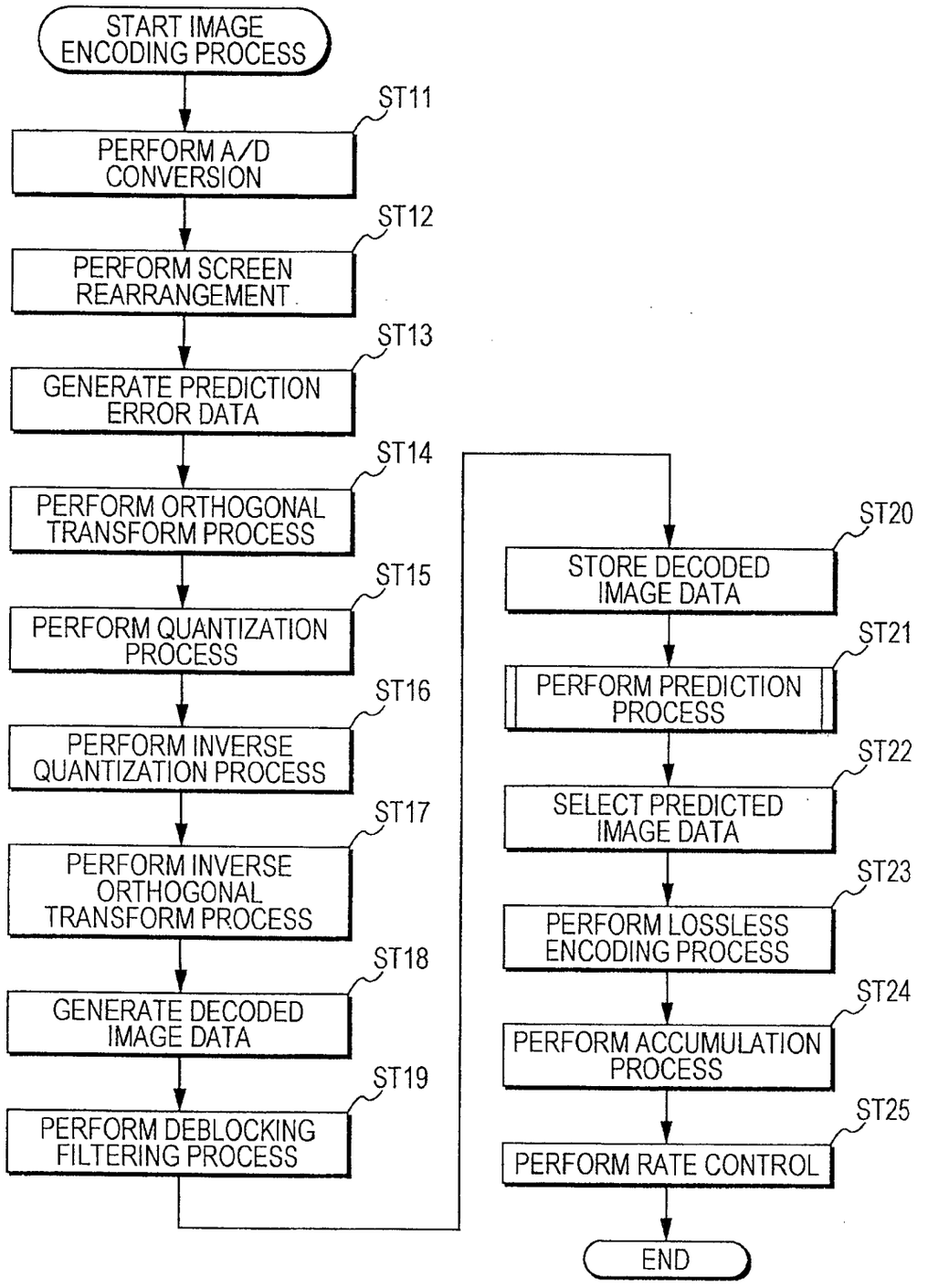

START IMAGE ENCODING PROCESS
ST11
PERFORM A/D CONVERSION
ST12
PERFORM SCREEN REARRANGEMENT
ST13
GENERATE PREDICTION ERROR DATA
ST14
PERFORM ORTHOGONAL TRANSFORM PROCESS
ST15
PERFORM QUANTIZATION PROCESS
ST16
PERFORM INVERSE QUANTIZATION PROCESS
ST17
PERFORM INVERSE ORTHOGONAL TRANSFORM PROCESS
ST18
GENERATE DECODED IMAGE DATA
ST19
PERFORM DEBLOCKING FILTERING PROCESS
ST20
STORE DECODED IMAGE DATA
ST21
PERFORM PREDICTION PROCESS
ST22
SELECT PREDICTED IMAGE DATA
ST23
PERFORM LOSSLESS ENCODING PROCESS
ST24
PERFORM ACCUMULATION PROCESS
ST25
PERFORM RATE CONTROL
END

FIG. 4

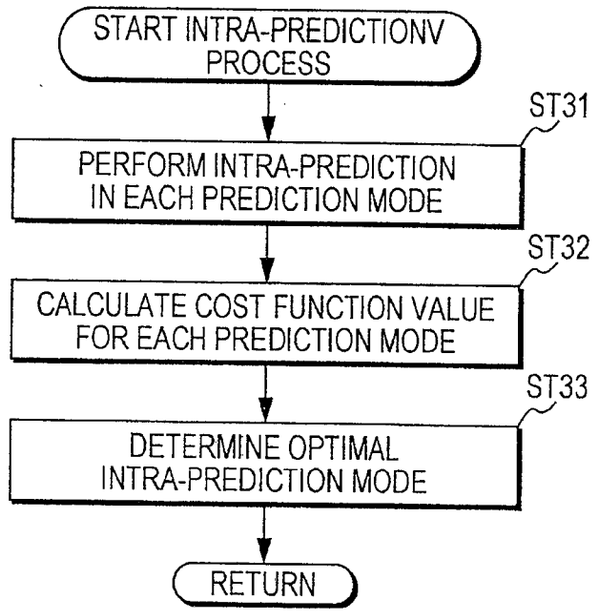

START INTRA-PREDICTIONV PROCESS
ST31
PERFORM INTRA-PREDICTION IN EACH PREDICTION MODE
ST32
CALCULATE COST FUNCTION VALUE FOR EACH PREDICTION MODE
ST33
DETERMINE OPTIMAL INTRA-PREDICTION MODE
RETURN

FIG. 5

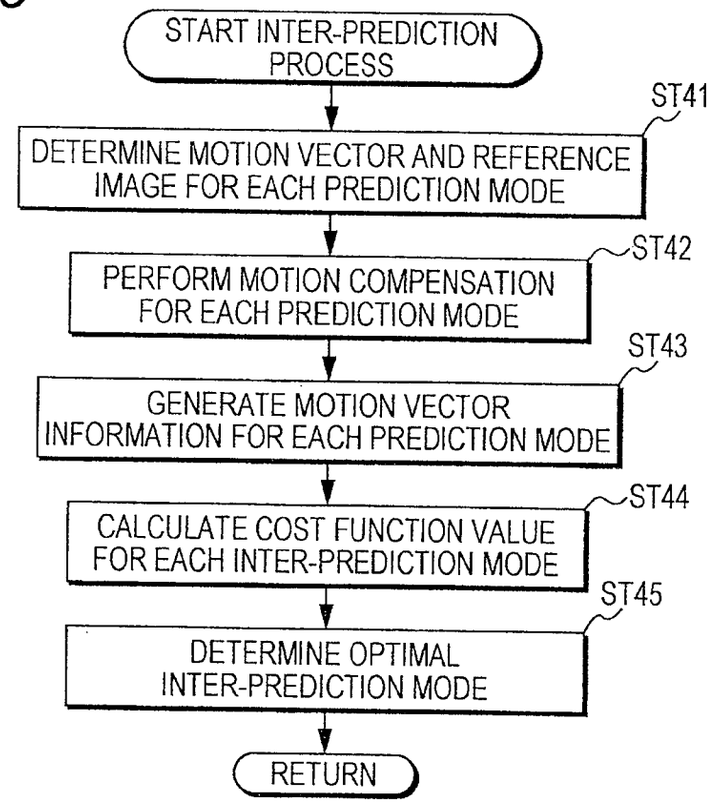

START INTER-PREDICTION PROCESS
ST41
DETERMINE MOTION VECTOR AND REFERENCE IMAGE FOR EACH PREDICTION MODE
ST42
PERFORM MOTION COMPENSATION FOR EACH PREDICTION MODE
ST43
GENERATE MOTION VECTOR INFORMATION FOR EACH PREDICTION MODE
ST44
CALCULATE COST FUNCTION VALUE FOR EACH INTER-PREDICTION MODE
ST45
DETERMINE OPTIMAL INTER-PREDICTION MODE
RETURN

FIG. 6

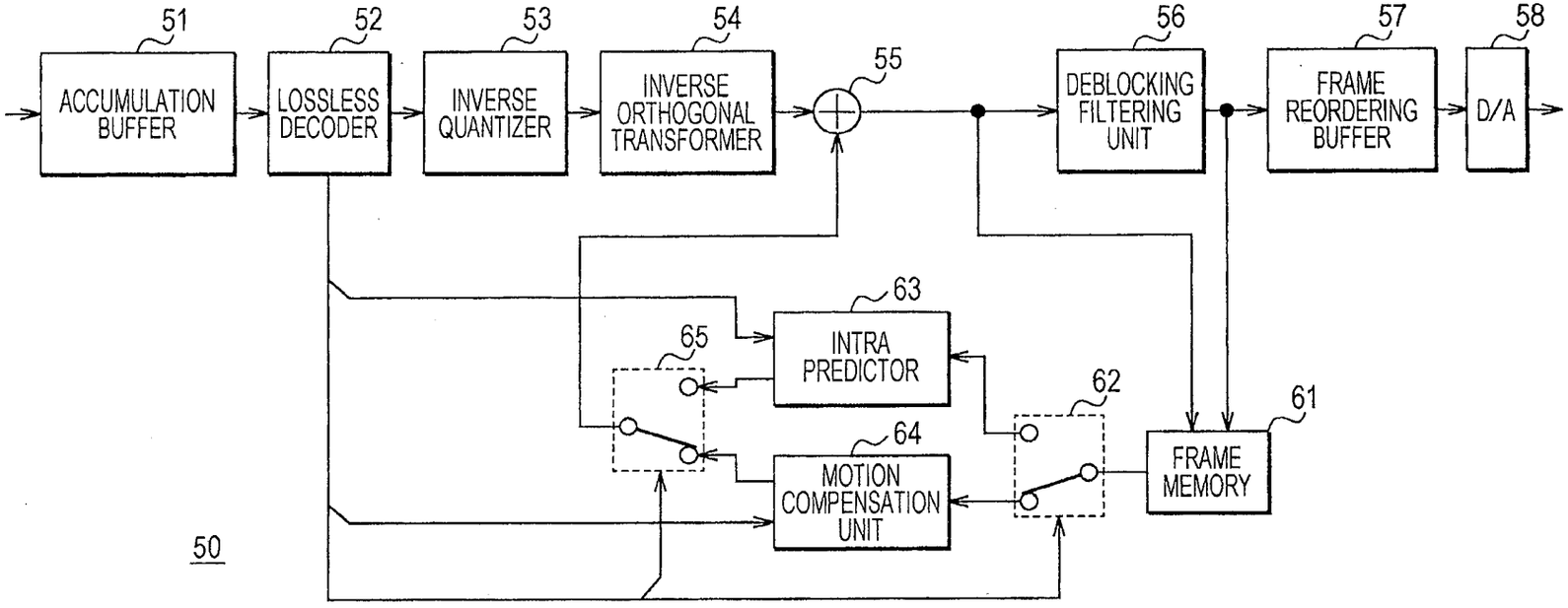

51
ACCUMULATION BUFFER
52
LOSSLESS DECODER
53
INVERSE QUANTIZER
54
INVERSE ORTHOGONAL TRANSFORMER
55
56
DEBLOCKING FILTERING UNIT
57
FRAME REORDERING BUFFER
58
D/A
63
INTRA PREDICTOR
65
62
61
64
MOTION COMPENSATION UNIT
FRAME MEMORY
50

FIG. 7

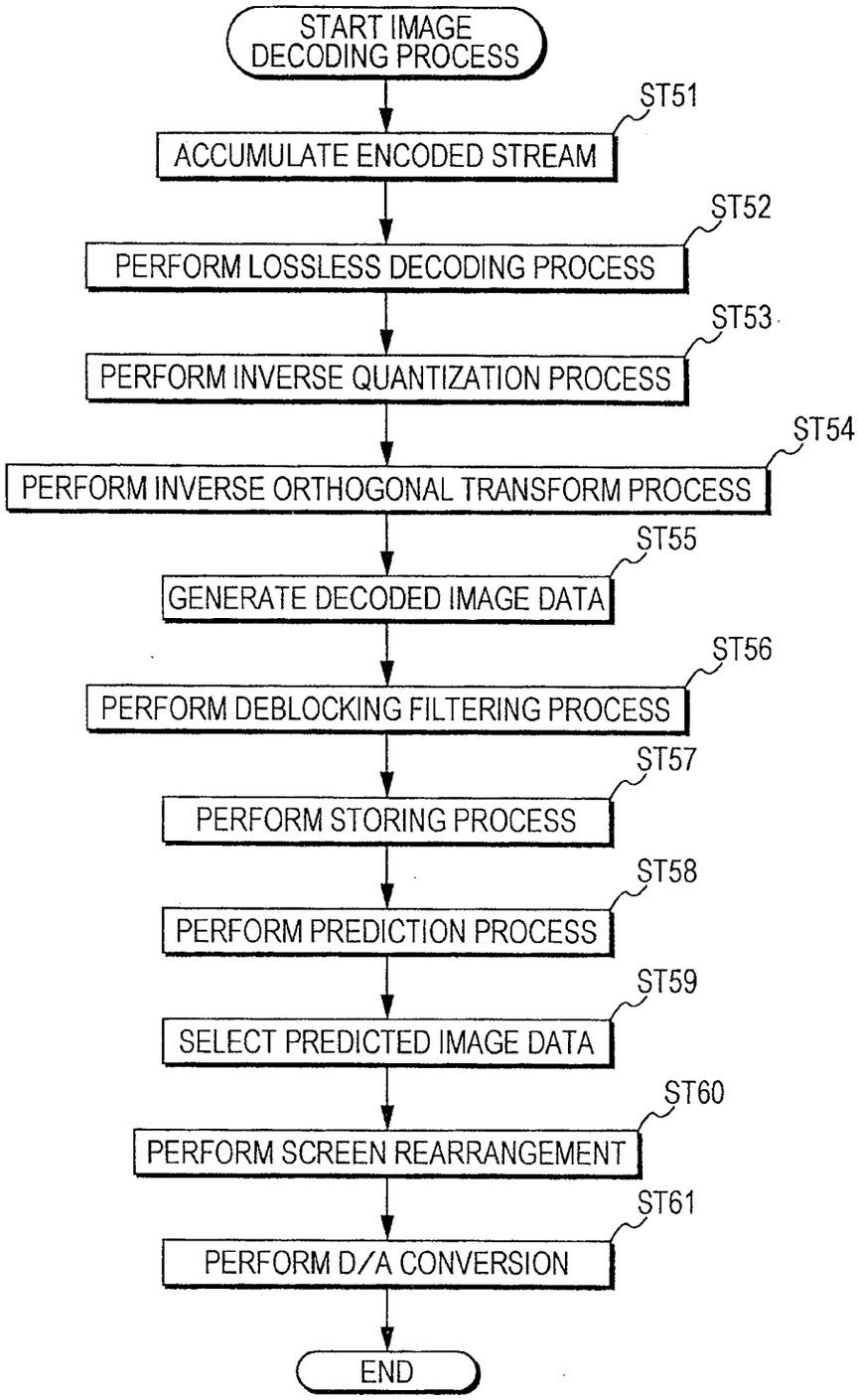

START IMAGE DECODING PROCESS
ST51
ACCUMULATE ENCODED STREAM
ST52
PERFORM LOSSLESS DECODING PROCESS
ST53
PERFORM INVERSE QUANTIZATION PROCESS
ST54
PERFORM INVERSE ORTHOGONAL TRANSFORM PROCESS
ST55
GENERATE DECODED IMAGE DATA
ST56
PERFORM DEBLOCKING FILTERING PROCESS
ST57
PERFORM STORING PROCESS
ST58
PERFORM PREDICTION PROCESS
ST59
SELECT PREDICTED IMAGE DATA
ST60
PERFORM SCREEN REARRANGEMENT
ST61
PERFORM D/A CONVERSION
END

FIG. 8

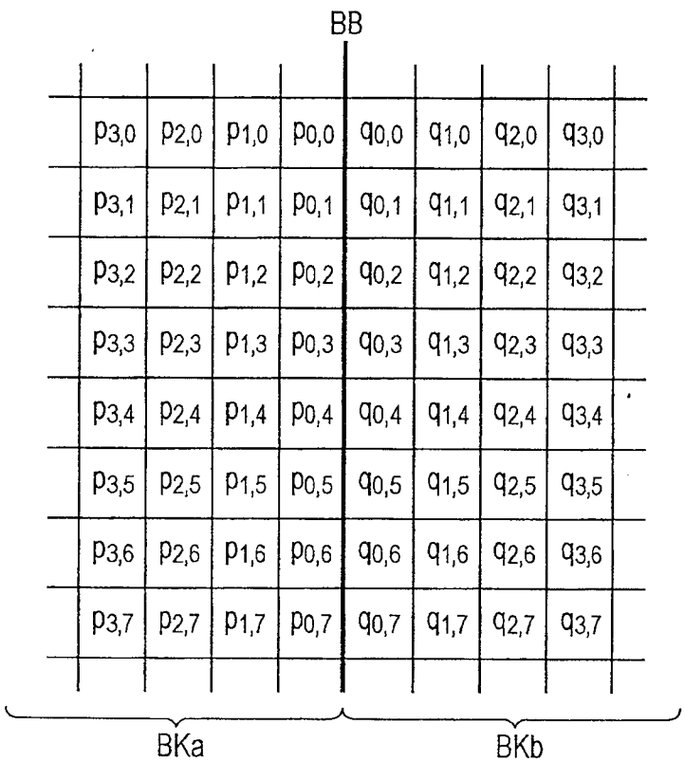

BB
p3,0 p2,0 p1,0 p0,0 q0,0 q1,0 q2,0 q3,0
p3,1 p2,1 p1,1 p0,1 q0,1 q1,1 q2,1 q3,1
p3,2 p2,2 p1,2 p0,2 q0,2 q1,2 q2,2 q3,2
p3,3 p2,3 p1,3 p0,3 q0,3 q1,3 q2,3 q3,3
p3,4 p2,4 p1,4 p0,4 q0,4 q1,4 q2,4 q3,4
p3,5 p2,5 p1,5 p0,5 q0,5 q1,5 q2,5 q3,5
p3,6 p2,6 p1,6 p0,6 q0,6 q1,6 q2,6 q3,6
p3,7 p2,7 p1,7 p0,7 q0,7 q1,7 q2,7 q3,7
BKa
BKb

FIG. 9

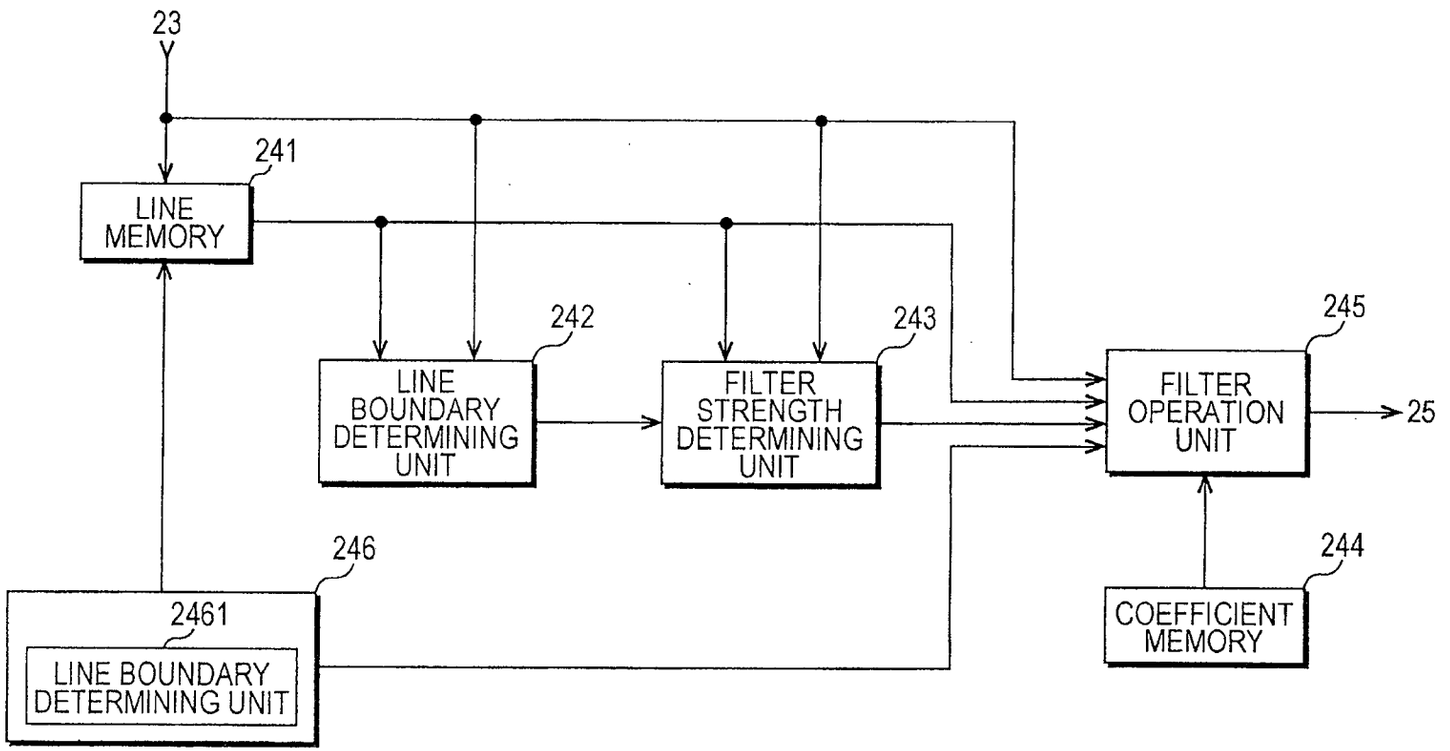

23
241
LINE MEMORY
242
LINE BOUNDARY DETERMINING UNIT
243
FILTER STRENGTH DETERMINING UNIT
245
FILTER OPERATION UNIT
25
246
2461
LINE BOUNDARY DETERMINING UNIT
244
COEFFICIENT MEMORY

FIG. 10

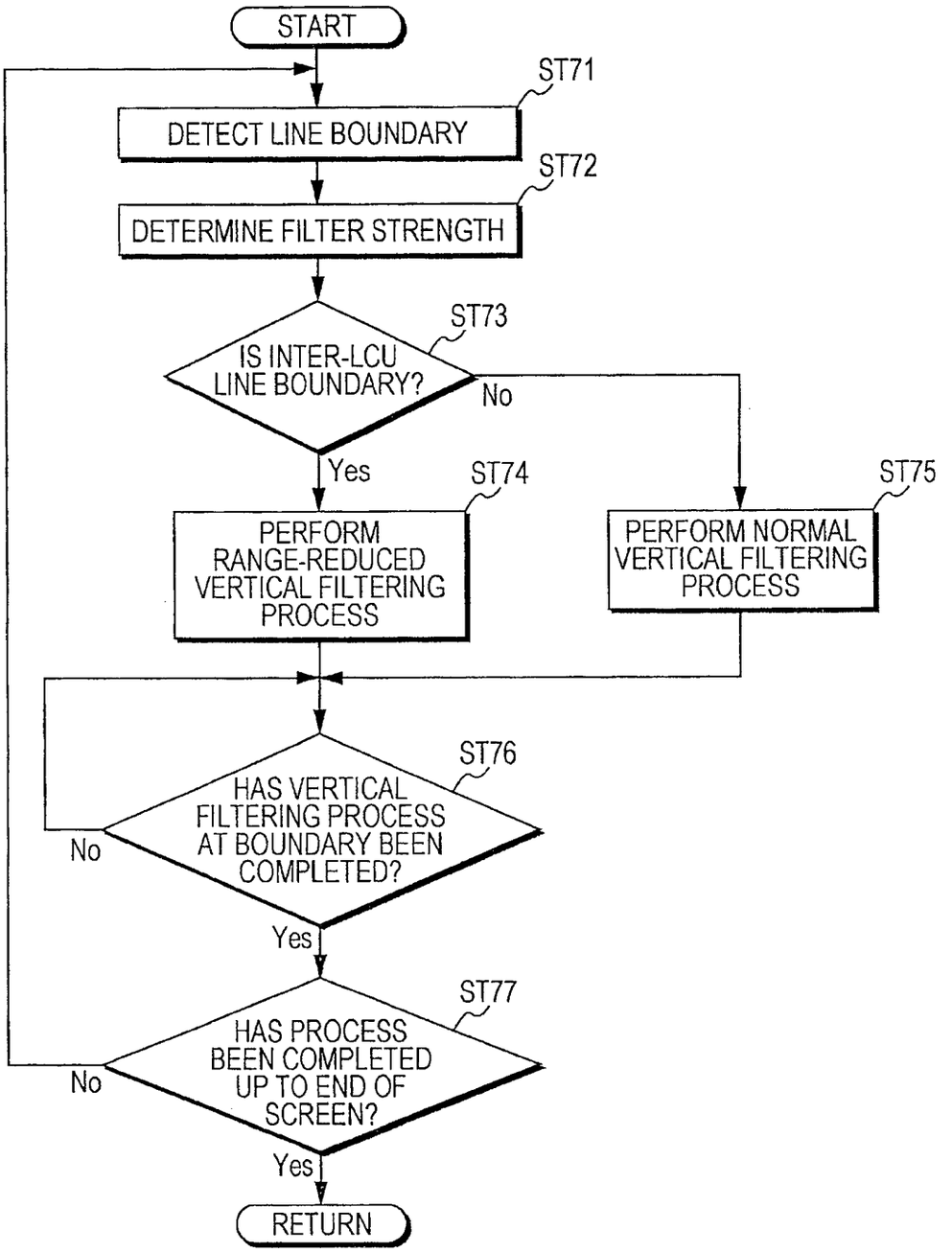

START
ST71
DETECT LINE BOUNDARY
ST72
DETERMINE FILTER STRENGTH
ST73
IS INTER-LCU LINE BOUNDARY?
No
Yes
ST74
PERFORM RANGE-REDUCED VERTICAL FILTERING PROCESS
ST75
PERFORM NORMAL VERTICAL FILTERING PROCESS
ST76
HAS VERTICAL FILTERING PROCESS AT BOUNDARY BEEN COMPLETED?
No
Yes
ST77
HAS PROCESS BEEN COMPLETED UP TO END OF SCREEN?
No
Yes
RETURN

FIG. 11

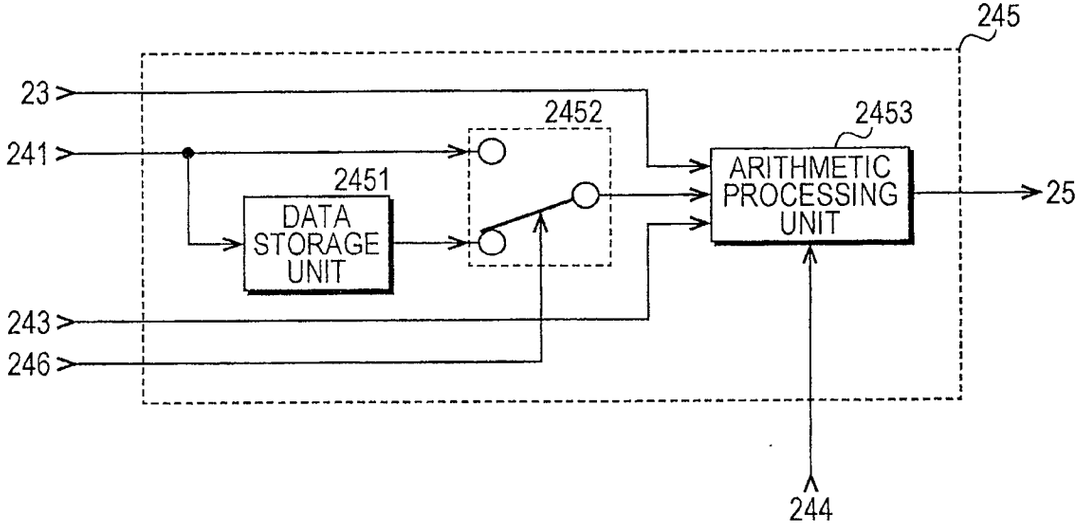

245
23
241
2452
2453
2451
DATA STORAGE UNIT
ARITHMETIC PROCESSING UNIT
25
243
246
244

FIG. 12

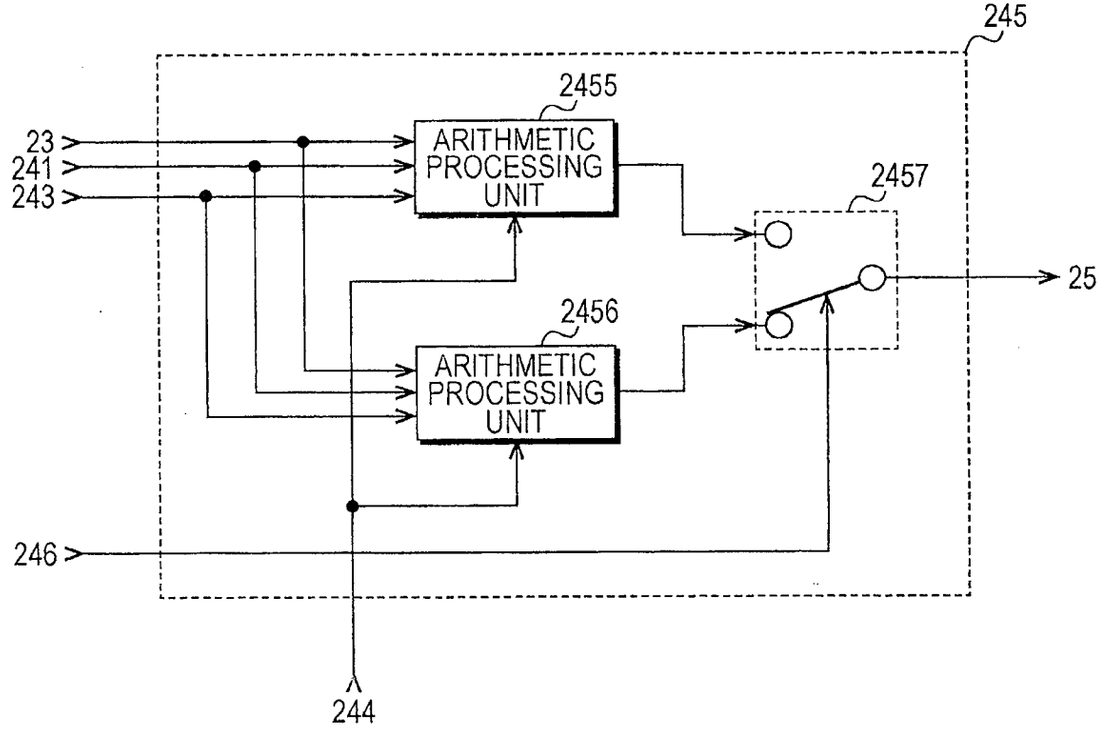

245
2455
ARITHMETIC PROCESSING UNIT
23
241
243
2457
25
2456
ARITHMETIC PROCESSING UNIT
246
244

FIG. 13

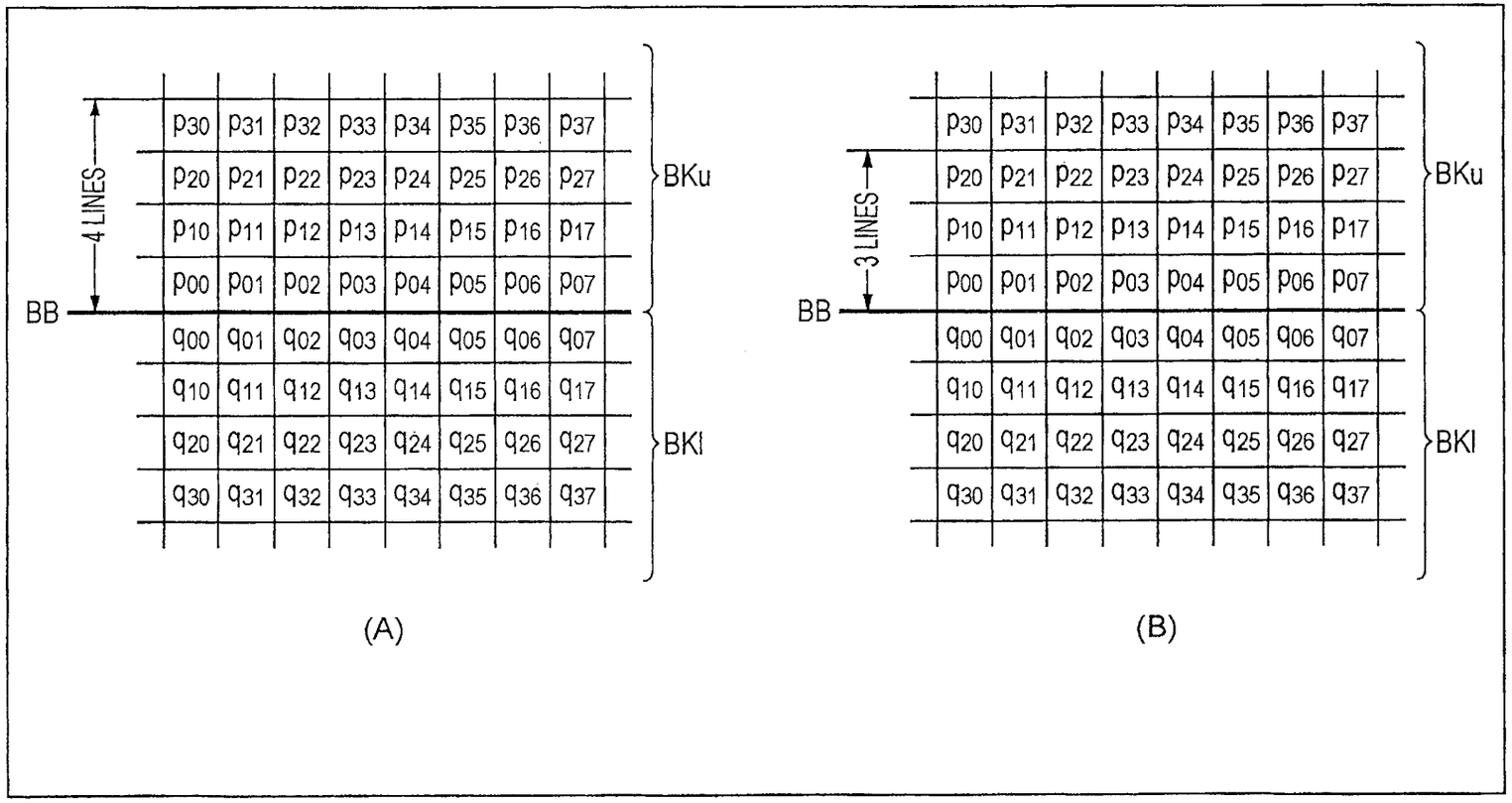

4 LINES
BB
p30 p31 p32 p33 p34 p35 p36 p37
p20 p21 p22 p23 p24 p25 p26 p27
p10 p11 p12 p13 p14 p15 p16 p17
p00 p01 p02 p03 p04 p05 p06 p07
q00 q01 q02 q03 q04 q05 q06 q07
q10 q11 q12 q13 q14 q15 q16 q17
q20 q21 q22 q23 q24 q25 q26 q27
q30 q31 q32 q33 q34 q35 q36 q37
BKu
BKl
(A)
3 LINES
BB
p30 p31 p32 p33 p34 p35 p36 p37
p20 p21 p22 p23 p24 p25 p26 p27
p10 p11 p12 p13 p14 p15 p16 p17
p00 p01 p02 p03 p04 p05 p06 p07
q00 q01 q02 q03 q04 q05 q06 q07
q10 q11 q12 q13 q14 q15 q16 q17
q20 q21 q22 q23 q24 q25 q26 q27
q30 q31 q32 q33 q34 q35 q36 q37
BKu
BKl
(B)

FIG. 14

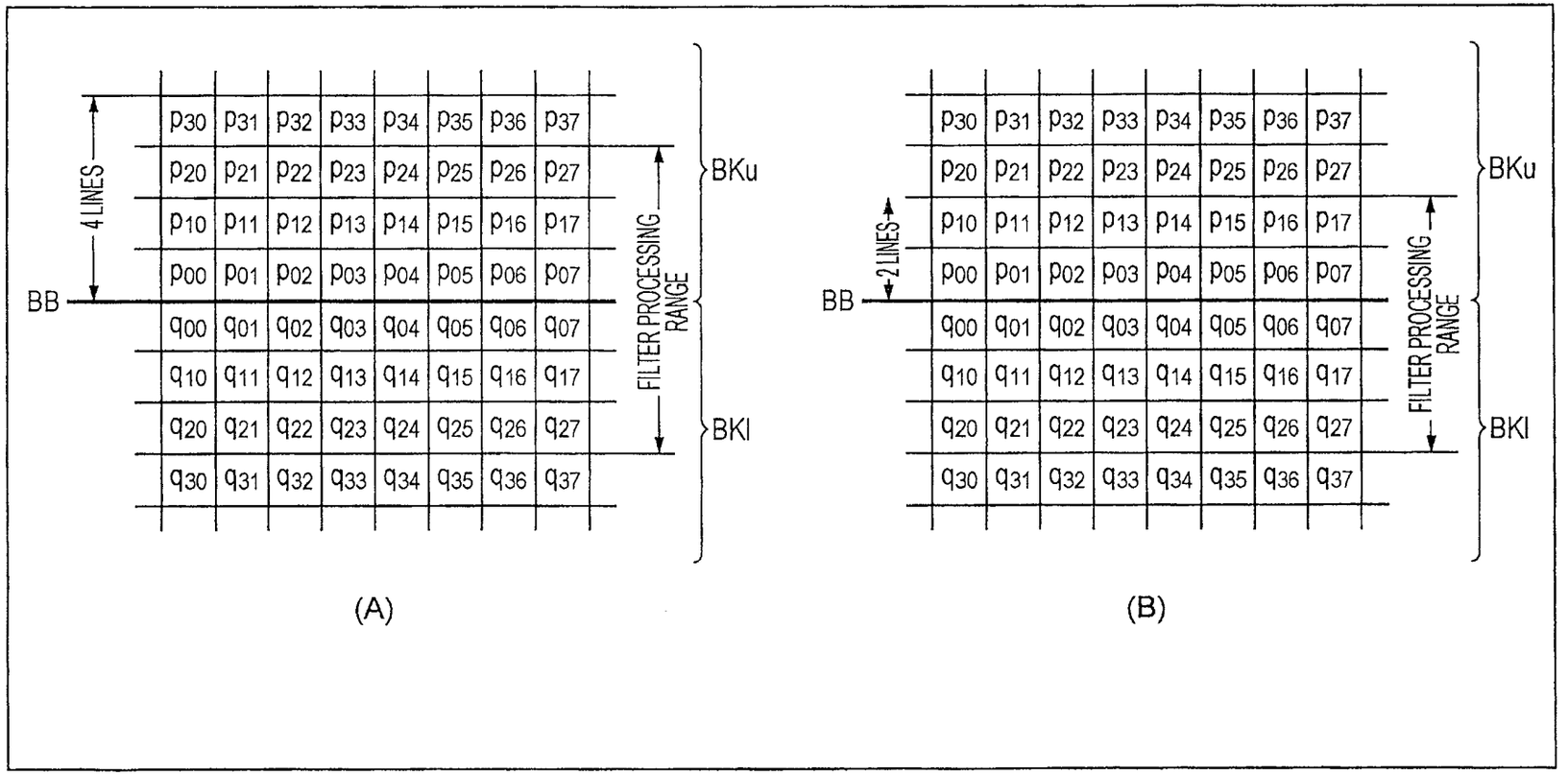

p30 p31 p32 p33 p34 p35 p36 p37
p20 p21 p22 p23 p24 p25 p26 p27
p10 p11 p12 p13 p14 p15 p16 p17
p00 p01 p02 p03 p04 p05 p06 p07
q00 q01 q02 q03 q04 q05 q06 q07
q10 q11 q12 q13 q14 q15 q16 q17
q20 q21 q22 q23 q24 q25 q26 q27
q30 q31 q32 q33 q34 q35 q36 q37
4 LINES
BB
FILTER PROCESSING RANGE
BKu
BKl
(A)
2 LINES
BB
FILTER PROCESSING RANGE
BKu
BKl
(B)

FIG. 15

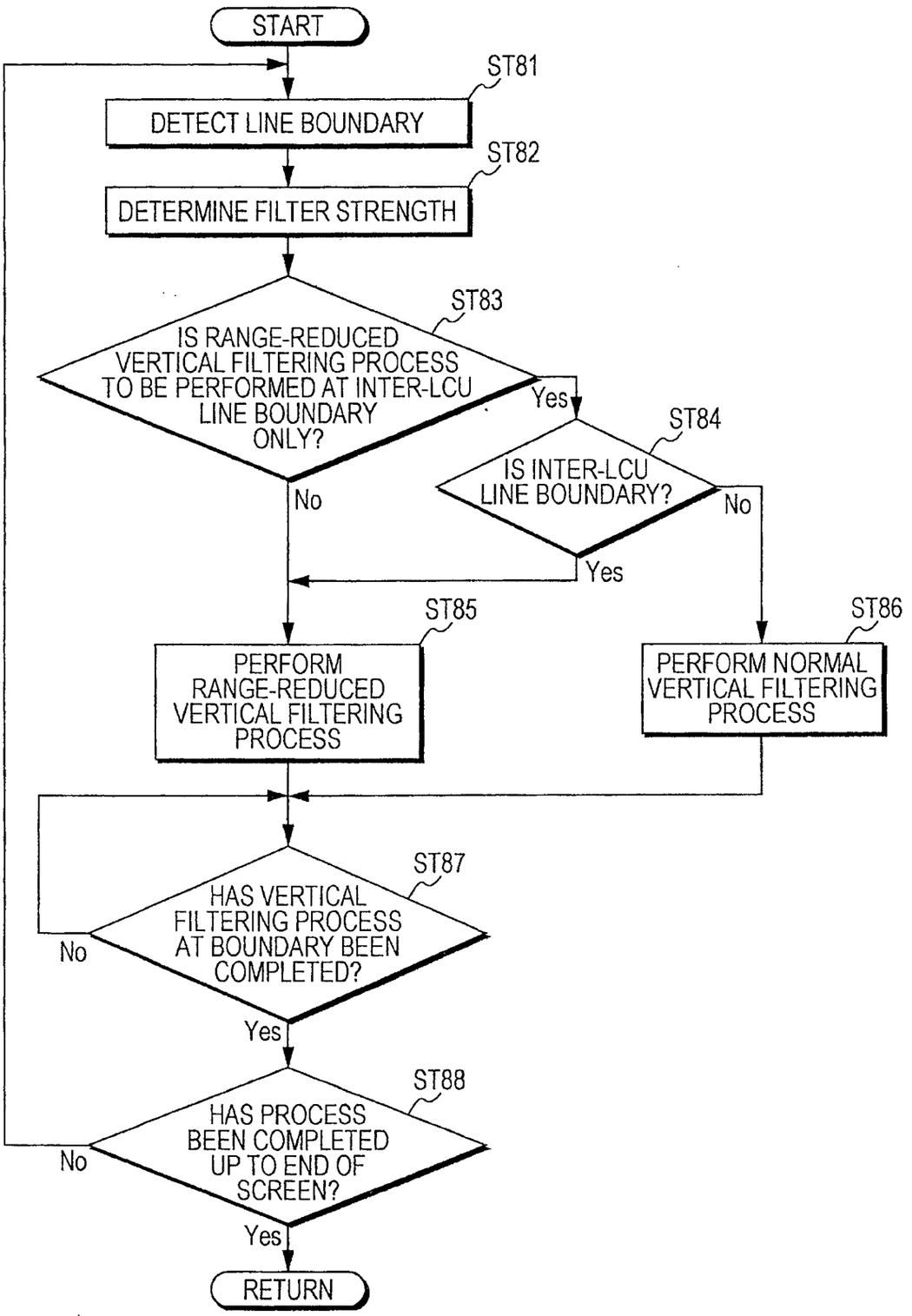

START
ST81
DETECT LINE BOUNDARY
ST82
DETERMINE FILTER STRENGTH
ST83
IS RANGE-REDUCED VERTICAL FILTERING PROCESS TO BE PERFORMED AT INTER-LCU LINE BOUNDARY ONLY?
Yes
No
ST84
IS INTER-LCU LINE BOUNDARY?
No
Yes
ST85
PERFORM RANGE-REDUCED VERTICAL FILTERING PROCESS
ST86
PERFORM NORMAL VERTICAL FILTERING PROCESS
ST87
HAS VERTICAL FILTERING PROCESS AT BOUNDARY BEEN COMPLETED?
No
Yes
ST88
HAS PROCESS BEEN COMPLETED UP TO END OF SCREEN?
No
Yes
RETURN

FIG. 16

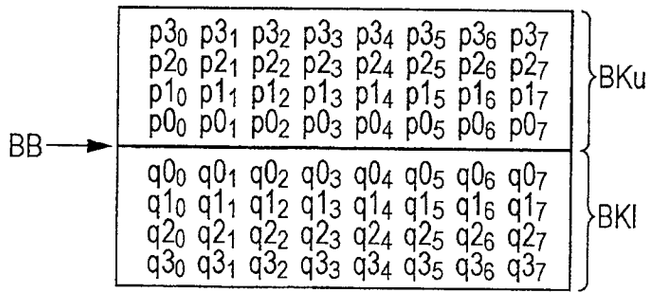

BB
BKu
BKl

FIG. 17

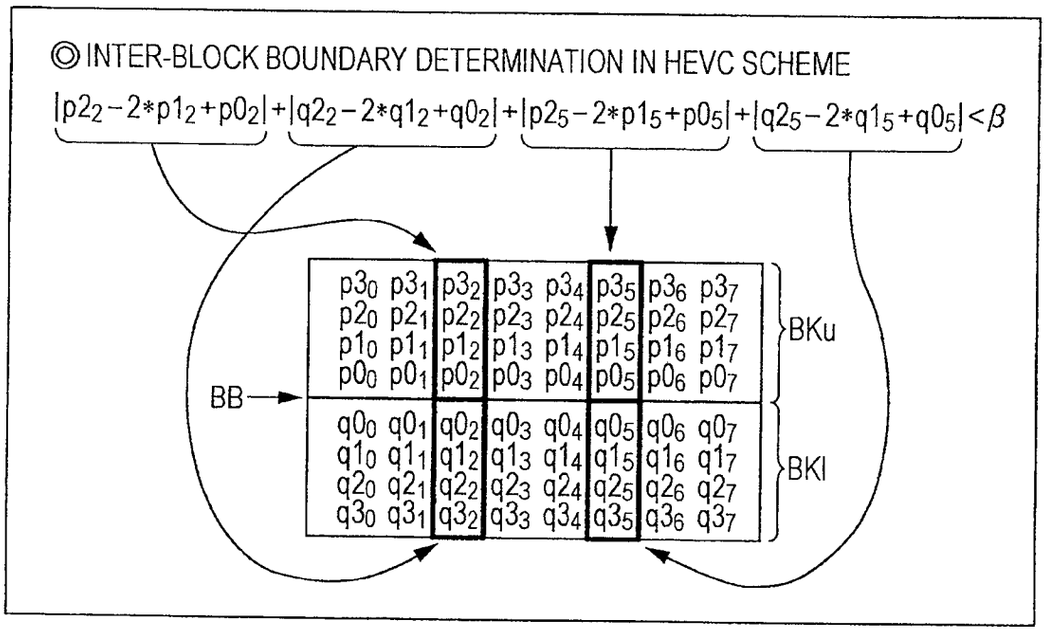

◎INTER-BLOCK BOUNDARY DETERMINATION IN HEVC SCHEME
|p2₂−2*p1₂+p0₂|+|q2₂−2*q1₂+q0₂|+|p2₅−2*p1₅+p0₅|+|q2₅−2*q1₅+q0₅|<β
BB
BKu
BKl

FIG. 18

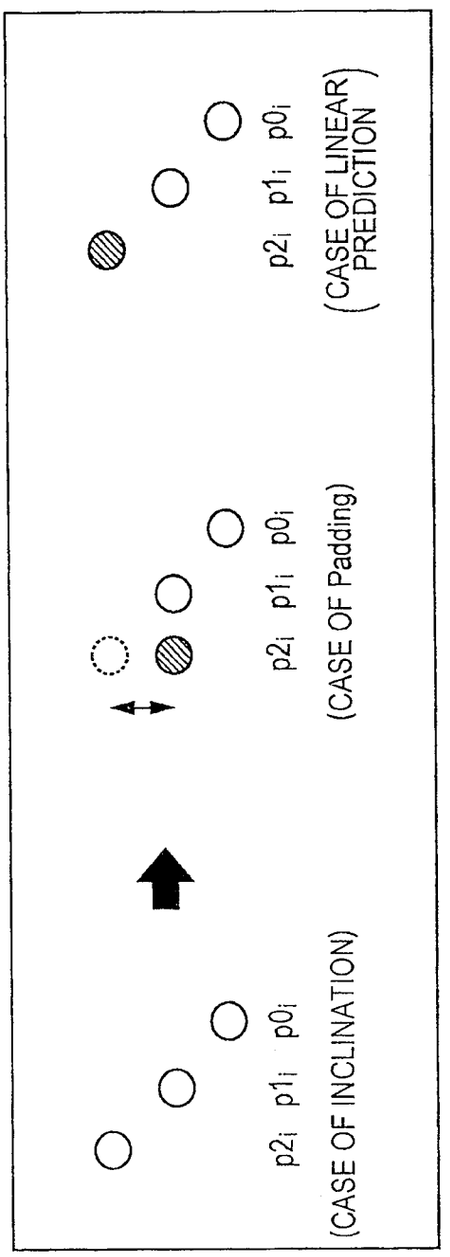

p2i p1i p0i
(CASE OF LINEAR PREDICTION)
p2i p1i p0i
(CASE OF Padding)
p2i p1i p0i
(CASE OF INCLINATION)

FIG. 19

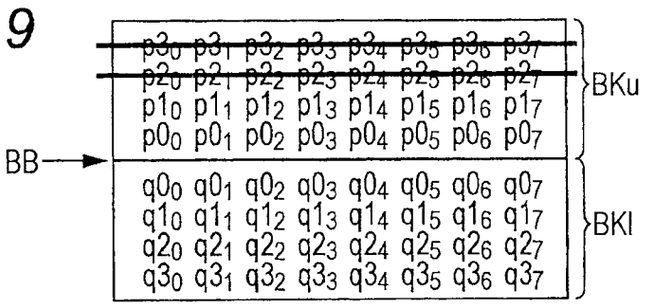

p3₀ p3₁ p3₂ p3₃ p3₄ p3₅ p3₆ p3₇
p2₀ p2₁ p2₂ p2₃ p2₄ p2₅ p2₆ p2₇
p1₀ p1₁ p1₂ p1₃ p1₄ p1₅ p1₆ p1₇
p0₀ p0₁ p0₂ p0₃ p0₄ p0₅ p0₆ p0₇
BKu
BB
q0₀ q0₁ q0₂ q0₃ q0₄ q0₅ q0₆ q0₇
q1₀ q1₁ q1₂ q1₃ q1₄ q1₅ q1₆ q1₇
q2₀ q2₁ q2₂ q2₃ q2₄ q2₅ q2₆ q2₇
q3₀ q3₁ q3₂ q3₃ q3₄ q3₅ q3₆ q3₇
BKl

FIG. 20

	AI_HE				RA_HE				LB_HE				LP_HE			
	22	27	32	37	22	27	32	37	22	27	32	37	22	27	32	37
Class A	16	25	23	29	12	19	19	18								
Class B	9	37	42	50	15	29	25	84	9	55	53	73	12	38	50	82
Class C	29	40	54	109	9	28	34	35	27	15	34	44	5	24	34	49
Class D	7	13	57	54	6	9	40	42	3	8	20	22	4	8	18	20
Class E	15	16	20	28					5	7	15	17	6	11	16	20
Class F	190	190	190	191	178	189	191	190	191	188	189	182	191	188	181	169

FIG. 21

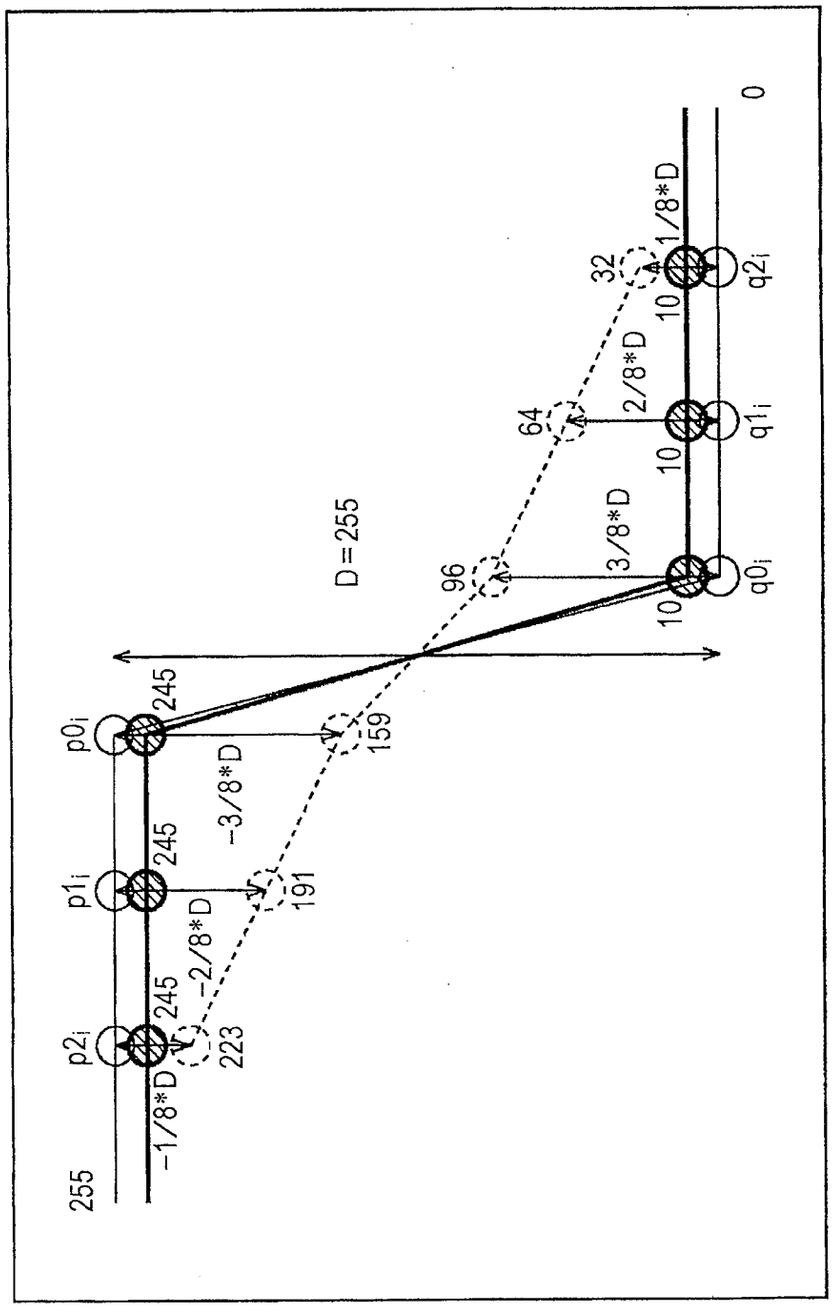

D=255
p0i
p1i
p2i
q0i
q1i
q2i
245
245
245
255
0
159
191
223
96
64
32
10
10
10
-3/8*D
-2/8*D
-1/8*D
3/8*D
2/8*D
1/8*D

FIG. 22

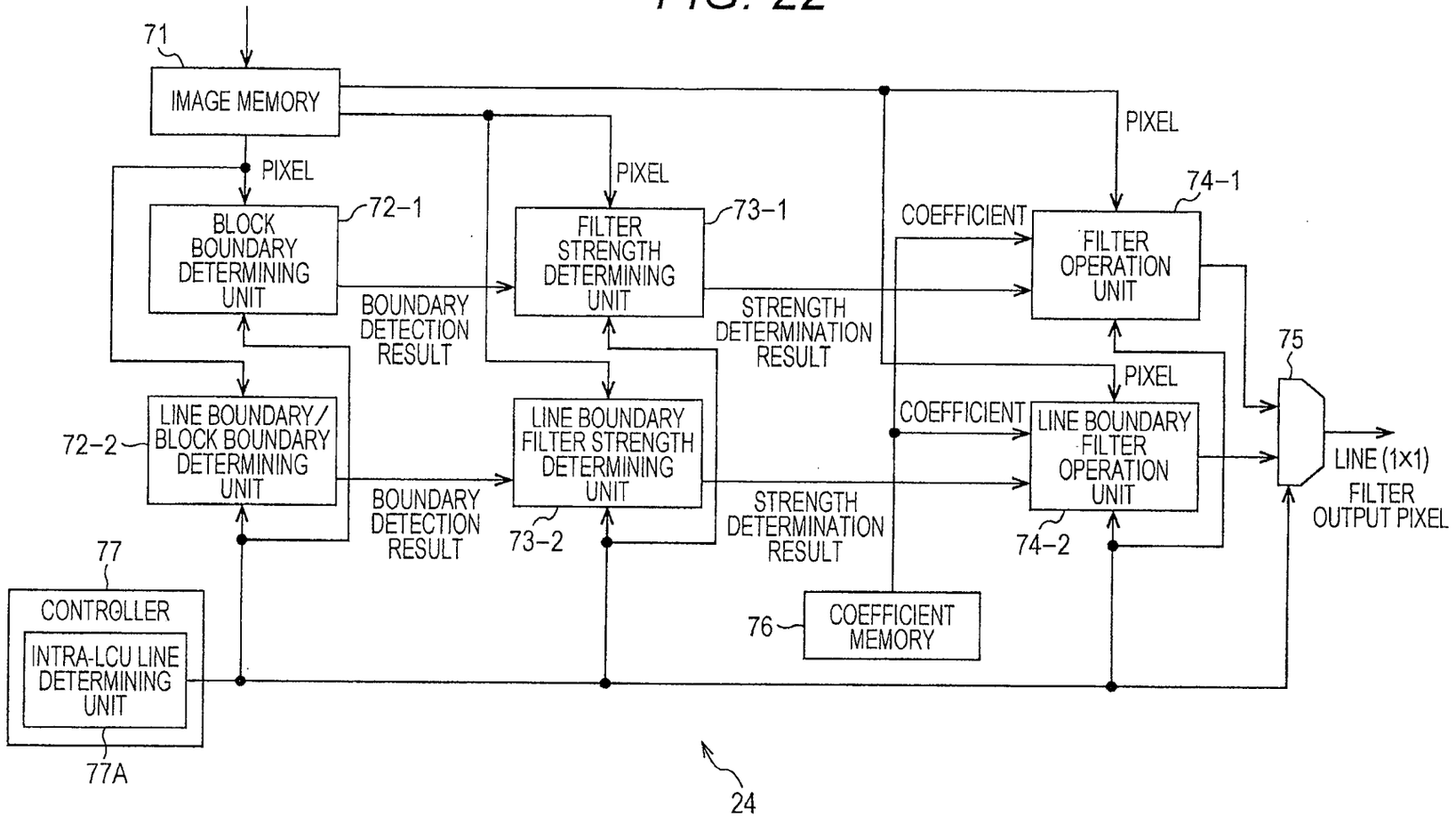

71
IMAGE MEMORY
PIXEL
PIXEL
PIXEL
BLOCK BOUNDARY DETERMINING UNIT
72-1
FILTER STRENGTH DETERMINING UNIT
73-1
COEFFICIENT
FILTER OPERATION UNIT
74-1
BOUNDARY DETECTION RESULT
STRENGTH DETERMINATION RESULT
75
PIXEL
LINE BOUNDARY / BLOCK BOUNDARY DETERMINING UNIT
72-2
LINE BOUNDARY FILTER STRENGTH DETERMINING UNIT
73-2
COEFFICIENT
LINE BOUNDARY FILTER OPERATION UNIT
74-2
BOUNDARY DETECTION RESULT
STRENGTH DETERMINATION RESULT
LINE (1x1) FILTER OUTPUT PIXEL
77
CONTROLLER
INTRA-LCU LINE DETERMINING UNIT
77A
76
COEFFICIENT MEMORY
24

FIG. 23

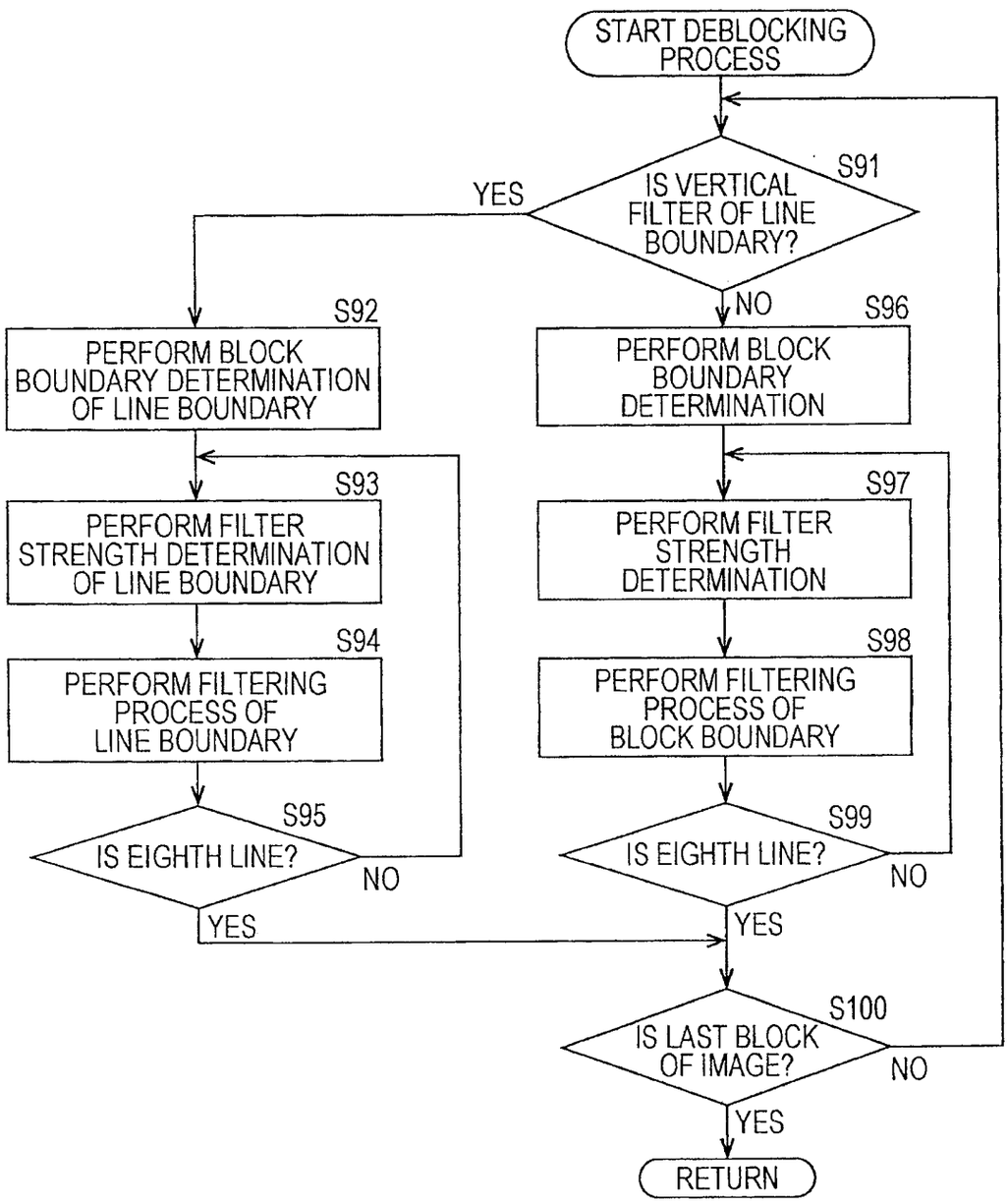

START DEBLOCKING PROCESS
S91
IS VERTICAL FILTER OF LINE BOUNDARY?
YES
NO
S92
PERFORM BLOCK BOUNDARY DETERMINATION OF LINE BOUNDARY
S93
PERFORM FILTER STRENGTH DETERMINATION OF LINE BOUNDARY
S94
PERFORM FILTERING PROCESS OF LINE BOUNDARY
S95
IS EIGHTH LINE?
NO
YES
S96
PERFORM BLOCK BOUNDARY DETERMINATION
S97
PERFORM FILTER STRENGTH DETERMINATION
S98
PERFORM FILTERING PROCESS OF BLOCK BOUNDARY
S99
IS EIGHTH LINE?
NO
YES
S100
IS LAST BLOCK OF IMAGE?
NO
YES
RETURN

FIG. 24

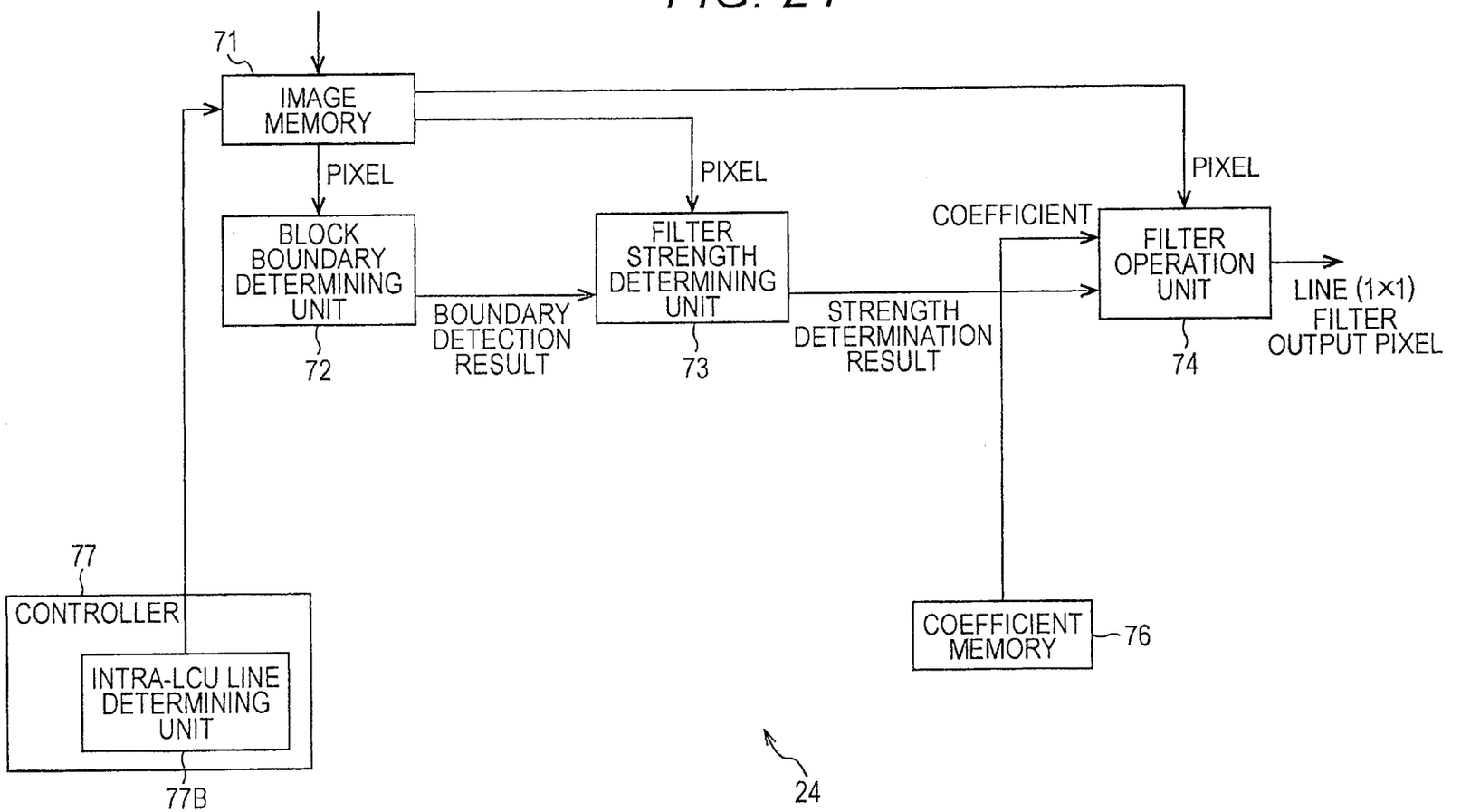

71
IMAGE MEMORY
PIXEL
PIXEL
PIXEL
BLOCK BOUNDARY DETERMINING UNIT
72
BOUNDARY DETECTION RESULT
FILTER STRENGTH DETERMINING UNIT
73
STRENGTH DETERMINATION RESULT
COEFFICIENT
FILTER OPERATION UNIT
74
LINE (1×1) FILTER OUTPUT PIXEL
77
CONTROLLER
INTRA-LCU LINE DETERMINING UNIT
77B
COEFFICIENT MEMORY
76
24

FIG. 25

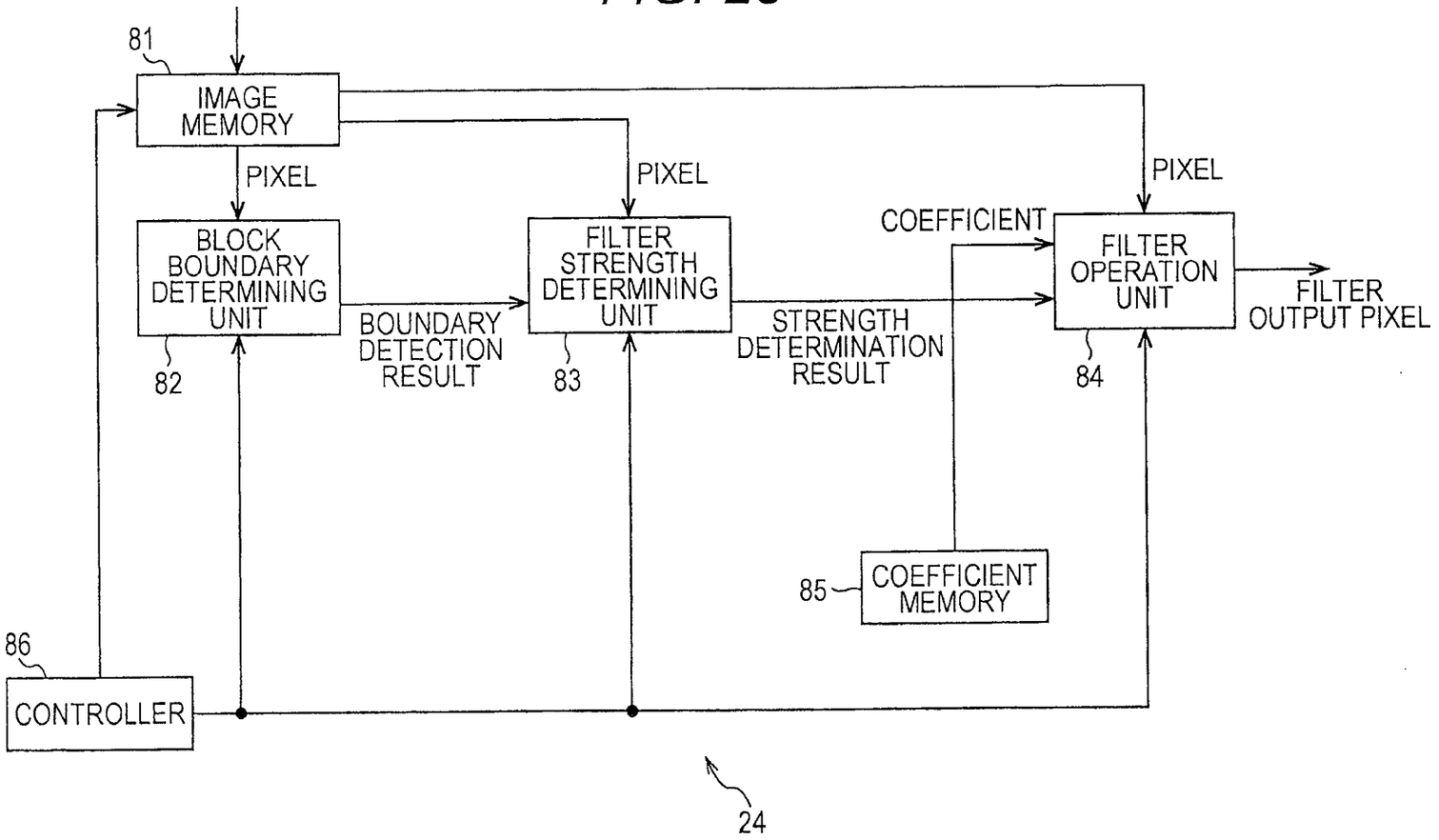

81
IMAGE MEMORY
PIXEL
PIXEL
PIXEL
BLOCK BOUNDARY DETERMINING UNIT
82
BOUNDARY DETECTION RESULT
FILTER STRENGTH DETERMINING UNIT
83
STRENGTH DETERMINATION RESULT
COEFFICIENT
FILTER OPERATION UNIT
84
FILTER OUTPUT PIXEL
85
COEFFICIENT MEMORY
86
CONTROLLER
24

FIG. 26

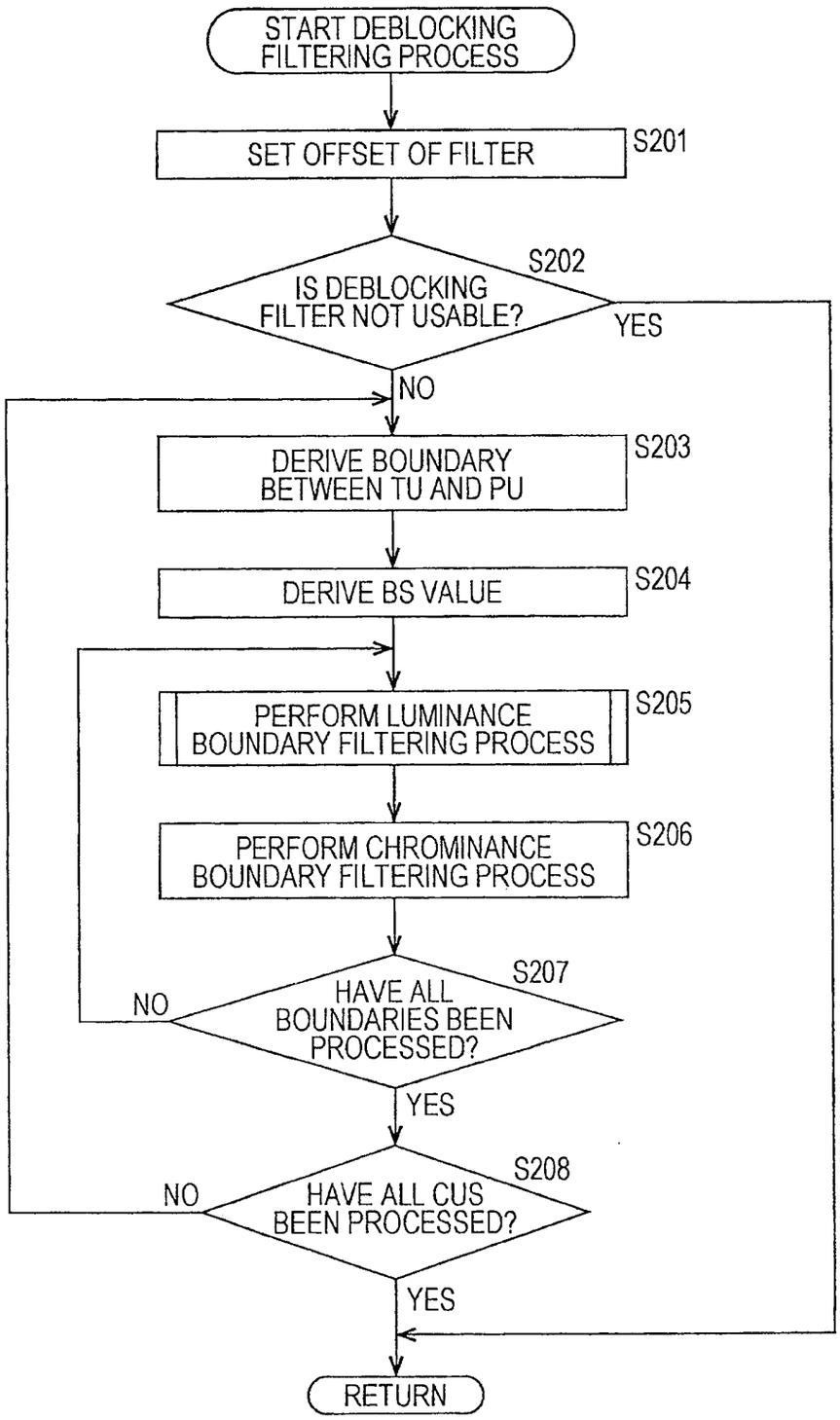

START DEBLOCKING FILTERING PROCESS
SET OFFSET OF FILTER
S201
IS DEBLOCKING FILTER NOT USABLE?
S202
YES
NO
DERIVE BOUNDARY BETWEEN TU AND PU
S203
DERIVE BS VALUE
S204
PERFORM LUMINANCE BOUNDARY FILTERING PROCESS
S205
PERFORM CHROMINANCE BOUNDARY FILTERING PROCESS
S206
HAVE ALL BOUNDARIES BEEN PROCESSED?
S207
NO
YES
HAVE ALL CUS BEEN PROCESSED?
S208
NO
YES
RETURN

FIG. 27

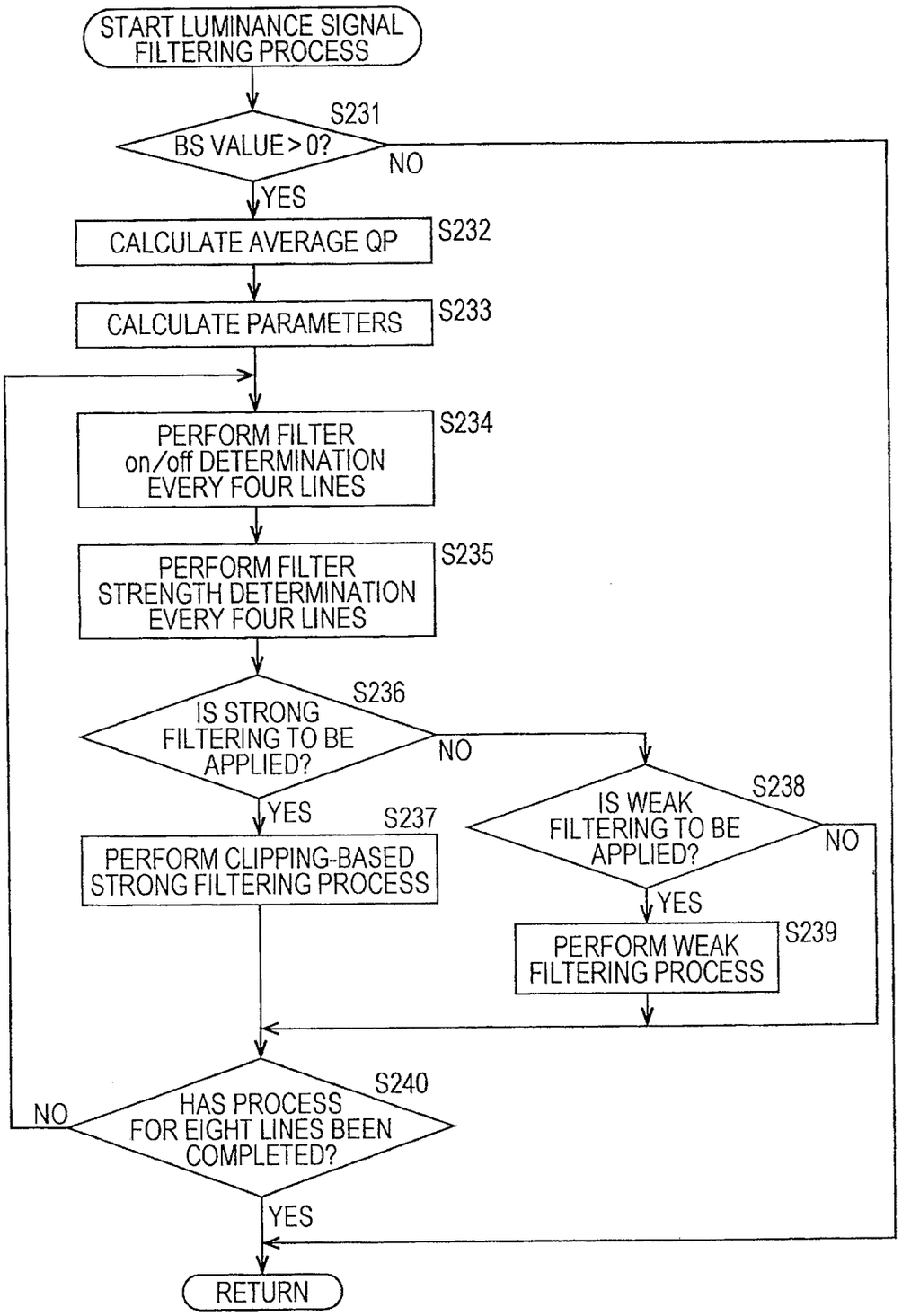

START LUMINANCE SIGNAL FILTERING PROCESS
S231
BS VALUE > 0?
NO
YES
CALCULATE AVERAGE QP
S232
CALCULATE PARAMETERS
S233
PERFORM FILTER on/off DETERMINATION EVERY FOUR LINES
S234
PERFORM FILTER STRENGTH DETERMINATION EVERY FOUR LINES
S235
S236
IS STRONG FILTERING TO BE APPLIED?
NO
YES
S237
PERFORM CLIPPING-BASED STRONG FILTERING PROCESS
S238
IS WEAK FILTERING TO BE APPLIED?
NO
YES
S239
PERFORM WEAK FILTERING PROCESS
S240
HAS PROCESS FOR EIGHT LINES BEEN COMPLETED?
NO
YES
RETURN

FIG. 28

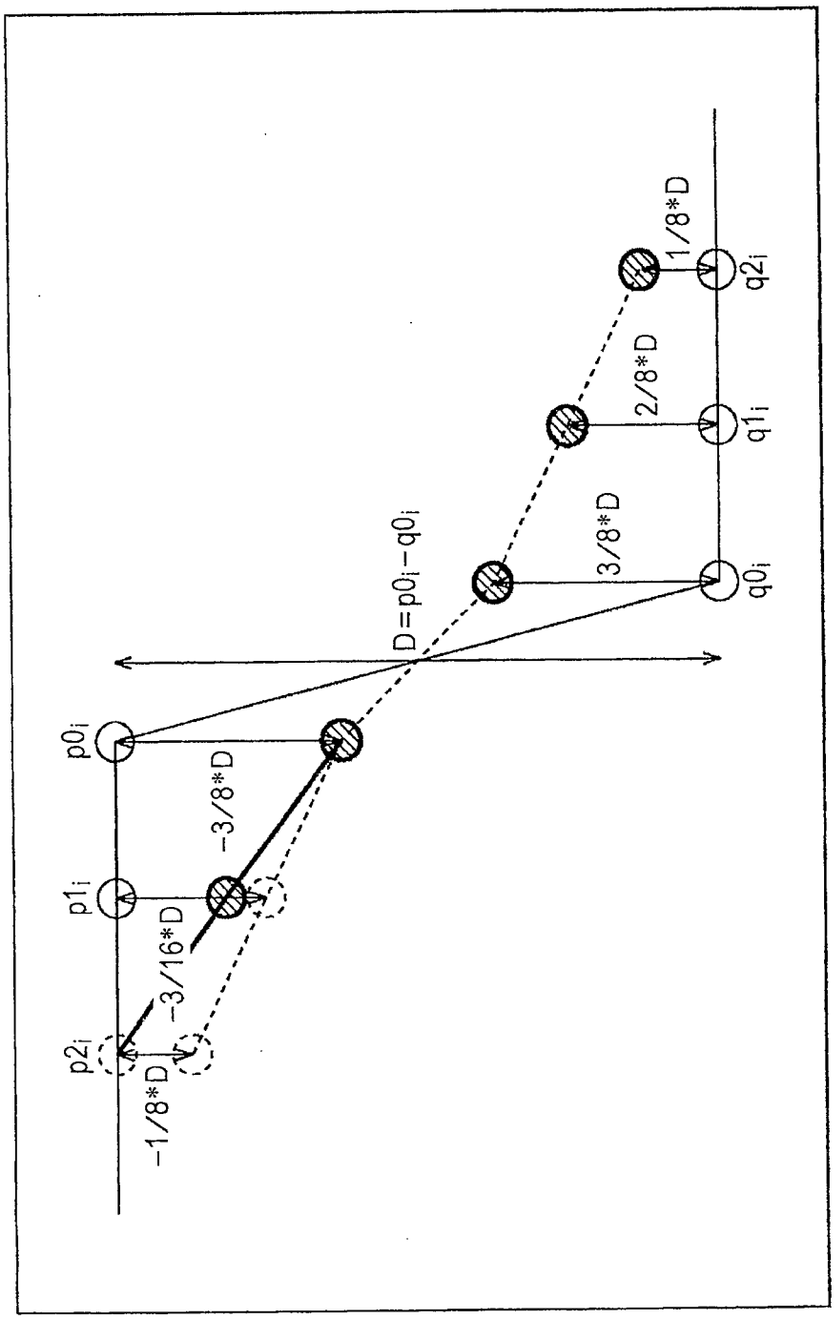


FIG. 29

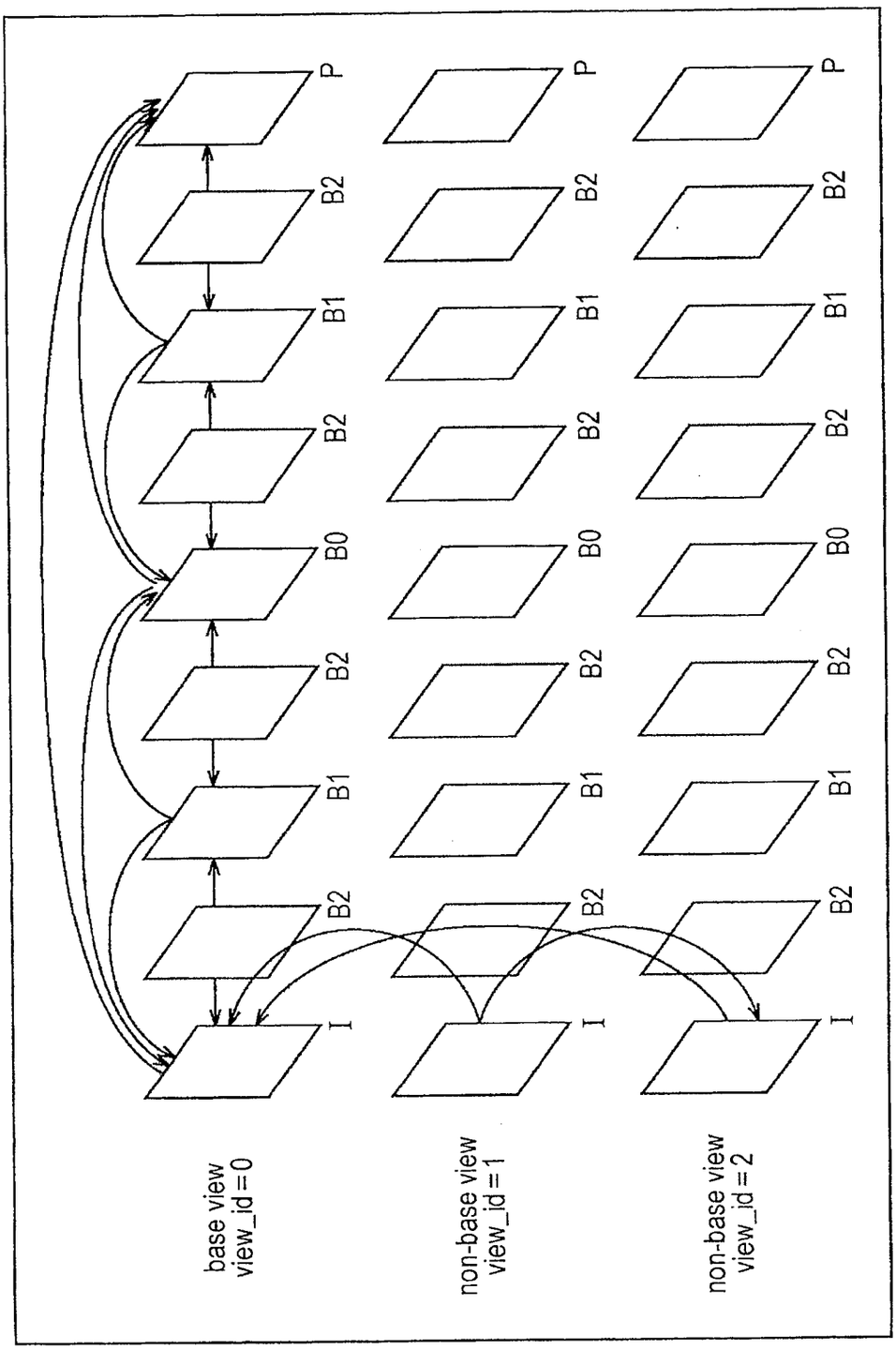

P
B2
B1
B2
B0
B2
B1
B2
I
base view
view_id = 0
non-base view
view_id = 1
non-base view
view_id = 2

FIG. 30

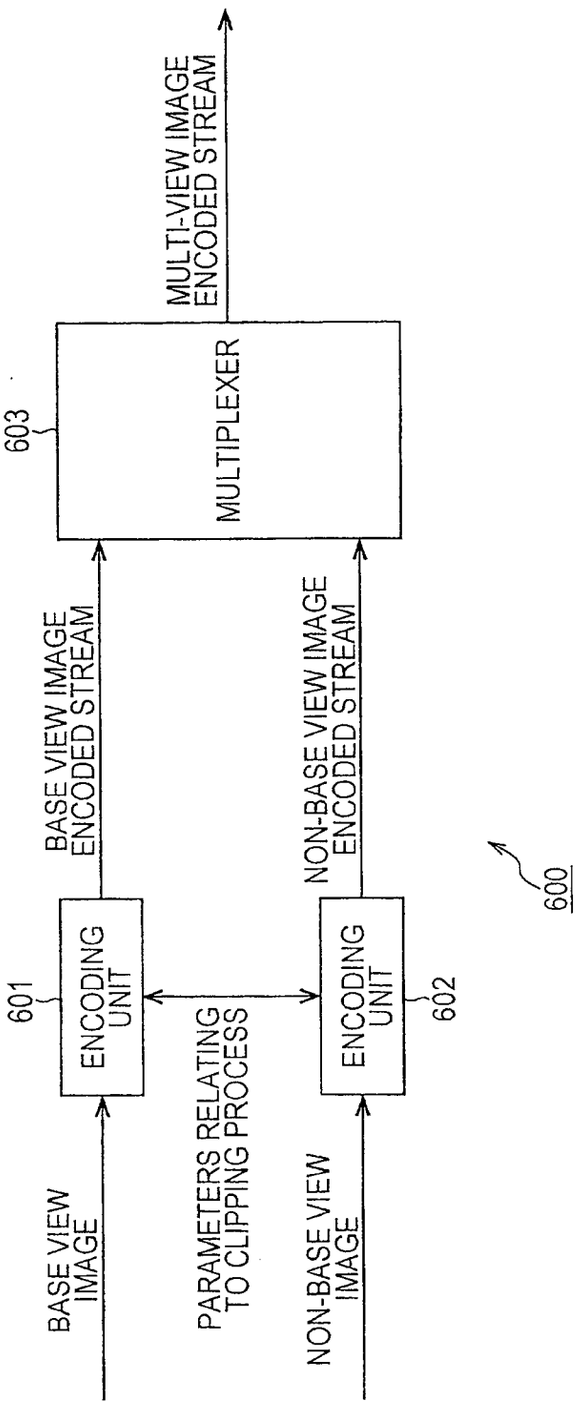

MULTI-VIEW IMAGE ENCODED STREAM
603
MULTIPLEXER
BASE VIEW IMAGE ENCODED STREAM
NON-BASE VIEW IMAGE ENCODED STREAM
600
601
ENCODING UNIT
602
ENCODING UNIT
BASE VIEW IMAGE
PARAMETERS RELATING TO CLIPPING PROCESS
NON-BASE VIEW IMAGE

FIG. 31

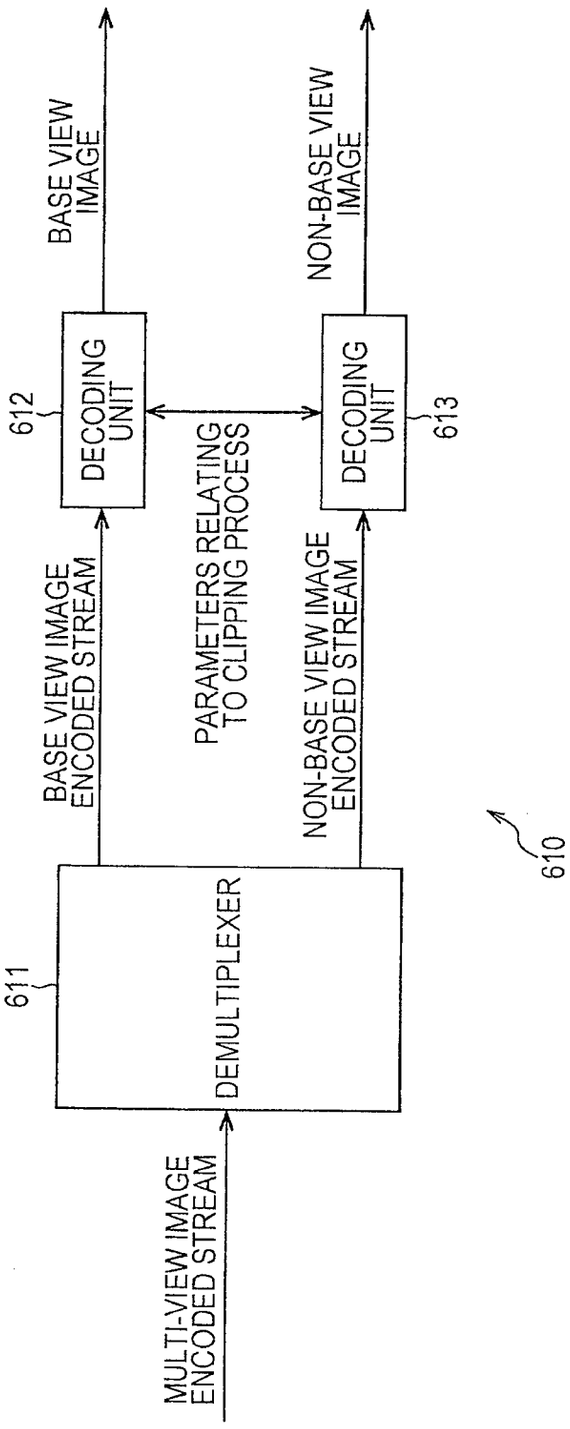

MULTI-VIEW IMAGE ENCODED STREAM
611
DEMULTIPLEXER
BASE VIEW IMAGE ENCODED STREAM
NON-BASE VIEW IMAGE ENCODED STREAM
PARAMETERS RELATING TO CLIPPING PROCESS
612
DECODING UNIT
613
DECODING UNIT
BASE VIEW IMAGE
NON-BASE VIEW IMAGE
610

FIG. 32

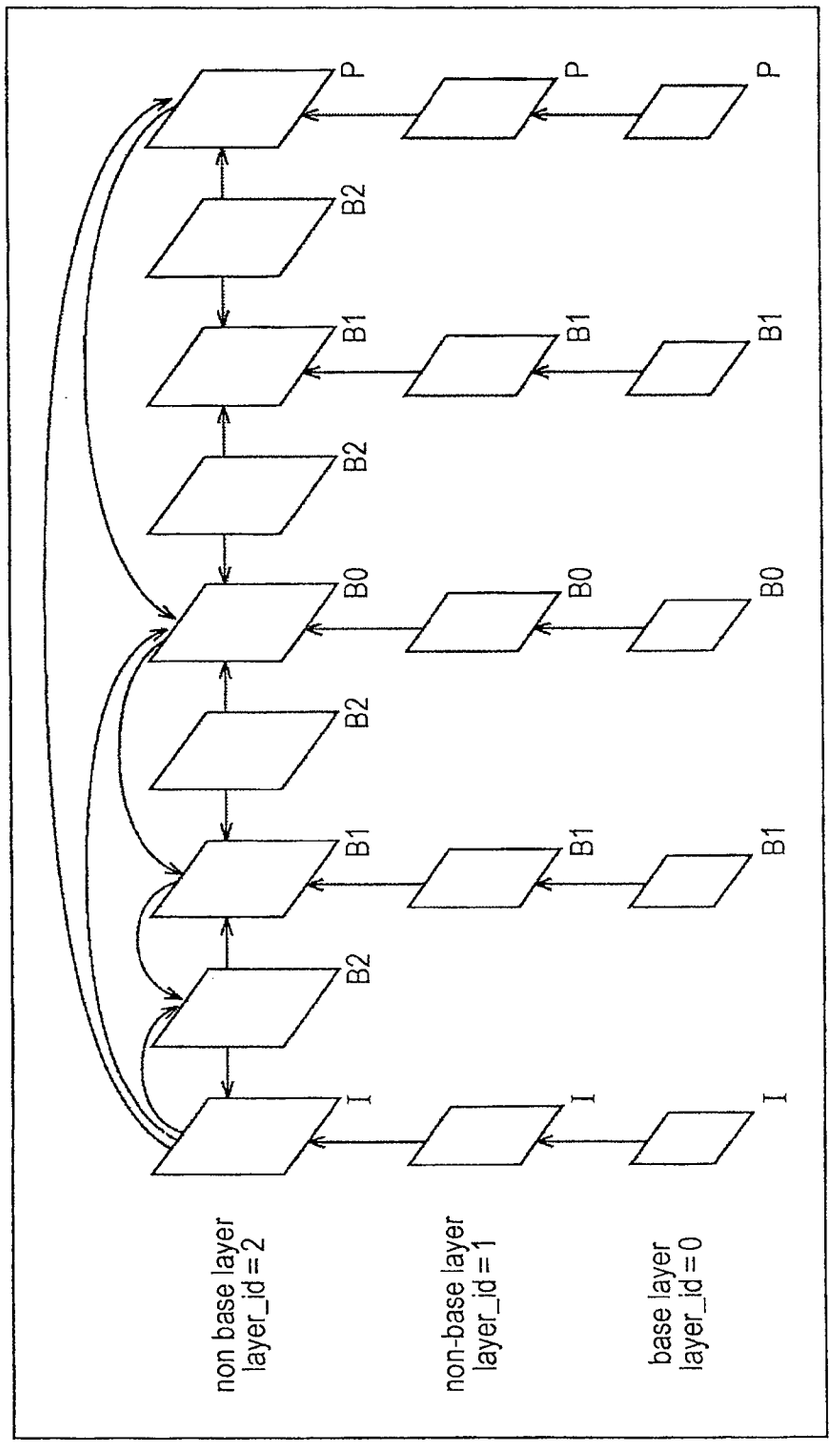

P
P
P
B2
B1
B1
B1
B2
B0
B0
B0
B2
B1
B1
B1
B2
I
I
I
non base layer
layer_id = 2
non-base layer
layer_id = 1
base layer
layer_id = 0

FIG. 33

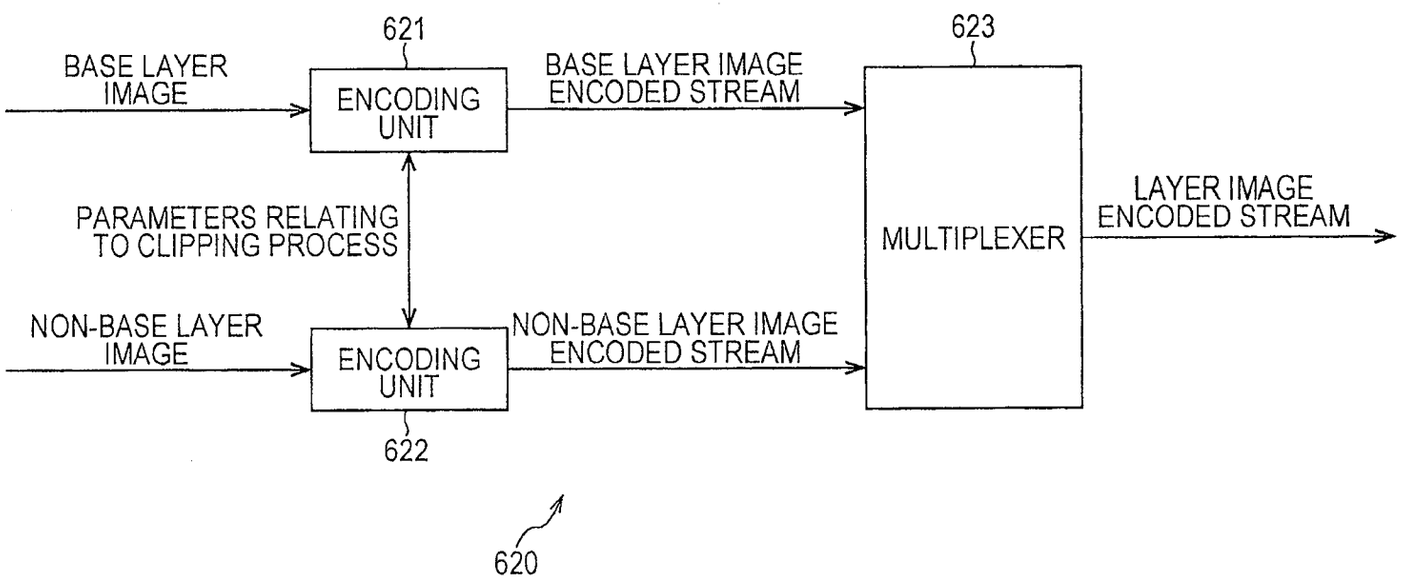

621
623
BASE LAYER IMAGE
ENCODING UNIT
BASE LAYER IMAGE ENCODED STREAM
PARAMETERS RELATING TO CLIPPING PROCESS
MULTIPLEXER
LAYER IMAGE ENCODED STREAM
NON-BASE LAYER IMAGE
ENCODING UNIT
NON-BASE LAYER IMAGE ENCODED STREAM
622
620

FIG. 34

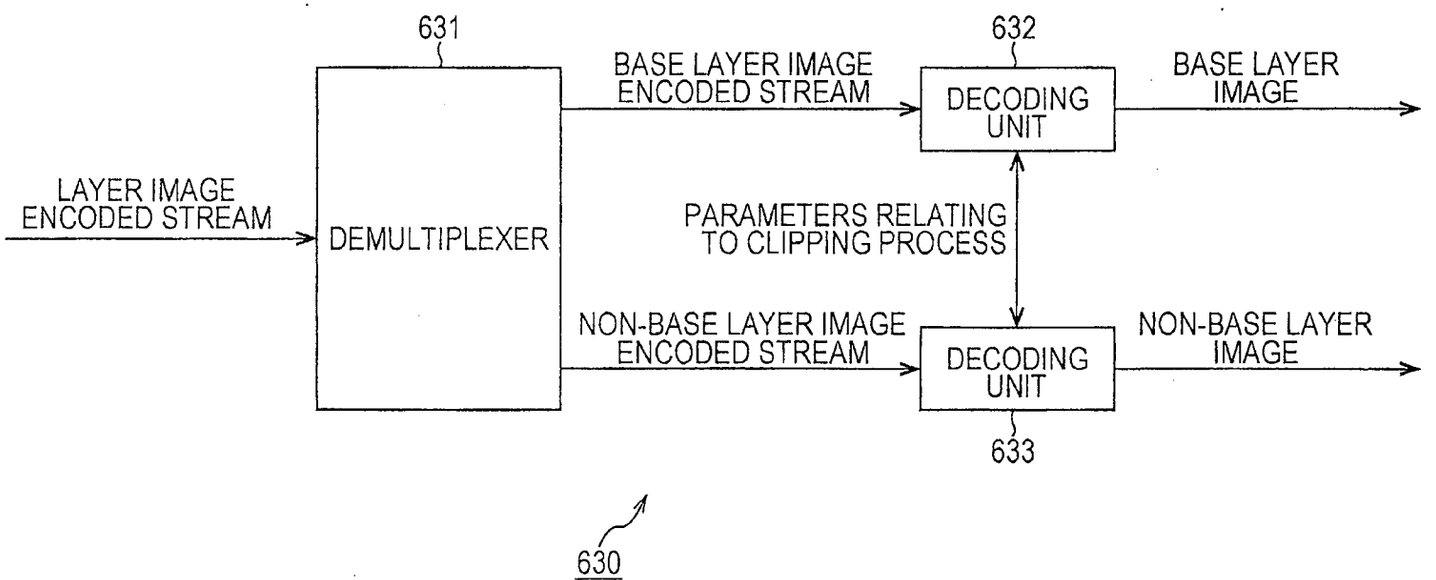

631
632
BASE LAYER IMAGE ENCODED STREAM
DECODING UNIT
BASE LAYER IMAGE
LAYER IMAGE ENCODED STREAM
DEMULTIPLEXER
PARAMETERS RELATING TO CLIPPING PROCESS
NON-BASE LAYER IMAGE ENCODED STREAM
DECODING UNIT
NON-BASE LAYER IMAGE
633
630

FIG. 35

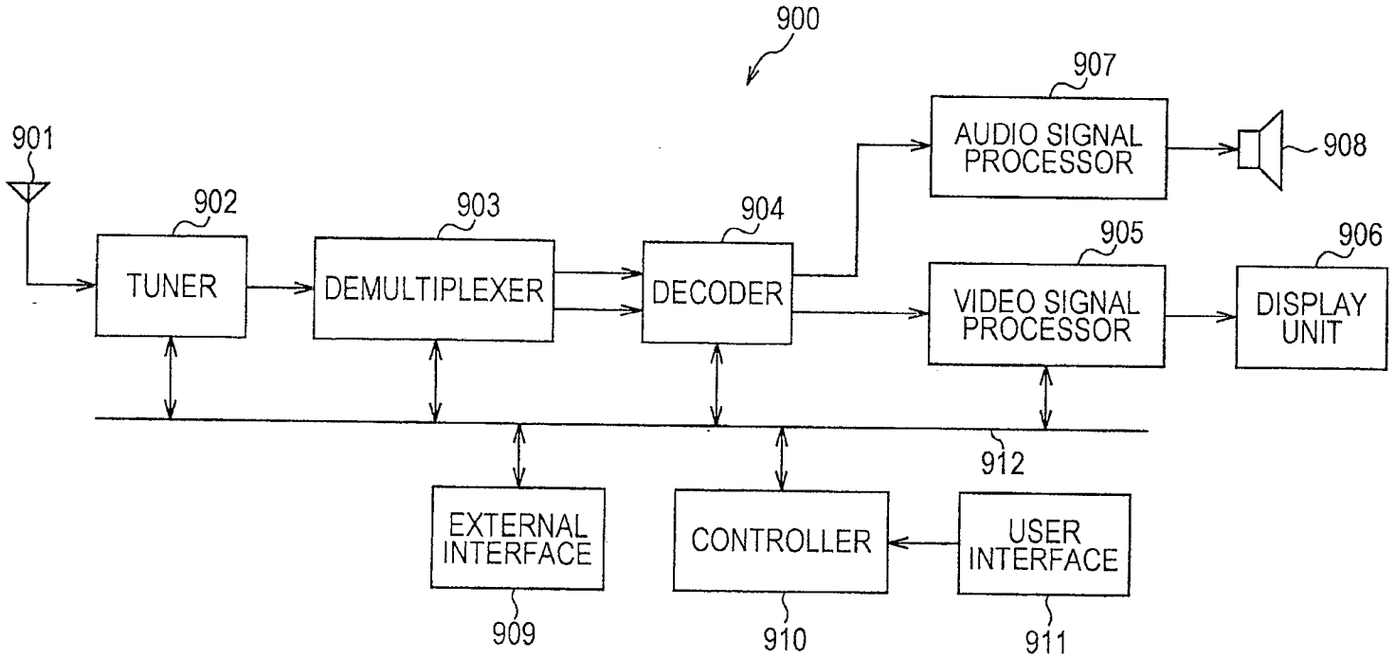

900
901
902
TUNER
903
DEMULTIPLEXER
904
DECODER
907
AUDIO SIGNAL PROCESSOR
908
905
VIDEO SIGNAL PROCESSOR
906
DISPLAY UNIT
912
EXTERNAL INTERFACE
909
CONTROLLER
910
USER INTERFACE
911

FIG. 36

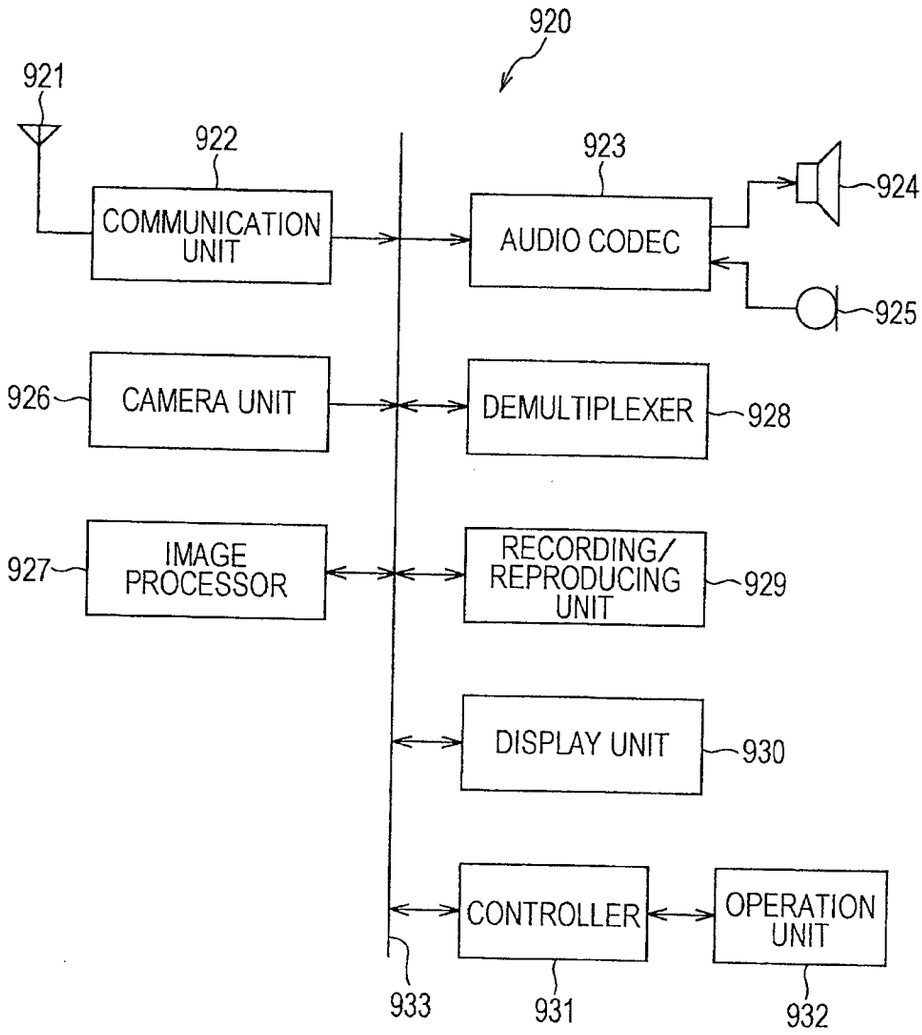

920
921
922
COMMUNICATION UNIT
923
AUDIO CODEC
924
925
926
CAMERA UNIT
DEMULTIPLEXER
928
927
IMAGE PROCESSOR
RECORDING/ REPRODUCING UNIT
929
DISPLAY UNIT
930
CONTROLLER
OPERATION UNIT
933
931
932

FIG. 37

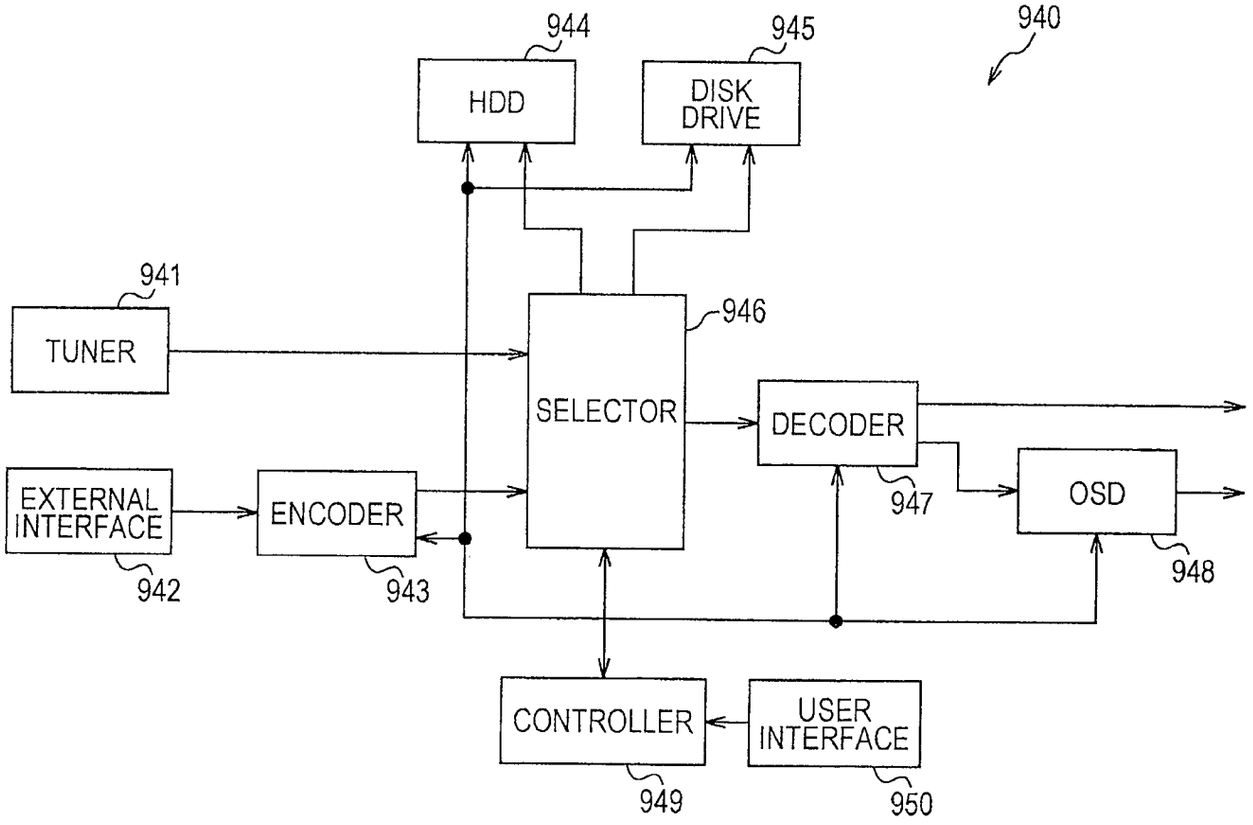
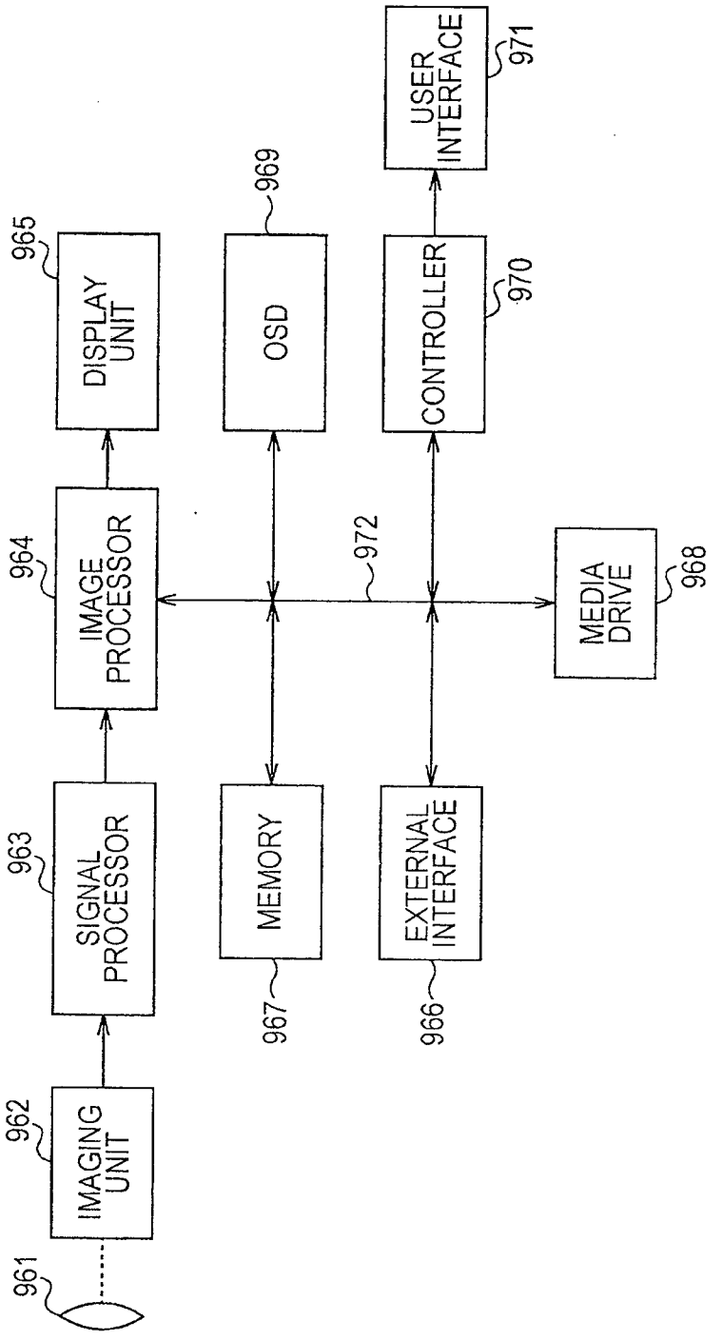

940
944
HDD
945
DISK DRIVE
941
TUNER
946
SELECTOR
DECODER
947
OSD
948
EXTERNAL INTERFACE
942
ENCODER
943
CONTROLLER
949
USER INTERFACE
950

FIG. 38


960
961
962
IMAGING UNIT
963
SIGNAL PROCESSOR
964
IMAGE PROCESSOR
965
DISPLAY UNIT
966
EXTERNAL INTERFACE
967
MEMORY
968
MEDIA DRIVE
969
OSD
970
CONTROLLER
971
USER INTERFACE
972

DESCRIPCIÓN

DISPOSITIVO DE PROCESAMIENTO DE IMAGEN Y MÉTODO DE PROCESAMIENTO DE IMAGEN

CAMPO TECNICO

[0001]

Esta técnica se refiere a un dispositivo de procesamiento de imagen BKu método de procesamiento de imagen. Específicamente, esta técnica permite un proceso de filtración de desbloqueo para aplicar apropiadamente la filtración.

ANTECEDENTES DEL ARTE

[0002]

En los últimos años, se han generado los dispositivos, que manejan información de imagen como datos digitales, que, en tal caso ayudan a transmitir y almacenar información con una alta eficiencia, y que se ajustan a un esquema, tal como MPEG2 (International Organization for Standardization and International Electrotechnical Cornmission (ISO/IEC) 13818-2), para la compresión de información de imagen usando transformación ortogonal, tal como la transformación discreta del coseno, y usando compensación de movimiento mediante la utilización de la redundancia que es única para la información de imagen tanto en la distribución de información en las estaciones de radiodifusión e información de recepción en los hogares ordinarios. Además, los esquemas llamados H.264 y MPEG4 Part10 (Advanced Video Coding (AVC) que requieren una mayor cantidad de operaciones para la codificación y decodificación, pero puede realizar una eficacia de codificación más alta que MPEG2 o los similares que también se han utilizado. Además, hoy en día, la estandarización trabaja para la codificación de vídeo de alta eficiencia (HEVC) que es un esquema de codificación de imagen de próxima generación están en curso de tal manera que se pueden realizar eficientemente la compresión, distribución, o similares de imagen de alta resolución de 4000 x 2000 píxeles que es cuatro veces la de las imagen de alta visión.

[0003]

En la estandarización que funciona para una alta eficiencia de codificación de vídeo (HEVC), que es un esquema de codificación de imagen de próxima generación, JCTVC – A119 (ver Documento de No Patente 1 más adelante) se propone aplicar un filtro de desbloqueo a cada bloque que tiene el tamaño de 8 x 8 píxeles o mayor. En el método propuesto en JCTVC - A119, mediante el aumento de una unidad de tamaño mínimo de bloque al que se va a aplicar

un filtro de desbloqueo, es posible ejecutar un proceso de filtración en paralelo en una pluralidad de límites de bloque en la misma dirección dentro de un macrobloque.

LISTA DE CITAS

DOCUMENTO DE NO PATENTE

[0004]

Documento de No Patente 1: K. Ugur (Nokia), K. R. Andersson (LM Ericsson), A. Fuldseth (Tandberg Telecom), "JCTVC - A119: vídeo de codificación propuesta por la tecnología de Tandberg, Nokia y Ericsson", Documentos de la primera reunión del Equipo de Colaboración Conjunta sobre Video Coding (JCT- VC), Dresden, Alemania, Abril 15-23 2010

RESUMEN DE LA INVENCIÓN

PROBLEMAS A SER RESUELTOS POR LA INVENCIÓN

[0005]

Sin embargo, en el proceso de filtración de desbloqueo convencional, la filtración no se realiza apropiadamente.

[0006]

Por lo tanto, un objeto de esta técnica es permitir un proceso de filtración de desbloqueo para aplicar la filtración apropiadamente.

SOLUCIONES A LOS PROBLEMAS

[0007]

Un dispositivo de procesamiento de imagen de acuerdo con un primer aspecto de esta técnica incluye: una unidad de decodificación que decodifica una corriente codificada, codificada en unidades que tiene cada una, una estructura de capa para generar una imagen; una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite del bloque que es un límite entre un bloque de la imagen generada por la unidad de decodificación BKu bloque adyacente al bloque, y un controlador que controla la unidad de filtración de manera que, cuando se aplica una filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de

REIVINDICACIONES

1. Un dispositivo de procesamiento de imagen que comprende:

una unidad de decodificación que decodifica una corriente codificada, codificada en unidades que tienen una estructura de capa para generar una imagen;

una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es in límite entre un bloque de la imagen generada por el unidad de decodificación y un bloque adyacente, adyacente al bloque; y

un controlador que controla la unidad de filtración de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

2. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 1, en el que,

el controlador aplica el proceso de recorte a los valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

3. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 2, en el que

el controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones:

[Fórmula Matemática 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0,7 y pv es un valor de recorte.

4. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 3 en el que,

el controlador establece un valor que es un múltiplo entero de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se aplica el proceso de recorte.

5. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 4 en el que,

el controlador establece un valor que es el doble de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se realiza el proceso de recorte.

6. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 1, que incluye además:

Una unidad que determina la fuerza del filtro, que determina la fuerza del filtro de desbloqueo aplicado al límite de bloque, en donde

la unidad de filtración aplica el filtro de desbloqueo al límite de bloque de acuerdo con la fuerza determinada por la unidad que determina la fuerza del filtro, y

el controlador controla la unidad de filtración de manera que, cuando la unidad que determina la fuerza filtración, determina que se va a aplicar filtración fuerte, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

7. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 6 en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando una pluralidad de líneas como unidades de procesamiento.

8. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 7, en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando cuatro líneas como las unidades de procesamiento.

9. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 7, que incluye además:

la unidad que determina la necesidad de filtración, que determina si el filtro de desbloqueo se va a aplicar al límite de bloque utilizando una pluralidad de líneas como unidades de procesamiento, en el que la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que se va a aplicar el filtro de desbloqueo.

10. Un método de procesamiento de imagen para permitir que un dispositivo de procesamiento de imagen ejecute:

La decodificación de una corriente codificada, codificada en unidades que tienen una estructura de capa para generar una imagen;

Aplicar un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de la imagen generada y un bloque adyacente, adyacente al bloque; y

Controlar de tal manera que cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada.

11. un dispositivo de procesamiento de imagen que incluye:

Una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de una imagen decodificada localmente cuando una imagen se codifica y un bloque adyacente, adyacente al bloque;

Un controlador que controla la unidad de filtración de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente; y

Una unidad de codificación que codifica la imagen en unidades que tienen una estructura de capa utilizando una imagen a la que se aplica el filtro de desbloqueo.

12. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 1), en el que,

El controlador aplica el proceso de recorte a valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

13. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 12, en el que,

El controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones

[Fórmula Matemática 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0,7 y pv es un valor de recorte.

C1

14. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 13 en el que,

el controlador establece un valor que es múltiplo entero de los parámetros del filtro de desbloqueo como el valor de recorte utilizado cuando se realiza el proceso de recorte.

15. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 14 en el que,

el controlador establece un valor que es el doble de los parámetros del filtro de desbloqueo como el valor de recorte usado cuando se realiza el proceso de recorte.

16. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 11 que incluye además:

Una unidad que determina la fuerza del filtro, que determina la fuerza del filtro de desbloqueo aplicada al límite de bloque donde la unidad de filtración aplica el filtro de desbloqueo al límite de bloque de acuerdo con la fuerza determinada por la unidad que determina la fuerza del filtro; y

el controlador controla la unidad de filtración de tal manera que,

cuando la unidad que determina la fuerza del filtro determina que se va a aplicar filtración fuerte, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por el unidad de decodificación.

17. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 16 en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo utilizando una pluralidad de líneas como unidades de procesamiento.

18. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 17 en el que,

La unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando cuatro líneas como las unidades de procesamiento.

19. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 17 que incluye además:

una unidad que determina la necesidad de filtración que determina si se va a aplicar el filtro de desbloqueo al límite de bloque utilizando una pluralidad de líneas como unidades de procesamiento, en donde

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que se va a aplicar el filtro de desbloqueo.

20. Un método de procesamiento de imagen para permitir que un dispositivo de procesamiento de imagen ejecute:

La aplicación de un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de una imagen decodificada localmente cuando una imagen se codifica y un bloque adyacente, adyacente al bloque;

Controlar de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente; y

codificar la imagen en unidades que tengan una estructura de capa utilizando una imagen a la que se le aplica el filtro de desbloqueo.

RESUMEN

La presente tecnología se refiere a un dispositivo de procesamiento de imagen y un método de procesamiento de imagen que permite que un proceso de filtración de desbloqueo aplique adecuadamente la filtración. Un píxel ($p0_i$) del cual el valor antes del proceso de desbloqueo es 255 (línea continua) cambia ampliamente a 159 (línea de puntos) después del proceso de desbloqueo convencional. Por lo tanto, se realiza en la filtración fuerte un proceso de recorte que tiene un valor de recorte de 10, con lo cual el pixel (p0i) del cual el valor es 255 (línea continua) antes del proceso de desbloqueo llega a ser 245 (línea gruesa). Por lo tanto, se puede suprimir un cambio en el valor de píxel que ocurre en la técnica convencional tanto como es posible. Por ejemplo, esta divulgación se puede aplicar a un dispositivo de procesamiento de imagen.

FIG. 1

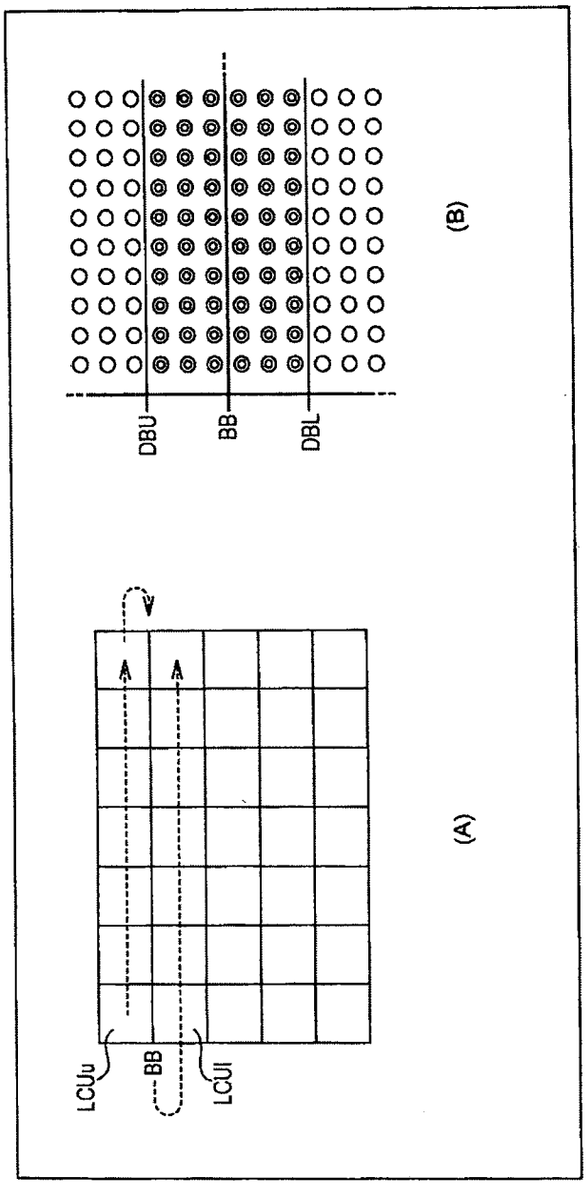
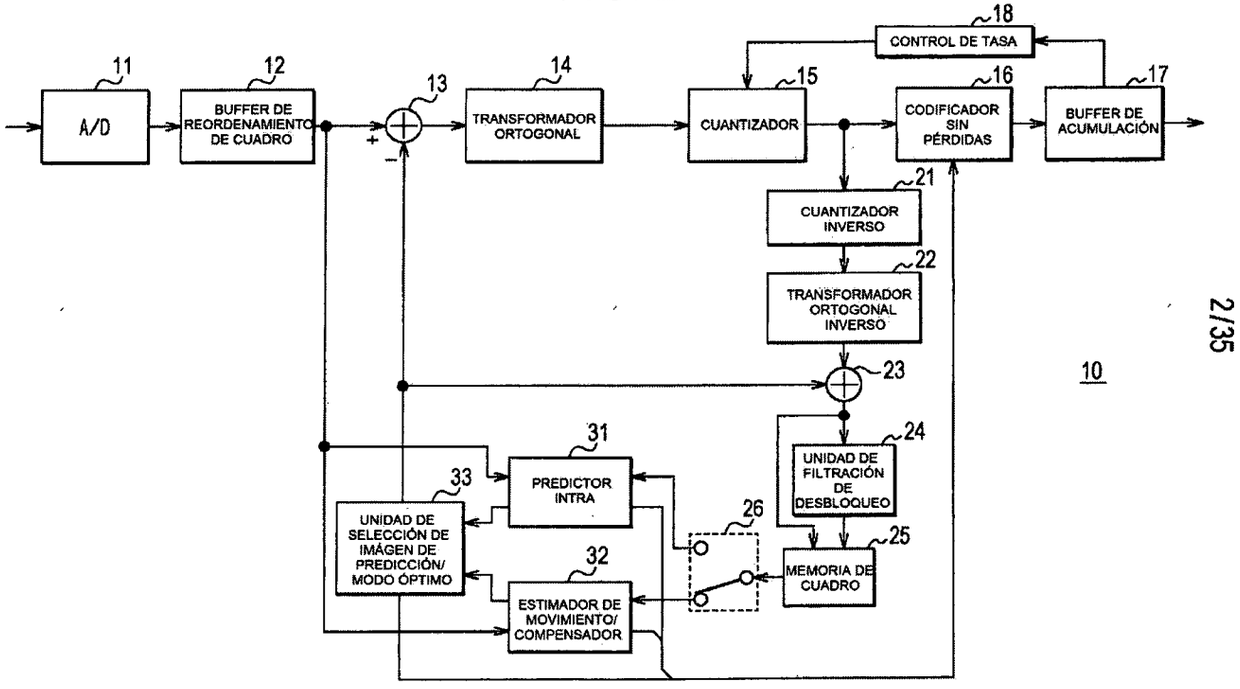


FIG. 2



SP334941

FIG. 3

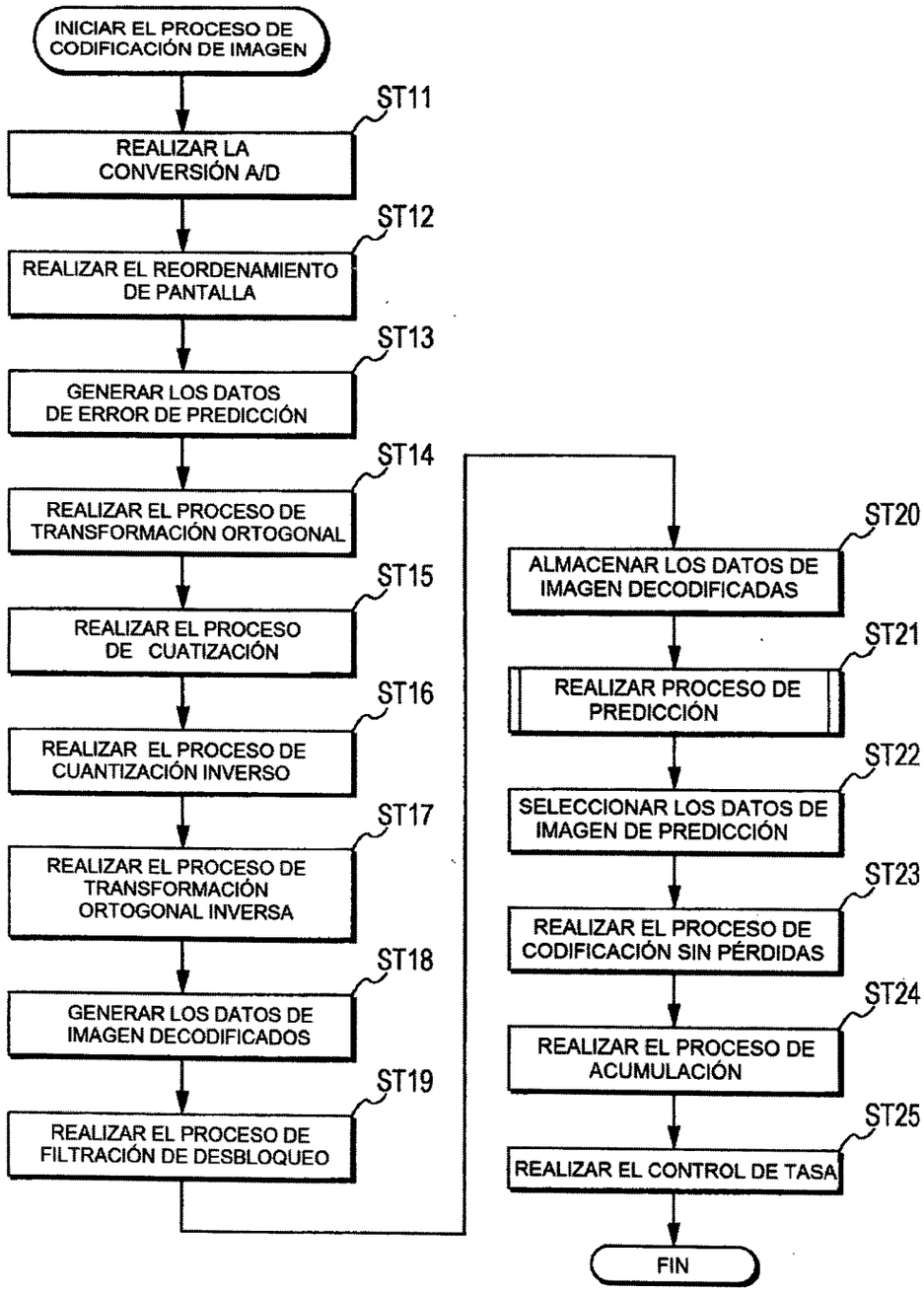

INICIAR EL PROCESO DE CODIFICACIÓN DE IMAGEN
ST11
REALIZAR LA CONVERSIÓN A/D
ST12
REALIZAR EL REORDENAMIENTO DE PANTALLA
ST13
GENERAR LOS DATOS DE ERROR DE PREDICCIÓN
ST14
REALIZAR EL PROCESO DE TRANSFORMACIÓN ORTOGONAL
ST15
REALIZAR EL PROCESO DE CUATIZACIÓN
ST16
REALIZAR EL PROCESO DE CUANTIZACIÓN INVERSO
ST17
REALIZAR EL PROCESO DE TRANSFORMACIÓN ORTOGONAL INVERSA
ST18
GENERAR LOS DATOS DE IMAGEN DECODIFICADOS
ST19
REALIZAR EL PROCESO DE FILTRACIÓN DE DESBLOQUEO
ST20
ALMACENAR LOS DATOS DE IMAGEN DECODIFICADAS
ST21
REALIZAR PROCESO DE PREDICCIÓN
ST22
SELECCIONAR LOS DATOS DE IMAGEN DE PREDICCIÓN
ST23
REALIZAR EL PROCESO DE CODIFICACIÓN SIN PÉRDIDAS
ST24
REALIZAR EL PROCESO DE ACUMULACIÓN
ST25
REALIZAR EL CONTROL DE TASA
FIN

FIG. 4

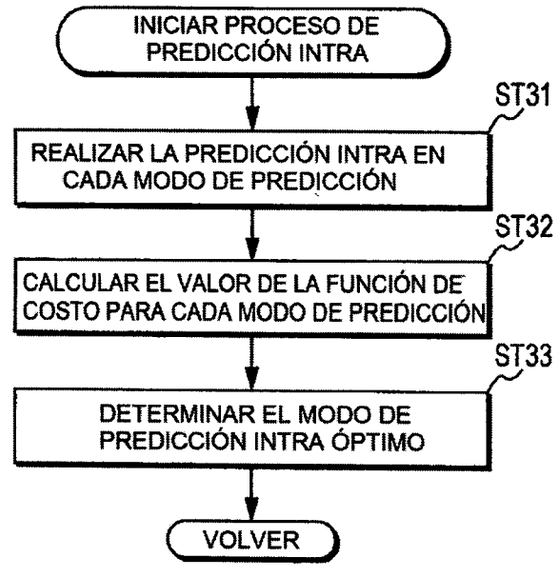

INICIAR PROCESO DE PREDICCIÓN INTRA
ST31
REALIZAR LA PREDICCIÓN INTRA EN CADA MODO DE PREDICCIÓN
ST32
CALCULAR EL VALOR DE LA FUNCIÓN DE COSTO PARA CADA MODO DE PREDICCIÓN
ST33
DETERMINAR EL MODO DE PREDICCIÓN INTRA ÓPTIMO
VOLVER

FIG. 5

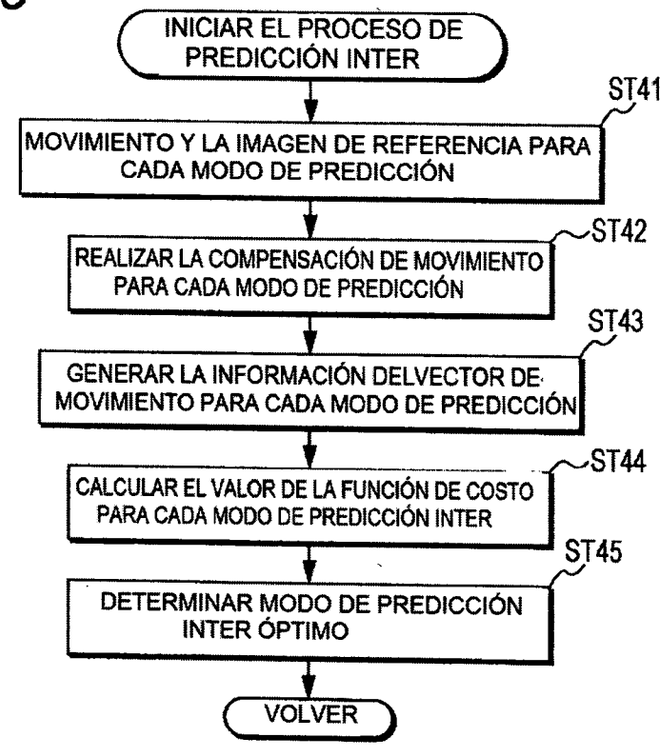

INICIAR EL PROCESO DE PREDICCIÓN INTER
ST41
MOVIMIENTO Y LA IMAGEN DE REFERENCIA PARA CADA MODO DE PREDICCIÓN
ST42
REALIZAR LA COMPENSACIÓN DE MOVIMIENTO PARA CADA MODO DE PREDICCIÓN
ST43
GENERAR LA INFORMACIÓN DELVECTOR DE MOVIMIENTO PARA CADA MODO DE PREDICCIÓN
ST44
CALCULAR EL VALOR DE LA FUNCIÓN DE COSTO PARA CADA MODO DE PREDICCIÓN INTER
ST45
DETERMINAR MODO DE PREDICCIÓN INTER ÓPTIMO
VOLVER

FIG. 6

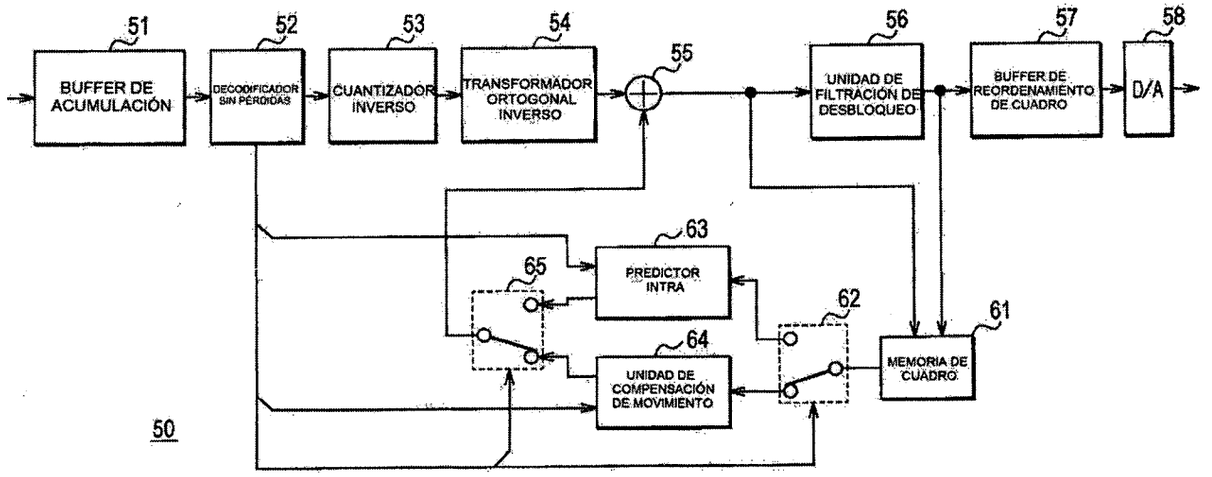

51
BUFFER DE ACUMULACIÓN
52
DECODIFICADOR SIN PÉRDIDAS
53
CUANTIZADOR INVERSO
54
TRANSFORMADOR ORTOGONAL INVERSO
55
56
UNIDAD DE FILTRACIÓN DE DESBLOQUEO
57
BUFFER DE REORDENAMIENTO DE CUADRO
58
D/A
63
PREDICTOR INTRA
65
62
61
64
UNIDAD DE COMPENSACIÓN DE MOVIMIENTO
MEMORIA DE CUADRO
50

FIG. 7

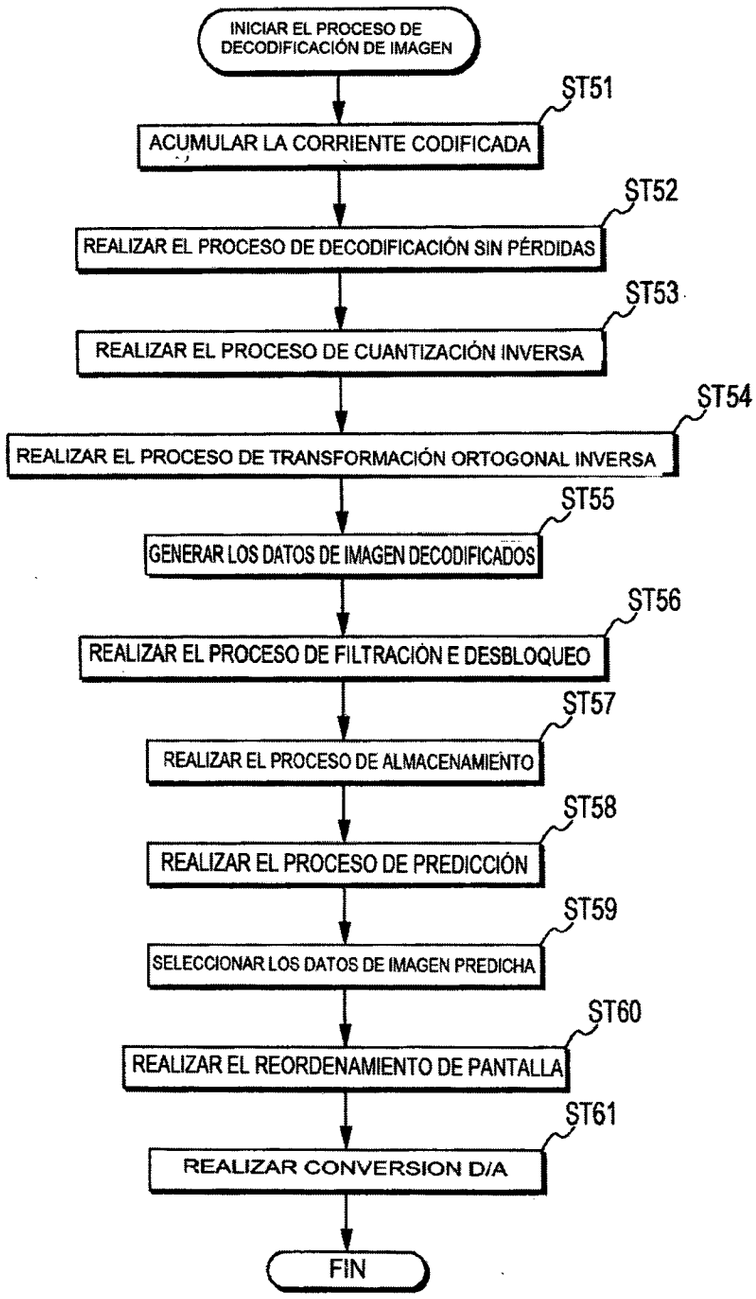

INICIAR EL PROCESO DE DECODIFICACIÓN DE IMAGEN
ST51
ACUMULAR LA CORRIENTE CODIFICADA
ST52
REALIZAR EL PROCESO DE DECODIFICACIÓN SIN PÉRDIDAS
ST53
REALIZAR EL PROCESO DE CUANTIZACIÓN INVERSA
ST54
REALIZAR EL PROCESO DE TRANSFORMACIÓN ORTOGONAL INVERSA
ST55
GENERAR LOS DATOS DE IMAGEN DECODIFICADOS
ST56
REALIZAR EL PROCESO DE FILTRACIÓN E DESBLOQUEO
ST57
REALIZAR EL PROCESO DE ALMACENAMIENTO
ST58
REALIZAR EL PROCESO DE PREDICCIÓN
ST59
SELECCIONAR LOS DATOS DE IMAGEN PREDICHA
ST60
REALIZAR EL REORDENAMIENTO DE PANTALLA
ST61
REALIZAR CONVERSION D/A
FIN

FIG. 8

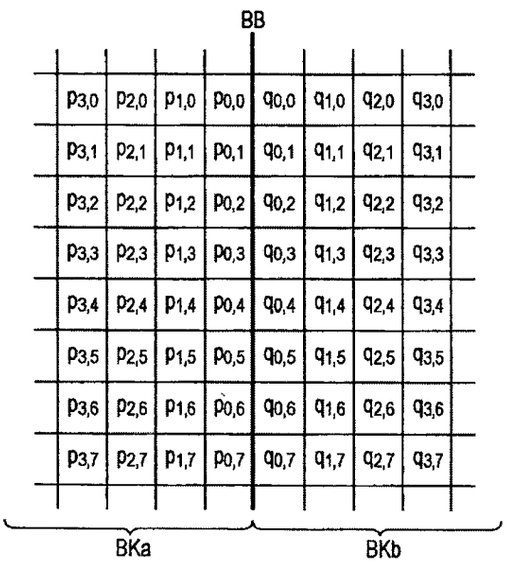

BB
p3,0 p2,0 p1,0 p0,0 q0,0 q1,0 q2,0 q3,0
p3,1 p2,1 p1,1 p0,1 q0,1 q1,1 q2,1 q3,1
p3,2 p2,2 p1,2 p0,2 q0,2 q1,2 q2,2 q3,2
p3,3 p2,3 p1,3 p0,3 q0,3 q1,3 q2,3 q3,3
p3,4 p2,4 p1,4 p0,4 q0,4 q1,4 q2,4 q3,4
p3,5 p2,5 p1,5 p0,5 q0,5 q1,5 q2,5 q3,5
p3,6 p2,6 p1,6 p0,6 q0,6 q1,6 q2,6 q3,6
p3,7 p2,7 p1,7 p0,7 q0,7 q1,7 q2,7 q3,7
BKa
BKb

FIG. 9

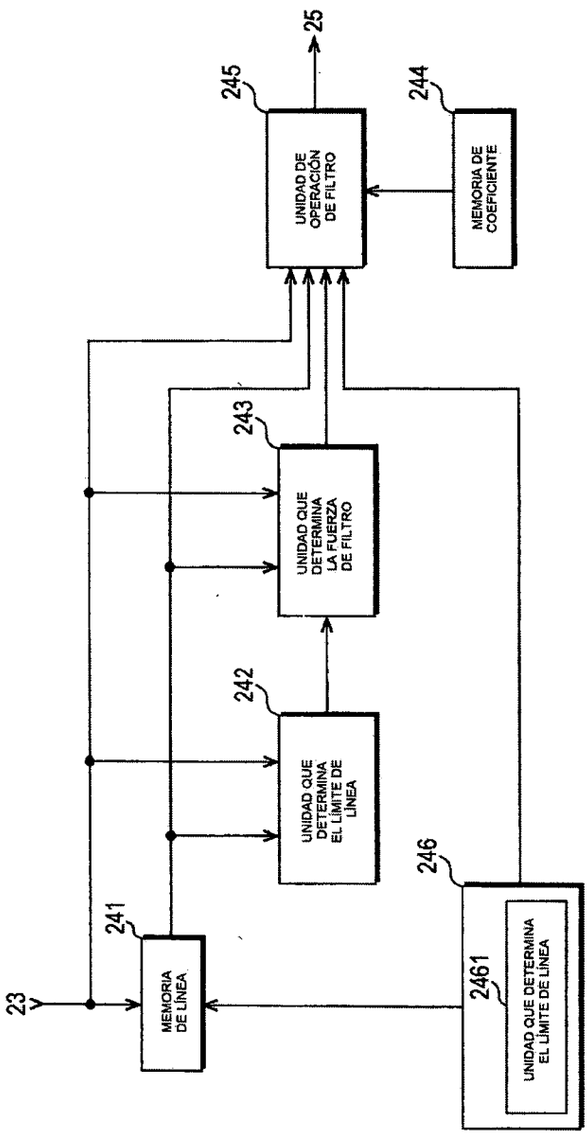

25
245
UNIDAD DE OPERACIÓN DE FILTRO
244
MEMORIA DE COEFICIENTE
243
UNIDAD QUE DETERMINA LA FUERZA DE FILTRO
242
UNIDAD QUE DETERMINA EL LÍMITE DE LÍNEA
246
2461
UNIDAD QUE DETERMINA EL LÍMITE DE LÍNEA
241
MEMORIA DE LÍNEA
23

FIG. 10

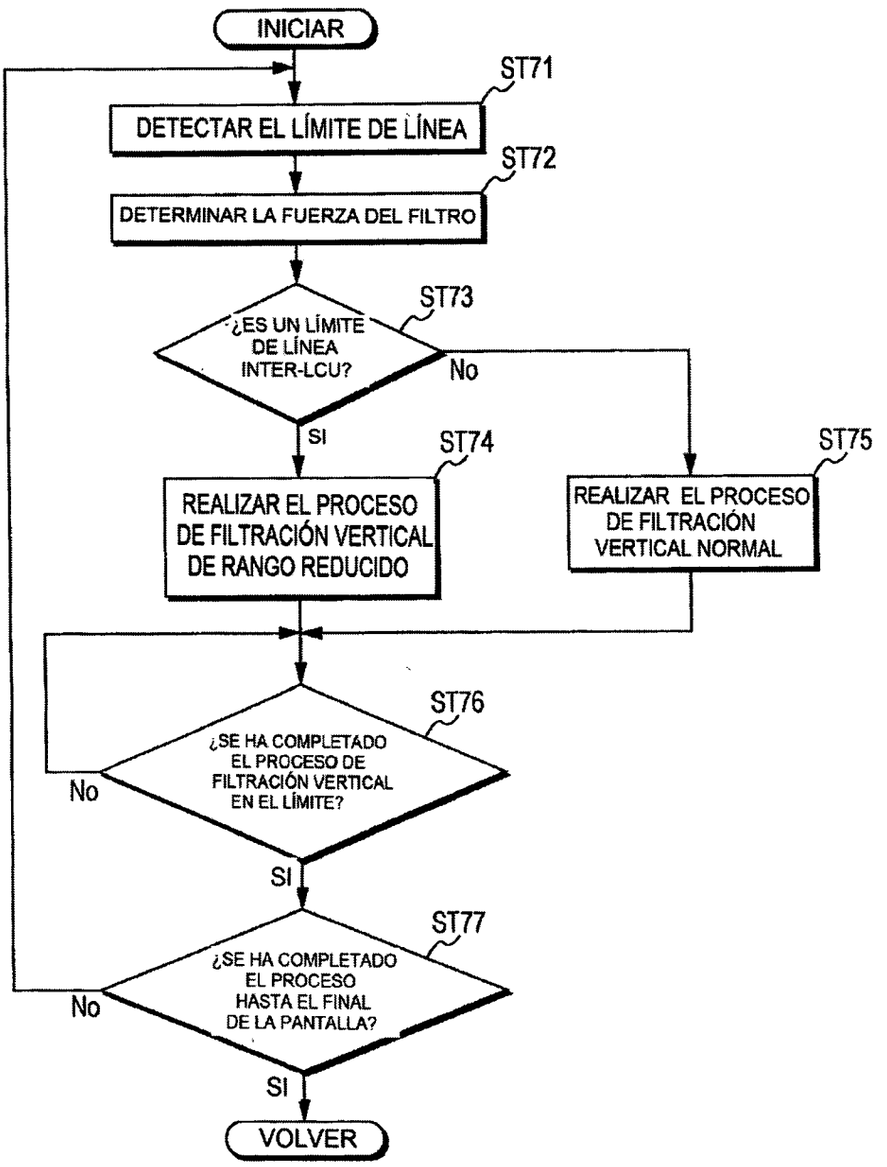

INICIAR
ST71
DETECTAR EL LÍMITE DE LÍNEA
ST72
DETERMINAR LA FUERZA DEL FILTRO
ST73
¿ES UN LÍMITE DE LÍNEA INTER-LCU?
No
SI
ST74
REALIZAR EL PROCESO DE FILTRACIÓN VERTICAL DE RANGO REDUCIDO
ST75
REALIZAR EL PROCESO DE FILTRACIÓN VERTICAL NORMAL
ST76
¿SE HA COMPLETADO EL PROCESO DE FILTRACIÓN VERTICAL EN EL LÍMITE?
No
SI
ST77
¿SE HA COMPLETADO EL PROCESO HASTA EL FINAL DE LA PANTALLA?
No
SI
VOLVER

FIG. 11

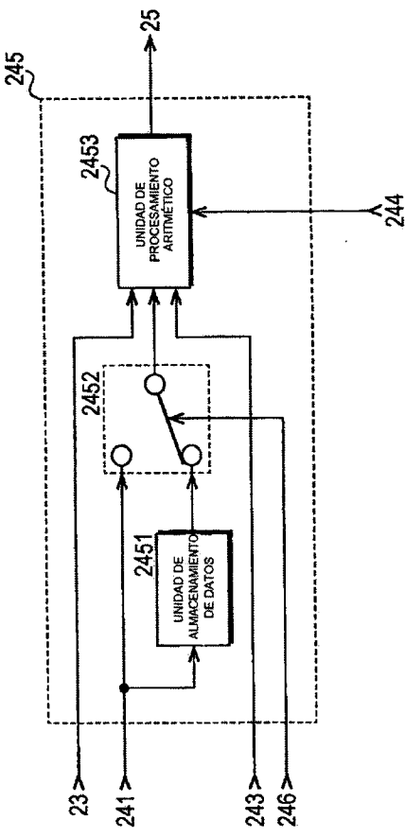

245
25
2453
UNIDAD DE PROCESAMIENTO ARITMÉTICO
244
2452
2451
UNIDAD DE ALMACENAMIENTO DE DATOS
23
241
243
246

FIG. 12

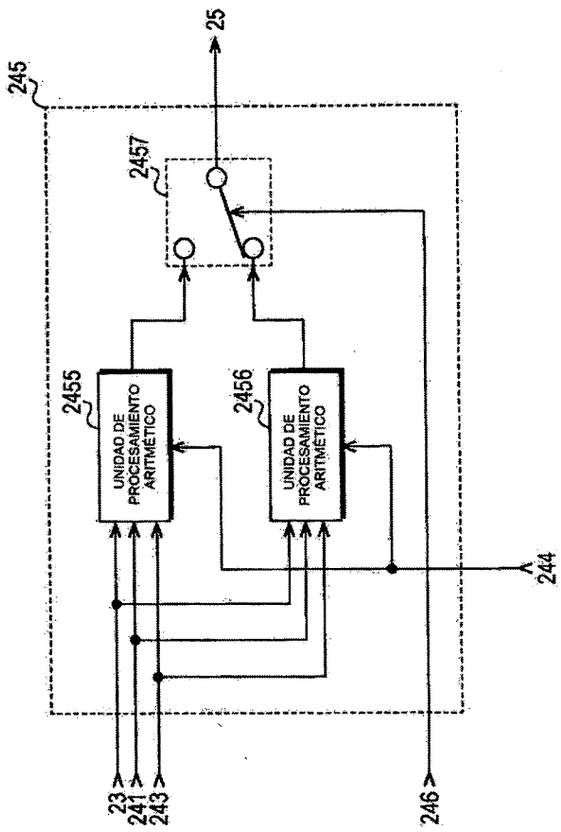

25
245
2457
2455
UNIDAD DE PROCESAMIENTO ARITMÉTICO
2456
UNIDAD DE PROCESAMIENTO ARITMÉTICO
244
23
241
243
246

FIG. 13

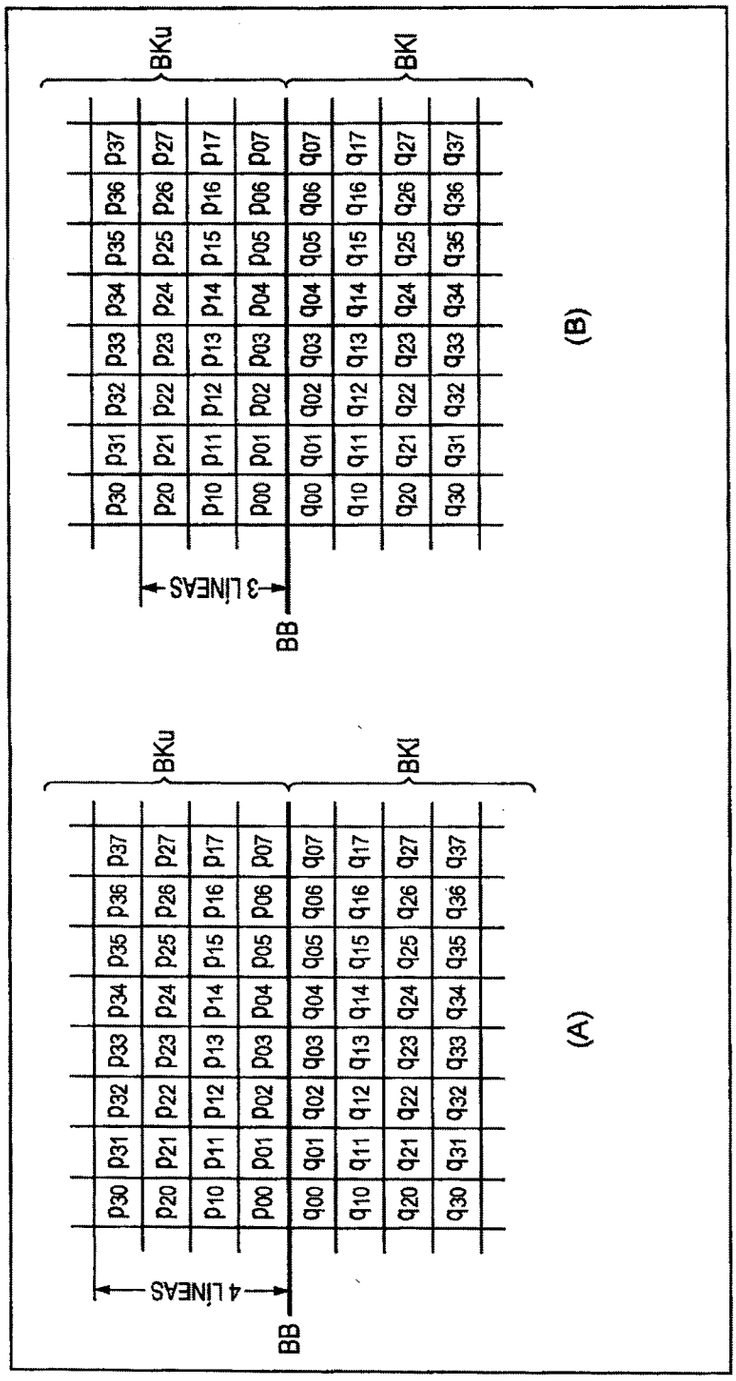


FIG. 14

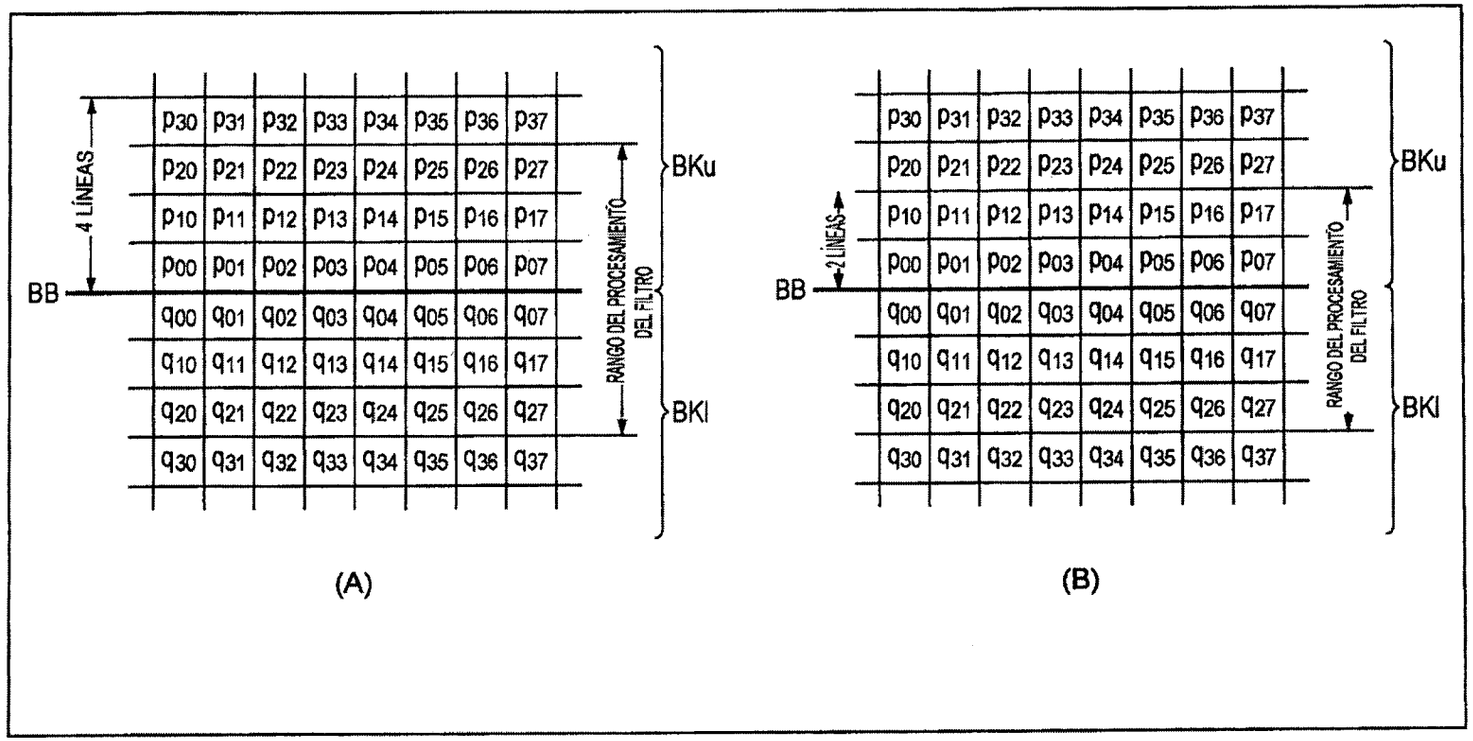

4 LÍNEAS
BB
p30 p31 p32 p33 p34 p35 p36 p37
p20 p21 p22 p23 p24 p25 p26 p27
p10 p11 p12 p13 p14 p15 p16 p17
p00 p01 p02 p03 p04 p05 p06 p07
q00 q01 q02 q03 q04 q05 q06 q07
q10 q11 q12 q13 q14 q15 q16 q17
q20 q21 q22 q23 q24 q25 q26 q27
q30 q31 q32 q33 q34 q35 q36 q37
RANGO DEL PROCESAMIENTO DEL FILTRO
BKu
BKl
(A)
2 LÍNEAS
BB
p30 p31 p32 p33 p34 p35 p36 p37
p20 p21 p22 p23 p24 p25 p26 p27
p10 p11 p12 p13 p14 p15 p16 p17
p00 p01 p02 p03 p04 p05 p06 p07
q00 q01 q02 q03 q04 q05 q06 q07
q10 q11 q12 q13 q14 q15 q16 q17
q20 q21 q22 q23 q24 q25 q26 q27
q30 q31 q32 q33 q34 q35 q36 q37
RANGO DEL PROCESAMIENTO DEL FILTRO
BKu
BKl
(B)

FIG. 15

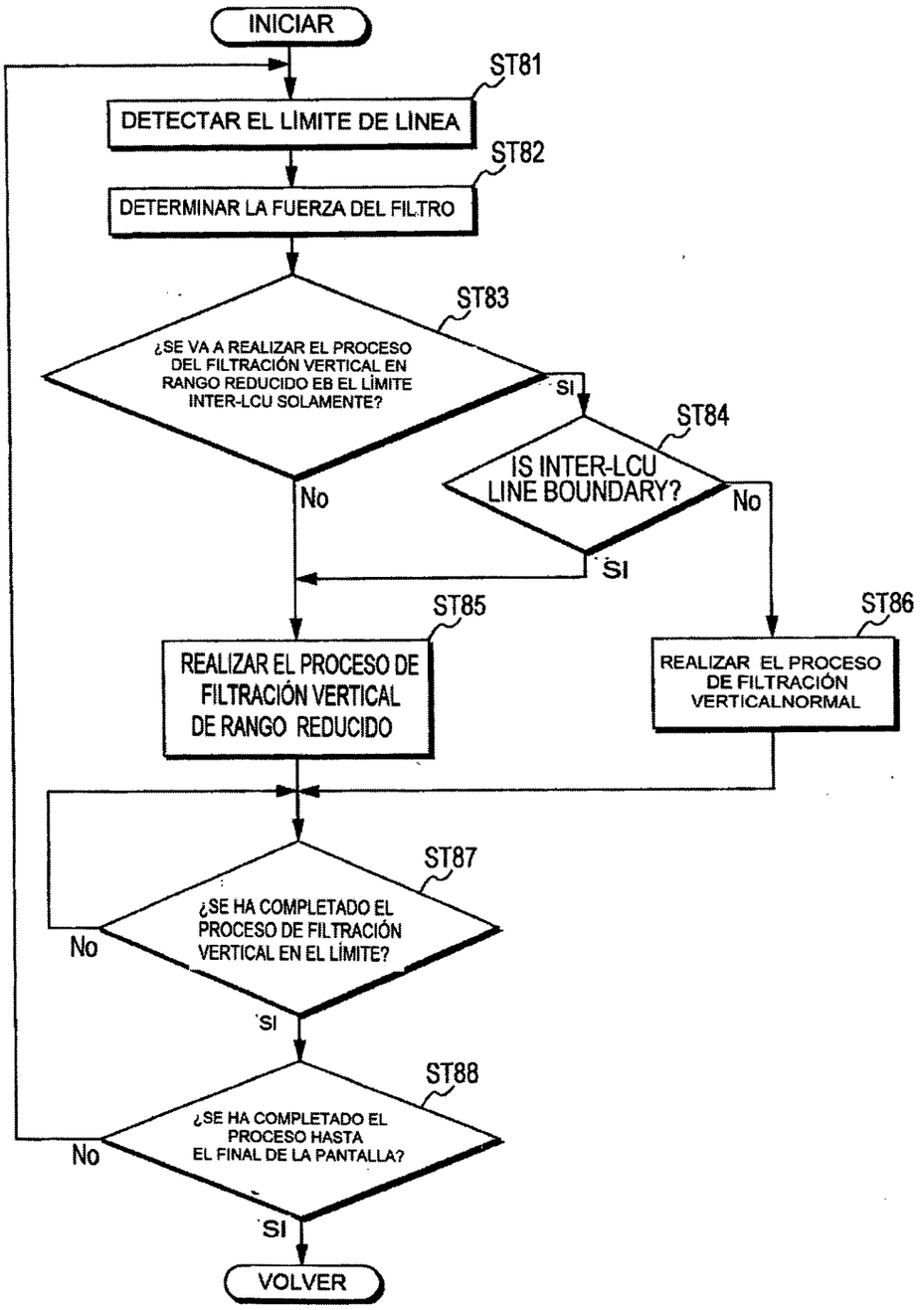

INICIAR
ST81
DETECTAR EL LÍMITE DE LÍNEA
ST82
DETERMINAR LA FUERZA DEL FILTRO
ST83
¿SE VA A REALIZAR EL PROCESO DEL FILTRACIÓN VERTICAL EN RANGO REDUCIDO EB EL LÍMITE INTER-LCU SOLAMENTE?
SI
No
ST84
IS INTER-LCU LINE BOUNDARY?
No
SI
ST85
REALIZAR EL PROCESO DE FILTRACIÓN VERTICAL DE RANGO REDUCIDO
ST86
REALIZAR EL PROCESO DE FILTRACIÓN VERTICALNORMAL
ST87
¿SE HA COMPLETADO EL PROCESO DE FILTRACIÓN VERTICAL EN EL LÍMITE?
No
SI
ST88
¿SE HA COMPLETADO EL PROCESO HASTA EL FINAL DE LA PANTALLA?
No
SI
VOLVER

FIG. 16

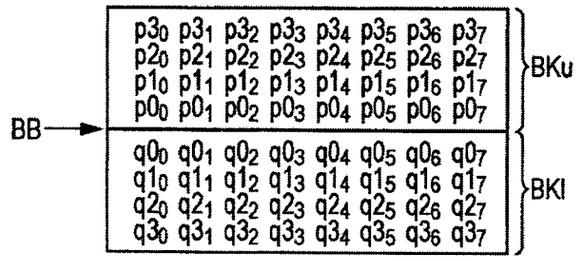

BB
BKu
BKl

FIG. 17

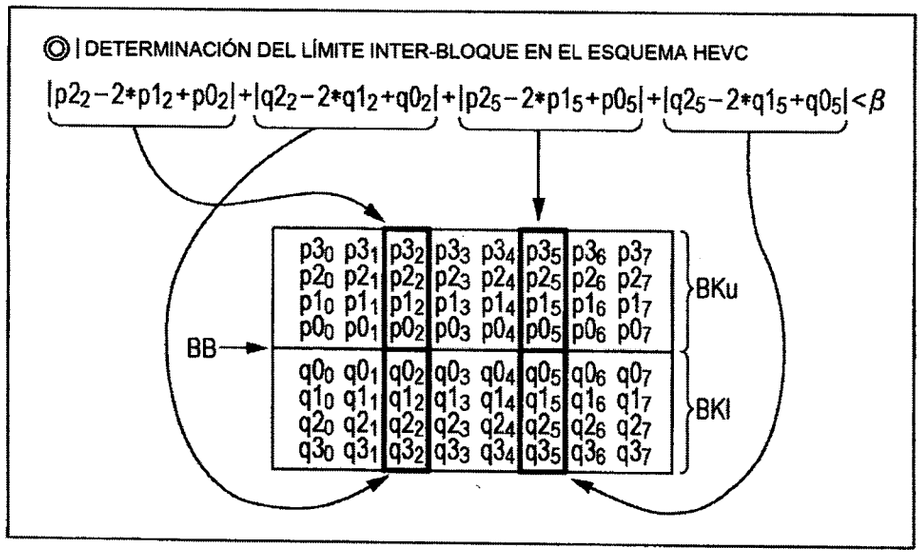

DETERMINACIÓN DEL LÍMITE INTER-BLOQUE EN EL ESQUEMA HEVC
|p22−2*p12+p02|+|q22−2*q12+q02|+|p25−2*p15+p05|+|q25−2*q15+q05|<β
BB
BKu
BKl

FIG. 18

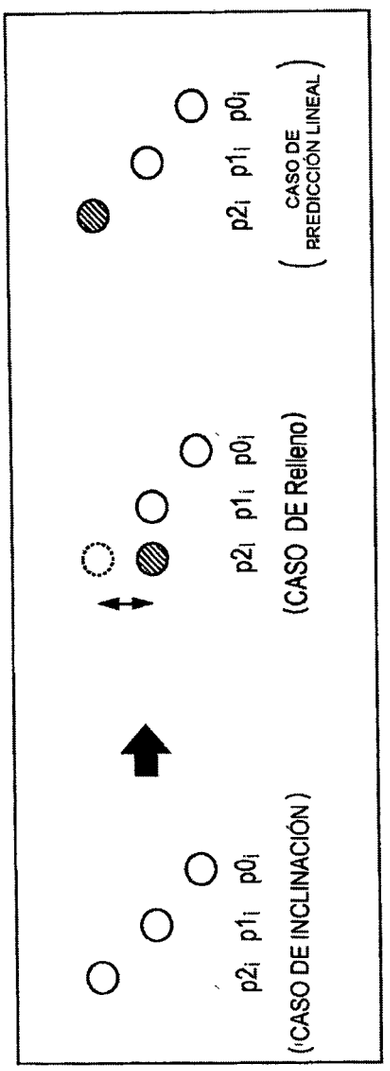

p0i p1i p2i
(CASO DE PREDICCIÓN LINEAL)
p0i p1i p2i
(CASO DE Relleno)
p0i p1i p2i
(CASO DE INCLINACIÓN)

FIG. 19

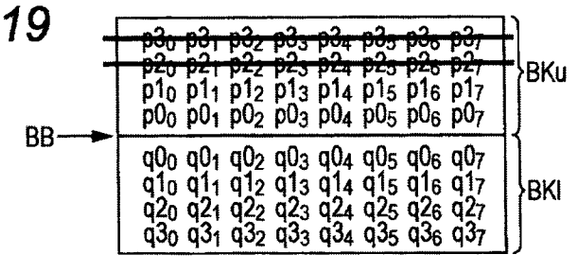

p3₀ p3₁ p3₂ p3₃ p3₄ p3₅ p3₆ p3₇
p2₀ p2₁ p2₂ p2₃ p2₄ p2₅ p2₆ p2₇
p1₀ p1₁ p1₂ p1₃ p1₄ p1₅ p1₆ p1₇
p0₀ p0₁ p0₂ p0₃ p0₄ p0₅ p0₆ p0₇
BKu
BB
q0₀ q0₁ q0₂ q0₃ q0₄ q0₅ q0₆ q0₇
q1₀ q1₁ q1₂ q1₃ q1₄ q1₅ q1₆ q1₇
q2₀ q2₁ q2₂ q2₃ q2₄ q2₅ q2₆ q2₇
q3₀ q3₁ q3₂ q3₃ q3₄ q3₅ q3₆ q3₇
BKl

FIG. 20

	AI_HE				RA_HE				LB_HE				LP_HE			
	22	27	32	37	22	27	32	37	22	27	32	37	22	27	32	37
Clase A	16	25	23	29	12	19	19	18								
Clase B	9	37	42	50	15	29	25	84	9	55	53	73	12	38	50	82
Clase C	29	40	54	109	9	28	34	35	27	15	34	44	5	24	34	49
Clase D	7	13	57	54	6	9	40	42	3	8	20	22	4	8	18	20
Clase E	15	16	20	28					5	7	15	17	6	11	16	20
Clase F	190	190	190	191	178	189	191	190	191	188	189	182	191	188	181	169

FIG. 21

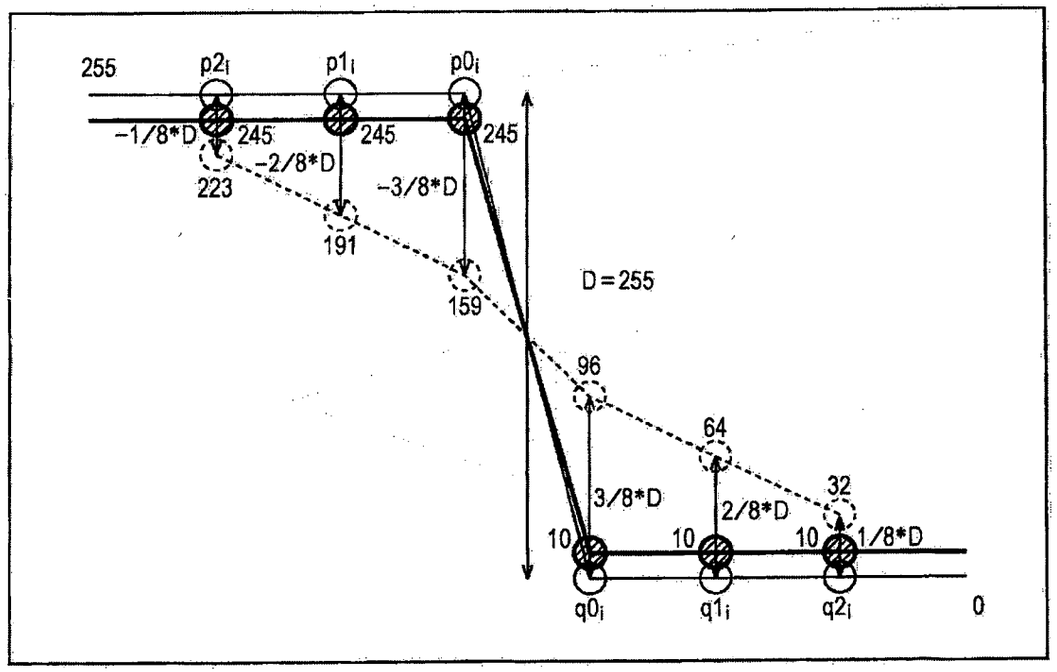

255
p2i
p1i
p0i
−1/8*D
245
245
245
−2/8*D
223
−3/8*D
191
159
D=255
96
64
32
3/8*D
2/8*D
1/8*D
10
10
10
q0i
q1i
q2i
0

FIG. 22

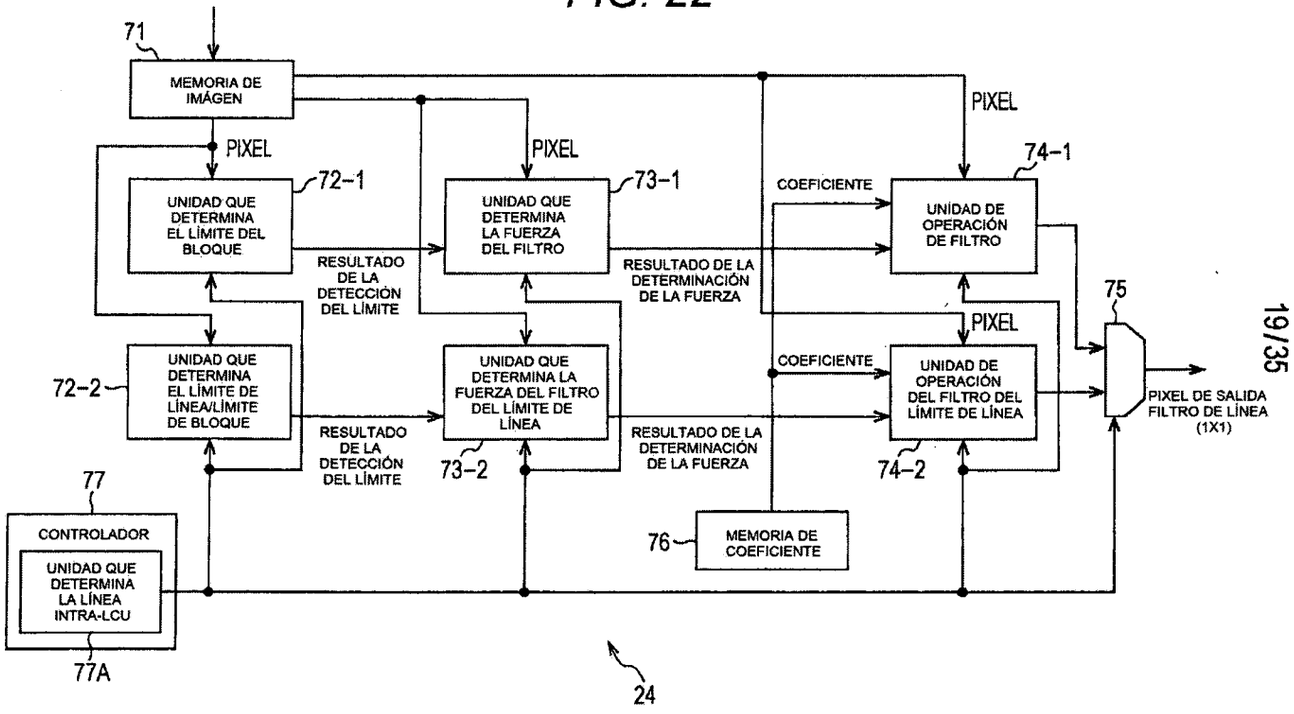

71
MEMORIA DE IMÁGEN
PIXEL
PIXEL
PIXEL
72-1
UNIDAD QUE DETERMINA EL LÍMITE DEL BLOQUE
73-1
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO
74-1
UNIDAD DE OPERACIÓN DE FILTRO
COEFICIENTE
RESULTADO DE LA DETECCIÓN DEL LÍMITE
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
75
PIXEL
72-2
UNIDAD QUE DETERMINA EL LÍMITE DE LÍNEA/LÍMITE DE BLOQUE
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO DEL LÍMITE DE LÍNEA
COEFICIENTE
UNIDAD DE OPERACIÓN DEL FILTRO DEL LÍMITE DE LÍNEA
PIXEL DE SALIDA FILTRO DE LÍNEA (1X1)
RESULTADO DE LA DETECCIÓN DEL LÍMITE
73-2
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
74-2
77
CONTROLADOR
UNIDAD QUE DETERMINA LA LÍNEA INTRA-LCU
77A
76
MEMORIA DE COEFICIENTE
24

FIG. 23

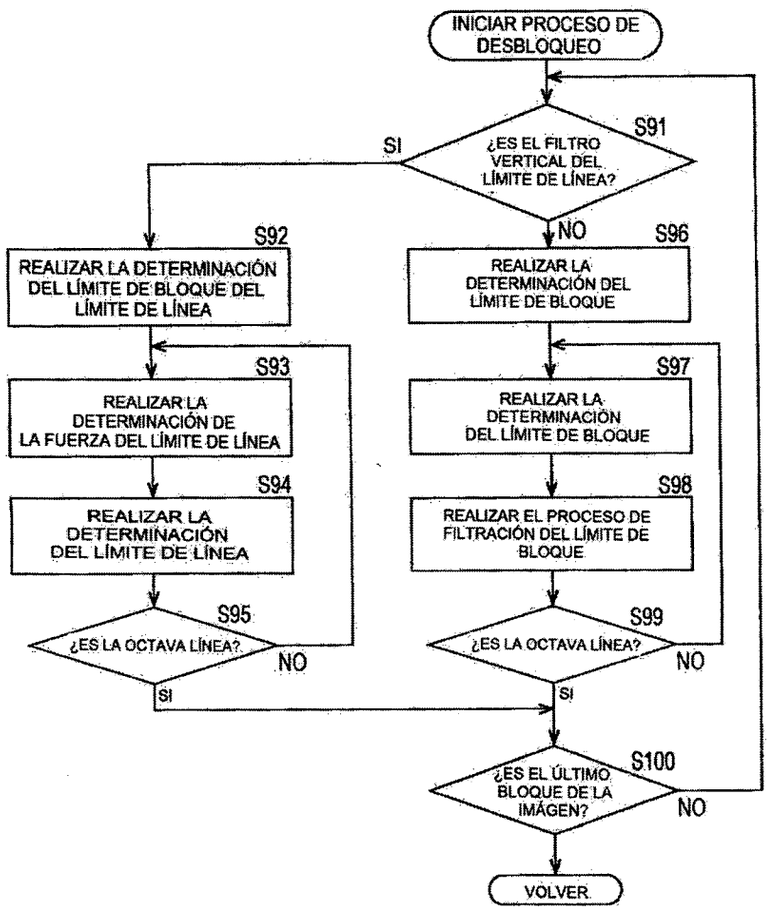

INICIAR PROCESO DE DESBLOQUEO
S91
¿ES EL FILTRO VERTICAL DEL LÍMITE DE LÍNEA?
SI
NO
S92
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE BLOQUE DEL LÍMITE DE LÍNEA
S93
REALIZAR LA DETERMINACIÓN DE LA FUERZA DEL LÍMITE DE LÍNEA
S94
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE LÍNEA
S95
¿ES LA OCTAVA LÍNEA?
NO
SI
S96
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE BLOQUE
S97
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE BLOQUE
S98
REALIZAR EL PROCESO DE FILTRACIÓN DEL LÍMITE DE BLOQUE
S99
¿ES LA OCTAVA LÍNEA?
NO
SI
S100
¿ES EL ÚLTIMO BLOQUE DE LA IMÁGEN?
NO
VOLVER

FIG. 24

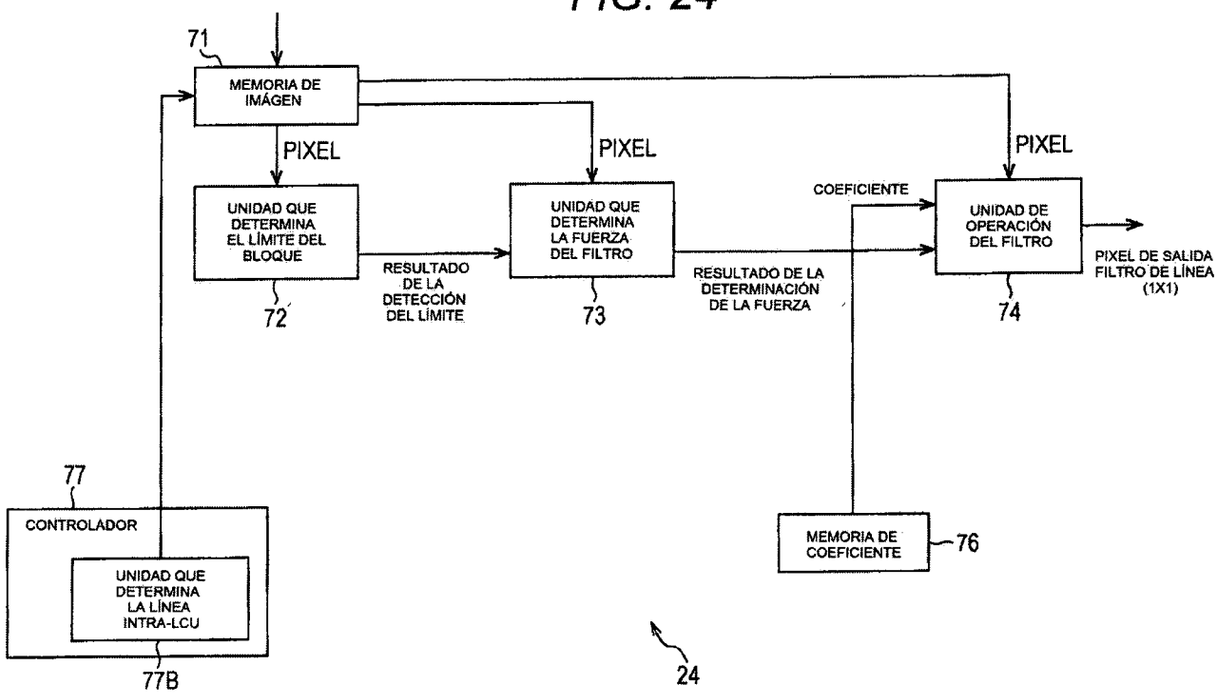

71
MEMORIA DE IMÁGEN
PIXEL
PIXEL
PIXEL
UNIDAD QUE DETERMINA EL LÍMITE DEL BLOQUE
72
RESULTADO DE LA DETECCIÓN DEL LÍMITE
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO
73
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
COEFICIENTE
UNIDAD DE OPERACIÓN DEL FILTRO
74
PIXEL DE SALIDA FILTRO DE LÍNEA (1X1)
77
CONTROLADOR
UNIDAD QUE DETERMINA LA LÍNEA INTRA-LCU
77B
MEMORIA DE COEFICIENTE
76
24

FIG. 25

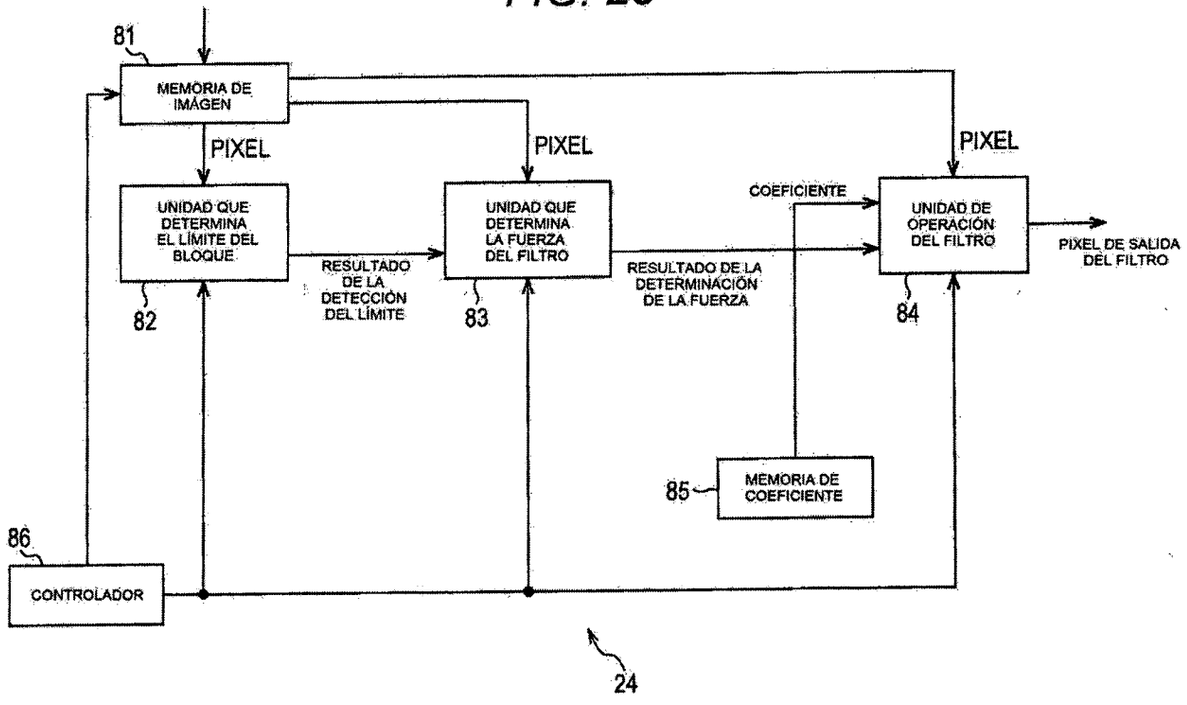

81
MEMORIA DE IMAGEN
PIXEL
PIXEL
PIXEL
UNIDAD QUE DETERMINA EL LÍMITE DEL BLOQUE
82
RESULTADO DE LA DETECCIÓN DEL LÍMITE
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO
83
COEFICIENTE
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
UNIDAD DE OPERACIÓN DEL FILTRO
84
PIXEL DE SALIDA DEL FILTRO
85
MEMORIA DE COEFICIENTE
86
CONTROLADOR
24

FIG. 26

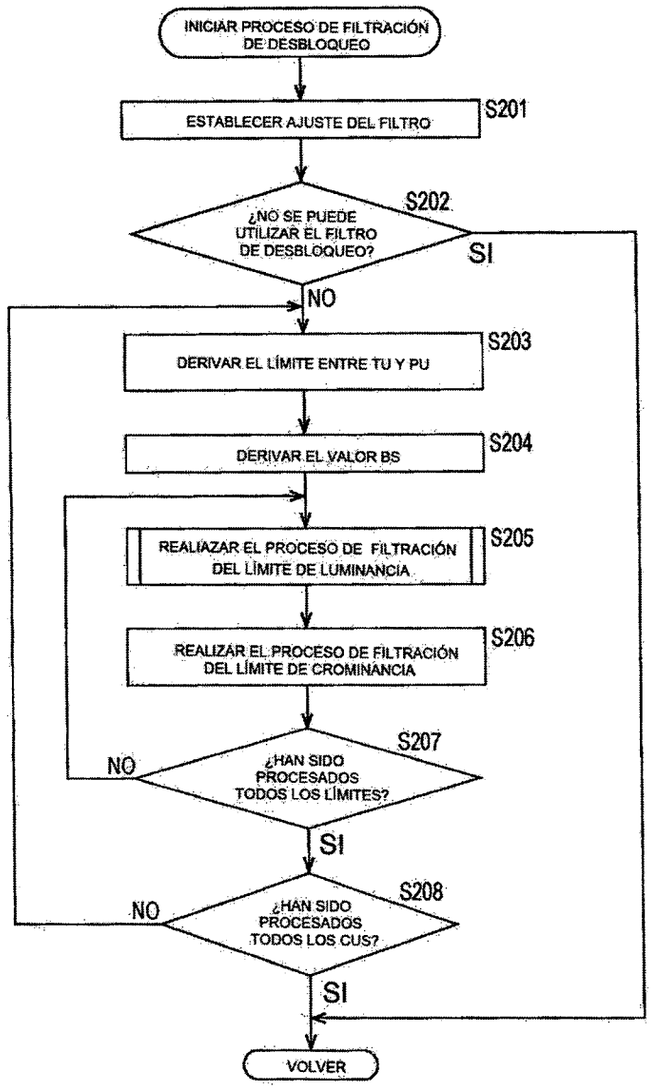

INICIAR PROCESO DE FILTRACIÓN DE DESBLOQUEO
ESTABLECER AJUSTE DEL FILTRO
S201
¿NO SE PUEDE UTILIZAR EL FILTRO DE DESBLOQUEO?
S202
SI
NO
DERIVAR EL LÍMITE ENTRE TU Y PU
S203
DERIVAR EL VALOR BS
S204
REALIAZAR EL PROCESO DE FILTRACIÓN DEL LÍMITE DE LUMINANCIA
S205
REALIZAR EL PROCESO DE FILTRACIÓN DEL LÍMITE DE CROMINANCIA
S206
¿HAN SIDO PROCESADOS TODOS LOS LÍMITES?
S207
NO
SI
¿HAN SIDO PROCESADOS TODOS LOS CUS?
S208
NO
SI
VOLVER

FIG. 27

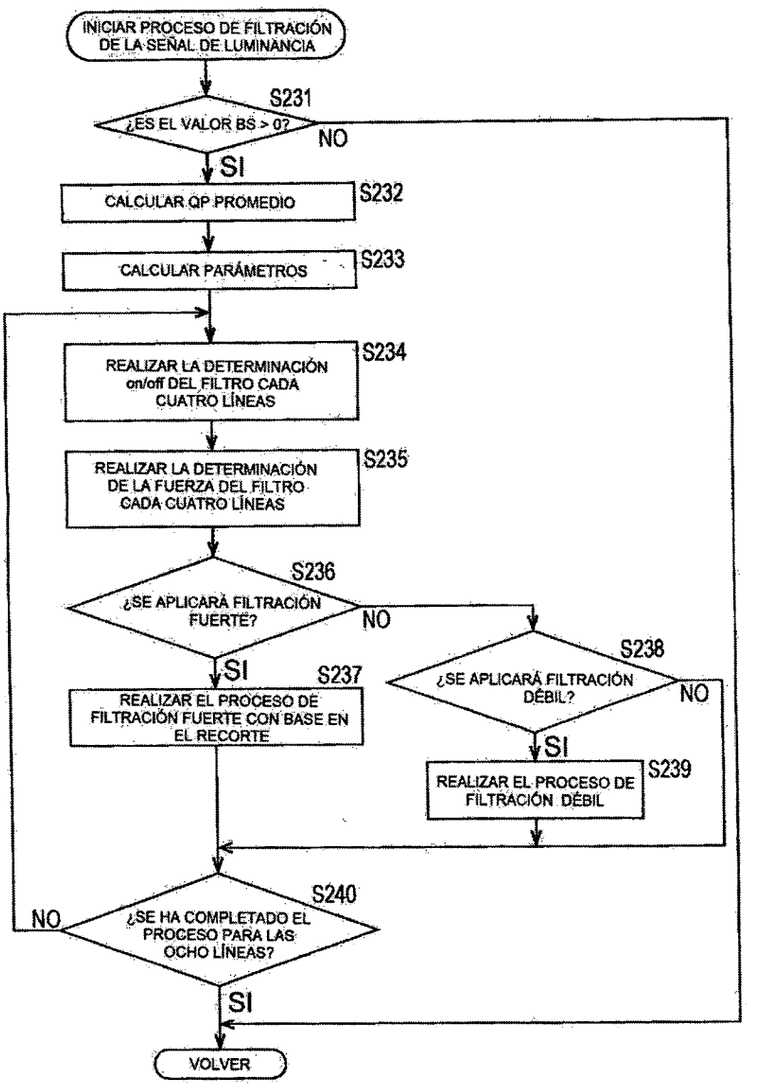

INICIAR PROCESO DE FILTRACIÓN DE LA SEÑAL DE LUMINANCIA
S231
¿ES EL VALOR BS > 0?
NO
SI
CALCULAR QP PROMEDIO
S232
CALCULAR PARÁMETROS
S233
REALIZAR LA DETERMINACIÓN on/off DEL FILTRO CADA CUATRO LÍNEAS
S234
REALIZAR LA DETERMINACIÓN DE LA FUERZA DEL FILTRO CADA CUATRO LÍNEAS
S235
S236
¿SE APLICARÁ FILTRACIÓN FUERTE?
NO
SI
S237
REALIZAR EL PROCESO DE FILTRACIÓN FUERTE CON BASE EN EL RECORTE
S238
¿SE APLICARÁ FILTRACIÓN DÉBIL?
NO
SI
S239
REALIZAR EL PROCESO DE FILTRACIÓN DÉBIL
S240
NO
¿SE HA COMPLETADO EL PROCESO PARA LAS OCHO LÍNEAS?
SI
VOLVER

FIG. 28

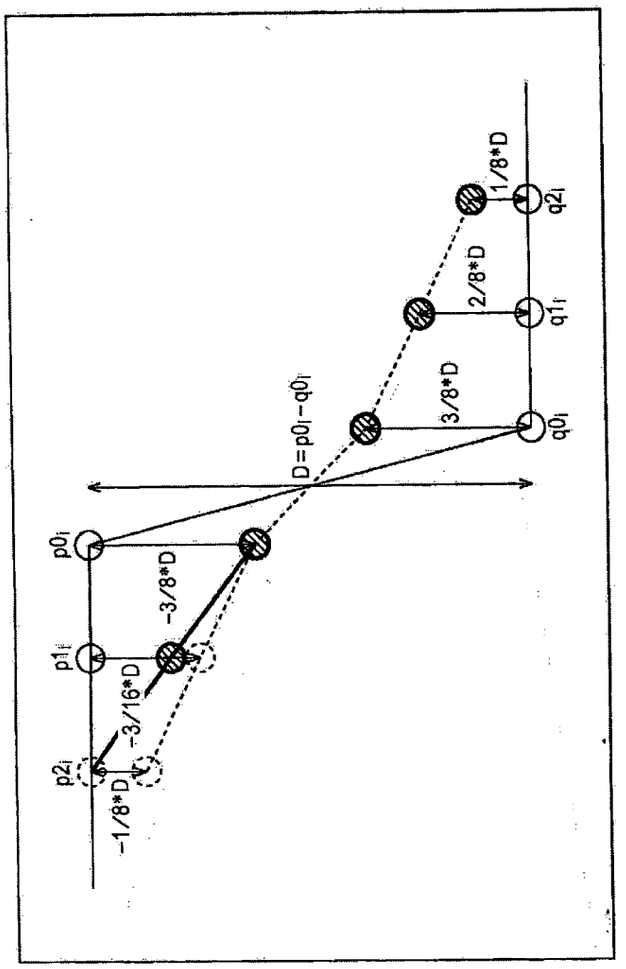


FIG. 29

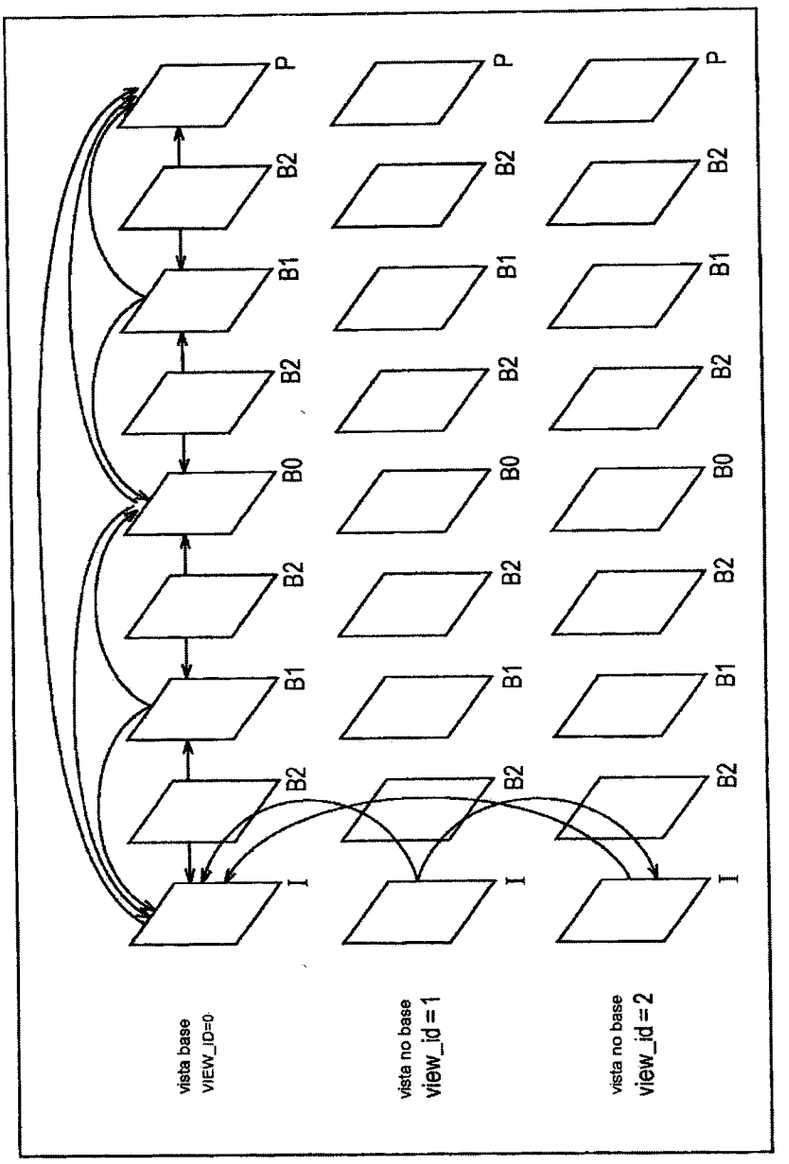

P
B2
B1
B2
B0
B2
B1
B2
I
vista base
VIEW_ID=0
vista no base
view_id=1
vista no base
view_id=2

FIG. 30

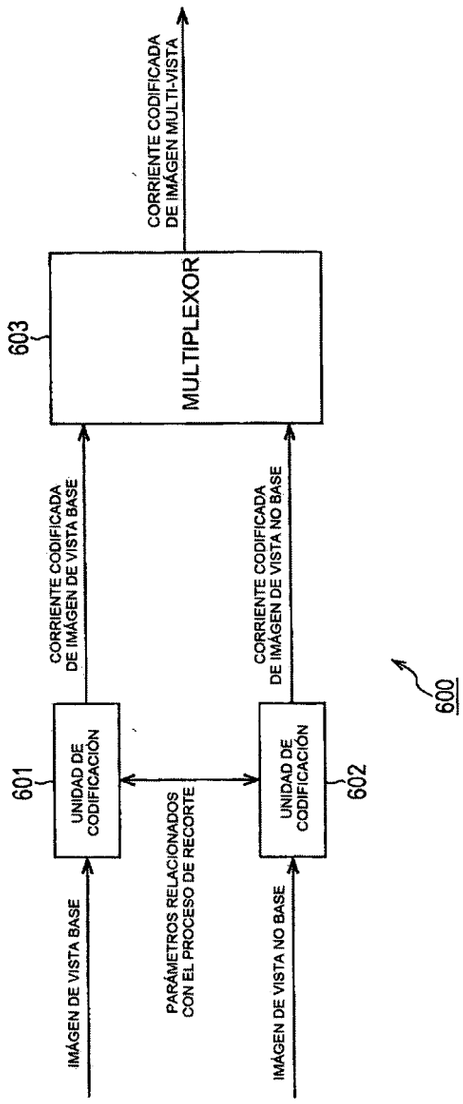

600
601
602
603
UNIDAD DE CODIFICACIÓN
UNIDAD DE CODIFICACIÓN
MULTIPLEXOR
IMÁGEN DE VISTA BASE
IMÁGEN DE VISTA NO BASE
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
CORRIENTE CODIFICADA DE IMÁGEN DE VISTA BASE
CORRIENTE CODIFICADA DE IMÁGEN DE VISTA NO BASE
CORRIENTE CODIFICADA DE IMÁGEN MULTI-VISTA

FIG. 31

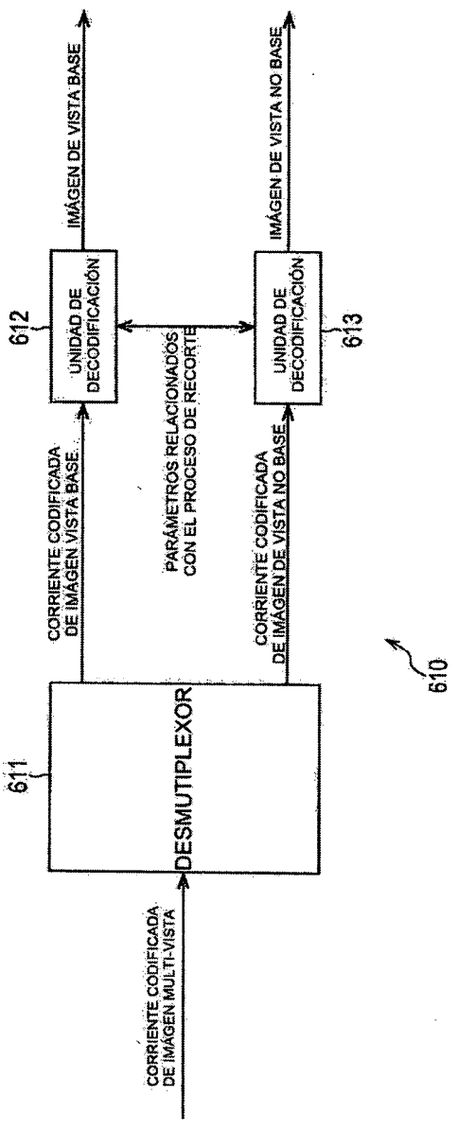

CORRIENTE CODIFICADA DE IMÁGEN MULTI-VISTA
DESMUTIPLEXOR
611
CORRIENTE CODIFICADA DE IMÁGEN VISTA BASE
CORRIENTE CODIFICADA DE IMÁGEN DE VISTA NO BASE
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
UNIDAD DE DECODIFICACIÓN
612
UNIDAD DE DECODIFICACIÓN
613
IMÁGEN DE VISTA BASE
IMÁGEN DE VISTA NO BASE
610

FIG. 32

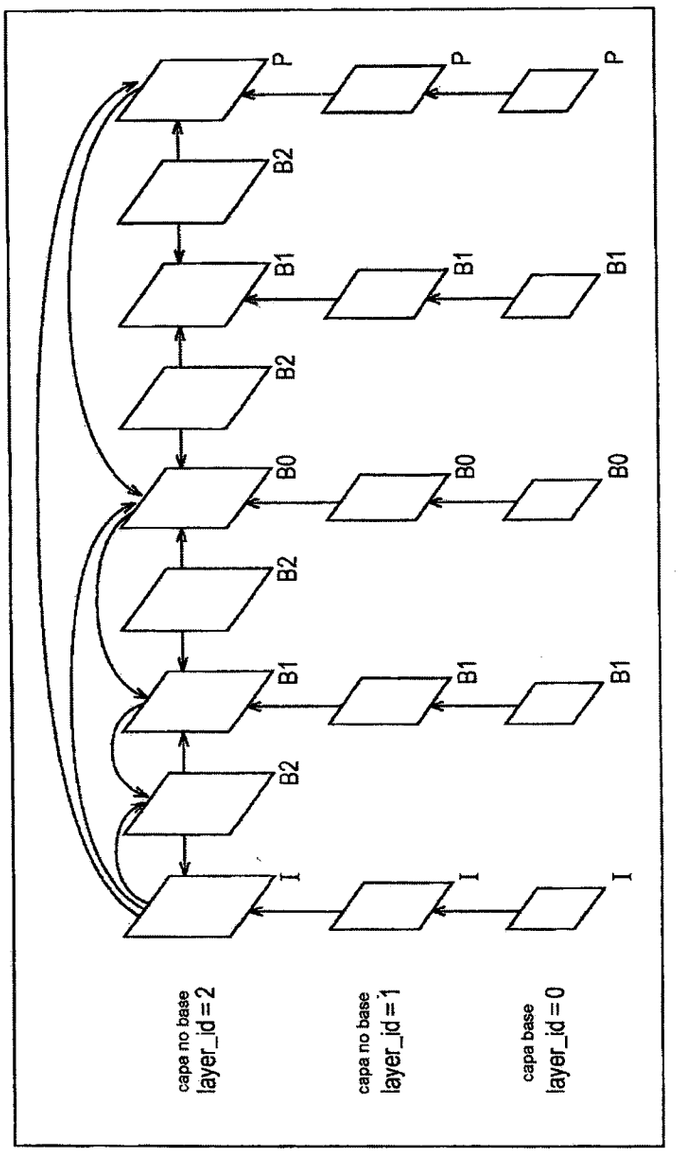

P
P
P
B2
B1
B1
B1
B2
B0
B0
B0
B2
B1
B1
B1
B2
I
I
I
capa no base
layer_id = 2
capa no base
layer_id = 1
capa base
layer_id = 0

FIG. 33

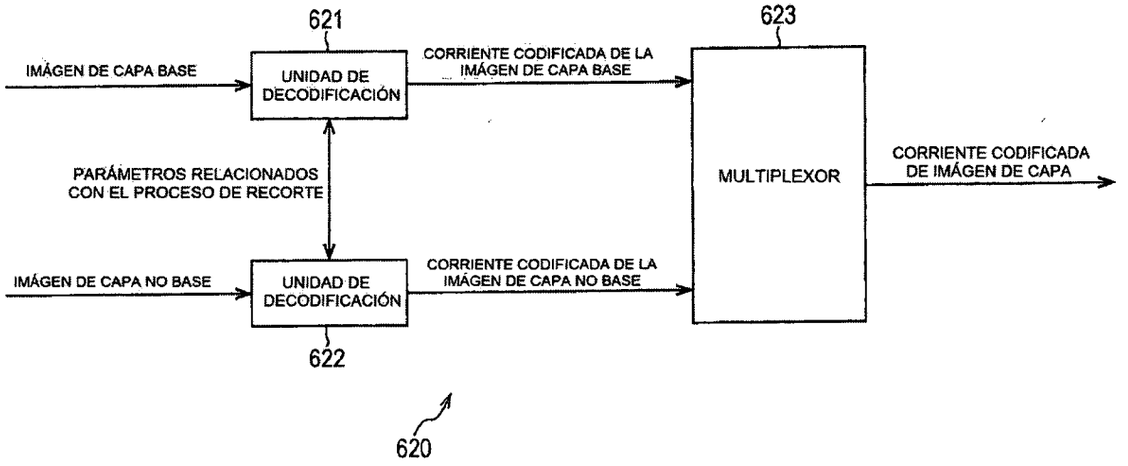

621
623
IMÁGEN DE CAPA BASE
UNIDAD DE DECODIFICACIÓN
CORRIENTE CODIFICADA DE LA IMÁGEN DE CAPA BASE
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
MULTIPLEXOR
CORRIENTE CODIFICADA DE IMÁGEN DE CAPA
IMÁGEN DE CAPA NO BASE
UNIDAD DE DECODIFICACIÓN
CORRIENTE CODIFICADA DE LA IMÁGEN DE CAPA NO BASE
622
620

FIG. 34

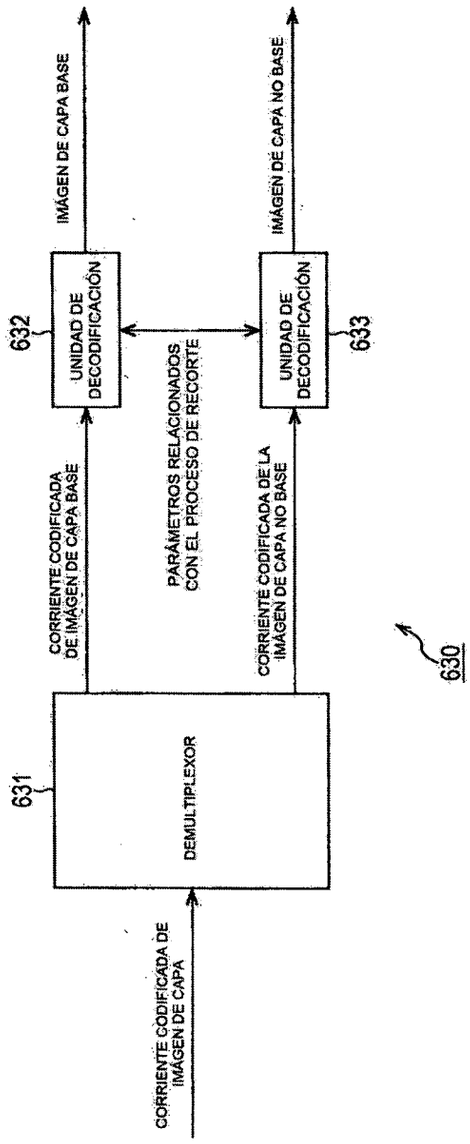

CORRIENTE CODIFICADA DE IMÁGEN DE CAPA
DEMULTIPLEXOR
631
CORRIENTE CODIFICADA DE IMÁGEN DE CAPA BASE
CORRIENTE CODIFICADA DE LA IMÁGEN DE CAPA NO BASE
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
UNIDAD DE DECODIFICACIÓN
632
UNIDAD DE DECODIFICACIÓN
633
IMÁGEN DE CAPA BASE
IMÁGEN DE CAPA NO BASE
630

FIG. 35

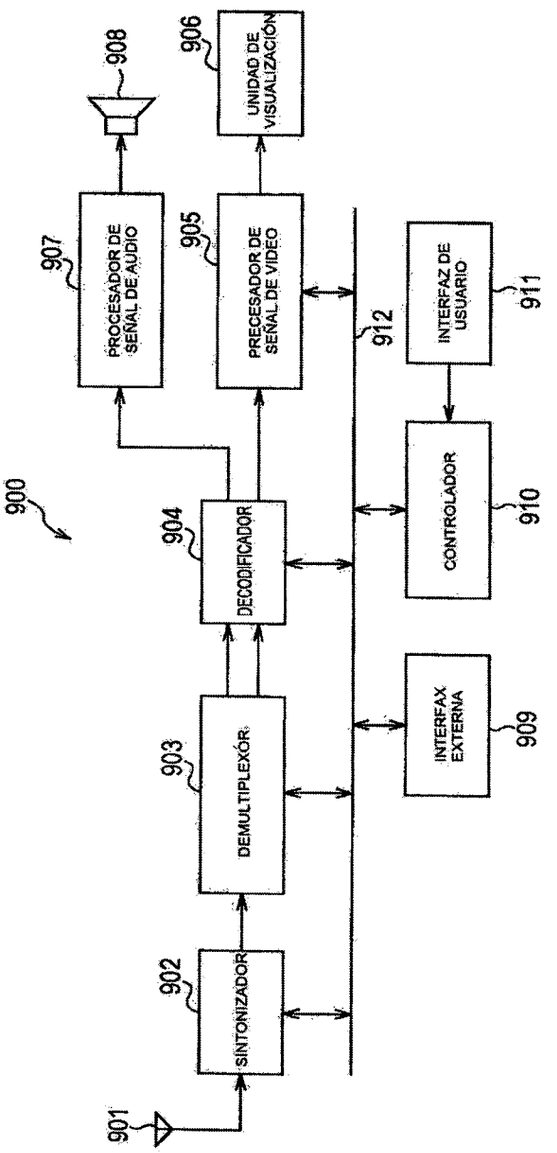

900
901
902
SINTONIZADOR
903
DEMULTIPLEXOR
904
DECODIFICADOR
905
PRECESADOR DE SEÑAL DE VIDEO
906
UNIDAD DE VISUALIZACIÓN
907
PROCESADOR DE SEÑAL DE AUDIO
908
909
INTERFAX EXTERNA
910
CONTROLADOR
911
INTERFAZ DE USUARIO
912

FIG. 36

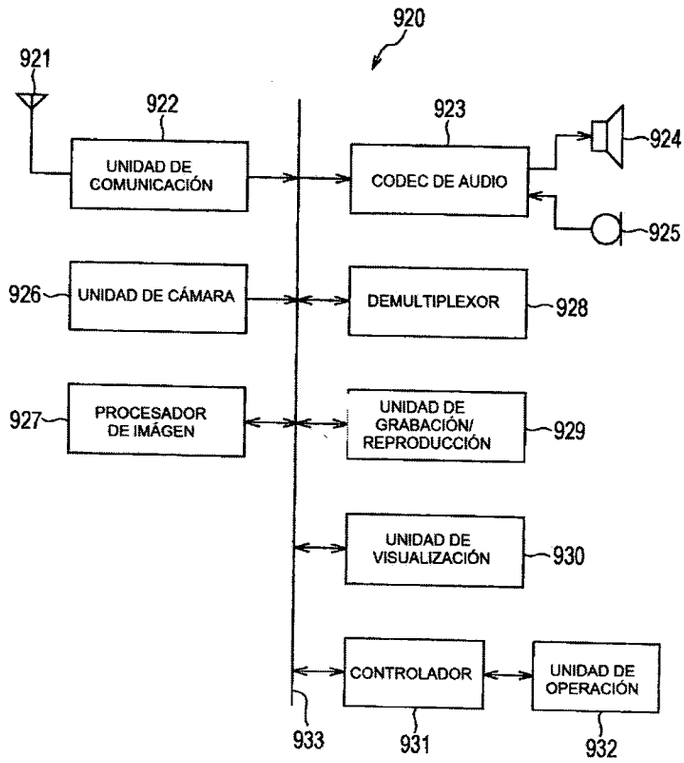

920
921
922
923
924
925
926
927
928
929
930
931
932
933
UNIDAD DE COMUNICACIÓN
CODEC DE AUDIO
UNIDAD DE CÁMARA
DEMULTIPLEXOR
PROCESADOR DE IMÁGEN
UNIDAD DE GRABACIÓN/ REPRODUCCIÓN
UNIDAD DE VISUALIZACIÓN
CONTROLADOR
UNIDAD DE OPERACIÓN

FIG. 37

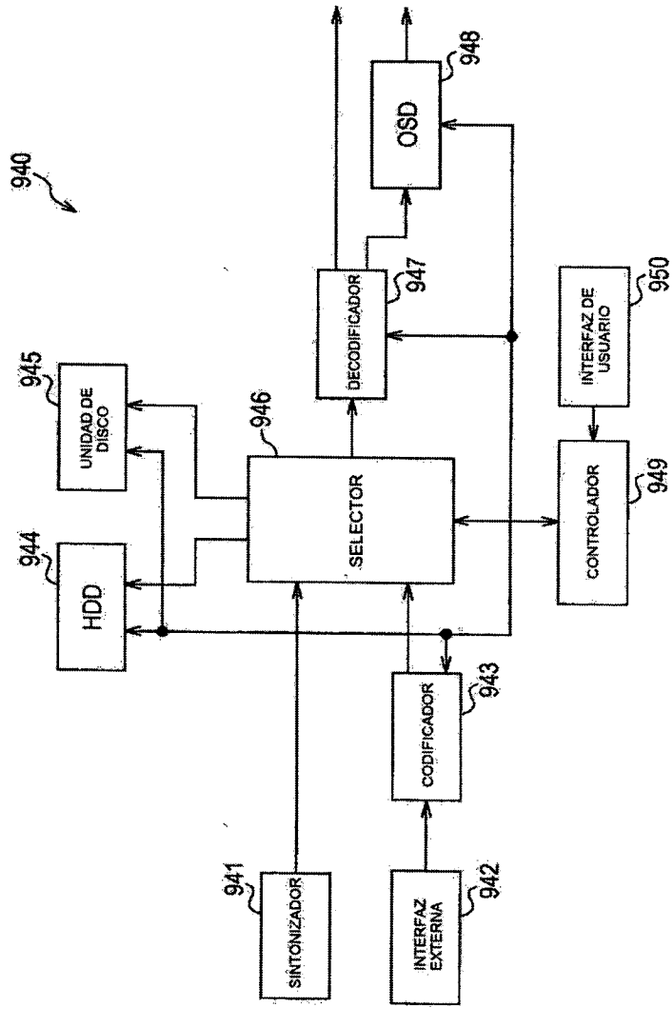
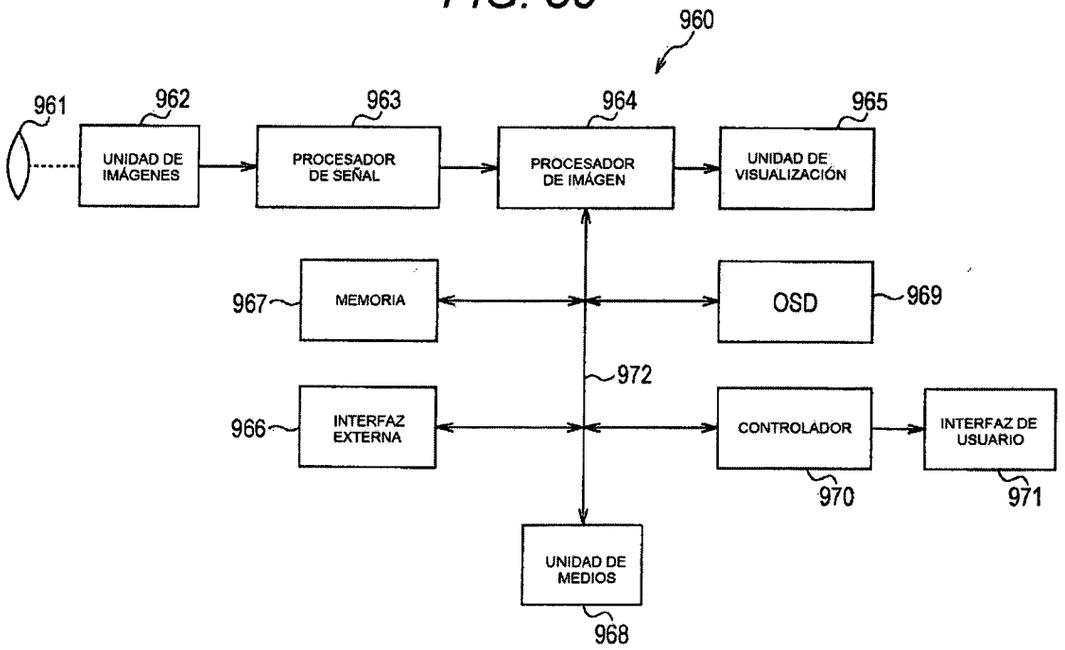

940
941
SINTONIZADOR
942
INTERFAZ EXTERNA
943
CODIFICADOR
944
HDD
945
UNIDAD DE DISCO
946
SELECTOR
947
DECODIFICADOR
948
OSD
949
CONTROLADOR
950
INTERFAZ DE USUARIO

FIG. 38


960
961
962
963
964
965
UNIDAD DE IMÁGENES
PROCESADOR DE SEÑAL
PROCESADOR DE IMÁGEN
UNIDAD DE VISUALIZACIÓN
967
MEMORIA
OSD
969
972
966
INTERFAZ EXTERNA
CONTROLADOR
INTERFAZ DE USUARIO
970
971
UNIDAD DE MEDIOS
968

100

Industria y Comercio
SUPERINTENDENCIA

PLANILLA DE RADICACIÓN - DIGITALIZACIÓN DIARIA
RELACIÓN DE ANEXOS

Dependencia:	Fecha: 13 / 12 / 26 / (AAAA MM DD)	Recorrido:
2020		3

Tipo de Radicación: Entrada: ☒ Salida: ☐ Traslado: ☐ Convenio: ☐

Radicador: | Planilla:

ID	*No. Radicación*	*Consecutivo*	*Item*
1	13-300252	0	7
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			

Item	*Unidad de Conservación*
1	Bolsa
2	Caja
3	Libro
4	Sobre
5	Tomo
6	**Otro**
Item	***Soporte Documental***
☒	CD
8	Disquete
9	DVD
10	Folleto
11	Foto
12	Periódico
13	Plano
14	Publicación Periódica
15	USB
16	**Otro**

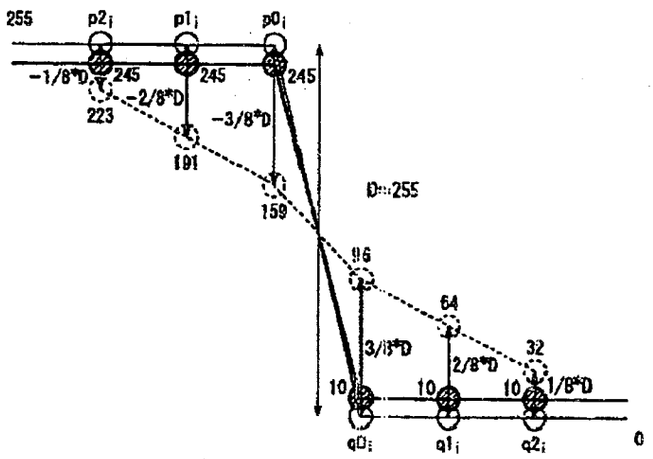
Observaciones:

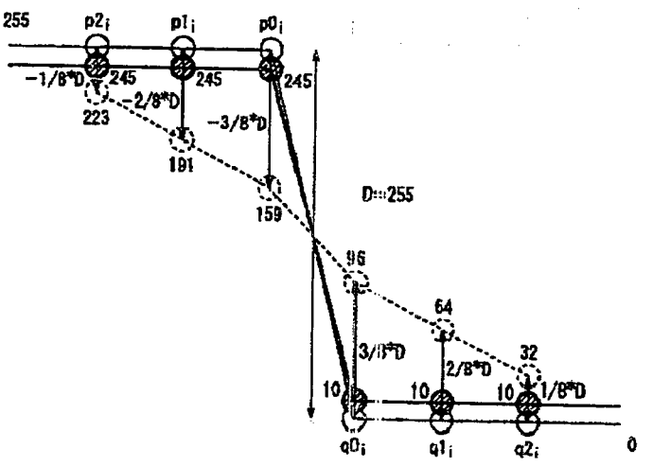
-Folio 100.

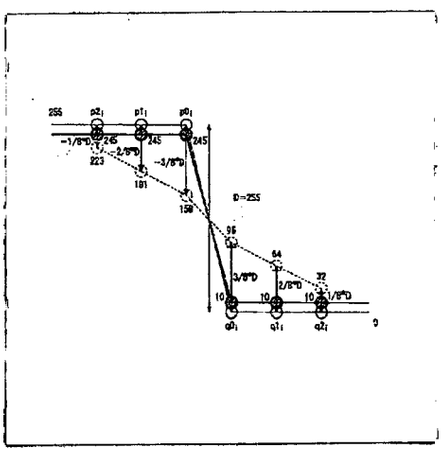
El co Anexo Contiene un total de 2 Archivos, 2 en formato pdf, los cuales fueron procesados.

grabado.

CASTELLANOS & CO LTDA
ABOGADOS
Calle 94 A No. 7 A-09
Bogotá, D.C. – COLOMBIA







Industria y Comercio
SUPERINTENDENCIA

SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
DIVISION DE NUEVAS CREACIONES
TARJETA ARCHIVO PROPIETARIOS

PATENTE DE INVENCION _____ PCT __X__ MODELO DE UTILIDAD _____ DISEÑO INDUSTRIAL _____

(21) No. de solicitud
(22) Fecha de solicitud
(51) Clasificación internacional

(71) Solicitante (s) **SONY CORPORATION**

(72) Inventor (es) **MASARU IKEDA**

(74) Apoderado **DRA. MARGARITA CASTELLANOS ABONDANO**

(30) Prioridad	(31)	No. Prioridad	(32)	Fecha	(33)	País
		2011-143461		2011-06-28		JAPON
		2011-240550		2011-11-01		JAPON
		2011-243839		2011-11-07		JAPON
		2012-009326		2012-01-09		JAPON

(85) Fecha inicio fase nacional:
(86) Datos relativos a la presentación de la solicitud PCT
Fecha de presentación de la solicitud: **28/05/2012**
No. de solicitud: **PCT/JP2012/063606**

(87) Publicación internacional
Fecha: **03/01/2013**
Nº publicación: **WO2013/001957 A1**

(54) Título: **DISPOSITIVO DE PROCESAMIENTO DE IMAGEN Y MÉTODO DE PROCESAMIENTO DE IMAGEN**

Industria y Comercio
SUPERINTENDENCIA

EXTRACTO PARA PUBLICACIÓN

___ Patente de Invención ___ Patente modelo de utilidad _X_ PCT

(21) No. De solicitud	(51) int. Cl:
(22) Fecha de solicitud	(71) Solicitante (s) **SONY CORPORATION**
(30) Prioridad (31) No. Prioridad **2011-143461** **2011-240550** **2011-243839** **2011-009326** (32) Fecha **28/06/2011** **01/11/2011** **07/11/2011** **19/01/2012** (33) País **JAPON**	(72) Interventor (es) **MASARU IKEDA**
	(74) Apoderado (s) **DRA. MARGARITA CASTELLANOS ABONDANO**
(54) Título **DISPOSITIVO DE PROCESAMIENTO DE IMAGEN Y METODO DE PROCESAMIENTO DE IMAGEN**	
Las casillas identificadas con 85, 86 y 87 sólo aplican para PCT	
(85) Fecha límite inicio Fase Nacional (86) Datos relativos a la presentación de la solicitud PCT Fecha de presentación de la solicitud **28/05/2012** No. de solicitud **PCT/JP2012/063606**	(87) Publicación internacional Fecha **03/01/2013** No. Publicación **WO2013/001957 A1**

Resumen reivindicación (es):

1. Un dispositivo de procesamiento de imagen que comprende:
una unidad de decodificación que decodifica una corriente codificada, codificada en unidades que tienen una estructura de capa para generar una imagen;
una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es in límite entre un bloque de la imagen generada por el unidad de decodificación y un bloque adyacente, adyacente al bloque; y
un controlador que controla la unidad de filtración de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

2. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 1, en el que,
el controlador aplica el proceso de recorte a los valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

3. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 2, en el que
el controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones:

[Fórmula Matemática 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

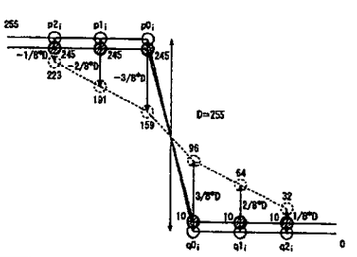
$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0,7 y pv es un valor de recorte

.....



Industria y Comercio SUPERINTENDENCIA	LISTA DE CHEQUEO ADMISIÓN A TRÁMITE – NUEVAS CREACIONES

PATENTE DE INVENCIÓN ____ **MODELO DE UTILIDAD** ____ **Art 33 Decisión 486/00**

Indicación que se solicita una patente.
Datos de identificación del solicitante o de la persona que presenta la solicitud
Descripción de la invención
Dibujos de ser estos pertinentes
Comprobante de pago de las tasas establecidas (De ser el caso formato de descuento)

Completa ____ Incompleta ____

PATENTE DE INVENCIÓN PCT X **MODELO DE UTILIDAD PCT** ___ **Art.33 Decisión 486/00, Circular Única**

X Indicación que se solicita una PCT
X Copia de la solicitud en español, tal como fue presentada inicialmente (capítulo descriptivo, reivindicatorio, resumen)
X Dibujos de ser estos pertinentes
X Comprobante de pago de las tasas establecidas (de ser el caso formato de descuento)

Completa X Incompleta ____

DISEÑO INDUSTRIAL ____ **(Art. 119 Decisión 486/00)**

Indicación que se solicita Diseño industrial
Datos de identificación del solicitante o de la persona que presenta la solicitud
Representación gráfica y fotográfica del Diseño industrial o muestra del material que incorpora el diseño

Comprobante de pago de las tasas establecidas

Completa ____ Incompleta ____

ESQUEMA DE TRAZADO ____ **(Art. 92 Decisión 486/00)**

Indicación que se solicita un esquema de trazado

Datos de identificación del solicitante o de la persona que presenta la solicitud

Representación gráfica de un esquema de trazado

Comprobante de pago de las tasas establecidas

Completa ____ Incompleta ____

SUPERINTENDENCIA DE INDUSTRIA Y COMERCIO
No. 13-300252- -00000-0000
Fecha: 2013-12-26 14:52:55 Dep. 2020 DIR.NUEVASCR
Tra. 11 PATENTEIYII Eve: 378 FASENACIONALI
Act. 411 PRESENTACION Folios: 103

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2013 年 1 月 3 日(03.01.2013)

WIPO | PCT

(10) 国際公開番号
WO 2013/001957 A1

(51) 国際特許分類:
H04N 7/26 (2006.01)

(21) 国際出願番号: PCT/JP2012/063606

(22) 国際出願日: 2012 年 5 月 28 日(28.05.2012)

(25) 国際出願の言語: 日本語

(26) 国際公開の言語: 日本語

(30) 優先権データ:
特願 2011-143461 2011 年 6 月 28 日(28.06.2011) JP
特願 2011-240550 2011 年 11 月 1 日(01.11.2011) JP
特願 2011-243839 2011 年 11 月 7 日(07.11.2011) JP
特願 2012-009326 2012 年 1 月 19 日(19.01.2012) JP

(71) 出願人 (米国を除く全ての指定国について): ソニー株式会社(SONY CORPORATION) [JP/JP]; 〒1080075 東京都港区港南 1 丁目 7 番 1 号 Tokyo (JP).

(72) 発明者; および

(75) 発明者/出願人 (米国についてのみ): 池田 優(IKEDA Masaru) [JP/JP]; 〒1080075 東京都港区港南 1 丁目 7 番 1 号 ソニー株式会社内 Tokyo (JP).

(74) 代理人: 稲本 義雄, 外(INAMOTO Yoshio et al.); 〒1600023 東京都新宿区西新宿 7 丁目 5 番 25 号 西新宿木村屋ビルディング 9 階 Tokyo (JP).

(81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

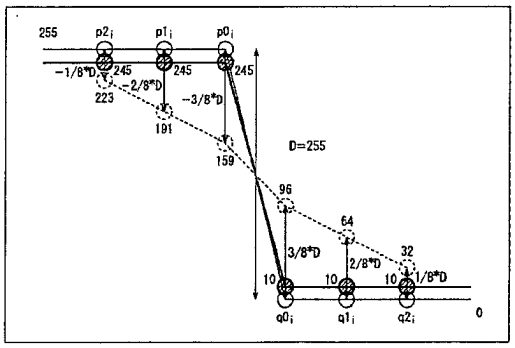
添付公開書類:

— 国際調査報告 (条約第 21 条(3))

(54) **Title**: IMAGE PROCESSING DEVICE AND IMAGE PROCESSING METHOD

(54) 発明の名称 : 画像処理装置と画像処理方法

図21



(57) **Abstract**: The present technology relates to an image processing device and an image processing method by which filtering can be appropriately performed in a deblocking filtering process. The value of a pixel ($p0_i$) which is 255 (solid line) prior to a deblocking process changes greatly after a conventional deblocking process, becoming 159 (dotted line). Therefore, a clipping process with a clipping value of 10 is performed in strong filtering, and thereby, the value of the pixel ($p0_i$) which is 255 (solid line) prior to the deblocking process becomes 245 (thick line), and the conventional large change in pixel values can be reduced to a minimum. The disclosure can be applied to, for example, an image processing device.

(57) 要約:

[続葉有]

WO 2013/001957 A1

本技術は、デブロッキングフィルタ処理において、適切にフィルタリングがかかるようにする画像処理装置と画像処理方法に関する。デブロッキング処理前には、２５５（実線）である画素ｐ０ｉの値は、従来のデブロッキング処理後に１５９（点線）となり、大きく変化してしまう。そこで、強フィルタリングにおいてクリップ値が１０のクリップ処理を行うことにより、デブロッキング処理前には、２５５（実線）である画素ｐ０ｉの値は、２４５（太線）となり、従来において生じていた画素値の変化を最小限に抑えることができる。本開示は、例えば、画像処理装置に適用することができる。

DESCRIPTION

IMAGE PROCESSING DEVICE AND IMAGE PROCESSING METHOD

TECHNICAL FIELD

[0001]

This technique relates to an image processing device and an image processing method. Specifically, this technique allows a deblocking filtering process to apply filtering appropriately.

BACKGROUND ART

[0002]

In recent years, devices, which handle image information as digital data, which, in such a case, aim to transmit and store information with a high efficiency, and which conform to a scheme, such as MPEG2 (International Organization for Standardization and International Electrotechnical Commission (ISO/IEC) 13818-2), for compressing image information using orthogonal transformation, such as discrete cosine transformation, and using motion compensation by utilizing redundancy that is unique to the image information have become widespread in both information distribution in broadcasting stations and information reception in ordinary homes. In addition, schemes called H.264 and MPEG4 Part10 (Advanced Video Coding (AVC) which require a larger amount of operation for coding and decoding but can realize a higher coding efficiency than MPEG2 or the like have been also used. Further, nowadays, standardization works for high efficiency video coding (HEVC) which is a next-generation image coding scheme are

in progress so that compression, distribution, or the like of high-resolution images of 4000 × 2000 pixels which is four times that of high-vision images can be performed efficiently.

[0003]

In the standardization works for high efficiency video coding (HEVC) which is a next-generation image coding scheme, JCTVC-A119 (see Non-Patent Document 1 below) proposes to apply a deblocking filter to each block having the size of 8 × 8 pixels or larger. In the method proposed in JCTVC-A119, by increasing a minimum unit block size to which a deblocking filter is to be applied, it is possible to execute a filtering process in parallel on a plurality of block boundaries in the same direction within one macroblock.

CITATION LIST

NON-PATENT DOCUMENT

[0004]

Non-Patent Document 1: K. Ugur (Nokia), K. R. Andersson (LM Ericsson), A. Fuldseth (Tandberg Telecom), "JCTVC-A119: Video coding technology proposal by Tandberg, Nokia, and Ericsson," Documents of the first meeting of the Joint Collaborative Team on Video Coding (JCT-VC), Dresden, Germany, 15-23 April, 2010

SUMMARY OF THE INVENTION

PROBLEMS TO BE SOLVED BY THE INVENTION

[0005]

However, in the conventional deblocking filtering process, filtering is not performed appropriately.

[0006]

Therefore, an object of this technique is to allow a deblocking filtering process to apply filtering appropriately.

SOLUTIONS TO PROBLEMS

[0007]

An image processing device according to a first aspect of this technique includes: a decoding unit that decodes an encoded stream encoded in units each having a layer structure to generate an image; a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the image generated by the decoding unit and an adjacent block adjacent to the block; and a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the image generated by the decoding unit.

[0008]

The controller may apply the clipping process to values of parts that change when the deblocking filter is applied.

[0009]

The controller applies the clipping process according to the following expressions:

[Mathematical formula 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

[0010]

The controller may set a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

[0011]

The controller may set a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

[0012]

The image processing device further includes: a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, and the filtering unit may apply the deblocking filter to the block boundary according to the strength determined by the filter strength determining unit, and the controller may control the filtering unit so that, when the filter strength determining unit determines that strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the

decoding unit.

[0013]

The filter strength determining unit may determine the strength of the deblocking filter using a plurality of lines as processing units.

[0014]

The filter strength determining unit may determine the strength of the deblocking filter using four lines as the processing units.

[0015]

The image processing device further includes: a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, and the filter strength determining unit may determine the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

[0016]

An image processing method according to the first aspect of this technique allows an image processing device to execute: decoding an encoded stream encoded in units each having a layer structure to generate an image; applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the generated image and an adjacent block adjacent to the block; and controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the generated image.

[0017]

An image processing device according to a second aspect of this technique includes: a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block; a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and an encoding unit that encodes the image in units each having a layer structure using an image to which the deblocking filter is applied.

[0018]

The controller may apply the clipping process to values of parts that change when the deblocking filter is applied.

[0019]

The controller applies the clipping process according to the following expressions:

[Mathematical formula 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

[0020]

The controller may set a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

[0021]

The controller may set a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

[0022]

The image processing device further includes: a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, and the filtering unit may apply the deblocking filter to the block boundary according to the strength determined by the filter strength determining unit, and the controller may control the filtering unit so that, when the filter strength determining unit determines that

strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the decoding unit.

[0023]

The filter strength determining unit may determine the strength of the deblocking filter using a plurality of lines as processing units.

[0024]

The filter strength determining unit may determine the strength of the deblocking filter using four lines as the processing units.

[0025]

The image processing device further includes: a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, and the filter strength determining unit may determine the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

[0026]

An image processing method according to the second aspect of this technique allows an image processing device to execute: applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block; controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process

is applied to the deblocking filter with respect to luminance components of the locally decoded image; and encoding the image in units each having a layer structure using an image to which the deblocking filter is applied.

[0027]

In a first aspect of the present technique, an encoded stream encoded in units each having a layer structure is decoded to generate an image; and a deblocking filter is applied to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the generated image and an adjacent block adjacent to the block. Moreover, control is performed so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the generated image.

[0028]

In a second aspect of the present technique, a deblocking filter is applied to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block. Moreover, control is performed so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and the image is encoded in units each having a layer structure using an image to which the deblocking filter is applied.

EFFECTS OF THE INVENTION

[0029]

According to this technique, it is possible to allow a deblocking filtering process to apply filtering appropriately.

BRIEF DESCRIPTION OF THE DRAWINGS

[0030]

Figs. 1(A) and 1(B) are diagrams for describing a conventional deblocking filtering process.

Fig. 2 is a diagram illustrating a configuration when applied to an image encoding device.

Fig. 3 is a flowchart illustrating an image encoding operation.

Fig. 4 is a flowchart illustrating an intra-prediction process.

Fig. 5 is a flowchart illustrating an inter-prediction process.

Fig. 6 is a diagram illustrating a configuration when applied to an image decoding device.

Fig. 7 is a flowchart illustrating an image decoding operation.

Fig. 8 is a diagram for describing a basic operation of a deblocking filtering unit.

Fig. 9 is a diagram illustrating the configuration of a first embodiment of the deblocking filtering unit.

Fig. 10 is a flowchart illustrating the operation of the first embodiment of the deblocking filtering unit.

Fig. 11 is a diagram illustrating the configuration of a filter operation unit.

Fig. 12 is a diagram illustrating the configuration of the filter operation unit.

Figs. 13(A) and 13(B) are diagrams for describing the operation of the filter operation unit.

Figs. 14(A) and 14(B) are diagrams for describing the operation of the filter operation unit.

Fig. 15 is a flowchart illustrating the operation of a fifth embodiment.

Fig. 16 is a diagram for describing a conventional deblocking filtering process.

Fig. 17 is a diagram for describing the present technique (sixth embodiment).

Fig. 18 is a diagram for describing the present technique (sixth embodiment).

Fig. 19 is a diagram illustrating pixels used in the deblocking filtering process of the present technique.

Fig. 20 is a diagram for describing pixel values that change due to conventional strong filtering.

Fig. 21 is a diagram for describing the effects of clipping-based strong filtering.

Fig. 22 is a diagram illustrating the configuration of a sixth embodiment of the deblocking filtering unit.

Fig. 23 is a flowchart illustrating the operation of the sixth embodiment of the deblocking filtering unit.

Fig. 24 is a diagram illustrating the configuration of a seventh embodiment of the deblocking filtering unit.

Fig. 25 is a diagram illustrating the configuration of an eighth embodiment of the deblocking filtering unit.

Fig. 26 is a flowchart illustrating the operation of the eighth embodiment of the deblocking filtering unit.

Fig. 27 is a flowchart for describing a luminance

signal deblocking process.

Fig. 28 is a diagram for describing an example of the R3W2 case.

Fig. 29 is a diagram illustrating an example of a multi-view image encoding scheme.

Fig. 30 is a diagram illustrating an example of main components of a multi-view image encoding device to which the present technique is applied.

Fig. 31 is a diagram illustrating an example of main components of a multi-view image decoding device to which the present technique is applied.

Fig. 32 is a diagram illustrating an example of a layer image encoding scheme.

Fig. 33 is a diagram illustrating an example of main components of a layer image encoding device to which the present technique is applied.

Fig. 34 is a diagram illustrating an example of main components of a layer image decoding device to which the present technique is applied.

Fig. 35 is a diagram illustrating an example of a schematic configuration of a television apparatus.

Fig. 36 is a diagram illustrating an example of a schematic configuration of a mobile phone.

Fig. 37 is a diagram illustrating an example of a schematic configuration of a recording/reproducing device.

Fig. 38 is a diagram illustrating an example of a schematic configuration of an imaging device.

MODE FOR CARRYING OUT THE INVENTION

[0031]

Hereinafter, modes for carrying out the present

technique will be described. The description will be given in the following order:

1. Conventional art
2. Configuration when applied to image encoding device
3. Operation of image encoding device
4. Configuration when applied to image decoding device
5. Operation of image decoding device
6. Basic operation of deblocking filtering unit
7. First embodiment of deblocking filtering unit
8. Second embodiment of deblocking filtering unit
9. Third embodiment of deblocking filtering unit
10. Fourth embodiment of deblocking filtering unit
11. Fifth embodiment of deblocking filtering unit
12. Description of sixth to eighth embodiments
13. Sixth embodiment of deblocking filtering unit
14. Seventh embodiment of deblocking filtering unit
15. Eighth embodiment of deblocking filtering unit
16. Ninth embodiment
17. Tenth embodiment (multi-view image encoding and decoding device)
18. Eleventh embodiment (layer image encoding and decoding device)
19. Application example

[0032]

[1. Conventional art]

A conventional deblocking filtering process will be described with reference to Figs. 1(A) and 1(B).

[0033]

As illustrated in Fig. 1(A), for example, when a

deblocking filtering process is performed in a raster order in respective largest coding units (LCUs), image data corresponding to a predetermined number of lines from an inter-block boundary BB, of LCUu which is an upper block is stored in a line memory, and a vertical filtering process is performed using the image data and image data of LCUl which is a lower block obtained after the storing. For example, as illustrated in Fig. 1(B), when filter operation is performed using image data corresponding to four lines from the boundary BB while using image data of the upper and lower blocks each corresponding to three lines from the inter-block boundary BB as a processing range of the vertical filtering process, image data corresponding to four lines from the boundary BB, of LCUu which is the upper block is stored in the line memory. In the drawings, target pixels of the deblocking filter are depicted by double circles, and the upper and lower boundaries of the filter processing range of the deblocking filter are indicated by "DBU" and "DBL," respectively.

[0034]

In this manner, since image data corresponding to a predetermined number of lines from the inter-block boundary BB is stored in the line memory so as to be used for the filter operation, if the number of pixels in the horizontal direction increases, the memory capacity of the line memory increases.

[0035]

[2. Configuration when applied to image encoding device]

Fig. 2 illustrates a configuration when the image processing device of the present technique is applied to

an image encoding device. An image encoding device 10 includes an analog/digital converter (A/D converter) 11, a frame reordering buffer 12, a subtractor 13, an orthogonal transformer 14, a quantizer 15, a lossless encoder 16, an accumulation buffer 17, and a rate controller 18. The image encoding device 10 further includes an inverse quantizer 21, an inverse orthogonal transformer 22, an adder 23, a deblocking filtering unit 24, a frame memory 25, a selector 26, an intra predictor 31, a motion estimator/compensator 32, and a predicted image/optimal mode selecting unit 33.

[0036]

The A/D converter 11 converts an analog image signal into digital image data and outputs the digital image data to the frame reordering buffer 12.

[0037]

The frame reordering buffer 12 reorders the frames of the image data output from the A/D converter 11. The frame reordering buffer 12 performs the frame reorder according to a GOP (group of pictures) structure related to an encoding process and outputs the reorderd image data to the subtractor 13, the intra predictor 31, and the motion estimator/compensator 32.

[0038]

The subtractor 13 is supplied with the image data output from the frame reordering buffer 12 and predicted image data selected by the predicted image/optimal mode selecting unit 33 described later. The subtractor 13 calculates prediction error data which is a difference between the image data output from the frame reordering buffer 12 and the predicted image data supplied from the

predicted image/optimal mode selecting unit 33 and outputs the prediction error data to the orthogonal transformer 14.

[0039]

The orthogonal transformer 14 performs an orthogonal transform process such as discrete cosine transform (DCT) or Karhunen-Loeve transform with respect to the prediction error data output from the subtractor 13. The orthogonal transformer 14 outputs transform coefficient data obtained by performing the orthogonal transform process to the quantizer 15.

[0040]

The quantizer 15 is supplied with the transform coefficient data output from the orthogonal transformer 14 and a rate control signal from the rate controller 18 described later. The quantizer 15 quantizes the transform coefficient data and outputs the quantized data to the lossless encoder 16 and the inverse quantizer 21. Moreover, the quantizer 15 switches a quantization parameter (quantization scale) based on the rate control signal from the rate controller 18 to change a bit rate of the quantized data.

[0041]

The lossless encoder 16 is supplied with the quantized data output from the quantizer 15 and prediction mode information from the intra predictor 31, the motion estimator/compensator 32, and the predicted image/optimal mode selecting unit 33 described later. The prediction mode information includes a macroblock type that allows a predicted block size to be identified, a prediction mode, motion vector information, reference

picture information, and the like according to intra-prediction or inter-prediction. The lossless encoder 16 performs a lossless encoding process such as, for example, variable-length encoding or arithmetic encoding on the quantized data to generate an encoded stream and outputs the encoded stream to the accumulation buffer 17. Moreover, the lossless encoder 16 losslessly encodes the prediction mode information to add the encoded prediction mode information to the header information of the encoded stream.

[0042]

The accumulation buffer 17 accumulates the encoded streams from the lossless encoder 16. Moreover, the accumulation buffer 17 outputs the accumulated encoded streams at a transfer rate corresponding to a transmission line.

[0043]

The rate controller 18 monitors a free space of the accumulation buffer 17, generates a rate control signal according to the free space, and outputs the rate control signal to the quantizer 15. The rate controller 18 acquires information indicating the free space from the accumulation buffer 17, for example. The rate controller 18 decreases the bit rate of the quantized data using the rate control signal when the free space decreases. Moreover, the rate controller 18 increases the bit rate of the quantized data using the rate control signal when the free space of the accumulation buffer 17 is sufficiently large.

[0044]

The inverse quantizer 21 performs an inverse

quantization process on the quantized data supplied from the quantizer 15. The inverse quantizer 21 outputs transform coefficient data obtained by performing the inverse quantization process to the inverse orthogonal transformer 22.

[0045]

The inverse orthogonal transformer 22 outputs data obtained by performing an orthogonal transform process on the transform coefficient data supplied from the inverse quantizer 21 to the adder 23.

[0046]

The adder 23 adds the data supplied from the inverse orthogonal transformer 22 and the predicted image data supplied from the predicted image/optimal mode selecting unit 33 to generate decoded image data and outputs the decoded image data to the deblocking filtering unit 24 and the frame memory 25.

[0047]

The deblocking filtering unit 24 performs a filtering process for reducing block distortion occurring during image encoding. The deblocking filtering unit 24 performs a filtering process to remove block distortion from the decoded image data supplied from the adder 23 and outputs the image data having been subjected to the filtering process to the frame memory 25. Moreover, the deblocking filtering unit 24 controls an image range used for a filter operation in a block positioned on the upper side of a boundary according to a boundary detected by performing inter-block boundary detection in a vertical direction. In this manner, by controlling the image range used for filter operation, the deblocking filtering

unit 24 allows a vertical filtering process to be performed even when a memory capacity of the line memory that stores image data is reduced. The details thereof will be described later.

[0048]

The frame memory 25 maintains the decoded image data supplied from the adder 23 and the decoded image data having been subjected to the filtering process, supplied from the deblocking filtering unit 24 as image data of a reference image.

[0049]

The selector 26 supplies the reference image data before the filtering process, read from the frame memory 25 to the intra predictor 31 in order to perform intra-prediction. Moreover, the selector 26 supplies the reference image data having been subjected to the filtering process, read from the frame memory 25 to the motion estimator/compensator 32 in order to perform inter-prediction.

[0050]

The intra predictor 31 performs an intra-prediction process in all candidate intra-prediction modes using the image data of an encoding target image output from the frame reordering buffer 12 and the reference image data before the filtering process, read from the frame memory 25. Further, the intra predictor 31 calculates a cost function value for each intra-prediction mode and selects an intra-prediction mode in which the calculated cost function value is the smallest (that is, an intra-prediction mode in which best encoding efficiency is obtained) as an optimal intra-prediction mode. The intra

predictor 31 outputs predicted image data generated in the optimal intra-prediction mode, the prediction mode information on the optimal intra-prediction mode, and a cost function value in the optimal intra-prediction mode to the predicted image/optimal mode selecting unit 33. Moreover, the intra predictor 31 outputs the prediction mode information on the intra-prediction mode in the intra-prediction process in the respective intra-prediction modes to the lossless encoder 16 in order to obtain a generated code amount used in calculation of the cost function value as described later.

[0051]

The motion estimator/compensator 32 performs a motion estimation/compensation process in all predicted block sizes corresponding to macroblocks. The motion estimator/compensator 32 detects a motion vector using the reference image data having been subjected to the filtering process, read from the frame memory 25 with respect to each of the encoding target images of the respective predicted block sizes, read from the frame reordering buffer 12. Further, the motion estimator/compensator 32 performs a motion compensation process on the decoded image based on the detected motion vector to generate a predicted image. Moreover, the motion estimator/compensator 32 calculates a cost function value for each predicted block size and selects a predicted block size in which the calculated cost function value is the smallest (that is, a predicted block size in which best encoding efficiency is obtained) as an optimal inter-prediction mode. The motion estimator/compensator 32 outputs the predicted image data

generated in the optimal inter-prediction mode, the prediction mode information on the optimal inter-prediction mode, and the cost function value in the optimal inter-prediction mode to the predicted image/optimal mode selecting unit 33. Moreover, the motion estimator/compensator 32 outputs the prediction mode information on the inter-prediction mode in the inter-prediction process in the respective predicted block sizes to the lossless encoder 16 in order to obtain a generated code amount used in calculation of the cost function value. The motion estimator/compensator 32 also performs the prediction in a skipped macroblock mode and a direct mode as the inter-prediction mode.

[0052]

The predicted image/optimal mode selecting unit 33 compares the cost function value supplied from the intra predictor 31 and the cost function value supplied from the motion estimator/compensator 32 in respective macroblock units and selects a mode in which the cost function value is the smaller as an optimal mode in which best encoding efficiency is obtained. Moreover, the predicted image/optimal mode selecting unit 33 outputs predicted image data generated in the optimal mode to the subtractor 13 and the adder 23. Further, the predicted image/optimal mode selecting unit 33 outputs the prediction mode information of the optimal mode to the lossless encoder 16. The predicted image/optimal mode selecting unit 33 may perform the intra-prediction or the inter-prediction in respective slice units.

[0053]

An encoding unit in the claims includes the intra

predictor 31 that generates predicted image data, the motion estimator/compensator 32, the predicted image/optimal mode selecting unit 33, the subtractor 13, the orthogonal transformer 14, the quantizer 15, the lossless encoder 16, and the like.

[0054]

[3. Operation of image encoding device]

Fig. 3 is a flowchart illustrating the operation of an image encoding process. In step ST11, the A/D converter 11 performs A/D conversion on an input image signal.

[0055]

In step ST12, the frame reordering buffer 12 performs screen reorder. The frame reordering buffer 12 stores the image data supplied from the A/D converter 11 and reorders the respective pictures so that the pictures arranged in the display order are reorderd in the encoding order.

[0056]

In step ST13, the subtractor 13 generates prediction error data. The subtractor 13 calculates a difference between the image data of the images reorderd in step ST12 and the predicted image data selected by the predicted image/optimal mode selecting unit 33 to generate prediction error data. The prediction error data has a data amount that is smaller than that of the original image data. Thus, it is possible to compress the data amount as compared to when the image is encoded as it is. When the predicted image/optimal mode selecting unit 33 selects the predicted image supplied from the intra predictor 31 and the predicted image from

the motion estimator/compensator 32 in respective slice units, the intra-prediction is performed in the slice in which the predicted image supplied from the intra predictor 31 is selected. Moreover, the inter-prediction is performed in the slice in which the predicted image from the motion estimator/compensator 32 is selected.

[0057]

In step ST14, the orthogonal transformer 14 performs an orthogonal transform process. The orthogonal transformer 14 performs orthogonal transform on the prediction error data supplied from the subtractor 13. Specifically, orthogonal transform such as discrete cosine transform or Karhunen-Loeve transform is performed on the prediction error data, and transform coefficient data is output.

[0058]

In step ST15, the quantizer 15 performs a quantization process. The quantizer 15 quantizes the transform coefficient data. When the quantization is performed, rate control is performed as will be described in the process of step ST25.

[0059]

In step ST16, the inverse quantizer 21 performs an inverse quantization process. The inverse quantizer 21 performs inverse quantization on the transform coefficient data quantized by the quantizer 15 according to a property corresponding to the property of the quantizer 15.

[0060]

In step ST17, the inverse orthogonal transformer 22 performs an orthogonal transform process. The inverse

orthogonal transformer 22 performs inverse orthogonal transform on the transform coefficient data inverse-quantized by the inverse quantizer 21 according to a property corresponding to the property of the orthogonal transformer 14.

[0061]

In step ST18, the adder 23 generates decoded image data. The adder 23 adds the predicted image data supplied from the predicted image/optimal mode selecting unit 33 and the data after the inverse orthogonal transform at the position corresponding to the predicted image to generate decoded image data.

[0062]

In step ST19, the deblocking filtering unit 24 performs a deblocking filtering process. The deblocking filtering unit 24 performs filtering on the decoded image data output from the adder 23 to remove block distortion. Moreover, the deblocking filtering unit 24 is configured to perform a vertical filtering process even when the memory capacity of the line memory that stores the image data is reduced. Specifically, the deblocking filtering unit 24 controls the image range used for a filter operation in a block positioned on the upper side of a boundary according to a boundary detected by performing inter-block boundary detection in the vertical direction.

[0063]

In step ST20, the frame memory 25 stores decoded image data. The frame memory 25 stores the decoded image data before the deblocking filtering process.

[0064]

In step ST21, the intra predictor 31 and the motion

estimator/compensator 32 perform prediction processes. That is, the intra predictor 31 performs an intra-prediction process in the intra-prediction mode and the motion estimator/compensator 32 performs a motion estimation/compensation process in the inter-prediction mode. By this process, the prediction processes are performed in all candidate prediction modes, and the cost function values in all candidate prediction modes are calculated. The optimal intra-prediction mode and the optimal inter-prediction mode are selected based on the calculated cost function values, and the predicted image generated in the selected prediction mode and the cost function and the prediction mode information are supplied to the predicted image/optimal mode selecting unit 33.

[0065]

In step ST22, the predicted image/optimal mode selecting unit 33 selects predicted image data. The predicted image/optimal mode selecting unit 33 determines an optimal mode in which best encoding efficiency is obtained based on the respective cost function values output from the intra predictor 31 and the motion estimator/compensator 32. Further, the predicted image/optimal mode selecting unit 33 selects the predicted image data in the determined optimal mode and supplies the predicted image data to the subtractor 13 and the adder 23. The predicted image is used for the operation in steps ST13 and ST18 as described above.

[0066]

In step ST23, the lossless encoder 16 performs a lossless encoding process. The lossless encoder 16 performs lossless encoding on the quantized data output

from the quantizer 15. That is, lossless encoding such as variable-length encoding or arithmetic encoding is performed on the quantized data whereby the quantized data is compressed. In this case, in step ST22 described above, the prediction mode information (for example, including the macroblock type, the prediction mode, the motion vector information, the reference picture information, and the like) input to the lossless encoder 16 is also subjected to lossless encoding. Further, lossless encoding data of the prediction mode information is added to the header information of the encoded stream generated by losslessly encoding the quantized data.

[0067]

In step ST24, the accumulation buffer 17 performs an accumulation process to accumulate the encoded stream. The encoded stream accumulated in the accumulation buffer 17 is appropriately read and is transmitted to a decoding side via a transmission line.

[0068]

In step ST25, the rate controller 18 performs rate control. The rate controller 18 controls the rate of the quantization operation of the quantizer 15 so that an overflow or an underflow does not occur in the accumulation buffer 17 when the accumulation buffer 17 accumulates the encoded stream.

[0069]

Next, the prediction process in step ST21 of Fig. 3 will be described. The intra predictor 31 performs an intra-prediction process. The intra predictor 31 performs intra-prediction on the image of a current block in all candidate intra-prediction modes. The image data

of a reference image referenced in the intra-prediction is not subjected to the filtering process of the deblocking filtering unit 24, but the reference image data stored in the frame memory 25 is used. Although the details of the intra-prediction process are described later, by this process, intra-prediction is performed in all candidate intra-prediction modes, and the cost function values in all candidate intra-prediction modes are calculated. Moreover, one intra-prediction mode in which best encoding efficiency is obtained is selected from all intra-prediction modes based on the calculated cost function values. The motion estimator/compensator 32 performs an inter-prediction process. The motion estimator/compensator 32 performs an inter-prediction process in all candidate inter-prediction modes (all predicted block sizes) using the reference image data having been subjected to the filtering process, stored in the frame memory 25. Although the details of the inter-prediction process are described later, by this process, the prediction process is performed in all candidate inter-prediction modes, and the cost function values in all candidate inter-prediction modes are calculated. Moreover, one inter-prediction mode in which best encoding efficiency is obtained is selected from all inter-prediction modes based on the calculated cost function values.

[0070]

The intra-prediction process will be described with reference to the flowchart of Fig. 4. In step ST31, the intra predictor 31 performs intra-prediction in the respective prediction modes. The intra predictor 31

generates predicted image data in each intra-prediction mode using the decoded image data before the filtering process, stored in the frame memory 25.

[0071]

In step ST32, the intra predictor 31 calculates the cost function values of the respective prediction modes. For example, the lossless encoding process is performed tentatively in all candidate prediction modes, the cost function value expressed by Expression (1) below is calculated in the respective prediction modes.

[0072]

$$Cost(Mode \in \Omega) = D + \lambda/R \cdots (1)$$

[0073]

Here, "Ω" represents a universal set of candidate prediction modes for encoding blocks and macroblocks. "D" represents differential energy (distortion) between a decoded image and an input image when encoding is performed in a prediction mode. "R" represents a generated code amount including orthogonal transform coefficients, prediction mode information, and the like, and "λ" represents the Lagrange multiplier given as a function of a quantization parameter QP.

[0074]

Moreover, with respect to all candidate prediction modes, the predicted images are generated and header bits such as motion vector information and prediction mode information are calculated, and the cost function value expressed by Expression (2) below is calculated in the respective prediction modes.

[0075]

$$Cost(Mode \in \Omega) = D + QPtoQuant(QP) \cdot Header\_Bit \cdots (2)$$

[0076]

Here, "Ω" represents a universal set of candidate prediction modes for encoding blocks and macroblocks. "D" represents differential energy (distortion) between a decoded image and an input image when encoding is performed in a prediction mode. "Header_Bit" is a header bit of a prediction mode, and "QPtoQuant" is a function given as a function of a quantization parameter QP.

[0077]

In step ST33, the intra predictor 31 determines an optimal intra-prediction mode. The intra predictor 31 selects one intra-prediction mode in which the calculated cost function value is the smallest based on the cost function values calculated in step ST32 and determines the intra-prediction mode as the optimal intra-prediction mode.

[0078]

Next, the inter-prediction process will be described with reference to the flowchart of Fig. 5. In step ST41, the motion estimator/compensator 32 determines a motion vector and a reference image in the respective prediction modes. That is, the motion estimator/compensator 32 determines the motion vector and the reference image of a current block of the respective prediction modes.

[0079]

In step ST42, the motion estimator/compensator 32 performs motion compensation in the respective prediction modes. The motion estimator/compensator 32 performs motion compensation on the reference image based on the motion vector determined in step ST41 in the respective

prediction modes (respective predicted block sizes) and generates predicted image data in the respective prediction modes.

[0080]

In step ST43, the motion estimator/compensator 32 generates motion vector information in the respective prediction modes. The motion estimator/compensator 32 generates motion vector information included in the encoded stream with respect to the motion vectors determined in the respective prediction modes. For example, a predicted motion vector is determined using median prediction and motion vector information indicating a difference between the motion vector detected by motion estimation and the predicted motion vector is generated. The motion vector information generated in this manner is also used in calculation of the cost function value in the next step ST44, and when the corresponding predicted image is ultimately selected by the predicted image/optimal mode selecting unit 33, the motion vector information is included in the prediction mode information and output to the lossless encoder 16.

[0081]

In step ST44, the motion estimator/compensator 32 calculates the cost function value in the respective inter-prediction modes. The motion estimator/compensator 32 calculates the cost function value using Expression (1) or (2) described above.

[0082]

In step ST45, the motion estimator/compensator 32 determines an optimal inter-prediction mode. The motion

estimator/compensator 32 selects one prediction mode in which the calculated cost function value is the smallest based on the cost function values calculated in step ST44.

[0083]

[4. Configuration when applied to image decoding device]

The encoded stream generated by encoding an input image is supplied to an image decoding device via a predetermined transmission line, a recording medium, or the like and is decoded.

[0084]

Fig. 6 illustrates the configuration of an image decoding device. An image decoding device 50 includes an accumulation buffer 51, a lossless decoder 52, an inverse quantizer 53, an inverse orthogonal transformer 54, an adder 55, a deblocking filtering unit 56, a frame reordering buffer 57, and a D/A converter 58. The image decoding device 50 further includes a frame memory 61, selectors 62 and 65, an intra predictor 63, and a motion compensation unit 64.

[0085]

The accumulation buffer 51 accumulates encoded streams transmitted thereto. The lossless decoder 52 decodes the encoded streams supplied from the accumulation buffer 51 according to a scheme corresponding to the encoding scheme of the lossless encoder 16 of Fig. 2. Moreover, the lossless decoder 52 outputs prediction mode information obtained by decoding the header information of the encoded streams to the intra predictor 63 and the motion compensation unit 64.

[0086]

The inverse quantizer 53 performs inverse

quantization on the quantized data decoded by the lossless decoder 52 according to a scheme corresponding to the quantization scheme of the quantizer 15 of Fig. 2. The inverse orthogonal transformer 54 performs inverse orthogonal transform on the output of the inverse quantizer 53 according to a scheme corresponding to the orthogonal transform scheme of the orthogonal transformer 14 of Fig. 2 and outputs the output to the adder 55.

[0087]

The adder 55 adds the data after the inverse orthogonal transform and the predicted image data supplied from the selector 65 to generate decoded image data and outputs the decoded image data to the deblocking filtering unit 56 and the frame memory 61.

[0088]

The deblocking filtering unit 56 performs a filtering process on the decoded image data supplied from the adder 55 similarly to the deblocking filtering unit 24 of Fig. 2 to remove block distortion and outputs the decoded image data to the frame reordering buffer 57 and the frame memory 61.

[0089]

The frame reordering buffer 57 performs screen reorder. That is, the frames reorderd from the encoding order used in the frame reordering buffer 12 of Fig. 2 are reorderd in the original display order and are output to the D/A converter 58.

[0090]

The D/A converter 58 performs D/A conversion on the image data supplied from the frame reordering buffer 57 and outputs the image data to a display (not illustrated)

to thereby display images.

[0091]

The frame memory 61 maintains the decoded image data before the filtering process, supplied from the adder 55 and the decoded image data having been subjected to the filtering process, supplied from the deblocking filtering unit 56 as image data of a reference image.

[0092]

The selector 62 supplies the reference image data before the filtering process, read from the frame memory 61 to the intra predictor 63 when a predicted block having been subjected to intra-prediction is decoded based on the prediction mode information supplied from the lossless decoder 52. Moreover, the selector 26 supplies the reference image data having been subjected to the filtering process, read from the frame memory 61 to the motion compensation unit 64 when a predicted block having been subjected to inter-prediction is decoded based on the prediction mode information supplied from the lossless decoder 52.

[0093]

The intra predictor 63 generates predicted images based on the prediction mode information supplied from the lossless decoder 52 and outputs the generated predicted image data to the selector 65.

[0094]

The motion compensation unit 64 performs motion compensation based on the prediction mode information supplied from the lossless decoder 52 to generate predicted image data and outputs the predicted image data to the selector 65. That is, the motion compensation

unit 64 performs motion compensation using a motion vector based on the motion vector information with respect to the reference image indicated by the reference frame information based on the motion vector information and the reference frame information included in the prediction mode information to generate predicted image data.

[0095]

The selector 65 supplies the predicted image data generated by the intra predictor 63 to the adder 55. Moreover, the selector 65 supplies the predicted image data generated by the motion compensation unit 64 to the adder 55.

[0096]

A decoding unit in the claims includes the lossless decoder 52, the inverse quantizer 53, the inverse orthogonal transformer 54, the adder 55, the intra predictor 63, the motion compensation unit 64, and the like.

[0097]

[5. Operation of image decoding device]

Next, an image processing device performed by the image decoding device 50 will be described with reference to the flowchart of Fig. 7.

[0098]

In step ST51, the accumulation buffer 51 accumulates encoded streams transferred thereto. In step ST52, the lossless decoder 52 performs a lossless decoding process. The lossless decoder 52 decodes the encoded streams supplied from the accumulation buffer 51. That is, the quantized data of the respective pictures

encoded by the lossless encoder 16 of Fig. 2 is obtained. Moreover, the lossless decoder 52 performs lossless decoding on the prediction mode information included in the header information of the encoded stream to obtain prediction mode information and supplies the prediction mode information to the deblocking filtering unit 56 and the selectors 62 and 65. Further, the lossless decoder 52 outputs the prediction mode information to the intra predictor 63 when the prediction mode information is information on the intra-prediction mode. In addition, the lossless decoder 52 outputs the prediction mode information to the motion compensation unit 64 when the prediction mode information is information on the inter-prediction mode.

[0099]

In step ST53, the inverse quantizer 53 performs an inverse quantization process. The inverse quantizer 53 performs inverse quantization on the quantized data decoded by the lossless decoder 52 according to a property corresponding to the property of the quantizer 15 of Fig. 2.

[0100]

In step ST54, the inverse orthogonal transformer 54 performs an orthogonal transform process. The inverse orthogonal transformer 54 performs inverse orthogonal transform on the transform coefficient data inverse-quantized by the inverse quantizer 53 according to a property corresponding to the property of the orthogonal transformer 14 of Fig. 2.

[0101]

In step ST55, the adder 55 generates decoded image

data. The adder 55 adds the data obtained by performing the orthogonal transform process and the predicted image data selected in step ST59 described later to generate the decoded image data. In this manner, the original image is decoded.

[0102]

In step ST56, the deblocking filtering unit 56 performs a deblocking filtering process. The deblocking filtering unit 56 performs a filtering process on the decoded image data output by the adder 55 to remove block distortion included in the decoded image.

[0103]

In step ST57, the frame memory 61 stores the decoded image data.

[0104]

In step ST58, the intra predictor 63 and the motion compensation unit 64 perform a prediction process. The intra predictor 63 and the motion compensation unit 64 perform the prediction process on the prediction mode information supplied from the lossless decoder 52.

[0105]

That is, when the prediction mode information of the intra-prediction is supplied from the lossless decoder 52, the intra predictor 63 performs an intra-prediction process based on the prediction mode information to generate the predicted image data. Moreover, when the prediction mode information of the inter-prediction is supplied from the lossless decoder 52, the motion compensation unit 64 performs motion compensation based on the prediction mode information to generate the predicted image data.

[0106]

In step ST59, the selector 65 selects predicted image data. That is, the selector 65 selects the predicted image supplied from the intra predictor 63 and the predicted image data generated by the motion compensation unit 64 and supplies the predicted image and the predicted image data to the adder 55 so as to be added to the output of the inverse orthogonal transformer 54 in step ST55 as described above.

[0107]

In step ST60, the frame reordering buffer 57 performs screen reorder. That is, the frame reordering buffer 57 reorders the frames so that the frames reorderd in the encoding order used in the frame reordering buffer 12 of the image encoding device 10 of Fig. 2 are reorderd in the original display order.

[0108]

In step ST61, the D/A converter 58 performs D/A conversion on the image data from the frame reordering buffer 57. This image is output to a display (not illustrated) and the image is displayed.

[0109]

[6. Basic operation of deblocking filtering unit]

In general, a deblocking filtering process according to an image encoding scheme such as H.264/AVC or HEVC involves determining whether filtering is necessary or not and performing a filtering process with respect to an inter-block boundary in which filtering is determined to be necessary.

[0110]

The inter-block boundary includes boundaries (that

is, horizontally adjacent inter-block boundaries (hereinafter referred to as "vertical boundaries") detected by performing inter-block boundary detection in the horizontal direction. Moreover, the inter-block boundary includes inter-block boundaries (that is, vertically adjacent inter-block boundaries (hereinafter referred to as "line boundaries") detected by performing inter-block boundary detection in the vertical direction.

[0111]

Fig. 8 is an explanatory diagram illustrating an example of pixels in two blocks BKa and BKb adjacent with a boundary interposed. In this example, although a vertical boundary is used as an example, the same can be applied to a line boundary. In the example of Fig. 8, image data of pixels in the block BKa is represented by symbol "pi,j." Here, "i" is a column index of pixel and "j" is a row index of pixel. Moreover, a smallest unit of the encoding process is a block of 8 × 8 pixels, and column indexes "i" of 0, 1, 2, and 3 are assigned sequentially (from left to right) from the column close to the vertical boundary. Row indexes "j" of 0, 1, 2, ..., and 7 are assigned from top to bottom. The left half of the block BKa is omitted in the drawing. On the other hand, image data of pixels in the block BKb is represented by symbol "qk,j." Here, "k" is a column index of pixel and "j" is a row index of pixel. The column indexes "k" of 0, 1, 2, and 3 are assigned sequentially (from left to right) from the column close to the vertical boundary. The right half of the block BKb is also omitted in the drawing.

[0112]

When it is determined that a deblocking filter is to be applied to a boundary and the boundary is a vertical boundary, for example, a filtering process is performed on the pixels on the left and right sides of the boundary. As for luminance components, a filter having strong filter strength and a filter having weak filter strength are switched according to the value of image data.

[0113]

[Luminance component filtering]

In selecting of strength, whether the conditions of Expressions (3) to (5) are satisfied is determined for each line (or each column). A strong filter is selected when all conditions of Expressions (3) to (5) are satisfied, and a weak filter is selected when the conditions of any one of the expressions are not satisfied.

[0114]

[Mathematical formula 1]

$$d < (\beta >> 2) \quad \cdots (3)$$

$$|p_3 - p_0| + |q_0 - q_3| < (\beta >> 3) \quad \cdots (4)$$

$$|p_0 - q_0| < ((5 * t_C + 1) >> 1) \quad \cdots (5)$$

[0115]

In Expression (3), "d" is a value calculated based on Expression (6). Moreover, "β" in Expressions (4) and (5) and "tc" in Expression (5) are values that are set based on a quantization parameter Q as indicated in Table 1.

[0116]

[Mathematical formula 2]

$$d = |p_{2,2} - 2*p_{1,2} + p_{0,2}| + |q_{2,2} - 2*q_{1,2} + q_{0,2}| + |p_{2,5} - 2*p_{1,5} + p_{0,5}| + |q_{2,5} - 2*q_{1,5} + q_{0,5}| \quad \cdots (6)$$

[0117]

[Table 1]

Q	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
β	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	7	8
t_C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Q	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
β	9	10	11	12	13	14	15	16	17	18	20	22	24	26	28	30	32	34	36
t_C	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4
Q	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	
β	38	40	42	44	46	48	50	52	54	56	58	60	62	64	64	64	64	64	
t_C	5	5	6	6	7	8	9	9	10	10	11	11	12	12	13	13	14	14	

[0118]

In weak filtering, the luminance components of the respective pixels in a filter processing range are calculated by performing the operations of Expressions (7) to (11).

[0119]

[Mathematical formula 3]

$$p_1' = Clip1_Y(p_1 + \Delta/2) \quad \cdots (7)$$
$$p_0' = Clip1_Y(p_0 + \Delta) \quad \cdots (8)$$
$$q_0' = Clip1_Y(q_0 - \Delta) \quad \cdots (9)$$
$$q_1' = Clip1_Y(q_1 - \Delta/2) \quad \cdots (10)$$

$$\Delta = Clip3(-t_C, t_C, (13*(q_0 - p_0) + 4*(q_1 - p_1) - 5*(q_2 - p_0) + 16) >> 5) \quad \cdots (11)$$

[0120]

In strong filtering, the luminance components of the respective pixels in a filter processing range are

calculated by performing the operations of Expressions (12) to (18).

[0121]

[Mathematical formula 4]

$$p_2' = Clip1_Y((2*p_3 + 3*p_2 + p_1 + p_0 + q_0 + 4) >> 3) \quad \cdots (12)$$
$$p_1' = Clip1_Y((p_2 + p_1 + p_0 + q_0 + 2) >> 2) \quad \cdots (13)$$
$$p_0' = Clip1_Y((p_2 + 2*p_1 + 2*p_0 + 2*q_0 + q_1 + 4) >> 3) \quad \cdots (14)$$
$$p_0' = Clip1_Y((p_2 + 2*p_1 + 2*p_0 + 2*q_0 + q_1 + 4) >> 3) \quad \cdots (15)$$
$$q_0' = Clip1_Y((p_1 + 2*p_0 + 2*q_0 + 2*q_1 + q_2 + 4) >> 3) \quad \cdots (16)$$
$$q_1' = Clip1_Y((p_0 + q_0 + q_1 + q_2 + 2) >> 2) \quad \cdots (17)$$
$$q_2' = Clip1_Y((p_0 + q_0 + q_1 + 3*q_2 + 2*q_3 + 4) >> 3) \quad \cdots (18)$$

[0122]

Moreover, in filtering of chrominance components, the chrominance components of the respective pixels in a filter processing range are calculated by performing the operations of Expressions (19) to (21).

[0123]

[Mathematical formula 5]

$$p_0' = Clip1_C(p_0 + \Delta) \quad \cdots (19)$$
$$q_0' = Clip1_C(q_0 - \Delta) \quad \cdots (20)$$

$$\Delta = Clip3(-t_C, t_C, ((((q_0 - p_0) << 2) + p_1 - q_1 + 4) >> 3)) \quad \cdots (21)$$

[0124]

In the above expressions, "Clip1Y" and "Clip1c" represent operations of Expressions (22) and (23), and "Clip3(x,y,z)" in Expressions (22) and (23) represents a value determined by Expression (24).

[0125]

[Mathematical formula 6]

$$\mathrm{Clip1_Y}(x) = \mathrm{Clip3}(0, (1 << \mathrm{BitDepth_Y}) - 1, x) \quad \cdots (22)$$

$$\mathrm{Clip1_C}(x) = \mathrm{Clip3}(0, (1 << \mathrm{BitDepth_C}) - 1, x) \quad \cdots (23)$$

$$\mathrm{Clip3}(x, y, z) = \begin{cases} x & ; \ z < x \\ y & ; \ z > y \\ z & ; \ \text{otherwise} \end{cases} \quad \cdots (24)$$

[0126]

Moreover, as for the line boundary, the operation performed for each line using the pixels in the horizontal direction at the vertical boundary is performed for each column using the pixels in the vertical direction, and the filtering process is performed.

[0127]

In the deblocking filtering process, a process that requires a line memory is the vertical filtering process, and hereinafter, a reduction in the memory capacity of the line memory in the vertical filtering process will be described in detail. Since the deblocking filtering unit 24 of the image encoding device 10 has the same configuration and performs the same operation as those of the deblocking filtering unit 56 of the image decoding device 50, the deblocking filtering unit 24 only will be described.

[0128]

[7. First embodiment of deblocking filtering unit]

Fig. 9 illustrates the configuration of a first embodiment of the deblocking filtering unit. The deblocking filtering unit 24 includes a line memory 241, a line boundary detecting unit 242, a filter strength determining unit 243, a coefficient memory 244, a filter operation unit 245, and a filter controller 246.

[0129]

The line memory 241 stores image data supplied from the adder 23 based on a control signal from the filter controller 246. Moreover, the line memory 241 reads the image data stored therein and outputs the read image data to the line boundary detecting unit 242, the filter strength determining unit 243, and the filter operation unit 245.

[0130]

The line boundary detecting unit 242 detects a line boundary in which the vertical filtering process is performed. The line boundary detecting unit 242 performs the filtering necessity determining process for each block using the image data supplied from the adder 23 and the image data read from the line memory 241 and detects a line boundary in which the vertical filtering process is performed. The line boundary detecting unit 242 outputs the detection results to the filter strength determining unit 243.

[0131]

The filter strength determining unit 243 determines the filter strength as described above. The filter strength determining unit 243 determines whether the vertical filtering process is to be performed in a strong filtering mode or a weak filtering mode using the image data of two blocks adjacent with the line boundary interposed in which the vertical filtering process is performed, and outputs the determination results to the filter operation unit 245.

[0132]

The coefficient memory 244 stores filter

coefficients used in the filtering operation of the deblocking filtering process.

[0133]

The filter operation unit 245 performs a filtering operation with the filter strength determined by the filter strength determining unit 243 using the image data supplied from the adder 23, the image data stored in the line memory 241, and the filter coefficients read from the coefficient memory 244. The filter operation unit 245 outputs the image data having been subjected to the vertical filtering process to the frame memory 25. Moreover, the filter operation unit 245 controls an image range used in the filtering operation in a block that is positioned on the upper side of the line boundary based on the block boundary determination result supplied from the filter controller 246.

[0134]

The filter controller 246 controls the line memory 241 so as to store image data corresponding to a predetermined number of lines on the lower side in the block. Moreover, the filter controller 246 reads the image data stored in the line memory 241. Further, the filter controller 246 includes a line boundary determining unit 2461. The line boundary determining unit 2461 determines whether a boundary is a line boundary (for example, inter-LCU line boundary) in respective block units in which the process is sequentially performed in a raster scan direction and outputs the determination results to the filter operation unit 245. The filter operation unit 245 performs controls such that the vertical filtering process can be

performed even when the memory capacity of the line memory is reduced.

[0135]

Fig. 10 illustrates the operation of the first embodiment of the deblocking filtering unit. In step ST71, the deblocking filtering unit 24 detects a line boundary. The deblocking filtering unit 24 detects a line boundary in which the vertical filtering process is performed.

[0136]

In step ST72, the deblocking filtering unit 24 determines a filter strength. The deblocking filtering unit 24 determines whether a strong filtering mode or a weak filtering mode will be used for the line boundary in which the vertical filtering process is performed.

[0137]

In step ST73, the deblocking filtering unit 24 determines whether the boundary in which the vertical filtering process is performed is an inter-LCU line boundary. The deblocking filtering unit 24 proceeds to step ST74 when the boundary in which the vertical filtering process is performed is the inter-LCU line boundary and proceeds to step ST75 when the boundary is not the inter-LCU line boundary.

[0138]

In step ST74, the deblocking filtering unit 24 performs a range-reduced vertical filtering process. The deblocking filtering unit 24 performs the vertical filtering process while reducing the image range used for the filtering operation of the upper adjacent LCU and the flow proceeds to step ST76.

[0139]

In step ST75, the deblocking filtering unit 24 performs a normal vertical filtering process. The deblocking filtering unit 24 performs the vertical filtering process using predetermined taps and coefficients without reducing the image range used for the filtering operation and the flow proceeds to step ST76.

[0140]

In step ST76, the deblocking filtering unit 24 determines whether the vertical filtering process at the boundary has been completed. When the vertical filtering process for the respective columns of the boundary has not been completed, the deblocking filtering unit 24 returns to step ST76 and performs the process for the next column. Moreover, when the vertical filtering process for the respective columns of the boundary has been completed, the flow proceeds to step ST77.

[0141]

In step ST77, the deblocking filtering unit 24 determines whether the process has been completed up to the end of the screen. When the process has not been completed up to the end of the screen, the deblocking filtering unit 24 returns to step ST71, detects a new boundary and performs the vertical filtering process. When the process has been completed up to the end of the screen, the deblocking filtering unit 24 ends the process for 11 screens.

[0142]

[Configuration and operation of filter operation unit]

Figs. 11 and 12 illustrate the configuration of the

filter operation unit. When a tap-changeable vertical filtering process is performed, the filter operation unit 245 changes the image data of a tap or changes the number of taps such that the vertical filtering process can be performed even when the number of lines that store image data is reduced. Fig. 11 illustrates the configuration of a filter operation unit when a vertical filtering process is performed while changing the image data of a tap. Moreover, Fig. 12 illustrates the configuration of a filter operation unit when a vertical filtering process is performed while changing the number of taps.

[0143]

In Fig. 11, the filter operation unit 245 includes a data storage unit 2451, a data selecting unit 2452, and an arithmetic processing unit 2453.

[0144]

When the memory capacity of a line memory is reduced, the data storage unit 2451 stores image data to be used for a tap at the position of the reduced line. The data storage unit 2451 outputs the image data stored therein to the data selecting unit 2452 as the image data of a tap at the position of the reduced line.

[0145]

The data selecting unit 2452 selects one of the image data stored in the data storage unit 2451 and the image data stored in the line memory and outputs the selected image data to the arithmetic processing unit 2453.

[0146]

The arithmetic processing unit 2453 performs an arithmetic process using the image data supplied from the

adder 23 and the data selecting unit 2452 and the filter coefficients read from the coefficient memory 244 to generate image data after the vertical filtering process and outputs the image data to the frame memory 25.

[0147]

In Fig. 12, the filter operation unit 245 includes an arithmetic processing unit 2455 and an arithmetic processing unit 2456.

[0148]

The arithmetic processing unit 2455 performs an arithmetic process with a predetermined number of taps and outputs image data after the vertical filtering process to the data selecting unit 2457.

[0149]

The arithmetic processing unit 2456 performs an arithmetic process without using the image data of the reduced line by reducing the number of taps according to a reduction in the memory capacity of the line memory and outputs the image data after the vertical filtering process to the data selecting unit 2457.

[0150]

The data selecting unit 2457 selects the image data depending on whether the boundary is an inter-LCU line boundary. When the boundary is not the boundary between LCUs, the data selecting unit 2457 selects the image data output from the arithmetic processing unit 2455. Moreover, when the boundary is the inter-LCU line boundary, the data selecting unit 2457 selects the image data output from the arithmetic processing unit 2456.

[0151]

Figs. 13(A) and 13(B) are diagrams for describing

the operation of the filter operation unit 245. The drawings illustrate the image data of respective pixels of two adjacent blocks BKu and BKl. Here, the filtering operation is performed using three pixels each from a boundary (line boundary) BB as a filter processing range of the vertical filtering process and four pixels each from the boundary BB as a tap. When it is determined that the filtering is to be performed in a strong filtering mode, the operations of Expressions (12) to (18) described above are performed for each column by the arithmetic processing units 2453 and 2455. Moreover, when it is determined that the filtering is to be performed in a weak filtering mode, the operations of Expressions (7) to (10) described above are performed for each column by the arithmetic processing units 2453 and 2455.

[0152]

When the deblocking filtering process is performed in a raster order in respective LCU units, it is determined that the vertical filtering process is performed in a strong filtering mode, and the operations of Expressions (12) to (18) described above are performed. Thus, as illustrated in Fig. 13(A), the image data of the block BKu on the upper side of the boundary BB corresponding to four lines from the boundary BB needs to be stored in the line memory.

[0153]

Here, the filter operation unit 245 controls the image range used for the filtering operation in the block positioned on the upper side of the boundary BB whereby the vertical filtering process can be performed even when

the memory capacity of the line memory is reduced. For example, as illustrated in Fig. 13(B), image data corresponding to three lines from the boundary BB is stored, and the vertical filtering process is performed using the stored image data. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the operation of Expression (25). Moreover, the arithmetic processing unit 2456 illustrated in Fig. 12 performs the operation of Expression (26), and the data selecting unit 2457 selects the image data from the arithmetic processing unit 2456 and outputs the image data. In Expressions (25) and (26), "i" indicates a column index of pixel, and when the filtering process is performed in units of blocks of 8 × 8 pixels, "i = 0 to 7."

[0154]

[Mathematical formula 7]

$$p_{2i}' = \mathrm{Clip1}_Y((2*p_{2i} + 3*p_{2i} + p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (25)$$

$$p_{2i}' = \mathrm{Clip1}_Y((\quad 5*p_{2i} + p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (26)$$

[0155]

The arithmetic processing unit 2453 copies the pixels in the upper block BKu on the upper end of the filter processing range in the upward direction and uses the same. That is, by using the image data p2 of the pixels at the upper end of the filter processing range stored in the data storage unit 2451 as the image data p3, the image data p2i' after the filtering process is calculated by the operation of Expression (25).

[0156]

The arithmetic processing unit 2456 reduces the

number of taps and calculates the image data p2i' after the filtering process by the operation of Expression (26) using the coefficients changes with the reduction in the number of taps. In this case, the coefficients are changed such that the coefficient of the image data p2 is changed from "3" to "5" so as to correspond to copying of the upper end pixels in the tap range.

[0157]

When the boundary BB is not the inter-LCU line boundary, since it is not necessary to use the image data of the line memory 241, the same filtering operation as the conventional operation is performed without narrowing the image range used for the filtering operation. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the same filtering operation as the conventional operation. Moreover, the arithmetic processing unit 2455 illustrated in Fig. 12 performs the same filtering operation as the conventional operation and the data selecting unit 2457 selects and outputs the image data from the arithmetic processing unit 2455.

[0158]

In this manner, when the boundary is a line boundary of blocks in which the process is sequentially performed in a raster scan order, by controlling the image range used for the filtering operation, even when the memory capacity of the line memory is reduced, the deblocking filtering process can be performed in the same manner as before the reduction. Moreover, for example, one line of image of 4K × 2K corresponds to two lines of image of 2K × 1K. Further, in the H.264/AVC scheme, a line memory corresponding to four lines is included, and

the memory capacity corresponding to one line of image of 4K × 2K corresponds to 50% of the memory capacity of the H.264/AVC scheme. Thus, the effect of reducing the memory capacity in a high-resolution image is improved.

[0159]

[8. Second embodiment of deblocking filtering unit]

A second embodiment of the deblocking filtering unit is different from the first embodiment in that the operations of the arithmetic processing unit 2453 and the arithmetic processing unit 2456 are different from those of the first embodiment.

[0160]

In strong filtering, when the boundary BB is the inter-LCU line boundary, and the operations of Expressions (12) to (18) described above are performed, as illustrated in Fig. 13(A), the image data of the block BKu on the upper side of the boundary BB corresponding to four lines from the boundary BB needs to be stored in the line memory.

[0161]

Here, the filter operation unit 245 controls the image range used for the filtering operation in the block positioned on the upper side of the boundary BB whereby the vertical filtering process can be performed even when the memory capacity of the line memory is reduced. For example, as illustrated in Fig. 13(B), image data corresponding to three lines from the boundary BB is stored, and the vertical filtering process is performed using the stored image data. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the operation of Expression (27). Moreover, the arithmetic

processing unit 2456 illustrated in Fig. 12 performs the operation of Expression (28), and the data selecting unit 2457 selects the image data from the arithmetic processing unit 2456 and outputs the image data. In Expressions (27) and (28), "i" indicates a column index of pixel, and when the filtering process is performed in units of blocks of 8 × 8 pixels, "i = 0 to 7."

[0162]

[Mathematical formula 8]

$$p_{2i}' = Clip1_Y((2*p_{1i} + 3*p_{2i} + p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (27)$$

$$p_{2i}' = Clip1_Y((3*p_{2i} + 3*p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (28)$$

[0163]

The arithmetic processing unit 2453 performs mirror copying about the pixels in the upper block BKu on the upper end of the filter processing range. That is, by storing the image data p1 in the data storage unit 2451, mirror-copying the image data p1 about the upper end pixels in the filter processing range to use the image data p1 as the image data p3, the image data p2i' after the filtering process is calculated by the operation of Expression (27).

[0164]

The arithmetic processing unit 2456 reduces the number of taps and calculates the image data p2i' after the filtering process by the operation of Expression (28) using the coefficients changes with the reduction in the number of taps. In this case, the coefficients are changed such that the coefficient of the image data p1 is changed from "2" to "3" so as to correspond to mirror-

copying about the upper end pixels in the filter processing range.

[0165]

When the boundary BB is not the inter-LCU line boundary, since it is not necessary to use the image data of the line memory 241, the same filtering operation as the conventional operation is performed without narrowing the image range used for the filtering operation. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the same filtering operation as the conventional operation. Moreover, the arithmetic processing unit 2455 illustrated in Fig. 12 performs the same filtering operation as the conventional operation and the data selecting unit 2457 selects and outputs the image data from the arithmetic processing unit 2455.

[0166]

In this manner, when the boundary is a line boundary of blocks in which the process is sequentially performed in a raster scan order, by controlling the image range used for the filtering operation, similarly to the first embodiment, even when the memory capacity is reduced, the deblocking filtering process can be performed.

[0167]

[9. Third embodiment of deblocking filtering unit]

A third embodiment of the deblocking filtering unit is different from the first and second embodiments in that the operations of the arithmetic processing unit 2453 and the arithmetic processing unit 2456 are different from those of the first embodiment.

[0168]

Figs. 14(A) and 14(B) are diagrams for describing the operation of the filter operation unit 245. The drawings illustrate the image data of respective pixels of two adjacent blocks BKu and BKl. Here, the filtering operation is performed using three pixels each from a boundary (line boundary) BB as a filter processing range of the vertical filtering process and four pixels each from the boundary BB as a tap. When it is determined that the filtering is to be performed in a strong filtering mode, the operations of Expressions (12) to (18) described above are performed for each column by the arithmetic processing units 2453 and 2455. Moreover, when it is determined that the filtering is to be performed in a weak filtering mode, the operations of Expressions (7) to (10) described above are performed for each column by the arithmetic processing units 2453 and 2455.

[0169]

When the deblocking filtering process is performed in a raster order in respective LCU units, it is determined that the vertical filtering process is performed in a strong filtering mode, and the operations of Expressions (12) to (18) described above are performed. Thus, as illustrated in Fig. 14(A), the image data of the block BKu on the upper side of the boundary BB corresponding to four lines from the boundary BB needs to be stored in the line memory.

[0170]

Here, the filter operation unit 245 controls the filter processing range and the image range used for the filtering operation in the block positioned on the upper

side of the boundary BB whereby the vertical filtering process can be performed even when the memory capacity of the line memory is reduced. For example, as illustrated in Fig. 14(B), the filter processing range in the upper block BKu is set to a range of two pixels from the boundary BB, and the image data corresponding to two lines from the boundary BB is stored. Further, the vertical filtering process is performed using the stored image data similarly to the first embodiment. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the operations of Expressions (29) and (30). Moreover, the arithmetic processing unit 2456 illustrated in Fig. 12 performs the operations of Expressions (31) and (32), and the data selecting unit 2457 selects the image data from the arithmetic processing unit 2456 and outputs the image data. In Expressions (29) to (32), "i" indicates a column index of pixel, and when the filtering process is performed in units of blocks of 8 × 8 pixels, "i = 0 to 7."

[0171]

[Mathematical formula 9]

$$p_{1i}' = \mathrm{Clip1_Y}((p_{1i} + p_{1i} + p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (29)$$

$$p_{0i}' = \mathrm{Clip1_Y}((p_{1i} + 2*p_{1i} + 2*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (30)$$

$$p_{1i}' = \mathrm{Clip1_Y}((2*p_{1i} + p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (31)$$

$$p_{0i}' = \mathrm{Clip1_Y}((3*p_{1i} + 2*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (32)$$

[0172]

The arithmetic processing unit 2453 copies the pixels in the upper block BKu on the upper end of the filter processing range in the upward direction and uses the same. That is, by using the image data p1 of the

pixels at the upper end of the filter processing range stored in the data storage unit 2451 as the image data p2, the image data p1i' and p0i' after the filtering process is calculated by the operations of Expressions (29) and (30).

[0173]

The arithmetic processing unit 2456 reduces the number of taps and calculates the image data p1i' and p0i' after the filtering process by the operations of Expressions (31) and (32) using the coefficients changes with the reduction in the number of taps. In this case, the coefficients are changed such that the coefficient of the image data p1 in Expression (31) is changed from "1" to "2" and the image data p1 in Expression (32) is changed from "2" to "3" so as to correspond to copying of the upper end pixels in the tap range.

[0174]

When the boundary BB is not the inter-LCU line boundary, since it is not necessary to use the image data of the line memory 241, the same filtering operation as the conventional operation is performed without narrowing the image range used for the filtering operation. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the same filtering operation as the conventional operation. Moreover, the arithmetic processing unit 2455 illustrated in Fig. 12 performs the same filtering operation as the conventional operation and the data selecting unit 2457 selects and outputs the image data from the arithmetic processing unit 2455.

[0175]

In this manner, when the boundary is a line

boundary of blocks in which the process is sequentially performed in a raster scan order, by controlling the filter processing range and the image range used for the filtering operation, even when the memory capacity is reduced, the deblocking filtering process can be performed in the same manner as before the reduction. Moreover, it is possible to reduce the memory capacity much more.

[0176]

[10. Fourth embodiment of deblocking filtering unit]

A fourth embodiment of the deblocking filtering unit is different from the first embodiment in that the operations of the arithmetic processing unit 2453 and the arithmetic processing unit 2456 are different from those of the third embodiment.

[0177]

In strong filtering, when the boundary BB is the inter-LCU line boundary, and the operations of Expressions (12) to (18) described above are performed, as illustrated in Fig. 14(A), the image data of the block BKu on the upper side of the boundary BB corresponding to four lines from the boundary BB needs to be stored in the line memory.

[0178]

Here, the filter operation unit 245 controls the filter processing range and the image range used for the filtering operation in the block positioned on the upper side of the boundary BB whereby the vertical filtering process can be performed even when the memory capacity of the line memory is reduced. For example, as illustrated in Fig. 14(B), the filter processing range in the upper

block BKu is set to a range of two pixels from the boundary BB, and the image data corresponding to two lines from the boundary BB is stored. Further, the vertical filtering process is performed using the stored image data similarly to the second embodiment. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the operations of Expressions (33) and (34). Moreover, the arithmetic processing unit 2456 illustrated in Fig. 12 performs the operations of Expressions (35) and (36), and the data selecting unit 2457 selects the image data from the arithmetic processing unit 2456 and outputs the image data. In Expressions (33) to (36), "i" indicates a column index of pixel, and when the filtering process is performed in units of blocks of 8 × 8 pixels, "i = 0 to 7."

[0179]

[Mathematical formula 10]

$$p_{1i}' = \mathrm{Clip1}_Y((p_{0i} + p_{1i} + p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (33)$$

$$p_{0i}' = \mathrm{Clip1}_Y((p_{0i} + 2*p_{1i} + 2*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (34)$$

$$p_{1i}' = \mathrm{Clip1}_Y((p_{1i} + 2*p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (35)$$

$$p_{0i}' = \mathrm{Clip1}_Y((2*p_{1i} + 3*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (36)$$

[0180]

The arithmetic processing unit 2453 performs mirror copying about the pixels in the upper block BKu on the upper end of the filter processing range. That is, by storing the image data p0 in the data storage unit 2451, mirror-copying the image data p0 about the upper end pixels in the filter processing range to use the image data p0 as the image data p2, the image data p1i' and p0i' after the filtering process is calculated by the

operations of Expressions (33) and (34).

[0181]

The arithmetic processing unit 2456 reduces the number of taps and calculates the image data p1i' and p0i' after the filtering process by the operations of Expressions (35) and (36) using the coefficients changes with the reduction in the number of taps. In this case, the coefficients are changed such that the coefficient of the image data p0 in Expression (35) is changed from "1" to "2" and the image data p0 in Expression (36) is changed from "2" to "3" so as to correspond to mirror-copying about the upper end pixels in the filter processing range.

[0182]

When the boundary BB is not the inter-LCU line boundary, since it is not necessary to use the image data of the line memory 241, the same filtering operation as the conventional operation is performed without narrowing the image range used for the filtering operation. That is, the arithmetic processing unit 2453 illustrated in Fig. 11 performs the same filtering operation as the conventional operation. Moreover, the arithmetic processing unit 2455 illustrated in Fig. 12 performs the same filtering operation as the conventional operation and the data selecting unit 2457 selects and outputs the image data from the arithmetic processing unit 2455.

[0183]

In this manner, when the boundary is a line boundary of blocks in which the process is sequentially performed in a raster scan order, by controlling the filter processing range and the image range used for the

filtering operation, even when the memory capacity is reduced, the deblocking filtering process can be performed in the same manner as before the reduction. Moreover, it is possible to reduce the memory capacity much more.

[0184]

[11. Fifth embodiment of deblocking filtering unit]

The deblocking filtering unit described above controls the image range used for the filtering operation in a block positioned on the upper side of a boundary and controls the filter processing range depending on whether the boundary BB is an inter-LCU line boundary. Next, in a fifth embodiment, a case where a mode where the image range is controlled when the boundary BB is the inter-LCU line boundary and a mode where the image range is controlled regardless of whether the boundary BB is the inter-LCU line boundary are provided will be described.

[0185]

Fig. 15 is a flowchart illustrating the operation of the fifth embodiment. In step ST81, the deblocking filtering unit 24 detects a line boundary. The deblocking filtering unit 24 detects a line boundary in which the vertical filtering process is performed.

[0186]

In step ST82, the deblocking filtering unit 24 determines a filter strength. The deblocking filtering unit 24 determines whether a strong filtering mode or a weak filtering mode will be used for the line boundary in which the vertical filtering process is performed.

[0187]

In step ST83, the deblocking filtering unit 24

determines whether a range-reduced vertical filtering process is to be performed with respect to the inter-LCU line boundary only (that is, whether a vertical filtering process is to be performed by reducing the image range used for the filtering operation). When it is determined that the range-reduced vertical filtering process is to be performed with respect to the line boundary of blocks smaller than the size of LCU as well as the inter-LCU line boundary, the deblocking filtering unit 24 proceeds to step ST84. Moreover, when it is determined that the range-reduced vertical filtering process is to be performed with respect to the inter-LCU line boundary only, the deblocking filtering unit 24 proceeds to step ST85.

[0188]

The deblocking filtering unit 24 determines whether the range-reduced vertical filtering process is to be performed with respect to the inter-LCU line boundary only based on the quantization parameters set in units of frames, for example. When the quantization parameter is small, better image quality is obtained as compared to when the quantization parameter is large. Thus, when the quantization parameter is larger than a predetermined threshold value, the deblocking filtering unit 24 determines that the vertical filtering process is to be performed in a mode where image quality is improved by performing the range-reduced vertical filtering process with respect to the inter-LCU line boundary only and performing a normal vertical filtering process with respect to a line boundary of blocks smaller than the size of LCU, and proceeds to step ST84. Moreover, when

the quantization parameter is equal to or smaller than the predetermined threshold value, the deblocking filtering unit 24 determines that the vertical filtering process is to be performed in a mode where control is made easy by performing the range-reduced vertical filtering process with respect to the line boundary between blocks smaller than the size of LCU as well as the inter-LCU line boundary, and proceeds to step ST85.

[0189]

In step ST84, the deblocking filtering unit 24 determines whether a boundary in which the vertical filtering process is performed is an inter-LCU line boundary. The deblocking filtering unit 24 proceeds to step ST85 when the boundary in which the vertical filtering process is performed is the inter-LCU line boundary and proceeds to step ST86 when the boundary is a line boundary of blocks smaller than the size of LCU.

[0190]

In step ST85, the deblocking filtering unit 24 performs a range-reduced vertical filtering process. The deblocking filtering unit 24 performs the vertical filtering process while reducing the image range used for the filtering operation of the upper adjacent LCU and proceeds to step ST87. In the range-reduced vertical filtering process, the filtering range may be reduced.

[0191]

In step ST86, the deblocking filtering unit 24 performs a normal vertical filtering process. The deblocking filtering unit 24 performs the vertical filtering process without reducing the image range used for the filtering operation and proceeds to step ST87.

[0192]

In step ST87, the deblocking filtering unit 24 determines whether the vertical filtering process at the boundary has been completed. When the vertical filtering process for the respective columns of the boundary has not been completed, the deblocking filtering unit 24 returns to step ST87 and performs the process for the next column. Moreover, when the vertical filtering process for the respective columns of the boundary has been completed, the flow proceeds to step ST88.

[0193]

In step ST88, the deblocking filtering unit 24 determines whether the process has been completed up to the end of the screen. When the process has not been completed up to the end of the screen, the deblocking filtering unit 24 returns to step ST81, detects a new boundary and performs the vertical filtering process. When the process has been completed up to the end of the screen, the deblocking filtering unit 24 ends the process for one screen.

[0194]

When such a process is performed, it is possible to reduce the memory capacity of the line memory. When the range-reduced vertical filtering process is performed with respect to other line boundaries as well as the inter-LCU line boundary, switching of filtering processes is not required and control is made easy. Moreover, when the range-reduced vertical filtering process is performed with respect to the inter-LCU line boundary only, better image quality can be obtained.

[0195]

[12. Description of sixth to eighth embodiments]

[Description of conventional technique]

In the above description, an example in which the image data of a tap in the image range is copied or mirror-copied and used as image data of the tap that is outside the image range due to the narrowing of the image range used for the filtering operation has been described. Here, in the present specification, copying is synonymous with padding.

[0196]

As in the present technique, a method of performing a filtering process using padding in the line boundary of LCUs only has been proposed in JCTVC-F053.

[0197]

Referring to Fig. 16, a filtering process of a strong filter and a filtering process using padding for luminance signals in the HEVC scheme will be described.

[0198]

Fig. 16 is an explanatory diagram illustrating an example of pixels in two blocks BKu and BKl that are adjacent vertically with a line boundary interposed. In the example of Fig. 16, image data of pixels in the block BKu is represented by symbol "pj_i." Here, "j" is a row index of pixel and "i" is column index of pixel. Moreover, a smallest unit of the encoding process is a block of 8 × 8 pixels, and row indexes "j" of 0, 1, 2, and 3 are assigned sequentially (from top to bottom) from the row close to the line boundary BB. Column indexes "i" of 0, 1, 2, ..., and 7 are assigned from left to right of the block. The upper half of the block BKu is omitted in the drawing. On the other hand, image data of

pixels in the block BK1 is represented by symbol "qk_i." Here, "k" is a row index of pixel and "i" is a column index of pixel. The row indexes "k" of 0, 1, 2, and 3 are assigned sequentially (from top to bottom) from the row close to the line boundary BB. The lower half of the block BK1 is also omitted in the drawing.

[0199]

In the strong filtering of the luminance signal in the HEVC scheme, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (37) to (42). Expressions (37) to (42) correspond to Expressions (14), (16), (13), (17), (12), and (18), respectively.

[0200]

[Mathematical formula 11]

$$p0_0 = Clip_{0\text{-}255}((p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(37)$$

$$q0_0 = Clip_{0\text{-}255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(38)$$

$$p1_0 = Clip_{0\text{-}255}((p2_i+p1_i+p0_i+q0_i+2)>>2); \quad i=0,7 \quad \cdots(39)$$

$$q1_0 = Clip_{0\text{-}255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(40)$$

$$p2_0 = Clip_{0\text{-}255}((2*p3_i+3*p2_i+p1_i+p0_i+q0_i+4)>>3); \quad i=0,7 \quad \cdots(41)$$

$$q2_0 = Clip_{0\text{-}255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(42)$$

Here, "$Clip_{0\text{-}255}$" indicates a clipping process of rounding a value up to "0" when the value is "0" or smaller and rounding a value down to "255" when the value is "255" or larger. The same is applied to the following expressions.

[0201]

In contrast, in the conventional proposal described above, the filtering process is performed using padding with R2W2 reduction although the filtering process is

performed at the line boundary of LCUs only. Here, R2W2 represents that the filtering process is applied to two pixels on the inter-LCU line boundary by referring to two pixels on the inter-LCU line boundary.

[0202]

In the strong filtering of luminance signals in the conventional proposal, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (43) to (47).

[0203]

[Mathematical formula 12]

$$\cdot p0_0 = Clip_{0\text{-}255}((p1_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3);\quad i=0,7 \quad \cdots(43)$$

$$\cdot q0_0 = Clip_{0\text{-}255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3);\quad i=0,7 \quad \cdots(44)$$

$$\cdot p1_0 = Clip_{0\text{-}255}((p1_i+p1_i+p0_i+q0_i+2)>>2);\quad i=0,7 \quad \cdots(45)$$

$$\cdot q1_0 = Clip_{0\text{-}255}((p0_i+q0_i+q1_i+q2_i+2)>>2);\quad i=0,7 \quad \cdots(46)$$

$$\cdot q2_0 = Clip_{0\text{-}255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3);\quad i=0,7 \quad \cdots(47)$$

[0204]

Here, Expression (43) of the strong filtering in the conventional proposal is different in that "$p2_i$" in Expression (37) of the strong filtering of the HEVC scheme is changed to "$p1_i$." Expression (45) of the strong filtering in the conventional proposal is different in that "$p2_i$" in Expression (39) of the strong filtering of the HEVC scheme is changed to "$p1_i$." The strong filtering in the conventional proposal is different in that an expression corresponding to Expression (41) of the strong filtering of the HEVC scheme is deleted.

[0205]

Expressions (44), (46), and (47) of the strong

filtering in the conventional proposal are the same as Expressions (38), (40), and (42), respectively, of the strong filtering of the HEVC scheme.

[0206]

That is, in the conventional proposal, since the pixel "$p2_i$" on the third row from the line boundary BB of the block Bku is not referred to, the pixel "$p1_i$" on a row immediately below the third row of the same column is padded and used instead.

[0207]

Moreover, the filtering process is not applied to the pixel "$p2_0$" on the third row from the line boundary BB of the block Bku. Therefore, in the strong filtering of the conventional proposal, an expression corresponding to Expression (41) of the strong filtering of the HEVC scheme is deleted.

[0208]

By doing so, in the conventional proposal, the memory capacity of the line memory is suppressed from becoming larger than that of the HEVC scheme.

[0209]

However, in the case of 4k images, since it is necessary to have a large line memory, it is requested to further reduce the line memory in the deblocking process. Moreover, in reducing the line memory, it is requested to maintain a better block noise reduction function of the deblocking process than the method of the conventional proposal.

[0210]

Here, in inter-block boundary determination in the HEVC scheme, as indicated by Expression (48) below, a

waveform having a constant inclination on both sides of a line boundary is subjected to the process. Therefore, in the present technique, the deblocking process at the inter-LCU line boundary is performed using the inclination of the waveform used in the inter-block boundary determination, which will be described in detail below.

[0211]

[Mathematical formula 13]

$$|p2_2-2*p1_2+p0_2|+|q2_2-2*q1_2+q0_2|+|p2_5-2*p1_5+p0_5|+|q2_5-2*q1_5+q0_5|<\beta \quad \cdots(48)$$

[0212]

[Description of present technique (linear approximation)]

In the example of Fig. 17, an inter-block boundary determination expression used in the HEVC scheme, of Expression (48) is illustrated.

[0213]

As illustrated in Fig. 17, the first term on the left side of Expression (48) is an expression for determining a pixel on the third column from the left side of the block Bku and can be expressed by a difference of differences (second-order difference) (that is, a second-order differential having a constant inclination.

$$p2_2 - 2*p1_2 + p0_2 = (p2_2 - p1_2) - (p1_2 - p0_2)$$

[0214]

The second to fourth terms on the left side of Expression (48) are an expression for determining a pixel on the sixth column from the left side of the block Bku, an expression for determining a pixel on the third column from the left side of the block Bkl, and an expression

for determining a pixel on the sixth column from the left side of the block Bk1, respectively. The same can be said for the second to fourth terms on the left side of Expression (48).

[0215]

As above, in the inter-block boundary determination in the HEVC scheme, since the waveform having a constant inclination is subjected to the process, the deblocking process at the inter-LCU line boundary uses the inclination of the waveform used in the inter-block boundary determination.

[0216]

In the conventional technique, as for pixels that could not be referred to, the deblocking process at the inter-LCU line boundary uses padding. In contrast, in the present technique, as for pixels that could not be referred to, the deblocking process at the inter-LCU line boundary performs linear approximation using the inclination of the waveform.

[0217]

That is, in the deblocking process at the inter-LCU line boundary in the present technique, Expression (49) below of linear approximation is used.

[0218]

[Mathematical formula 14]

$$p2_i = p1_i + (p2_i - p1_i) \fallingdotseq p1_i + (p1_i - p0_i) = 2*p1_i - p0_i \quad \cdots(49)$$

[0219]

As illustrated in Fig. 18, when the pixel values of $p2_i$, $p1_i$, and $p0_i$ have an inclination, the pixel value of $p2_i$ could not be referred to in the conventional R2W2-based padding. Thus, although the pixel value of $p1_i$

immediately below the pixel $p2_i$ is padded and used instead of the pixel value of $p2_i$ as indicated by the hatched circle, since the actual pixel value of $p2_i$ is at the position of the dot circle, an error occurs.

[0220]

In contrast, in the R2W2-based linear prediction of the present technique, since the pixel value of $p2_i$ could not be referred to, a pixel value (= the same pixel value as the actual pixel value) predicted by linear prediction from the inclination based on the pixel values of $p1_i$ and $p0_i$ is used as the pixel value of $p2_i$ as indicated by the hatched circle.

[0221]

By doing so, when pixel values have an inclination, the present technique does not incur more errors than the conventional padding method. Thus, although the R2W2-based technique is used, it is possible to reduce the line memory while maintaining the block noise reduction function of the deblocking process.

[0222]

Moreover, for example, one line of image of 4K × 2K corresponds to two lines of image of 2K × 1K. Further, in the H.264/AVC scheme, a line memory corresponding to four lines is included, and the memory capacity corresponding to one line of image of 4K × 2K corresponds to 50% of the memory capacity of the H.264/AVC scheme. Thus, the effect of reducing the memory capacity in a high-resolution image is improved.

[0223]

When the pixel values of $p2_i$, $p1_i$, and $p0_i$ do not have an inclination, substantially the same result is

obtained as when the pixel value of $p1_i$ immediately below the pixel $p2_i$ is used. In this case, although the R2W2-based technique is used, it is possible to reduce the line memory while maintaining the block noise reduction function of the deblocking process.

[0224]

[Filtering process for R2W2 case]

Next, the filtering process of luminance signals for the R2W2 case will be described. Here, R2 represents the number of pixels referred to and W2 represents the number of pixels to which the filtering process is applied. That is, R2W2 represents that the filtering process is applied to two pixels on the inter-LCU line boundary by referring to two pixels on the inter-LCU line boundary as illustrated in Fig. 19.

[0225]

In the example of Fig. 19, it is illustrated that pixels on the third and fourth columns from the line boundary BB in the block BKu in Fig. 16 could not be referred to.

[0226]

As illustrated again for comparison, in the strong filtering of the luminance signal in the HEVC scheme, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (50) to (55). Expressions (50) to (55) correspond to Expressions (37) to (42), respectively.

[0227]

[Mathematical formula 15]

$$p0_0 = Clip_{0-255}((p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(50)$$
$$q0_0 = Clip_{0-255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(51)$$
$$p1_0 = Clip_{0-255}((p2_i+p1_i+p0_i+q0_i+2)>>2); \quad i=0,7 \quad \cdots(52)$$
$$q1_0 = Clip_{0-255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(53)$$
$$p2_0 = Clip_{0-255}((2*p3_i+3*p2_i+p1_i+p0_i+q0_i+4)>>3); \quad i=0,7 \quad \cdots(54)$$
$$q2_0 = Clip_{0-255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(55)$$

[0228]

In contrast, in the strong filtering of the luminance signal in the R2W2 case of the present technique, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (56) to (60).

[0229]

[Mathematical formula 16]

$$p0_0 = Clip_{0-255}((4*p1_i+p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(56)$$
$$q0_0 = Clip_{0-255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(57)$$
$$p1_0 = Clip_{0-255}((3*p1_i+q0_i+2)>>2); \quad i=0,7 \quad \cdots(58)$$
$$q1_0 = Clip_{0-255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(59)$$
$$q2_0 = Clip_{0-255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(60)$$

[0230]

Here, Expression (56) of the strong filtering of the R2W2 case is different in that "$p2_i + 2*p1_i + 2*p0_i$" in Expression (50) of the strong filtering of the HEVC scheme is changed to "$4*p1_i + p0_i$" according to Expression (49) of linear approximation.

[0231]

Expression (58) of the strong filtering of the R2W2 case is different in that "$p2_i + 2*p1_i + p0_i$" in Expression (52) of the strong filtering of the HEVC scheme is changed to "$3*p1_i$" according to Expression (49) of linear approximation.

[0232]

The strong filtering of the R2W2 case is different in that an expression corresponding to Expression (54) of the strong filtering of the HEVC scheme is deleted.

[0233]

That is, in the R2W2 case of the present technique, since the pixel "$p2_i$" on the third row from the line boundary BB of the block Bku is not referred to, Expression (49) of linear approximation is substituted and used instead.

[0234]

Moreover, the filtering process is not applied to the pixel "$p2_0$" on the third row from the line boundary BB of the block Bku. Therefore, in the strong filtering of the R2W2 case, an expression corresponding to Expression (54) of the strong filtering of the HEVC scheme is deleted.

[0235]

Next, a weak filtering determination expression for luminance signals at a line boundary and weak filtering will be described.

[0236]

First, in the weak filtering determination expression for luminance signals in the HEVC scheme and the weak filtering, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (61) to (63). The first terms of Expressions (61) to (63) are weak filtering determination expressions.

[0237]

[Mathematical formula 17]

$$\text{if}(\text{abs}(\text{delta}) < \text{iThrCut}),\ \text{delta} = (9*(q0_i - p0_i) - 3*(q1_i - p1_i))$$

$$\cdot\ p0_0 = \text{Clip}_{0\text{-}255}(p0_i + \text{Clip}_{(-tc)\text{-}tc}(\text{delta})); \qquad i = 0,7$$

$$\cdot\ q0_0 = \text{Clip}_{0\text{-}255}(q0_i - \text{Clip}_{(-tc)\text{-}tc}(\text{delta})); \qquad i = 0,7 \quad \cdots(61)$$

$$\text{if}(\text{abs}((p2_2 - 2*p1_2 + p0_2) + \text{abs}(p2_5 - 2*p1_5 + p0_5) < \text{iSideThreshold}))$$

$$\cdot\ p1_0 = \text{Clip}_{0\text{-}255}(p1_i + \text{Clip}_{(-tc2)\text{-}tc2}((((p2_i + p0_i + 1) >> 1) - p1_i + \text{delta}) >> 1); \quad i = 0,7 \quad \cdots(62)$$

$$\text{if}(\text{abs}((q2_2 - 2*q1_2 + q0_2) + \text{abs}(q2_5 - 2*q1_5 + q0_5) < \text{iSideThreshold}))$$

$$\cdot\ q1_0 = \text{Clip}_{0\text{-}255}(q1_i + \text{Clip}_{(-tc2)\text{-}tc2}((((q2_i + q0_i + 1) >> 1) - q1_i - \text{delta}) >> 1); \quad i = 0,7 \quad \cdots(63)$$

Here, as described above, "$\text{Clip}_{0\text{-}255}$" indicates a clipping process of rounding a value up to "0" when the value is "0" or smaller and rounding a value down to "255" when the value is "255" or larger. Moreover, $\text{Clip}_{(-tc)\text{-}tc}$ indicates a clipping process of rounding a value up to "-tc" when the value is "-tc" or smaller and rounding a value down to "tc" when the value is "tc" or larger. The "tc2" clipping is the same. The same is applied to the following expressions. Here, "tc" is a value set based on the parameter Q as indicated in Table 1.

[0238]

In contrast, in the weak filtering determination expression for luminance signals in the R2W2 case and the weak filtering, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (64) to (66). The first terms of Expressions (64) to (66) are weak filtering determination expressions.

[0239]

[Mathematical formula 18]

$$\text{if}(\text{abs}(\text{delta}) < \text{iThrCut}),\ \text{delta} = (9*(q0_i - p0_i) - 3*(q1_i - p1_i))$$

$$\cdot\ p0_0 = \text{Clip}_{0-255}(p0_i + \text{Clip}_{(-tc)-tc}(\text{delta})); \qquad i = 0,7$$

$$\cdot\ q0_0 = \text{Clip}_{0-255}(q0_i - \text{Clip}_{(-tc)-tc}(\text{delta})); \qquad i = 0,7 \quad \cdots(64)$$

$$\text{if}(\text{abs}(0-(p1_2-p0_2)) + \text{abs}(0-(p1_5-p0_5)) < \text{iSideThreshold}))$$

$$\cdot\ p1_0 = \text{Clip}_{0-255}(p1_i + \text{Clip}_{(-tc2)-tc2}((((p1_i+p1_i+1)>>1) - p1_i + \text{delta})>>1)); \quad i = 0,7 \quad \cdots(65)$$

$$\text{if}(\text{abs}((q2_2 - 2*q1_2 + q0_2) + \text{abs}(q2_5 - 2*q1_5 + q0_5) < \text{iSideThreshold}))$$

$$\cdot\ q1_0 = \text{Clip}_{0-255}(q1_i + \text{Clip}_{(-tc2)-tc2}((((q2_i+q0_i+1)>>1) - q1_i - \text{delta})>>1)); \quad i = 0,7 \quad \cdots(66)$$

[0240]

Here, an expression within the tc2 clipping of the second term which is a weak filtering expression within Expression (65) of the R2W2 case is different from an expression within the tc2 clipping of the second term in Expression (62) of the HEVC scheme according to Expression (49) of linear approximation. That is, the expression "((($p2_i$ + $p0_i$ + 1)>>1)-$p1_i$ + delta)>>1)" within the tc2 clipping of the second term of the HEVC scheme is changed to the expression "((($p1_i$ + $p1_i$ + 1)>>1)-$p1_i$ + delta)>>1)" within the tc2 clipping of the second term.

[0241]

Moreover, an expression within the if clause of the first term which is a weak filtering determination expression within Expression (65) of the R2W2 case is different from an expression within the if clause of the first term in Expression (62) of the HEVC scheme. That is, the expression "abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ - 2*$p1_5$ + $p0_5$)" within the if clause of the first term of the HEVC scheme is changed to the expression "abs(0 - ($p1_2$ - $p0_2$)) + abs(0 - ($p1_5$ - $p0_5$))" within the if clause of the first term.

[0242]

That is, in the first term of Expression (65), a pixel immediately below a current pixel is padded as indicated in Expression (67) below rather than linearly approximating the same. In other words, "0" in the expression "abs(0 - ($p1_2$ - $p0_2$))" represents "0 = $p2_2$ - $p1_2$" and "0" in the expression "abs(0 - ($p1_5$ - $p0_5$))" represents "0 = $p2_5$ - $p1_5$."

[0243]

[Mathematical formula 19]

$$p2_i \doteqdot p1_i \quad \cdots(67)$$

This is because as for inclination determination in the filtering determination expression, the use of linear approximation allows to ignore determination which actually cannot be ignored. Thus, in the inclination determination of the weak filtering determination expression, a pixel immediately below a current pixel is padded as indicated in Expression (67). The same can be said for the strong filtering determination expression and the inter-block boundary determination expression described below.

[0244]

Next, an inter-block boundary determination expression and a strong filtering determination expression for luminance signals at a line boundary will be described.

[0245]

First, an inter-block boundary determination expression for luminance signals in the HEVC scheme is expressed by Expression (68) below and a strong filtering determination expression is expressed by Expression (69) below.

[Mathematical formula 20]

$$\text{if}(\text{abs}((p2_2-2*p1_2+p0_2)+\text{abs}(p2_5-2*p1_5+p0_5)+\text{abs}((q2_2-2*q1_2+q0_2)+\text{abs}((q2_5-2*q1_5+q0_5)+<\beta) \quad \cdots(68)$$

[Mathematical formula 21]

$$d<(\beta>>2),\quad d=\text{abs}((p2_2-2*p1_2+p0_2)+\text{abs}(p2_5-2*p1_5+p0_5)+\text{abs}((q2_2-2*q1_2+q0_2)+\text{abs}(q2_5-2*q1_5+q0_5)$$
$$\text{and}\,(|p3_i-p0_i|+|q0_i-q3_i|)<(\beta>>3$$
$$\text{and}\,|p0_i-q0_i|<((5*t_c+1)>>1) \quad \cdots(69)$$

[0246]

In contrast, an inter-block boundary determination expression in the R2W2 case of the present technique is expressed by Expression (70) below and a strong filtering determination expression is expressed by Expression (71) below.

[Mathematical formula 22]

$$\text{if}(\text{abs}(0-(p1_2-p0_2))+\text{abs}(0-(p1_5-p0_5))+\text{abs}((q2_2-2*q1_2+q0_2)+\text{abs}(q2_5-2*q1_5+q0_5)<\beta) \quad \cdots(70)$$

[Mathematical formula 23]

$$d<(\beta>>2),\quad d=\text{abs}(0-(p1_2-p0_2))+\text{abs}(0-(p1_5-p0_5))+\text{abs}((q2_2-2*q1_2+q0_2)+\text{abs}(q2_5-2*q1_5+q0_5)$$
$$\text{and}\,(|(p1_i-p0_i)<<1|+|q0_i-q3_i|)<(\beta>>3)$$
$$\text{and}\,|p0_i-q0_i|<((5*t_c+1)>>1) \quad \cdots(71)$$

[0247]

The first and second terms in the if clause of Expression (70) for inter-block boundary determination of the R2W2 case are different from the first and second terms in the if clause of Expression (68) for inter-block boundary determination of the HEVC scheme according to Expression (67) for padding. That is, the first and

second terms "abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ - 2*$p1_5$ + $p0_5$)" in the if clause of the HEVC scheme are changed to the first and second terms "abs(0 - ($p1_2$ - $p0_2$)) + abs(0 - ($p1_5$ - $p0_5$))" in the if clause.

[0248]

Moreover, an expression in the absolute value on the first row of Expression (71) for strong filtering determination of the R2W2 case is different from an expression in the absolute value on the first row of Expression (69) for strong filtering determination in the HEVC scheme according to Expression (67) for padding. That is, the expression "abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ - 2*$p1_5$ + $p0_5$)" in the absolute value on the first row of Expression (69) is changed to the expression "abs(0 - ($p1_2$ - $p0_2$)) + abs(0 - ($p1_5$ - $p0_5$))" in the absolute value on the first row of Expression (71).

[0249]

Further, an expression appearing after the first "and" of Expression (71) for the strong filtering determination of the R2W2 case is different from an expression appearing after the first "and" of Expression (69) for the strong filtering determination in the HEVC scheme. That is, the expression "(|$p3_i$ - $p0_i$ << 1| + |$q0_i$ - $q3_i$|) < (β >> 3))" appearing after the first "and" of Expression (69) is changed to the expression "(|$p1_i$ - $p0_i$ << 1| + |$q0_i$ - $q3_i$|) < (β >> 3))" appearing after the first "and" of Expression (71).

[0250]

Since the determination of this part is determination of the magnitudes of pixel values, Expression (49) for linear approximation and Expression

(67) for padding are used. That is, the expression appearing after the first "and" of Expression (71) is generated in such a manner that "$p3_i$" in the expression appearing after the first "and" of Expression (69) is first approximated as "$p2_i$" by padding and "$p2_i$" is approximated as "$2*p1_i - p0_i$" by linear approximation.

[0251]

[Filtering process for R2W1 case]

Next, the filtering process of luminance signals for the R2W1 case will be described. Here, R2 represents the number of pixels referred to and W1 represents the number of pixels to which the filtering process is applied. That is, R2W1 represents that the filtering process is applied to one pixel on the inter-LCU line boundary by referring to two pixels on the inter-LCU line boundary as illustrated in Fig. 19.

[0252]

In the strong filtering of the luminance signal in the R2W1 case of the present technique, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (72) to (75).

[0253]

[Mathematical formula 24]

$$p0_0 = Clip_{0\text{-}255}((4*p1_i+p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(72)$$

$$q0_0 = Clip_{0\text{-}255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(73)$$

$$q1_0 = Clip_{0\text{-}255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(74)$$

$$q2_0 = Clip_{0\text{-}255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(75)$$

[0254]

Here, Expression (72) of the strong filtering of the R2W1 case is different in that "$p2_i + 2*p1_i + 2*p0_i$"

in Expression (50) of the strong filtering of the HEVC scheme is changed to "$4*p1_i + p0_i$" according to Expression (49) of linear approximation.

[0255]

The strong filtering of the R2W1 case is different in that expressions corresponding to Expressions (52) and (54) of the strong filtering of the HEVC scheme are deleted.

[0256]

That is, in the present technique, since the pixel "$p2_i$" on the third row from the line boundary BB of the block Bku is not referred to, Expression (49) of linear approximation is substituted and used instead.

[0257]

Moreover, the filtering process is not applied to the pixels "$p1_0$" and "$p2_0$" on the second and third rows from the line boundary BB of the block Bku. Therefore, in the strong filtering of the R2W1 case, expressions corresponding to Expressions (52) and (54) of the strong filtering of the HEVC scheme is deleted.

[0258]

Next, a weak filtering determination expression for luminance signals at a line boundary and weak filtering will be described.

[0259]

In the weak filtering determination expression for luminance signals in the R2W1 case and the weak filtering, the luminance components of the respective pixels in the filter processing range are calculated by performing the operations of Expressions (76) and (77).

[0260]

[Mathematical formula 25]

$$\begin{aligned}&\text{if}(\text{abs}(\text{delta}) < \text{iThrCut}),\ \text{delta} = (9*(q0_i - p0_i) - 3*(q1_i - p1_i))\\&\quad p0_0 = \text{Clip}_{0\text{-}255}(p0_i + \text{Clip}_{(-tc)\text{-}tc}(\text{delta})); \qquad i = 0,7\\&\quad q0_0 = \text{Clip}_{0\text{-}255}(q0_i - \text{Clip}_{(-tc)\text{-}tc}(\text{delta})); \qquad i = 0,7 \quad \cdots(76)\\&\text{if}(\text{abs}((q2_2 - 2*q1_2 + q0_2) + \text{abs}(q2_5 - 2*q1_5 + q0_5) < \text{iSideThreshold}))\\&\quad \cdot\ q1_0 = \text{Clip}_{0\text{-}255}(q1_i + \text{Clip}_{(-tc2)\text{-}tc2}((((q2_i + q0_i + 1) >> 1) - q1_i - \text{delta}) >> 1);\ i = 0,7 \quad \cdots(77)\end{aligned}$$

[0261]

Here, in the R2W1 case, the filtering process is not applied to the pixel "$p1_0$" on the second row from the line boundary BB of the block Bku. Therefore, in the weak filtering determination expression for the R2W1 case and the weak filtering, an expression corresponding to Expression (62) of the HEVC scheme is deleted.

[0262]

Next, an inter-block boundary determination expression and a strong filtering determination expression for luminance signals at a line boundary will be described.

[0263]

An inter-block boundary determination expression in the R2W1 case is expressed by Expression (78) below and a strong filtering determination expression is expressed by Expression (79) below.

[Mathematical formula 26]

$$\begin{aligned}&\text{if}(\text{abs}(0 - (p1_2 - p0_2)) + \text{abs}(0 - (p1_5 - p0_5)) + \text{abs}((q2_2 - 2*q1_2 + q0_2)\\&+ \text{abs}(q2_5 - 2*q1_5 + q0_5) < \beta)\end{aligned} \quad \cdots(78)$$

[Mathematical formula 27]

$$d<(\beta>>2),\quad d=abs(0-(p1_2-p0_2))+abs(0-(p1_5-p0_5))+$$
$$abs((q2_2-2*q1_2+q0_2)+abs(q2_5-2*q1_5+q0_5)$$
$$and\,(|(p1_i-p0_i)<<1|+|q0_i-q3_i|)<(\beta>>3)$$
$$and\,|p0_i-q0_i|<((5*t_c+1)>>1) \quad \cdots(79)$$

[0264]

The first and second terms in the if clause of Expression (78) for inter-block boundary determination of the R2W1 case are different from the first and second terms in the if clause of Expression (68) for inter-block boundary determination of the HEVC scheme according to Expression (67) for padding. That is, the first and second terms "abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ - 2*$p1_5$ + $p0_5$)" in the if clause of the HEVC scheme are changed to the first and second terms "abs(0 - ($p1_2$ - $p0_2$)) + abs(0 - ($p1_5$ - $p0_5$))" in the if clause.

[0265]

Moreover, an expression in the absolute value on the first row of Expression (79) for strong filtering determination of the R2W1 case is different from an expression in the absolute value on the first row of Expression (69) for strong filtering determination in the HEVC scheme according to Expression (67) for padding. That is, the expression "abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ - 2*$p1_5$ + $p0_5$)" in the absolute value on the first row of Expression (69) is changed to the expression "abs(0 - ($p1_2$ - $p0_2$)) + abs(0 - ($p1_5$ - $p0_5$))" in the absolute value on the first row of Expression (79).

[0266]

Further, an expression appearing after the first "and" of Expression (79) for the strong filtering determination of the R2W1 case is different from an

expression appearing after the first "and" of Expression (69) for the strong filtering determination in the HEVC scheme. That is, the expression "($|p3_i - p0_i << 1| + |q0_i - q3_i|) < (\beta >> 3)$)" appearing after the first "and" of Expression (69) is changed to the expression "($|p1_i - p0_i << 1| + |q0_i - q3_i|) < (\beta >> 3)$)" appearing after the first "and" of Expression (79).

[0267]

As described above in connection with the R2W2 case, since the determination of this part is determination of the magnitudes of pixel values, Expression (49) for linear approximation and Expression (67) for padding are used. That is, the expression appearing after the first "and" of Expression (79) is generated in such a manner that "$p3_i$" in the expression appearing after the first "and" of Expression (69) is first approximated as "$p2_i$" and "$p2_i$" is approximated as "$2*p1_i - p0_i$."

[0268]

A configuration example and an operation for realizing the above determination process based on linear approximation will be described with reference to Figs. 22 to 24 as sixth and seventh embodiments of the deblocking filtering unit.

[0269]

As described above, in the present technique, the inter-block boundary determination at the line boundary and the strong filtering determination expression are changed using Expression (49) for linear approximation and Expression (67) for padding. Thus, a clipping process is added to a strong filtering process as a countermeasure to a case where the determination

expression is erroneously ignored and the deblocking filtering process is erroneously performed.

[0270]

[Description of present technique (Clipping process in strong filtering)]

Next, a clipping process in strong filtering which is an eighth embodiment will be described. The strong filtering for luminance signals in the R2W2 case may also use Expressions (80) to (84) in which a clipping process is added.

[Mathematical formula 28]

$$p0_0 = Clip_{0-255}(p0_0 + Clip_{(-tc)-tc}(((4*p1_i + p0_i + 2*q0_i + q1_i + 4) >> 3) - p0_0)); \quad i = 0,7 \quad \cdots(80)$$

$$q0_0 = Clip_{0-255}(p0_0 + Clip_{(-tc)-tc}(((p1_i + 2*p0_i + 2*q0_i + 2*q1_i + q2_i + 4) >> 3) - p0_0)); \quad i = 0,7 \quad \cdots(81)$$

$$p1_0 = Clip_{0-255}(p1_0 + Clip_{(-tc)-tc}(((3*p1_i + q0_i + 2) >> 2) - p1_0)); \quad i = 0,7 \quad \cdots(82)$$

$$q1_0 = Clip_{0-255}(q1_0 + Clip_{(-tc)-tc}(((p0_i + q0_i + q1_i + q2_i + 2) >> 2) - q1_0)); \quad i = 0,7 \quad \cdots(83)$$

$$q2_0 = Clip_{0-255}(q2_0 + Clip_{(-tc)-tc}(((p1_i + q0_i + q1_i + 3*q2_i + 2*q3_i + 4) >> 3) - q2_0)); \quad i = 0,7 \quad \cdots(84)$$

[0271]

Expression (80) for strong filtering is a calculation expression for $p0_0$ after filtering. In Expression (80) for strong filtering, a clipping process of (-tc) to tc is applied to a difference between the expression in the clipping process of 0 to 255 in Expression (56) and $p0_0$ to obtain a value, the value is added to $p0_0$ to obtain another value, and the other value is subjected to a clipping process of 0 to 255.

[0272]

Expression (81) for strong filtering is a calculation expression for $q0_0$ after filtering. In Expression (81) for strong filtering, a clipping process of (-tc) to tc is applied to a difference between the expression in the clipping process of 0 to 255 in

Expression (57) and $q0_0$ to obtain a value, the value is added to $q0_0$ to obtain another value, and the other value is subjected to a clipping process of 0 to 255.

[0273]

Expression (82) for strong filtering is a calculation expression for $p1_0$ after filtering. In Expression (82) for strong filtering, a clipping process of (-tc) to tc is applied to a difference between the expression in the clipping process of 0 to 255 in Expression (58) and $p1_0$ to obtain a value, the value is added to $p1_0$ to obtain another value, and the other value is subjected to a clipping process of 0 to 255.

[0274]

Expression (83) for strong filtering is a calculation expression for $q1_0$ after filtering. In Expression (83) for strong filtering, a clipping process of (-tc) to tc is applied to a difference between the expression in the clipping process of 0 to 255 in Expression (59) and $q1_0$ to obtain a value, the value is added to $q1_0$ to obtain another value, and the other value is subjected to a clipping process of 0 to 255.

[0275]

Expression (84) for strong filtering is a calculation expression for $q2_0$ after filtering. In Expression (84) for strong filtering, a clipping process of (-tc) to tc is applied to a difference between the expression in the clipping process of 0 to 255 in Expression (60) and $q2_0$ to obtain a value, the value is added to $q2_0$ to obtain another value, and the other value is subjected to a clipping process of 0 to 255.

[0276]

By doing so, it is possible to suppress excessive application of the filtering process.

[0277]

Here, Matthias Narroschke, Tomas Wedi, Semih Esenlik, "Results for modified decisions for deblocking," JCTVC-G590, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 7th Meeting, Geneva, CH, 21-30 November, 2011 (hereinafter referred to as Non-Patent Document 2) is proposed.

[0278]

In Non-Patent Document 2, it is described that a strong/weak filtering determination process is performed in respective block units. However, in the determination process described in Non-Patent Document 2, when determination results on a current line and a block unit to which the line belongs are different (for example, when weak filtering is determined for the current line whereas strong filtering is determined for a block that includes the line, article information is applied to the current line), an excessive filtering process is applied.

[0279]

Moreover, in the Non-Patent Document 2, since the strong/weak filtering determination process is performed in units of four lines, a mismatch occurs between the determination and the filtering process. As an example of the mismatch, weak filtering is determined although it is necessary to determine strong filtering and strong filtering is determined although it is necessary to determine weak filtering.

[0280]

In strong filtering of luminance signals in the

HEVC scheme (HM-5.0), Expressions (85) to (89) below are employed.

[Mathematical formula 29]

$$p0_i=(p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3; \quad i=0,7 \quad \cdots(85)$$
$$q0_i=(p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3; \quad i=0,7 \quad \cdots(86)$$
$$p1_i=(3*p1_i+q0_i+2)>>2; \quad i=0,7 \quad \cdots(87)$$
$$q1_i=(p0_i+q0_i+q1_i+q2_i+2)>>2; \quad i=0,7 \quad \cdots(88)$$
$$q2_i=(p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3; \quad i=0,7 \quad \cdots(89)$$

[0281]

In the present technique, in order to eliminate a mismatch of the determination process and an incompleteness in the determination process, the clipping process indicated in Expressions (90) to (94) below. In Expression (90), since a clipping process is to be performed on a partial Δ-value that changes with the filtering process, $p0_i$ is outside the clipping process. The same is applied to Expressions (91) to (94).

[Mathematical formula 30]

$$p0_i=p0_i+Clip_{(-pv)-(pv)}((p2_i+2*p1_i-6*p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(90)$$
$$q0_i=q0_i+Clip_{(-pv)-(pv)}((p1_i+2*p0_i-6*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(91)$$
$$p1_i=p1_i+Clip_{(-pv)-(pv)}((-p1_i+q0_i+2)>>2); \quad i=0,7 \quad \cdots(92)$$
$$q1_i=q1_i+Clip_{(-pv)-(pv)}((p0_i+q0_i-3*q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(93)$$
$$q2_i=q2_i+Clip_{(-pv)-(pv)}((p0_i+q0_i+q1_i-5*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(94)$$

Here, a clipping value "pv" is any one of 1 time, 2 times, 3 times, 4 times, 5 times, 6 times, 7 times, and 8 times of the parameter tc. That is, pv = tc, 2*tc, 3*tc, 4*tc, 5*tc, 6*tc, 7*tc, 8*tc. Although the clipping value "pv" is preferably 2 times of the parameter tc, the clipping value may be any one of 1 time to 8 times of the parameter. Moreover, the clipping value is not limited to 1 time to 8 times as long as the same effects are

obtained.

[0282]

Moreover, "tc" changes with the quantization parameter QP. Thus, the multiplication factor of the clipping value applied to "tc" may also be increased (that is, be changed with QP) as QP increases. The parameter (the clipping value or the value of a multiplication factor of the clipping value applied to "tc") applied to the clipping process may be set in advance and may be added to an encoded stream and transmission terminator to a decoding side.

[0283]

Expressions (90) to (94) described above may be modified so that the clipping process is performed as in Expressions (95) to (99) below.

[Mathematical formula 31]

$$p0_i = Clip_{0\text{-}255}(p0_i + Clip_{(-pv)\text{-}(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3)); \quad i = 0,7 \quad \cdots(95)$$

$$q0_i = Clip_{0\text{-}255}(q0_i + Clip_{(-pv)\text{-}(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3)); \quad i = 0,7 \quad \cdots(96)$$

$$p1_i = Clip_{0\text{-}255}(p1_i + Clip_{(-pv)\text{-}(pv)}((-p1_i + q0_i + 2) >> 2)); \quad i = 0,7 \quad \cdots(97)$$

$$q1_i = Clip_{0\text{-}255}(q1_i + Clip_{(-pv)\text{-}(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2)); \quad i = 0,7 \quad \cdots(98)$$

$$q2_i = Clip_{0\text{-}255}(q2_i + Clip_{(-pv)\text{-}(pv)}((p0_i + q0_i + q1_i - 5*q2_i + 2*q3_i + 4) >> 3)); \quad i = 0,7 \quad \cdots(99)$$

[0284]

Here, Expressions (85) to (89) for the strong filtering of luminance signals of the HEVC scheme (HM-5.0) are expressions for strong filtering of luminance signals at a line boundary. That is, in the HEVC scheme (HM-5.0), the strong filtering of luminance signals is actually expressed by Expressions (100) to (105) below.

[Mathematical formula 32]

$$p0_i = (p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3; \quad i = 0,7 \quad \cdots (100)$$

$$q0_i = (p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3; \quad i = 0,7 \quad \cdots (101)$$

$$p1_i = (p2_i+p1_i+p0_i+q0_i+2)>>2; \quad i = 0,7 \quad \cdots (102)$$

$$q1_i = (p0_i+q0_i+q1_i+q2_i+2)>>2; \quad i = 0,7 \quad \cdots (103)$$

$$p2_i = (2*p3_i+3*p2_i+p1_i+p0_i+q0_i+4)>>3; \quad i = 0,7 \quad \cdots (104)$$

$$q2_i = (p0_i+q0_i+p1_i+3*q2_i+2*q3_i+4)>>3; \quad i = 0,7 \quad \cdots (105)$$

[0285]

In the present technique, in order to eliminate a mismatch of the determination process and an incompleteness of the determination process, a clipping process may be applied to the strong filtering for luminance signals as indicated in Expressions (106) to (111) so as to correspond to Expressions (100) to (105). In the first expression of Expression (106), similarly to Expressions (90) to (94) described above, since a clipping process is to be performed on the partial Δ-value that changes with the filtering process, "$p0_i$" is outside the clipping process. The same is applied to the first expressions of Expressions (107) to (111).

[0286]

[Mathematical formula 33]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3); \quad i = 0, 7$$
$$= Clip_{(-pv)-(pv)}((p2_i + 2*p1_i + 2*p0_i + 2*q0_i + q1_i + 4) >> 3); \quad \cdots (106)$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3); \quad i = 0, 7$$
$$= Clip_{(-pv)-(pv)}((p1_i + 2*p0_i + 2*q0_i + 2*q1_i + q2_i + 4) >> 3); \quad \cdots (107)$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2); \quad i = 0, 7$$
$$= Clip_{(-pv)-(pv)}((p2_i + p1_i + p0_i + q0_i + 2) >> 2); \quad \cdots (108)$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2); \quad i = 0, 7$$
$$= Clip_{(-pv)-(pv)}((p0_i + q0_i + q1_i + q2_i + 2) >> 2); \quad \cdots (109)$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3); \quad i = 0, 7$$
$$= Clip_{(-pv)-(pv)}((2*p3_i + 3*p2_i + p1_i + p0_i + q0_i + 4) >> 3); \quad \cdots (110)$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3); \quad i = 0, 7$$
$$= Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i + 3*q2_i + 2*q3_i + 4) >> 3); \quad \cdots (111)$$

Here, a clipping value "pv" is any one of 1 time, 2 times, 3 times, 4 times, 5 times, 6 times, 7 times, and 8 times of the parameter tc. That is, pv = tc, 2*tc, 3*tc, 4*tc, 5*tc, 6*tc, 7*tc, 8*tc. Although the clipping value "pv" is preferably 2 times of the parameter tc, the clipping value may be any one of 1 time to 8 times of the parameter. Moreover, the clipping value is not limited to 1 time to 8 times as long as the same effects are obtained.

[0287]

Moreover, "tc" changes with the quantization parameter QP. Thus, the multiplication factor of the clipping value applied to "tc" may also be increased (that is, be changed with QP) as QP increases. The parameter (the clipping value or the value of a multiplication factor of the clipping value applied to "tc") applied to the clipping process may be set in

advance and may be added to an encoded stream and transmission terminator to a decoding side.

[0288]

Fig. 20 is a diagram illustrating the largest values of the pixel values that change with the strong filtering of luminance signals in the HEVC scheme (HM-5.0) of Expressions (100) to (105) described above, obtained through an experiment.

[0289]

AI_HE, RA_HE, LB_HE, and LP_HE indicate conditions of this experiment. AI_HE indicates that the experiment was performed in ALL INTRA High Efficiency mode. RA_HE indicates that the experiment was performed in Random Access High Efficiency mode. LB_HE indicates that the experiment was performed in Low Delay B High Efficiency mode. LP_HE indicates that the experiment was performed in Low Delay P High Efficiency mode.

[0290]

"22, 27, 32, and 37" appearing under AI_HE, RA_HE, LB_HE, and LP_HE are values of the quantization parameters used in the experiment. Moreover, Class A to Class F indicate the type of test sequence used in the experiment.

[0291]

As illustrated in Fig. 20, a pixel value that changes with the strong filtering of luminance signals in the HEVC scheme (HM-5.0) (that is, a pixel value that changes with the strong filtering when the clipping process is not present) changes greatly by 100 or more in some cases.

[0292]

Therefore, a clipping process is applied to the strong filtering of luminance signals. Due to this, as illustrated in Fig. 21, it is possible to suppress the influence of a mismatch of the determination process and an incompleteness of the determination process as much as possible.

[0293]

In the example of Fig. 21, when there is an edge (that is not a block boundary) indicated by a solid line, although a deblocking process is not to be performed, if the deblocking process is performed according to the technique of Non-Patent Document 2, the pixel values are changed greatly as indicated by dot lines.

[0294]

For example, as indicated by solid line, before the deblocking process is performed, the values of the pixels $p2_i$, $p1_i$, and $p0_i$ are 255, the values of the pixels $q0_i$, $q1_i$, and $q2_i$ are 0, and the difference D between the values of the pixels $p0_i$ and $q0_i$ is 255.

[0295]

In contrast, as indicated by dot line, after the deblocking process is performed, the value of the pixel $p2_i$ is 223, the value of the pixel $p1_i$ is 191, and the value of the pixel $p0_i$ is 159. Thus, the pixel values of the pixels $p2_i$, $p1_i$, and $p0_i$ are changed greatly. Moreover, after the deblocking process is performed, the value of the pixel $q0_i$ is 96, the value of the pixel $q1_i$ is 64, and the value of the pixel $q2_i$ is 32. Thus, the pixel values of the pixels $q0_i$, $q1_i$, and $q2_i$ are changed greatly.

[0296]

In such a case, in the strong filtering of Expressions (106) to (111) described above, a clipping process of which the clipping value is 10 is performed.

[0297]

In contrast, as indicated by bold line, after the deblocking process is performed, the values of the pixels $p2_i$, $p1_i$, and $p0_i$ are 245, and the values of the pixels $q0_i$, $q1_i$, and $q2_i$ are 255. Thus, the change in the pixel values can be suppressed as much as possible.

[0298]

A configuration example and an operation of the clipping process in the strong filtering will be described with reference to Figs. 25 to 27 described later as an eighth embodiment of the deblocking filtering unit.

[0299]

[13. Sixth embodiment of deblocking filtering unit]

[Configuration example of deblocking filtering unit]

Fig. 22 illustrates the configuration of the sixth embodiment of the deblocking filtering unit. The deblocking filtering unit 24 is configured to include an image memory 71, a block boundary determining unit 72-1, a filter strength determining unit 73-1, a filter operation unit 74-1, a selector 75, a coefficient memory 76, and a filter controller 77. The deblocking filtering unit 24 is configured to further include a line boundary/block boundary determining unit 72-2, a line boundary filter strength determining unit 73-2, and a line boundary filter operation unit 74-2.

[0300]

The image memory 71 is a unit that corresponds to

the line memory 241 of Fig. 9 and is configured as a line memory. The image memory 71 stores image data supplied from the adder 23. The image memory 71 reads the image data stored therein and supplies the read image data to the block boundary determining unit 72-1, the filter strength determining unit 73-1, and the filter operation unit 74-1. Moreover, the image memory 71 reads the image data stored therein and supplies the read image data to the line boundary/block boundary determining unit 72-2, the line boundary filter strength determining unit 73-2, and the line boundary filter operation unit 74-2.

[0301]

At positions other than the line boundary, the image data is not stored in the image memory 71, but there may be case where the image data supplied from the adder 23 is supplied to the respective units and processed. However, in the example of Fig. 22, for the sake of convenience, it is described that the image data having passed through the image memory 71 is processed.

[0302]

The block boundary determining unit 72-1 performs inter-block boundary determination under the control of the controller 77. That is, the block boundary determining unit 72-1 performs the inter-block boundary determination process for each block using the image data read from the image memory 71 and detects an inter-block boundary in which the filtering process is performed. The block boundary determining unit 72-1 outputs the detection results to the filter strength determining unit 73-1.

[0303]

The filter strength determining unit 73-1 determines the filter strength in the above-described manner under the control of the controller 77. That is, the filter strength determining unit 73-1 determines whether the filtering process will be performed in a strong filtering mode or a weak filtering mode for each line using the image data of two blocks with an inter-block boundary interposed, supplied from the image memory 71 and outputs the determination results to the filter operation unit 74-1.

[0304]

The filter operation unit 74-1 performs a filtering operation with the filter strength determined by the filter strength determining unit 73-1 for each line under the control of the controller 77 using the image data stored in the image memory 71 and the filter coefficients read from the coefficient memory 76. The filter operation unit 74-1 outputs the image data having been subjected to the filtering process to the selector 75.

[0305]

When the boundary is an inter-LCU line boundary, the line boundary/block boundary determining unit 72-2 performs inter-block boundary determination under the control of the controller 77. That is, the line boundary/block boundary determining unit 72-2 performs the inter-block boundary determination process for each block using the image data read from the image memory 71 and detects an inter-block boundary in which the filtering process is performed. The line boundary/block boundary determining unit 72-2 outputs the detection results to the line boundary filter strength determining

unit 73-2.

[0306]

When the boundary is an inter-LCU line boundary, the line boundary filter strength determining unit 73-2 determines the filter strength in the above-described manner under the control of the controller 77. That is, the line boundary filter strength determining unit 73-2 determines whether the filtering process will be performed in a strong filtering mode or a weak filtering mode for each line using the image data of two blocks adjacent with an inter-block boundary interposed, supplied from the image memory 71 and outputs the determination results to the line boundary filter operation unit 74-2.

[0307]

The line boundary filter operation unit 74-2 performs the filtering operation with the filter strength determined by the line boundary filter strength determining unit 73-2 for each line under the control of the controller 77 using the image data stored in the image memory 71 and the filter coefficients read from the coefficient memory 76. The line boundary filter operation unit 74-2 outputs the image data having been subjected to the filtering process to the selector 75.

[0308]

The selector 75 selects any one of the image data from the filter operation unit 74-1 and the image data from the line boundary filter operation unit 74-2 under the control of the controller 77 and outputs the selected image data to the frame memory 25.

[0309]

The coefficient memory 76 stores filter coefficients used for the filtering operation of the deblocking filtering process. The coefficient memory 76 reads the filter coefficients stored therein and supplies the read filter coefficients to the filter operation unit 74-1 and the line boundary filter operation unit 74-2.

[0310]

The controller 77 controls the respective units of the deblocking filtering unit 24. For example, the controller 77 controls the image memory 71 so as to store image data corresponding to a predetermined number of lines on the lower side in the block and reads the image data stored in the image memory 71. The controller 77 is configured to include an intra-LCU line determining unit 77A.

[0311]

The intra-LCU line determining unit 77A determines whether a boundary is an inter-LCU line boundary in respective block units in which the process is sequentially performed in a raster scan direction, and supplies the determination results to the block boundary determining unit 72-1, the filter strength determining unit 73-1, and the filter operation unit 74-1 so that the units perform the process when the boundary is not the line boundary. Moreover, the intra-LCU line determining unit 77A also supplies the determination results to the line boundary/block boundary determining unit 72-2, the line boundary filter strength determining unit 73-2, and the line boundary filter operation unit 74-2 so that the units perform the process when the boundary is the line boundary. Further, the intra-LCU line determining unit

77A supplies the determination results to the selector 75 so that the selector 75 selects the image data from the line boundary filter operation unit 74-2 when the boundary is the line boundary and selects the image data from the filter operation unit 74-1 when the boundary is not the line boundary.

[0312]

The deblocking filtering unit 24 illustrated in Fig. 22 may be configured in advance to perform the R2W2 process, for example, and may be configured so as to be able to control the filter processing range and the filtering operation range similarly to the example of Fig. 9.

[0313]

[Operation of deblocking filtering unit]

Next, the deblocking process of the deblocking filtering unit 24 of Fig. 22 will be described with reference to the flowchart of Fig. 23. In the example of Fig. 23, a case where the R2W2 process is performed will be described.

[0314]

In step S91, the intra-LCU line determining unit 77A determines whether a filter applied is a vertical filter of a LCU boundary in respective block units in which the process is sequentially performed in a raster scan direction. That is, in step S91, it is determined whether a boundary is an inter-LCU line boundary and the filter applied is a vertical filter.

[0315]

When it is determined in step S91 that the filter applied is a vertical filter of a LCU boundary, the flow

proceeds to step S92. In step S92, the line boundary/block boundary determining unit 72-2 performs inter-block boundary determination of a line boundary under the control of the controller 77 when the boundary is an inter-LCU line boundary.

[0316]

That is, the line boundary/block boundary determining unit 72-2 performs the inter-block boundary determination process for each block according to Expression (70) described above using the image data read from the image memory 71 and detects an inter-block boundary in which the filtering process is performed. The line boundary/block boundary determining unit 72-2 outputs the detection results to the line boundary filter strength determining unit 73-2.

[0317]

In step S93, the line boundary filter strength determining unit 73-2 determines the filter strength of the line boundary as described above under the control of the controller 77 when the boundary is the inter-LCU line boundary.

[0318]

That is, the line boundary filter strength determining unit 73-2 determines whether the filtering process will be performed in a strong filtering mode or a weak filtering mode using the image data of two blocks adjacent with the inter-block boundary, supplied from the image memory 71. A strong filtering mode is determined according to Expression (71) described above. Moreover, a weak filtering mode is determined according to the first terms of Expressions (64) to (66) described above.

The line boundary filter strength determining unit 73-2 outputs the determination results to the line boundary filter operation unit 74-2.

[0319]

In step S94, the line boundary filter operation unit 74-2 performs a filtering process of the line boundary under the control of the controller 77. That is, the line boundary filter operation unit 74-2 performs a filtering operation with the filter strength determined by the line boundary filter strength determining unit 73-2 using the image data stored in the image memory 71 and the filter coefficient read from the coefficient memory 76.

[0320]

In the case of the strong filtering, the filtering operation is performed according to Expressions (56) to (60) described above. Moreover, in the case of the weak filtering, the filtering operation is performed according to the second and third terms of Expressions (64) to (66) described above. The line boundary filter operation unit 74-2 outputs the image data having been subjected to the filtering process to the selector 75.

[0321]

In step S95, the intra-LCU line determining unit 77A determines whether a current line is the last eighth line of the LCU. When it is determined in step S95 that the current line is not the last eighth line of the LCU, the flow returns to step S93 and the subsequent processes are repeated.

[0322]

Moreover, when it is determined in step S95 that

the current line is the last eighth line of the LCU, the flow proceeds to step S100.

[0323]

On the other hand, when it is determined in step S91 that the filter applied is not the vertical filter of the LCU boundary, the flow proceeds to step S96. In step S96, the block boundary determining unit 72-1 performs inter-block boundary determination under the control of the controller 77.

[0324]

That is, the block boundary determining unit 72-1 performs the inter-block boundary determination process for each block according to Expression (68) described above using the image data read from the image memory 71 and detects an inter-block boundary in which the filtering process is performed. The block boundary determining unit 72-1 outputs the detection results to the filter strength determining unit 73-1.

[0325]

In step S97, the filter strength determining unit 73-1 determines the filter strength in the above-described manner under the control of the controller 77. That is, the filter strength determining unit 73-1 determines whether the filtering process will be performs in a strong filtering mode or a weak filtering mode for each line using the image data of two blocks adjacent with the inter-block boundary interposed, supplied from the image memory 71. A strong filtering mode is determined according to Expression (69) described above. A weak filtering mode is determined according to the first terms of Expressions (61) to (63) described above.

The filter strength determining unit 73-1 outputs the determination results to the filter operation unit 74-1.

[0326]

In step S98, the filter operation unit 74-1 performs a filtering operation of the block boundary under the control of the controller 77. That is, the filter operation unit 74-1 performs the filtering operation with the filter strength determined by the filter strength determining unit 73-1 for each line using the image data stored in the image memory 71 and the filter coefficients read from the coefficient memory 76.

[0327]

In the case of the strong filtering, the filtering operation is performed according to Expressions (50) to (55) described above. Moreover, in the case of the weak filtering, the filtering operation is performed according to the second and third terms of Expressions (61) to (63) described above. The filter operation unit 74-1 outputs the image data having been subjected to the filtering process to the selector 75.

[0328]

In step S99, the intra-LCU line determining unit 77A determines whether a current line is the last eighth line of the LCU. When it is determined in step S99 that the current line is not the last eighth line of the LCU, the flow returns to step S97 and the subsequent processes are repeatedly performed.

[0329]

Moreover, when it is determined in step S99 that the current line is the last eighth line of the LCU, the flow proceeds to step S100.

[0330]

In step S100, the intra-LCU line determining unit 77A determines whether a current block is the last block of the image. That is, in step S100, it is determined whether the boundary is a line boundary of another block in the screen.

[0331]

When it is determined in step S100 that the block is the last block of the image, the deblocking process ends. When it is determined in step S100 that the block is not the last block of the image, the flow returns to step S91, and the subsequent processes are repeatedly performed on the next LCU in the raster scan order.

[0332]

In the example of Fig. 22, the intra-LCU line determining unit 77A selects whether the coefficients used for the case other than the line boundary in the filter operation unit 74-1 or the coefficients for the case of the line boundary in the line boundary filter operation unit 74-2 will be used. In this manner, the filtering process at the line boundary and the filtering process at the positions other than the line boundary are switched.

[0333]

In contrast, only one type of coefficients may be used and the address of the pixel to be read may be controlled so that the filtering process at the line boundary and the filtering process at the positions other than the line boundary are switched.

[0334]

For example, calculation of $P0_0$ in the R2W2-based

strong filtering will be described as an example. In Expression (50) for the strong filtering of the HEVC scheme,

a term relating to "p(block BKu)" is "$p2_i + 2*P1_i + 2*q0_i$."

[0335]

On the other hand, in Expression (56) for the strong filtering of the HEVC scheme,

a term relating to "p(block BKu)" is "$4*P1_i + q0_i$."

[0336]

Here, $4*P1_i + q0_i = q0_i + 2*P1_i + 2*P1_i$.

[0337]

That is, the term relating to "p" in Expression (50) and the term relating to "p" in Expression (56) also use coefficients 1, 2, and 2 in that order from the left side. Thus, by replacing (changing the read-out addresses of) the pixel values to be multiplied with the coefficients using the same coefficients, it is possible to switch between the filtering process at the line boundary and the filtering process at the positions other than the line boundary.

[0338]

[14. Seventh embodiment of deblocking filtering unit]

[Configuration example of deblocking filtering unit]

Fig. 24 illustrates the configuration of a seventh embodiment of the deblocking filtering unit.

[0339]

In the example of Fig. 24, the deblocking filtering unit 24 is configured to include an image memory 71, a block boundary determining unit 72, a filter strength determining unit 73, a filter operation unit 74, a

coefficient memory 76, and a controller 77.

[0340]

In the deblocking filtering unit 24 of Fig. 24, the image memory 71, the coefficient memory 76, and the controller 77 are the same as those of the deblocking filtering unit 24 of Fig. 22.

[0341]

The deblocking filtering unit 24 of Fig. 24 is different from the deblocking filtering unit 24 of Fig. 22 in that the block boundary determining unit 72-1 and the line boundary/block boundary determining unit 72-2 are collectively changed to the block boundary determining unit 72. The deblocking filtering unit 24 of Fig. 24 is different from the deblocking filtering unit 24 of Fig. 22 in that the filter strength determining unit 73-1 and the line boundary filter strength determining unit 73-2 are collectively changed to the filter strength determining unit 73. The deblocking filtering unit 24 of Fig. 24 is different from the deblocking filtering unit 24 of Fig. 22 in that the filter operation unit 74-1 and the line boundary filter operation unit 74-2 are collectively changed to the filter operation unit 74.

[0342]

Further, the deblocking filtering unit 24 of Fig. 24 is different from the deblocking filtering unit 24 of Fig. 22 in that the intra-LCU line determining unit 77A of the controller 77 is changed to an intra-LCU line determining unit 77B.

[0343]

That is, the intra-LCU line determining unit 77B

controls the address of the pixel read from the image memory 71 as described above by way of an example of calculation of $P0_0$ in the R2W2-based strong filtering. Thus, the image memory 71 reads pixels based on the control of the intra-LCU line determining unit 77B and supplies the read pixels to the block boundary determining unit 72, the filter strength determining unit 73, and the filter operation unit 74.

[0344]

The block boundary determining unit 72 performs inter-block boundary determination. That is, the block boundary determining unit 72 performs the inter-block boundary determination process described above for each block using the image data read from the image memory 71 and detects an inter-block boundary in which the filtering process is performed. The block boundary determining unit 72 outputs the detection results to the filter strength determining unit 73.

[0345]

The filter strength determining unit 73 determines the filter strength in the above-described manner. That is, the filter strength determining unit 73 determines whether the filtering process will be performs in a strong filtering mode or a weak filtering mode for each line using the image data of two blocks adjacent with the inter-block boundary interposed, supplied from the image memory 71 and outputs the determination results to the filter operation unit 74.

[0346]

The filter operation unit 74 performs the filtering operation with the filter strength determined by the

filter strength determining unit 73 for each line using the image data stored in the image memory 71 and the filter coefficients read from the coefficient memory 76. The filter operation unit 74 outputs the image data having been subjected to the filtering process to the frame memory 25.

[0347]

The deblocking filtering process of the deblocking filtering unit 24 of Fig. 24 is basically the same as the deblocking filtering process described above with reference to Fig. 23, and the description thereof will not be provided.

[0348]

As above, pixels which could not be referred to due to the narrowed image range used for the filtering operation are interpolated using the inclination of the waveform used in the inter-block boundary determination. As a result, the deblocking process when the pixel values have an inclination can be performed even when the image range in the inter-LCU line boundary is narrowed.

[0349]

Moreover, in inter-block boundary determination and strong/weak filtering determination, the pixels which could not be referred to due to the narrowed image range used for the filtering operation are interpolated using the inclination and padding of the waveform used in the inter-block boundary determination. As a result, the deblocking determination process when the pixel values have an inclination can be performed even when the image range in the inter-LCU line boundary is narrowed.

[0350]

In this manner, in the inter-LCU line boundary, even when the image range is narrowed, it is possible to reduce the line memory while maintaining the function of the deblocking process.

[0351]

[15. Eighth embodiment of deblocking filtering unit]

[Configuration example of deblocking filtering unit]

Fig. 25 illustrates the configuration of an eighth embodiment of the deblocking filtering unit.

[0352]

In the example of Fig. 25, the deblocking filtering unit 24 is configured to include an image memory 81, a block boundary determining unit 82, a filter strength determining unit 83, a filter operation unit 84, a coefficient memory 85, and a controller 86.

[0353]

The image memory 81 is a unit that corresponds to the image memory 71 of Fig. 22 and is configured as a line memory. The image memory 81 stores image data supplied from the adder 23. The image memory 81 reads the image data stored therein and supplies the read image data to the block boundary determining unit 82, the filter strength determining unit 83, and the filter operation unit 84.

[0354]

At positions other than the line boundary, the image data is not stored in the image memory 81, but there may be case where the image data supplied from the adder 23 is supplied to the respective units and processed. However, in the example of Fig. 25, for the sake of convenience, it is described that the image data

having passed through the image memory 81 is processed.

[0355]

The block boundary determining unit 82 derives boundaries every eight lines under the control of the controller 86 to calculate parameters used for determination and performs inter-block boundary determination every four lines. That is, the block boundary determining unit 82 derives a boundary between TU and PU using the image data read from the image memory 81 and derives a BS value. Further, the block boundary determining unit 82 obtains an average of the quantization parameters QP of two regions adjacent with a current boundary interposed to calculate an average QP (quantization parameter) and calculates the parameters tc and β based on the calculated average QP.

[0356]

Moreover, the block boundary determining unit 82 determines whether or not to perform filtering every four lines using the image data of two blocks adjacent with the inter-block boundary interposed, supplied from the image memory 81 and the calculated parameters. The block boundary determining unit 82 supplies the calculated parameters to the filter strength determining unit 83 together with the boundary determination results.

[0357]

The filter strength determining unit 83 determines the filter strength every four lines under the control of the controller 86. That is, when the block boundary determining unit 82 determines that filtering is to be performed, the filter strength determining unit 83 determines whether the filtering process will be

performed in a strong filtering mode or a weak filtering mode and outputs the determination results to the filter operation unit 84.

[0358]

The filter operation unit 84 performs the filtering operation with the filter strength determined by the filter strength determining unit 83 every four lines using the image data stored in the image memory 81 and the filter coefficients read from the coefficient memory 85 under the control of the controller 86. In particular, when the filter strength determining unit 83 determines that strong filtering is to be performed, the filter operation unit 84 performs a clipping-based filtering process using Expressions (106) to (111) described above. The filter operation unit 84 outputs the image data having been subjected to the filtering process to the frame memory 25 in the subsequent stage.

[0359]

The coefficient memory 85 stores filter coefficients used in the filtering operation of the deblocking filtering process. The coefficient memory 85 reads the filter coefficients stored therein and supplies the filter coefficients to the filter operation unit 84.

[0360]

The controller 86 receives information (ON/OFF information of deblocking filter, offset values of parameters β and tc, and clipping parameters) from an operation unit (not illustrated) (in the case of the decoding side, the lossless decoder 52). Moreover, the controller 86 is also supplied with prediction mode information and parameters necessary for deblocking

filter such as quantization parameters.

[0361]

For example, the controller 86 supplies the input information to the corresponding unit and controls the respective units of the deblocking filtering unit 24 based on the input ON/OFF information.

[0362]

The controller 86 controls the image memory 81 so as to store image data corresponding to a predetermined number of lines on the lower side in the block and reads the image data stored in the image memory 81. Moreover, when strong filtering is performed, the controller 86 controls the filter operation unit 84 to perform a clipping-based strong filtering process. In this case, the controller 86 supplies parameters (for example, a clipping value pv and a multiplication value of tc of the clipping value) relating to the clipping process to the filter operation unit 84.

[0363]

[Operation of deblocking filtering unit]

Next, the deblocking process of the deblocking filtering unit 24 of Fig. 25 will be described with reference to the flowchart of Fig. 26.

[0364]

For example, ON/OFF information, a β offset value, a Tc offset value, and parameters relating to the clipping process are input to the controller 86 via an operation unit (not illustrated) (in the case of the decoding side, the lossless decoder 52). Moreover, the controller 86 is also supplied with prediction mode information and parameters such necessary for the

deblocking filter such as quantization parameters.

[0365]

In step S201, the controller 86 sets filter offsets (β and Tc offsets) and supplies the set offset information to the filter strength determining unit 83.

[0366]

In step S202, the controller 86 determines whether the deblocking filter can be used based on the ON/OFF information. When it is determined in step S202 that the deblocking filter cannot be used, the deblocking filtering process ends.

[0367]

When it is determined in step S202 that the deblocking filter can be used, the controller 86 sends a notification thereof to the block boundary determining unit 82, the filter strength determining unit 83, and the filter operation unit 84, and the flow proceeds to step S203. In this case, the necessary parameters and the like are also supplied from the controller 86 to the respective units.

[0368]

In step S203, the block boundary determining unit 82 derives a boundary between TU and PU every 8 lines. In step S204, the block boundary determining unit 82 derives a boundary strength (BS) value based on the information on the boundary between TU and PU derived in step S203 and the prediction mode information and the like supplied from the controller 86.

[0369]

In step S205, the block boundary determining unit 82, the filter strength determining unit 83, and the

filter operation unit 84 perform a luminance boundary filtering process. In the luminance boundary filtering process, the luminance boundary filtering is performed on the luminance signals by the process of step S205, which will be described with reference to Fig. 27.

[0370]

In step S206, the block boundary determining unit 82, the filter strength determining unit 83, and the filter operation unit 84 perform a chrominance boundary filtering process. By the process of step S206, chrominance boundary filtering is performed on chrominance signals.

[0371]

In step S207, the controller 86 determines whether all boundaries have been processed. When it is determined in step S207 that all boundaries have not been processed, the flow returns to step S205 and the subsequent processes are repeatedly performed. When it is determined in step S207 that all boundaries have been processed, the flow proceeds to step S208.

[0372]

In step S208, the controller 86 determines whether all CUs have been processed. When it is determined in step S208 that all CUs have not been processed, the flow returns to step S203 and the subsequent processes are repeatedly performed.

[0373]

When it is determined in step S208 that all CUs have been processed, the deblocking filtering process ends.

[0374]

[Example of luminance boundary filtering process]

Next, the luminance boundary filtering process in step S205 of Fig. 26 will be described with reference to the flowchart of Fig. 27.

[0375]

In step S231, the block boundary determining unit 82 determines whether the BS value is larger than 0 every eight lines. When it is determined in step S231 that the BS value is not larger than 0, the luminance boundary filtering process ends. That is, in this case, the filtering process of the filter operation unit 84 is not performed on the luminance boundary, and the pixel values before the filtering are output as they are.

[0376]

When it is determined in step S231 that the BS value is larger than 0, the flow proceeds to step S232. In step S232, the block boundary determining unit 82 obtains an average of quantization parameters QP of two regions adjacent to a current boundary using the pixel values from the image memory 81 every eight lines to calculate an average QP (quantization parameter). In this case, the quantization parameters QP are supplied from the controller 86.

[0377]

In step S233, the block boundary determining unit 82 calculates the parameters tc and β based on the average QP calculated in step S232 every eight lines.

[0378]

In step S234, the block boundary determining unit 82 performs filter on/off determination every four lines. That is, the block boundary determining unit 82 performs

filter on/off determination every four lines using the calculated parameters β and the like. This determination process uses Expression (68) described above, for example.

[0379]

The on/off determination results in step S234 are supplied to the filter strength determining unit 83 together with the calculated parameters as the boundary determination results.

[0380]

Moreover, in step S235, the filter strength determining unit 83 performs filter strength determination every four lines. That is, the filter strength determining unit 83 performs filter strength determination every four lines using the parameters β, tc, and the like calculated by the block boundary determining unit 82. This determination process uses the first terms of Expressions (69) and (61) to (63) described above, for example.

[0381]

The on/off determination results in step S234 and the filter strength determination results in step S235 are supplied to the filter operation unit 84. In this case, the parameter tc is also supplied to the filter operation unit 84.

[0382]

In step S236, the filter operation unit 84 determines whether strong filtering is to be applied to the current four lines based on the determination results from the filter strength determining unit 83. When it is determined in step S236 that strong filtering is to be applied to the current four lines, the flow proceeds to

step S237.

[0383]

In step S237, the filter operation unit 84 performs a clipping-based strong filtering process using Expressions (106) to (111) described above under the control of the controller 86. The filter operation unit 84 outputs the pixel values after the filtering process to the subsequent stage.

[0384]

When it is determined in step S236 that strong filtering is not to be applied, the filter operation unit 84 determines in step S238 whether weak filtering is to be applied to the current four lines based on the determination results from the filter strength determining unit 83. When it is determined in step S238 that weak filtering is to be applied, the flow proceeds to step 239.

[0385]

In step S239, the filter operation unit 84 performs a weak filtering process. The filter operation unit 84 outputs the pixel values after the filtering process to the subsequent stage.

[0386]

When it is determined in step S238 that weak filtering is not to be applied, the flow proceeds to step 240. In step S240, the filter operation unit 84 does not perform the process on the current four lines based on the determination results from the filter strength determining unit 83 but outputs the pixel values after the filtering process to the subsequent stage as they are.

[0387]

In step S240, the filter operation unit 84 determines whether the process for eight lines has been completed, and ends the luminance signal deblocking process when it is determined that the process for eight lines has been completed. When it is determined in step S240 that the process for eight lines has not been completed, the flow returns to step S234 and the subsequent processes are repeatedly performed.

[0388]

As above, since the clipping-based strong filtering process is performed when it is determined that strong filtering is to be applied, it is possible to suppress the influence of a mismatch of the determination process and an incompleteness of the determination process as much as possible. In this way, it is possible to allow the deblocking process to apply filtering appropriately.

[0389]

In the above description, the linear approximation and clipping-based strong filtering process has been described. In the description of linear approximation, although the cases of R2W2 and R2W1 have been described, the linear approximation process for the R3W2 and R1W1 cases can be performed in the same manner.

[0390]

[16. Ninth embodiment]

[Example of R3W2]

Next, the R3W2 case will be described with reference to Fig. 28. In the example of Fig. 28, white circles on solid line indicate pixel values of the respective pixels $p2_i$, $p1_i$, $p0_i$, $q0_i$, $q1_i$, and $q2_i$ near the boundary BB illustrated in Fig. 16 before the deblocking

filter. As illustrated in Fig. 28, a gap between the blocks BKu and BKl is $D = p0_i - q0_i$.

[0391]

Moreover, circles on dot line indicate the respective pixel values after strong filtering of luminance signals in the HEVC scheme. That is, in this case, the difference between the pixel values before and after the strong filtering is 1/8*D for $q2_i$, 2/8*D for $q1_i$, and 3/8*D for $q0_i$. Similarly, the difference between the pixel values before and after the strong filtering is 3/8*D for $p0_i$, 2/8*D for $p1_i$, and 1/8*D for $p2_i$.

[0392]

Here, considering the luminance signal filtering process for the R3W2 case, the pixel $p2_i$ is not subjected to filtering. That is, the pixel $p2_i$ has a pixel value indicated by a dot circle on the solid line, and a difference to the pixel value $p1_i$ indicated by the dot line increases. Thus, the pixel values have an abrupt slope only between the pixel values of $p2_i$ indicated by the dot circle on the solid line and $p1_i$ indicated by the dot line.

[0393]

Thus, in the R3W2 case, in order to make this slope gentle, a difference $\Delta p1_i$ between the pixel value (white circle) of $p0_i$ before filtering and the pixel value (hatched circle) after strong filtering is expressed by Expression (112) below.

[Mathematical formula 34]

$$\begin{aligned}\Delta p1_i &= 1/16*(-3*D+2*4)\\ &= (-3*(p0_i+q0_i)+8)>>4\\ &= (13*p0_i+3*q0_i+8)>>4-p1_i\\ &\fallingdotseq (4*p2_i+5*p1_i+4*p0_i+3*q0_i+8)>>4-p1_i \quad \cdots (112)\end{aligned}$$

[0394]

That is, Expression (112) is derived so that $\Delta p1_i$ takes 3/16*D which is a half of the difference $\Delta p0_i$ between the pixel value (white circle) of $p0_i$ before filtering and the pixel value (hatched circle) after strong filtering.

[0395]

Since the second term on the right side of the last approximation equation of Expression (112) is the pixel value (white circle) before filtering, the first term on the right side of the last approximation equation of Expression (112) represents the pixel value after strong filtering, of $p1_i$ indicated by the hatched circle.

[0396]

That is, the coefficients multiplied to the respective pixel values in the first term on the right side of the last approximation equation of Expression (112) are used as the coefficients of the strong filtering for the R3W2 case. As a result, the results of the strong filtering for the R3W2 case can be made gentle as compared to the slope of the dot line of the conventional technique as indicated by the slope of the bold line in Fig. 28.

[0397]

The determination expression for strong filtering of the R3W2 case is expressed by Expression (113) below.

[Mathematical formula 35]

$$d < (\beta >> 2)$$
$$d = abs(q2_2 - 2*q1_2 + q0_2) + abs(p2_5 - 2*p1_5 + p0_5) + abs(q2_2 - 2*q1_2 + q0_2) + abs(q2_5 - 2*q1_5 + q0_5)$$
$$and\ (|p2_i - p0_i| + |q0_i - q3_i|) < (\beta >> 3)$$
$$and\ |p0_i - q0_i| < ((5*t_c + 1) >> 1) \quad \cdots (113)$$

An expression appearing after the first "and" of Expression (113) for determining the strong filtering of the R3W2 case is different from an expression appearing after the first "and" of Expression (69) for determining the strong filtering of the HEVC scheme. In the case of R3W2, since $p3_i$ could not be referred, padding (that is, $p3_i = p2_i$) is not applied.

[0398]

Thus, the two expressions are different only in that the expression "($|p3_i - p0_i << 1| + |q0_i - q3_i|) < (\beta >> 3)$)" appearing after the first "and" of Expression (69) is changed to the expression "($|p2_i - p0_i << 1| + |q0_i - q3_i|) < (\beta >> 3)$)" appearing after the first "and" of Expression (113).

[0399]

Moreover, in the strong filtering of the R3W2 case, an expression relating to $p1_0$ is different from that of Expression (52) for the strong filtering of the HEVC scheme according to Expression (112) described above as illustrated in Expression (114).

[Mathematical formula 36]

$$p1_0 = Clip_{0-255}((4*p2_i + 5*p1_i + 4*p0_i + 3*q0_i + 8) >> 4); \quad i = 0, 7 \quad \cdots (114)$$

[0400]

In the strong filtering of the R3W2 case, Expression (54) relating to $P2_0$ of the strong filtering

of the HEVC scheme is not necessary and is omitted.

[0401]

In this manner, the strong filtering of the R3W2 case can be performed without deteriorating the accuracy.

[0402]

Moreover, in the present specification, the size of the processing units of the deblocking filter may be the size of macroblocks of the H.264/AVC, the size of coding tree blocks of the HEVC coding units, and the like, for example. Moreover, in the present specification, the term block or macroblock also includes a coding unit (CU), a prediction unit (PU), and a transform unit (TU) in the context of HEVC.

[0403]

A series of processes described in this specification can be performed by hardware, software, or a combination thereof. When the processes are performed by software, a program including the process sequence can be installed in and executed by a memory of a computer assembled into exclusive hardware. Alternatively, the program can be installed in and executed by a general-purpose computer performing various processes.

[0404]

For example, the program may be recorded in advance in a hard disk or a read only memory (ROM) as a recording medium. Alternatively, the program may be recorded temporarily or permanently on a magnetic disk (such as a flexible disk), an optical disk (such as a CD-ROM (compact disc read only memory), a MO (magneto optical) disc, or a DVD (digital versatile disc)), and a removable storage medium (not shown) (such as a semiconductor

memory), for example. Such a removable storage medium may be provided as so-called package software.

[0405]

The program may be wirelessly transferred from a download site to the computer or may be transferred through wires to the computer via a network such as a local area network (LAN) or the Internet in addition to installing the program in the computer from the removable recording medium described above. The computer can receive the program transferred in this manner and install the same in a recording medium such as a built-in hard disk.

[0406]

[19. Tenth embodiment]

[Application to multi-view image encoding and decoding]

The series of processes can be applied to multi-view image encoding and decoding. Fig. 29 illustrates an example of a multi-view image encoding scheme.

[0407]

As illustrated in Fig. 29, a multi-view image includes images of a plurality of views, and an image of a predetermined single view among the plurality of views is designated as a base view image. The respective view images other than the base view image are treated as non-base view images.

[0408]

When the multi-view image as illustrated in Fig. 29 is encoded, in the respective views (same views), the parameters (for example, a clipping value and a multiplication factor relating to tc of the clipping value) relating to the clipping process for strong

filtering can be set. Moreover, in the respective views (different views), the parameters relating the clipping process for strong filtering set to the other views can be shared.

[0409]

In this case, the parameters relating to the clipping process set to a base view are used in at least one non-base view. Alternatively, for example, the parameters relating to the clipping process set to a non-base view (view_id = 0) are used in at least one of the base view and a non-base view (view_id = 1).

[0410]

Further, in the respective views (same views), the parameters relating to the clipping process for strong filtering can be set. Moreover, in the respective views (different views), the parameters relating the clipping process for strong filtering set to the other views can be shared.

[0411]

In this case, the parameters relating to the clipping process set to a base view are used in at least one non-base view. Alternatively, for example, the parameters relating to the clipping process set to a non-base view (view_id = 0) are used in at least one of the base view and a non-base view (view_id = 1).

[0412]

In this manner, it is possible to allow the deblocking process to apply filtering appropriately.

[0413]

[Multi-view image encoding device]

Fig. 30 is a diagram illustrating a multi-view

image encoding device that performs the multi-view image encoding described above. As illustrated in Fig. 30, a multi-view image encoding device 600 includes an encoding unit 601, an encoding unit 602, and a multiplexer 603.

[0414]

The encoding unit 601 encodes a base view image to generate a base view image encoded stream. The encoding unit 602 encodes a non-base view image to generate a non-base view image encoded stream. The multiplexer 603 multiplexes the base view image encoded stream generated by the encoding unit 601 and the non-base view image encoded stream generated by the encoding unit 602 to generate a multi-view image encoded stream.

[0415]

The image encoding device 10 (Fig. 1) can be applied to the encoding units 601 and 602 of the multi-view image encoding device 600. In this case, the multi-view image encoding device 600 sets the parameters relating to the clipping process for strong filtering set by the encoding unit 601 and the parameters relating to the clipping process for strong filtering set by the encoding unit 602 and transfers the parameters.

[0416]

As described above, the parameters relating to the clipping process for strong filtering set by the encoding unit 601 may be transferred so as to be shared by the encoding units 601 and 602. Conversely, the parameters relating to the clipping process for strong filtering set by the encoding unit 602 may be transferred so as to be shared by the encoding units 601 and 602.

[0417]

[Multi-view image decoding device]

Fig. 31 is a diagram illustrating a multi-view image decoding device that performs the multi-view image decoding described above. As illustrated in Fig. 31, a multi-view image decoding device 610 includes a demultiplexer 611, a decoding unit 612, and a decoding unit 613.

[0418]

The demultiplexer 611 demultiplexes the multi-view image encoded stream in which the base view image encoded stream and the non-base view image encoded stream are multiplexed to extract the base view image encoded stream and the non-base view image encoded stream. The decoding unit 612 decodes the base view image encoded stream extracted by the demultiplexer 611 to obtain a base view image. The decoding unit 613 decodes the non-base view image encoded stream extracted by the demultiplexer 611 to obtain a non-base view image.

[0419]

The image decoding device 50 (Fig. 6) can be applied to the decoding units 612 and 613 of the multi-view image decoding device 610. In this case, the multi-view image decoding device 610 performs the process using the parameters relating to the clipping process for strong filtering, that are set by the encoding unit 601 and decoded by the decoding unit 612 and the parameters relating to the clipping process for strong filtering, that are set by the encoding unit 602 and decoded by the decoding unit 613.

[0420]

As described above, there is a case where the

parameters relating to the clipping process for strong filtering set by the encoding unit 601 (or the encoding 602) are transferred so as to be shared by the encoding units 601 and 602. In this case, the multi-view image decoding device 610 performs the process using the parameters relating to the clipping process for strong filtering, that are set by the encoding unit 601 (or the encoding 602) and decoded by the decoding unit 612 (or the decoding unit 613).

[0421]

[18. Eleventh embodiment]

[Application to layer image encoding and decoding]

The above series of processes can be applied to layer image encoding and decoding. Fig. 32 illustrates an example of a multi-view image encoding scheme.

[0422]

As illustrated in Fig. 32, a layer image includes images of a plurality of layers (resolutions), and an image of a predetermined single layer among the plurality of resolutions is designated as a base layer image. The respective layer images other than the base layer image are treated as non-base layer images.

[0423]

When the layer image encoding (space scalability) as illustrated in Fig. 32 is performed, in the respective layers (same layers), the parameters relating to the clipping process for strong filtering can be set. Moreover, in the respective layers (different layers), the parameters relating to the clipping process for strong filtering set to the other layers can be shared.

[0424]

In this case, the parameters relating to the clipping process for strong filtering set to a base layer are used in at least one non-base layer. Alternatively, for example, the parameters relating to the clipping process for strong filtering set to a non-base layer (layer_id = 0) are used in at least one of the base layer and a non-base layer (layer_id = 1).

[0425]

Further, in the respective layers (same layers), the parameters relating to the clipping process for strong filtering can be set. Moreover, in the respective layers (different layers), the parameters relating to the clipping process for strong filtering set to the other views can be shared.

[0426]

In this case, the parameters relating to the clipping process for strong filtering set to a base layer are used in at least one non-base layer. Alternatively, for example, the parameters relating to the clipping process for strong filtering set to a non-base layer (layer_id = 0) are used in at least one of the base layer and a non-base layer (layer_id = 1).

[0427]

In this manner, it is possible to allow the deblocking process to apply filtering appropriately.

[0428]

[Layer image encoding device]

Fig. 33 is a diagram illustrating a layer image encoding device that performs the layer image encoding described above. As illustrated in Fig. 33, a layer image encoding device 620 includes an encoding unit 621,

an encoding unit 622, and a multiplexer 623.

[0429]

The encoding unit 621 encodes a base layer image to generate a base layer image encoded stream. The encoding unit 622 encodes a non-base layer image to generate a non-base layer image encoded stream. The multiplexer 623 multiplexes the base layer image encoded stream generated by the encoding unit 621 and the non-base layer image encoded stream generated by the encoding unit 622 to generate a layer image encoded stream.

[0430]

The image encoding device 10 (Fig. 1) can be applied to the encoding units 621 and 622 of the layer image encoding device 620. In this case, the layer image encoding device 620 sets the parameters relating to the clipping process for strong filtering set by the encoding unit 621 and the parameters relating to the clipping process for strong filtering set by the encoding unit 602 and transfers the parameters.

[0431]

As described above, the parameters relating to the clipping process for strong filtering set by the encoding unit 621 may be transferred so as to be shared by the encoding units 621 and 622. Conversely, the parameters relating to the clipping process for strong filtering set by the encoding unit 622 may be transferred so as to be shared by the encoding units 621 and 622.

[0432]

[Layer image decoding device]

Fig. 34 is a diagram illustrating a layer image decoding device that performs the layer image decoding

described above. As illustrated in Fig. 34, a layer image decoding device 630 includes a demultiplexer 631, a decoding unit 632, and a decoding unit 633.

[0433]

The demultiplexer 631 demultiplexes the layer image encoded stream in which the base layer image encoded stream and the non-base layer image encoded stream are multiplexed to extract the base layer image encoded stream and the non-base layer image encoded stream. The decoding unit 632 decodes the base layer image encoded stream extracted by the demultiplexer 631 to obtain a base layer image. The decoding unit 633 decodes the non-base layer image encoded stream extracted by the demultiplexer 631 to obtain a non-base layer image.

[0434]

The image decoding device 50 (Fig. 6) can be applied to the decoding units 632 and 633 of the multi-view image decoding device 630. In this case, the multi-view image decoding device 630 performs the process using the parameter relating to the clipping process for strong filtering, that are set by the encoding unit 621 and decoded by the decoding unit 632 and the parameters relating to the clipping process for strong filtering, that are set by the encoding unit 622 and decoded by the decoding unit 633.

[0435]

As described above, there is a case where the parameters relating to the clipping process for strong filtering set by the encoding unit 621 (or the encoding 622) are transferred so as to be shared by the encoding units 621 and 622. In this case, the multi-view image

decoding device 630 performs the process using the parameters relating to the clipping process for strong filtering, that are set by the encoding unit 621 (or the encoding 622) and decoded by the decoding unit 632 (or the decoding unit 633).

[0436]

[19. Application example]

The image encoding device 10 and the image decoding device 50 according to the above-described embodiments can be applied to various electronic apparatuses such as a transmitter or a receiver that distributes signals on cable broadcasting (such as satellite broadcasting or a cable TV) or on the Internet and distributes signals to a terminal by cellular communication, a recording device that records images on a medium such as an optical disc, a magnetic disk, or a flash memory, or a reproducing device that reproduces images from these storage media. Four application examples will be described below.

[0437]

[First application example]

Fig. 35 illustrates an example of a schematic configuration of a television apparatus to which the above-described embodiment is applied. A television apparatus 900 includes an antenna 901, a tuner 902, a demultiplexer 903, a decoder 904, a video signal processor 905, a display unit 906, an audio signal processor 907, a speaker 908, and an external interface 909. The television apparatus 900 further includes a controller 910, a user interface 911, and the like.

[0438]

The tuner 902 extracts a signal of a desired

channel from a broadcast signal received through the antenna 901 and demodulates the extracted signal. Then, the tuner 902 outputs an encoded bit stream obtained by demodulation to the demultiplexer 903. That is, the tuner 902 serves as transmitting means in the television apparatus 900, which receives the encoded stream in which the image is encoded.

[0439]

The demultiplexer 903 separates a video stream and an audio stream of a program to be watched from the encoded bit stream and outputs each separated stream to the decoder 904. Moreover, the demultiplexer 903 extracts auxiliary data such as EPG (Electronic Program Guide) from the encoded bit stream and supplies the extracted data to the controller 910. The demultiplexer 903 may descramble when the encoded bit stream is scrambled.

[0440]

The decoder 904 decodes the video stream and the audio stream input from the demultiplexer 903. Then, the decoder 904 outputs video data generated by a decoding process to the video signal processor 905. Moreover, the decoder 904 outputs audio data generated by the decoding process to the audio signal processor 907.

[0441]

The video signal processor 905 reproduces the video data input from the decoder 904 and allows the display unit 906 to display video. The video signal processor 905 may also allow the display unit 906 to display an application screen supplied through the network. The video signal processor 905 may also perform an additional

process such as noise removal, for example, to the video data according to setting. Further, the video signal processor 905 may generate a GUI (Graphical User Interface) image such as a menu, a button, and a cursor, for example, and superimpose the generated image on an output image.

[0442]

The display unit 906 is driven by a drive signal supplied from the video signal processor 905 to display the video or image on a video screen of a display device (for example, a liquid crystal display, a plasma display, an OLED, and the like).

[0443]

The audio signal processor 907 performs a reproducing process such as D/A conversion and amplification to the audio data input from the decoder 904 and allows the speaker 908 to output the audio. The audio signal processor 907 may also perform an additional process such as the noise removal to the audio data. The external interface 909 is the interface for connecting the television apparatus 900 and an external device or the network. For example, the video stream or the audio stream received through the external interface 909 may be decoded by the decoder 904. That is, the external interface 909 also serves as the transmitting means in the television apparatus 900, which receives the encoded stream in which the image is encoded.

[0444]

The controller 910 includes a processor such as a central processing unit (CPU) and a memory such as a random access memory (RAM) and a read only memory (ROM).

The memory stores the program executed by the CPU, program data, the EPG data, data obtained through the network and the like. The program stored in the memory is read by the CPU at startup of the television apparatus 900 to be executed, for example. The CPU controls operation of the television apparatus 900 according to an operation signal input from the user interface 911, for example, by executing the program.

[0445]

The user interface 911 is connected to the controller 910. The user interface 911 includes a button and a switch for the user to operate the television apparatus 900, a receiver of a remote control signal and the like, for example. The user interface 911 detects operation by the user through the components to generate the operation signal and outputs the generated operation signal to the controller 910.

[0446]

The bus 912 connects the tuner 902, the demultiplexer 903, the decoder 904, the video signal processor 905, the audio signal processor 907, the external interface 909, and the controller 910 to one another.

[0447]

In the television apparatus 900 configured in this manner, the decoder 904 has the functions of the image decoding device 50 according to the above-described embodiment. Therefore, when images are decoded in the television apparatus 900, the deblocking filtering process can apply filtering appropriately.

[0448]

[Second application example]

Fig. 36 illustrates an example of a schematic configuration of a mobile phone to which the above-described embodiment is applied. A mobile phone 920 includes an antenna 921, a communication unit 922, an audio codec 923, a speaker 924, a microphone 925, a camera unit 926, an image processor 927, a demultiplexer 928, a recording/reproducing unit 929, a display unit 930, a controller 931, an operation unit 932, and a bus 933.

[0449]

The antenna 921 is connected to the communication unit 922. The speaker 924 and the microphone 925 are connected to the audio codec 923. The operation unit 932 is connected to the controller 931. The bus 933 connects the communication unit 922, the audio codec 923, the camera unit 926, the image processor 927, the demultiplexer 928, the recording/reproducing unit 929, the display unit 930, and the controller 931 to one another.

[0450]

The mobile phone 920 performs operation such as transmission/reception of an audio signal, transmission/reception of an e-mail or image data, image taking, and recording of data in various operation modes including an audio communication mode, a data communication mode, an imaging mode, and a television-phone mode.

[0451]

In the audio communication mode, an analog audio signal generated by the microphone 925 is supplied to the audio codec 923. The audio codec 923 converts the analog

audio signal to the audio data and A/D converts the converted audio data to compress. Then, the audio codec 923 outputs the compressed audio data to the communication unit 922. The communication unit 922 encodes and modulates the audio data to generate a transmission signal. Then, the communication unit 922 transmits the generated transmission signal to a base station (not illustrated) through the antenna 921. Moreover, the communication unit 922 amplifies a wireless signal received through the antenna 921 and applies frequency conversion to the same to obtain a reception signal. Then, the communication unit 922 generates the audio data by demodulating and decoding the reception signal and outputs the generated audio data to the audio codec 923. The audio codec 923 expands the audio data and D/A converts the same to generate the analog audio signal. Then, the audio codec 923 supplies the generated audio signal to the speaker 924 to allow the same to output the audio.

[0452]

In the data communication mode, for example, the controller 931 generates character data composing the e-mail according to the operation by the user through the operation unit 932. Moreover, the controller 931 allows the display unit 930 to display characters. The controller 931 generates e-mail data according to a transmission instruction from the user through the operation unit 932 to output the generated e-mail data to the communication unit 922. The communication unit 922 encodes and modulates the e-mail data to generate the transmission signal. Then, the communication unit 922

transmits the generated transmission signal to the base station (not illustrated) through the antenna 921. Moreover, the communication unit 922 amplifies the wireless signal received through the antenna 921 and applies the frequency conversion to the same to obtain the reception signal. Then, the communication unit 922 demodulates and decodes the reception signal to restore the e-mail data and outputs the restored e-mail data to the controller 931. The controller 931 allows the display unit 930 to display contents of the e-mail data and allows the storage medium of the recording/reproducing unit 929 to store the e-mail data.

[0453]

The recording/reproducing unit 929 includes an arbitrary readable/writable storage medium. For example, the storage medium may be a built-in storage medium such as the RAM and the flash memory and may be an externally-mounted storage medium such as the hard disc, the magnetic disc, the magneto-optical disc, the optical disc, a USB memory, and a memory card.

[0454]

In the imaging mode, for example, the camera unit 926 takes an image of an object to generate the image data and outputs the generated image data to the image processor 927. The image processor 927 encodes the image data input from the camera unit 926 and stores the encoded stream in the storage medium of the recording/reproducing unit 929.

[0455]

Moreover, in the television-phone mode, for example, the demultiplexer 928 multiplexes the video stream

encoded by the image processor 927 and the audio stream input from the audio codec 923 and outputs the multiplexed stream to the communication unit 922. The communication unit 922 encodes and modulates the stream to generate the transmission signal. Then, the communication unit 922 transmits the generated transmission signal to the base station (not illustrated) through the antenna 921. Moreover, the communication unit 922 amplifies the wireless signal received through the antenna 921 and applies the frequency conversion to the same to obtain the reception signal. The transmission signal and the reception signal may include the encoded bit stream. Then, the communication unit 922 restores the stream by demodulating and decoding the reception signal and outputs the restored stream to the demultiplexer 928. The demultiplexer 928 separates the video stream and the audio stream from the input stream and outputs the video stream and the audio stream to the image processor 927 and the audio codec 923, respectively. The image processor 927 decodes the video stream to generate the video data. The video data is supplied to the display unit 930 and a series of images is displayed by the display unit 930. The audio codec 923 expands the audio stream and D/A converts the same to generate the analog audio signal. Then, the audio codec 923 supplies the generated audio signal to the speaker 924 to output the audio.

[0456]

In the mobile phone 920 configured in this manner, the image processor 927 has the functions of the image encoding device 10 and the image decoding device 50

according to the above-described embodiment. Therefore, when images are encoded and decoded in the mobile phone 920, the deblocking filtering process can apply filtering appropriately.

[0457]

[Third application example]

Fig. 37 illustrates an example of a schematic configuration of the recording/reproducing device to which the above-described embodiment is applied. The recording/reproducing device 940 encodes the audio data and the video data of a received broadcast program to record on the recording medium, for example. Moreover, the recording/reproducing device 940 may encode the audio data and the video data obtained from another apparatus to record on the recording medium, for example. Moreover, the recording/reproducing device 940 reproduces the data recorded on the recording medium by a monitor and the speaker according to the instruction of the user. In this case, the recording/reproducing device 940 decodes the audio data and the video data.

[0458]

The recording/reproducing device 940 includes a tuner 941, an external interface 942, an encoder 943, a HDD (Hard Disk Drive) 944, a disk drive 945, a selector 946, a decoder 947, an OSD (On-Screen Display) 948, a controller 949, and a user interface 950.

[0459]

The tuner 941 extracts a signal of a desired channel from the broadcast signal received through an antenna (not illustrated) and demodulates the extracted signal. Then, the tuner 941 outputs the encoded bit

stream obtained by the demodulation to the selector 946. That is, the tuner 941 serves as the transmitting means in the recording/reproducing device 940.

[0460]

The external interface 942 is the interface for connecting the recording/reproducing device 940 and the external device or the network. The external interface 942 may be an IEEE1394 interface, a network interface, a USB interface, a flash memory interface and the like, for example. For example, the video data and the audio data received through the external interface 942 are input to the encoder 943. That is, the external interface 942 serves as the transmitting means in the recording/reproducing device 940.

[0461]

The encoder 943 encodes the video data and the audio data when the video data and the audio data input from the external interface 942 are not encoded. Then, the encoder 943 outputs the encoded bit stream to the selector 946.

[0462]

The HDD 944 records the encoded bit stream in which content data such as the video and the audio are compressed, various programs and other data on an internal hard disc. The HDD 944 reads the data from the hard disc when reproducing the video and the audio.

[0463]

The disk drive 945 records and reads the data on and from the mounted recording medium. The recording medium mounted on the disk drive 945 may be the DVD disc (DVD-Video, DVD-RAM, DVD-R, DVD-RW, DVD+R, DVD+RW and the

like), a Blu-ray (registered trademark) disc and the like, for example. The selector 946 selects the encoded bit stream input from the tuner 941 or the encoder 943 and outputs the selected encoded bit stream to the HDD 944 or the disk drive 945 when recording the video and the audio. Moreover, the selector 946 outputs the encoded bit stream input from the HDD 944 or the disk drive 945 to the decoder 947 when reproducing the video and the audio.

[0464]

The decoder 947 decodes the encoded bit stream to generate the video data and the audio data. Then, the decoder 947 outputs the generated video data to the OSD 948. Moreover, the decoder 904 outputs the generated audio data to an external speaker. The OSD 948 reproduces the video data input from the decoder 947 to display the video. The OSD 948 may also superimpose the GUI image such as the menu, the button, and the cursor, for example, on the displayed video.

[0465]

The controller 949 includes the processor such as the CPU and the memory such as the RAM and ROM. The memory stores the program executed by the CPU, the program data and the like. The program stored in the memory is read by the CPU to be executed on activation of the recording/reproducing device 940, for example. The CPU controls operation of the recording/reproducing device 940 according to an operation signal input from the user interface 950, for example, by executing the program.

[0466]

The user interface 950 is connected to the

controller 949. The user interface 950 includes a button and a switch for the user to operate the recording/reproducing device 940 and a receiver of a remote control signal, for example. The user interface 950 detects operation by the user through the components to generate the operation signal and outputs the generated operation signal to the controller 949.

[0467]

In the recording/reproducing device 940 configured in this manner, the encoder 943 has the functions of the image encoding device 10 according to the above-described embodiment. Moreover, the decoder 947 has the functions of the image decoding device 50 according to the above-described embodiment. Therefore, when images are encoded and decoded in the recording/reproducing device 940, the deblocking filtering process can apply filtering appropriately.

[0468]

[Fourth application example]

Fig. 38 illustrates an example of a schematic configuration of an imaging device to which the above-described embodiment is applied. An imaging device 960 images an object to generate the image, encodes the image data, and records the same on a recording medium.

[0469]

The imaging device 960 includes an optical block 961, an imaging unit 962, a signal processor 963, an image processor 964, a display unit 965, an external interface 966, a memory 967, a media drive 968, an OSD 969, a controller 970, a user interface 971, and a bus 972.

[0470]

The optical block 961 includes a focus lens, a diaphragm mechanism, and the like. The optical block 961 forms an optical image of the object on an imaging surface of the imaging unit 962. The imaging unit 962 includes an image sensor such as a CCD and a CMOS and converts the optical image formed on the imaging surface to an image signal as an electric signal by photoelectric conversion. Then, the imaging unit 962 outputs the image signal to the signal processor 963.

[0471]

The signal processor 963 performs various camera signal processes such as knee correction, gamma correction, or color correction to the image signal input from the imaging unit 962. The signal processor 963 outputs the image data after the camera signal process to the image processor 964.

[0472]

The image processor 964 encodes the image data input from the signal processor 963 to generate the encoded data. Then, the image processor 964 outputs the generated encoded data to the external interface 966 or the media drive 968. Moreover, the image processor 964 decodes the encoded data input from the external interface 966 or the media drive 968 to generate the image data. Then, the image processor 964 outputs the generated image data to the display unit 965. The image processor 964 may also output the image data input from the signal processor 963 to the display unit 965 to display the image. The image processor 964 may also superimpose data for display obtained from the OSD 969 on

the image output to the display unit 965.

[0473]

The OSD 969 generates the GUI image such as the menu, the button, and the cursor, for example, and outputs the generated image to the image processor 964.

[0474]

The external interface 966 is configured as an USB input/output terminal, for example. The external interface 966 connects the imaging device 960 and a printer when printing the image, for example. Moreover, a drive is connected to the external interface 966 as necessary. The removable medium such as the magnetic disc and the optical disc is mounted on the drive, for example, and the program read from the removable medium may be installed on the imaging device 960. Further, the external interface 966 may be configured as a network interface connected to the network such as a LAN and the Internet. That is, the external interface 966 serves as the transmitting means in the imaging device 960.

[0475]

The recording medium mounted on the media drive 968 may be an arbitrary readable/writable removable medium such as the magnetic disc, the magneto-optical disc, the optical disc, and the semiconductor memory, for example. Moreover, the recording medium may be fixedly mounted on the media drive 968 to form a non-portable storage unit such as a built-in hard disk drive or SSD (Solid State Drive), for example.

[0476]

The controller 970 includes the processor such as the CPU and the memory such as the RAM and the ROM. The

memory stores the program executed by the CPU and the program data. The program stored in the memory is read by the CPU at startup of the imaging device 960 to be executed, for example. The CPU controls operation of the imaging device 960 according to the operation signal input from the user interface 971, for example, by executing the program.

[0477]

The user interface 971 is connected to the controller 970. The user interface 971 includes a button, a switch and the like for the user to operate the imaging device 960, for example. The user interface 971 detects the operation by the user through the components to generate the operation signal and outputs the generated operation signal to the controller 970.

[0478]

The bus 972 connects the image processor 964, the external interface 966, the memory 967, the media drive 968, the OSD 969, and the controller 970 to one another.

[0479]

In the imaging device 960 configured in this manner, the image processor 964 has the functions of the image encoding device 10 and the image decoding device 50 according to the above-described embodiment. Therefore, when images are encoded and decoded in the imaging device 960, the deblocking filtering process can apply filtering appropriately.

[0480]

In the present specification, an example in which various types of information such as the parameters relating to the clipping process for strong filtering are

multiplexed into encoded streams and are transmitted from the encoding side to the decoding side has been described. However, a method of transmitting these items of information is not limited to this example. For example, these items of information may be transmitted or recorded as separate data associated with the encoded bit stream rather than being multiplexed into the encoded bit stream. Here, the term "associate" means that the image (or part of the image such as a slice and a block) included in the bit stream and information corresponding to the image can be linked with each other at the time of decoding. That is, the information may be transmitted on a transmission line other than that of the image (or bit stream). Moreover, the information may be recorded on another recording medium (or another recording area of the same recording medium) other than that of the image (or bit stream). Further, the information and the image (or bit stream) may be associated with each other in optional units such as a plurality of frames, one frame, or a part of the frame, for example.

[0481]

Furthermore, the present technique is not to be interpreted as being limited to the above-described embodiments. The embodiments disclose the present technique in an exemplary and illustrative form, and it is obvious that modifications and substitutions may occur to those skilled in the art without departing from the spirit of the present technique. In other words, the spirit of the present technique should be determined in consideration of the claims.

[0482]

Note that the image processing device of this technique may include the following constitutions.

(1) An image processing device including:

a decoding unit that decodes an encoded stream encoded in units each having a layer structure to generate an image;

a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the image generated by the decoding unit and an adjacent block adjacent to the block; and

a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the image generated by the decoding unit.

(2) The image processing device according to (1), wherein

the controller applies the clipping process to values of parts that change when the deblocking filter is applied.

(3) The image processing device according to (2), wherein

the controller applies the clipping process according to the following expressions:

[Mathematical formula 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

(4) The image processing device according to (3), wherein

the controller sets a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

(5) The image processing device according to (4), wherein

the controller sets a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

(6) The image processing device according to any of (1) to (5), further including:

a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, wherein

the filtering unit applies the deblocking filter to the block boundary according to the strength determined by the filter strength determining unit, and

the controller controls the filtering unit so that, when the filter strength determining unit determines that

strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the decoding unit.

(7) The image processing device according to (6), wherein

the filter strength determining unit determines the strength of the deblocking filter using a plurality of lines as processing units.

(8) The image processing device according to (7), wherein

the filter strength determining unit determines the strength of the deblocking filter using four lines as the processing units.

(9) The image processing device according to (7), further including:

a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, wherein

the filter strength determining unit determines the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

(10) An image processing method for allowing an image processing device to execute:

decoding an encoded stream encoded in units each having a layer structure to generate an image;

applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block

of the generated image and an adjacent block adjacent to the block; and

controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the generated image.

(11) An image processing device including:

a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block;

a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and

an encoding unit that encodes the image in units each having a layer structure using an image to which the deblocking filter is applied.

(12) The image processing device according to (11), wherein

the controller applies the clipping process to values of parts that change when the deblocking filter is applied.

(13) The image processing device according to (12), wherein

the controller applies the clipping process according to the following expressions:

[Mathematical formula 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

(14) The image processing device according to (13), wherein

the controller sets a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

(15) The image processing device according to (14), wherein

the controller sets a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

(16) The image processing device according to any of (11) to (15), further including:

a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, wherein

the filtering unit applies the deblocking filter to the block boundary according to the strength determined

by the filter strength determining unit, and

the controller controls the filtering unit so that, when the filter strength determining unit determines that strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the decoding unit.

(17) The image processing device according to (16), wherein

the filter strength determining unit determines the strength of the deblocking filter using a plurality of lines as processing units.

(18) The image processing device according to (17), wherein

the filter strength determining unit determines the strength of the deblocking filter using four lines as the processing units.

(19) The image processing device according to (17), further including:

a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, wherein

the filter strength determining unit determines the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

(20) An image processing method for allowing an image processing device to execute:

applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied

to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block;

controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and

encoding the image in units each having a layer structure using an image to which the deblocking filter is applied.

REFERENCE SIGNS LIST

[0483]

10: Image encoding device
11: A/D converter
12, 57: Frame reordering buffer
13: Subtractor
14: Orthogonal transformer
15: Quantizer
16: Lossless encoder
17: Accumulation buffer
18: Rate controller
21, 53: Inverse quantizer
22, 54: Inverse orthogonal transformer
23, 55: Adder
24, 56: Deblocking filtering unit
25, 61: Frame memory
26, 62, 65: Selector
31, 63: Intra predictor
32: Motion estimator/compensator

33: Predicted image/optimal mode selecting unit
50: Image decoding device
51: Accumulation buffer
52: Lossless decoder
61: Frame memory
64: Motion compensation unit
71: Image memory
72, 72-1: Block boundary determining unit
72-2: Line boundary/block boundary determining unit
73, 73-1: Filter strength determining unit
73-2: Line boundary filter strength determining unit
74, 74-1: Filter operation unit
74-2: Line boundary filter operation unit
75: Selector
76: Coefficient memory
77: Controller
77A, 77B: Intra-LCU line determining unit
241: Line memory
242: Line boundary detecting unit
243: Filter strength determining unit
244: Coefficient memory
245: Filter operation unit
246: Filter controller
2451: Data storage unit
2452: Data selecting unit
2453, 2455, 2456: Arithmetic processing unit
2457: Data selecting unit
2461: Line boundary determining unit

CLAIMS

1. An image processing device comprising:

a decoding unit that decodes an encoded stream encoded in units each having a layer structure to generate an image;

a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the image generated by the decoding unit and an adjacent block adjacent to the block; and

a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the image generated by the decoding unit.

2. The image processing device according to claim 1, wherein

the controller applies the clipping process to values of parts that change when the deblocking filter is applied.

3. The image processing device according to claim 2, wherein

the controller applies the clipping process according to the following expressions:

[Mathematical formula 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

4. The image processing device according to claim 3, wherein

the controller sets a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

5. The image processing device according to claim 4, wherein

the controller sets a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

6. The image processing device according to claim 1, further comprising:

a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, wherein

the filtering unit applies the deblocking filter to the block boundary according to the strength determined

by the filter strength determining unit, and

the controller controls the filtering unit so that, when the filter strength determining unit determines that strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the decoding unit.

7. The image processing device according to claim 6, wherein

the filter strength determining unit determines the strength of the deblocking filter using a plurality of lines as processing units.

8. The image processing device according to claim 7, wherein

the filter strength determining unit determines the strength of the deblocking filter using four lines as the processing units.

9. The image processing device according to claim 7, further comprising:

a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, wherein

the filter strength determining unit determines the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

10. An image processing method for allowing an image processing device to execute:

decoding an encoded stream encoded in units each having a layer structure to generate an image;

applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of the generated image and an adjacent block adjacent to the block; and

controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the generated image.

11. An image processing device comprising:

a filtering unit that applies a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block;

a controller that controls the filtering unit so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and

an encoding unit that encodes the image in units each having a layer structure using an image to which the deblocking filter is applied.

12. The image processing device according to claim 11,

wherein

the controller applies the clipping process to values of parts that change when the deblocking filter is applied.

13. The image processing device according to claim 12, wherein

the controller applies the clipping process according to the following expressions:

[Mathematical formula 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

where i = 0, 7 and pv is a clipping value.

14. The image processing device according to claim 13, wherein

the controller sets a value that is integer multiples of parameters of the deblocking filter as the clipping value used when the clipping process is performed.

15. The image processing device according to claim 14, wherein

the controller sets a value that is twice the parameters of the deblocking filter as the clipping value used when the clipping process is performed.

16. The image processing device according to claim 11, further comprising:

a filter strength determining unit that determines the strength of the deblocking filter applied to the block boundary, wherein

the filtering unit applies the deblocking filter to the block boundary according to the strength determined by the filter strength determining unit, and

the controller controls the filtering unit so that, when the filter strength determining unit determines that strong filtering is to be applied, a clipping process is applied to the deblocking filter with respect to the luminance components of the image generated by the decoding unit.

17. The image processing device according to claim 16, wherein

the filter strength determining unit determines the strength of the deblocking filter using a plurality of lines as processing units.

18. The image processing device according to claim 17, wherein

the filter strength determining unit determines the strength of the deblocking filter using four lines as the

processing units.

19. The image processing device according to claim 17, further comprising:

a filtering necessity determining unit that determines whether the deblocking filter is to be applied to the block boundary using a plurality of lines as processing units, wherein

the filter strength determining unit determines the strength of the deblocking filter when the filtering necessity determining unit determines that the deblocking filter is to be applied.

20. An image processing method for allowing an image processing device to execute:

applying a deblocking filter to a block boundary according to a strength of the deblocking filter applied to the block boundary which is a boundary between a block of an image locally decoded when an image is encoded and an adjacent block adjacent to the block;

controlling so that, when strong filtering is applied as the strength of the deblocking filter, a clipping process is applied to the deblocking filter with respect to luminance components of the locally decoded image; and

encoding the image in units each having a layer structure using an image to which the deblocking filter is applied.

ABSTRACT

The present technology relates to an image processing device and an image processing method which allow a deblocking filtering process to apply filtering appropriately. A pixel ($p0_i$) of which the value is 255 (solid line) before a deblocking process changes greatly to 159 (dot line) after a conventional deblocking process. Therefore, a clipping process having a clipping value of 10 is performed in strong filtering, whereby the pixel ($p0_i$) of which the value is 255 (solid line) before the deblocking process becomes 245 (bold line). Thus, a change in the pixel value occurring in the conventional technique can be suppressed as much as possible. This disclosure can be applied to an image processing device, for example.

FIG. 1

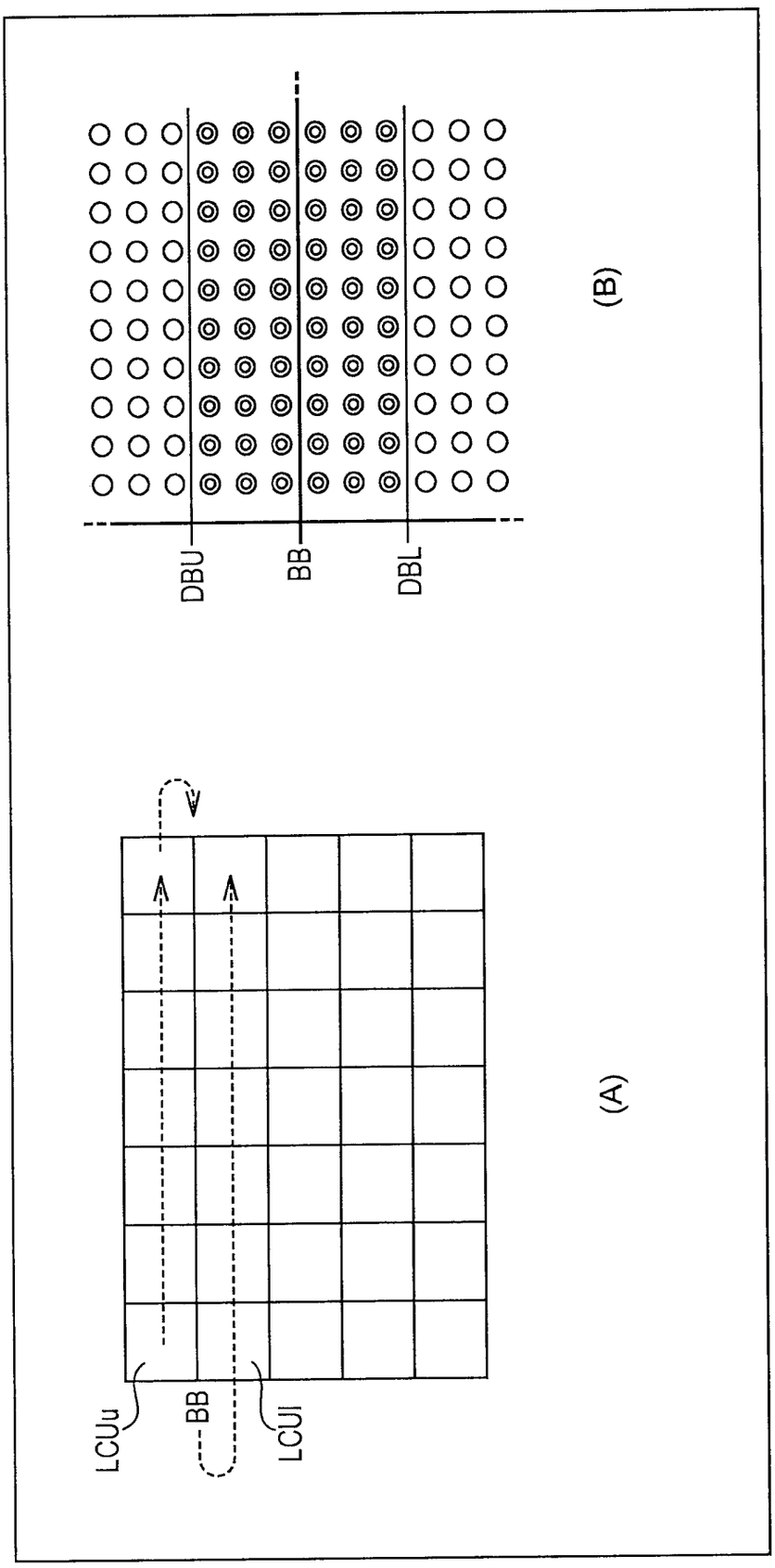


FIG. 2

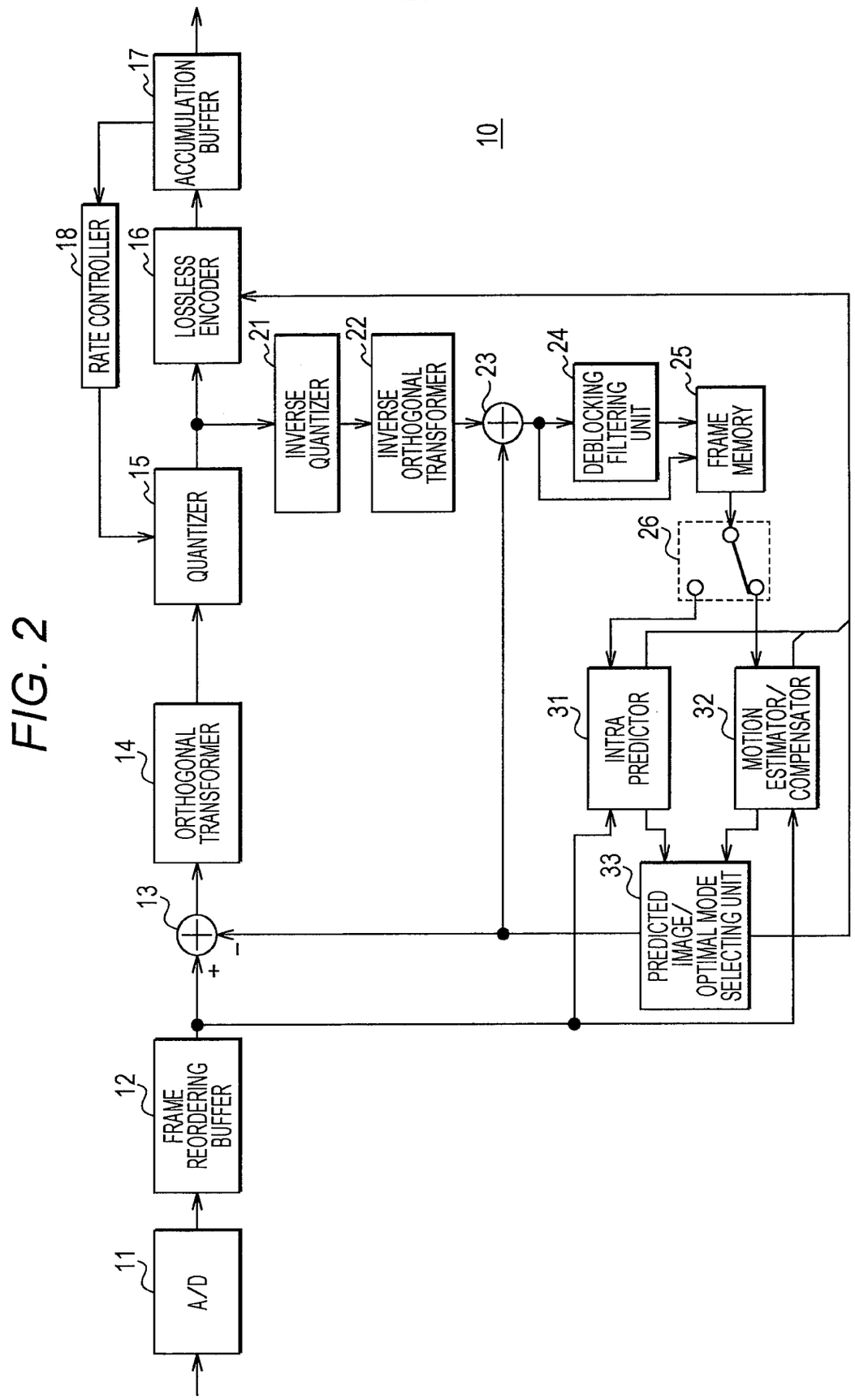

10
11
A/D
12
FRAME REORDERING BUFFER
13
+
-
14
ORTHOGONAL TRANSFORMER
15
QUANTIZER
16
LOSSLESS ENCODER
17
ACCUMULATION BUFFER
18
RATE CONTROLLER
21
INVERSE QUANTIZER
22
INVERSE ORTHOGONAL TRANSFORMER
23
24
DEBLOCKING FILTERING UNIT
25
FRAME MEMORY
26
31
INTRA PREDICTOR
32
MOTION ESTIMATOR/ COMPENSATOR
33
PREDICTED IMAGE/ OPTIMAL MODE SELECTING UNIT

FIG. 3

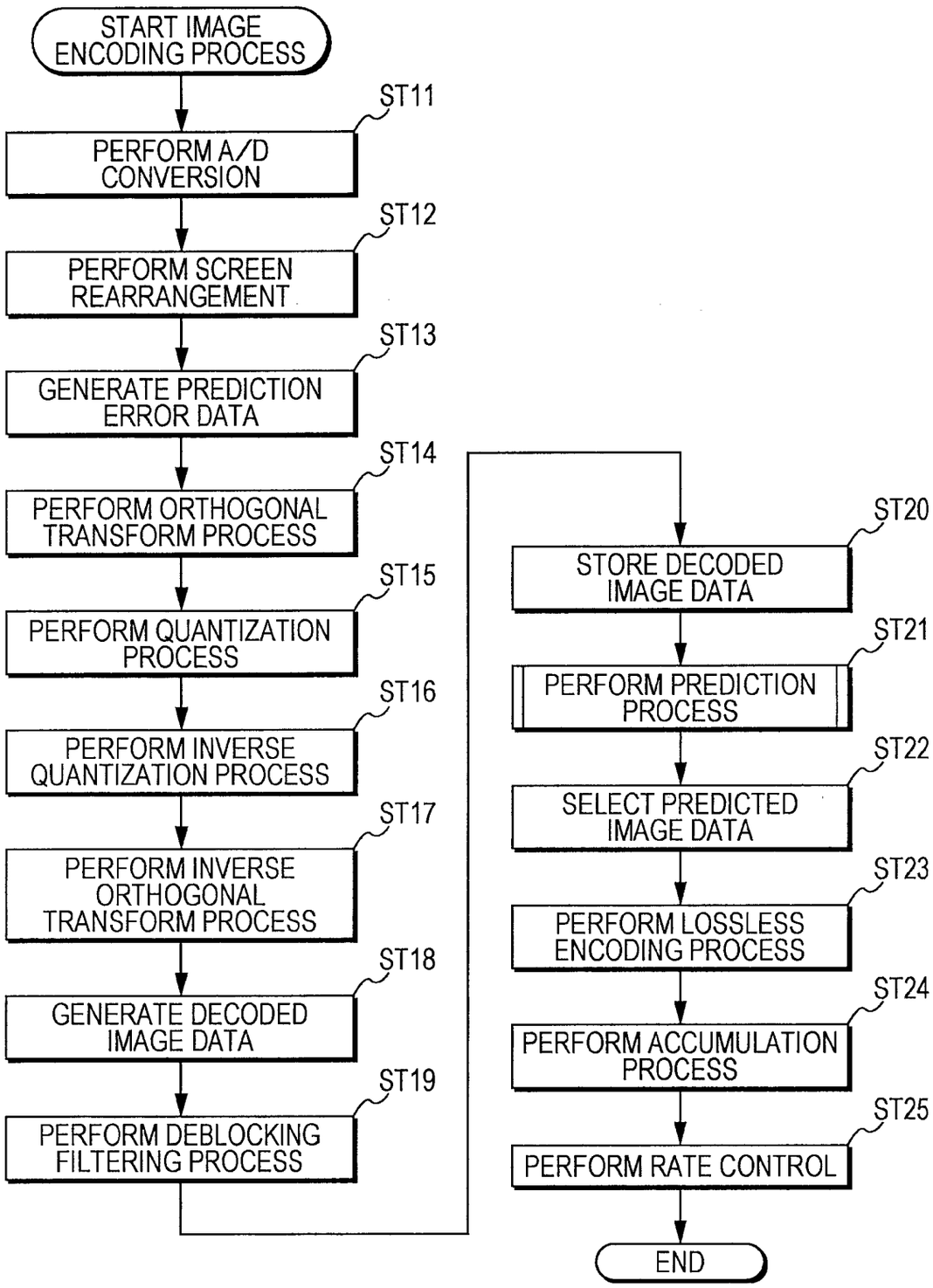

START IMAGE ENCODING PROCESS
ST11
PERFORM A/D CONVERSION
ST12
PERFORM SCREEN REARRANGEMENT
ST13
GENERATE PREDICTION ERROR DATA
ST14
PERFORM ORTHOGONAL TRANSFORM PROCESS
ST15
PERFORM QUANTIZATION PROCESS
ST16
PERFORM INVERSE QUANTIZATION PROCESS
ST17
PERFORM INVERSE ORTHOGONAL TRANSFORM PROCESS
ST18
GENERATE DECODED IMAGE DATA
ST19
PERFORM DEBLOCKING FILTERING PROCESS
ST20
STORE DECODED IMAGE DATA
ST21
PERFORM PREDICTION PROCESS
ST22
SELECT PREDICTED IMAGE DATA
ST23
PERFORM LOSSLESS ENCODING PROCESS
ST24
PERFORM ACCUMULATION PROCESS
ST25
PERFORM RATE CONTROL
END

FIG. 4

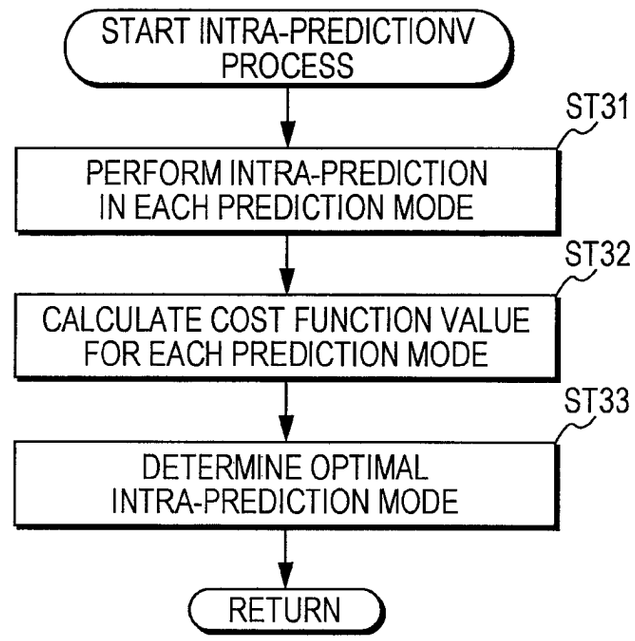

START INTRA-PREDICTIONV PROCESS
ST31
PERFORM INTRA-PREDICTION IN EACH PREDICTION MODE
ST32
CALCULATE COST FUNCTION VALUE FOR EACH PREDICTION MODE
ST33
DETERMINE OPTIMAL INTRA-PREDICTION MODE
RETURN

FIG. 5

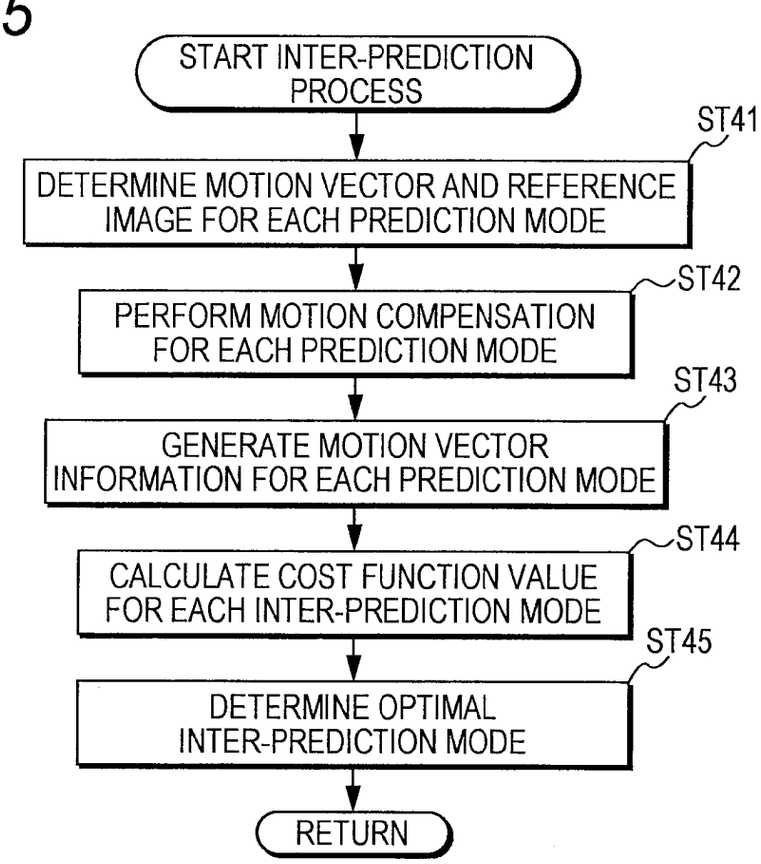

START INTER-PREDICTION PROCESS
ST41
DETERMINE MOTION VECTOR AND REFERENCE IMAGE FOR EACH PREDICTION MODE
ST42
PERFORM MOTION COMPENSATION FOR EACH PREDICTION MODE
ST43
GENERATE MOTION VECTOR INFORMATION FOR EACH PREDICTION MODE
ST44
CALCULATE COST FUNCTION VALUE FOR EACH INTER-PREDICTION MODE
ST45
DETERMINE OPTIMAL INTER-PREDICTION MODE
RETURN

FIG. 6

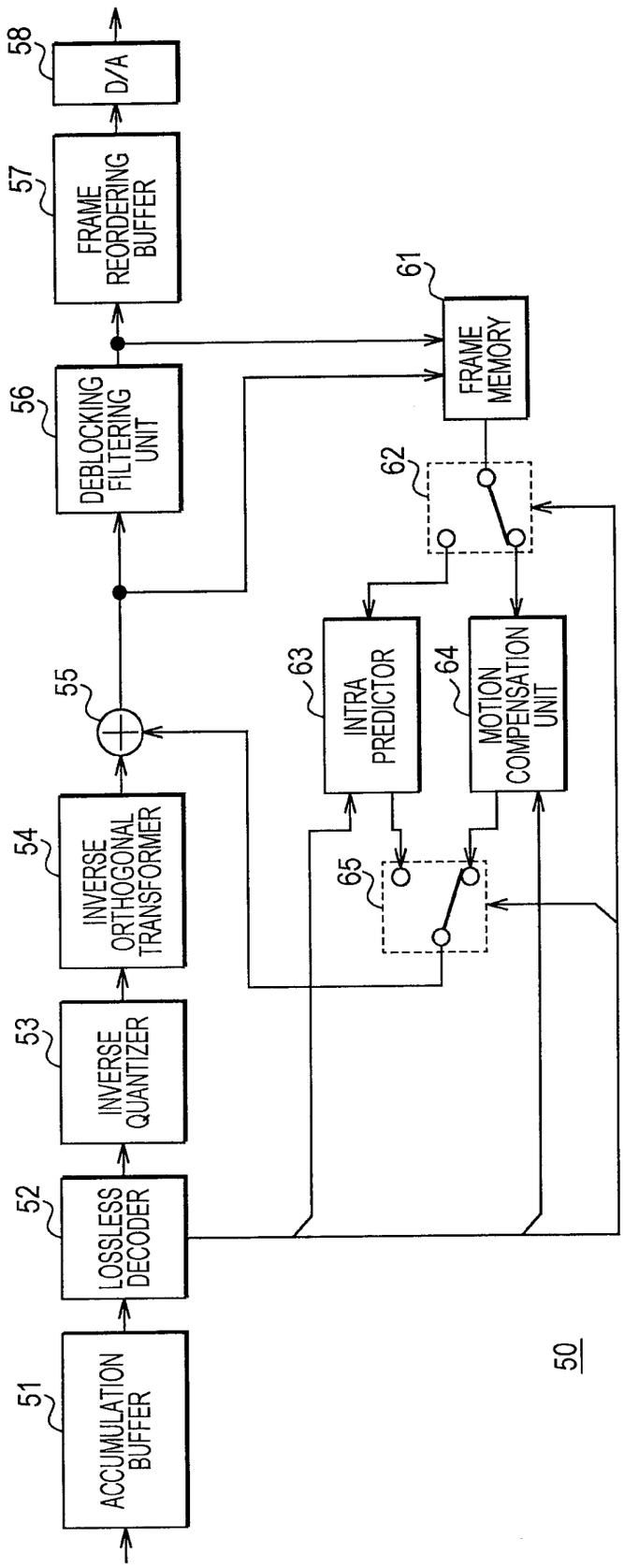

ACCUMULATION BUFFER
51
LOSSLESS DECODER
52
INVERSE QUANTIZER
53
INVERSE ORTHOGONAL TRANSFORMER
54
55
DEBLOCKING FILTERING UNIT
56
FRAME REORDERING BUFFER
57
D/A
58
FRAME MEMORY
61
62
INTRA PREDICTOR
63
MOTION COMPENSATION UNIT
64
65
50

FIG. 7

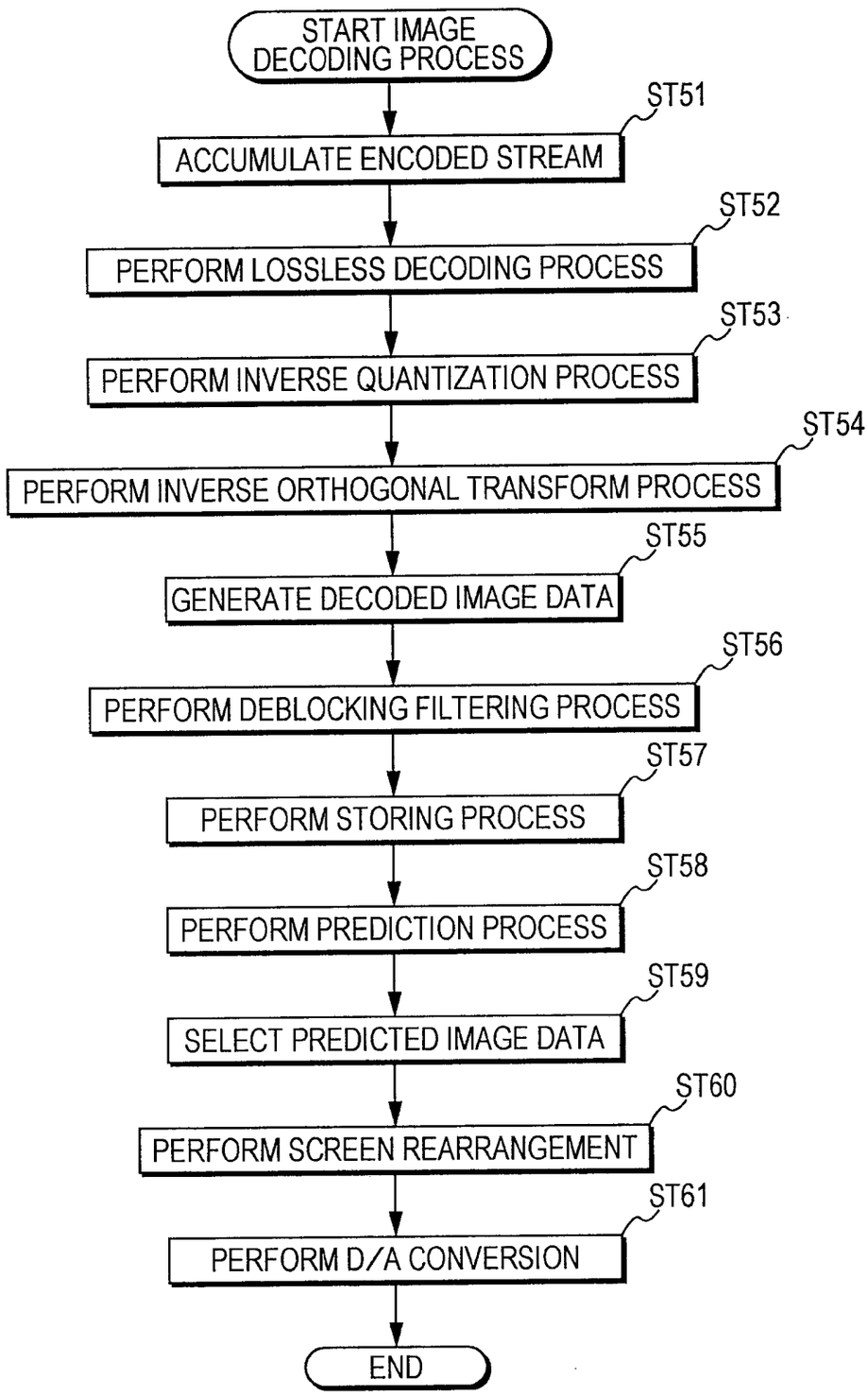

START IMAGE DECODING PROCESS
ST51
ACCUMULATE ENCODED STREAM
ST52
PERFORM LOSSLESS DECODING PROCESS
ST53
PERFORM INVERSE QUANTIZATION PROCESS
ST54
PERFORM INVERSE ORTHOGONAL TRANSFORM PROCESS
ST55
GENERATE DECODED IMAGE DATA
ST56
PERFORM DEBLOCKING FILTERING PROCESS
ST57
PERFORM STORING PROCESS
ST58
PERFORM PREDICTION PROCESS
ST59
SELECT PREDICTED IMAGE DATA
ST60
PERFORM SCREEN REARRANGEMENT
ST61
PERFORM D/A CONVERSION
END

FIG. 8

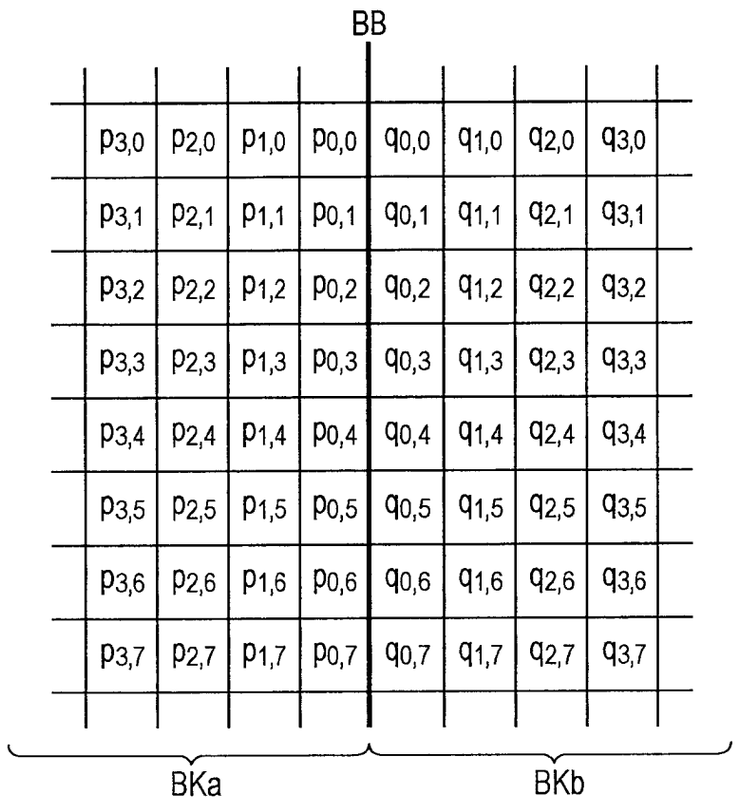

BB
p3,0 p2,0 p1,0 p0,0 q0,0 q1,0 q2,0 q3,0
p3,1 p2,1 p1,1 p0,1 q0,1 q1,1 q2,1 q3,1
p3,2 p2,2 p1,2 p0,2 q0,2 q1,2 q2,2 q3,2
p3,3 p2,3 p1,3 p0,3 q0,3 q1,3 q2,3 q3,3
p3,4 p2,4 p1,4 p0,4 q0,4 q1,4 q2,4 q3,4
p3,5 p2,5 p1,5 p0,5 q0,5 q1,5 q2,5 q3,5
p3,6 p2,6 p1,6 p0,6 q0,6 q1,6 q2,6 q3,6
p3,7 p2,7 p1,7 p0,7 q0,7 q1,7 q2,7 q3,7
BKa
BKb

FIG. 9

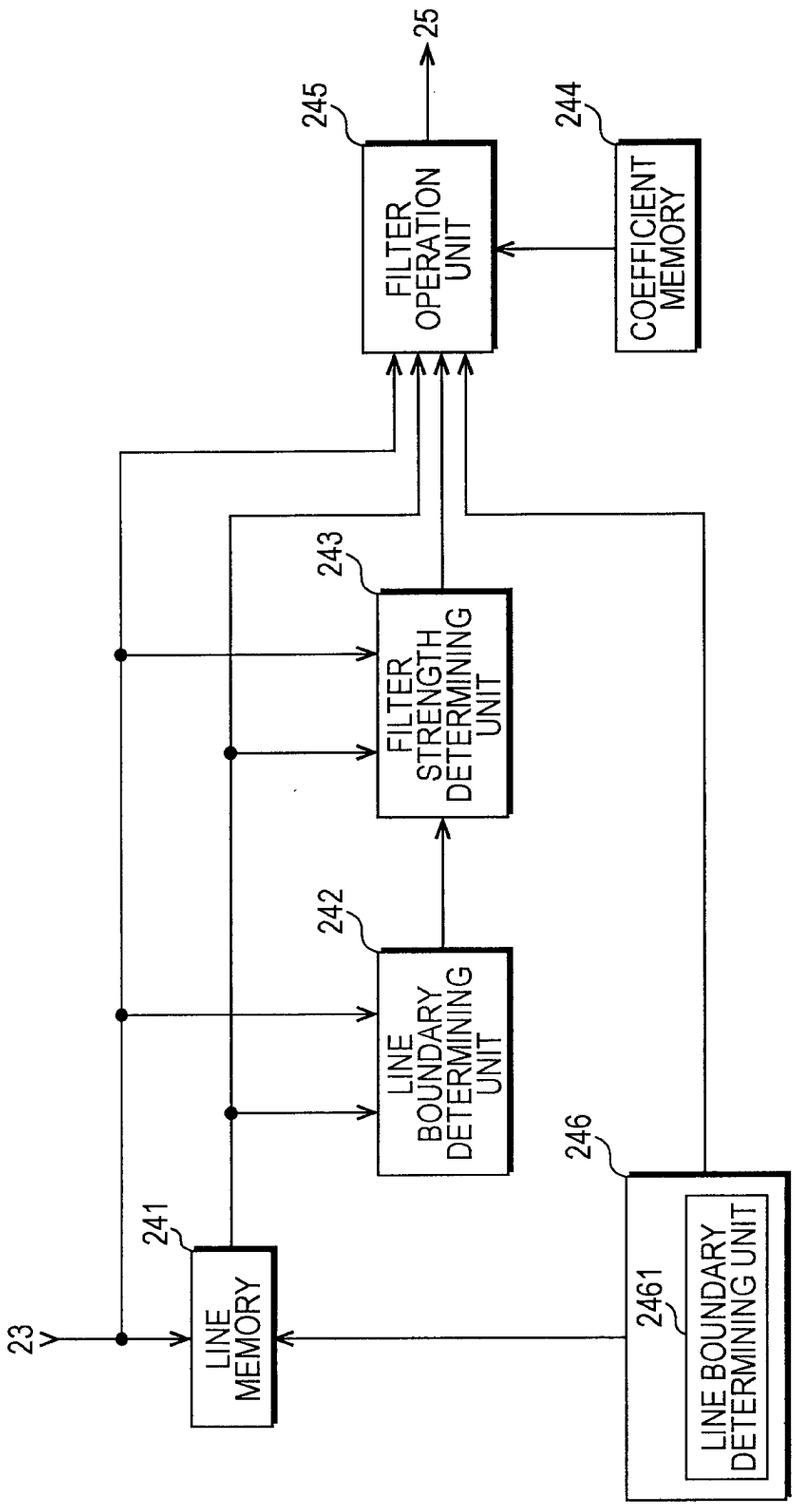

23
241
LINE MEMORY
242
LINE BOUNDARY DETERMINING UNIT
243
FILTER STRENGTH DETERMINING UNIT
244
COEFFICIENT MEMORY
245
FILTER OPERATION UNIT
25
246
2461
LINE BOUNDARY DETERMINING UNIT

FIG. 10

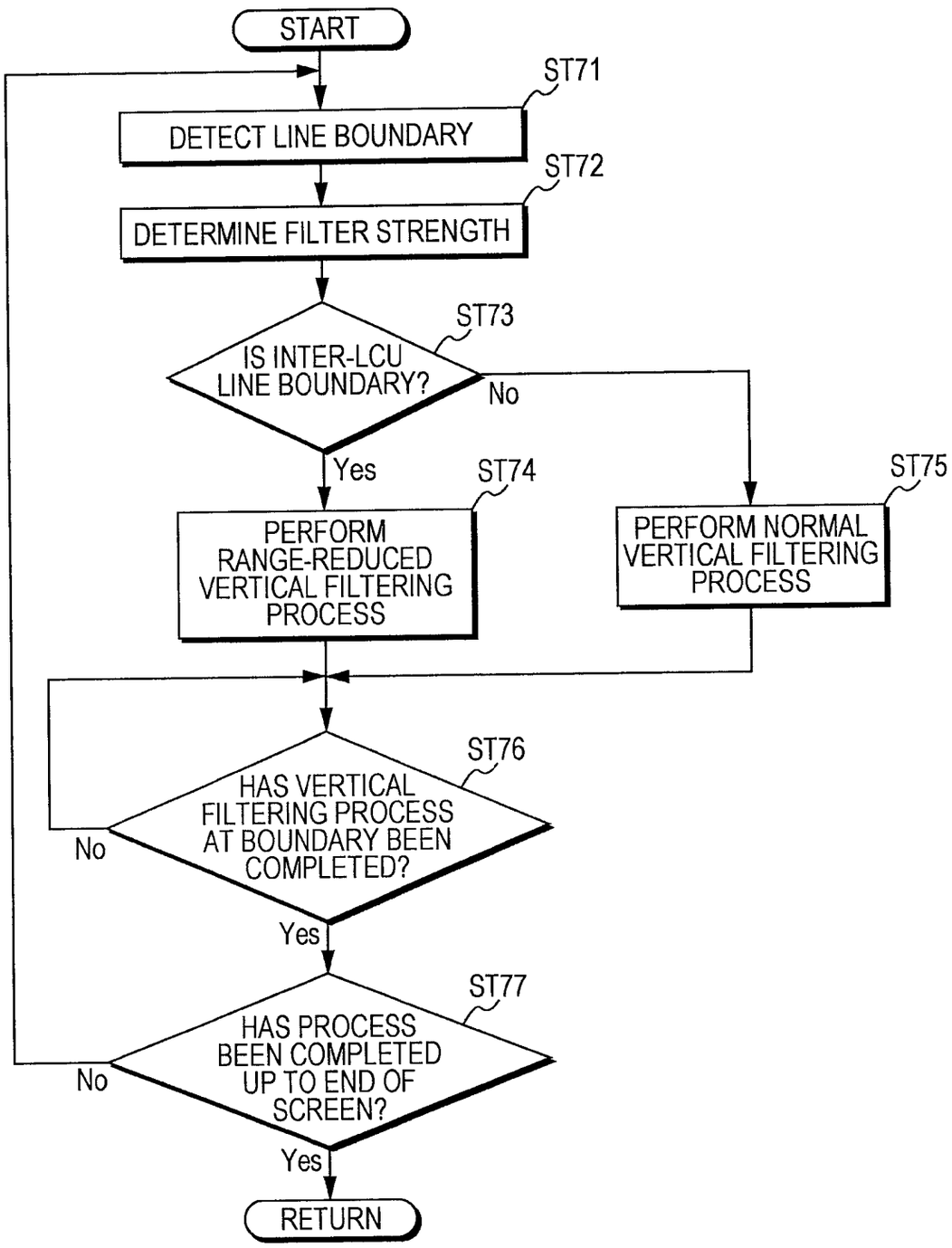

START
ST71
DETECT LINE BOUNDARY
ST72
DETERMINE FILTER STRENGTH
ST73
IS INTER-LCU LINE BOUNDARY?
No
Yes
ST74
PERFORM RANGE-REDUCED VERTICAL FILTERING PROCESS
ST75
PERFORM NORMAL VERTICAL FILTERING PROCESS
ST76
HAS VERTICAL FILTERING PROCESS AT BOUNDARY BEEN COMPLETED?
No
Yes
ST77
HAS PROCESS BEEN COMPLETED UP TO END OF SCREEN?
No
Yes
RETURN

FIG. 11

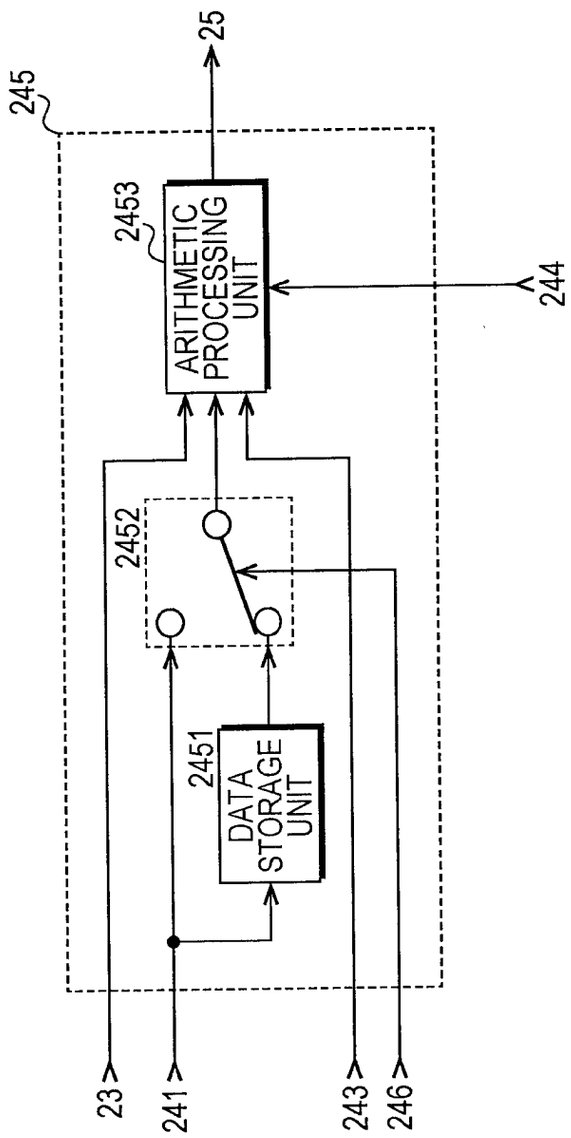

245
25
2453
ARITHMETIC PROCESSING UNIT
244
2452
2451
DATA STORAGE UNIT
23
241
243
246

FIG. 12

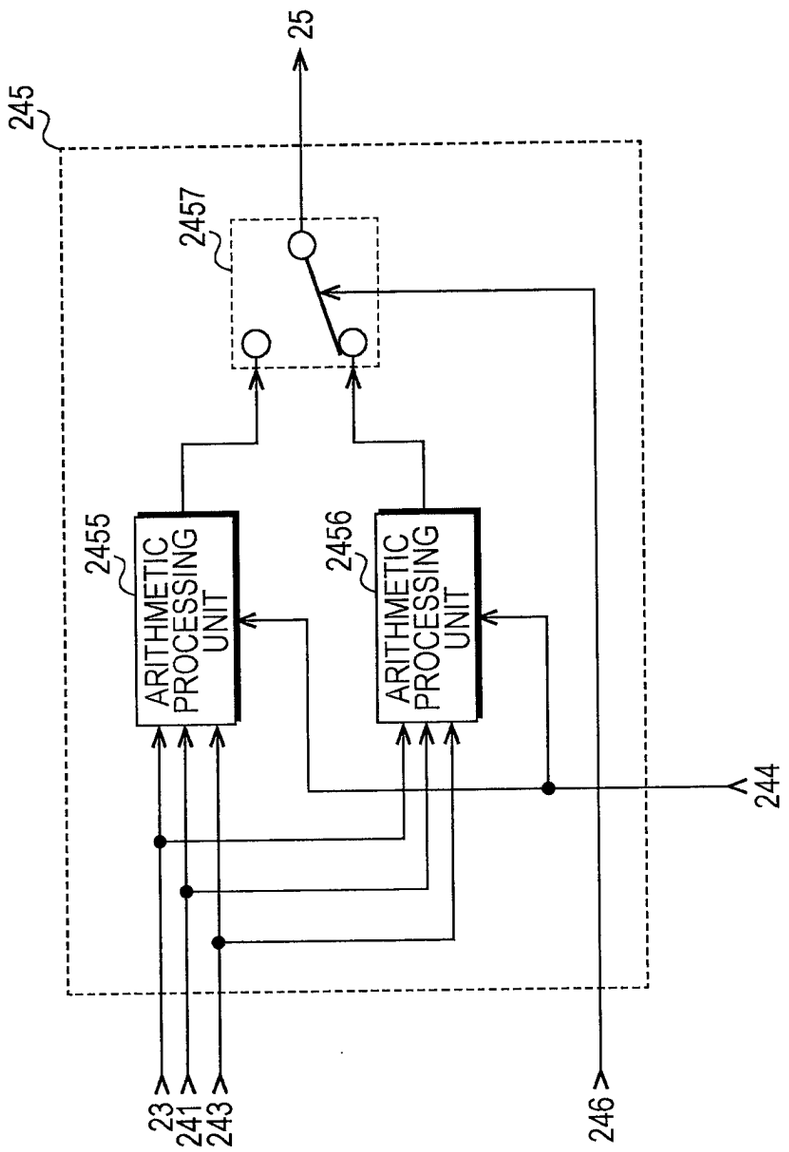

25
245
2457
2455
ARITHMETIC PROCESSING UNIT
2456
ARITHMETIC PROCESSING UNIT
244
23
241
243
246

FIG. 13

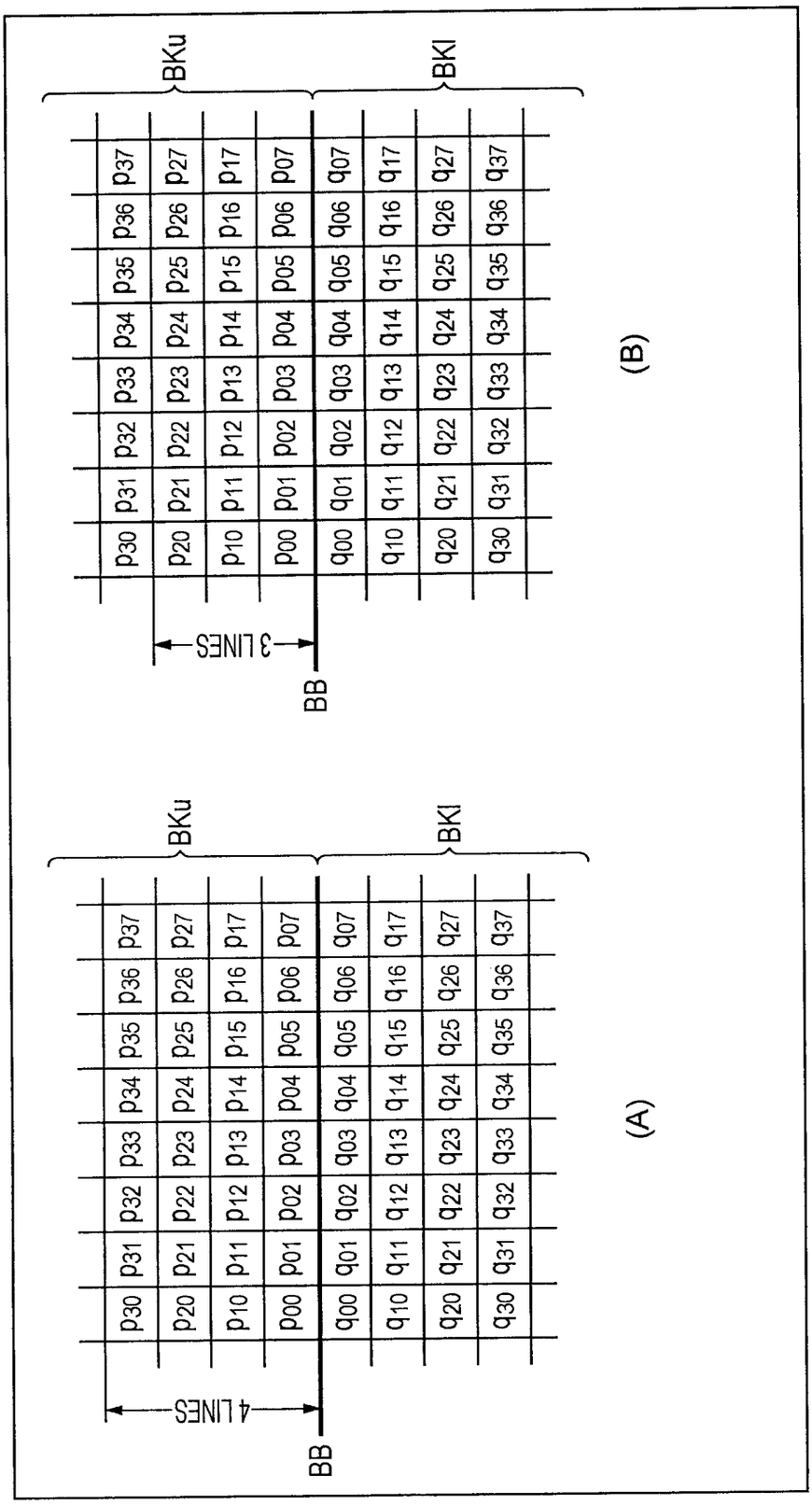


FIG. 14

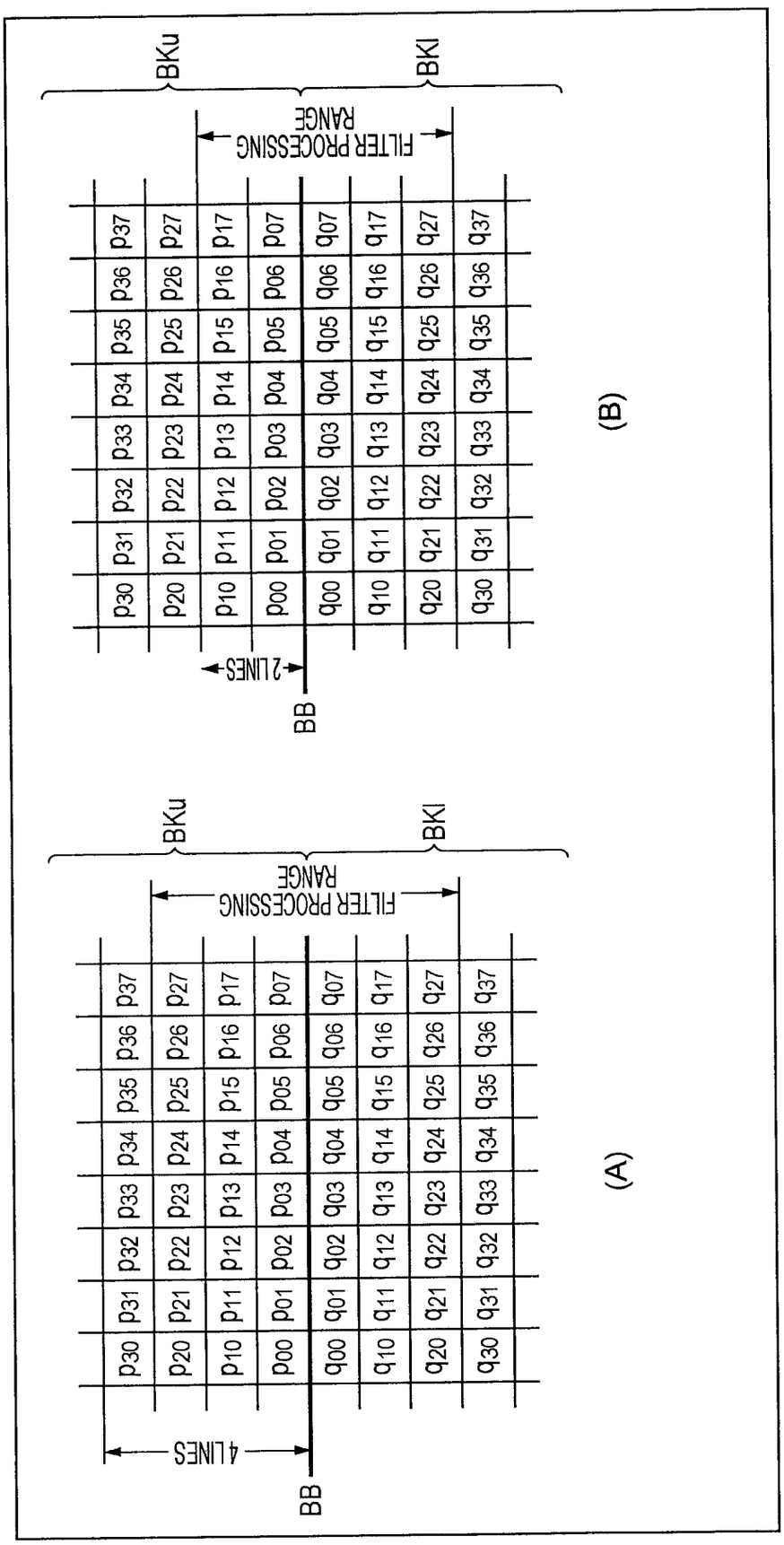

BKu
BKl
FILTER PROCESSING RANGE
4 LINES
BB
(A)
BKu
BKl
FILTER PROCESSING RANGE
2 LINES
BB
(B)

FIG. 15

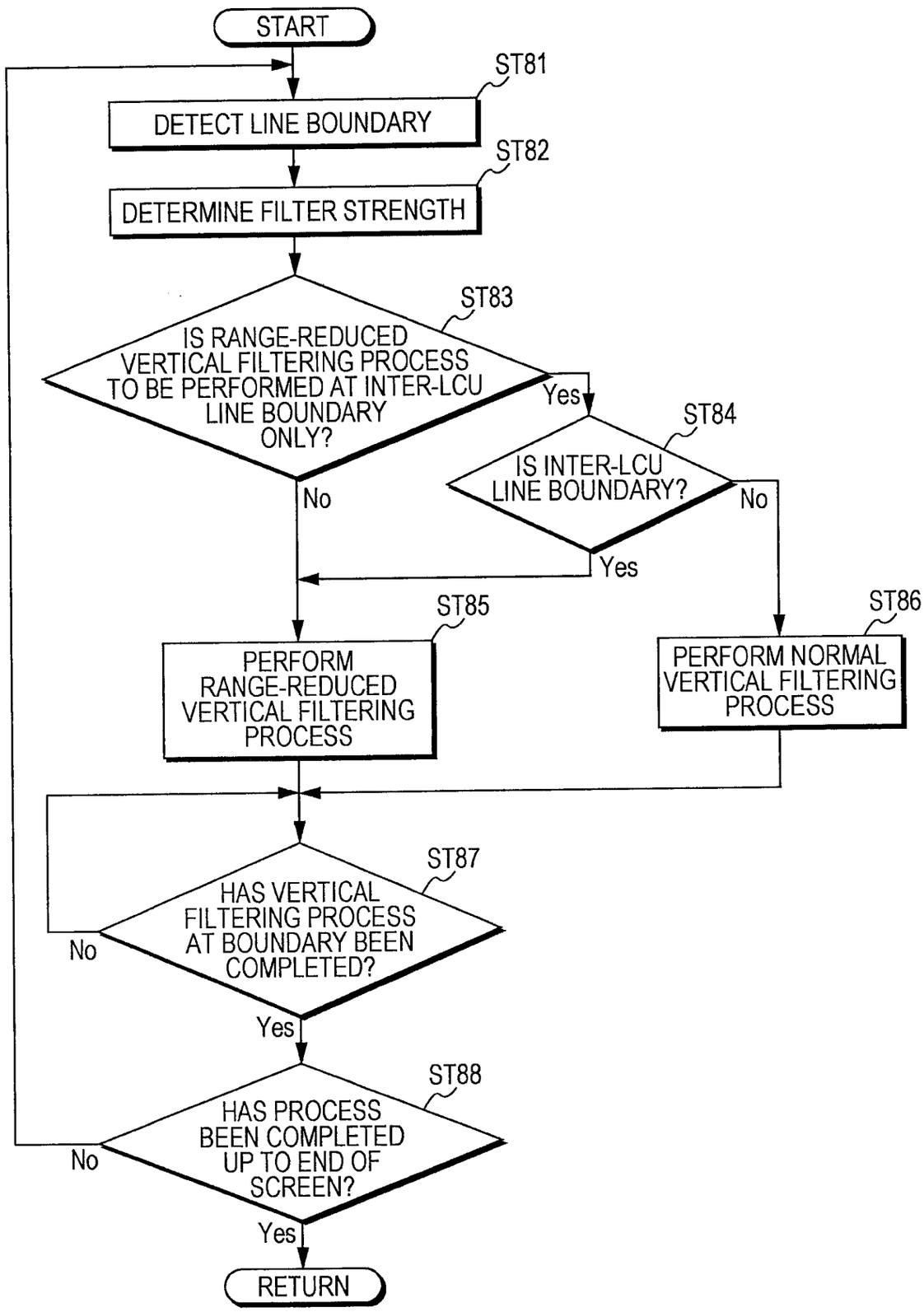

START
ST81
DETECT LINE BOUNDARY
ST82
DETERMINE FILTER STRENGTH
ST83
IS RANGE-REDUCED VERTICAL FILTERING PROCESS TO BE PERFORMED AT INTER-LCU LINE BOUNDARY ONLY?
Yes
No
ST84
IS INTER-LCU LINE BOUNDARY?
No
Yes
ST85
PERFORM RANGE-REDUCED VERTICAL FILTERING PROCESS
ST86
PERFORM NORMAL VERTICAL FILTERING PROCESS
ST87
HAS VERTICAL FILTERING PROCESS AT BOUNDARY BEEN COMPLETED?
No
Yes
ST88
HAS PROCESS BEEN COMPLETED UP TO END OF SCREEN?
No
Yes
RETURN

FIG. 16

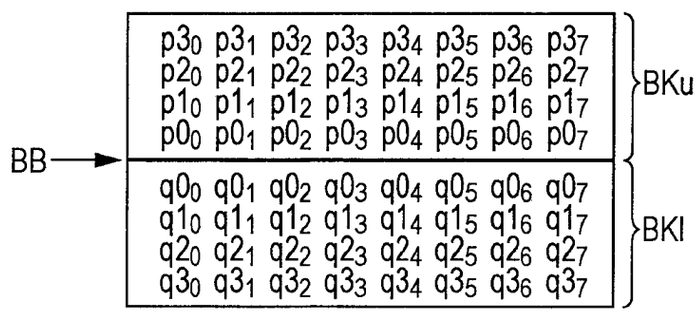

BB
BKu
BKl

FIG. 17

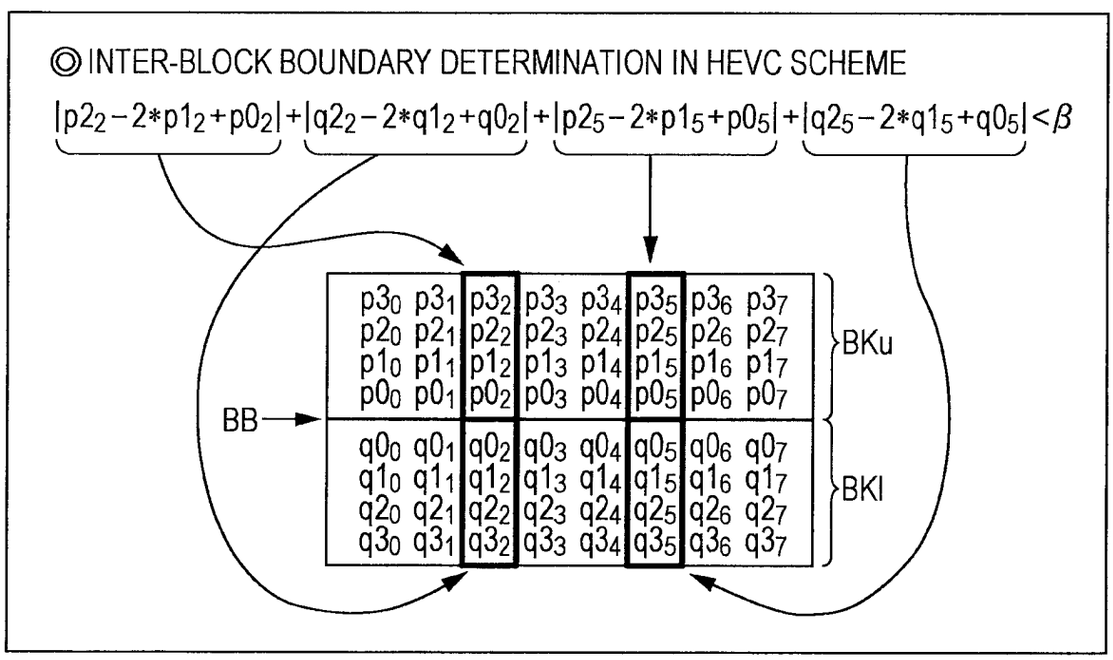

◎INTER-BLOCK BOUNDARY DETERMINATION IN HEVC SCHEME
|p22−2*p12+p02|+|q22−2*q12+q02|+|p25−2*p15+p05|+|q25−2*q15+q05|<β
BB
BKu
BKl

FIG. 18

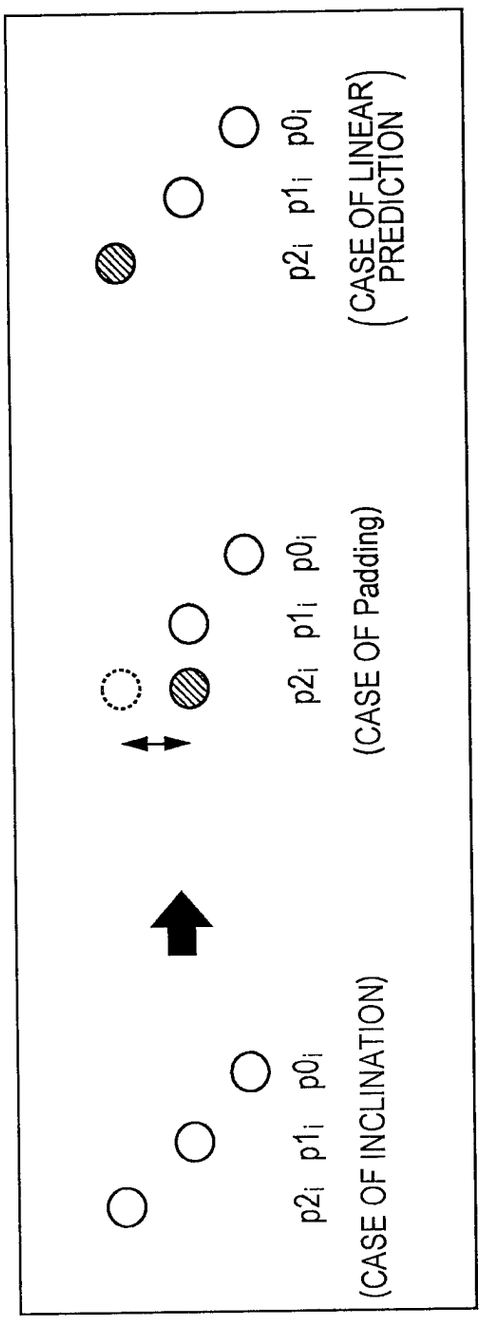

p2i p1i p0i
(CASE OF LINEAR PREDICTION)
p2i p1i p0i
(CASE OF Padding)
p2i p1i p0i
(CASE OF INCLINATION)

FIG. 19

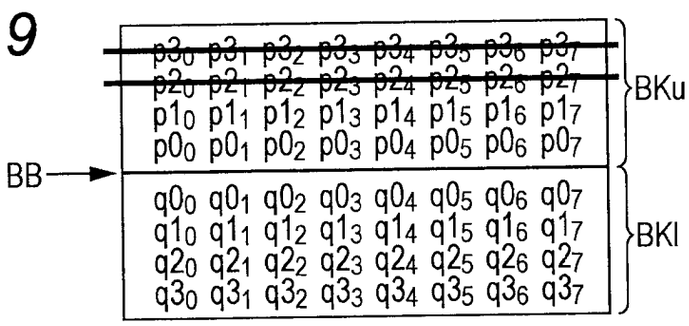

p30 p31 p32 p33 p34 p35 p36 p37
p20 p21 p22 p23 p24 p25 p26 p27
p10 p11 p12 p13 p14 p15 p16 p17
p00 p01 p02 p03 p04 p05 p06 p07
BKu
BB
q00 q01 q02 q03 q04 q05 q06 q07
q10 q11 q12 q13 q14 q15 q16 q17
q20 q21 q22 q23 q24 q25 q26 q27
q30 q31 q32 q33 q34 q35 q36 q37
BKl

FIG. 20

	AI_HE				RA_HE				LB_HE				LP_HE			
	22	27	32	37	22	27	32	37	22	27	32	37	22	27	32	37
Class A	16	25	23	29	12	19	19	18								
Class B	9	37	42	50	15	29	25	84	9	55	53	73	12	38	50	82
Class C	29	40	54	109	9	28	34	35	27	15	34	44	5	24	34	49
Class D	7	13	57	54	6	9	40	42	3	8	20	22	4	8	18	20
Class E	15	16	20	28					5	7	15	17	6	11	16	20
Class F	190	190	190	191	178	189	191	190	191	188	189	182	191	188	181	169

FIG. 21

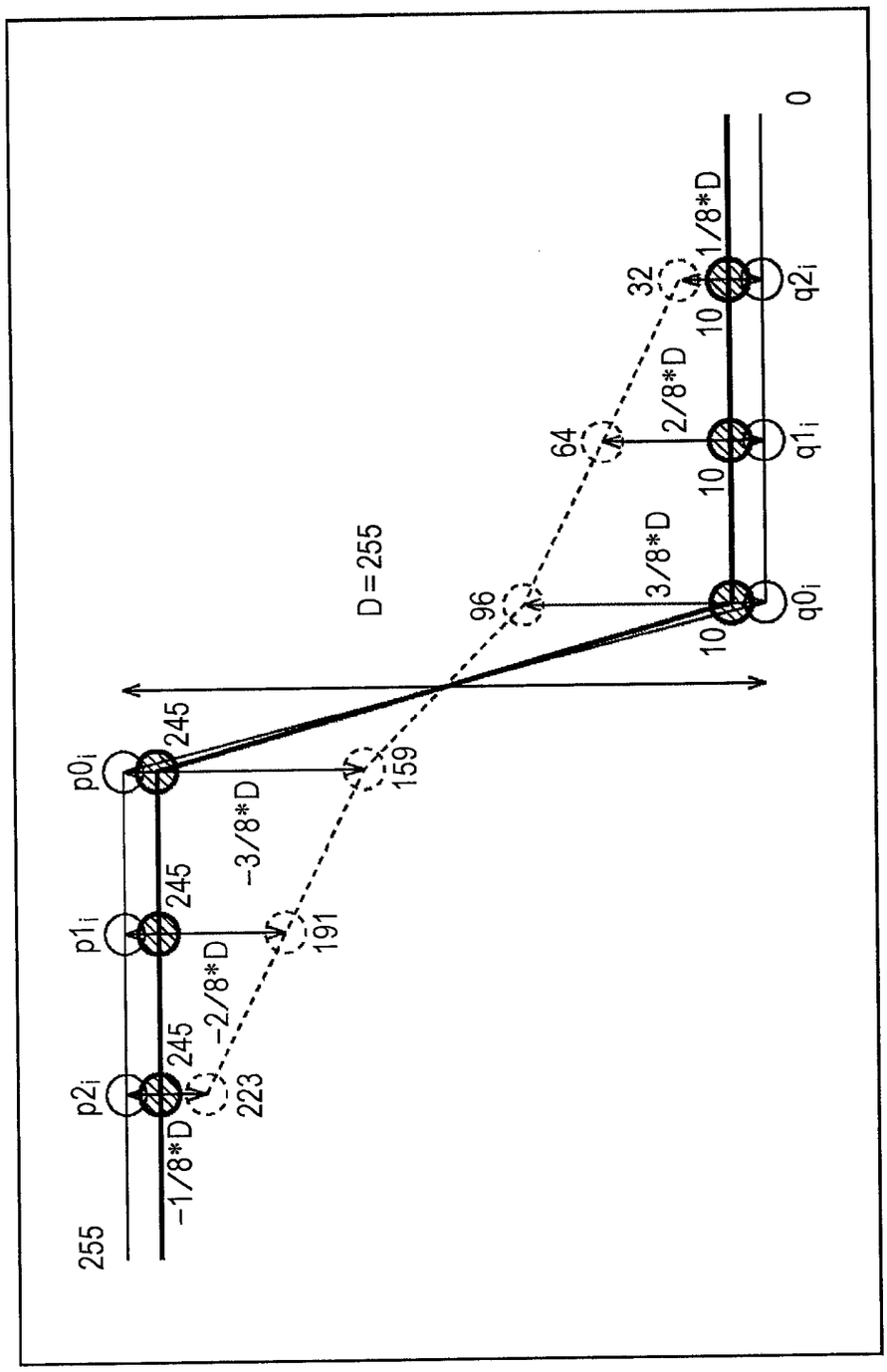

255
p2i
p1i
p0i
245
245
245
−1/8*D
−2/8*D
−3/8*D
223
191
159
D=255
96
64
32
10
10
10
3/8*D
2/8*D
1/8*D
q0i
q1i
q2i
0

FIG. 22

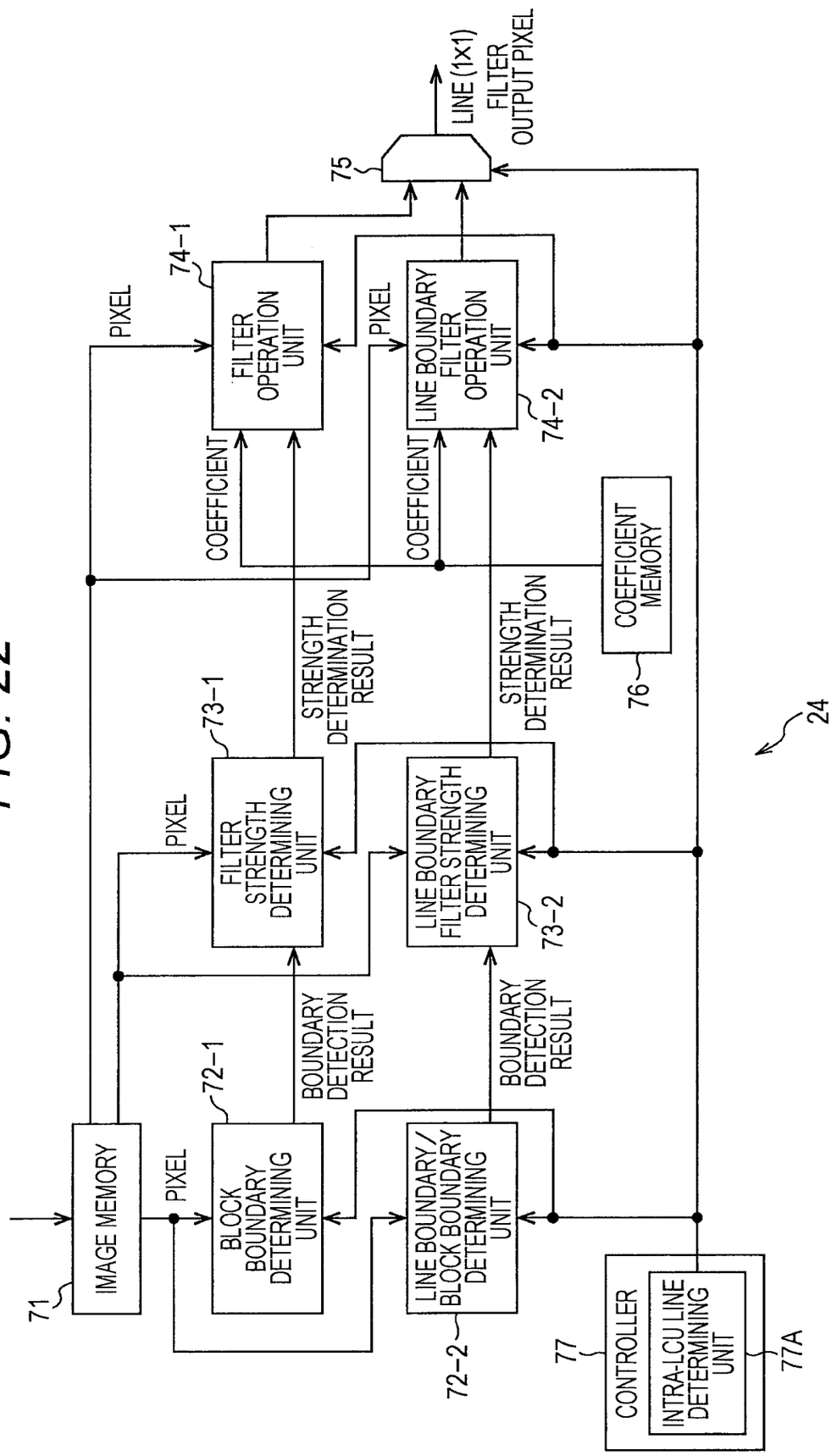

71
IMAGE MEMORY
PIXEL
72-1
BLOCK BOUNDARY DETERMINING UNIT
BOUNDARY DETECTION RESULT
72-2
LINE BOUNDARY/ BLOCK BOUNDARY DETERMINING UNIT
BOUNDARY DETECTION RESULT
73-1
FILTER STRENGTH DETERMINING UNIT
STRENGTH DETERMINATION RESULT
73-2
LINE BOUNDARY FILTER STRENGTH DETERMINING UNIT
STRENGTH DETERMINATION RESULT
74-1
FILTER OPERATION UNIT
COEFFICIENT
74-2
LINE BOUNDARY FILTER OPERATION UNIT
COEFFICIENT
76
COEFFICIENT MEMORY
75
LINE (1x1) FILTER OUTPUT PIXEL
24
77
CONTROLLER
77A
INTRA-LCU LINE DETERMINING UNIT

FIG. 23

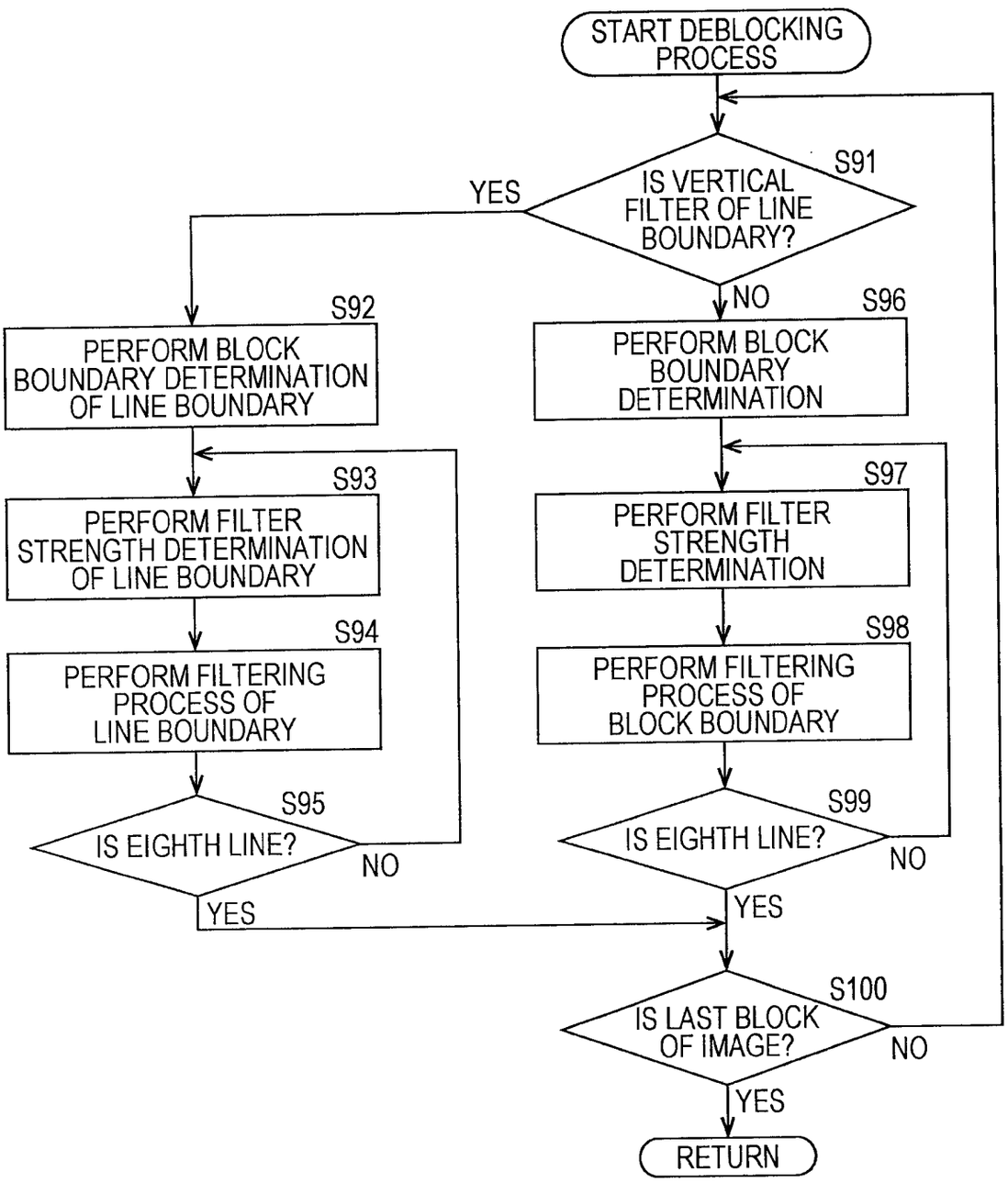

START DEBLOCKING PROCESS
S91
IS VERTICAL FILTER OF LINE BOUNDARY?
YES
NO
S92
PERFORM BLOCK BOUNDARY DETERMINATION OF LINE BOUNDARY
S93
PERFORM FILTER STRENGTH DETERMINATION OF LINE BOUNDARY
S94
PERFORM FILTERING PROCESS OF LINE BOUNDARY
S95
IS EIGHTH LINE?
NO
YES
S96
PERFORM BLOCK BOUNDARY DETERMINATION
S97
PERFORM FILTER STRENGTH DETERMINATION
S98
PERFORM FILTERING PROCESS OF BLOCK BOUNDARY
S99
IS EIGHTH LINE?
NO
YES
S100
IS LAST BLOCK OF IMAGE?
NO
YES
RETURN

FIG. 24

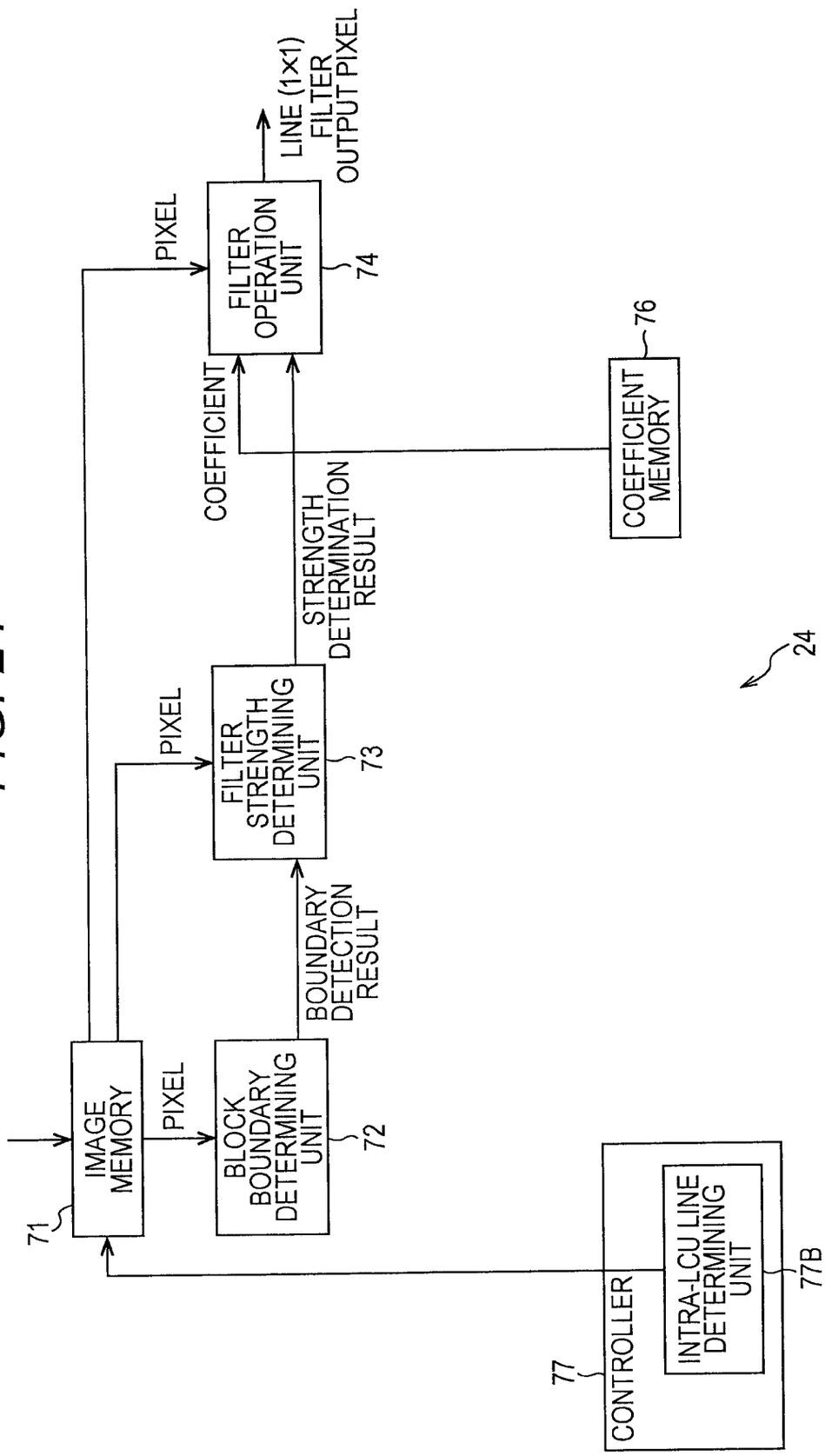

IMAGE MEMORY
71
PIXEL
BLOCK BOUNDARY DETERMINING UNIT
72
BOUNDARY DETECTION RESULT
FILTER STRENGTH DETERMINING UNIT
73
PIXEL
STRENGTH DETERMINATION RESULT
FILTER OPERATION UNIT
74
PIXEL
COEFFICIENT
LINE (1×1) FILTER OUTPUT PIXEL
COEFFICIENT MEMORY
76
24
CONTROLLER
77
INTRA-LCU LINE DETERMINING UNIT
77B

FIG. 25

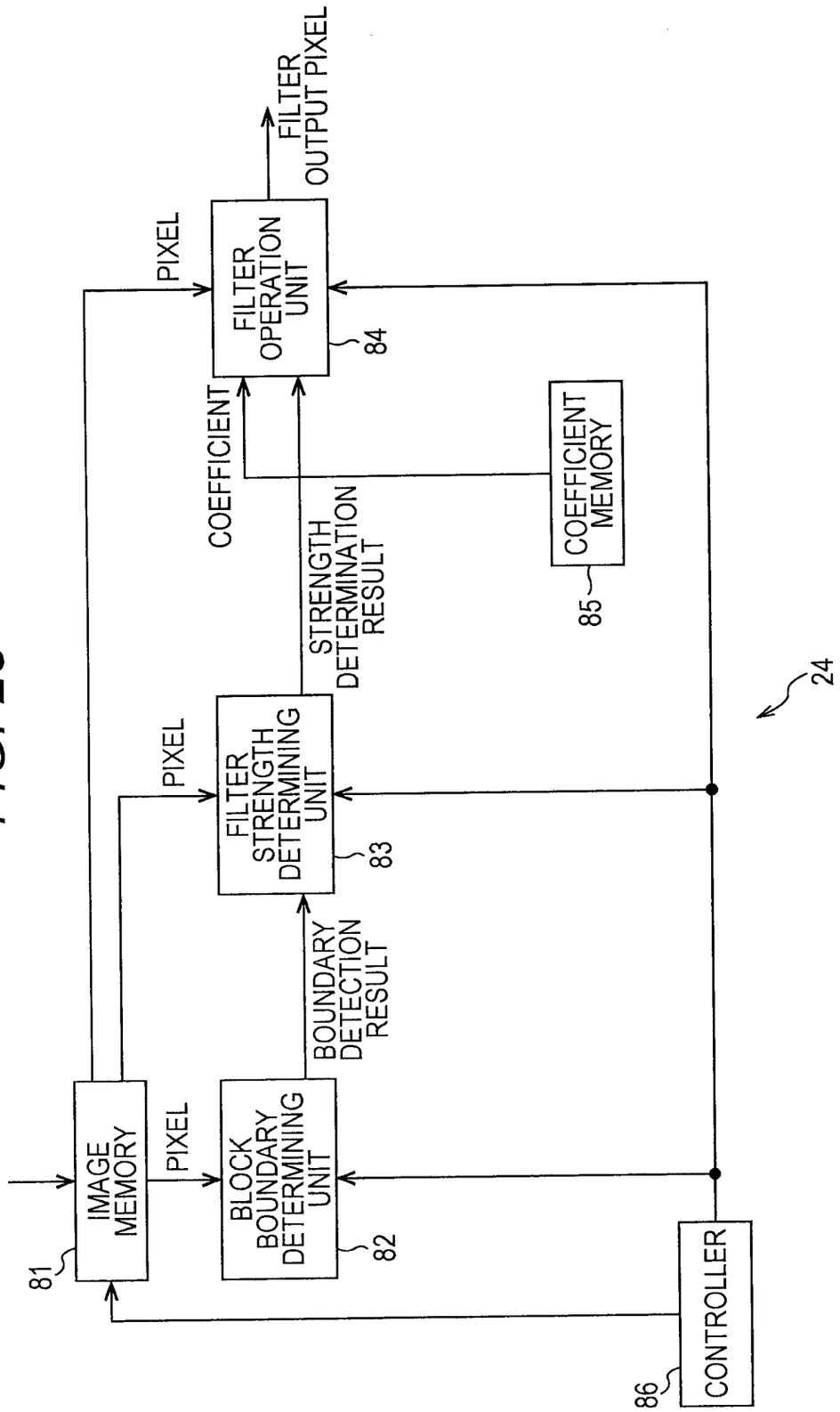

IMAGE MEMORY
81
PIXEL
BLOCK BOUNDARY DETERMINING UNIT
82
BOUNDARY DETECTION RESULT
FILTER STRENGTH DETERMINING UNIT
83
PIXEL
STRENGTH DETERMINATION RESULT
COEFFICIENT
COEFFICIENT MEMORY
85
FILTER OPERATION UNIT
84
PIXEL
FILTER OUTPUT PIXEL
CONTROLLER
86
24

FIG. 26

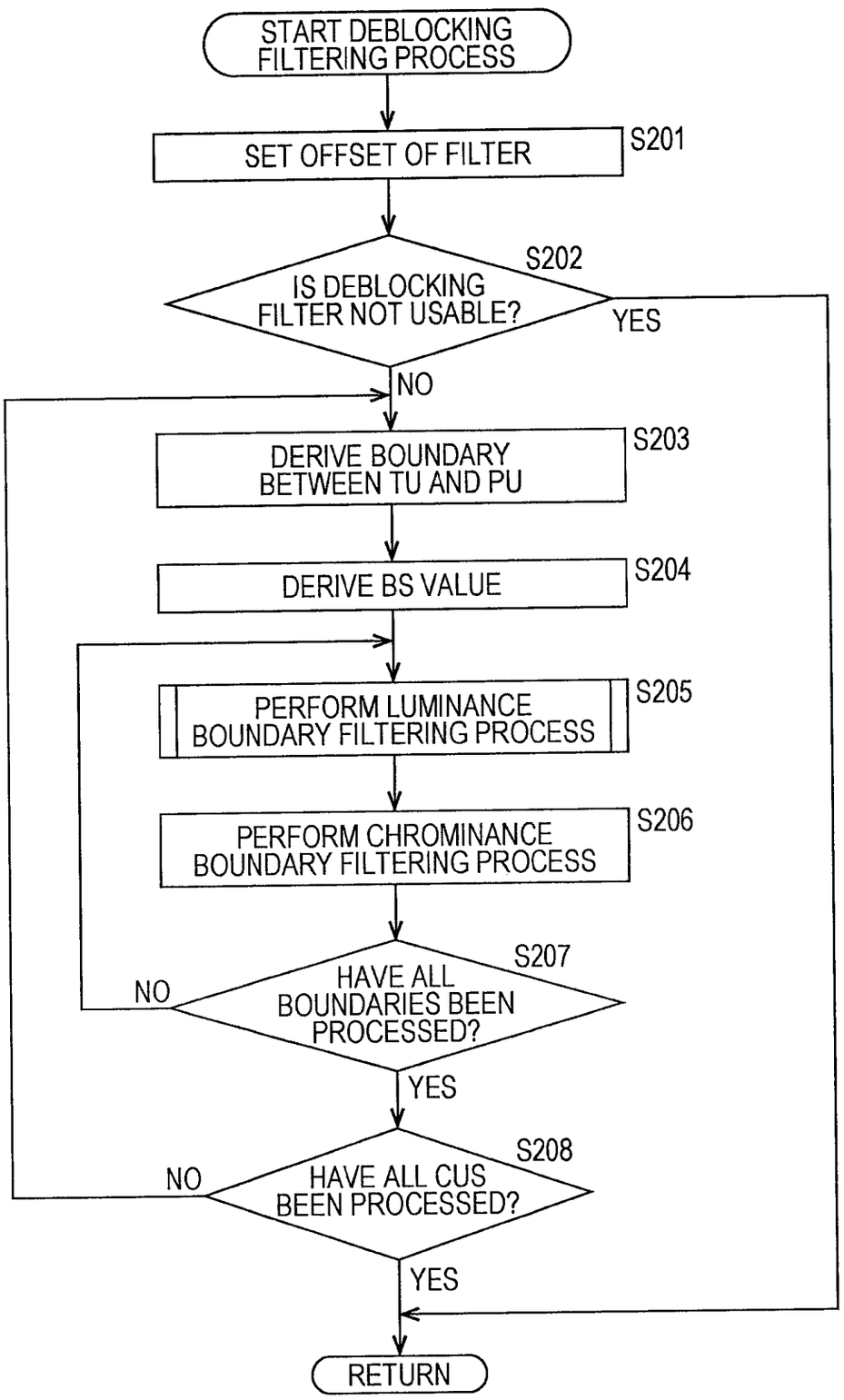

START DEBLOCKING FILTERING PROCESS
SET OFFSET OF FILTER
S201
IS DEBLOCKING FILTER NOT USABLE?
S202
YES
NO
DERIVE BOUNDARY BETWEEN TU AND PU
S203
DERIVE BS VALUE
S204
PERFORM LUMINANCE BOUNDARY FILTERING PROCESS
S205
PERFORM CHROMINANCE BOUNDARY FILTERING PROCESS
S206
HAVE ALL BOUNDARIES BEEN PROCESSED?
S207
NO
YES
HAVE ALL CUS BEEN PROCESSED?
S208
NO
YES
RETURN

FIG. 27

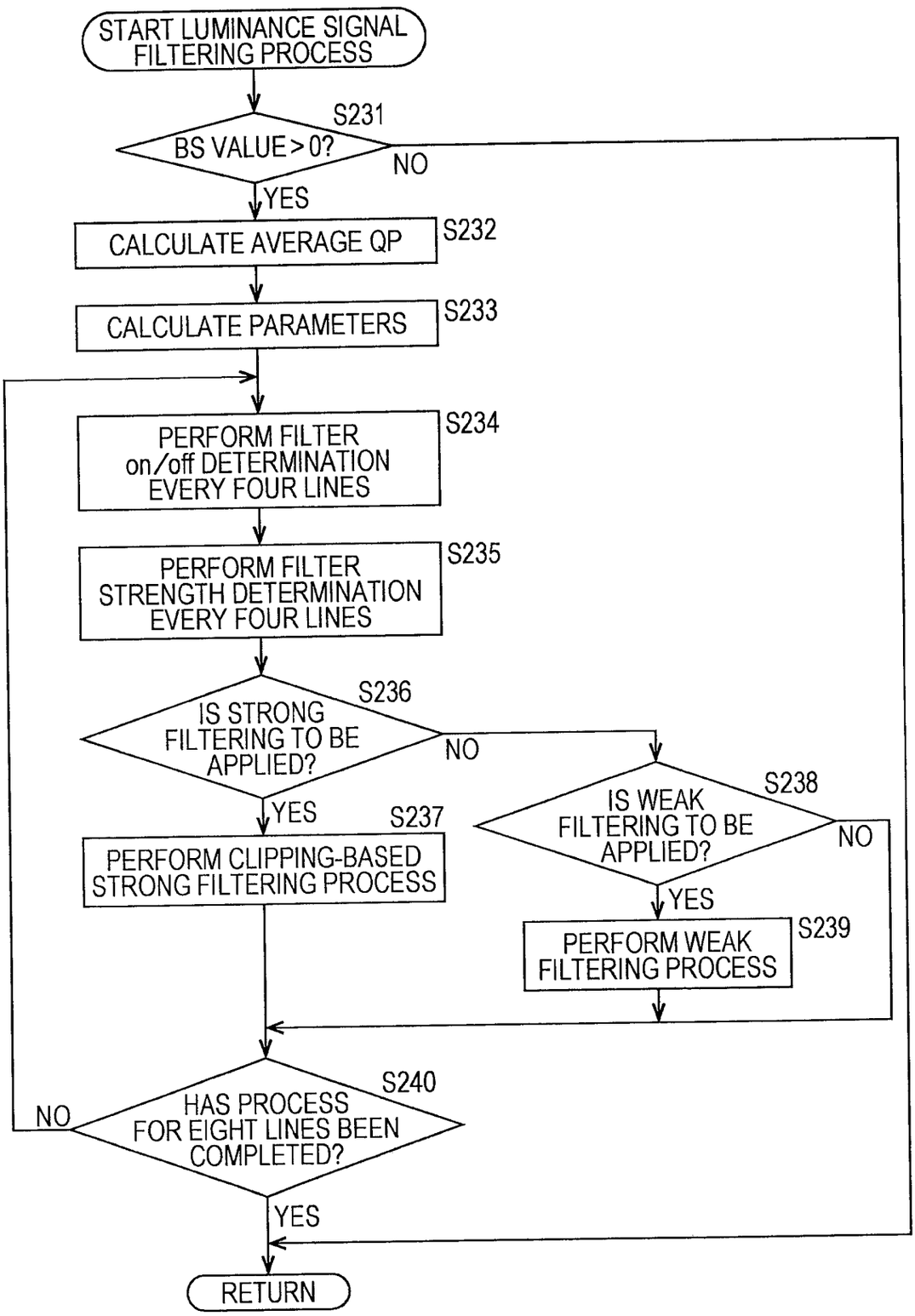

START LUMINANCE SIGNAL FILTERING PROCESS
S231
BS VALUE > 0?
NO
YES
CALCULATE AVERAGE QP
S232
CALCULATE PARAMETERS
S233
PERFORM FILTER on/off DETERMINATION EVERY FOUR LINES
S234
PERFORM FILTER STRENGTH DETERMINATION EVERY FOUR LINES
S235
S236
IS STRONG FILTERING TO BE APPLIED?
NO
YES
S237
PERFORM CLIPPING-BASED STRONG FILTERING PROCESS
S238
IS WEAK FILTERING TO BE APPLIED?
NO
YES
S239
PERFORM WEAK FILTERING PROCESS
S240
HAS PROCESS FOR EIGHT LINES BEEN COMPLETED?
NO
YES
RETURN

FIG. 28

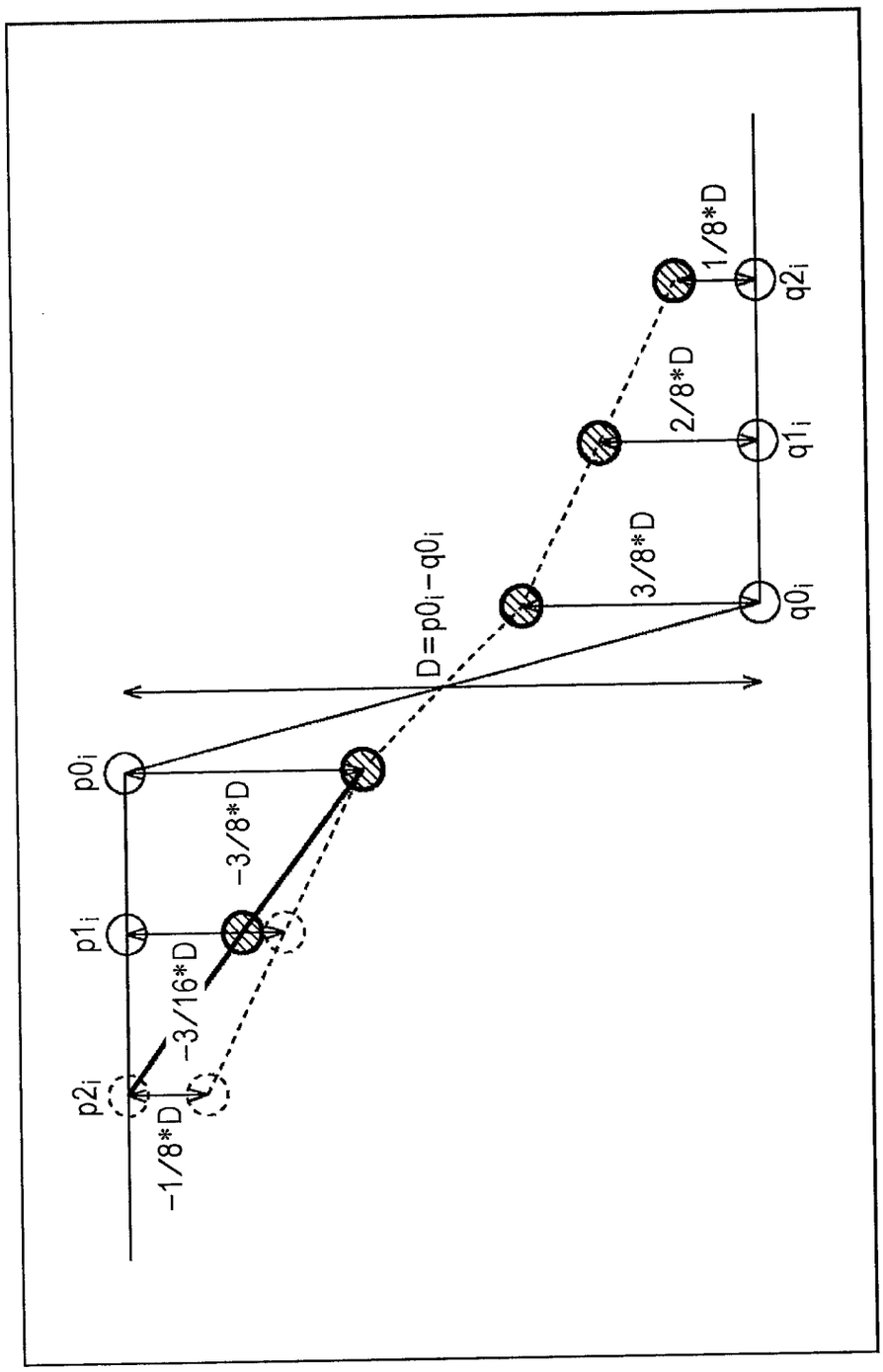


FIG. 29

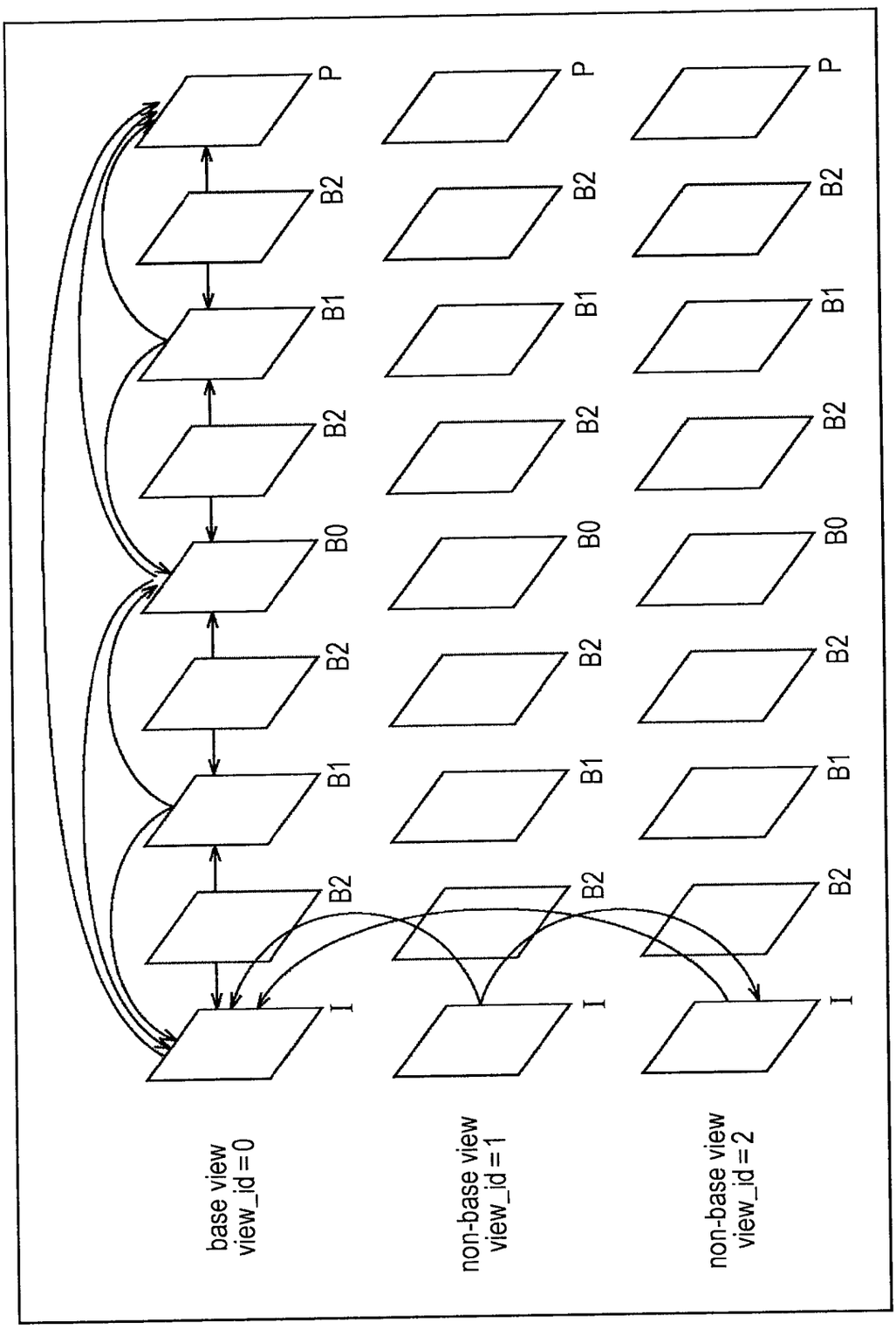

base view
view_id = 0
non-base view
view_id = 1
non-base view
view_id = 2
I
B2
B1
B2
B0
B2
B1
B2
P

FIG. 30

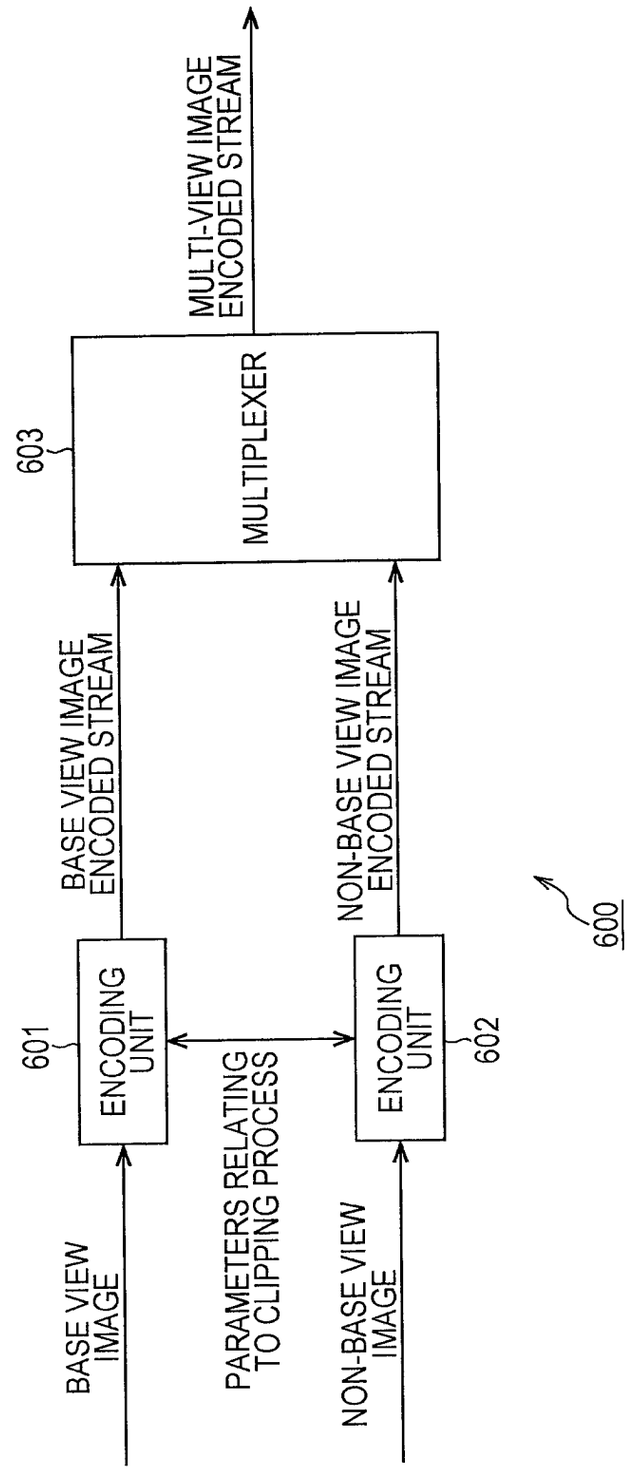

MULTI-VIEW IMAGE ENCODED STREAM
603
MULTIPLEXER
BASE VIEW IMAGE ENCODED STREAM
NON-BASE VIEW IMAGE ENCODED STREAM
601
ENCODING UNIT
602
ENCODING UNIT
600
BASE VIEW IMAGE
PARAMETERS RELATING TO CLIPPING PROCESS
NON-BASE VIEW IMAGE

FIG. 31

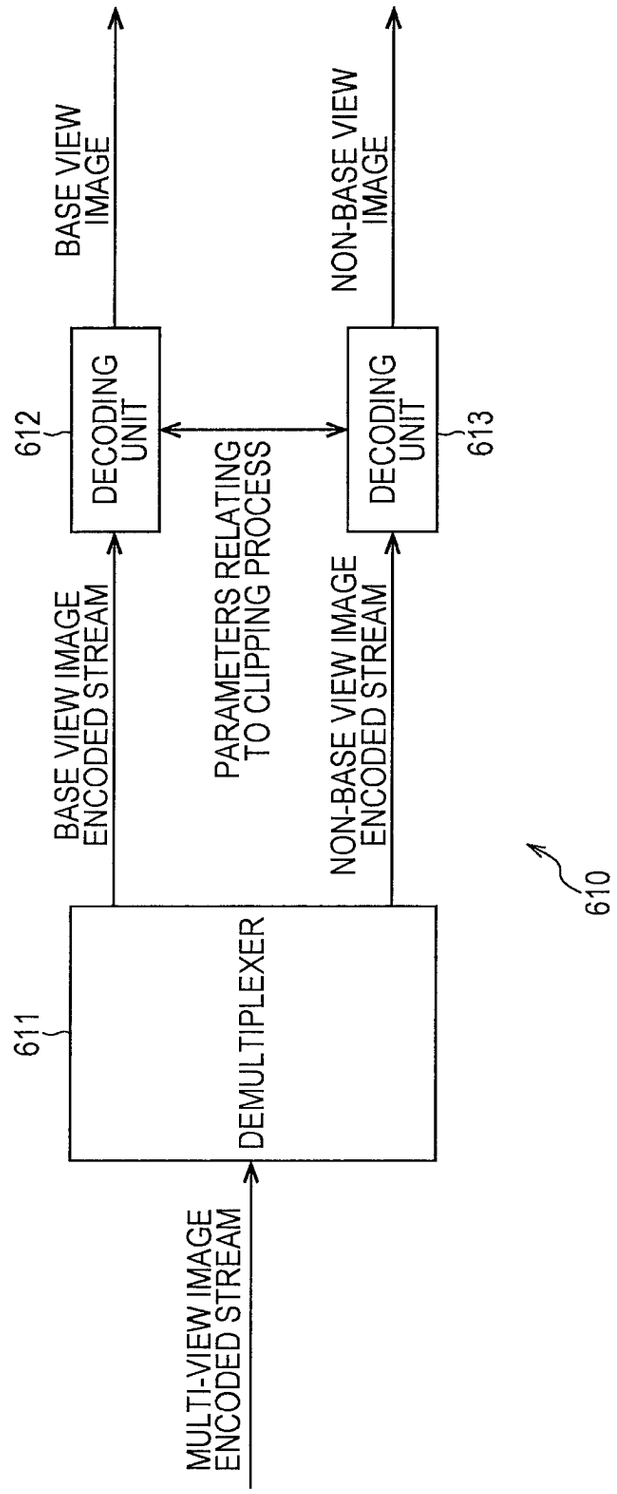

MULTI-VIEW IMAGE ENCODED STREAM
DEMULTIPLEXER
611
BASE VIEW IMAGE ENCODED STREAM
NON-BASE VIEW IMAGE ENCODED STREAM
PARAMETERS RELATING TO CLIPPING PROCESS
DECODING UNIT
612
DECODING UNIT
613
BASE VIEW IMAGE
NON-BASE VIEW IMAGE
610

FIG. 32

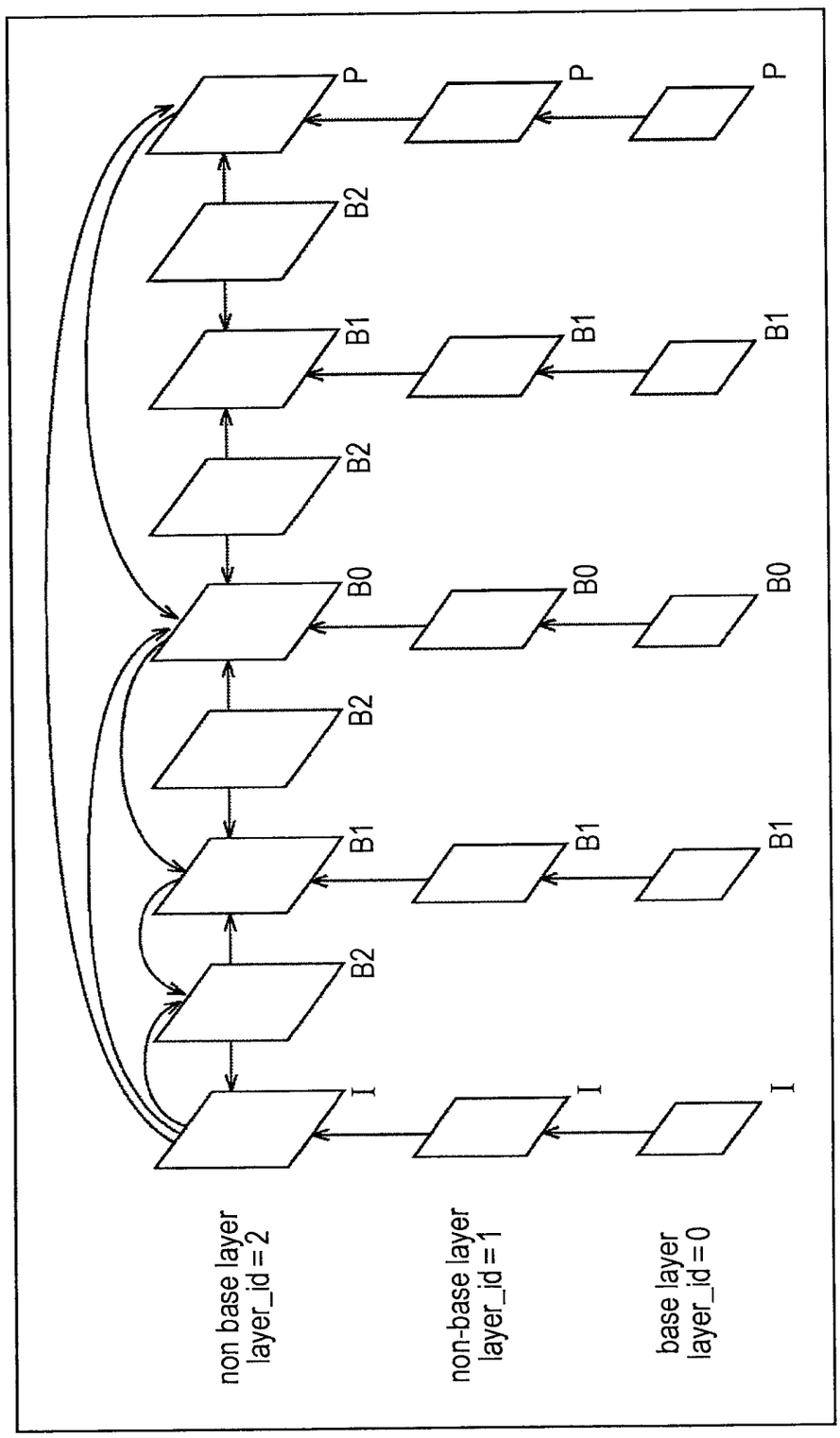

P
P
P
B2
B1
B1
B1
B2
B0
B0
B0
B2
B1
B1
B1
B2
I
I
I
non base layer
layer_id = 2
non-base layer
layer_id = 1
base layer
layer_id = 0

FIG. 33

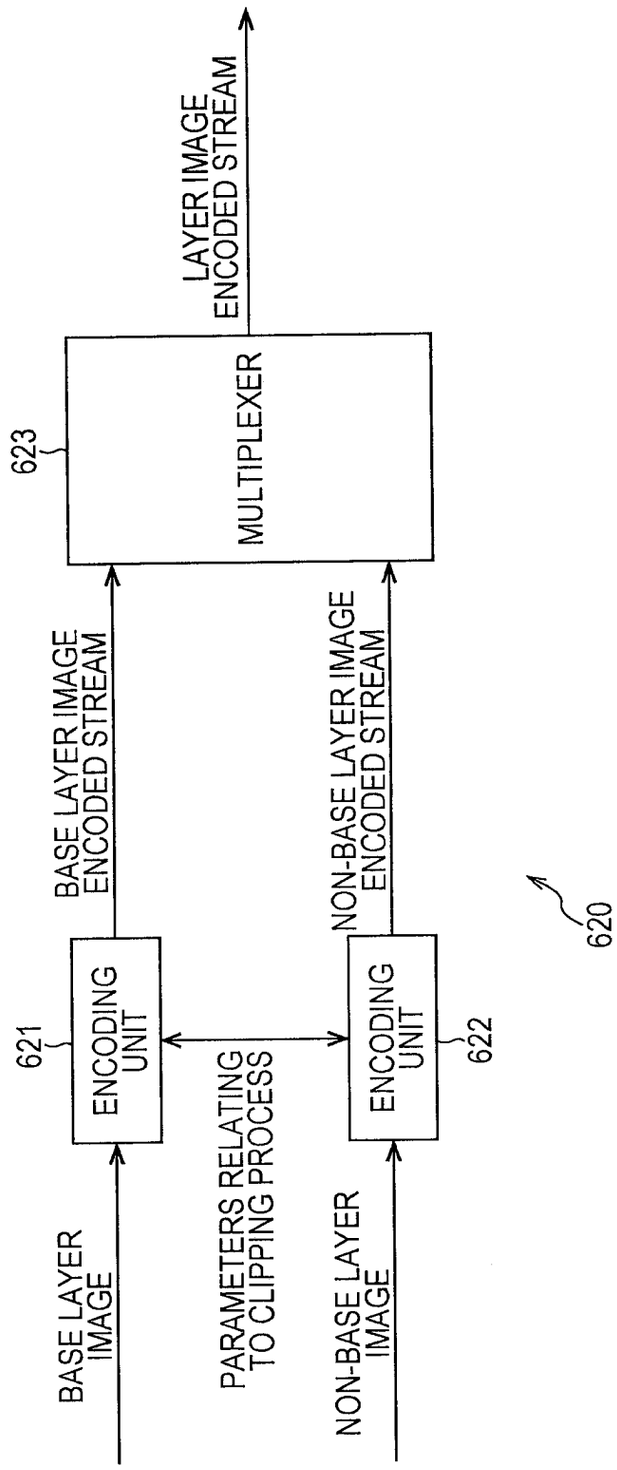

LAYER IMAGE ENCODED STREAM
MULTIPLEXER
623
BASE LAYER IMAGE ENCODED STREAM
NON-BASE LAYER IMAGE ENCODED STREAM
620
621
ENCODING UNIT
622
ENCODING UNIT
BASE LAYER IMAGE
PARAMETERS RELATING TO CLIPPING PROCESS
NON-BASE LAYER IMAGE

FIG. 34

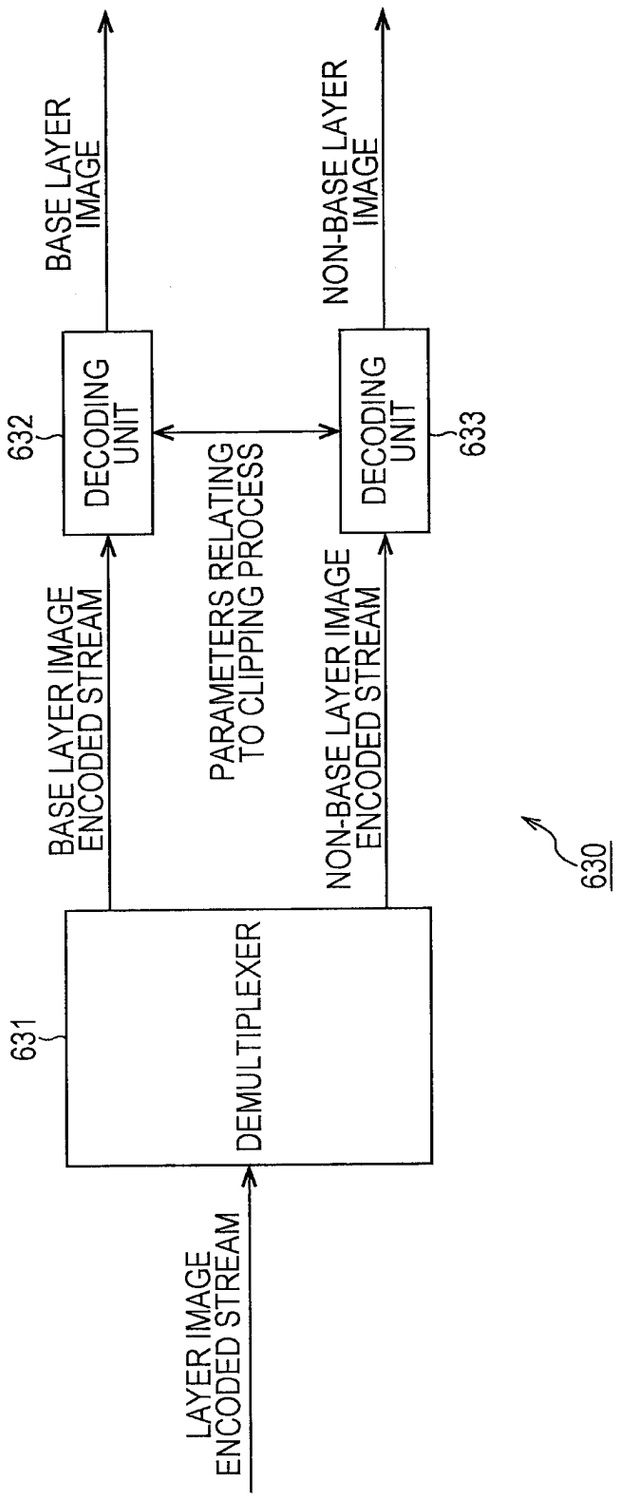

LAYER IMAGE ENCODED STREAM
631
DEMULTIPLEXER
BASE LAYER IMAGE ENCODED STREAM
NON-BASE LAYER IMAGE ENCODED STREAM
PARAMETERS RELATING TO CLIPPING PROCESS
632
DECODING UNIT
633
DECODING UNIT
BASE LAYER IMAGE
NON-BASE LAYER IMAGE
630

FIG. 35

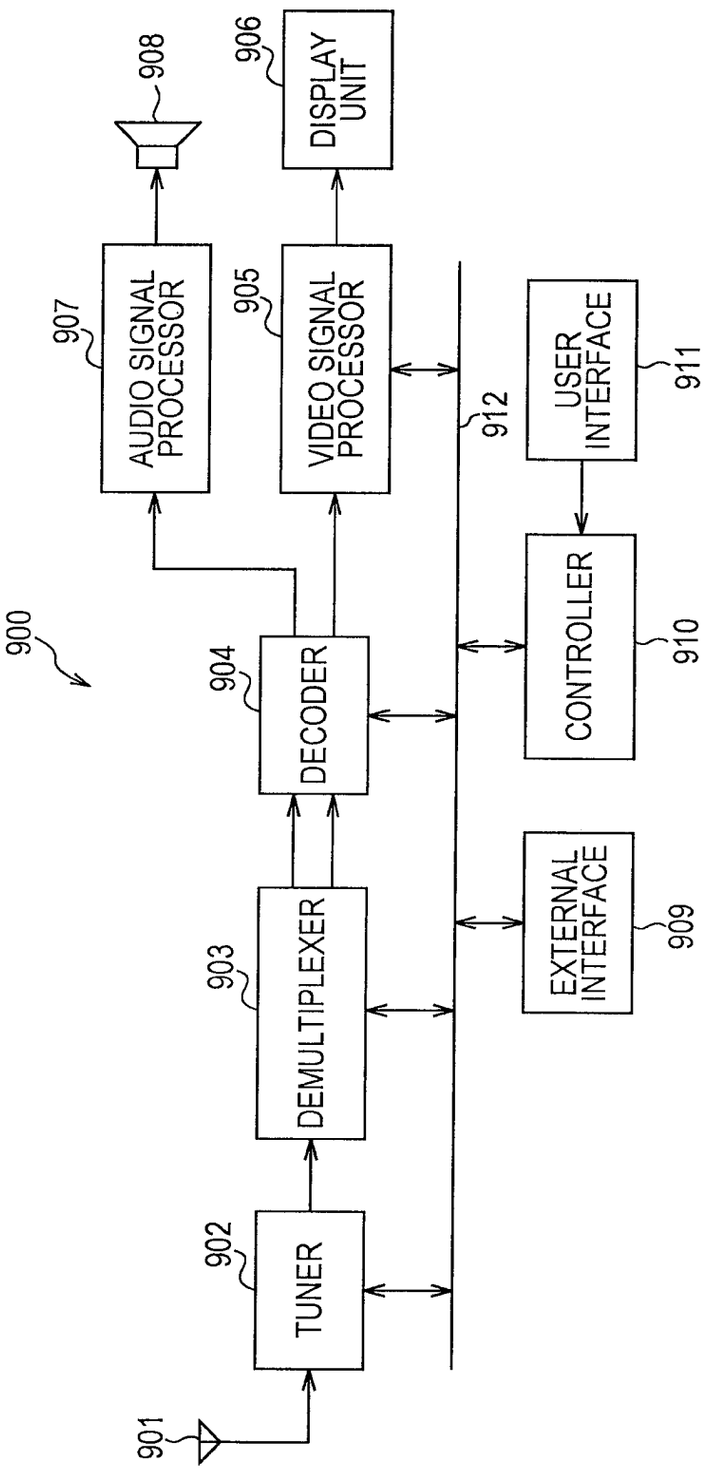

900
901
902
TUNER
903
DEMULTIPLEXER
904
DECODER
905
VIDEO SIGNAL PROCESSOR
906
DISPLAY UNIT
907
AUDIO SIGNAL PROCESSOR
908
909
EXTERNAL INTERFACE
910
CONTROLLER
911
USER INTERFACE
912

FIG. 36

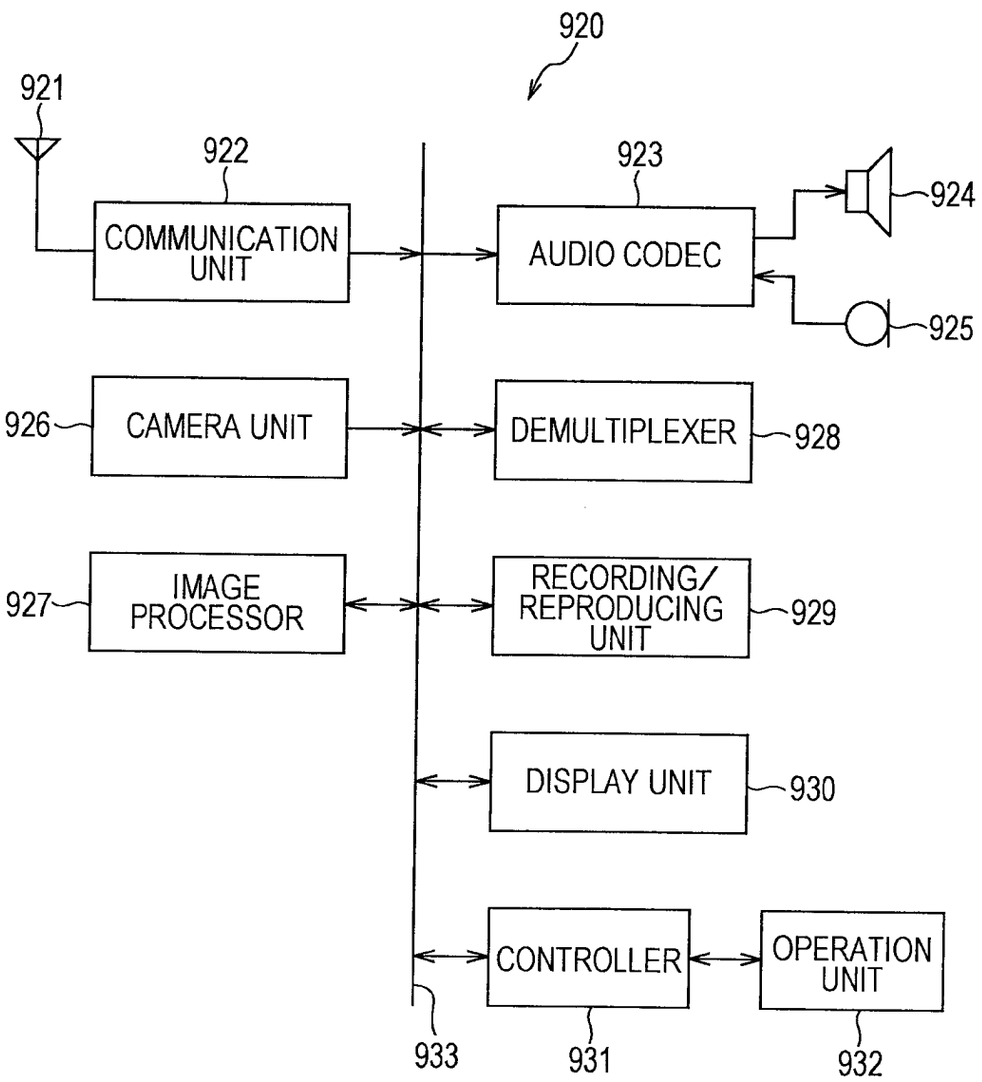

920
921
922
923
924
925
COMMUNICATION UNIT
AUDIO CODEC
926
CAMERA UNIT
DEMULTIPLEXER
928
927
IMAGE PROCESSOR
RECORDING/ REPRODUCING UNIT
929
DISPLAY UNIT
930
CONTROLLER
OPERATION UNIT
933
931
932

FIG. 37

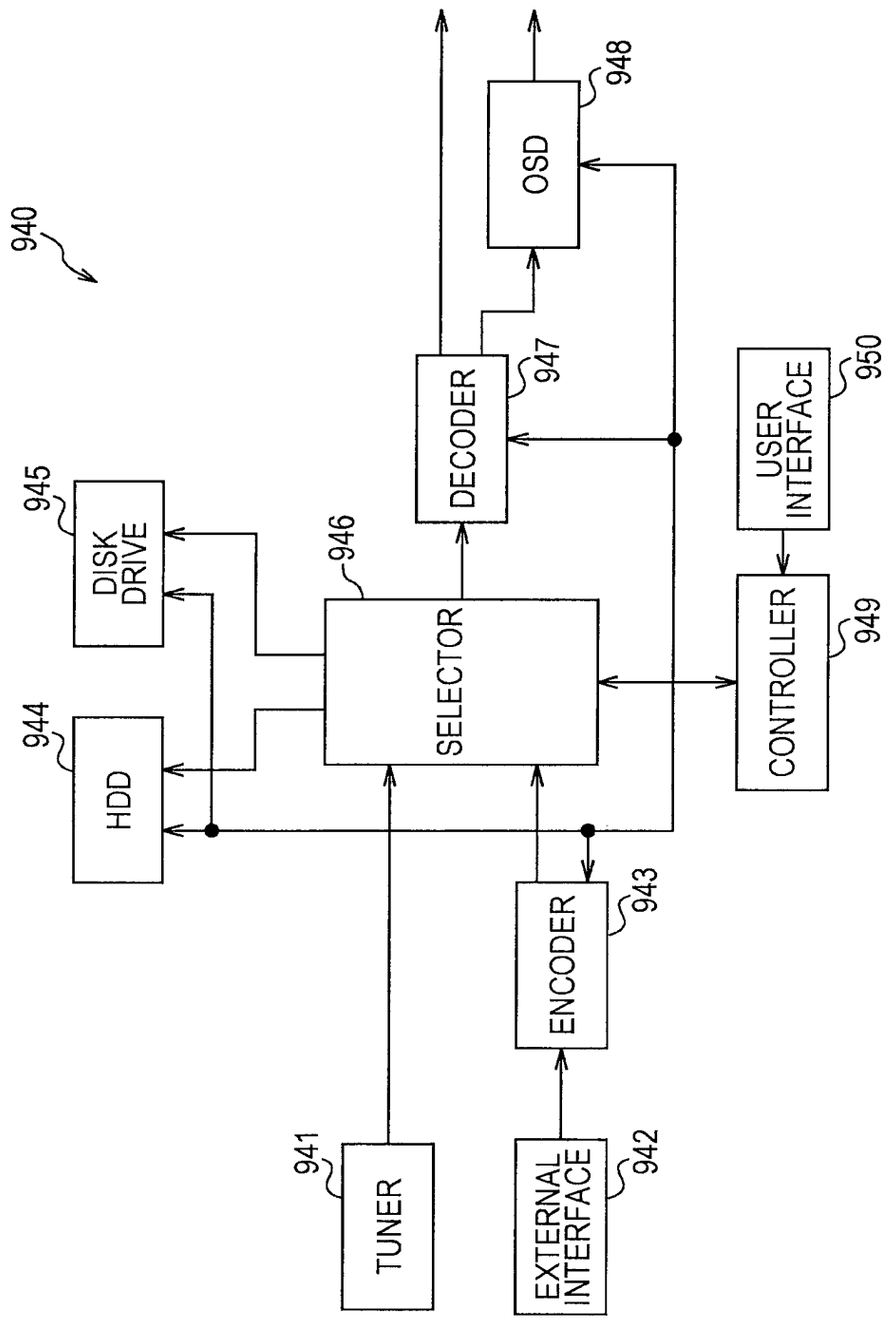
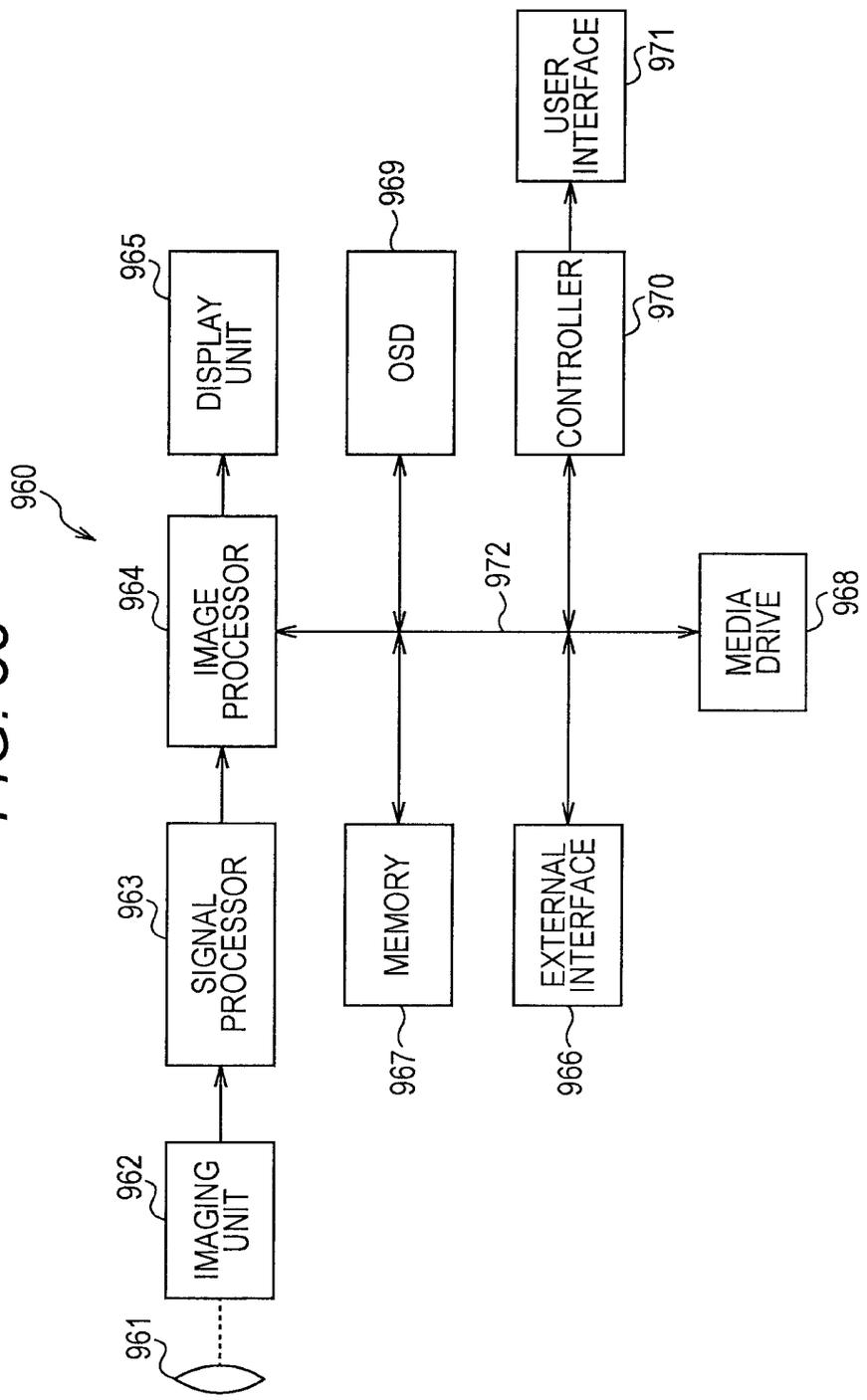

940
941
TUNER
942
EXTERNAL INTERFACE
943
ENCODER
944
HDD
945
DISK DRIVE
946
SELECTOR
947
DECODER
948
OSD
949
CONTROLLER
950
USER INTERFACE

FIG. 38


960
961
962
IMAGING UNIT
963
SIGNAL PROCESSOR
964
IMAGE PROCESSOR
965
DISPLAY UNIT
966
EXTERNAL INTERFACE
967
MEMORY
968
MEDIA DRIVE
969
OSD
970
CONTROLLER
971
USER INTERFACE
972

DESCRIPCIÓN

DISPOSITIVO DE PROCESAMIENTO DE IMAGEN Y MÉTODO DE PROCESAMIENTO DE IMAGEN

CAMPO TECNICO

[0001]

Esta técnica se refiere a un dispositivo de procesamiento de imagen BKu método de procesamiento de imagen. Específicamente, esta técnica permite un proceso de filtración de desbloqueo para aplicar apropiadamente la filtración.

ANTECEDENTES DEL ARTE

[0002]

En los últimos años, se han generado los dispositivos, que manejan información de imagen como datos digitales, que, en tal caso ayudan a transmitir y almacenar información con una alta eficiencia, y que se ajustan a un esquema, tal como MPEG2 (International Organization for Standardization and International Electrotechnical Cornmission (ISO/IEC) 13818-2), para la compresión de información de imagen usando transformación ortogonal, tal como la transformación discreta del coseno, y usando compensación de movimiento mediante la utilización de la redundancia que es única para la información de imagen tanto en la distribución de información en las estaciones de radiodifusión e información de recepción en los hogares ordinarios. Además, los esquemas llamados H.264 y MPEG4 Part10 (Advanced Video Coding (AVC) que requieren una mayor cantidad de operaciones para la codificación y decodificación, pero puede realizar una eficacia de codificación más alta que MPEG2 o los similares que también se han utilizado. Además, hoy en día, la estandarización trabaja para la codificación de vídeo de alta eficiencia (HEVC) que es un esquema de codificación de imagen de próxima generación están en curso de tal manera que se pueden realizar eficientemente la compresión, distribución, o similares de imagen de alta resolución de 4000 x 2000 píxeles que es cuatro veces la de las imagen de alta visión.

[0003]

En la estandarización que funciona para una alta eficiencia de codificación de vídeo (HEVC), que es un esquema de codificación de imagen de próxima generación, JCTVC – A119 (ver Documento de No Patente 1 más adelante) se propone aplicar un filtro de desbloqueo a cada bloque que tiene el tamaño de 8 x 8 píxeles o mayor. En el método propuesto en JCTVC - A119, mediante el aumento de una unidad de tamaño mínimo de bloque al que se va a aplicar

un filtro de desbloqueo, es posible ejecutar un proceso de filtración en paralelo en una pluralidad de límites de bloque en la misma dirección dentro de un macrobloque.

LISTA DE CITAS

DOCUMENTO DE NO PATENTE

[0004]

Documento de No Patente 1: K. Ugur (Nokia), K. R. Andersson (LM Ericsson), A. Fuldseth (Tandberg Telecom), “JCTVC - A119: vídeo de codificación propuesta por la tecnología de Tandberg, Nokia y Ericsson”, Documentos de la primera reunión del Equipo de Colaboración Conjunta sobre Video Coding (JCT- VC), Dresden, Alemania, Abril 15-23 2010

RESUMEN DE LA INVENCIÓN

PROBLEMAS A SER RESUELTOS POR LA INVENCIÓN

[0005]

Sin embargo, en el proceso de filtración de desbloqueo convencional, la filtración no se realiza apropiadamente.

[0006]

Por lo tanto, un objeto de esta técnica es permitir un proceso de filtración de desbloqueo para aplicar la filtración apropiadamente.

SOLUCIONES A LOS PROBLEMAS

[0007]

Un dispositivo de procesamiento de imagen de acuerdo con un primer aspecto de esta técnica incluye: una unidad de decodificación que decodifica una corriente codificada, codificada en unidades que tiene cada una, una estructura de capa para generar una imagen; una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite del bloque que es un límite entre un bloque de la imagen generada por la unidad de decodificación BKu bloque adyacente al bloque, y un controlador que controla la unidad de filtración de manera que, cuando se aplica una filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de

desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

[0008]

El controlador puede aplicar el proceso de recorte para los valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

[0009]

El controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones:

[Fórmula Matemática 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0,7 y PV son un valor de recorte.

[0010]

El controlador puede establecer un valor que es un múltiplo entero de parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se realiza el proceso de recorte.

[0011]

El controlador puede establecer un valor que es el doble de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se realiza el proceso de recorte.

[0012]

El dispositivo de procesamiento de imagen incluye además: una unidad que determina la fuerza del filtro que determina la fuerza del filtro de desbloqueo aplicada al límite de bloque, y la unidad de filtración puede aplicar el filtro de desbloqueo al límite de bloque de acuerdo con la

fuerza determinada por la unidad que determina la fuerza del filtro, y el controlador puede controlar la unidad de filtración de manera que, cuando la unidad que determina la fuerza del filtro determina qué fuerza de filtración se va a aplicar, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por el unidad de decodificación.

[0013]

La unidad que determina la fuerza del filtro puede determinar la fuerza del filtro de desbloqueo usando una pluralidad de líneas como unidades de procesamiento.

[0014]

La unidad que determina la fuerza del filtro puede determinar la fuerza del filtro de desbloqueo usando cuatro líneas como las unidades de procesamiento.

[0015]

El dispositivo de procesamiento de imagen incluye además: una unidad que determina la necesidad de filtración que determina si el filtro de desbloqueo se va a aplicar al límite del bloque usando una pluralidad de líneas como unidades de procesamiento, y la unidad que determina la fuerza del filtro puede determinar la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que el filtro de desbloqueo sea aplicado.

[0016]

Un método de procesamiento de imagen de acuerdo con el primer aspecto de esta técnica permite que un dispositivo de procesamiento de imagen ejecute: la decodificación de una corriente codificada en unidades, teniendo cada uno una estructura de capa para generar una imagen; la aplicación de un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque, que es un límite entre un bloque de la imagen generada y un bloque adyacente al lado del bloque, y que controla de manera que, cuando se aplique una filtración fuerte como la fuerza del filtro de desbloqueo, se aplique un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada.

[0017]

Un dispositivo de procesamiento de imagen de acuerdo con un segundo aspecto de esta técnica incluye: una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo a una fuerza del filtro de desbloqueo aplicada al límite de bloque, que es un límite entre un bloque de una imagen decodificada localmente cuando se codifica una imagen BKu

bloque adyacente al bloque; un controlador que controla la unidad de filtración de manera que, cuando se aplique una filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente; y una unidad de codificación que codifica la imagen en unidades que tienen cada una, una estructura de capa usando una imagen a la que se aplica el filtro de desbloqueo.

[0018]

El controlador puede aplicar el proceso de recorte a los valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

[0019]

El controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones:

[Fórmula Matemática 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0, 7 y PV son un valor de recorte.

[0020]

El controlador puede establecer un valor que es un múltiplo entero de parámetros del filtro de desbloqueo como el valor de recorte utilizado cuando se realiza el proceso de recorte.

[0021]

El controlador puede establecer un valor que es el doble de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se realiza el proceso de recorte.

[0022]

El dispositivo de procesamiento de imagen incluye además: una unidad que determina la fuerza del filtro que determina la fuerza del filtro de desbloqueo aplicada al límite de bloque, y la

unidad de filtración puede aplicar el filtro de desbloqueo al límite de bloque de acuerdo con la fuerza determinada por la unidad que determina la fuerza del filtro, y el controlador puede controlar la unidad de filtración de manera que, cuando la unidad que determina la fuerza del filtro determina que se aplica una filtración fuerte, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

[0023]

La unidad que determina la fuerza del filtro puede determinar la fuerza del filtro de desbloqueo usando una pluralidad de líneas como unidades de procesamiento.

[0024]

La unidad que determina la fuerza del filtro puede determinar la fuerza del filtro de desbloqueo usando cuatro líneas como las unidades de procesamiento.

[0025]

El dispositivo de procesamiento de imagen incluye además: una unidad que determina la necesidad de filtración que determina si el filtro de desbloqueo se va a aplicar al límite de bloque usando una pluralidad de líneas como unidades de procesamiento, y la unidad que determina la fuerza del filtro puede determinar la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que se aplique el filtro de desbloqueo.

[0026]

Un método de procesamiento de imagen de acuerdo con el segundo aspecto de esta técnica permite que un dispositivo de procesamiento de imagen ejecute: la aplicación de un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque, que es un límite entre un bloque de una imagen decodificada localmente cuando se codifica una imagen BKu bloque adyacente al bloque; controlar de manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente, y que codifica la imagen en unidades que tienen cada uno una estructura de capa con una imagen a la que se aplica el filtro de desbloqueo.

[0027]

En un primer aspecto de la presente técnica, una corriente codificada en unidades que tienen cada una, una estructura de capa se decodificada para generar una imagen, y se aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo

aplicada al límite de bloque que es un límite entre un bloque de la imagen generada y un bloque adyacente, adyacente al bloque. Por otra parte, se realiza el control de manera que, cuando se aplica una filtración fuerte como la fuerza de la filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada.

[0028]

En un segundo aspecto de la presente técnica, se aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque, que es un límite entre un bloque de una imagen decodificada localmente cuando se codifica una imagen BKu bloque adyacente, adyacente al bloque. Por otra parte, se realiza un control de manera que, cuando se aplica una filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente, y la imagen se codifica en unidades que tienen cada una, una estructura de capa usando una imagen a la que se aplica el filtro de desbloqueo.

EFECTOS DE LA INVENCIÓN

[0029]

De acuerdo con esta técnica, es posible permitir un proceso de filtración de desbloqueo para aplicar la filtración de forma adecuada.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0030]

Las Figuras 1(A) y 1(B) son diagramas para describir un proceso de filtración de desbloqueo convencional.

La figura 2 es un diagrama que ilustra una configuración cuando se aplica a un dispositivo de codificación de imagen.

La figura 3 es un diagrama de flujo que ilustra una operación de codificación de imagen.

La figura 4 es un diagrama de flujo que ilustra un proceso de predicción-intra.

La figura 5 es un diagrama de flujo que ilustra un proceso de predicción-inter.

La figura 6 es un diagrama que ilustra una configuración cuando se aplica a un dispositivo de decodificación de imagen.

La figura 7 es un diagrama de flujo que ilustra una operación de decodificación imagen.

La figura 8 es un diagrama para describir una operación básica de una unidad de filtración de desbloqueo.

La figura 9 es un diagrama que ilustra la configuración de una primera implementación de la unidad de filtración de desbloqueo.

La figura 10 es un diagrama de flujo que ilustra la operación de la primera implementación de la unidad de filtración de desbloqueo.

La figura 11 es un diagrama que ilustra la configuración de una unidad de operación de filtro.

La figura 12 es un diagrama que ilustra la configuración de la unidad de operación de filtro.

Las Figuras 13(A) y 13(B) son diagramas para describir la operación de la unidad de operación de filtro.

Las Figuras 14(A) y 14(B) son diagramas para describir la operación de la unidad de operación del filtro.

La figura 15 es un diagrama de flujo que ilustra la operación de una quinta implementación.

La figura 16 es un diagrama para describir un proceso de filtración de desbloqueo convencional.

La figura 17 es un diagrama para describir la presente técnica (sexta implementación).

La figura 18 es un diagrama para describir la presente técnica (sexta implementación).

La figura 19 es un diagrama que ilustra los píxeles usados en el proceso de filtración de desbloqueo de la presente técnica.

La figura 20 es un diagrama para describir valores de los píxeles que cambian debido a la filtración fuerte convencional.

La figura 21 es un diagrama para describir los efectos de la filtración fuerte con base en el recorte.

La figura 22 es un diagrama que ilustra la configuración de una sexta implementación de la unidad de filtración de desbloqueo.

La figura 23 es un diagrama de flujo que ilustra la operación de la sexta implementación de la unidad de filtración de desbloqueo.

La figura 24 es un diagrama que ilustra la configuración de una séptima implementación de la unidad de filtración de desbloqueo.

La figura 25 es un diagrama que ilustra la configuración de una octava implementación de la unidad de filtración de desbloqueo.

La figura 26 es un diagrama de flujo que ilustra la operación de la octava implementación de la unidad de filtración de desbloqueo.

La figura 27 es un diagrama de flujo para describir un proceso de desbloqueo de la señal de luminancia.

La figura 28 es un diagrama para describir un ejemplo del caso R3W2.

La figura 29 es un diagrama que ilustra un ejemplo de un esquema de codificación de imagen en multi-vista.

La figura 30 es un diagrama que ilustra un ejemplo de los principales componentes de un dispositivo de codificación de imagen en multi-vista al que se aplica la presente técnica.

La figura 31 es un diagrama que ilustra un ejemplo de los componentes principales de un dispositivo de decodificación de imagen en multi-vista al que se aplica la presente técnica.

La figura 32 es un diagrama que ilustra un ejemplo de un esquema de codificación de imagen en capa.

La figura 33 es un diagrama que ilustra un ejemplo de los componentes principales de un dispositivo de codificación de imagen en capa al que se aplica la presente técnica.

La figura 34 es un diagrama que ilustra un ejemplo de los componentes principales de un dispositivo de decodificación de imagen en capa al que se aplica la presente técnica.

La figura 35 es un diagrama que ilustra un ejemplo de configuración esquemática de un aparato de televisión.

La figura 36 es un diagrama que ilustra un ejemplo de una configuración esquemática de un teléfono móvil.

La figura 37 es un diagrama que ilustra un ejemplo de una configuración esquemática de un dispositivo de grabación/reproducción.

La figura 38 es un diagrama que ilustra un ejemplo de una configuración esquemática de un dispositivo de formación de imagen.

MODO PARA LLEVAR A CABO LA INVENCIÓN

[0031]

En lo sucesivo, se describirán los modos de llevar a cabo la presente técnica. La descripción se dará en el siguiente orden:

1. Arte convencional
2. Configuración cuando se aplica al dispositivo de codificación de imagen.
3. Operación del dispositivo de codificación de imagen
4. Configuración cuando se aplica al dispositivo de decodificación de imagen
5. La operación del dispositivo de decodificación de imagen
6. Operación básico de la unidad de filtración de desbloqueo
7. Primera implementación de la unidad de filtración de desbloqueo
8. Segunda implementación de la unidad de filtración de desbloqueo
9. Tercera implementación de la unidad de filtración de desbloqueo
10. Cuarta implementación de la unidad de filtración de desbloqueo
11. Quinta implementación de la unidad de filtración de desbloqueo

12. Descripción de la sexta a la octava implementaciones.
13. Sexta implementación de la unidad de filtración de desbloqueo
14. Séptima implementación de la unidad de filtración de desbloqueo
15. Octava implementación de la unidad de filtración de desbloqueo
16. Novena implementación
17. Décima implementación (dispositivo de codificación y de decodificación de imagen en multi-vista)
18. Undécima implementación (dispositivo de codificación y de decodificación de la imagen en capa)
19. ejemplo de aplicación

[0032]

[1. Arte Convencional]

Un proceso de filtración de desbloqueo convencional se describirá con referencia a las Figuras 1(A) y 1(B).

[0033]

Como se ilustra en la figura 1(A), por ejemplo, cuando un proceso de filtración de desbloqueo se lleva a cabo en un orden de trama en respectivas unidades de codificación más grandes (LCUs), los datos de imagen correspondientes a un número predeterminado de líneas desde un límite interbloque BB, de la LCUu que es un bloque superior se almacena en una memoria de línea, y un proceso de filtración vertical se realiza utilizando los datos de imagen y datos de imagen de la LCU1 que es un bloque inferior obtenido después del almacenamiento. Por ejemplo, como se ilustra en la figura 1(B), cuando se realiza la operación de filtración utilizando datos de imagen correspondientes a cuatro líneas desde el límite BB a la vez que usando datos de imagen de los bloques superior e inferior, cada uno corresponden a tres líneas desde el límite interbloque BB como un rango de procesamiento del proceso de filtración vertical, los datos de imagen correspondientes a las cuatro líneas desde el límite BB, de la LCUu que es el bloque superior se almacenan en la memoria de línea. En los dibujos, los píxeles objetivo del filtro de desbloqueo se representan por círculos dobles, y los límites superior e inferior del rango de procesamiento de filtro del filtro de desbloqueo se indican por “DBU” y “DBL”, respectivamente.

[0034]

De esta manera, ya que los datos de imagen correspondientes a un número predeterminado de líneas desde el límite del interbloque BB se almacenan en la memoria de línea con el fin de ser utilizado para la operación del filtro, si el número de píxeles en la dirección horizontal aumenta, la capacidad de memoria de la memoria de línea aumenta.

[0035]

[2. Configuración cuando se aplica al dispositivo de codificación de imagen]

La figura 2 ilustra una configuración cuando el dispositivo de procesamiento de imagen de la presente técnica se aplica a un dispositivo de codificación de imagen. Un dispositivo de codificación de imagen 10 incluye un convertidor analógico/digital (convertidor A/D) 11, un buffer de reordenamiento de cuadro 12, un sustractor 13, un transformador ortogonal 14, un cuantizador 15, un cuantizador sin pérdidas 16, un buffer de acumulación 17 y un controlador de tasa 18. El dispositivo de codificación de imagen 10 incluye además un cuantizador inverso 21, un transformador ortogonal inverso 22, un sumador 23, una unidad de filtración de desbloqueo 24, una memoria de cuadro 25, un selector 26, un predictor intra 31, un estimador de movimiento/compensador 32, y una unidad de selección de imagen de predicción/modo optimo 33.

[0036]

El convertidor A/D 11 convierte una señal de imagen analógica en datos de imagen digitales y envía los datos digitales de la imagen al buffer de reordenamiento de cuadro 12.

[0037]

El buffer de reordenamiento de cuadro 12 reordena los cuadros de la salida de datos de imagen desde el convertidor A/D 11. El Buffer de reordenamiento de cuadro 12 lleva a cabo la reordenación del cuadro de acuerdo con un GOP (grupo de imágenes) estructura relacionada con un proceso de codificación y envía los datos de imagen reordenados al sustractor 13, el predictor intra 31, y el estimador de movimiento/compensador 32.

[0038]

El sustractor 13 se suministra con la salida de datos de imagen desde el buffer de reordenamiento de cuadro 12 y predice datos de imagen seleccionados por la unidad de selección de imagen de predicción/modo óptimo 33 descrito posteriormente. El sustractor 13 calcula los datos de error de predicción, que es una diferencia entre la salida de datos de imagen desde el buffer de reordenamiento de cuadro 12 y los datos de imagen de predicción suministrados desde la unidad de selección de imagen de predicción/modo optimo 33 y envía a los datos de error de predicción al transformador ortogonal 14.

[0039]

El transformador ortogonal 14 realiza un proceso de transformación ortogonal tal como una transformada discreta del coseno (DCT) o transformada de Karhunen - Loeve con respecto a la salida de datos de error de predicción desde el sustractor 13. El transformador ortogonal 14 envía los datos del coeficiente de transformación obtenidos mediante la realización del proceso de transformación ortogonal al cuantizador 15.

[0040]

El cuantizador 15 se suministra con la salida de datos del coeficiente de transformación desde el transformador ortogonal 14 y una señal de control de tasa desde el controlador de tasa 18 se describe más adelante. El cuantizador 15 cuantiza los datos del coeficiente de transformación y envía los datos cuantizados al codificador sin pérdidas 16 y el cuantizador inverso 21. Por otra parte, el cuantizador conmuta un parámetro de cuantización (escala de cuantización) con base en la señal de control de tasa desde el controlador de tasa 18 para cambiar una tasa de bits de los datos cuantizados.

[0041]

El codificador sin pérdidas 16 se suministra con la salida de datos cuantizados desde el cuantizador 15 y la información de modo de predicción desde el predictor intra 31, el estimador de movimiento/compensador 32, y la unidad de selección de imagen de predicción/modo optimo se describe más adelante. La información del modo de predicción incluye un macrobloque tipo que permite un tamaño de bloque de predicción a ser identificado, un modo de predicción, información del vector de movimiento, la información de la imagen de referencia, y similares de acuerdo con la predicción intra o la predicción inter. El codificador sin pérdidas 16 realiza un proceso de codificación sin pérdidas tal como, por ejemplo, de codificación de longitud variable o de codificación aritmética sobre los datos cuantizados para generar una corriente codificada y emitir la corriente codificada al buffer de acumulación 17. Por otra parte, el codificador sin pérdidas 16 codifica sin pérdida de la información de modo de predicción para añadir la información de modo de predicción codificada a la información de encabezamiento de la corriente codificada.

[0042]

El buffer acumulación 17 acumula las corrientes codificadas del codificador sin pérdidas 16. Por otra parte, el buffer de acumulación 17 las corrientes codificadas a una tasa de transferencia que corresponde a una línea de transmisión.

[0043]

El controlador de tasa 18 monitorea un espacio libre del buffer de acumulación 17, genera una señal de control de tasa de acuerdo con el espacio libre, y envía la señal de control

de tasa al cuantizador 15. El controlador de tasa 18 adquiere información que indica el espacio libre del buffer de acumulación 17, por ejemplo. El controlador de tasa 18 disminuye la tasa de bit de los datos cuantizados utilizando la señal de control de tasa, cuando el espacio libre se reduce.

Por otra parte, el controlador de tasa 18 aumenta la tasa de bit de los datos cuantizados utilizando la señal de control de tasa cuando el espacio libre del buffer de acumulación 17 es suficientemente grande.

[0044]

El cuantizador inverso 21 realiza un proceso de cuantización inverso sobre los datos cuantizados suministrados desde el cuantizador 15. El cuantizador inverso 21 envía los datos de coeficiente de transformación obtenidos por la realización del proceso de cuantización inversa al transformador ortogonal inverso 22.

[0045]

El transformador ortogonal inverso 22 envía los datos obtenidos mediante la realización de un proceso de transformar ortogonal en los datos de coeficiente de transformada suministrados desde el cuantizador inverso 21 al sumador 23.

[0046]

El sumador 23 agrega los datos suministrados desde el transformador ortogonal inverso 22 y los datos de imagen predicha suministrados desde la unidad de selección de imagen de predicción/modo óptimo 33 para generar datos de imagen decodificados y envía los datos de imagen decodificados a la unidad de filtración de desbloqueo 24 y la memoria de cuadro 25.

[0047]

La unidad de filtración de desbloqueo 24 realiza un proceso de filtración para reducir la distorsión de bloques que ocurre durante la codificación de imagen. La unidad de filtración de desbloqueo 24 realiza un proceso de filtración para eliminar la distorsión de bloques de los datos de imagen decodificados suministrados desde el sumador 23 y envía los datos de imagen que han sido sometidos al proceso de filtración a la memoria de cuadro 25. Por otra parte, la unidad de filtración de desbloqueo 24 controla un rango de imagen usada para una operación de filtro en un bloque situado en el lado superior de un límite de acuerdo con un límite detectado mediante la realización de detección de límite inter-bloque en una dirección vertical. De esta manera, mediante el control del rango de imagen usado para la operación de filtro, la unidad de filtración de desbloqueo 24 permite que se realice un proceso de filtración vertical, incluso

cuando se reduce la capacidad de la memoria de la memoria línea que almacena los datos de imagen. Los detalles de la misma se describirán más adelante.

[0048]

La memoria de cuadro 25 mantiene los datos de imagen decodificados suministrados desde el sumador 23 y los datos de imagen decodificados que han sido sometidos al proceso de filtración, suministrados desde la unidad de filtración de desbloqueo 24 como datos de imagen de una imagen de referencia.

[0049]

El selector 26 suministra los datos de imagen de referencia antes del proceso de filtración, lee desde la memoria de cuadro 25 para el predictor intra 31 con el fin de llevar a cabo la predicción intra. Por otra parte, el selector 26 suministra los datos de imagen de referencia después que han sido sometidos al proceso de filtración, lee desde la memoria de cuadro 25 al estimador de movimiento/compensador 32 con el fin de realizar la predicción inter.

[0050]

El predictor intra 31 realiza un proceso de predicción intra en todos los modos de predicción intra candidatos utilizando los datos de imagen de una salida de imagen objetivo de codificación del buffer de reordenamiento de cuadro 12 y los datos de imagen de referencia antes del proceso de filtración, lee desde la memoria de cuadro 25. Además, el predictor intra 31 calcula un valor de función de costo para cada modo de predicción intra y selecciona un modo de predicción intra en el que el valor de la función costo calculado es el más pequeño (es decir, un modo de predicción intra en el que se obtiene la mejor eficiencia de codificación) como un modo de predicción intra óptimo. El predictor intra 31 envía los datos de imagen de predicción generados en el modo óptimo de predicción intra, la información de modo de predicción en el modo de predicción intra óptimo, y un valor de función de costo en el modo de predicción intra óptimo a la unidad de selección de imagen de predicción/modo optimo 33.

Por otra parte, el predictor intra 31 envía la información de modo de predicción en el modo de predicción intra en el proceso de predicción intra en los respectivos modos de predicción intra al codificador sin pérdidas 16 con el fin de obtener una cantidad de código generada usada en el cálculo del valor de la función de costos como se describe más adelante.

[0051]

El estimador de movimiento/compensador 32 realiza un proceso de estimación de movimiento/compensación en todos los tamaños de predicción de bloque correspondientes a macrobloques. El estimador de movimiento/ compensador 32 detecta un vector de movimiento

utilizando los datos de la imagen de referencia que han sido sometidos al proceso de filtración, lee desde la memoria de cuadro 25 con respecto a cada una de las imágenes objetivo de codificación de los respectivos tamaños de bloques de predicción, leídos desde el buffer de reordenamiento de cuadro 12. Además, el estimador de movimiento/compensador 32 realiza un proceso de compensación de movimiento en la imagen decodificada con base en el vector de movimiento detectado para generar una imagen de predicción. Por otra parte, el estimador de movimiento/compensador 32 calcula un valor de función de costo para cada tamaño de bloque de predicción y selecciona un tamaño de bloque de predicción en el que el valor de la función costo calculado es el más pequeño (es decir, un tamaño de bloque de predicción en el que se obtiene la mejor eficiencia en la codificación) como un modo de predicción inter óptima. El estimador de movimiento/compensador 32 envía los datos de imagen de predicción generados en el modo de predicción inter óptimo, la información de modo de predicción en el modo de predicción inter óptimo, y el valor de la función de costo en el modo de predicción inter óptimo a la unidad de selección de imagen de predicción/modo optimo 33. Por otra parte, el estimador de movimiento/compensador 32 envía la información de modo de predicción del modo de predicción inter en el proceso de predicción inter en los respectivos tamaños de bloque de predicción al codificador sin pérdidas 16 con el fin de obtener una cantidad de código generada usada en el cálculo del valor de la función de costo. El estimador de movimiento/compensador 32 también realiza la predicción en un modo de macrobloque recortado y un modo directo como el modo de predicción inter.

[0052]

La unidad de selección de imagen de predicción/modo optimo 33 compara el valor de función de costo suministrado desde el predictor intra 31 y el valor de función de costo suministrado desde el estimador de movimiento/compensador 32 en unidades de macro-bloques respectivos y selecciona un modo en el que el valor de la función de costo es el más pequeño como un modo óptimo en el cual se obtiene mejor la eficiencia de codificación. Por otra parte, la unidad de selección de imagen de predicción/modo óptimo 33 envía los datos de imagen de predicción generados en el modo óptimo al sustractor 13 y el sumador 23. Además, la unidad de selección de imagen de predicción/modo óptimo 33 envía la información de modo de predicción del modo óptimo para el codificador sin pérdidas 16. La unidad de selección de imagen de predicción/modo óptimo 33 puede realizar la predicción intra o la predicción inter en unidades de segmento respectivos.

[0053]

Una unidad de codificación en las reivindicaciones incluye el predictor intra 31 que genera datos de imagen de predicción, el estimador de movimiento/compensador 32, la unidad

de selección de imagen de predicción/modo optimo 33, el sustractor 13, el transformador ortogonal 14, el cuantizador 15, el codificador sin pérdida 16, y similares.

[0054]

[3. Operación del dispositivo de codificación de imagen]

La figura 3 es un diagrama de flujo que ilustra la operación de un proceso de codificación de imagen. En la etapa ST11, el convertidor A/D 11 realiza la conversión A/D en una imagen de señal de entrada.

[0055]

En la etapa ST12, el buffer de reordenamiento de cuadro 12 realiza el reordenamiento de la pantalla. El buffer de reordenamiento de cuadro 12 almacena los datos de imagen suministrados desde el convertidor A/D 11 y reordena las respectivas imágenes de manera que las imágenes dispuestas en la pantalla se reordenen en el orden de codificación.

[0056]

En la etapa ST13, el sustractor 13 genera datos de error de predicción. El sustractor 13 calcula una diferencia entre los datos de imagen de las imágenes reordenadas en la etapa ST12 y los datos de imagen de predicción seleccionados por la unidad de selección de imagen de predicción/modo óptimo 33 a generar datos de error de predicción. El error de predicción de datos tiene una cantidad de datos que es más pequeña que el de los datos de imagen originales. Por lo tanto, es pasible comprimir la cantidad de datos como se compara cuando la imagen se codifica tal como es. Cuando la unidad de selección de imagen de predicción/modo optimo 33 selecciona la imagen de predicción suministrada desde el predictor intra 31 y la imagen predicha a partir del estimador de movimiento/compensador 32 en unidades de segmento respectivos, la predicción intra se realiza en el segmento en el que se selecciona la imagen de predicción suministrada desde el predictor intra 31. Por otra parte, la predicción inter se lleva a cabo en el sector en el que se selecciona la imagen de predicción desde el estimador de movimiento/compensador 32.

[0057]

En la etapa ST14, el transformador ortogonal 14 realiza un proceso de transformación ortogonal. El transformador ortogonal 14 realiza la transformación ortogonal sobre los datos de error de predicción suministrados desde el sustractor 13. Específicamente, la transformación ortogonal tal como la transformación discreta del coseno o transformación Karhunen - Loeve se

realiza sobre los datos de error de predicción, y se sacan los datos de coeficiente de transformación.

[0058]

En la etapa ST15, el cuantizador 15 realiza un proceso de cuantización. El cuantizador 15 cuantiza los datos de coeficiente de transformada. Cuando la cuantización es realizada, la tasa de control se lleva a cabo como se describe en el procedimiento de la etapa ST25.

[0059]

En la etapa ST16, el cuantizador inverso 21 lleva a cabo un proceso de cuantización inversa. El cuantizador inverso 21 realiza cuantización inversa sobre los datos de coeficiente de transformación mediante el cuantizador 15 de acuerdo a una propiedad correspondiente a la propiedad del cuantizador 15.

[0060]

En la etapa ST17, el transformador ortogonal inverso 22 realiza un proceso de transformación ortogonal. El transformador ortogonal inverso 22 realiza la transformación ortogonal inversa sobre los datos cuantizados inversos del coeficiente de transformación por el cuantizador inverso 21 de acuerdo con una propiedad correspondiente a la propiedad del transformador ortogonal 14.

[0061]

En la etapa ST18, el sumador 23 genera datos de imagen decodificados. El sumador 23 agrega los datos de imagen de predicción suministrados desde la unidad de selección de imagen de predicción/modo óptimo 33 y los datos después de la transformación ortogonal inversa en la posición correspondiente a la imagen de predicción para generar datos de imagen decodificados.

[0062]

En la etapa ST19, la unidad de filtración de desbloqueo 24 realiza un proceso de filtración de desbloqueo. La unidad de filtración de desbloqueo 24 realiza la filtración de la salida de datos de imagen decodificados por el sumador 23 para eliminar la distorsión de bloque. Por otra parte, la unidad de filtración de desbloqueo 24 está configurada para realizar un proceso de filtración vertical incluso cuando se reduce la capacidad de memoria de la memoria de línea que almacena los datos de imagen. Específicamente, la unidad de filtración de desbloqueo 24 controla el rango de imagen usado para una operación de filtro en un bloque situado en el lado superior de un límite de acuerdo con un límite detectado mediante la realización de una detección del límite interbloque en la dirección vertical.

[0063]

En la etapa ST20, la memoria de cuadro 25 almacena datos de imagen decodificados. La memoria de cuadro 25 almacena los datos de imagen decodificados antes del proceso de filtración de desbloqueo.

[0064]

En la etapa ST21, el predictor intra 31 y el estimador de movimiento/compensador 32 realizan los procesos de predicción. Es decir, el predictor intra 31 realiza un proceso de predicción intra en el modo de predicción intra y el estimador de movimiento/compensador 32 realiza un proceso de estimación de movimiento/compensación en el modo de predicción inter. Por este proceso, los procesos de predicción se llevan a cabo en todos los modos de predicción candidatos, y se calculan los valores de la función costo en todos los modos de predicción candidato. El modo de predicción intra óptimo y el modo de predicción inter óptimo se seleccionan con base en los valores de la función de costo calculados y la imagen de predicción se genera en el modo de predicción seleccionado y la función de costos y la información del modo de predicción se suministran a la unidad de selección de imagen de predicción/modo optimo 33.

[0065]

En la etapa ST22, la unidad de selección de imagen de predicción/modo óptimo 33 selecciona los datos de imagen de predicción. La unidad de selección de imagen de predicción/modo optimo 33 determina un modo óptimo en el que se obtiene la mejor eficiencia de codificación sobre la base de los valores de la función de costo respectivos sacados desde el predictor intra 31 y el estimador de movimiento/compensador 32. Además, la unidad de selección de imagen de predicción/modo óptimo 33 selecciona los datos de imagen de predicción en el modo óptimo determinado y suministra los datos de imagen de predicción al sustractor 13 y el sumador 23. La imagen de predicción se utiliza para la operación en las etapas ST13 y ST18, como se describe más arriba.

[0066]

En la etapa ST23, el codificador sin pérdidas 16 realiza un proceso de codificación sin pérdidas. El codificador sin pérdidas 16 realiza codificación sin pérdida en la salida de datos cuantizados desde el cuantizador 15. Es decir, la codificación sin pérdidas como la codificación de longitud variable o codificación aritmética se realiza sobre los datos cuantizados con lo que se comprimen los datos cuantizados. En este caso, en la etapa ST22 descrita anteriormente, la información de modo de predicción (por ejemplo, incluyendo el tipo de macrobloque, el modo de predicción, la información del vector de movimiento, la información de imagen de referencia, y

similares) entra al codificador sin pérdidas 16 también se somete a codificación sin pérdida. Además, los datos de codificación sin pérdidas de la información de modo de predicción se agregan a la información de encabezamiento de la corriente codificada generada mediante la codificación sin pérdidas de los datos cuantizados.

[0067]

En la etapa ST24, el buffer de acumulación 17 realiza un proceso de acumulación para acumular la corriente codificada. La corriente codificada acumulada en el buffer de acumulación 17 se lee de manera adecuada y se transmite a un lado de decodificación a través de una línea de transmisión.

[0068]

En la etapa ST25, el controlador de tasa 18 realiza el control de tasa. El controlador de tasa 18 controla la tasa de las operaciones de cuantización del cuantizador 15 para que no ocurra un desbordamiento o un subdesbordamiento en el buffer de acumulación 17 cuando el buffer de acumulación 17 acumule la corriente codificada.

[0069]

Más adelante, se describirá el proceso de predicción en la etapa ST21 de la figura 3. El predictor intra 31 realiza un proceso de predicción intra. El predictor intra 31 realiza predicción intra sobre la imagen de un bloque actual en todos los modos de predicción intra - candidatos. Los datos de imagen de una imagen de referencia, referencia en la predicción intra no se somete al proceso de filtración de la unidad de filtración de desbloqueo 24, pero se utilizan los datos de imagen de referencia almacenados en la memoria de cuadro 25. Aunque los detalles del proceso de predicción intra se describen más adelante, por este proceso, se realiza predicción intra en todos los modos de predicción intra candidatos, y se calculan los valores de la función de costo en todos los modos de predicción intra candidatos. Por otra parte, un modo de predicción intra en el que se obtiene la mejor eficiencia de la codificación se selecciona de entre todos los modos de predicción intra con base en los valores de la función de costos calculados. El estimador de movimiento/compensador 32 realiza un proceso de predicción inter en todos los modos de predicción inter candidatos (todos los tamaños de bloque de predicción) utilizando los datos de la imagen de referencia que han sido sometidos al proceso de filtración, almacenado en la memoria de cuadro 25. Aunque los detalles del proceso de predicción inter se describen más adelante, por este proceso, el proceso de predicción se realiza en todos los modos de predicción inter candidatos, y se calculan los valores de la función de costo en todos los modos de predicción inter candidatos. Por otra parte, un modo de predicción inter en el que se obtiene la mejor eficiencia de codificación se selecciona de entre todos los modos de predicción inter con base en los valores de la función de costo calculados.

[0070]

El proceso de predicción intra se describirá con referencia al diagrama de flujo de la Fig. 4. En la etapa ST31, el predictor intra 31 realiza predicción intra en los respectivos modos de predicción. El predictor intra 31 genera datos de imagen de predicción en cada modo de predicción intra utilizando los datos de imagen decodificados antes del proceso de filtración, almacenado en la memoria de cuadro 25.

[0071]

En la etapa ST32, el predictor intra 31 calcula los valores de la función de costo de los modos de predicción respectivos. Por ejemplo, el proceso de codificación sin pérdida se lleva a cabo tentativamente en todos los modos de predicción candidatos, se calcula el valor de la función de costo en todos los modos de predicción respectivos mediante la Expresión (1) siguiente.

[0072]

$$\text{Costo (Modo } \epsilon\ \Omega) = D + \lambda/R \qquad \ldots (1)$$

[0073]

Aquí, “Ω” representa un conjunto universal de modos candidato de predicción para los bloques de codificación y macrobloques. “D” representa diferencial de energía (distorsión) entre una imagen decodificada y una imagen de entrada cuando la codificación se lleva a cabo en un modo de predicción. “R” representa una cantidad de código generado incluyendo los coeficientes de transformación ortogonal, información de modo de predicción, y similares, y “λ” representa el multiplicador de Lagrange dado como una función de un parámetro de cuantización QP.

[0074]

Por otra parte, con respecto a todos los modos de predicción candidatos, las imágenes de predicción se generan y se calculan los bits de encabezamiento como información de vector de movimiento y la información de modo de predicción, y se calcula el valor de la función de costo expresada por la Expresión (2) siguiente en los respectivos modos de predicción.

[0075]

$$\text{Costo (Modo}\epsilon\Omega) = D + \text{QPtoQuant (QP)} \bullet \text{Header\_Bit} \qquad \ldots (2)$$

[0076]

Aquí, “Ω” representa un conjunto universal de modos candidatos de predicción para codificar bloques y macro-bloques. “D” representa la energía diferencial (distorsión) entre una imagen decodificada y una imagen de entrada cuando la codificación se realiza en un modo de predicción. “Header_Bit es un bit de encabezamiento de un modo de predicción”, y “QPtoQuant” es una función dada como una función de un parámetro de cuantización QP.

[0077]

En la etapa ST33, el predictor intra 31 determina un modo de predicción intra óptimo. El predictor intra 31 selecciona un modo de predicción intra en el que el valor de la función de costo calculado es el más pequeño con base en los valores de función de costo calculados en la etapa ST32 y determina el modo de predicción intra como el modo de predicción intra óptimo.

[0078]

Más adelante, se describirá el proceso de predicción inter con referencia al diagrama de flujo de la figura 5, en la etapa ST41, el estimador de movimiento/ compensador 32 determina un vector de movimiento y una imagen de referencia en los respectivos modos de predicción. Es decir, el estimador de movimiento/compensador 32 determina el vector de movimiento y la imagen de referencia de un bloque actual de los respectivos modos de predicción.

[0079]

En la etapa ST42, el estimador de movimiento/compensador 32 realiza la compensación de movimiento en los respectivos modos de predicción. El estimador de movimiento/compensador 32 realiza de compensación de movimiento sobre la imagen de referencia con base en el vector de movimiento determinado en la etapa ST41 en los respectivos modos de predicción (respectivos tamaños de los bloques de predicción) y genera datos de imagen de predicción en los respectivos modos de predicción.

[0080]

En la etapa ST43, el estimador de movimiento/compensador 32 genera información del vector de movimiento en los respectivos modos de predicción. El estimador de movimiento/compensador 32 genera información del vector de movimiento incluida en la corriente codificada con respecto a los vectores de movimiento determinados en los respectivos modos de predicción. Por ejemplo, un vector de movimiento de predicción se determina utilizando la predicción mediana y la información del vector de movimiento que indica una diferencia entre el vector de movimiento detectado por la estimación de movimiento y se general vector de movimiento de predicción. La información del vector de movimiento generada de esta manera también se utiliza en el cálculo del valor de la función de costo en la siguiente etapa ST44, y cuando la imagen de predicción correspondiente se selecciona finalmente mediante la

unidad de selección de imagen de predicción/modo optimo 33, la información del vector de movimiento se incluye en la información de modo de predicción y se envía al codificador sin pérdidas 16.

[0081]

En la etapa ST44, el estimador de movimiento/compensador 32 calcula el valor de la función de costo en los respectivos modos de predicción inter. El estimador de movimiento/compensador 32 calcula el valor de la función de costo utilizando la Expresión (1) o (2) descrita anteriormente.

[0082]

En la etapa ST45, el estimador de movimiento/compensador 32 determina un modo de predicción inter óptimo. El estimador de movimiento/compensador 32 selecciona un modo de predicción en el que el valor de la función de costo calculado es el más pequeño sobre la base de los valores de la función de costo calculados en la etapa ST44.

[0083]

[4. Configuración cuando se aplica al dispositivo de decodificación de imagen]

La corriente codificada generada mediante la codificación de una imagen de entrada se suministra a un dispositivo de decodificación de imagen a través de una línea de transmisión predeterminada, un medio de grabación, o similar, y se decodifica.

[0084]

La figura 6 ilustra la configuración de un dispositivo de decodificación de imagen. Un dispositivo de decodificación de imagen 50 incluye un buffer de acumulación 51, un decodificador sin pérdidas 52, un cuantizador inverso 53, un transformador ortogonal inverso 54, un sumador 55, una unidad de filtración de desbloqueo 56, un buffer de reordenamiento de cuadro 57, y un convertidor D/A 58. El dispositivo de decodificación de imagen 50 incluye además una memoria de cuadro 61, selectores 62 y 65, un predictor intra 63, y una unidad de compensación de movimiento 64.

[0085]

El buffer de acumulación 51 acumula corrientes codificadas transmitidas al mismo. El decodificador sin pérdidas 52 decodifica las corrientes codificadas suministrados desde el buffer de acumulación 51 de acuerdo con un esquema correspondiente al esquema de codificación del codificador sin pérdida 16 de la figura 2. Por otra parte, el decodificador sin pérdidas 52 envía la información del modo de predicción obtenida mediante la decodificación de la información de

encabezamiento de las corrientes codificadas al predictor intra 63 y a la unidad de compensación de movimiento 64.

[0086]

El cuantizador inverso 53 realiza la cuantización inversa sobre los datos cuantizados decodificados por el decodificador sin pérdidas 52 de acuerdo con un esquema que corresponde al esquema de cuantización del cuantizador 15 de la figura 2. El transformador ortogonal inverso 54 realiza la transformación ortogonal inversa sobre la salida del cuantizador inverso 53 de acuerdo con un esquema que corresponde al esquema de transformación ortogonal del transformador ortogonal 14 de la Fig. 2 y envía la salida al sumador 55.

[0087]

El sumador 55 agrega los datos después de la transformación ortogonal inversa y los datos de imagen de predicción suministrados desde el selector 65 para generar datos de la imagen decodificados y envía los datos de imagen decodificados a la unidad de filtración de desbloqueo 56 y la memoria de cuadro 61.

[0088]

La unidad de filtración de desbloqueo 56 realiza un proceso de filtración sobre los datos de imagen decodificados suministrados desde el sumador 55 de manera similar a la unidad de filtración de desbloqueo 24 de la Fig. 2 para eliminar la distorsión de bloques y envía los datos de imagen decodificados al buffer de reordenamiento de cuadro 57 y la memoria de cuadro 61.

[0089]

El buffer de reordenamiento de cuadro 57 realiza el reordenamiento de la pantalla. Es decir, los cuadros reordenados a partir del orden de codificación usado en el buffer de reordenamiento de cuadro 12 de la Fig. 2 se reordenan en el orden de presentación original y se envían al convertidor D/A 58.

[0090]

El convertidor D/A 58 realiza la conversión D/A sobre los datos de imagen suministrados desde el buffer de reordenamiento de cuadro 57 y envía los datos de imagen a una pantalla (no ilustrada) para mostrar por este modo la imagen.

[0091]

La memoria de cuadro 61 mantiene los datos de imagen decodificada antes del proceso de filtración, suministrados desde el sumador 55 y los datos de imagen decodificados que han

sido sometidos al proceso de filtración, suministrados desde la unidad de filtración de desbloqueo 56 como datos de imagen de una imagen de referencia.

[0092]

El selector 62 suministra los datos de imagen de referencia antes del proceso de filtración, lee desde la memoria de cuadro 61 para el predictor intra 63 cuando un bloque de predicción que ha sido sometido a predicción intra se decodifica con base en la información de modo de predicción suministrada desde el decodificador sin pérdidas 52. Por otra parte, el selector 26 proporciona los datos de la imagen de referencia que han sido sometidos al proceso de filtración, lee de la memoria de cuadro 61 para la unidad de compensación de movimiento 64, cuando un bloque de predicción que ha sido sometido a predicción inter se decodifica con base en la información de modo de predicción suministrada desde el decodificador sin pérdidas 52.

[0093]

El predictor intra 63 genera imágenes de predicción con la base en la información de modo de predicción suministrada desde el decodificador sin pérdidas 52 y envía los datos de imagen de predicción al selector 65.

[0094]

La unidad de compensación de movimiento 64 realiza la compensación de movimiento con base en la información de modo de predicción suministrada desde el decodificador sin pérdidas 52 para generar datos de imagen de predicción y envía los datos de imagen de predicción al selector 65. Es decir, la unidad de compensación de movimiento 64 realiza la compensación de movimiento utilizando un vector de movimiento con base en la información del vector de movimiento con respecto a la imagen de referencia indicada por la información del cuadro de referencia basada en la información del vector de movimiento y la información del cuadro de referencia incluido en la información del modo de predicción para generar datos de imagen de predicción.

[0095]

El selector 65 suministra los datos de imagen de predicción generados por el predictor intra 63 al sumador 55.

Por otra parte, el selector 65 suministra los datos de imagen de predicción generados por la unidad de compensación de movimiento 64 al sumador 55.

[0096]

Una unidad de decodificación en las reivindicaciones incluye el decodificador sin pérdidas 52, el cuantizador inverso 53, el transformador ortogonal inverso 54, el sumador 55, el predictor intra 63, la unidad de compensación de movimiento 64, y similares.

[0097]

[5. Operación del dispositivo de decodificación de imagen]

Más adelante, se describirá un dispositivo de procesamiento de imagen realizado por el dispositivo de decodificación de imagen 50 con referencia al diagrama de flujo de la figura 7.

[0098]

En la etapa ST51, el buffer de acumulación 51 acumula corrientes codificadas transferidas al mismo. En la etapa ST52, el decodificador sin pérdida 52 realiza un proceso de decodificación sin pérdida. El decodificador sin pérdidas 52 decodifica las corrientes codificadas suministradas desde el buffer de acumulación 51. Es decir, se obtienen los datos cuantizados de las imágenes respectivas codificadas por el codificador sin pérdidas 16 de la figura 2. Por otra parte, el decodificador sin pérdidas 52 realiza la decodificación sin pérdida en la información de modo de predicción incluida en la información de encabezamiento de la corriente codificada para obtener Información del modo de predicción y suministra la información de modo de predicción a la unidad de filtración de desbloqueo 56 y a los selectores 62 y 65. Además, el decodificador sin pérdidas 52 envía la información de modo de predicción al predictor intra 63 cuando la información de modo de predicción es información sobre el modo de predicción intra. Además, el decodificador sin pérdidas 52 envía a la información de modo de predicción a la unidad de compensación de movimiento 64 cuando la información de modo de predicción es información sobre el modo de predicción inter.

[0099]

En la etapa ST53, el cuantizador inverso 53 realiza un proceso de cuantización inversa. El cuantizador inverso 53 realiza la cuantización inversa de los datos cuantizados decodificados por el decodificador sin pérdidas 52 de acuerdo con una propiedad que corresponde a la propiedad del cuantizador 15 de la Fig. 2.

[0100]

En la etapa ST54, el transformador ortogonal inverso 54 lleva a cabo un proceso de transformación ortogonal. El transformador ortogonal inverso 54 lleva a cabo la transformación ortogonal inversa sobre los datos del coeficiente de transformación de cuantización inversa por el cuantizador inverso 53 de acuerdo a una propiedad correspondiente a la propiedad de transformación ortogonal 14 de la figura 2.

[0101]

En la etapa ST55, el sumador 55 genera datos de imagen decodificados. El sumador 55 agrega los datos obtenidos mediante la realización del proceso de transformación ortogonal y los datos de predicción de imagen seleccionados en la etapa ST59 descrita más adelante para generar los datos de imagen decodificados. De esta manera, se decodifica la imagen original.

[0102]

En la etapa ST56, la unidad de filtración de desbloqueo 56 realiza un proceso de filtración de desbloqueo. La unidad de filtración de desbloqueo 56 realiza un proceso de filtración sobre la salida de datos de imagen decodificados por el sumador 55 para eliminar la distorsión de bloques incluida en la imagen decodificada.

[0103]

En la etapa ST57, la memoria de cuadro 61 almacena los datos de imagen decodificados.

[0104]

En la etapa ST58, el predictor intra 63 y la unidad de compensación de movimiento 64 realizan un proceso de predicción. El predictor intra 63 y la unidad de compensación de movimiento 64 realizan el proceso de predicción sobre la información del modo de predicción suministrada desde el decodificador sin pérdidas 52.

[0105]

Es decir, cuando la información del modo de predicción de la predicción intra se suministra desde el decodificador sin pérdidas 52, el predictor intra 63 realiza un proceso de predicción intra con base en la información del modo de predicción para generar los datos de imagen de predicción. Por otra parte, cuando la información del modo de predicción de la predicción inter se suministra desde el decodificador sin pérdidas 52, la unidad de compensación de movimiento 64 realiza la compensación de movimiento con base en la información del modo de predicción para generar los datos de imagen de predicción.

[0106]

En la etapa ST59, el selector 65 selecciona los datos de la imagen de predicción. Es decir, el selector 65 selecciona la imagen de predicción suministrada desde el predictor intra 63 y los datos de imagen de predicción generados por la unidad de compensación de movimiento 64 y suministra la imagen de predicción y los datos de imagen de predicción al sumador 55 de

manera que sean añadidos a la salida del transformador ortogonal inverso 54 en la etapa ST55 como se describió anteriormente.

[0107]

En la etapa ST60, el buffer de reordenamiento de cuadro 57 realiza el reordenamiento de la pantalla. Es decir, el buffer de reordenamiento de cuadro 57 reordena los cuadros de manera que los cuadros reordenados en el orden de codificación utilizados en el buffer de reordenamiento 12 del dispositivo de codificación de imagen 10 de la figura 2 se reordenan en el orden de visualización original.

[0108]

En la etapa ST61, el convertidor D/A 58 realiza la conversión D/A sobre los datos de imagen del buffer de reordenamiento de cuadro 57. Esta imagen se envía a una pantalla (no ilustrada) y se muestra la imagen.

[0109]

[6. Operación básico de la unidad de filtración de desbloqueo]

En general, un proceso de filtración de desbloqueo de acuerdo con un esquema de codificación de imagen como H.264/AVC o HEVC, involucra la determinación de si es o no necesaria la filtración y realiza un proceso de filtración con respecto a un límite de inter-bloque en el que se determina que la filtración es necesaria.

[0110]

El límite inter-bloque incluye límites (es decir, los límites inter-bloques adyacentes horizontalmente (denominado en lo sucesivo “límites verticales”) detectados mediante la realización de la detección del límite inter-bloque en la dirección horizontal. Por otra parte, el límite inter-bloque incluye límites inter-bloques (es decir, límites inter-bloques adyacentes verticalmente (en adelante denominados “límites de línea”) detectados mediante la realización de la detección del límite inter-bloque en la dirección vertical.

[0111]

La figura 8 es un diagrama explicativo que ilustra un ejemplo de píxeles en dos bloques BKa y BKb adyacentes con un límite interpuesto. En este ejemplo, aunque un límite vertical se utiliza como un ejemplo, el mismo se puede aplicar a un límite de línea. En el ejemplo de la figura 8, los datos de imagen de píxeles en el bloque BKa están representados por el símbolo “Pi, j”. En este caso, “i” es un índice de la columna de píxeles y “j” es un índice de la fila de píxel. Por otra parte, una unidad más pequeña del proceso de codificación es un bloque de 8 x 8 píxeles, y se asignan secuencialmente índices de columna “i” de 0, 1, 2 y 3 (de izquierda a

derecha) desde la columna cerca al límite vertical. Los índices de fila “j” de 0, 1, 2,..., y 7 se asignan de arriba hacia abajo. La mitad izquierda del bloque BKa se omite en el dibujo. Por otro lado, los datos de imagen de píxeles en el bloque BKb se representan por el símbolo “qk", "j”. Aquí, “k” es un índice de columna de pixel y “j” es un índice de la fila de píxel. Los índices de columna “k” de 0, 1, 2 y 3 se asignan secuencialmente (de izquierda a derecha) a partir de la columna cerca del límite vertical. La mitad derecha del bloque BKb también se omite en el dibujo.

[0112]

Cuando se determina que debe ser aplicado un filtro de desbloqueo a un límite y el límite es un límite vertical, por ejemplo, el proceso de filtración se realiza sobre los píxeles en los lados izquierdo y derecho del límite. En cuanto a los componentes de luminancia, se conmutan un filtro que tiene una fuerza de filtro fuerte y un filtro que tiene una fuerza de filtro débil de acuerdo con el valor de los datos de imagen.

[0113]

[Filtración del componente de Luminancia]

En la selección de fuerza, si las condiciones de las expresiones (3) a (5) se satisfacen, se determina para cada línea (o para cada columna). Un filtro fuerte se selecciona cuando se satisfacen todas las condiciones de las Expresiones (3) a (5), y un filtro débil se selecciona cuando las condiciones de cualquiera de las expresiones no se satisfacen.

[0114]

[Fórmula Matemática 1]

$$d < (\beta >> 2) \quad \cdots (3)$$

$$|p_3 - p_0| + |q_0 - q_3| < (\beta >> 3) \quad \cdots (4)$$

$$|p_0 - q_0| < ((5 * t_C + 1) >> 1) \quad \cdots (5)$$

[0115]

En la Expresión (3), “d” es un valor calculado con base en la Expresión (6). Por otra parte, “β” en las expresiones (4) y (5) y “tc” en la Expresión (5) son valores que se establecen con base en un parámetro de cuantización Q como se indica en la Tabla 1.

[0116]

[Fórmula Matemática 2]

$$d = |p_{2,2} - 2*p_{1,2} + p_{0,2}| + |q_{2,2} - 2*q_{1,2} + q_{0,2}| + |p_{2,5} - 2*p_{1,5} + p_{0,5}| + |q_{2,5} - 2*q_{1,5} + q_{0,5}| \quad \cdots (6)$$

[0117]

[Tabla 1]

Q	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
β	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	7	8
tc	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Q	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
β	9	10	11	12	13	14	15	16	17	18	20	22	24	26	28	30	32	34	36
Tc	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4
Q	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	
B	38	40	42	44	46	48	50	52	54	56	58	60	62	64	64	64	64	64	
tc	5	5	6	6	7	8	9	9	10	10	11	11	12	12	13	13	14	14	

[0118]

En una filtración débil, los componentes de luminancia de los respectivos píxeles en un rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las Expresiones (7) a (11).

[0119]

[Fórmula Matemática 3]

$$p_1' = Clip1_Y(p_1 + \Delta/2) \quad \cdots (7)$$
$$p_0' = Clip1_Y(p_0 + \Delta) \quad \cdots (8)$$
$$q_0' = Clip1_Y(q_0 - \Delta) \quad \cdots (9)$$
$$q_1' = Clip1_Y(q_1 - \Delta/2) \quad \cdots (10)$$

$$\Delta = Clip3(-t_C, t_C, (13*(q_0 - p_0) + 4*(q_1 - p_1) - 5*(q_2 - p_0) + 16) >> 5) \quad \cdots (11)$$

[0120]

En una filtración fuerte, los componentes de luminancia de los respectivos píxeles en un rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las Expresiones (12) a (18).

[0121]

[Fórmula Matemática 4]

$$p_2' = Clip1_Y((2*p_3 + 3*p_2 + p_1 + p_0 + q_0 + 4) >> 3) \quad \cdots (12)$$
$$p_1' = Clip1_Y((p_2 + p_1 + p_0 + q_0 + 2) >> 2) \quad \cdots (13)$$
$$p_0' = Clip1_Y((p_2 + 2*p_1 + 2*p_0 + 2*q_0 + q_1 + 4) >> 3) \quad \cdots (14)$$
$$p_0' = Clip1_Y((p_2 + 2*p_1 + 2*p_0 + 2*q_0 + q_1 + 4) >> 3) \quad \cdots (15)$$
$$q_0' = Clip1_Y((p_1 + 2*p_0 + 2*q_0 + 2*q_1 + q_2 + 4) >> 3) \quad \cdots (16)$$
$$q_1' = Clip1_Y((p_0 + q_0 + q_1 + q_2 + 2) >> 2) \quad \cdots (17)$$
$$q_2' = Clip1_Y((p_0 + q_0 + q_1 + 3*q_2 + 2*q_3 + 4) >> 3) \quad \cdots (18)$$

[0122]

Por otra parte, en la filtración de componentes de crominancia, los componentes de crominancia de los respectivos píxeles en un rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las Expresiones (19) a (21).

[0123]

[Fórmula matemática 5]

$$p_0' = Clip1_C(p_0 + \Delta) \quad \cdots (19)$$
$$q_0' = Clip1_C(q_0 - \Delta) \quad \cdots (20)$$

$$\Delta = Clip3(-t_C, t_C, ((((q_0 - p_0) << 2) + p_1 - q_1 + 4) >> 3)) \quad \cdots (21)$$

[0124]

En las expresiones anteriores, “Clip1Y” y”Clip1c” representan operaciones de las expresiones (22) y (23), y”Clip3(x,y,z)” en las expresiones (22) y (23) representan un valor determinado por la Expresión (24).

[0125]

[Fórmula matemática 6]

$Clip1_Y(x) = Clip3(0,(1<<BitDeopthy) – 1,x)$ …(22)

$Clip1_c(x) = Clip3(0,(1<<BitDeopthy) – 1,x)$ …(23)

$$Clip3(x,y,z) = \begin{cases} x ; z<x \\ y ; z>y \\ Z ; \text{de otra manera} \end{cases} \quad \ldots(24)$$

[0126]

Además, en cuanto al límite de línea, la operación se realiza para cada línea utilizando los píxeles en la dirección horizontal, en el límite vertical se realiza para cada columna usando los píxeles en la dirección vertical, y se realiza el proceso de filtración.

[0127]

En el proceso de filtración de desbloqueo, un proceso que requiere una memoria de línea es el proceso de filtración vertical, y en lo sucesivo se describirá en detalle una reducción en la capacidad de memoria de la memoria de línea en el proceso de filtración vertical. Puesto que la unidad de filtración de desbloqueo 24 del dispositivo de codificación de imagen 10 tiene la misma configuración y la misma operación que el de la unidad de filtración de desbloqueo 56 del dispositivo de decodificación de imagen 50, la unidad de filtración de desbloqueo 24 solamente se describirá.

[0128]

[7. Primera implementación de la unidad de filtración de desbloqueo]

La figura 9 ilustra la configuración de una primera implementación de la unidad de filtración de desbloqueo. La unidad de filtración de desbloqueo 24 incluye una memoria de línea 241, una unidad que detecta el límite de línea 242, una unidad que determina la fuerza del filtro 243, una memoria de coeficiente 244, una unidad de operación de filtro 245, y un controlador de filtro 246.

[0129]

La memoria de línea 241 almacena los datos de imagen suministrados desde el sumador 23 sobre la base de una señal de control desde el controlador de filtro 246. Por otra parte, la memoria de línea 241 lee los datos de imagen almacenados en ella y envía los datos de imagen leídos a la unidad que detecta el límite de línea 242, la unidad que determina la fuerza del filtro 243, y la unidad de operación de filtro 245.

[0130]

La unidad que detecta el límite de línea 242 detecta un límite de línea en el que se realiza el proceso de filtración vertical. La unidad que detecta el límite de línea 242 realiza el proceso que determina la necesidad de filtración para cada bloque usando los datos de imagen suministrados desde el sumador 23 y los datos de imagen leídos de la memoria de línea 241 y detecta el límite de línea en el que se realiza el proceso de filtración vertical. La unidad que detecta el límite de línea 242 envía los resultados de la detección a la unidad que determina la fuerza del filtro 243.

[0131]

La unidad que determina la fuerza del filtro 243 determina la fuerza del filtro como se describió anteriormente. La unidad que determina la fuerza del filtro 243 determina si se realiza el proceso de filtración vertical debe realizar en un modo de filtración fuerte o en un modo de filtración débil utilizando los datos de imagen de dos bloques adyacentes con el límite de línea

interpuesto en el que se realiza el proceso de filtración vertical, y envía los resultados de la determinación a la unidad de operación de filtro 245.

[0132]

La memoria de coeficiente 244 almacena los coeficientes de filtro usados en la operación de filtración del proceso de filtración de desbloqueo.

[0133]

La unidad de operación de filtro 245 realiza una operación de filtración con el filtro fuerte determinado por la unidad que determina la fuerza del filtro 243 usando los datos de imagen suministrados por el sumador 23, los datos de imagen almacenados en la memoria de línea 241, y los coeficientes de filtro leídos desde la memoria de coeficiente 244. La unidad de operación de filtro 245 envía los datos de imagen que han sido sometidos al proceso de filtración vertical a la memoria de cuadro 25. Por otra parte, la unidad de operación de filtro 245 controla un rango de imagen utilizado en la operación de filtración en un bloque que está ubicado sobre la parte superior del límite de línea con base en el resultado de la determinación del límite de bloque suministrado desde el controlador de filtro 246.

[0134]

El controlador de filtro 246 controla la memoria de línea 241 de tal manera que almacena los datos de imagen correspondientes a un número predeterminado de líneas en la parte inferior del bloque. Por otra parte, el controlador de filtro 246 lee los datos de imagen almacenados en la memoria de línea 241. Además, el controlador de filtro 246 incluye una unidad que determina el límite de línea 2461. La unidad que determina el límite de línea 2461 determina si un límite es un límite de línea (por ejemplo, un límite de línea inter-LCU) en las unidades del bloque respectivo en el que se realiza el proceso secuencialmente en una dirección de exploración de trama y envía los resultados de la determinación a la unidad de operación de filtro 245. La unidad de operación de filtro 245 realiza control del proceso de filtración vertical que se puede realizar incluso cuando está reducida la capacidad de memoria de la memoria de línea.

[0135]

La figura 10 ilustra la operación de la primera implementación de la unidad de filtración de desbloqueo. En la etapa ST71, la unidad de filtración de desbloqueo 24 detecta un límite de línea. La unidad de filtración de desbloqueo 24 detecta un límite de línea en el que se realiza el proceso de filtración vertical.

[0136]

En la etapa ST72, la unidad de filtración de desbloqueo 24 determina una fuerza de filtro. La unidad de filtración de desbloqueo 24 determina si se usará un modo de filtración fuerte o un modo de filtración débil para el límite de línea en el que se realizará el proceso de filtración vertical.

[0137]

En la etapa ST73, la unidad de filtración de desbloqueo 24 determina si el límite en el que se realiza el proceso de filtración vertical es un límite de línea inter-LCU. La unidad de filtración de desbloqueo 24 procede a la etapa ST74 cuando el límite en el que se realiza el proceso de filtración vertical es el límite de línea inter-LCU y procede a la etapa ST75 cuando el límite no es un límite de línea inter-LCU.

[0138]

En la etapa ST74, la unidad de filtración de desbloqueo 24 realiza un proceso de filtración vertical de rango reducido. La unidad de filtración de desbloqueo 24 realiza el proceso de filtración vertical a la vez que reduce el rango de imagen, usado para la operación de filtración del LCU adyacente superior y el flujo procede a la etapa ST76.

[0139]

En la etapa ST75, la unidad de filtración de desbloqueo 24 realiza un proceso de filtración vertical normal. La unidad de filtración de desbloqueo 24 realiza el proceso de filtración vertical por medio de toma predeterminada y coeficientes sin reducir el rango de imagen usado para la operación de filtración y el flujo procede a la etapa ST76.

[0140]

En la etapa ST76, la unidad de filtración de desbloqueo 24 determina si se ha completado el proceso de filtración vertical en el límite. Cuando no se ha completado el proceso de filtración vertical para las respectivas columnas del límite, la unidad de filtración de desbloqueo 24 vuelve a la etapa ST76 y lleva a cabo el proceso para la siguiente columna. Por otra parte, cuando el proceso de filtración vertical para las respectivas columnas del límite se ha completado, el flujo procede a la etapa ST77.

[0141]

En la etapa ST77, la unidad de filtración de desbloqueo 24 determina si el proceso se he completado hasta el final de la pantalla. Cuando el proceso no se ha completado hasta el final de la pantalla, la unidad de filtración de desbloqueo 24 vuelve a la etapa ST71, detecta un nuevo límite y realiza el proceso de filtración vertical. Cuando el proceso se ha completado

hasta el final de la pantalla, la unidad de filtración de desbloqueo 24 terminal proceso para 11 pantallas.

[0142]

[Configuración y Operación de la Unidad de Operación de filtro]

Las figuras 11 y 12 ilustran la configuración de la unidad de operación de filtro. Cuando se realiza el proceso de filtración vertical de toma cambiable, la unidad de operación de filtro 245 cambia los datos de imagen de una toma o cambial número de toma de tal manera que se realiza el proceso de filtración vertical pueda realizar incluso cuando se reduce el número de líneas que almacenan datos de imagen. La figura 11 ilustra la configuración de una unidad de operación de filtro cuando se realiza un proceso de filtración vertical mientras que cambia los datos de imagen de una toma. Por otra parte, la figura 12 ilustra la configuración de la unidad de operación de filtro cuando se realiza un proceso de filtración vertical mientras que cambial número de toma.

[0143]

En la figura 11, la unidad de operación de filtro 245 incluye una unidad de almacenamiento de datos 2451, una unidad de selección de datos 2452, y una unidad de procesamiento aritmético 2453.

[0144]

Cuando la capacidad de memoria de una memoria de línea se reduce, la unidad de almacenamiento de datos 2451 almacena los datos de imagen a ser usados para una toma en la posición de la línea reducida. La unidad de almacenamiento de datos 2451 envía los datos de imagen almacenados en ella a la unidad de selección de datos 2452 como los datos de imagen de una toma en la posición de la línea reducida.

[0145]

La unidad de selección de datos 2452 selecciona uno de los datos de imagen almacenados en la unidad de almacenamiento de datos 2451 y los datos de imagen almacenados en la memoria de línea y envía de los datos de imagen seleccionados a la unidad de procesamiento aritmético 2453.

[0146]

La unidad de procesamiento aritmético 2453 realiza un proceso aritmético utilizando los datos de imagen suministrados desde el sumador 23 y la unidad de selección de datos 2452 y

los coeficientes de filtro leídos desde la memoria de coeficiente 244 para generar datos de imagen después del proceso de filtración vertical y envía los datos de imagen a la memoria de cuadro 25.

[0147]

En la figura 12, la unidad de operación de filtro 245 incluye una unidad de procesamiento aritmético 2455 y una unidad de procesamiento aritmético 2456.

[0148]

La unidad de procesamiento aritmético 2455 realiza un proceso aritmético con un número predeterminado de tomas y envía los datos de imagen después del proceso de filtración vertical a la unidad de selección de datos 2457.

[0149]

La unidad de procesamiento aritmético 2456 realiza un proceso aritmético sin usar los datos de imagen de la línea reducida mediante la reducción del número de tomas según una reducción de la capacidad de memoria de la memoria de línea y envía los datos de imagen después del proceso de filtración vertical a la unidad de selección de datos 2457.

[0150]

La unidad de selección de datos 2457 selecciona los datos de imagen en función de si el límite es un límite de línea inter-LCU. Cuando el límite no es el límite inter-LCUs, la unidad de selección de datos 2457 selecciona la salida de los datos de imagen desde la unidad de procesamiento aritmético 2455. Además, cuando el límite es el límite de línea inter-LCU, la unidad de selección de datos 2457 selecciona la salida de los datos de imagen desde la unidad de procesamiento aritmético 2456.

[0151]

Las Figuras 13(A) y 13(B) son diagramas para describir la operación de la unidad de operación de filtro 245. Los dibujos ilustran los datos de imagen de los respectivos píxeles de dos bloques adyacentes BKu y BKl. En este caso, la operación de filtración se realiza usando tres píxeles cada uno de un límite (límite de línea) BB como un rango de procesamiento de filtro del proceso de filtración vertical y cuatro píxeles cada uno del límite BB como una toma. Cuando se determina que la filtración se va a realizar en un modo de filtración fuerte, se realizan las operaciones de las expresiones (12) a (18) descritas anteriormente para cada columna mediante las unidades de procesamiento aritmético 2453 y 2455. Por otra parte, cuando se determina que la filtración se va a realizar en un modo de filtración débil, se realizan las operaciones de las expresiones (7) a (10) descritas anteriormente para cada columna mediante las unidades de procesamiento aritmético 2453 y 2455.

[0152]

Cuando el proceso de filtración de desbloqueo se realiza en un orden de trama en las unidades LCU respectivas, se determina que se realiza el proceso de filtración vertical realiza en un modo de filtración fuerte, y se hacen las operaciones de las expresiones (12) a (18) descritas anteriormente. Por lo tanto, como se ilustra en la figura 13 (A), los datos de imagen del bloque BKu sobre el lado superior del límite BB correspondiente a cuatro líneas del límite BB necesitan ser almacenado en la memoria de línea.

[0153]

En este caso, la unidad de operación de filtro 245 controla el rango de imagen usado para la operación de filtración en el bloque posicionado sobre el lado superior del límite BB por lo cual se realiza el proceso de filtración vertical puede realizar incluso cuando la capacidad de memoria de la memoria de línea es reducida. Por ejemplo, como se ilustra en la figura 13 (B), se almacenan los datos de imagen correspondientes a tres líneas del límite BB, y se realiza el proceso de filtración vertical usando los datos almacenados. Es decir, la unidad de procesamiento aritmético 2453 ilustrado en la figura 11 realiza la operación de la Expresión (25). Por otra parte, la unidad de procesamiento aritmético 2456 ilustrada en la figura 12 realiza la operación de la Expresión (26), y la unidad de selección de datos 2457 selecciona los datos de imagen desde la unidad de procesamiento aritmético 2456 y envía los datos de imagen. En las expresiones (25) y (26), “i” indica un índice de columna de píxel, y cuando se realiza el proceso de filtración en unidades de bloques de 8 x 8 píxeles, “i = 0 a 7”.

[0154]

[Fórmula matemática 7]

$$p_{2i}' = Clip1_V((2*p_{2i} + 3*p_{2i} + p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (25)$$

$$p_{2i}' = Clip1_V((\quad 5*p_{2i} + p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (26)$$

[0155]

La unidad de procesamiento aritmético 2453 copia los píxeles en el bloque superior BKu en el extremo superior del rango de procesamiento de filtro en la dirección hacia arriba y utiliza los mismos. Es decir, mediante el uso de los datos de imagen p2 de los píxeles en el extremo superior del rango de procesamiento de filtro almacenado en la unidad de almacenamiento de datos 2451 como los datos de imagen p3, los datos de imagen p2i' después del proceso de filtración se calculan mediante la operación de la Expresión (25).

[0156]

La unidad de procesamiento aritmético 2456 reduce el número de tomas y calcula los datos de imagen p2i' después del proceso de filtración mediante la operación de la Expresión (26) utilizando los coeficientes de cambios con la reducción en el número de tomas. En este caso, los coeficientes se modifican de tal manera que el coeficiente de los datos de imagen p2 se cambia desde “3” a “5” o de modo que se correspondan a la copia de los pixeles del extremo superior en el rango de la toma.

[0157]

Cuando el límite BB no es el límite de línea inter-LCU, no es necesario usar los datos de imagen de la memoria de línea 241, se realiza la misma operación de filtración que en la operación convencional sin estrechar el rango de imagen usado para la operación de filtración. Es decir, la unidad de procesamiento aritmético 2453 ilustrada en la figura 11 realiza la misma operación de filtración que la operación convencional. Por otra parte, la unidad de procesamiento aritmético 2455 ilustrada en la figura 12 realiza la misma operación de filtración que la operación convencional y la unidad de selección de datos 2453 seleccionan y envía los datos de imagen desde la unidad de procesamiento aritmético 2455.

[0158]

De esta manera, cuando el límite es un límite de línea de los bloques en los que se ha realizado secuencialmente el proceso con el propósito de escanear la trama, mediante el control del rango de imagen usado para la operación de filtración, incluso cuando se reduce la capacidad de memoria de la línea de memoria, se puede realizar el proceso de filtración de desbloqueo de la misma manera que antes de la reducción. Por otra parte, por ejemplo, una línea de imagen de 4K x 2K corresponde a dos líneas de imagen de 2K x 1K. Además, en el esquema de H.264/AVC, una memoria de línea corresponde a cuatro líneas incluidas, y la capacidad de memoria correspondiente a una línea de la imagen de 4K x 2K corresponde al 50% de la capacidad de memoria del esquema H.264/AVC. Por lo tanto, se mejora el efecto de reducción de la capacidad de memoria en una imagen de alta resolución.

[0159]

[8. Segunda implementación de la unidad de filtración de desbloqueo]

Una segunda implementación de la unidad de filtración de desbloqueo es diferente de la primera implementación en que las operaciones de la unidad de procesamiento aritmético 2453 y la unidad de procesamiento aritmético 2456 son diferentes de las de la primera implementación.

[0160]

En la filtración fuerte, cuando el límite BB es el límite de línea inter-LCU, y se realizan las operaciones de las Expresiones (12) a (18) descritas anteriormente, como se ilustran en la figura 13(A), los datos de imagen del bloque BKu en la parte superior del límite BB correspondiente a cuatro líneas desde el límite BB necesita ser almacenada en la memoria de línea.

[0161]

En este caso, la unidad de operación de filtro 245 controla el rango de imagen usado para la operación de filtración en el bloque situado en el lado superior del límite BB por lo cual se puede realizar el proceso de filtración vertical, incluso cuando está reducida la capacidad de memoria de la memoria de línea. Por ejemplo, como se ilustra en la figura 13(B), se almacenan los datos de imagen correspondiente a tres líneas desde el límite BB, y se realiza el proceso de filtración vertical utilizando los datos de imagen almacenados. Es decir, la unidad de procesamiento aritmético 2453 ilustrada en la fig. 11 realiza la operación de la Expresión (27). Por otra parte, la unidad de procesamiento aritmético 2456 ilustrada en la fig. 12 realiza la operación de la Expresión (28), y la unidad de selección de datos 2457 selecciona los datos de imagen a partir de la unidad de procesamiento aritmético 2456 y envía los datos de imagen. En las Expresiones (27) y (28), “i” indica un índice columna de píxel, y cuando se realiza el proceso de filtración en unidades de bloques de 8 X 8 pixeles, “i = 0 a 7”.

[0162]

[Formula Matemática 8]

$$p_{2i}' = \mathrm{Clip1}_Y((2*p_{1i} + 3*p_{2i} + p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (27)$$

$$p_{2i}' = \mathrm{Clip1}_Y((3*p_{2i} + 3*p_{1i} + p_{0i} + q_{0i} + 4) >> 3) \quad \cdots (28)$$

[0163]

La unidad de procesamiento aritmético 2453 realiza copiado de espejo sobre los píxeles en el bloque superior BKu en el extremo superior del rango de procesamiento de filtro. Es decir, mediante el almacenamiento de los datos de imagen p1 en la unidad de almacenamiento de datos 2451, la copia de espejo de los datos de imagen p1 sobre los píxeles del extremo superior en el rango de procesamiento de filtro usa los datos de imagen p1 como los datos de imagen p3, los datos de imagen p2i' después del proceso de filtración se calculan por la operación de la Expresión (27).

[0164]

La unidad de procesamiento aritmético 2456 reduce el número de tomas y calcula los datos de imagen p2i' después del proceso de filtración por la operación de la Expresión (28)

utilizando los coeficientes de cambios con la reducción del número de tomas. En este caso, los coeficientes se modifican de tal manera que el coeficiente de los datos de imagen p1 se cambia de “2” a “3” o de manera que correspondan a la copia de espejo sobre los píxeles del extremo superior del rango de procesamiento de filtro.

[0165]

Cuando el límite BB no es el límite de línea inter-LCU, no es necesario usar los datos de imagen de la memoria de línea 241, se realiza la misma operación de filtración que en la operación convencional sin estrechar el rango de imagen utilizado para la operación de filtración. Es decir, la unidad de procesamiento aritmético 2453 ilustrada en la figura 11 realiza la misma operación de filtración que la operación convencional. Por otra parte, la unidad de procesamiento aritmético 2455 ilustrada en la figura 12 realiza la misma operación de filtración que la operación convencional y la unidad de selección de datos 2457 seleccionan y envía los datos de imagen desde la unidad de procesamiento aritmético 2455.

[0166]

De esta manera, cuando el límite es un límite de línea de bloques en el que el proceso se realiza secuencialmente en un orden de escaneo de trama, mediante el control del rango de imagen usado para la operación de filtración, de manera similar a la primera implementación, incluso cuando se reduce la capacidad de memoria, se puede realizar el proceso de filtración de desbloqueo.

[0167]

[9. Tercera implementación de la unidad de filtración de desbloqueo]

Una tercera implementación de la unidad de filtración de desbloqueo es diferente de la primera y segunda implementaciones en que las operaciones de la unidad de procesamiento aritmético 2453 y la unidad de procesamiento aritmético 2456 son diferentes de la primera implementación.

[0168]

Las Figuras 14(A) y 14(B) son diagramas para describir la operación de la unidad de operación de filtro 245. Los dibujos ilustran los datos de imagen de los respectivos píxeles de dos bloques adyacentes BKu y BKl. Aquí, la operación de filtración se realiza utilizando tres píxeles cada uno de un límite (límite de línea) BB como un rango de procesamiento de filtro y cuatro pixeles cada uno desde el límite BB como una toma. Cuando se determina que la filtración se va a realizar en el modo de filtración fuerte, se realizan las operaciones de las expresiones (12) a (18) descritas anteriormente para cada columna mediante las unidades de procesamiento aritmético 2453 y 2455. Por otra parte, cuando se determina que se va a realizar

un proceso de filtración en el modo de filtración débil, se realizan las operaciones de las Expresiones (7) a (10) descritas anteriormente para cada columna por las unidades de procesamiento aritmético 2453 y 2455.

[0169]

Cuando se realiza el proceso de filtración de desbloqueo en un orden de trama en unidades LCU respectivas, se determina que el proceso de filtración vertical se realiza en un modo de filtración fuerte, y se realizan las operaciones de las expresiones (12) a (18) descritas anteriormente. Por lo tanto, como se ilustra en la figura 14 (A), los datos de imagen del bloque BKu en la parte superior del límite BB correspondiente a cuatro líneas a partir del límite BB necesitan ser almacenadas en la memoria de línea.

[0170]

Aquí, la unidad de operación de filtro 245 controla el rango de procesamiento de filtro y el rango de imagen utilizado para la operación de filtración en el bloque situado en la parte superior del límite BB con lo cual el proceso de filtración vertical se puede llevar a cabo incluso cuando está reducida la capacidad de memoria de la memoria de línea. Por ejemplo, como se ilustra en la figura 14 (B), el rango de procesamiento de filtro en el bloque superior BKu se establece en un rango de dos píxeles desde el límite BB, y se almacenan los datos de imagen correspondientes a dos líneas desde el límite BB. Además, el proceso de filtración vertical se realiza usando los datos de imagen almacenados de manera similar a la primera implementación. Es decir, la unidad de procesamiento aritmético 2453 ilustrada en la figura 11 realiza las operaciones de las expresiones (29) y (30). Por otra parte, la unidad de procesamiento aritmético 2456 ilustrada en la figura 12 realiza las operaciones de las expresiones (31) y (32), y la unidad de selección de datos 2457 selecciona los datos de imagen desde la unidad de procesamiento aritmético 2456 y envía los datos de imagen. En las expresiones (29) a (32), “i” indica un índice de columna de píxel, y cuando se realiza el proceso de filtración en unidades de bloques de 8 x 8 píxeles, “i = 0 a 7”.

[0171]

[Fórmula Matemática 9]

$$p_{1i}' = \mathrm{Clip1}_Y((p_{1i} + p_{1i} + p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (29)$$

$$p_{0i}' = \mathrm{Clip1}_Y((p_{1i} + 2*p_{1i} + 2*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (30)$$

$$p_{1i}' = \mathrm{Clip1}_Y((2*p_{1i} + p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (31)$$

$$p_{0i}' = \mathrm{Clip1}_Y((3*p_{1i} + 2*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (32)$$

[0172]

La unidad de procesamiento aritmético 2453 copia los pixeles en el bloque superior BKu sobre el extremo superior del rango de procesamiento de filtro en la dirección hacia arriba y utiliza la misma. Es decir, mediante el uso de los datos de imagen p1 de los pixeles del extremo superior del rango de procesamiento de filtro almacenados en la unidad de almacenamiento 2451 como los datos de imagen p2, los datos de imagen pl_i' y p0o' después del proceso de filtración se calcula por las operaciones de las expresiones (29) y (30).

[0173]

La unidad de procesamiento aritmético 2456 reduce el número de tomas y calcula los datos de imagen pli' y p0i' después del proceso de filtración por las operaciones de las Expresiones (31) y (32) utilizando los coeficientes de cambios con la reducción en el número de tomas. En este caso, los coeficientes se modifican de tal manera que el coeficiente del dato de imagen p1 en la Expresión (31) se cambia de “1” a “2” y el dato de imagen p1 en la Expresión (32) se cambia de “2” a “3” por lo que corresponde a la copia de los píxeles del extremo superior del rango de la toma.

[0174]

Cuando el límite BB no es el límite de línea inter-LCU, no es necesario utilizar los datos de imagen de la memoria de línea 241, se realiza la misma operación de filtración que en la operación convencional sin reducir el rango de imagen usado para la operación de filtración. Es decir, la unidad de procesamiento aritmético 2453 ilustrada en la figura 11 realiza la misma operación de filtración que la operación convencional. Por otra parte, la unidad de procesamiento aritmético 2455 ilustrada en la figura 12 realiza la misma operación de filtración que la operación convencional y la unidad de selección de datos 2457 seleccionan y envía los datos de imagen desde la unidad de procesamiento aritmético 2455.

[0175]

De esta manera, cuando el límite es un límite de línea de bloques en el que se realiza secuencialmente el proceso con el propósito de escanear la trama, por control del rango de procesamiento de filtro y el rango de imagen utilizado para la operación de filtración, incluso cuando la capacidad de la memoria es reducida, se puede realizar el proceso de filtración de desbloqueo de la misma manera que antes de la reducción. Por otra parte, es posible reducir la capacidad de memoria mucho más.

[0176]

[10. Cuarta implementación de la unidad de filtración de desbloqueo]

Una cuarta implementación de la unidad de filtración de desbloqueo es diferente de la primera implementación en que las operaciones de la unidad de procesamiento aritmético 2453

y la unidad de procesamiento aritmético 2456 son diferentes de las de la tercera implementación.

[0177]

En la filtración fuerte, cuando el límite BB en un límite de línea LCU, y se realizan las operaciones de las Expresiones (12) a (18) descritas anteriormente, como se ilustra en la figura 14 (A), los datos de imagen del bloque BKu sobre la parte superior del límite BB correspondiente a cuatro líneas desde el límite BB necesita ser almacenada en la memoria de línea.

[0178]

Aquí, la unidad de operación de filtro 245 controla el rango de procesamiento de filtro y el rango de imagen usados para la operación de filtración en el bloque situado en el lado superior del límite BB por lo cual se puede realizar el proceso de filtración vertical incluso cuando se reduce la capacidad de memoria de la memoria de línea. Por ejemplo, como se ilustra en la figura 14 (B), el rango de procesamiento de filtro en el bloque superior BKu se establece en un rango de dos píxeles desde el límite BB, y se almacenan los datos de imagen correspondientes a dos líneas desde el límite BB. Además, se realiza el proceso de filtración vertical usando los datos de imagen almacenados de manera similar a la segunda implementación. Es decir, la unidad de procesamiento aritmético 2453 ilustrada en la figura 11 realiza las operaciones de las expresiones (33) y (34). Por otra parte, la unidad de procesamiento aritmético 2456 ilustrada en la figura 12 realiza las operaciones de las expresiones (35) y (36), y la unidad de selección de datos 2457 selecciona los datos de imagen desde la unidad de procesamiento aritmético 2456 y envía los datos de imagen. En las expresiones (33) a (36), “i” indica un índice de columna de píxel, y cuando se realiza el proceso de filtración en unidades de bloques de 8 x 8 píxeles, “i = 0 a 7”.

[0179]

[Fórmula Matemática 10]

$$p_{1i}' = Clip1_Y((p_{0i} + p_{1i} + p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (33)$$

$$p_{0i}' = Clip1_Y((p_{0i} + 2*p_{1i} + 2*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (34)$$

$$p_{1i}' = Clip1_Y((p_{1i} + 2*p_{0i} + q_{0i} + 2) >> 2) \quad \cdots (35)$$

$$p_{0i}' = Clip1_Y((2*p_{1i} + 3*p_{0i} + 2*q_{0i} + q_{1i} + 4) >> 3) \quad \cdots (36)$$

[0180]

La unidad de procesamiento aritmético 2453 realiza copiado de espejo sobre los píxeles en el bloque superior BKu en el extremo superior del rango de procesamiento de filtro. Es decir,

mediante el almacenamiento de los datos de imagen p0 en la unidad de almacenamiento de datos 2451, la copia de espejo de los datos de imagen p0 sobre los píxeles del extremo superior en el rango de procesamiento de filtro usa los datos de imagen p0 como los datos de imagen p2, los datos de imagen p1i' y p0i' después del proceso de filtración se calculan por la operación de las Expresiones (33) y (34).

[0181]

La unidad de procesamiento aritmético 2456 reduce el número de tomas y calcula los datos de imagen p1i' y p0i' después del proceso de filtración por la operación de las Expresiones (35) y (36) utilizando los coeficientes de cambios con la reducción del número de tomas. En este caso, los coeficientes se modifican de tal manera que el coeficiente de los datos de imagen p0 en la Expresión (35) se cambian “1” a “2” y los datos de imagen p0 en la Expresión (36) se cambian de “2” a “3” de manera que correspondan a la copia de espejo sobre los píxeles del extremo superior del rango de procesamiento de filtro.

[0182]

Cuando el límite BB no es el límite de línea inter-LCU, no es necesario usar los datos de imagen de la memoria de línea 241, se realiza la misma operación de filtración que en la operación convencional sin estrechar el rango de imagen utilizado para la operación de filtración. Es decir, la unidad de procesamiento aritmético 2453 ilustrada en la figura 11 realiza la misma operación de filtración que la operación convencional. Por otra parte, la unidad de procesamiento aritmético 2455 ilustrada en la figura 12 realiza la misma operación de filtración que la operación convencional y la unidad de selección de datos 2457 seleccionan y envía los datos de imagen desde la unidad de procesamiento aritmético 2455.

[0183]

De esta manera, cuando el límite es un límite de línea de bloques en el que el proceso se realiza secuencialmente en un orden de escaneo de trama, mediante el control del rango de procesamiento de filtro y el rango de imagen usados para la operación de filtración, incluso cuando se reduce la capacidad de memoria, se puede realizar el proceso de filtración de desbloqueo de la misma manera que antes de la reducción. Por otra parte, es posible reducir la capacidad de memoria mucho más.

[0184]

[11. Quinta implementación de la unidad de filtración de desbloqueo]

La unidad de filtración de desbloqueo descrita anteriormente controla el rango de imagen que se utiliza para la operación de filtración en un bloque situado en el lado superior de un límite y controla el rango de procesamiento de filtro dependiendo de si el límite BB es un límite de

línea inter-LCU. Más adelante, en la quinta implementación, se proporcionará la descripción de un caso en el que un modo donde el rango de imagen se controla cuando el límite BB es el límite de la línea inter-LCU y un modo donde el rango de imagen se controla dependiendo de si el límite BB es el límite de línea inter-LCU.

[0185]

La figura 15 es un diagrama de flujo que ilustra la operación de la quinta implementación. En la etapa ST81, la unidad de filtración de desbloqueo 24 detecta un límite de línea. La unidad de filtración de desbloqueo 24 detecta un límite de línea en el que se realiza el proceso de filtración vertical.

[0186]

En la etapa ST82, la unidad de filtración de desbloqueo 24 determina una fuerza del filtro. La unidad de filtración de desbloqueo 24 determina si se va a utilizar el modo de filtración fuerte o el modo de filtración débil para el límite de línea en el que se realiza el proceso de filtración vertical.

[0187]

En la etapa ST83, la unidad de filtración de desbloqueo 24 determina si se va a realizar un proceso de filtración vertical de rango reducido con respecto al límite de línea inter-LCU solamente (es decir, si un proceso de filtración vertical si se va a realizar un proceso de filtración vertical mediante la reducción del rango de imagen usado para la operación de filtración). Cuando se determina que el proceso de filtración vertical de rango reducido se va a realizar con respecto al límite de línea de los bloques más pequeños que el tamaño de la LCU así como también el límite de línea inter-LCU, la unidad de filtración de desbloqueo 24 procede a la etapa ST84. Por otra parte, cuando se determina que se va a realizar el proceso de filtración vertical de rango reducido con respecto al límite de línea inter-LCU solamente, la unidad de filtración de desbloqueo 24 procede a la etapa ST85.

[0188]

La unidad de filtración de desbloqueo 24 determina si se va a realizar el proceso de filtración vertical de rango reducido con respecto al límite de línea inter-LCU solamente basado en los parámetros de cuantización establecidos en unidades de cuadro, por ejemplo. Cuando el parámetro de cuantización es pequeño, se obtiene mejor calidad de imagen comparada con la que se obtiene cuando el parámetro de cuantización es grande. Así, cuando el parámetro de cuantización es mayor que el valor umbral predeterminado, la unidad de filtración de desbloqueo 24 determina que se va a realizar el proceso de filtración vertical en un modo en el que la calidad de imagen se mejora mediante la realización del proceso de filtración vertical de

rango reducido con respecto al límite de línea inter-LCU solamente y realiza un proceso de filtración vertical normal con respecto a un límite de línea de los bloques más pequeños que el tamaño de la LCU, y procede a la etapa ST84. Además, cuando el parámetro de cuantización es igual o menor que el valor umbral predeterminado, la unidad de filtración de desbloqueo 24 determina que se va a realizar el proceso de filtración vertical en un modo en el que el control se hace de forma fácil mediante la realización de un proceso de filtración vertical de rango reducido con respecto a la línea límite inter-bloque más pequeños que el tamaño de la LCU así como también el límite de línea inter-LCU, y procede a la etapa ST85.

[0189]

En la etapa ST84, la unidad de filtración de desbloqueo 24 determina si un límite en el que se ve a realizar el proceso de filtración vertical es un límite de línea inter-LCU. La unidad de filtración de desbloqueo 24 procede a la etapa ST85 cuando el límite en el que se realiza el proceso de filtración vertical es un límite de línea inter-LCU y procede a la etapa ST86 cuando el límite es un límite de línea de bloques más pequeños que el tamaño de la LCU.

[0190]

En la etapa ST85, la unidad de filtración de desbloqueo 24 realiza un proceso de filtración vertical de rango reducido. La unidad de filtración de desbloqueo 24 realiza el proceso de filtración vertical a la vez que reduce el rango de imagen usado para la operación de filtración de la LCU adyacente superior y procede a la etapa ST87. En el proceso de filtración vertical de rango reducido, se puede reducir el rango de filtración.

[0191]

En la etapa ST86, la unidad de filtración de desbloqueo 24 realiza un proceso de filtración vertical normal. La unidad de filtración de desbloqueo 24 realiza el proceso de filtración vertical sin reducir el rango de imagen que se utiliza para la operación de filtración y procede a la etapa ST87.

[0192]

En la etapa ST87, la unidad de filtración de desbloqueo 24 determina si se ha completado el proceso de filtración vertical en el límite. Cuando no se ha completado el proceso de filtración vertical para las respectivas columnas del límite, la unidad de filtración de desbloqueo 24 vuelve a la etapa ST87 y realiza el proceso para la próxima columna. Además, cuando se ha completado el proceso de filtración vertical para las respectivas columnas del límite, el flujo procede a la etapa ST88.

[0193]

En la etapa ST88, la unidad de filtración de desbloqueo 24 determina si el proceso se ha completado hasta el final de la pantalla. Cuando el proceso no se ha completado hasta el final de la pantalla, la unidad de filtración de desbloqueo 24 vuelve a la etapa ST81, detecta un nuevo límite y realiza el proceso de filtración vertical. Cuando el proceso se ha completado hasta el final de la pantalla, la unidad de filtración de desbloqueo 24 termina el proceso para una pantalla.

[0194]

Cuando se realiza un proceso de este tipo, es posible reducir la capacidad de memoria de la memoria de línea. Cuando se realiza el proceso de filtración vertical de rango reducido con respecto a la otra línea así como también los límites de línea inter-LCU, no se requiere la conmutación del proceso de filtro y el control se hace fácil. Por otra parte, cuando el proceso de filtración vertical de rango reducido se realiza únicamente con respecto al límite de línea inter-LCU, se puede obtener mejor calidad de imagen.

[0195]

[12. Descripción de la sexta a la octava implementación]

[Descripción de la Técnica Convencional]

En la descripción anterior, se ha descrito un ejemplo en el que se copian los datos de imagen de una toma en el rango de imagen o una copia de espejo y se utilizan como datos de imagen de la toma que esta fuera del rango de imagen debido al estrechamiento del rango de imagen utilizado para la operación de filtración. Aquí, en la presente especificación, la copia es sinónimo de relleno.

[0196]

Como en la presente técnica, solamente se ha propuesto un método de realización de un proceso de filtración utilizando copiado en el límite de línea de las LCUs en JCTVC - F053.

[0197]

Haciendo referencia a la figura 16, se describirá en el esquema HEVC un proceso de filtración de un filtro fuerte y un proceso de filtración usando copiado para las señales de luminancia.

[0198]

La figura 16 es un diagrama explicativo que ilustra un ejemplo de pixeles en dos bloques BKu y BKl que están adyacentes verticalmente con un límite de línea interpuesto. En el ejemplo de la figura 16, los datos de imagen de píxeles en el bloque BKu se representan por el símbolo

“pj_i”. Aquí, “j” es un índice de fila de píxeles y "i” es el índice de la columna de píxeles. Por otra parte, la unidad más pequeña del proceso de codificación es un bloque de 8 x 8 píxeles, y se asignan secuencialmente los índices de fila “j” de 0, 1, 2, y 3 (desde la parte superior hasta el fondo) desde la fila cerca al límite de línea BB. Se asignan los índices de columna “i” de 0, 1, 2,..., y 7 desde la izquierda hacia la derecha del bloque. La mitad superior del bloque BKu se omite en el dibujo. Por otro lado, los datos de imagen de los pixeles en el bloque BKl se representan por el símbolo “qk_i”. Aquí, “k” es un índice de fila de píxel y "i” es un índice de columna de píxel. Los índices de fila “k” de 0, 1, 2 y 3 se asignan secuencialmente (desde la parte superior hacia el fondo) desde la fila cerca al límite de línea BB. La mitad inferior del bloque BKl se omite en el dibujo.

[0199]

En la filtración fuerte de la señal de luminancia en el esquema HEVC, los componentes de luminancia de los respectivos píxeles en el rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las expresiones (37) a (42). Las Expresiones (37) a (42) corresponden a las expresiones (14), (16), (13), (17), (12), y (18), respectivamente.

[0200]

[Fórmula Matemática 11]

$$p0_0 = Clip_{0-255}((p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(37)$$
$$q0_0 = Clip_{0-255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(38)$$
$$p1_0 = Clip_{0-255}((p2_i+p1_i+p0_i+q0_i+2)>>2); \quad i=0,7 \quad \cdots(39)$$
$$q1_0 = Clip_{0-255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(40)$$
$$p2_0 = Clip_{0-255}((2*p3_i+3*p2_i+p1_i+p0_i+q0_i+4)>>3); \quad i=0,7 \quad \cdots(41)$$
$$q2_0 = Clip_{0-255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(42)$$

Aquí, “$Clip_{0-255}$” indica un proceso de recorte que redondea un valor hasta “0” cuando el valor es “0” o más pequeño y redondea un valor hasta “255” cuando el valor es “255” o más grande. Lo mismo se aplica a las siguientes expresiones.

[0201]

En contraste, en la propuesta convencional descrita anteriormente, el proceso de filtración se realiza utilizando el relleno con reducción R2W2 aunque el proceso de filtración se realice en el límite de línea de las CLU solamente. Aquí, R2W2 representa que el proceso de filtración se aplica a dos píxeles en el límite de línea inter-LCU haciendo referencia a dos píxeles en el límite de línea inter-LCU.

[0202]

En la filtración fuerte de las señales de luminancia en la propuesta convencional, los componentes de luminancia de los píxeles respectivos en el rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las Expresiones (43) a (47).

[0203]

[Fórmula Matemática 12]

$$\cdot\ p0_0 = Clip_{0-255}((p1_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3);\quad i=0,7 \quad \cdots(43)$$
$$\cdot\ q0_0 = Clip_{0-255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3);\quad i=0,7 \quad \cdots(44)$$
$$\cdot\ p1_0 = Clip_{0-255}((p1_i+p1_i+p0_i+q0_i+2)>>2);\quad i=0,7 \quad \cdots(45)$$
$$\cdot\ q1_0 = Clip_{0-255}((p0_i+q0_i+q1_i+q2_i+2)>>2);\quad i=0,7 \quad \cdots(46)$$
$$\cdot\ q2_0 = Clip_{0-255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3);\quad i=0,7 \quad \cdots(47)$$

[0204]

Aquí, la Expresión (43) de la filtración fuerte en la propuesta convencional es diferente en que “p2i” en la Expresión (37) de la filtración fuerte del esquema HEVC se cambia a la expresión “$p1_i$” la Expresión (45) de la filtración fuerte en la propuesta convencional en que “p2i” en la Expresión (39) de la filtración fuerte del esquema HEVC se cambia a “$p1_i$”. La filtración fuerte en la propuesta convencional es diferente en que se suprime una expresión correspondiente a la Expresión (41) de la filtración fuerte de la HEVC.

[0205]

Las Expresiones (44), (46), y (47) de la filtración fuerte en la propuesta convencional son las mismas que las Expresiones (38), (40), y (42), respectivamente, de la filtración fuerte del esquema HEVC.

[0206]

Es decir, en la propuesta convencional, puesto que el pixel “p2i” en la tercera fila desde el límite de línea BB del bloque BKu no se refiere al píxel “$p1_i$” sobre una fila inmediatamente por debajo de la tercera fila de la misma columna se rellena y se usa en su lugar.

[0207]

Por otra parte, el proceso de filtración no se aplica al píxel “$p2_0$” en la tercera fila del límite de línea BB del bloque BKu. Por lo tanto, en la filtración fuerte del proceso convencional, se suprime una expresión que corresponde a la Expresión (41) de la filtración fuerte del esquema HEVC.

[0208]

De esta manera, en la propuesta convencional, la capacidad de memoria de la memoria de línea se suprime puesto que llega a ser más grande que la del esquema HEVC.

[0209]

Sin embargo, en el caso de imágenes 4k, puesto que es necesario tener una memoria de línea grande, se requiere además reducir la memoria de línea en el proceso de desbloqueo. Por otra parte, en la reducción de la memoria de línea, se requiere mantener una mejor función de reducción de ruido en el bloque del proceso de desbloqueo que en el método de la propuesta convencional.

[0210]

Aquí, en la determinación del límite inter-bloque en el esquema HEVC, como se indica por la Expresión (48) más adelante, se somete al proceso una forma de onda que tiene una inclinación constante en ambos lados de un límite de línea. Por lo tanto, en la presente técnica, el proceso de desbloqueo en el límite de línea inter-LCU se realiza utilizando la inclinación de la forma de onda usada en la determinación del límite inter-bloque, que se describirá en detalle más adelante.

[0211]

[Fórmula Matemática 13]

$$|p2_2-2*p1_2+p0_2|+|q2_2-2*q1_2+q0_2|+|p2_5-2*p1_5+p0_5|+|q2_5-2*q1_5+q0_5|<\beta \quad \cdots(48)$$

[0212]

[Descripción de la presente técnica (Aproximación lineal)]

En el ejemplo de la figura 17, se ilustra una expresión para la determinación del límite inter–bloque usado en el esquema HEVC, de la Expresión (48).

[0213]

Como se ilustra en la figura 17, el primer término en el lado izquierdo de la Expresión (48) es una expresión para la determinación de un píxel en la tercera columna desde el lado izquierdo del bloque BKu y puede expresarse por una diferencia de diferencias (diferencia de segundo orden) (es decir, un diferencial de segundo orden que tiene una inclinación constante.

$$p2_2 - 2*p1_2 + p0_2 = (p2_2 - p1_2) - (p1_2 - p0_2)$$

[0214]

Los términos segundo a cuarto en el lado izquierdo de la Expresión (48) son una expresión para la determinación de un píxel en la sexta columna desde el lado izquierdo del bloque BKu, una expresión para la determinación de un píxel en la tercera columna desde el lado izquierdo del bloque BKl, y una expresión para la determinación de un píxel en la sexta columna de la izquierda del bloque BKl, respectivamente. Lo mismo se puede decir para los términos segundo a cuarto en el lado izquierdo de Expresión (48).

[0215]

Como anteriormente, en la determinación del límite inter-bloque en el esquema HEVC, puesto que la forma de onda que tiene una inclinación constante se somete al proceso, el proceso de desbloqueo en el límite de línea inter-LCU usa la inclinación de la forma de onda utilizada en la determinación del límite inter–bloque.

[0216]

En la técnica convencional, en cuanto a los píxeles que no pudieron ser contemplados, el proceso de desbloqueo en el límite de línea inter-LCU utiliza relleno. En contraste, en la presente técnica, incluso para los píxeles que no pudieron ser contemplados, el proceso de desbloqueo en el límite inter-LCU realiza aproximación lineal utilizando la inclinación de la forma de onda.

[0217]

Es decir, en el proceso de desbloqueo en el límite de línea inter-LCU en la presente técnica, se utiliza la Expresión (49) siguiente de aproximación lineal.

[0218]

[Fórmula Matemática 14]

$$p2_i = p1_i + (p2_i - p1_i) \fallingdotseq p1_i + (p1_i - p0_i) = 2*p1_i - p0_i \qquad \cdots(49)$$

[0219]

Como se ilustra en la figura 18, cuando los valores de píxeles de $p2_i$, $p1_i$, y $p0_i$ tienen una inclinación, el valor de píxel de $p2_i$ no se puede referir en el relleno a base de R2W2 convencional. Por lo tanto, aunque el valor de píxel de $p1_i$ inmediatamente por debajo del píxel $p2_i$ esté relleno y se use en lugar del valor de píxel de $p2_i$ como se indica mediante el círculo sombreado, puesto que el valor real de píxel de $p2_i$ está en la posición del círculo de puntos, se produce un error.

[0220]

En contraste, en la predicción lineal basada en R2W2 de la presente técnica, puesto que el valor de píxel de $p2_i$ no se puede referir, se utiliza un valor de píxel (= el mismo valor de píxel que el valor real de píxel) predicho por la predicción lineal a partir de la inclinación basada en los valores de píxel de $p1_i$ y $p0_i$ como el valor de píxel de $p2_i$ como se indica por el círculo sombreado.

[0221]

De esta manera, cuando los valores de píxel tienen una inclinación, la presente técnica no incurre en más errores que el método de relleno convencional. Por lo tanto, aunque se utiliza la técnica basada en R2W2, es posible reducir la memoria de línea a la vez que se mantiene la función de reducción de ruido de bloque del proceso de desbloqueo.

[0222]

Por otra parte, por ejemplo, una línea de imagen de 4K x 2K corresponde a dos líneas de imagen de 2K x 1K. Además, en el esquema H.264/AVC, se incluye una memoria de línea correspondiente a cuatro líneas, y la capacidad de memoria correspondiente a una línea de imagen de 4K x 2K corresponde al 50% de la capacidad de memoria del sistema H.264/AVC. Por lo tanto, se mejora el efecto de reducción de la capacidad de memoria en una imagen de alta resolución.

[0223]

Cuando los valores de píxel de $p2_i$, $p1_i$, y $p0_i$ no tienen una inclinación, se obtiene sustancialmente el mismo resultado que cuando se utiliza el valor de píxel $p1_i$ inmediatamente por debajo del píxel $p2_i$. En este caso, si bien se utiliza la técnica basada en R2W2, es posible reducir la memoria de línea a la vez que mantener la función de reducción de ruido del bloque del proceso de desbloqueo.

[0224]

[Proceso de filtración para el caso R2W2]

Más adelante, se describirá el proceso de filtración de las señales de luminancia para el caso R2W2. Aquí, R2 representa el número de píxeles a que se refieren y W2 representa el número de píxeles a los cuales se aplica el proceso de filtración. Es decir, R2W2 representa que el proceso de filtración se aplica a dos píxeles en el límite de línea inter-LCU haciendo referencia a dos píxeles en el límite de línea inter–LCU que se ilustra en la fig. 19.

[0225]

En el ejemplo de la figura 19, se ilustra que los píxeles en la tercera y cuarta columnas desde el límite de línea BB en el bloque BKu en la figura 16 no podrían ser a los que se refiere.

[0226]

Como se ilustra de nuevo por comparación, en la filtración fuerte de la señal de luminancia en el esquema HEVC, los componentes de luminancia de los píxeles respectivos en el rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las expresiones (50) a (55). Las Expresiones (50) a (55) corresponden a las expresiones (37) a (42), respectivamente.

[0227]

[Fórmula Matemática 15]

$$p0_0 = Clip_{0-255}((p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(50)$$
$$q0_0 = Clip_{0-255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(51)$$
$$p1_0 = Clip_{0-255}((p2_i+p1_i+p0_i+q0_i+2)>>2); \quad i=0,7 \quad \cdots(52)$$
$$q1_0 = Clip_{0-255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(53)$$
$$p2_0 = Clip_{0-255}((2*p3_i+3*p2_i+p1_i+p0_i+q0_i+4)>>3); \quad i=0,7 \quad \cdots(54)$$
$$q2_0 = Clip_{0-255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(55)$$

[0228]

En contraste, en la filtración fuerte de la señal de luminancia en el caso R2W2 de la presente técnica, los componentes de luminancia de los respectivos píxeles en el rango del procesamiento de filtro se calculan mediante la realización de las operaciones de las expresiones (56) a (60).

[0229]

[Fórmula Matemática 16]

$$p0_0 = Clip_{0-255}((4*p1_i+p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(56)$$
$$q0_0 = Clip_{0-255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(57)$$
$$p1_0 = Clip_{0-255}((3*p1_i+q0_i+2)>>2); \quad i=0,7 \quad \cdots(58)$$
$$q1_0 = Clip_{0-255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(59)$$
$$q2_0 = Clip_{0-255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(60)$$

[0230]

Aquí, la Expresión (56) de la filtración fuerte del caso R2W2 es diferente en que "$p2_i$ + 2*$p1_i$ + 2*$p0_i$" en la Expresión (50) de la filtración fuerte del esquema HEVC se cambia a "4* $p1_i$ + $p0_i$", de acuerdo con la Expresión (49) de la aproximación lineal.

[0231]

La Expresión (58) de la filtración fuerte del caso R2W2 es diferente en que "$p2_i$ + 2*$p1_i$ + $p0_i$" en la Expresión (52) de la filtración fuerte del esquema HEVC se cambia a "3* $p1_i$" de acuerdo con la Expresión (49) de la aproximación lineal.

[0232]

La filtración fuerte del caso R2W2 es diferente en que se suprime una expresión correspondiente a la Expresión (54) de la filtración fuerte el esquema HEVC.

[0233]

Es decir, en el caso R2W2 de la presente técnica, puesto que no se conoce el pixel "$p2_i$" en la tercera fila desde el límite de línea BB del bloque BKu, la Expresión (49) de la aproximación lineal se sustituye y se usa en su lugar.

[0234]

Por otra parte, el proceso de filtración no se aplica al píxel "$p2_0$" en la tercera fila desde el límite de línea BB del bloque BKu. Por lo tanto, en la filtración fuerte del caso R2W2, se elimina una expresión correspondiente a la Expresión (54) de la filtración fuerte del esquema HEVC.

[0235]

Más adelante, se describirá una expresión para la determinación de la filtración débil para las señales de luminancia en un límite de línea y una filtración débil.

[0236]

En primer lugar, en la expresión para la determinación de la filtración débil para las señales de luminancia en el esquema HEVC y la filtración débil, los componentes de luminancia de los píxeles respectivos en el rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las expresiones (61) a (63). Los primeros términos de las expresiones (61) a (63) son expresiones de determinación de filtración débil.

[0237]

[Fórmula Matemática 17]

si (abs(delta) < iThrCut), delta = (9*(q0$_i$ − p0$_i$) − 3*(q1$_i$ − p1$_i$))

- p0$_0$ = Clip$_{0-255}$(p0$_i$ + Clip$_{(-tc)-tc}$(delta)); i = 0,7
- q0$_0$ = Clip$_{0-255}$(q0$_i$ − Clip$_{(-tc)-tc}$(delta)); i = 0,7 ···(61)

si (abs((p2$_2$ − 2*p1$_2$ + p0$_2$) + abs(p2$_5$ − 2*p1$_5$ + p0$_5$) < iSideThreshold))

- p1$_0$ = Clip$_{0-255}$(p1$_i$ + Clip$_{(-tc2)-tc2}$((((p2$_i$ + p0$_i$ + 1) >> 1) − p1$_i$ + delta) >> 1); i = 0,7 ···(62)

si (abs((q2$_2$ − 2*q1$_2$ + q0$_2$) + abs(q2$_5$ − 2*q1$_5$ + q0$_5$) < iSideThreshold))

- q1$_0$ = Clip$_{0-255}$(q1$_i$ + Clip$_{(-tc2)-tc2}$((((q2$_i$ + q0$_i$ + 1) >> 1) − q1$_i$ − delta) >> 1); i = 0,7 ···(63)

Aquí, como se describió anteriormente “Clip$_{0-255}$” indica un proceso de recorte que redondea un valor hasta “0” cuando el valor es “0” o más pequeño y redondea un valor hasta “255” cuando el valor es “255” o más grande. Por otro lado Clip$_{(tc)(tc)}$ indica un proceso de recorte que redondea un valor hasta “-tc” cuando el valor es “tc” o más pequeño y redondea hasta un valor “tc” cuando el valor es “tc” o más grande. El recorte “tc2” es el mismo. Lo mismo se aplica a las siguientes expresiones. Aquí, “tc” es un valor que se establece con base en el parámetro Q como se indica en la Tabla 1.

[0238]

En contraste, en la expresión de la determinación de filtración débil para las señales de luminancia en el caso R2W2 y en la filtración débil, los componentes de luminancia de los respectivos píxeles en el rango de procesamiento de filtro se calculan mediante la realización de las operaciones de las expresiones (64) a (66). Los primeros términos de las expresiones (64) a (66) son expresiones para la determinación de la filtración débil.

[0239]

[Fórmula Matemática 18]

si (abs(delta) < iThrCut), delta = (9*(q0$_i$ − p0$_i$) − 3*(q1$_i$ − p1$_i$))

- p0$_0$ = Clip$_{0-255}$(p0$_i$ + Clip$_{(-tc)-tc}$(delta)); i = 0,7
- q0$_0$ = Clip$_{0-255}$(q0$_i$ − Clip$_{(-tc)-tc}$(delta)); i = 0,7 ···(64)

si (abs(0 − (p1$_2$ − p0$_2$)) + abs(0 − (p1$_5$ − p0$_5$)) < iSideThreshold))

- p1$_0$ = Clip$_{0-255}$(p1$_i$ + Clip$_{(-tc2)-tc2}$((((p1$_i$ + p1$_i$ + 1) >> 1) − p1$_i$ + delta) >> 1); i = 0,7 ···(65)

si (abs((q2$_2$ − 2*q1$_2$ + q0$_2$) + abs(q2$_5$ − 2*q1$_5$ + q0$_5$) < iSideThreshold))

- q1$_0$ = Clip$_{0-255}$(q1$_i$ + Clip$_{(-tc2)-tc2}$((((q2$_i$ + q0$_i$ + 1) >> 1) − q1$_i$ − delta) >> 1); i = 0,7 ···(66)

[0240]

Aquí, una expresión dentro del recorte tc2 del segundo término que es una expresión de filtración débil dentro de la Expresión (65) del caso R2W2 es diferente de una expresión dentro del recorte tc2 del segundo término en la Expresión (62) del esquema HEVC de acuerdo con la

Expresión (49) de la aproximación lineal. Es decir, la expresión “((((p2$_i$ + p0$_i$ +1)>>1)- p1$_i$ + delta)>>1)” dentro del recorte tc2 del segundo término del esquema HEVC se cambia a la expresión “((((p1$_i$ + p1$_i$ +1)>>1)-p1$_i$ + delta)>>1)” dentro del recorte tc2 del segundo término.

[0241]

Por otra parte, una expresión dentro de la cláusula sí del primer término que es una expresión para la determinación de la filtración débil dentro de la Expresión (65) del caso R2W2 es diferente de una expresión dentro de la cláusula sí del primer término de la Expresión (62) del esquema HEVC. Es decir, la expresión “abs(p2$_2$ - 2*p1$_2$ + p0$_2$) + abs(p2$_5$ -2*p1$_5$ + p0$_5$)” dentro de la cláusula sí del primer término del esquema HEVC se cambia a la expresión “abs(0-(p1$_2$ – p0$_2$)) + abs(0-(p1$_5$ – p0$_5$))” dentro de la cláusula sí del primer término.

[0242]

Es decir, en el primer término de la Expresión (65), un píxel inmediatamente por debajo de un píxel actual se rellena como se indica en la Expresión (67) por debajo en lugar de aproximar linealmente el mismo. En otras palabras, “0” en la expresión “abs(0-(p1$_2$ – p0$_2$))” representa “0 = p2$_2$ - p1$_2$” y”0” en la expresión “abs(0-(p1$_5$ - p0$_5$)” representa “0 = p2$_5$ - p1$_5$”.

[0243]

[Fórmula Matemática 19]

$$p2_i \fallingdotseq p1_i \qquad \cdots(67)$$

Esto es debido a que la determinación de la inclinación en la expresión de la determinación de filtración, el uso de aproximación lineal permite ignorar la determinación que en realidad no se puede ignorar. Por lo tanto, en la determinación de la inclinación de la expresión de determinación de filtración débil, un píxel inmediatamente por debajo del píxel se rellena como se indica en la Expresión (67). Lo mismo se puede decir para la expresión de la determinación de filtración fuerte y la expresión de determinación del límite inter-bloque descritos más adelante.

[0244]

Más adelante, se describirán la expresión para la determinación del límite inter-bloque y una expresión para la determinación de filtración fuerte para las señales de luminancia en un límite de línea.

[0245]

En primer lugar, la expresión para la determinación del límite inter-bloque para las señales de luminancia en el esquema HEVC se expresa mediante la Expresión (68) más adelante y una expresión para la determinación de filtración fuerte se expresa mediante la Expresión (69) más adelante.

[Fórmula Matemática 20]

$$\text{si } (abs((p2_2-2*p1_2+p0_2)+abs(p2_5-2*p1_5+p0_5)+abs((q2_2-2*q1_2+q0_2)+abs((q2_5-2*q1_5+q0_5)+<\beta) \quad \cdots(68)$$

[Fórmula Matemática 21]

$$d<(\beta>>2),\quad d=abs((p2_2-2*p1_2+p0_2)+abs(p2_5-2*p1_5+p0_5)+abs((q2_2-2*q1_2+q0_2)+abs(q2_5-2*q1_5+q0_5)$$
$$\text{y } (|p3_i-p0_i|+|q0_i-q3_i|)<(\beta>>3$$
$$\text{y } |p0_i-q0_i|<((5*t_c+1)>>1) \quad \cdots(69)$$

[0246]

En contraste, una expresión de determinación del límite inter-bloque en el caso R2W2 de la presente técnica se expresa mediante la Expresión (70) más adelante y una expresión para la determinación de filtración fuerte se expresa mediante la Expresión (71) más adelante.

[Fórmula Matemática 22]

$$\text{si } (abs(0-(p1_2-p0_2))+abs(0-(p1_5-p0_5))+abs((q2_2-2*q1_2+q0_2)+abs(q2_5-2*q1_5+q0_5)<\beta) \quad \cdots(70)$$

[Fórmula Matemática 23]

$$d<(\beta>>2),\quad d=abs(0-(p1_2-p0_2))+abs(0-(p1_5-p0_5))+abs((q2_2-2*q1_2+q0_2)+abs(q2_5-2*q1_5+q0_5)$$
$$\text{y } (|(p1_i-p0_i)<<1|+|q0_i-q3_i|)<(\beta>>3)$$
$$\text{y } |p0_i-q0_i|<((5*t_c+1)>>1) \quad \cdots(71)$$

[0247]

El primero y el segundo término en la cláusula sí de la Expresión (70) para la determinación del límite inter-bloque del caso R2W2 son diferentes del primero y segundo término en la cláusula sí de la Expresión (68) para la determinación del límite inter-bloque del esquema HEVC de acuerdo con la Expresión (67) para el relleno. Es decir, el primero y segundo término “$abs(p2_2 - 2*p1_2 + p0_2) + abs(p2_5 -2*p1_5 + p0_5)$” en la cláusula sí del esquema

HEVC se cambian al primero y segundo términos “abs(0-($p1_2$ – $p0_2$)) + abs(0-($p1_5$ - $p0_5$))” en la cláusula sí.

[0248]

Por otra parte, una expresión en valor absoluto en la primera fila de la Expresión (71) para determinación de filtración fuerte del caso R2W2 es diferente de una expresión en valor absoluto en la primera fila de la Expresión (69) para la determinación de filtración fuerte en el esquema HEVC de acuerdo con la Expresión (67) para el relleno. Es decir, la expresión “abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ -2*$p1_5$ + $p0_5$)” en el valor absoluto de la primera fila de la Expresión (69) se cambia a la expresión “abs(0-($p1_2$ – $p0_2$)) + abs(0-($p1_5$ –$p0_5$))” en el valor absoluto de la primera fila de la Expresión (71).

[0249]

Además, una expresión que aparece después del primer “y” de la Expresión (71) para la determinación de la filtración fuerte del caso R2W2 es diferente de una expresión que aparece después del primer “y” de la Expresión (69) para la determinación de filtración fuerte en el esquema HEVC. Es decir, la expresión “(|$p3_i$ – $p0_i$ <<1| + |$q0_i$ –$q3_i$|)< (β>>3))” que aparece después de la primera “y” de La Expresión (69) se cambia a la expresión “(|$p1_i$ – $p0_i$ <<1| + |$q0_i$ – $q3_i$|)<(β>>3))” que aparece después de la primera “y” de la Expresión (71).

[0250]

Puesto que la determinación de esta parte es la determinación de las magnitudes de los valores de píxel, se pueden utilizar la Expresión (49) para la aproximación lineal y la Expresión (67) para el relleno. Es decir, la expresión que aparece después de la primera “y” de la Expresión (71) se genera de tal manera que “$p3_i$” en la expresión que aparece después de la primera “y” de la Expresión (69) se aproxima primero como “$p2_i$” por relleno y”$p2_i$” se aproxima como “2*$pP1_i$-$p0_i$” por aproximación lineal.

[0251]

[Proceso de filtración para el caso R2W1]

Más adelante, se describirá el proceso de filtración de las señales de luminancia para el caso R2W1. Aquí, R2 representa el número de píxeles a que hace referencia y W1 representa el número de píxeles a los que se aplica el proceso de filtración. Es decir, R2W1 representa que el proceso de filtración se aplica a un píxel en el límite de línea inter–LCU haciendo referencia a dos píxeles en el límite de línea inter-LCU como se ilustra en la fig. 19.

[0252]

En la filtración fuerte de la señal de luminancia en el caso R2W1 de la presente técnica, los componentes de luminancia de los respectivos píxeles en el rango de procesamiento de filtro se calculan por la realización de las operaciones de las Expresiones (72) a (75).

[0253]

[Fórmula Matemática 24]

$$p0_0 = Clip_{0-255}((4*p1_i+p0_i+2*q0_i+q1_i+4)>>3); \quad i=0,7 \quad \cdots(72)$$
$$q0_0 = Clip_{0-255}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad i=0,7 \quad \cdots(73)$$
$$q1_0 = Clip_{0-255}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad i=0,7 \quad \cdots(74)$$
$$q2_0 = Clip_{0-255}((p0_i+q0_i+q1_i+3*q2_i+2*q3_i+4)>>3); \quad i=0,7 \quad \cdots(75)$$

[0254]

Aquí, la Expresión (72) de la filtración fuerte del caso R2W1 es diferente en que "$p2_i$ + $2*p1_i$ +$2*p0_i$ + 0" en la Expresión (50) de la filtración fuerte del esquema HEVC se cambia a "$4*p1_i + p0_i$" de acuerdo con la Expresión (49) de aproximación lineal.

[0255]

La filtración fuerte del caso R2W1 es diferente en que se suprimen las expresiones que corresponden a las Expresiones (52) y (54) de la filtración fuerte del esquema HEVC.

[0256]

Es decir, en la presente técnica, el píxel ""$p2_i$" en la tercera fila desde el límite de línea BB del bloque BKu que no se conoce, la Expresión (49) de la aproximación lineal se sustituye y utiliza en su lugar.

[0257]

Por otra parte, el proceso de filtración no se aplica a los píxeles "$p1_0$" y"$p2_0$" en la segunda y tercera filas del límite de línea BB del bloque BKu. Por lo tanto, en la filtración fuerte del caso R2W1, se suprimen las expresiones correspondientes a las Expresiones (52) y (54) de la filtración fuerte del esquema HEVC.

[0258]

Más adelante, se describirá una expresión para la determinación de filtración débil para las señales de luminancia en el límite de línea y la filtración débil.

[0259]

En la expresión para la determinación de filtración débil para las señales de luminancia en el caso R2W1 y la filtración débil, los componentes de luminancia de los píxeles respectivos en el rango de procesamiento de filtro se calculan mediante la realización de las operaciones de la Expresiones (76) y (77).

[0260]

[Fórmula Matemática 25]

$$\begin{aligned} &\text{si } (\text{abs}(\text{delta}) < \text{iThrCut}),\ \text{delta} = (9*(q0_i - p0_i) - 3*(q1_i - p1_i)) \\ &\quad p0_0 = \text{Clip}_{0-255}(p0_i + \text{Clip}_{(-tc)-tc}(\text{delta})); \qquad i = 0,7 \\ &\quad q0_0 = \text{Clip}_{0-255}(q0_i - \text{Clip}_{(-tc)-tc}(\text{delta})); \qquad i = 0,7 \quad \cdots(76) \\ &\text{si } (\text{abs}((q2_2 - 2*q1_2 + q0_2) + \text{abs}(q2_5 - 2*q1_5 + q0_5) < \text{iSideThreshold})) \\ &\quad \cdot\ q1_0 = \text{Clip}_{0-255}(q1_i + \text{Clip}_{(-tc2)-tc2}((((q2_i + q0_i + 1) >> 1) - q1_i - \text{delta}) >> 1);\ i = 0,7 \quad \cdots(77) \end{aligned}$$

[0261]

Aquí, en el caso R2W1, el proceso de filtración no se aplica al píxel “$p1_0$” en la segunda fila a partir del límite de línea BB del bloque BKu. Por lo tanto, en la expresión para la determinación de la filtración débil para el caso R2W1 y la filtración débil, se elimina una expresión correspondiente a la Expresión (62) del esquema HEVC.

[0262]

Más adelante, se describirá una expresión para la determinación del límite inter-bloque y una expresión para la determinación de la filtración fuerte para las señales de luminancia en un límite de línea.

[0263]

Una expresión para la determinación del límite inter-bloque en el caso R2W1 se expresa mediante la Expresión (78) más adelante y una expresión para la determinación de la filtración fuerte se expresa mediante la Expresión (79) más adelante.

[Fórmula Matemática 26]

$$\begin{aligned} &\text{si } (\text{abs}(0 - (p1_2 - p0_2)) + \text{abs}(0 - (p1_5 - p0_5)) + \text{abs}((q2_2 - 2*q1_2 + q0_2) \\ &+ \text{abs}(q2_5 - 2*q1_5 + q0_5) < \beta) \qquad \cdots(78) \end{aligned}$$

[Fórmula Matemática 27]

$$d < (\beta >> 2), \quad d = abs(0-(p1_2-p0_2))+abs(0-(p1_5-p0_5))+ abs((q2_2-2*q1_2+q0_2)+abs(q2_5-2*q1_5+q0_5)$$
$$y\ (|(p1_i-p0_i) << 1|+|q0_i-q3_i|) < (\beta >> 3)$$
$$y\ |p0_i-q0_i| < ((5*t_c+1) >> 1) \quad \cdots(79)$$

[0264]

El primero y segundo términos en la cláusula sí de La Expresión (78) para la determinación del límite inter-bloque del caso R2W1 son diferentes del primero y segundo términos en la cláusula sí de la Expresión (68) para la determinación del límite inter-bloque del esquema HEVC de acuerdo con la Expresión (67) para el relleno. Es decir, el primero y segundo términos “abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ -2*$p1_5$ + $p0_5$)” en la cláusula sí del esquema HEVC se cambian al primero y segundo términos “abs(0-($p1_2$ – $p0_2$))” + abs(0-($p1_5$ - $p0_5$))” en la cláusula sí

[0265]

Por otra parte, una expresión en el valor absoluto en la primera fila de la Expresión (79) para la determinación de la filtración fuerte del caso R2W1 es diferente de una expresión en el valor absoluto de la primera fila de la Expresión (69) para la determinación de la filtración fuerte en el esquema HEVC de acuerdo con la Expresión (67) para el relleno. Es decir, la expresión “abs($p2_2$ - 2*$p1_2$ + $p0_2$) + abs($p2_5$ -2*$p1_5$ + $p0_5$)” en el valor absoluto de la primera fila de la Expresión (69) se cambia a la expresión “abs(0-($p1_2$ – $p0_2$)) + abs(0-($p1_5$ - $p0_5$))” en el valor absoluto de la primera fila de la Expresión (79) .

[0266]

Además, una expresión que aparece después de la primera “y” de la Expresión (79) para la determinación de la filtración fuerte del caso R2W1 es diferente de una expresión que aparece después de la primera “y” de la Expresión (69) para la determinación de la filtración fuerte en el esquema HEVC. Es decir, la expresión “(|$p3_i$ – $p0_i$ <<1| + |$q0_i$ – $q3_i$|)< (β>>3))”que aparece después de la primera “y” de La Expresión (69) se cambia a la expresión “(|$p1_i$ – $p0_i$ <<1| + |$q0_i$ – $q3_i$|)<(β>>3))” que aparece después de la primera “y” de la Expresión (79) .

[0267]

Como se describió anteriormente en relación con el caso R2W2, puesto que la determinación de esta parte es la determinación de las magnitudes de los valores de píxel, se utiliza la Expresión (49) para la aproximación lineal y la Expresión (67) para el relleno. Es decir, la expresión que aparece después de la primera “y” de la Expresión (79) se genera de tal manera que “$p3_i$” en la expresión que aparece después de la primera “y” de la Expresión (69) primero se aproxima como “$p2_i$” y “$p2_i$” se aproxima como “2*$p1_i$ – $p0_i$”.

[0268]

Con referencia a las figuras 22 a 24 se describirá un ejemplo de configuración y una operación para realizar los procesos de determinación anteriores con base en aproximación lineal, como la sexta y séptima implementaciones de la unidad de filtración de desbloqueo.

[0269]

Como se describió anteriormente, en la presente técnica, la determinación del límite inter-bloque en el límite de línea y la expresión para la determinación de filtración fuerte se cambian utilizando la Expresión (49) para la aproximación lineal y la Expresión (67) para el relleno. Por lo tanto, se agrega un proceso de recorte a un proceso de filtración fuerte como una contramedida a un caso en el que la Expresión de la determinación de ignora erróneamente y el proceso de filtración de desbloqueo se realiza erróneamente.

[0270]

[Descripción de la presente técnica (Proceso de Recorte en la Filtración Fuerte)]

Más adelante, se describirá un proceso de recorte en la filtración fuerte que es una octava implementación. La filtración fuerte para las señales luminancia en el caso R2W2 también puede utilizar las expresiones (80) a (84) en las cuales se agrega el proceso de recorte.

[Fórmula Matemática 28]

$$p0_0 = Clip_{0-255}(p0_0 + Clip_{(-tc)-tc}(((4*p1_i + p0_i + 2*q0_i + q1_i + 4) >> 3) - p0_0)); \quad i = 0,7 \quad \cdots(80)$$

$$q0_0 = Clip_{0-255}(p0_0 + Clip_{(-tc)-tc}(((p1_i + 2*p0_i + 2*q0_i + 2*q1_i + q2_i + 4) >> 3) - p0_0)); \quad i = 0,7 \quad \cdots(81)$$

$$p1_0 = Clip_{0-255}(p1_0 + Clip_{(-tc)-tc}(((3*p1_i + q0_i + 2) >> 2) - p1_0)); \quad i = 0,7 \quad \cdots(82)$$

$$q1_0 = Clip_{0-255}(q1_0 + Clip_{(-tc)-tc}(((p0_i + q0_i + q1_i + q2_i + 2) >> 2) - q1_0)); \quad i = 0,7 \quad \cdots(83)$$

$$q2_0 = Clip_{0-255}(q2_0 + Clip_{(-tc)-tc}(((p1_i + q0_i + q1_i + 3*q2_i + 2*q3_i + 4) >> 3) - q2_0)); \quad i = 0,7 \quad \cdots(84)$$

[0271]

La Expresión (80) para la filtración fuerte es una expresión del cálculo para $p0_0$ después de la filtración. En la Expresión (80) para la filtración fuerte, se aplica un proceso de recorte de (-tc) a tc a una diferencia entre la expresión en el proceso de recorte de 0 a 255 en la Expresión (56) y $p0_0$ para obtener un valor, el valor se agrega a $p0_0$ para obtener otro valor, y el otro valor se somete a un proceso de recorte de 0 a 255.

[0272]

La Expresión (81) para la filtración fuerte es una expresión de cálculo para $q0_0$ después de la filtración. En la Expresión (81) para la filtración fuerte, se aplica un proceso de recorte de

(-tc) a tc a una diferencia entre la expresión en el proceso de recorte de 0 a 255 en la Expresión (57) y $q0_0$ para obtener un valor, el valor se agrega a $q0_0$ para obtener otro valor, y el otro valor se somete a un proceso de recorte de 0 a 255.

[0273]

La Expresión (82) para la filtración fuerte es una expresión de cálculo para $p1_0$ después de la filtración. En la Expresión (82) para la filtración fuerte, se aplica un proceso de recorte de (-tc) a tc a una diferencia entre la expresión en el proceso de recorte de 0 a 255 en la Expresión (58) y $p1_0$ para obtener un valor, el valor se agrega a $p1_0$ para obtener otro valor, y el otro valor se somete a un proceso de recorte de 0 a 255.

[0274]

La Expresión (83) para la filtración fuerte es una expresión de cálculo para $q1_0$ después de la filtración. En la Expresión (83) para la filtración fuerte, se aplica un proceso de recorte de (-tc) a tc a una diferencia entre la expresión en el proceso de recorte de 0 a 255 en la Expresión (59) y $q1_0$ para obtener un valor, el valor se agrega a $q1_0$ para obtener otro valor, y el otro valor se somete a un proceso de recorte de 0 a 255.

[0275]

La Expresión (84) para la filtración fuerte es una expresión de cálculo para $q2_0$ después de la filtración. En la Expresión (84) para la filtración fuerte, se aplica un proceso de recorte de (-tc) a tc a una diferencia entre la expresión en el proceso de recorte de 0 a 255 en la Expresión (60) y $q2_0$ para obtener un valor, el valor se agrega a $q2_0$ para obtener otro valor, y el otro valor se somete a un proceso de recorte de 0 a 255..

[0276]

De esta manera, es posible suprimir la aplicación excesiva del proceso de filtración.

[0277]

Aquí, se propone Matthias Narroschke, Thomas Wedi, Semih Esenlik, “Results for modified decisions for deblocking” “JCTVC-G590, Joint Collaborative Team on Video coding (JCT-VC) del ITU-T SG16 WP3 e ISO/IEC JTC1/SC29/WG11 séptima reunión, Ginebra, CH, 21 - 30 Noviembre, 2011 (de aquí en adelante, referido como Documento 2 de no Patente).

[0278]

En el Documento 2 de no Patente, se describe que un proceso para la determinación de filtración fuerte/débil se realiza en las unidades de bloques respectivas. Sin embargo, en el proceso de determinación descrito en el Documento 2 de no Patente, cuando los resultados de la determinación en una línea actual y una unidad de bloque a la cual la línea pertenece son diferentes (por ejemplo, cuando la filtración es débil se determina para la línea actual mientras que la filtración fuerte se determina para un bloque que determina la línea, se aplica la información del artículo a la línea actual), se aplica un proceso de filtración excesivo.

[0279]

Por otra parte, en el Documento 2 de no patente, puesto que proceso de determinación de filtración fuerte/débil se realiza en unidades de cuatro líneas, no hay correspondencia entre la determinación y el proceso de filtración. Como un ejemplo de la falta de correspondencia, la filtración débil se determina aunque sea necesario determinar la filtración fuerte y la filtración fuerte se determina aunque sea necesario determinar la filtración débil.

[0280]

En la filtración fuerte de señales de luminancia en el esquema HEVC (HM- 5.0), se emplean las Expresiones (85) a (89) siguientes.

[Fórmula Matemática 29]

$$p0_i = (p2_i + 2*p1_i + 2*p0_i + 2*q0_i + q1_i + 4) >> 3; \quad i = 0,7 \quad \cdots(85)$$
$$q0_i = (p1_i + 2*p0_i + 2*q0_i + 2*q1_i + q2_i + 4) >> 3; \quad i = 0,7 \quad \cdots(86)$$
$$p1_i = (3*p1_i + q0_i + 2) >> 2; \quad i = 0,7 \quad \cdots(87)$$
$$q1_i = (p0_i + q0_i + q1_i + q2_i + 2) >> 2; \quad i = 0,7 \quad \cdots(88)$$
$$q2_i = (p0_i + q0_i + q1_i + 3*q2_i + 2*q3_i + 4) >> 3; \quad i = 0,7 \quad \cdots(89)$$

[0281]

En la presente técnica, con el propósito de eliminar una falta de correspondencia del proceso de determinación y una incompletitud en el proceso de determinación, se indica, el proceso de recorte de las Expresiones (90) a (94) más adelante. En La Expresión (90), puesto que el proceso de recorte se va a realizar en un valor-Δ parcial que cambia con el proceso de filtración, $p0_i$ está fuera del proceso de recorte. Lo mismo se aplica a las Expresiones (91) a (94).

[Fórmula Matemática 30]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3); \quad i = 0,7 \quad \cdots (90)$$
$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3); \quad i = 0,7 \quad \cdots (91)$$
$$p1_i = p1_i + Clip_{(-pv)-(pv)}((-p1_i + q0_i + 2) >> 2); \quad i = 0,7 \quad \cdots (92)$$
$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2); \quad i = 0,7 \quad \cdots (93)$$
$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + q1_i - 5*q2_i + 2*q3_i + 4) >> 3); \quad i = 0,7 \quad \cdots (94)$$

En este caso, un valor de corte “pv” es cualquiera de 1 vez, 2 veces, 3 veces, 4 veces, 5 veces, 6 veces, 7 veces y 8 veces del parámetro tc. Es decir, el pv = tc, 2*tc, 3*tc, 4*tc, 5*tc, 6*tc, 7*tc 8*tc. Aunque el valor de recorte “pv” preferiblemente es 2 veces el parámetro tc, el valor de recorte puede ser cualquiera de 1 vez a 8 veces del parámetro. Por otra parte, el valor de recorte no se limita a 1vez hasta 8 veces siempre que se obtengan efectos iguales.

[0282]

Por otra parte, “tc” cambia con el parámetro de cuantización QP. Por lo tanto, el factor de multiplicación del valor de recorte aplicado a “tc” también se puede aumentar (es decir, se cambia con QP) en la medida que aumenta QP. El parámetro (el valor de recorte o el valor de un factor de multiplicación del valor de recorte aplicado a “tc”) aplicado al proceso de recorte se puede establecer por adelantado y se puede agregar a una corriente codificada y al terminador de transmisión a un lado de decodificación.

[0283]

Las Expresiones (90) a (94) descritas anteriormente se pueden modificar de tal manera que el proceso de recorte se realice como en las Expresiones (95) a (99) más adelante.

[Fórmula Matemática 31]

$$p0_i = Clip_{0-255}(p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3)); \quad i = 0,7 \quad \cdots (95)$$
$$q0_i = Clip_{0-255}(q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3)); \quad i = 0,7 \quad \cdots (96)$$
$$p1_i = Clip_{0-255}(p1_i + Clip_{(-pv)-(pv)}((-p1_i + q0_i + 2) >> 2)); \quad i = 0,7 \quad \cdots (97)$$
$$q1_i = Clip_{0-255}(q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2)); \quad i = 0,7 \quad \cdots (98)$$
$$q2_i = Clip_{0-255}(q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + q1_i - 5*q2_i + 2*q3_i + 4) >> 3)); \quad i = 0,7 \quad \cdots (99)$$

[0284]

Aquí, las Expresiones (85) a (89) para la filtración fuerte de las señales de luminancia del esquema HEVC (HM-5.0) son expresiones para filtración fuerte de las señales de luminancia en un límite de línea. Es decir, en el esquema HEVC (HM-5.0), la filtración fuerte de las señales de luminancia se expresa realmente mediante las Expresiones (100) a (105) más adelante.

[Fórmula Matemática 32]

$$p0_i = (p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3; \quad i = 0,7 \quad \cdots (100)$$

$$q0_i = (p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3; \quad i = 0,7 \quad \cdots (101)$$

$$p1_i = (p2_i+p1_i+p0_i+q0_i+2)>>2; \quad i = 0,7 \quad \cdots (102)$$

$$q1_i = (p0_i+q0_i+q1_i+q2_i+2)>>2; \quad i = 0,7 \quad \cdots (103)$$

$$p2_i = (2*p3_i+3*p2_i+p1_i+p0_i+q0_i+4)>>3; \quad i = 0,7 \quad \cdots (104)$$

$$q2_i = (p0_i+q0_i+p1_i+3*q2_i+2*q3_i+4)>>3; \quad i = 0,7 \quad \cdots (105)$$

[0285]

En la presente técnica, con el propósito de eliminar una falta de correspondencia del proceso de determinación y una incompletitud en el proceso de determinación, se puede aplicar un proceso de recorte a la filtración fuerte para las señales de luminancia como se indica en las Expresiones (106) a (111) a fin de que correspondan a las Expresiones (100) a (105). En la primera Expresión de la Expresión (106), de manera similar a las Expresiones (90) a (94) descritas anteriormente puesto que se realiza un proceso de recorte sobre el valor-Δ parcial que cambia con el proceso de filtración, $p0_i$ está fuera del proceso de recorte. Lo mismo se aplica a la primera expresión de las Expresiones (107) a (111) .

[0286]

[Fórmula Matemática 33]

$$\begin{aligned} p0_i &= p0_i+Clip_{(-pv)-(pv)}((p2_i+2*p1_i-6*p0_i+2*q0_i+q1_i+4)>>3); \quad i = 0,7 \\ &= Clip_{(-pv)-(pv)}((p2_i+2*p1_i+2*p0_i+2*q0_i+q1_i+4)>>3); \quad \cdots (106) \end{aligned}$$

$$\begin{aligned} q0_i &= q0_i+Clip_{(-pv)-(pv)}((p1_i+2*p0_i-6*q0_i+2*q1_i+q2_i+4)>>3); \quad i = 0,7 \\ &= Clip_{(-pv)-(pv)}((p1_i+2*p0_i+2*q0_i+2*q1_i+q2_i+4)>>3); \quad \cdots (107) \end{aligned}$$

$$\begin{aligned} p1_i &= p1_i+Clip_{(-pv)-(pv)}((p2_i-3*p1_i+p0_i+q0_i+2)>>2); \quad i = 0,7 \\ &= Clip_{(-pv)-(pv)}((p2_i+p1_i+p0_i+q0_i+2)>>2); \quad \cdots (108) \end{aligned}$$

$$\begin{aligned} q1_i &= q1_i+Clip_{(-pv)-(pv)}((p0_i+q0_i-3*q1_i+q2_i+2)>>2); \quad i = 0,7 \\ &= Clip_{(-pv)-(pv)}((p0_i+q0_i+q1_i+q2_i+2)>>2); \quad \cdots (109) \end{aligned}$$

$$\begin{aligned} p2_i &= p2_i+Clip_{(-pv)-(pv)}((2*p2_i-5*p2_i+p1_i+p0_i+q0_i+4)>>3); \quad i = 0,7 \\ &= Clip_{(-pv)-(pv)}((2*p3_i+3*p2_i+p1_i+p0_i+q0_i+4)>>3); \quad \cdots (110) \end{aligned}$$

$$\begin{aligned} q2_i &= q2_i+Clip_{(-pv)-(pv)}((p0_i+q0_i+p1_i-5*q2_i+2*q3_i+4)>>3); \quad i = 0,7 \\ &= Clip_{(-pv)-(pv)}((p0_i+q0_i+p1_i+3*q2_i+2*q3_i+4)>>3); \quad \cdots (111) \end{aligned}$$

En este caso, un valor de corte “pv” es cualquiera de 1 vez, 2 veces, 3 veces, 4 veces, 5 veces, 6 veces, 7 veces y 8 veces del parámetro tc. Es decir, el pv = tc, 2*tc, 3*tc, 4*tc, 5*tc, 6*tc, 7*tc, 8*tc. Aunque el valor de recorte “pv” preferiblemente es 2 veces el parámetro tc, el valor de recorte puede ser cualquiera de 1 vez a 8 veces del parámetro. Por otra parte, el valor de recorte no se limita a 1vez hasta 8 veces siempre que se obtengan efectos iguales.

[0287]

Por otra parte, “tc” cambia con el parámetro de cuantización QP. Por lo tanto, el factor de multiplicación del valor de recorte aplicado a “tc” también se puede aumentar (es decir, se cambia con QP) en la medida que aumenta QP. El parámetro (el valor de recorte o el valor de un factor de multiplicación del valor de recorte aplicado a “tc”) aplicado al proceso de recorte se puede establecer por adelantado y se puede agregar a una corriente codificada y al terminador de transmisión a un lado de decodificación.

[0288]

La figura 20 es un diagrama que ilustra los valores más grandes de los valores de píxel que cambian con la filtración fuerte de las señales de luminancia en el esquema HEVC (HM-5.0) de las Expresiones (100) a (105) descritas anteriormente, obtenidas a través de un experimento

[0289]

Al_HE, RA_HE, LB_HE y LP_HE indican las condiciones de este experimento AI_HE indica que el experimento se realizó en el modo de alta eficiencia TODO lNTRA. RA_HE indica que el experimento se realizó en el modo de eficiencia de alto acceso al AZAR. LB_HE indica que el experimento se realizó en modo de alta eficiencia bajo retardo B. LP_HE indica que el experimento se realizó en el modo de alta eficiencia bajo retardo P.

[0290]

“22, 27, 32 y 37” que aparecen bajo AI_HE, RA_HE, LB_HE, y LP_HE son los valores de los parámetros de cuantización usados en el experimento. Por otra parte, la Clase A a la Clase F indican el tipo de secuencia de prueba utilizada en el experimento.

[0291]

Como se ilustra en la figura 20, un valor de píxel que cambia con la filtración fuerte de las señales de luminancia en el esquema HEVC (HM- 5.0) (es decir, un valor de píxel que cambia con la filtración fuerte cuando el proceso de recorte no está presente) cambia ampliamente por 100 o más en algunos casos.

[0292]

Por lo tanto, se aplica un proceso de recorte a la filtración fuerte de las señales de luminancia. Debido a esto, como se ilustra en la figura 21, es posible suprimir la influencia de un desajuste del proceso de determinación y una incompletitud del proceso de determinación tanto como es posible.

[0293]

En el ejemplo de la figura 21 cuando hay un borde (que no es un límite de bloque) indicado mediante una línea continua, aunque el proceso de desbloqueo no se vaya a realizar, si el proceso de desbloqueo se realiza de acuerdo con la técnica del Documento 2 de no patente, los valores de los píxel se cambian ampliamente como se indica por las líneas de puntos.

[0294]

Por ejemplo, como se indica por la línea continua, antes que se realice el proceso de desbloqueo, los valores de los píxeles $p2_i$, $p1_i$, y $p0_i$ son 255, los valores de los pixeles $q0_i$, $q1_i$ y $q2_i$ son O, y la diferencia D entre los valores de píxeles $p0_i$ y $q0_i$ es 255.

[0295]

En contraste, como se indica por la línea de puntos, después que se realiza el proceso de desbloqueo, el valor del píxel $p2_i$ es 223, el valor del píxel $p1_i$ es 191, y el valor del pixel $p0_i$ es 159. Por lo tanto, los valores de pixel de los píxeles $p2_i$, $p1_i$ y $p0_i$ se cambian ampliamente.

Por otra parte, después que se realiza el proceso de desbloqueo, el valor del píxel $q0_i$ es 96, el valor del píxel $q1_i$ es 64, y el valor del píxel $q2_i$ es 32. Por lo tanto, los valores de pixel de los píxeles $q0_i$, $q1_i$ y $q2_i$ se cambian ampliamente.

[0296]

En tal caso, en la filtración fuerte de las Expresiones (106) a (111) descritas anteriormente, se realiza un proceso de recorte del cual el valor de recorte es 10.

[0297]

En contraste, como se indica por la línea en negrita, después que se realiza el proceso de desbloqueo, los valores de los píxeles $p2_i$, $p1_i$ y $p0_i$ son 245, y los valores de los pixeles $q0_i$, $q1_i$ y $q2_i$ son 255. Por lo tanto, el cambio en el valor de pixel se puede suprimir tanto como es posible.

[0298]

Un ejemplo de configuración y una operación del proceso de recorte en la filtración fuerte se describirán con referencia a las Figs. 25 al 27 descritas más adelante como una octava implementación de la unidad de filtración de desbloqueo.

[0299]

[13. Sexta implementación de la unidad de filtración de desbloqueo] [Ejemplo de configuración de la unidad de filtración de desbloqueo]

La figura 22 ilustra la configuración de la sexta implementación de la unidad de filtración de desbloqueo. La unidad de filtración de desbloqueo 24 está configurada para incluir una memoria de imagen 71, una unidad que determina el límite de bloque 72-1, una unidad que determina la fuerza del filtro 73-1, una unidad de operación de filtro 74-1, un selector 75, una memoria de coeficiente 76, y un controlador de filtro 77. La unidad de filtración de desbloqueo 24 está configurada además para incluir una unidad que determina el límite de línea/límite de bloque 72-2, una unidad que determina la fuerza del filtro del límite de línea 73-2, y una unidad de operación del filtro del límite de la línea 74-2.

[0300]

La memoria de imagen 71 es una unidad que corresponde a la memoria de línea 241 de la fig. 9 y está configurada como una memoria de línea. La memoria de imagen 71 almacena los datos de imagen suministrados desde el sumador 23. La memoria de imagen 71 lee los datos de imagen almacenados en la misma y suministra los datos de imagen leídos a la unidad que determina el límite de bloque 72-1, la unidad que determina la fuerza del filtro 73-1 y la unidad de operación de filtro 74-1. Por otra parte, la memoria de imagen 71 lee los datos de imagen almacenados en ella y proporciona los datos de imagen leídos a la unidad que determina el límite de línea/límite de bloque 72-2, la unidad que determina la fuerza del filtro del límite de línea 73-2 y la unidad de operación del filtro del límite de línea 74-2.

[0301]

En posiciones diferentes al límite de línea, los datos de imagen no se almacenan en la memoria de imagen 71, pero puede ser el caso cuando los datos de imagen suministrados desde el sumador 23 se suministran a las unidades respectivas y se procesan. Sin embargo, en el ejemplo de la figura 22, por razones de conveniencia, se describe que los datos de imagen que tienen que pasar a través de la memoria de imagen 71 se procesen.

[0302]

La unidad que determina el límite de bloque 72-1 realiza la determinación del límite inter-bloque bajo el control del controlador 77. Es decir, la unidad que determina el límite de bloque 72-1 realiza el proceso de determinación del límite inter–bloque para cada bloque utilizando los datos de imagen leídos desde la memoria de imagen 71 y detecta un límite inter-bloque en el que se realiza el proceso de filtración. La unidad que determina el límite de bloque 72-1 envía los resultados de la detección a la unidad que determina la fuerza de filtro 73-1.

[0303]

La unidad que determina la fuerza del filtro 73-1 determina la fuerza del filtro de la manera descrita anteriormente bajo el control del controlador 77. Es decir, la unidad que determina la fuerza del filtro 73-1 determina si se realizará el proceso de filtración en un modo de filtración fuerte o en un modo de filtración débil para cada línea utilizando los datos de imagen de dos bloques con un límite de bloque interpuesto, suministrados desde la memoria de imagen 71 y envía los resultados de la determinación a la unidad de operación de filtro 74-1.

[0304]

La unidad de operación del filtro 74-1 realiza una operación de filtración con la fuerza de filtro determinada por la unidad que determina la fuerza de filtro 73-1 para cada línea bajo el control del controlador 77 utilizando los datos de imagen almacenados en la memoria de imagen 71 y los coeficientes de filtro leídos desde la memoria de coeficientes 76. La unidad de operación de filtro 74-1 envía de los datos de imagen que han sido sometidos al proceso de filtración al selector 75.

[0305]

Cuando el límite es un límite de línea inter-LCU, la unidad que determina el límite de línea/límite de bloque 72-2 realiza una determinación del límite inter-bloque bajo el control del controlador 77. Es decir, la unidad que determina el límite de línea/límite de bloque 72-2 realiza el proceso de determinación del límite inter-bloque para cada bloque usando los datos de imagen leídos desde la memoria de imagen 71 y detecta un límite inter-bloque en el que se realiza el proceso de filtración. La unidad que determina el límite de línea/límite de bloque 72-2 envía los resultados de la detección a la unidad que determina la fuerza del filtro del límite de línea 73-2.

[0306]

Cuando el límite es un límite de línea inter-LCU, la unidad que determina la fuerza del filtro del límite de línea 73-2 determina la fuerza del filtro de la manera descrita anteriormente bajo el control del controlador 77. Es decir, la unidad que determina la fuerza del filtro del límite de línea 73-2 determina si el proceso de filtración se realizará en un modo de filtración fuerte o en un modo de filtración débil para cada línea usando los datos de imagen de dos bloques adyacentes con un límite inter-bloque interpuesto, suministrado desde la memoria de imagen 71 y envía los resultados de la determinación a la unidad de operación del filtro del límite de línea 74-2.

[0307]

La unidad de operación del filtro del límite de línea 74-2 realiza la operación de filtración con la fuerza del filtro Determinada por la unidad que determina la fuerza del filtro del límite de

línea 73-2 para cada línea bajo el control del controlador 77 utilizando los datos de imagen almacenados en la memoria de imagen 71 y los coeficientes de filtro leídos desde la memoria de coeficientes 76. La unidad de operación del filtro del límite de línea 74-2 envía los datos de imagen que han sido sometidos al proceso de filtración al selector 75.

[0308]

El selector 75 selecciona cualquiera de los datos de imagen de la unidad de operación del filtro 74-1 y los datos de imagen desde la unidad de operación del filtro del límite de línea 74-2 bajo el control del controlador 77 y envía los datos de imagen seleccionados a la memoria de cuadro 25.

[0309]

La memoria de coeficientes 76 almacena los coeficientes de filtro usados para la operación de filtración del proceso de filtración de desbloqueo. La memoria de coeficientes 76 lee los coeficientes de filtro almacenados en ella y suministra los coeficientes de filtro leídos a la unidad de operación del filtro 74-1 y a la unidad de operación del filtro del límite de línea 74-2.

[0310]

El controlador 77 controla las respectivas unidades de la unidad de filtración de desbloqueo 24. Por ejemplo, el controlador 77 controla la memoria de imagen 71 a fin de almacenar los datos de imagen correspondientes a un número predeterminado de líneas en el lado inferior en el bloque y lee los datos de imagen almacenados en la memoria de imagen 71. El controlador 77 está configurado para incluir una unidad que determina la línea intra-LCU 77A.

[0311]

La unidad que determina la línea intra-LCU 77A determina si un límite es un límite de línea inter-LCU en las unidades de bloque respectivas en las que se realiza secuencialmente el proceso en una dirección de exploración de trama, y suministra los resultados de la determinación a la unidad que determina el límite de bloque 72-1, la unidad que determina la fuerza del filtro 73-1, y la unidad de operación del filtro 74-1 de tal manera que las unidades realicen el proceso cuando el límite no es el límite de línea. Por otra parte, la unidad que determina la línea intra-LCU 77A suministra también los resultados de la determinación a la unidad que determina el límite de línea/límite de bloque 72-2, la unidad que determina la fuerza del filtro del límite de línea 73-2, y la unidad de operación del filtro del límite de línea 74-2 de tal manera que las unidades realizan el proceso cuando el límite es el límite de línea. Además, la unidad que determina la línea intra-LCU 77A suministra los resultados de la determinación al selector 75 de modo que el selector 75 selecciona los datos de imagen desde la unidad de operación del filtro del límite de línea 74-2 cuando el límite es el límite de línea y selecciona los

datos de imagen a partir de la unidad de operación del filtro 74-1 cuando el límite no es un límite de línea.

[0312]

La unidad de filtración de desbloqueo 24 ilustrada en la figura 22 se puede configurar por adelantado para realizar el proceso R2W2, por ejemplo, y se puede configurar de tal manera que pueda controlar el tango de procesamiento del filtro y el rango de la operación de filtración de manera similar al ejemplo de la figura 9.

[0313]

[Operación de la unidad de filtración de desbloqueo]

A continuación, se describirá el proceso de desbloqueo de la unidad de filtración de desbloqueo 24 de la figura 22 con referencia al diagrama de flujo de la figura 23. En el ejemplo de la figura 23, se describe un caso en el que se realiza el proceso R2W2.

[0314]

En la etapa S91, la unidad que determina la línea intra-LCU 77A determina si el filtro aplicado es un filtro vertical del límite LCU en las respectivas unidades de bloque en las que se realiza el proceso secuencialmente en una dirección de escaneo de trama. Es decir, en la etapa S91, se determina si un límite es un límite de línea inter-LCU y el filtro aplicado es un filtro vertical.

[0315]

Cuando se determina en la etapa S91 que el filtro aplicado es un filtro vertical de un límite LCU, el flujo procede a la etapa S92. En la etapa S92, la unidad que determina el límite de línea/límite de bloque 72-2 realiza la determinación del límite inter-bloque de un límite de línea bajo el control del controlador 77 cuando el límite es un límite de línea inter-LCU.

[0316]

Es decir, la unidad que determina el límite de línea/límite de bloque 72-2 realiza el proceso de determinación del límite inter-bloque para cada bloque de acuerdo con la Expresión (70) descrita anteriormente utilizando los datos de imagen leídos desde la memoria de imagen 71 y detecta un límite inter–bloque en el que se realiza el proceso de filtración. La unidad que determina el límite de línea/límite de bloque 72-2 envía los resultados de la detección a la unidad que determina la fuerza del filtro del límite de línea 73-2.

[0317]

En la etapa S93, la unidad que determina la fuerza del filtro del límite de línea 73-2 determina la fuerza del filtro del límite de línea como se describió anteriormente bajo el control del controlador 77 cuando el límite es el límite de línea inter-LCU.

[0318]

Es decir, la unidad que determina la fuerza del filtro del límite de línea 73-2 determina si el proceso de filtración se realizará en un modo de filtración fuerte o en un modo de filtración débil usando los datos de imagen de dos bloques adyacentes con el límite inter-bloque, suministrados desde la memoria de imagen 71. Se determina un modo de filtración fuerte de acuerdo con la Expresión (71) descrita anteriormente. Por otra parte, se determina un modo de filtración débil de acuerdo con los primeros términos de las Expresiones (64) a (66) descritas anteriormente. La unidad que determina la fuerza del filtro del límite de línea 73-2 envía los resultados de la determinación a la unidad de operación del filtro del límite de línea 74-2.

[0319]

En la etapa S94, la unidad de operación del filtro del límite de línea 74-2 realiza un proceso de filtración del límite de línea bajo el control del controlador 77. Es decir, la unidad de operación del filtro del límite de línea 74-2 realiza una operación de filtración con la fuerza del filtro determinada por la unidad que determina la fuerza del filtro del límite de línea 73-2 utilizando los datos de imagen almacenados en la memoria de imagen 71 y el coeficiente de filtro leído a partir de la memoria de coeficientes 76.

[0320]

En el caso de la filtración fuerte, la operación de filtración se realiza de acuerdo con las Expresiones (56) a (60) descritas anteriormente. Por otra parte, en el caso de la filtración débil, la operación de filtración se realiza de acuerdo con el segundo y tercer términos de las Expresiones (64) a (66) descritas anteriormente. La unidad de operación del filtro del límite de línea 74-2 envía los datos de imagen que han sido sometidos al proceso de filtración al selector 75.

[0321]

En la etapa S95, la unidad que determina la línea intra-LCU 77A determina si una línea actual es la última línea octava de la LCU. Cuando se determina en la etapa S95 que la línea actual no es la última línea octava de la LCU, el flujo vuelve a la etapa S93 y el proceso se repite posteriormente.

[0322]

Por otra parte, cuando se determina en la etapa S95 que la línea actual es la última línea octava de la LCU, el flujo procede a la etapa S100.

[0323]

Por otro lado, cuando se determina en la etapa S91 que el filtro aplicado no es el filtro vertical del límite LCU, el flujo procede a la etapa S96. En la Etapa S96, la unidad que determina el límite de bloque 72-1 realiza la determinación del límite inter-bloque bajo el control del controlador 77.

[0324]

Es decir, la unidad que determina el límite de bloque 72-1 realiza el proceso de determinación del límite inter-bloque para cada bloque de acuerdo con la Expresión (68) descrita anteriormente usando los datos de imagen leídos de la memoria de imagen 71 y detecta un límite inter-bloque en el que se realiza el proceso de filtración. La unidad que determina el límite de bloque 72-1 envía los resultados a la unidad que determina la fuerza del filtro 73-1.

[0325]

En la etapa S97, la unidad que determina la fuerza del filtro 73-1 determina la fuerza del filtro de la manera descrita anteriormente bajo el control del controlador 77. Es decir, la unidad que determina la fuerza del filtro 73-1 determina si el proceso de filtración se realizará en un modo de filtración fuerte o en un modo de filtración débil para cada línea utilizando los datos de imagen de dos bloques adyacentes al límite inter-bloque interpuesto, suministrado desde la memoria de imagen 71. Un modo de filtración fuerte se determina de acuerdo con la Expresión (69) descrita anteriormente. Un modo de filtración débil se determina de acuerdo al primer término de las expresiones (61) a (63) descritas anteriormente. La unidad que determina la fuerza del filtro 73-1 envía los resultados a la unidad de operación del filtro 74-1.

[0326]

En la etapa S98, la unidad de operación del filtro 74-1 realiza la operación de filtración del límite de bloque bajo el control del controlador 77. Es decir, la unidad de operación del filtro 74-1 realiza la operación de filtración con la fuerza del filtro determinada por la unidad que determina la fuerza del filtro 73-1 para cada línea utilizando los datos de imagen almacenados en la memoria de imagen 71 y los coeficientes de filtro leídos de la memoria de coeficientes 76.

[0327]

En el caso de la filtración fuerte, la operación de filtración se realiza de acuerdo con las Expresiones (50) a (55) descritas anteriormente. Por otra parte, en el caso de la filtración débil,

la operación de filtración se realiza de acuerdo con el segundo y el tercer términos de las Expresiones (61) a (63) descritas anteriormente. La unidad de operación del filtro 74-1 envía los datos de imagen que han sido sometidos al proceso de filtración al selector 75.

[0328]

En la etapa S99, la unidad que determina la línea intra-LCU 77A determina si una línea actual es la última línea octava de la LCU. Cuando se determina en la etapa S99 que la línea actual no es la última línea octava de la LCU, el flujo vuelve a la etapa S97 y se realizan repetidamente los procesos posteriores.

[0329]

Por otra parte, cuando se determina en la etapa S99 que la línea actual es la última línea octava de la LCU, el flujo procede a la etapa S100.

[0330]

En la etapa S100, la unidad que determina la línea intra-LCU 77A determina si un bloque actual es el último bloque de la imagen. Es decir, en la etapa S100, se determina si el límite es un límite de línea de otro bloque en la pantalla.

[0331]

Cuando se determina en la etapa S100 que el bloque es el último bloque de la imagen, termina el proceso de desbloqueo. Cuando se determina en la etapa S100 que el bloque no es el último bloque de la imagen, el flujo vuelve a la etapa S91, y se realizan repetidamente los procesos posteriores en la próxima LCU con el propósito de escanear la trama.

[0332]

En el ejemplo de la figura 22, la unidad que determina la línea intra-LCU 77A selecciona si se utilizaran los coeficientes para otros casos diferentes del límite de línea en la unidad de operación del filtro 74-1 o si se utilizarán los coeficientes para el caso del límite de línea en la unidad de operación del filtro del límite de línea 74-2 . De esta manera el proceso de filtración en el límite de línea y el proceso de filtración en posiciones diferentes al límite de línea se conmutan.

[0333]

Por el contrario, solamente se puede usar un tipo de coeficientes y la dirección del pixel a ser leído se puede controlar de tal manera que el proceso de filtración en el límite de línea y el proceso de filtración en las posiciones distintas del límite de línea se conmuta.

[0334]

Por ejemplo, el cálculo de $p0_0$ en la filtración fuerte basada en R2W2 se describirá como un ejemplo. En la Expresión (50) para la filtración fuerte del esquema HEVC, un término que relaciona a “p (bloque BKu)” es “$p2_i + 2*p1_i + 2*q0_i$”.

[0335]

Por otro lado, en la Expresión (56) para la filtración fuerte del esquema HEVC, un término que relaciona a “p (bloque BKu)” es “$4*p1_i + q0_i$”.

[0336]

Aquí, “$4* P1_i + q0_i$” = “$q0_i + 2*p1_i + 2*p1_i$”.

[0337]

Es decir, el término que relaciona a “p” en la Expresión (50) y el término que relaciona a “p” en la Expresión (56) también utilizan los coeficientes 1, 2 y 2 en este orden desde el lado izquierdo. De este modo, reemplazando (cambiando la dirección de lectura) los valores de píxel a ser multiplicados con los coeficientes utilizan los mismos coeficientes, es posible conmutar entre el proceso de filtración en el límite de línea y el proceso de filtración en posiciones diferentes del límite de línea.

[0338]

[14. Séptima implementación de la unidad de filtración de desbloqueo] [Ejemplo de configuración de la unidad de filtración de desbloqueo]

La figura 24 ilustra la configuración de una séptima implementación de la unidad de filtración de desbloqueo.

[0339]

En el ejemplo de la figura 24, la unidad de filtración de desbloqueo 24 está configurada para incluir una memoria de imagen 71, una unidad que determina el límite de bloque 72, una unidad que determina la fuerza del filtro 73, una unidad de operación del filtro 74, una memoria de coeficiente 76, y un controlador 77.

[0340]

En la unidad de filtración de desbloqueo 24 de la figura 24, la memoria de imagen 71, la memoria de coeficiente 76, y el controlador 77 son los mismos que aquellos de la unidad de filtración de desbloqueo 24 de la figura 22.

[0341]

La unidad de filtración de desbloqueo 24 de la figura 24 es diferente de la unidad de filtración de desbloqueo 24 de la figura 22 en que la unidad que determina el límite de bloque 72-1 y la unidad que determina el límite de línea/límite de bloque 72-2 se cambian colectivamente a la unidad que determina el límite de bloque 72. La unidad de filtración de desbloqueo 24 de la figura 24 es diferente de la unidad de filtración de desbloqueo 24 de la figura 22 en que la unidad que determina la fuerza del filtro 73-1 y la unidad que determina la fuerza del filtro del límite de línea 73-2 se cambian colectivamente con la unidad que determina la fuerza del filtro 73. La unidad de filtración de desbloqueo 24 de la figura 24 es diferente de la unidad de filtración de desbloqueo 24 de la figura 22 en que la unidad de operación del filtro 74-1 y la unidad de operación del filtro del límite de línea 74-2 se cambian colectivamente a la unidad de operación de filtro 74.

[0342]

Además, la unidad de filtración de desbloqueo 24 de la figura 24 es diferente de la unidad de filtración de desbloqueo 24 de la figura 22 en que la unidad que determina la línea intra-LCU 77A del controlador 77 se cambia a una unidad que determina la línea intra-LCU 77B.

[0343]

Es decir, la unidad que determina la línea intra-LCU 77B controla la dirección de la lectura de pixel desde la memoria de imagen 71 como se describió anteriormente por medio de un ejemplo de cálculo del $p0_0$ en la filtración fuerte basada en R2W2. Por lo tanto, la memoria de imagen 71 lee los pixeles basada en el control de la unidad que determina la línea intra-LCU 77B y suministra los pixeles leídos a la unidad que determina el límite de bloque 72, la unidad que determina la fuerza del filtro 73, y la unidad de operación del filtro 74.

[0344]

La unidad que determina el límite de bloque 72 realiza la determinación del límite inter-bloque. Es decir, la unidad que determina el límite de bloque 72 realiza el proceso de determinación del límite inter-bloque descrito anteriormente para cada bloque utilizando los datos de imagen leídos de la memoria de imagen 71 y detecta un límite inter-bloque en el que se realiza el proceso de filtración. La unidad que determina el límite de bloque 72 envía los resultados de detección a la unidad que determina la fuerza del filtro 73.

[0345]

La unidad que determina la fuerza del filtro 73 determina la fuerza del filtro de la manera descrita anteriormente. Es decir, la unidad que determina la fuerza del filtro 73 determina si el proceso de filtración se realizará en un modo de filtración fuerte o en un modo de filtración débil para cada línea usando los datos de imagen de dos bloques adyacentes con el límite inter-

bloque interpuesto, suministrado desde la memoria de imagen 71 y envía los resultados de la determinación a la unidad de operación del filtro 74.

[0346]

La unidad de operación del filtro 74 realiza la operación de filtración con la fuerza del filtro determinada por la unidad que determina la fuerza del filtro 73 para cada línea utilizando los datos de imagen almacenados en la memoria de imagen 71 y los coeficientes de filtro leídos desde la memoria de coeficientes 76. La unidad de operación del filtro 74 envía de los datos de imagen que han sido sometidos al proceso de filtración a la memoria de cuadro 25.

[0347]

El proceso de filtración de desbloqueo de la unidad de filtración de desbloqueo 24 de la figura 24 es básicamente el mismo que el proceso de filtración de desbloqueo descrito anteriormente con referencia a la figura 23, y la descripción de este no se proporcionará.

[0348]

Como antes, los píxeles que no podían ser remitidos debido al rango de imagen estrechado usado para la operación de filtración se interpolan utilizando la inclinación de la forma de onda utilizada en la determinación del límite inter-bloque. Como resultado, el proceso de desbloqueo cuando los valores de píxel tienen una inclinación se puede realizar incluso cuando el rango de imagen en el límite de línea inter-LCU es estrecho.

[0349]

Por otra parte, en la determinación del límite inter-bloque y la determinación de la filtración fuerte/débil, los píxeles que no se pueden ser referidos debido al rango de imagen estrecho utilizado para la operación de filtración se interpolan utilizando la inclinación y el relleno de la forma de onda utilizada en la determinación del límite inter-bloque. Como resultado, el proceso de determinación de desbloqueo cuando los valores de píxel tienen una inclinación se puede realizar incluso cuando el rango de imagen en el límite de línea inter-LCU es estrecho.

[0350]

De la misma manera, en el límite de línea inter-LCU, incluso cuando el rango de imagen es estrecho, es posible reducir la memoria de línea a la vez que se mantiene la función del proceso de desbloqueo.

[0351]

[15. Octava implementación de la unidad de filtración de desbloqueo] [Ejemplo de configuración de la unidad de filtración de desbloqueo]

La figura 25 ilustra la configuración de una octava implementación de la unidad de filtración de desbloqueo.

[0352]

En el ejemplo de la figura 25, la unidad de filtración de desbloqueo 24 está configurada para incluir una memoria de imagen 81, una unidad que determina el límite de bloque 82, una unidad que determina la fuerza del filtro 83, una unidad de operación de filtro 84, una memoria de coeficiente 85, y un controlador 86.

[0353]

La memoria de imagen 81 es una unidad que corresponde a la memoria de imagen 71 de la figura 22 y está configurada como una memoria de línea. La memoria de imagen 81 almacena los datos de imagen suministrados desde el sumador 23. La memoria de imagen 81 lee los datos de imagen almacenados en ella, y suministra los datos de imagen leídos a la unidad que determina el límite de bloque 82, la unidad que determina la fuerza del filtro 83, y la unidad de operación de filtro 84.

[0354]

En posiciones diferentes al límite de línea, los datos de imagen no se almacenan en la memoria de imagen 81, pero puede haber casos donde los datos de imagen suministrados desde el sumador 23 se suministren a las unidades respectivas y se procesen. Sin embargo, en el ejemplo de la figura 25, por razones de conveniencia, se describe que los datos de imagen que han pasado a través de la memoria de imagen 81 se procesan.

[0355]

La unidad que determina el límite de bloque 82 deriva límites cada ocho líneas bajo el control del controlador 86 para calcular los parámetros usados para la determinación y realiza la determinación del límite inter-bloque para cada cuatro líneas. Es decir, la unidad que determina el límite de bloque 82 deriva un límite entre TU y PU utilizando los datos de imagen leídos de la memoria de imagen 81 y deriva un valor BS. Además, la unidad que determina el límite de bloque 82 obtiene un promedio de los parámetros de cuantización QP de dos regiones adyacentes con el límite actual interpuesto para calcular un promedio QP (parámetro de cuantización) y calcula los parámetros tc y β con base en el QP promedio calculado.

[0356]

Por otra parte, la unidad que determina el límite de bloque 82 determina si debe o no realizar la filtración de cada cuatro líneas utilizando los datos de imagen de dos bloques adyacentes con el límite inter-bloque interpuesto, suministrados desde la memoria de imagen 81 y los parámetros calculados. La unidad que determina el límite de bloque 82 suministra los parámetros calculados a la unidad que determina la fuerza del filtro junto con los resultados de la determinación del límite.

[0357]

La unidad que determina la fuerza del filtro 83 determina la fuerza del filtro cada cuatro líneas bajo el control del controlador 86. Es decir, cuando la unidad que determina el límite de bloque 82 determina que la filtración se va a realizar, la unidad que determina la fuerza del filtro 83 determina si se realizará el proceso de filtración en el modo de filtración fuerte o en el modo de filtración débil y envía los resultados de la determinación a la unidad de operación de filtro 84.

[0358]

La unidad de operación de filtro 84 realiza la operación de filtración con el filtro fuerte determinado por la unidad que determina la fuerza del filtro 83 cada cuatro líneas utilizando los datos de imagen almacenados en la memoria de imagen 81 y los coeficientes de filtro leídos de la memoria de coeficiente 85 bajo el control del controlador 86. En especial, la unidad que determina la fuerza del filtro 83 determina que fuerza de filtración se va a realizar, la unidad de operación de filtro 84 realiza un proceso de filtración con base en el recorte utilizando las Expresiones (106) a (111) descritas anteriormente. La unidad de operación de filtro 84 envía los datos de imagen que han sido sometidos al proceso de filtración a la memoria de cuadro 25 en la etapa posterior.

[0359]

La memoria de coeficiente 85 almacena los coeficientes de filtro usados en la operación de filtración del proceso de filtración de desbloqueo. La memoria de coeficiente 85 lee los coeficientes de filtro almacenados en el mismo y suministra los coeficientes de filtro a la unidad de operación de filtro 84.

[0360]

El controlador 86 recibe la información (la información On/Off del filtro de desbloqueo, los valores de compensación de los parámetros β y tc, y los parámetros de recorte) de una unidad de operación (no ilustrada) (en el caso del lado de decodificación, del decodificador sin pérdidas 52). Por otra parte, el controlador 86 también se suministra con la información del

modo de predicción necesaria y los parámetros necesarios para el filtro de desbloqueo tal como los parámetros de cuantización.

[0361]

Por ejemplo, el controlador 86 suministra la información de entrada a la unidad correspondiente y controla las respectivas unidades de la unidad de filtración de desbloqueo 24 con base en la información de entrada ON/OFF.

[0362]

El controlador 86 controla la memoria de imagen 81 para almacenar los datos de imagen correspondiente a un número predeterminado de líneas en la parte inferior del bloque y lee los datos de imagen almacenados en la memoria de imagen 81. Por otra parte, cuando se realiza la filtración, el controlador 86 controla la unidad de operación de filtro 84 para realizar un proceso de filtración fuerte con base en el recorte. En este caso, el controlador 86 suministra los parámetros (por ejemplo, un valor de recorte pv y un valor de multiplicación de tc del valor de recorte) relacionados con el proceso de recorte a la unidad de operación de filtro 84.

[0363]

[Operación de la unidad de filtración de desbloqueo]

A continuación, el proceso de desbloqueo de la unidad de filtración de desbloqueo 24 de la figura 25 se describirá con referencia al diagrama de flujo de la figura 26.

[0364]

Por ejemplo, la información ON/OFF, un valor de compensación β, un valor de compensación Tc, y los parámetros relacionados con el proceso de recorte se entran al controlador 86 a través de una unidad de operación (no ilustrada) (en el caso del lado de decodificación, el decodificador sin pérdidas 52). Por otra parte, el controlador 86 también se suministra con la información de modo de predicción y los parámetros necesarios para tal filtro de desbloqueo tal como los parámetros de cuantización.

[0365]

En la etapa S201, el controlador 86 establece las compensaciones de filtro (compensaciones β y Tc) y suministros la información de compensación establecida a la unidad que determina la fuerza del filtro 83.

[0366]

En la etapa S202, el controlador 86 determina si se puede utilizar el filtro de desbloqueo con base en la información ON/OFF. Cuando se determina en la etapa S202 que no se puede utilizar el filtro de desbloqueo, el proceso de filtración de desbloqueo termina.

[0367]

Cuando se determina en la etapa S202 que se puede usar el filtro de desbloqueo, el controlador 86 envía una notificación de esto a la unidad que determina el límite de bloque 82, la unidad que determina la fuerza del filtro 83, y la unidad de operación de filtro 84, y el flujo procede a la etapa S203. En este caso, los parámetros necesarios y similares también se suministran desde el controlador 86 a las unidades respectivas.

[0368]

En la etapa S203, la unidad que determina el límite de bloque 82 deriva un límite entre TU y PU cada 8 líneas. En la etapa S204, la unidad que determina el límite de bloque 82 deriva un valor de fuerza del límite (BS) con base en la información en el límite entre TU y PU derivada en la etapa S203 y la información del modo de predicción y los similares se suministran desde el controlador 86.

[0369]

En la etapa S205, la unidad que determina el límite de bloque 82, la unidad que determina la fuerza del filtro 83, y la unidad de operación de filtro 84 realizan un proceso de filtración de límite de luminancia. En el proceso de filtración límite de luminancia, la filtración límite de luminancia se realiza en las señales de luminancia por el proceso de la etapa S205, que se describirá con referencia a la figura 27.

[0370]

En la etapa S206, la unidad que determina el límite de bloque 82, la unidad que determina la fuerza del filtro 83 y la unidad de operación de filtro 84 realizan un proceso de filtración de límite de crominancia. Mediante el proceso de la etapa S206, se realiza la filtración del límite de crominancia sobre las señales de crominancia.

[0371]

En la etapa S207, el controlador 86 determina si todos los límites han sido procesados. Cuando se determina en la etapa S207 que todos los límites no han sido procesados, el flujo vuelve a la etapa S205 y se realizan repetidamente los procesos posteriores. Cuando se determina en la etapa S207 que todos los límites han sido procesados , el flujo procede a la etapa S208.

[0372]

En la etapa S208, el controlador 86 determina si han sido procesados todos los CUs. Cuando se determina en la etapa S208 que no han sido procesados todos los CUs, el flujo vuelve a la etapa S203 y realiza repetidamente todos los procesos posteriores.

[0373]

Cuando se determine en la etapa S208 que han sido procesados todos los CUs, el proceso de filtración de desbloqueo termina.

[0374]

[Ejemplo del proceso de filtración del límite de luminancia]

A continuación, se describirá el proceso de filtración límite de luminancia en la etapa S205 de la figura 26 con referencia al diagrama de flujo de la figura 27.

[0375]

En la etapa S231, la unidad que determina el límite de bloque 82 determina si el valor BS es mayor que 0 cada ocho líneas. Cuando se determine en la etapa S231 que el valor BS no es mayor que 0, el proceso de filtración del límite de luminancia termina. Es decir, en este caso, no se realiza el proceso de filtración de la unidad de operación de filtro 84 sobre el límite de luminancia, y los valores de píxel antes de la filtración se envían como son.

[0376]

Cuando se determina en la etapa S231 que el valor BS es mayor que 0, el flujo procede a la etapa S232. En la etapa S232, la unidad que determina el límite de bloque 82 obtiene un promedio de los parámetros de cuantización QP de dos regiones adyacentes a un límite actual utilizando los valores de pixel de la memoria de imagen 81 cada ocho líneas para calcular un promedio QP (parámetro de cuantización). En este caso, los parámetros de cuantización QP se suministran desde el controlador 86.

[0377]

En la etapa S233, la unidad que determina el límite de bloque 82 calcula los parámetros tc y β con base en el QP promedio calculado en la etapa S232 cada ocho líneas.

[0378]

En la etapa S234, la unidad que determina el límite de bloque 82 realiza la determinación ON/OFF de filtro cada cuatro líneas. Es decir, la unidad que determina el límite de bloque 82 realiza la determinación ON/OFF del filtro cada cuatro líneas utilizando los

parámetros calculados β y los similares. Este proceso de determinación usa la Expresión (68) descrita anteriormente, por ejemplo.

[0379]

Los resultados de la determinación ON/OFF en la etapa S234 se suministran a la unidad que determina la fuerza del filtro 83 junto con los parámetros calculados, como los resultados de la determinación del límite.

[0380]

Por otra parte, en la etapa S235, la unidad que determina la fuerza del filtro 83 realiza la determinación de la fuerza de filtro cada cuatro líneas. Es decir, la unidad que determina la fuerza del filtro 83 realiza la determinación de la fuerza de filtro cada cuatro líneas utilizando los parámetros β, tc, y similares calculados por la unidad que determina el límite de bloque 82. Este proceso de determinación utiliza el primer término de las Expresiones (69) y (61) a (63) descritas anteriormente, por ejemplo.

[0381]

Los resultados de la determinación ON/OFF en la etapa S234 y los resultados de la determinación de la fuerza del filtro en la etapa S235 se suministran a la unidad de operación de filtro 84. En este caso, el parámetro tc también se suministra a la unidad de operación de filtro 84.

[0382]

En la etapa S236, la unidad de operación de filtro 84 determina si se va a aplicar filtración fuerte a las cuatro líneas actuales sobre la base de los resultados de la determinación de la unidad que determina la fuerza del filtro 83. Cuando se determina en la etapa S236 que se va a aplicar filtración fuerte a las cuatro líneas actuales, el flujo procede a la etapa S237.

[0383]

En la etapa S237, la unidad de operación de filtro 84 realiza un proceso de filtración fuerte con base en el recorte utilizando las Expresiones (106) a (111) descritas anteriormente bajo el control del controlador 86. La unidad de operación de filtro 84 envía los valores de píxel después del proceso de filtración a la etapa siguiente.

[0384]

Cuando se determine en la S236 que no se va a aplicar filtración fuerte, la unidad de operación de filtro 84 determina en la etapa S238 si se va a aplicar filtración débil a las cuatro líneas actuales con base en los resultados de la determinación de la unidad que determina la

fuerza del filtro 83. Cuando se determine en la etapa S238 que se va a aplicar filtración débil, el flujo procede a la etapa 239.

[0385]

En la etapa S239, la unidad de operación de filtro 84 realiza un proceso de filtración débil. La unidad de operación de filtro 84 envía de los valores de píxel después del proceso de filtración a la etapa siguiente.

[0386]

Cuando se determine en la etapa S238 que no se va a aplicar filtración débil, el flujo procede a la etapa 240. En la etapa S240, la unidad de operación de filtro 84 no realiza el proceso sobre las cuatro líneas actuales con base en los resultados de la determinación en la unidad que determina la fuerza del filtro 83, pero envía los valores de píxel después del proceso de filtración a la etapa siguiente como son.

[0387]

En la etapa S240, la unidad de operación de filtro 84 determina si se ha completado el proceso para ocho líneas, y termina el proceso de desbloque de la señal de luminancia cuando se determina que se ha completado el proceso para ocho líneas. Cuando se determine en la etapa S240 que el proceso no se ha completado para las ocho líneas, el flujo vuelve a la etapa S234 y se realizan repetidamente los procesos posteriores.

[0388]

Como antes, puesto que el proceso de filtración fuerte con base en el recorte se realiza cuando se determina que se va a aplicar filtración fuerte, es posible suprimir la influencia de un desajuste del proceso de determinación y una incompletitud del proceso de determinación tanto como es posible. De esta manera, es posible permitir que el proceso de filtración de desbloqueo aplique apropiadamente la filtración.

[0389]

En las descripciones anteriores, se ha descrita la aproximación lineal y el proceso de filtración fuerte con base en el recorte. En la descripción de la aproximación lineal, aunque se han descrito los casos R2W2 y R2W1, el proceso de aproximación lineal para los casos R3W2 y R1W1 se puede realizar de la misma manera.

[0390]

[16. Novena implementación] [Ejemplo de R3W2]

A continuación, se describirá el caso R3W2 con referencia a la figura 28. En el ejemplo de la figura 28, los círculos blancos en la línea continua indican valores de píxel de los respectivos pixeles $p2_i$, $p1_i$, $p0_i$, $q0_i$, $q1_i$ y $q2_i$ cerca del límite BB ilustrado en la figura 16 antes del filtro de desbloqueo. Como se ilustra en la figura 28, un espacio entre los bloques BKu y BKI es $D = p0_i – q0_i$.

[0391]

Por otra parte, los círculos sobre la línea de puntos indican los valores de pixel respectivos después de la filtración fuerte de las señales de luminancia en el esquema HEVC. Es decir, en este caso, la diferencia entre los valores de píxel antes y después de la filtración fuerte es 1/8*D para $q2_i$, 2/8*D para $q1_i$ y 3/8*D para $q0_i$. Del mismo modo, la diferencia entre los valores de píxel antes y después de la filtración fuerte 3/8*D para $p0_i$, 2/8*D para $p1_i$, y 1/8*D para $P2_i$.

[0392]

En este caso, teniendo en cuenta el proceso de filtración de la señal de luminancia para el caso R3W2, el pixel $p2_i$ no se somete a filtración. Es decir, el píxel $p2_i$ tiene un valor de píxel indicado por el círculo de puntos en la línea continua, y la diferencia para el valor de píxel $p1_i$ indicada por la línea de puntos aumenta. Por lo tanto, los valores de píxel sólo tienen una pendiente abrupta entre los valores de píxel de $p2_i$ indicados por el círculo de puntos en la línea sólida y $P1_i$ indicado por la línea de puntos.

[0393]

Por lo tanto, en el caso R3W2, con el fin de hacer está pendiente suave, se expresa una diferencia $\Delta p1_i$ entre el valor de píxel (círculo blanco) de $p0_i$ antes de la filtración y el valor del píxel (círculo rayado) después de la filtración fuerte se expresa por la Expresión (112) más adelante.

[Fórmula Matemática 34]

$$\begin{aligned}\Delta p1_i &= 1/16*(-3*D+2*4)\\ &= (-3*(p0_i+q0_i)+8)>>4\\ &= (13*p0_i+3*q0_i+8)>>4-p1_i\\ &\fallingdotseq (4*p2_i+5*p1_i+4*p0_i+3*q0_i+8)>>4-p1_i \quad \cdots (112)\end{aligned}$$

[0394]

Es decir, la Expresión (112) se deriva de tal manera queΔ p1 $_i$ toma 3/16*D que es la mitad de la diferencia Δ $p0_i$ entre el valor del píxel (círculo blanco) de $p0_i$ antes de la filtración y el valor del píxel (círculo rayado) después de la filtración fuerte.

[0395]

Dado que el segundo término del lado derecho de la última ecuación de aproximación de la Expresión (112) es el valor del píxel (círculo blanco) antes de la filtración, el primer término en el lado derecho de la última ecuación de aproximación de la Expresión (112) representa el valor del píxel después de la filtración fuerte, de $p1_i$ indicado por el círculo rayado.

[0396]

Es decir, los coeficientes multiplicados por los valores de píxel respectivos en el primer término del lado derecho de la última ecuación de aproximación de la Expresión (112) se utilizan como los coeficientes de filtración fuerte para el caso R3W2. Por lo tanto, los resultados de la filtración fuerte para el caso R3W2 se pueden suavizar cuando se comparan con la pendiente de la línea de puntos de la técnica convencional, como se indica por la pendiente de la línea en negrita en la figura 28.

[0397]

La expresión de determinación para la filtración fuerte del caso R3W2 se expresa mediante la Expresión (113) siguiente.

[Fórmula Matemática 35]

$$
\begin{aligned}
&d < (\beta >> 2) \\
&d = abs(q2_2 - 2*q1_2 + q0_2) + abs(p2_5 - 2*p1_5 + p0_5) \\
&\quad + abs(q2_2 - 2*q1_2 + q0_2) + abs(q2_5 - 2*q1_5 + q0_5) \\
&\text{si } (|p2_i - p0_i| + |q0_i - q3_i|) < (\beta >> 3) \\
&\text{si } |p0_i - q0_i| < ((5*t_c + 1) >> 1) \qquad \cdots (113)
\end{aligned}
$$

Una expresión que aparece después de la primera “y” de la Expresión (113) para la determinación de la filtración fuerte del caso R3W2 es diferente de una expresión que aparece después de la primera “y” de la Expresión (69) para determinar la filtración fuerte del esquema HEVC. En el caso R3W2, ya que $p3_i$ no podría ser referido, no se aplica el relleno (es decir, $p3_i$ = $p2_i$).

[0398]

Por lo tanto, las dos expresiones son diferentes sólo en que la expresión “(lp3_i - $p0_i$ <<1I $q0_i$ - $q3_i$l)<(β>>3))” que aparece después de la primera “y” de la Expresión (69) se cambia a la

expresión “(lp3$_i$ - p0$_i$ <<1I q0$_i$ - q3$_i$l)<(β>>3))” que aparece después de la primera “y” de la Expresión (113).

[0399]

Por otra parte, en la filtración fuerte del caso R3W2, una expresión relacionada con p1$_0$ es diferente de ésa en la Expresión (52) para la filtración fuerte del esquema HEVC de acuerdo con la Expresión (112) descrita anteriormente como se ilustra en la Expresión (114).

[Fórmula Matemática 36]

$$p1_0 = Clip_{0-255}((4*p2_i+5*p1_i+4*p0_i+3*q0_i+8)>>4); \quad i = 0,7 \qquad \cdots (114)$$

[0400]

En la filtración fuerte del caso R3W2, la Expresión (54) relacionada con p2$_0$ de la filtración fuerte del esquema HEVC no es necesaria y se omite.

[0401]

De esta manera, se puede realizar la filtración fuerte del caso R3W2 sin el deterioro de la exactitud.

[0402]

Por otra parte, en la presente especificación, el tamaño de las unidades de procesamiento de filtro de desbloqueo puede ser del tamaño de los macro-bloques del H.264/AVC, el tamaño de los bloques de árbol de codificación de las unidades de codificación del HEVC y los similares, por ejemplo. Por otra parte, en la presente especificación, el término bloque o macro-bloque también incluye una unidad de codificación (CU), una unidad de predicción (PU), y una unidad de transformación (TU) en el contexto del HEVC.

[0403]

Se pueden realizar una serie de procesos descritos en esta especificación mediante hardware, software, o una combinación de los mismos. Cuando los procesos se realizan mediante software, se puede instalar un programa que incluye la secuencia del proceso y se ejecuta mediante una memoria de un computador ensamblado dentro del hardware exclusivo. Alternativamente, el programa se puede instalar en, y ejecutar mediante un computador de propósito general realizando varios procesos.

[0404]

Por ejemplo, el programa se puede grabar por adelantado en un disco duro o una memoria de sólo lectura (ROM) o un medio de grabación. Por otra parte, el programa se puede grabar temporalmente o permanentemente en un disco magnético (tal como un disco flexible), un disco óptico (como un CD- ROM (disco compacto de memoria de sólo lectura), un disco MO (magneto óptico), o un DVD (disco versátil digital)), y un medio de almacenamiento extraíble (no mostrado) (tal como una memoria semiconductora), por ejemplo. Dicho medio de almacenamiento extraíble puede ser suministrado como el llamado paquete de software.

[0405]

El programa se puede transferir de forma inalámbrica desde un sitio de descarga al computador o se puede transferir a través de cables al computador a través de una red como una red de área local (LAN) o de Internet, además de instalar el programa en el computador desde el medio de grabación extraíble descrito anteriormente. El computador puede recibir el programa transferido de esta manera e instalar el mismo en un medio de grabación tal como un disco duro incorporado.

[0406]

[19. Décima implementación]

[Aplicación para codificación y decodificación de imagen en multi-vista]

Las series de procesos se pueden aplicar a la codificación y decodificación de imagen en multi-vista. La figura 29 ilustra un ejemplo de un esquema de codificación de imagen en multi-vista.

[0407]

Como se ilustra en la figura 29, una imagen multi-vista incluye imágenes de una pluralidad de vista, y una imagen de una vista única predeterminada entre la pluralidad de vista se designa como una imagen de vista base. Las imágenes de vista respectivas diferentes de la imagen de vista base se tratan como imágenes de vista no base.

[0408]

Cuando se codifica la imagen multi-vista como se ilustra en la figura 29, en las vistas respectivas (las mismas vistas), se pueden establecer los parámetros (por ejemplo, un valor de recorte y un factor de multiplicación en relación con tc del valor de recorte) en relación con el proceso de recorte para el proceso de filtración fuerte. Por otra parte, en las vistas respectivas (visitas diferentes), los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos para las otras vistas se pueden compartir.

[0409]

En este caso, los parámetros relacionados con el proceso de recorte establecidos para una vista base se utilizan en por lo menos una vista no base. Alternativamente, por ejemplo, los parámetros relacionados con el proceso de recorte establecidos para una vista no base (view_id = 0) se utilizan en por lo menos una vista base y una vista no base (view_id = 1).

[0410]

Además, en las respectivas vistas (las mismas vistas), se pueden establecer los parámetros relacionados con el proceso de recorte para la filtración fuerte. Por otra parte, en las respectivas vistas (visitas diferentes), los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos para las otras vistas se pueden compartir.

[0411]

En este caso, los parámetros relacionados con el proceso de recorte establecidos para una vista base se utilizan en por lo menos una vista no base. Alternativamente, por ejemplo, los parámetros relacionados con el proceso de recorte establecidos para una vista no base (view_id = 0) se utilizan en por lo menos una vista base y una vista no base (view_id = 1).

[0412]

De esta manera, es posible permitir que el proceso de desbloqueo aplique adecuadamente la filtración.

[0413]

[Dispositivos de codificación de imagen Multi-vista]

La figura 30 es un diagrama que ilustra un dispositivo de codificación de imagen multi-vista que realiza la codificación de imagen multi-vista descrita anteriormente. Como se ilustra en la figura 30, un dispositivo de codificación de imagen multi-vista 600 incluye una unidad de codificación 601, una unidad de codificación 602, y un multiplexor 603.

[0414]

La unidad de codificación 601 codifica una imagen de vista base para generar una corriente codificada de imagen de vista base. La unidad de codificación 602 codifica una imagen de vista no-base para generar una corriente codificada de imagen de vista no-base. El multiplexor 603 multiplexa la corriente codificada de imagen de vista base generada por la unidad de codificación 601 y la corriente codificada de imagen de vista no-base generada por la unidad de codificación 602 para generar una corriente codificada de imagen multi- vista.

[0415]

El dispositivo de codificación de imagen 10 (Figura 1) se puede aplicar a las unidades de codificación 601 y 602 del dispositivo de codificación de imagen multi-vista 600. En este caso, el dispositivo de codificación de imagen multi-vista 600 establece los parámetros en relación con el proceso de recorte para la filtración fuerte establecida por la unidad de codificación 601 y los parámetros relacionados con el proceso de recorte para la filtración fuerte establecida por la unidad de

codificación 602 y transfiere los parámetros.

[0416]

Como se describió anteriormente, los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos por la unidad de codificación 601 se pueden transferir para ser compartidos por las unidades de codificación 601 y 602. Por el contrario, los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos por la unidad de codificación 602 se pueden transferir para que sean compartidos por las unidades de codificación 601 y 602.

[0417]

[Dispositivos de decodificación de imagen Multi-vista]

La figura 31 es un diagrama que ilustra un dispositivo de decodificación de imagen multi-vista que realiza la decodificación de imagen multi-vista descrita anteriormente. Como se ilustra en la figura 31, un dispositivo de decodificación de imagen multi-vista 610 incluye un demultiplexor 611, una unidad de decodificación 612, y una unidad de decodificación 613.

[0418]

El demultiplexor 611 demultiplexa la corriente codificada de imagen multi vista en la que la corriente codificada de imagen de vista base y la corriente codificada de imagen de vista no-base se multiplexan para extraer la corriente codificada de imagen de vista base y la corriente codificada de imagen de vista no-base. La unidad de decodificación 612 decodifica la corriente codificada de imagen de vista base extraída por el demultiplexor 611 para obtener una imagen de vista base. La unidad de decodificación 613 decodifica la corriente codificada de imagen de vista no-base extraída por el demultiplexor 611 para obtener una imagen de vista no- base.

[0419]

El dispositivo de decodificación de imagen 50 (figura 6) se puede aplicar a las unidades de decodificación 612 y 613 del dispositivo de decodificación de imagen multi-vista 610. En este caso, el dispositivo de decodificación de imagen multi-vista 610 realiza el proceso utilizando los

parámetros relacionados con el proceso de recorte para la filtración fuerte, que se establece por la unidad de codificación 601 y se decodifican por la unidad de decodificación 612 y los parámetros relacionados con el proceso de recorte para la filtración fuerte, que se establece por la unidad de codificación 602 y se decodifican por la unidad de decodificación 613.

[0420]

Como se describió anteriormente, hay un caso en el que el parámetro relacionado con el proceso de recorte para la filtración fuerte establecido por la unidad de codificación 601 (o la de codificación 602) se transfieren para ser compartidas por las unidades de codificación 601 y 602. En este caso, el dispositivo de decodificación de imagen multi-vista 610 realiza el proceso utilizando los parámetros relacionados con el proceso de recorte para la filtración fuerte, que se establece por la unidad de codificación 601 (o la de codificación 602) y decodificados por la unidad de decodificación 612 (o la unidad de decodificación 613).

[0421]

[18. Undécima implementación]

[Aplicación a la codificación y decodificación de imagen de capa]

Las series anteriores de procesos se pueden aplicar a la codificación y decodificación de imagen de capa. La figura 32 ilustra un ejemplo de un esquema de codificación de imagen multi-vista.

[0422]

Como se ilustra en la figura 32, una imagen de capa incluye imágenes de una pluralidad de capas (resoluciones), y una imagen de una capa única predeterminada entre la pluralidad de resoluciones se designa como una imagen de capa base. Las respectivas imágenes de capas diferentes de la imagen de capa base se tratan como imágenes de capa no-base.

[0423]

Cuando se realiza la codificación de imagen de capa (escalabilidad espacio) como se ilustra en la figura 32 en las respectivas capas (las mismas capas), se pueden establecer los parámetros relacionados con el proceso de recorte para la filtración fuerte. Por otra parte, en las respectivas capas (capas diferentes), los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos para las otras capas se pueden compartir.

[0424]

En este caso, los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos para una capa base se utilizan en por lo menos una capa no base. Alternativamente, por ejemplo, los parámetros relacionados con el proceso de recorte para la

filtración fuerte establecidos para una capa no base (layer_id = 0) se utilizan en por lo menos una de las capas base y una capa no base (layer_id = 1).

[0425]

Además, en las respectivas capas (las mismas capas), se pueden establecer los parámetros relacionados con el proceso de recorte para la filtración fuerte. Por otra parte, en las respectivas capas (capas diferentes), los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos para las otras capas se pueden compartir.

[0426]

En este caso, los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos para una capa base se utilizan en por lo menos una capa no base. Alternativamente, por ejemplo, los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos para una capa no base (layer_id = 0) se utilizan en por lo menos una capa base y una capa no base (layer_id = 1).

[0427]

De esta manera, es posible permitir que el proceso de desbloqueo aplique adecuadamente la filtración.

[0428]

[Dispositivo de codificación de imagen de capa]

La figura 33 es un diagrama que ilustra un dispositivo de codificación de imagen de capa que realiza la codificación de imagen de capa descrita anteriormente. Como se ilustra en la figura 33, un dispositivo de codificación de imagen de capa 620 incluye una unidad de codificación 621,una unidad de codificación 622, y un multiplexor 623.

[0429]

La unidad de codificación 621 codifica una imagen de capa base para generar una corriente codificada de imagen de capa base. La unidad de codificación 622 codifica una imagen de capa no-base para generar una corriente codificada de imagen de capa no-base. El multiplexor 623 multiplexa la corriente codificada de imagen de capa base generada por la unidad de codificación 621 y la corriente codificada de imagen de capa no-base generada por la unidad de codificación 622 para generar una corriente codificada de imagen de capa.

[0430]

El dispositivo de codificación de imagen 10 (Figura 1) se puede aplicar a las unidades de codificación 621 y 622 del dispositivo de codificación de imagen de capa 620. En este caso, el dispositivo de codificación de imagen de capa 620 establece los parámetros en relación con el

proceso de recorte para la filtración fuerte establecida por la unidad de codificación 621 y los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos por los parámetros en relación con el proceso de recorte para la filtración fuerte establecidos por la unidad de codificación 602 y transfiere los parámetros.

[0431]

Como se describió anteriormente, los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos por la unidad de codificación 621 se pueden transferir para ser compartidos por las unidades de codificación 621 y 622. Por el contrario, los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos por la unidad de codificación 622 se pueden transferir para que sean compartidos por las unidades de codificación 621 y 622.

[0432]

[Dispositivo de decodificación de imagen de capar]

La figura 34 es un diagrama que ilustra un dispositivo de decodificación de imagen de capa que realiza la decodificación de imagen de capa descrita anteriormente. Como se ilustra en la figura 34, un dispositivo de decodificación de imagen de capa 630 incluye un demultiplexor 631, una unidad de decodificación 632, y una unidad de decodificación 633.

[0433]

El demultiplexor 631 demultiplexa la corriente codificada de imagen de capa en la que la corriente codificada de imagen de capa base y la corriente codificada de imagen de capa no-base se multiplexan para extraer la corriente codificada de imagen de capa base y la corriente codificada de imagen de capa no-base. La unidad de decodificación 632 decodifica la corriente codificada de imagen de capa base extraída por el demultiplexor 631 para obtener una imagen de capa base. La unidad de decodificación 633 decodifica la corriente codificada de imagen de capa no-base extraída por el demultiplexor 631 para obtener una imagen de capa no- base.

[0434]

El dispositivo de decodificación de imagen 50 (Figura 6) se puede aplicar a las unidades de decodificación 632 y 633 del dispositivo de decodificación de imagen multi-vista 630. En este caso, el dispositivo de decodificación de imagen multi-vista 630 realiza el proceso utilizando los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos por la unidad de codificación 621 y decodificados por la unidad de decodificación 632 y los parámetros relacionados con el proceso de recorte para la filtración fuerte que se establecen por la unidad de codificación 622 y decodificados por la unidad de decodificación 633.

[0435]

Como se describió anteriormente, hay un caso en el que los parámetros relacionados con el proceso de recorte para la filtración fuerte establecidos por la unidad de codificación 621 (o la de codificación 622) se transfieren para ser compartidos por las unidades de codificación 621 y 622. En este caso, el dispositivo de decodificación de imagen multi-vista 630 realiza el proceso utilizando los parámetros relacionados con el proceso de recorte para la filtración fuerte, que se establecen por la unidad de codificación 621 (o la de codificación 622) y decodificados por la unidad de decodificación 632 (o la unidad de decodificación 633).

[0436]

[19. Ejemplo de Aplicación]

El dispositivo de codificación de imagen 10 y el dispositivo de decodificación de imagen 50 según las formas de implementación descrita anteriormente se puede aplicar a diversos aparatos electrónicos, como un transmisor o un receptor que distribuye señales de radiodifusión por cable (tales como radiodifusión satelital o una TV por cable) o en Internet y distribuye señales a un terminal por comunicación celular, un dispositivo de grabación que graba imagen en un medio tal como un disco óptico, un disco magnético o una memoria flash o un dispositivo de reproducción que reproduce imagen desde estos medios de almacenamiento. A continuación se explicarán cuatro ejemplos de aplicación.

[0437]

[Primer ejemplo de aplicación]

La figura 35 ilustra un ejemplo de una configuración esquemática de un aparato de televisión al que se aplica la implementación descrita anteriormente. Un aparato de televisión 900 incluye una antena 901, un sintonizador 902, un demultiplexor 903, un decodificador 904, un procesador de señal de vídeo 905, una unidad de visualización 906, un procesador de señal de audio 907, un altavoz 908, y una interfaz externa 909. El aparato de televisión 900 incluye además un controlador 910, una interfaz de usuario 911, y similares.

[0438]

El sintonizador 902 extrae una señal de un canal deseado desde una señal de radiodifusión recibida a través de la antena 901 y demodula la señal extraída. Luego, el sintonizador 902 envía una corriente de bit codificada obtenida por demodulación al demultiplexor 903. Es decir, el sintonizador 902 sirve como medio de transmite en el aparato de televisión 900 que recibe la corriente codificada en la que se codifica la imagen.

[0439]

El demultiplexor 903 separa un corriente de vídeo y una corriente de audio de un programa a ser visto desde la corriente de bit codificada y envía cada corriente separada al decodificador 904. Por otra parte, el demultiplexor 903 extrae los datos auxiliares como EPG (Electronic Program Guide) de la corriente de bit codificada y suministra los datos extraídos al controlador 910. El demultiplexor 903 puede decodificar cuando la corriente de bit codificada está codificada..

[0440]

El decodificador 904 decodifica la entrada de la corriente de vídeo y la corriente de audio desde el demultiplexor 903. Entonces, el decodificador 904 envía los datos de vídeo generados por un proceso de decodificación al procesador de señal de vídeo 905. Por otra parte, el decodificador 904 envía los datos de audio generados por el proceso de decodificación al procesador de señal de audio 907.

[0441]

El procesador de señal de vídeo 905 reproduce la entrada de datos de vídeo del decodificador 904 y permite que la unidad de visualización 906 muestre el video. El procesador de señal de vídeo 905 también puede permitir que la unidad de visualización 906 muestre una pantalla de aplicación suministrada a través de la red. El procesador de señal de vídeo 905 también puede realizar un proceso adicional tal como retirar el ruido, por ejemplo, a los datos de vídeo según el ajuste. Además, el procesador de señal de vídeo 905 puede generar una imagen GUI (Graphical User Interface) tal como un menú, un botón y un cursor, por ejemplo, y superponer la imagen generada en una imagen de salida.

[0442]

La unidad de visualización 906 es impulsada por una señal de accionamiento suministrada desde el procesador de señal de vídeo 905 para mostrar el vídeo o la imagen en una pantalla de vídeo de un dispositivo de visualización (por ejemplo, una pantalla de cristal líquido, una pantalla de plasma, un OLED, y similares).

[0443]

El procesador de señal de audio 907 realiza un proceso de reproducción tal como conversión D/A y la amplificación para la entrada de los datos de audio desde el decodificador 904 y permite que el altavoz 908 emita el audio. El procesador de señal de audio 907 también puede realizar un proceso adicional tal como retirar el ruido de los datos de audio. La interfaz externa 909 es la interfaz para conectar el aparato de televisión 900 y un dispositivo externo o la red. Por ejemplo, la corriente de vídeo o la corriente de audio recibidas a través de la interfaz externa 909 puede ser decodificada por el decodificador 904. Es decir, la interfaz externa 909

también sirve como el medio de transmisión en el aparato de televisión 900 que recibe la corriente codificada en la que se codifica la imagen.

[0444]

El controlador 910 incluye un procesador tal como un Unidad Central de Procesamiento (CPU) y una memoria tal como una memoria de acceso aleatoria (RAM) y una memoria de sólo lectura (ROM). La memoria almacena el programa ejecutado por la CPU, los datos del programa, los datos de la EPG, los datos obtenidos a través de la red y similares. El programa almacenado en la memoria es leído por la CPU en el arranque del aparato de televisión 900 para ser ejecutado, por ejemplo. La CPU controla la operación del aparato de televisión 900 de acuerdo a una entrada de señal de operación desde la interfaz de usuario 911, por ejemplo, mediante la ejecución del programa.

[0445]

La interfaz de usuario 911 está conectada al controlador 910. La interfaz de usuario 911 incluye un botón y un interruptor para el usuario para utilizar el aparato de televisión 900, un receptor de la señal de control remoto y los similares, por ejemplo. La interfaz de usuario 911 detecta la operación del usuario a través de los componentes para generar la señal de operación y envía la señal de operación generada al controlador 910.

[0446]

El bus 912 conecta el sintonizador 902, el demultiplexor 903, el decodificador 904, el procesador de señal de vídeo 905, el procesador de señal de audio 907, la interfaz externa 909, y el controlador 910 el uno al otro.

[0447]

En el aparato de televisión 900 configurado de esta manera, el decodificador 904 tiene las funciones del dispositivo de decodificación de imagen 50 según la implementación descrita anteriormente. Por lo tanto, cuando las imágenes se decodifican en el aparato de televisión 900, el proceso de filtración de desbloqueo puede aplicar adecuadamente la filtración.

[0448]

[Segundo ejemplo de aplicación]

La figura 36 ilustra un ejemplo de una configuración esquemática de un teléfono móvil al que se aplica la implementación descrita anteriormente. Un teléfono móvil 920 incluye una antena 921, una unidad de comunicación 922, un Códec de audio 923, un altavoz 924, un micrófono 925, una unidad de cámara 926, un procesador de imagen 927, un demultiplexor 928,

una unidad de grabación/reproducción 929, una unidad de pantalla 930, un controlador 931, una unidad de operación 932, y un bus 933.

[0449]

La antena 921 está conectada a la unidad de comunicación 922. El altavoz 924 y el micrófono 925 están conectados al Códec de audio 923. La unidad operación 932 está conectada al controlador 931. El bus 933 conecta la unidad de comunicación 922, el Códec de audio 923, la unidad de cámara 926, el procesador de imagen 927, el demultiplexor 928, la unidad de grabación/reproducción 929, la unidad de pantalla 930, y el controlador 931 uno al otro.

[0450]

El teléfono móvil 920 realiza la operación, como la transmisión/recepción de una señal de audio, transmisión/recepción de un e-mail o datos de imagen, la toma de la imagen, y el registro de datos en varios modos de operación que incluyen un modo de comunicación de audio, un modo de comunicación de datos, un modo de imágenes, un modo de televisión-teléfono.

[0451]

En el modo de comunicación de audio, una señal de audio analógica generada por el micrófono 925 es suministrada al Códec de audio 923. El Códec de audio 923 convierte la señal de audio analógica a los datos de audio y hace la conversión A/D de los datos de audio convertidos para comprimir. Entonces, el Códec de audio 923 envía los datos de audio comprimidos a la unidad de comunicación 922. La unidad de comunicación 922 codifica y modula los datos de audio para generar una señal de transmisión. Entonces, la unidad de comunicación 922 transmite la señal de transmisión generada a una estación base (no ilustrada) a través de la antena 921. Por otra parte, la unidad de comunicación 922 amplifica una señal inalámbrica recibida a través de la antena 921 y aplica la conversión de frecuencia a la misma para obtener una señal de recepción. Entonces, la unidad de comunicación 922 genera los datos de audio por demodulación y decodificación de la señal de recepción y envía los datos de audio generados al Códec de audio 923. El Códec de audio 923 expande los datos de audio y Hace la conversión D/A de los mismos para generar la señal de audio analógica. Más adelante, el Códec de audio 923 suministra la señal de audio generada al altavoz 924 para permitir que el mismo de salida al audio.

[0452]

En el modo de comunicación de datos, por ejemplo, el controlador 931 genera la composición de datos de carácter del e-mail de acuerdo con la operación por el usuario a través

de la unidad de operación 932. Por otra parte, el controlador 931 permite que la unidad de pantalla 930 visualice los caracteres. El controlador 931 genera datos de e-mail de acuerdo con una instrucción de transmisión desde el usuario a través de la unidad operación 932 envía los datos de e-mail generados a la unidad de comunicación 922. La unidad de comunicación 922 codifica y modula los datos de e-mail para generar la señal de transmisión. Entonces, la unidad de comunicación 922 transmite la señal de transmisión generada a la estación base (no ilustrada) a través de la antena 921. Por otra parte, la unidad de comunicación 922 amplifica la señal inalámbrica recibida a través de la antena 921 y aplica la conversión de frecuencia a la misma para obtener la señal de recepción. Entonces, la unidad de comunicación 922 demodula y decodifica la señal de recepción para restaurar los datos de e-mail y envía los datos de e-mail restaurados al controlador 931. El controlador 931 permite que la unidad de pantalla 930 visualice el contenido de los datos de e-mail y permite que el medio de almacenamiento de la unidad de grabación/reproducción 929 almacene los datos de e-mail.

[0453]

La unidad de grabación/reproducción 929 incluye un medio arbitrario de almacenamiento de lectura/escritura. Por ejemplo, el medio de almacenamiento puede ser un medio de almacenamiento incorporado tal como la memoria RAM y la memoria flash y puede ser un medio de almacenamiento montado externamente tal como el disco duro, el disco magnético, disco magneto-óptico, el disco óptico, o una memoria USB, y una tarjeta de memoria

[0454]

En el modo de imágenes, por ejemplo, la unidad de cámara 926 toma una imagen de un objeto para generar los datos de imagen y envía los datos de imagen generados al procesador de imagen 927. El procesador de imagen 927 codifica la entrada de datos de imagen de la unidad de cámara 926 y almacena la corriente codificada en el medio de almacenamiento de la unidad de grabación/reproducción 929.

[0455]

Por otra parte, en el modo de televisión-teléfono, por ejemplo, el demultiplexor 928 multiplexa la corriente de vídeo codificada por el procesador de imagen 927 y la entrada de la corriente de audio desde el Códec 923 y envía la corriente multiplexada a la unidad comunicación 922. La unidad comunicación 922 codifica y modula la corriente para generar la señal transmisión. Entonces, la unidad comunicación 922 transmite la señal de transmisión generada a la estación base (no ilustrada) a través de la antena 921. Por otra parte, la unidad de comunicación 922 amplifica la señal inalámbrica recibida a través de la antena 921 y aplica la conversión de frecuencia a la misma para obtener la señal de recepción. La señal transmisión y la señal de recepción pueden incluir la corriente de bit codificada. Entonces, la unidad de

comunicación 922 restaura la corriente por demodulación y decodificación de la señal de recepción y envía ñas corriente restaurada al demultiplexor 928. El demultiplexor 928 separa la corriente de vídeo y la corriente de audio de la corriente de entrada y envía la corriente de vídeo y la corriente de audio al procesador de imagen 927 y el Código de audio 923, respectivamente. El procesador de imagen 927 decodifica la corriente de video para generar los datos de vídeo. Los datos de vídeo se suministran a la unidad de pantalla 930 y se muestra una serie de imágenes por la unidad de pantalla 930. El Códec de audio 923 expande la corriente de audio y hace la conversión D/A de la misma para generar la señal de audio analógica. Más adelante, el Códec de audio 923 suministra la señal de audio generada al altavoz 924 para dar salida al audio.

[0456]

En el teléfono móvil 920 configurado de esta manera, el procesador de imagen 927 tiene las funciones del dispositivo de codificación de imagen 10 y el dispositivo de decodificación de imagen 50 según la implementación descrita anteriormente. Por lo tanto, cuando las imagen se codifican y decodifican en el teléfono móvil 920, el proceso de filtración de desbloqueo se puede aplicar adecuadamente.

[0457]

[Tercer ejemplo de aplicación]

La figura 37 ilustra un ejemplo de una configuración esquemática del dispositivo de grabación/reproducción al que se le aplica la implementación descrita anteriormente. El dispositivo de grabación/reproducción 940 codifica los datos de audio y los datos de video de un programa de radiodifusión recibido para grabar sobre un medio de grabación, por ejemplo. Por otra parte, el dispositivo de grabación/reproducción 940 puede codificar los datos de audio y los datos de vídeo obtenidos de otro aparato para grabarlos en el medio de grabación, por ejemplo. Por otra parte, el dispositivo de grabación/reproducción 940 reproduce los datos grabados en el medio de grabación por un monitor y el altavoz de acuerdo a las instrucciones del usuario. En este caso, el dispositivo de grabación/reproducción 940 decodifica los datos de audio y los datos de vídeo.

[0458]

El dispositivo de grabación/reproducción 940 incluye un sintonizador 941, una interfaz externa 942, un codificador 943, un HDD (Hard Disk Drive) 944, una unidad de disco 945, un selector 946, un decodificador 947, una OSD (On-Screen Display) 948, un controlador 949, y una interfaz de usuario 950.

[0459]

El sintonizador 941 extrae una señal de un canal deseado a partir de la señal de emisión recibida a través de una antena (no ilustrado) y demodula la señal extraída. Entonces, el sintonizador 941 envía la corriente de bit codificada obtenida por la demodulación al selector 946. Es decir, el sintonizador 941 sirve como el medio de transmisión en el dispositivo de grabación/reproducción 940.

[0460]

La interfaz externa 942 es la interfaz para conectar el dispositivo de grabación/reproducción 940 y el dispositivo externo o la red. La interfaz externa 942 puede ser una interfaz IEEE 1394, una interfaz de red, una interfaz USB, una interfaz de memoria flash y similares, por Ejemplo. Por ejemplo, los datos de vídeo y los datos de audio recibidos a través de la interfaz externa 942 se introducen en el codificador 943. Es decir, la interfaz externa 942 sirve como medio de transmisión en el dispositivo de grabación/reproducción 940.

[0461]

El codificador 943 codifica los datos de vídeo y los datos de audio cuando no se codifican los datos de vídeo y la entrada de datos de audio desde la interfaz externa 942. Entonces, el codificador 943 envía la corriente de bit codificada al selector 946.

[0462]

Los HDD 944 graban la corriente de bit codificada en la que se comprimen los datos de contenido tales como el vídeo y el audio, varios programas y otros datos en un disco duro interno. El HDD 944 lee los datos desde el disco duro cuando se reproduce el vídeo y el audio.

[0463]

La unidad de disco 945 lee y registra los datos en y desde el medio de grabación montado. El medio de grabación montado en la unidad de disco 945 puede ser un disco DVD (DVD -Video, DVD -RAM, DVD -R, DVD- RW, DVD+R, DVD+RW y los similares), un disco Blu-ray (marca registrada) y similares, por ejemplo. El selector 946 selecciona y codifica la entrada de corriente de bit desde el sintonizador 941 o el codificador 943 y envía la corriente de bit codificada seleccionada al HDD 944 o a la unidad de disco 945 cuando graba el vídeo y el audio. Por otra parte, el selector 946 envía la entrada de corriente de bit codificada desde el HDD 944 o la unidad de disco 945 al decodificador 947 cuando se reproduce el vídeo y el audio.

[0464]

El decodificador 947 decodifica la corriente de bit codificada para generar los datos de vídeo y los datos de audio. Luego, el decodificador 947 envía los datos de vídeo generados en el OSD 948. Por otra parte, el decodificador 904 envía los datos de audio generados a un

altavoz externo. El OSD 948 reproduce la entrada de datos de vídeo desde el decodificador 947 para mostrar el vídeo. El OSD 948 también puede superponer la imagen GUI tal como el menú, el botón y el cursor, por ejemplo, sobre el video mostrado.

[0465]

El controlador 949 incluye el procesador como la CPU y la memoria, tal como la RAM y la ROM. La memoria almacena el programa ejecutado por la CPU, los datos del programa y otros similares. El programa almacenado en la memoria es leído por la CPU para ser ejecutado con la activación del dispositivo de grabación/reproducción 940, por ejemplo. La CPU controla la operación del dispositivo de grabación/reproducción 940 de acuerdo a una entrada de señal de operación de la interfaz de usuario 950, por ejemplo, mediante la ejecución del programa.

[0466]

La interfaz de usuario 950 está conectada al controlador 949. La interfaz de usuario 950 incluye un botón y un interruptor para que el usuario opere el dispositivo de grabación/reproducción 940 y un receptor de una señal de control remoto, por ejemplo. La interfaz de usuario 950 detecta la operación por parte del usuario a través de los componentes para generar la señal de operación y envía la señal de operación generada al controlador 949.

[0467]

En el dispositivo de grabación/reproducción 940 configurado de esta manera, el codificador 943 tiene las funciones del dispositivo de codificación de imagen 10 según la implementación descrita anteriormente. Por otra parte, el decodificador 947 tiene las funciones del dispositivo de decodificación de imagen 50 según la implementación descrita anteriormente. Por lo tanto, cuando las imágenes se codifican y decodifican en el dispositivo de grabación/reproducción 940, el proceso de filtración de desbloqueo puede aplicar adecuadamente la filtración.

[0468]

[Cuarto ejemplo de aplicación]

La figura 38 ilustra un ejemplo de una configuración esquemática de un dispositivo de formación de imágenes al que se aplica la implementación descrita anteriormente. Un dispositivo de formación de imágenes 960 forma la imagen de un objeto para generar la imagen, codifica los datos de imagen, y graba la misma en un medio de grabación.

[0469]

El dispositivo de formación de imágenes 960 incluye un bloque óptico 961, una unidad formación de imagen 962, un procesador de señales 963, un procesador de imagen 964, una unidad de visualización 965, una interfaz externa 966, una memoria 967, una unidad de medios 968, un OSD 969, un controlador 970, una interfaz de usuario 971, y un bus 972.

[0470]

El bloque óptico 961 incluye una lente de enfoque, un mecanismo de diafragma, y similares. El bloque óptico 961 forma una imagen óptica del objeto en una superficie de formación de imagen de la unidad de formación de imágenes 962. La unidad de formación de imágenes 962 incluye un sensor de imagen tal como un CCD y un CMOS y convierte la imagen óptica formada sobre la superficie de formación de imágenes en una señal de imagen como una señal eléctrica por conversión fotoeléctrica. Entonces, la unidad de formación de imágenes 962 envía la señal de imagen al procesador de señal 963.

[0471]

El procesador de señal 963 realiza varios procesos de señal de cámara tales como la corrección de rodilla, corrección gamma, o corrección de color a la entrada de la señal de imagen de la unidad de formación de imágenes 962. El procesador de señal 963 envía los datos de imagen después del proceso de señal de la cámara al procesador de imagen 964.

[0472]

El procesador de imagen 964 codifica la entrada de los datos de imagen del procesador de señal 963 para generar los datos codificados. Entonces, el procesador de imagen 964 envía los datos codificados generados a la interfaz externa 966 o a la unidad de medios 968. Por otra parte, el procesador de imagen 964 decodifica la entrada de datos codificados de la interfaz externa 966 o la unidad de medios 968 para generar los datos de imagen. Luego, el procesador de imagen 964 envía los datos de imagen generados a la unidad de visualización 965. El procesador de imagen 964 también puede enviar la entrada de datos de imagen desde el procesador de señal 963 a la unidad de visualización 965 para visualizar la imagen. El procesador de imagen 964 también puede superponer los datos para la visualización obtenida desde el OSD 969 en la salida de imagen a la unidad de visualización 965.

[0473]

El OSD 969 genera la imagen GUI tal como por ejemplo el menú, el botón, y el cursor, y envía la imagen generada al procesador de imagen 964.

[0474]

La interfaz externa 966 está configurada como una terminal USB de entrada/salida, por ejemplo. La interfaz externa 966 conecta el dispositivo de formación de imagen 960 y una

impresora imprime la imagen, por ejemplo. Por otra parte, una unidad está conectada a la interfaz externa 966 según sea necesario. Los medios extraíbles tales como disco magnético y disco óptico se montan en la unidad, por ejemplo, y el programa leído desde el medio extraíble se puede instalar en el dispositivo de formación de imágenes 960. Además, la interfaz externa 966 se puede configurar como una interfaz de red conectada a la red como una LAN y la Internet. Es decir, la interfaz externa 966 sirve como el medio de transmisión en el dispositivo de formación de imágenes 960.

[0475]

El medio de grabación montado en la unidad de medios 968 puede ser un medio arbitrariamente de lectura/escritura tal como el disco magnético, disco magneto-óptico, el disco óptico, y la memoria semiconductora, por ejemplo. Por otra parte, el medio de grabación puede ser montado de forma fija en la unidad de medios 968 para formar una unidad de almacenamiento no portátil tal como una unidad de disco duro incorporado o SSD (Solid State Drive), por ejemplo.

[0476]

El controlador 970 incluye el procesador como la CPU y la memoria, como la memoria RAM y la ROM. La memoria almacena el programa ejecutado por la CPU y los datos del programa. El programa almacenado en la memoria es leído por la CPU en el arranque del dispositivo de formación de imagen 960 para ser ejecutado, por ejemplo. La CPU controla la operación del dispositivo de formación de imágenes 960 de acuerdo con la entrada de señal de operación desde la interfaz de usuario 971, por ejemplo, mediante la ejecución del programa.

[0477]

La interfaz de usuario 971 está conectada al controlador 970. La interfaz de usuario 971 incluye un botón, un interruptor y similares para que el usuario opere el dispositivo de formación de imágenes 960, por ejemplo. La interfaz de usuario 971 detecta la operación por el usuario a través de los componentes para generar la señal de operación y envía la señal de operación generada al controlador 970.

[0478]

El bus 972 conecta el procesador de imagen 964, la interfaz externa 966, la memoria 967, la unidad de medios 968, el OSD 969, y el controlador 970 el uno al otro.

[0479]

En el dispositivo de formación de imagen 960 configurado de esta manera, el procesador de imagen 964 tiene las funciones del dispositivo de codificación de imagen 10 y el dispositivo

de decodificación de imagen 50 según la implementación descrita anteriormente. Por lo tanto, cuando las imágenes se codifican y decodifican en el dispositivo de formación de imágenes 960, el proceso de filtración de desbloqueo puede aplicar adecuadamente la filtración.

[0480]

En la presente especificación, se ha descrito un ejemplo en el que se multiplexan varios tipos de información, tal como los parámetros relacionados con el proceso de recorte para la filtración fuerte en corrientes codificadas y se transmiten desde el lado de codificación hasta el lado de decodificación. Sin embargo, un método de transmisión de estos elementos de información no se limita a este ejemplo. Por ejemplo, estos elementos de información se pueden transmitir o grabar como datos separados asociados con la corriente de bit codificada en lugar de ser multiplexados en la corriente de bit codificada. Aquí, el término “asociado” significa que la imagen (o parte de la imagen tal como un segmento y un bloque) incluidos en la corriente de bit y la información correspondiente a la imagen se pueden unir entre sí en el momento de la decodificación. Es decir, la información puede ser transmitida en una línea de transmisión que no sea la de la imagen (o corriente de bit). Por otra parte, la información se puede grabar sobre otro medio de grabación (o de otra zona de grabación del mismo medio de grabación), excepto el de la imagen (o corriente de bit). Además, la información y la imagen (o corriente de bit) pueden estar asociados entre sí en unidades opcionales tal como una pluralidad de cuadros, una trama, o una parte de la trama, por ejemplo.

[0481]

Además, la presente técnica no debe ser interpretada como limitada a las implementaciones descritas anteriormente. Las implementaciones divulgan la presente técnica en una ilustrativa y de ejemplo, y es obvio que aquellos versados en el arte puedan hacer modificaciones y sustituciones sin que se aparten del espíritu de la presente técnica. En otras palabras, el espíritu de la presente técnica se debe determinar en consideración de las reivindicaciones.

[0482]

Tenga en cuenta que el dispositivo de procesamiento de imágenes de esta técnica puede incluir las siguientes constituciones. (1) Un dispositivo de procesamiento de imagen que incluye:

una unidad de decodificación que decodifica una corriente codificada, codificada en unidades que tienen una estructura de capa para generar una imagen;

una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es in límite

entre un bloque de la imagen generada por el unidad de decodificación y un bloque adyacente, adyacente al bloque; y

un controlador que controla la unidad de filtración de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación. (2) El dispositivo de procesamiento de imagen de acuerdo con (1), en el que, el controlador aplica el proceso de recorte a los valores de las partes que cambian cuando se aplica el filtro de desbloqueo. (3) El dispositivo de procesamiento de imagen de acuerdo con (2), en el que el controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones:

[Fórmula Matemática 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0, 7 y pv es un valor de recorte.

(4) El dispositivo de procesamiento de imagen de acuerdo con (3), en el que

el controlador establece un valor que es un múltiplo entero de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se aplica el proceso de recorte.

(5) El dispositivo de procesamiento de imagen de acuerdo con (4), en el que,

el controlador establece un valor que es el doble de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se realiza el proceso de recorte.

(6) El dispositivo de procesamiento de imagen de acuerdo con cualquiera de (1) a (5), que incluye además:

Una unidad que determina la fuerza del filtro, que determina la fuerza del filtro de desbloqueo aplicado al límite de bloque, en donde

la unidad de filtración aplica el filtro de desbloqueo al límite de bloque de acuerdo con la fuerza determinada por la unidad que determina la fuerza del filtro, y

el controlador controla la unidad de filtración de manera que, cuando la unidad que determina la fuerza filtración, determina que se va a aplicar filtración fuerte, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

(7) El dispositivo de procesamiento de imagen de acuerdo con (6), en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando una pluralidad de líneas como unidades de procesamiento.

(8) El dispositivo de procesamiento de imagen de acuerdo con (7), en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando cuatro líneas como unidades de procesamiento.

(9) El dispositivo de procesamiento de imagen de acuerdo con (7), que incluye además:

la unidad que determina la necesidad de filtración, que determina si el filtro de desbloqueo se va a aplicar al límite de bloque utilizando una pluralidad de líneas como unidades de procesamiento, en el que la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que se va a aplicar el filtro de desbloqueo.

(10) Un método de procesamiento de imagen para permitir que un dispositivo de procesamiento de imagen ejecute:

La decodificación de una corriente codificada, codificada en unidades que tienen una estructura de capa para generar una imagen;

Aplicar un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de la imagen generada y un bloque adyacente, adyacente al bloque; y

Controlar de tal manera que cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada.

(11) un dispositivo de procesamiento de imagen que incluye:

Una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de una imagen decodificada localmente cuando una imagen se codifica y un bloque adyacente, adyacente al bloque;

Un controlador que controla la unidad de filtración de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente; y

Una unidad de codificación que codifica la imagen en unidades que tienen una estructura de capa utilizando una imagen a la que se aplica el filtro de desbloqueo

(12) el dispositivo de procesamiento de imagen de acuerdo con (11), en el que,

El controlador aplica el proceso de recorte a valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

(13) el dispositivo de procesamiento de imagen de acuerdo con (12), en el que,

El controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones

[Fórmula Matemática 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0,7 y pv es un valor de recorte.

(14) El dispositivo de procesamiento de imagen de acuerdo con la (13), en el que,

el controlador establece un valor que es múltiplo entero de los parámetros del filtro de desbloqueo como el valor de recorte utilizado cuando se realiza el proceso de recorte.

(15) El dispositivo de procesamiento de imagen de acuerdo con la (14), en el que,

el controlador establece un valor que es el doble de los parámetros del filtro de desbloqueo como el valor de recorte usado cuando se realiza el proceso de recorte.

(16) El dispositivo de procesamiento de imagen de acuerdo con la cualquiera de (11) a (15) que incluye además:

Una unidad que determina la fuerza del filtro, que determina la fuerza del filtro de desbloqueo aplicada al límite de bloque donde la unidad de filtración aplica el filtro de desbloqueo al límite de bloque de acuerdo con la fuerza determinada por la unidad que determina la fuerza del filtro; y

el controlador controla la unidad de filtración de tal manera que,

cuando la unidad que determina la fuerza del filtro determina que se va a aplicar filtración fuerte, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por el unidad de decodificación.

(17) El dispositivo de procesamiento de imagen de acuerdo con (16), en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo utilizando una pluralidad de líneas como unidades de procesamiento.

(18) El dispositivo de procesamiento de imagen de acuerdo con (17), en el que,

La unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando cuatro líneas como las unidades de procesamiento.

(19) El dispositivo de procesamiento de imagen de acuerdo con (17), que incluye además:

una unidad que determina la necesidad de filtración que determina si se va a aplicar el filtro de desbloqueo al límite de bloque utilizando una pluralidad de líneas como unidades de procesamiento, en donde

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que se va a aplicar el filtro de desbloqueo.

(20) Un método de procesamiento de imagen para permitir que un dispositivo de procesamiento de imagen ejecute:

La aplicación de un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de una imagen decodificada localmente cuando una imagen se codifica y un bloque adyacente, adyacente al bloque;

Controlar de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente; y

codificar la imagen en unidades que tengan una estructura de capa utilizando una imagen a la que se le aplica el filtro de desbloqueo.

LISTA DE REFERENCIA DE SIGNOS

[0483]

Dispositivo de codificación de imagen 10
Convertidor analógico/digital (convertidor A/D) 11
Buffer de reordenamiento de cuadro 12, 57
Sustractor 13
Transformador ortogonal 14
Cuantizador 15
Cuantizador sin pérdidas 16
Buffer de acumulación 17
Controlador de tasa 18
Cuantizador inverso 21, 53
Transformador ortogonal inverso 22, 54
Sumador 23, 55
Unidad de filtración de desbloqueo 24, 56
Memoria de cuadro 25, 61
Selector 26, 62, 65
Predictor intra 31, 63
Estimador de movimiento/compensador 32

Unidad de selección de imagen de predicción/modo optimo 33
Dispositivo de decodificación de imagen 50
Buffer de acumulación 51,
Decodificador sin pérdidas 52,
Memoria de cuadro 61,
Unidad de compensación de movimiento 64
Memoria de imagen 71,
Unidad que determina el límite de bloque 72, 72-1
Unidad que determina el límite de línea/límite de bloque 72-2,
Unidad que determina la fuerza del filtro 73, 73-1
Unidad que determina la fuerza del filtro del límite de línea 73-2
Unidad de operación del filtro 74, 74-1
Unidad de operación del filtro del límite de línea 74-2.
Selector 75.
Memoria de coeficiente 76
Controlador 77
Unidad que determina la línea intra-LCU 77A, 77B
Memoria de línea 241
Unidad que detecta el límite de línea 242
Unidad que determina la fuerza del filtro 243
Memoria de coeficiente 244
Unidad de operación de filtro 245
Controlador de filtro 246
Unidad de almacenamiento de datos 2451
Unidad de selección de datos 2452
Unidad de procesamiento aritmético 2453, 2455, 2456
Unidad de selección de datos 2457
Unidad que determina el límite de línea 2461

REIVINDICACIONES

1. Un dispositivo de procesamiento de imagen que comprende:

una unidad de decodificación que decodifica una corriente codificada, codificada en unidades que tienen una estructura de capa para generar una imagen;

una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es in límite entre un bloque de la imagen generada por el unidad de decodificación y un bloque adyacente, adyacente al bloque; y

un controlador que controla la unidad de filtración de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

2. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 1, en el que,

el controlador aplica el proceso de recorte a los valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

3. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 2, en el que

el controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones:

[Fórmula Matemática 37]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0,7 y pv es un valor de recorte.

4. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 3 en el que,

el controlador establece un valor que es un múltiplo entero de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se aplica el proceso de recorte.

5. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 4 en el que,

el controlador establece un valor que es el doble de los parámetros del filtro de desbloqueo según el valor de recorte utilizado cuando se realiza el proceso de recorte.

6. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 1, que incluye además:

Una unidad que determina la fuerza del filtro, que determina la fuerza del filtro de desbloqueo aplicado al límite de bloque, en donde

la unidad de filtración aplica el filtro de desbloqueo al límite de bloque de acuerdo con la fuerza determinada por la unidad que determina la fuerza del filtro, y

el controlador controla la unidad de filtración de manera que, cuando la unidad que determina la fuerza filtración, determina que se va a aplicar filtración fuerte, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por la unidad de decodificación.

7. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 6 en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando una pluralidad de líneas como unidades de procesamiento.

8. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 7, en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando cuatro líneas como las unidades de procesamiento.

9. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 7, que incluye además:

la unidad que determina la necesidad de filtración, que determina si el filtro de desbloqueo se va a aplicar al límite de bloque utilizando una pluralidad de líneas como unidades de procesamiento, en el que la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que se va a aplicar el filtro de desbloqueo.

10. Un método de procesamiento de imagen para permitir que un dispositivo de procesamiento de imagen ejecute:

La decodificación de una corriente codificada, codificada en unidades que tienen una estructura de capa para generar una imagen;

Aplicar un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de la imagen generada y un bloque adyacente, adyacente al bloque; y

Controlar de tal manera que cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada.

11. un dispositivo de procesamiento de imagen que incluye:

Una unidad de filtración que aplica un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de una imagen decodificada localmente cuando una imagen se codifica y un bloque adyacente, adyacente al bloque;

Un controlador que controla la unidad de filtración de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente; y

Una unidad de codificación que codifica la imagen en unidades que tienen una estructura de capa utilizando una imagen a la que se aplica el filtro de desbloqueo.

12. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 1), en el que,

El controlador aplica el proceso de recorte a valores de las partes que cambian cuando se aplica el filtro de desbloqueo.

13. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 12, en el que,

El controlador aplica el proceso de recorte de acuerdo con las siguientes expresiones

[Fórmula Matemática 38]

$$p0_i = p0_i + Clip_{(-pv)-(pv)}((p2_i + 2*p1_i - 6*p0_i + 2*q0_i + q1_i + 4) >> 3);$$

$$q0_i = q0_i + Clip_{(-pv)-(pv)}((p1_i + 2*p0_i - 6*q0_i + 2*q1_i + q2_i + 4) >> 3);$$

$$p1_i = p1_i + Clip_{(-pv)-(pv)}((p2_i - 3*p1_i + p0_i + q0_i + 2) >> 2);$$

$$q1_i = q1_i + Clip_{(-pv)-(pv)}((p0_i + q0_i - 3*q1_i + q2_i + 2) >> 2);$$

$$p2_i = p2_i + Clip_{(-pv)-(pv)}((2*p2_i - 5*p2_i + p1_i + p0_i + q0_i + 4) >> 3);$$

$$q2_i = q2_i + Clip_{(-pv)-(pv)}((p0_i + q0_i + p1_i - 5*q2_i + 2*q3_i + 4) >> 3);$$

Donde i = 0,7 y pv es un valor de recorte.

14. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 13 en el que,

el controlador establece un valor que es múltiplo entero de los parámetros del filtro de desbloqueo como el valor de recorte utilizado cuando se realiza el proceso de recorte.

15. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 14 en el que,

el controlador establece un valor que es el doble de los parámetros del filtro de desbloqueo como el valor de recorte usado cuando se realiza el proceso de recorte.

16. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 11 que incluye además:

Una unidad que determina la fuerza del filtro, que determina la fuerza del filtro de desbloqueo aplicada al límite de bloque donde la unidad de filtración aplica el filtro de desbloqueo al límite de bloque de acuerdo con la fuerza determinada por la unidad que determina la fuerza del filtro; y

el controlador controla la unidad de filtración de tal manera que,

cuando la unidad que determina la fuerza del filtro determina que se va a aplicar filtración fuerte, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen generada por el unidad de decodificación.

17. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 16 en el que,

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo utilizando una pluralidad de líneas como unidades de procesamiento.

18. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 17 en el que,

La unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo usando cuatro líneas como las unidades de procesamiento.

19. El dispositivo de procesamiento de imagen de acuerdo con la reivindicación 17 que incluye además:

una unidad que determina la necesidad de filtración que determina si se va a aplicar el filtro de desbloqueo al límite de bloque utilizando una pluralidad de líneas como unidades de procesamiento, en donde

la unidad que determina la fuerza del filtro determina la fuerza del filtro de desbloqueo cuando la unidad que determina la necesidad de filtración determina que se va a aplicar el filtro de desbloqueo.

20. Un método de procesamiento de imagen para permitir que un dispositivo de procesamiento de imagen ejecute:

La aplicación de un filtro de desbloqueo a un límite de bloque de acuerdo con una fuerza del filtro de desbloqueo aplicada al límite de bloque que es un límite entre un bloque de una imagen decodificada localmente cuando una imagen se codifica y un bloque adyacente, adyacente al bloque;

Controlar de tal manera que, cuando se aplica filtración fuerte como la fuerza del filtro de desbloqueo, se aplica un proceso de recorte al filtro de desbloqueo con respecto a los componentes de luminancia de la imagen decodificada localmente; y

codificar la imagen en unidades que tengan una estructura de capa utilizando una imagen a la que se le aplica el filtro de desbloqueo.

RESUMEN

La presente tecnología se refiere a un dispositivo de procesamiento de imagen y un método de procesamiento de imagen que permite que un proceso de filtración de desbloqueo aplique adecuadamente la filtración. Un píxel ($p0_i$) del cual el valor antes del proceso de desbloqueo es 255 (línea continua) cambia ampliamente a 159 (línea de puntos) después del proceso de desbloqueo convencional. Por lo tanto, se realiza en la filtración fuerte un proceso de recorte que tiene un valor de recorte de 10, con lo cual el pixel (p0i) del cual el valor es 255 (línea continua) antes del proceso de desbloqueo llega a ser 245 (línea gruesa). Por lo tanto, se puede suprimir un cambio en el valor de píxel que ocurre en la técnica convencional tanto como es posible. Por ejemplo, esta divulgación se puede aplicar a un dispositivo de procesamiento de imagen.

FIG. 1

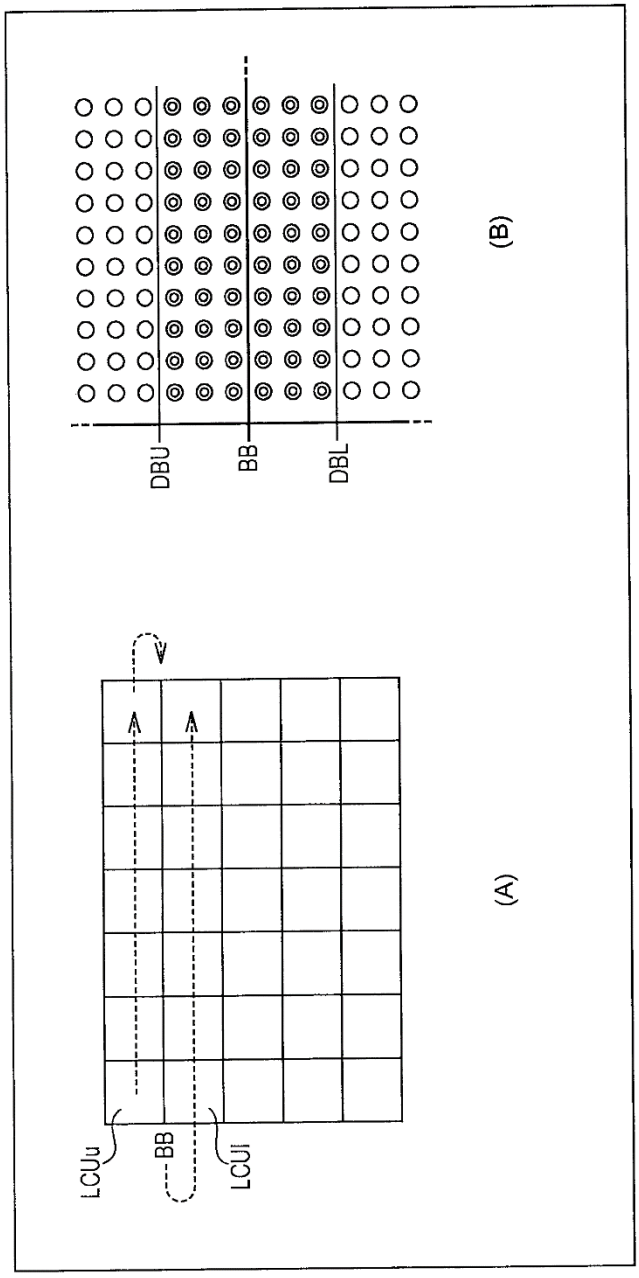


FIG. 2

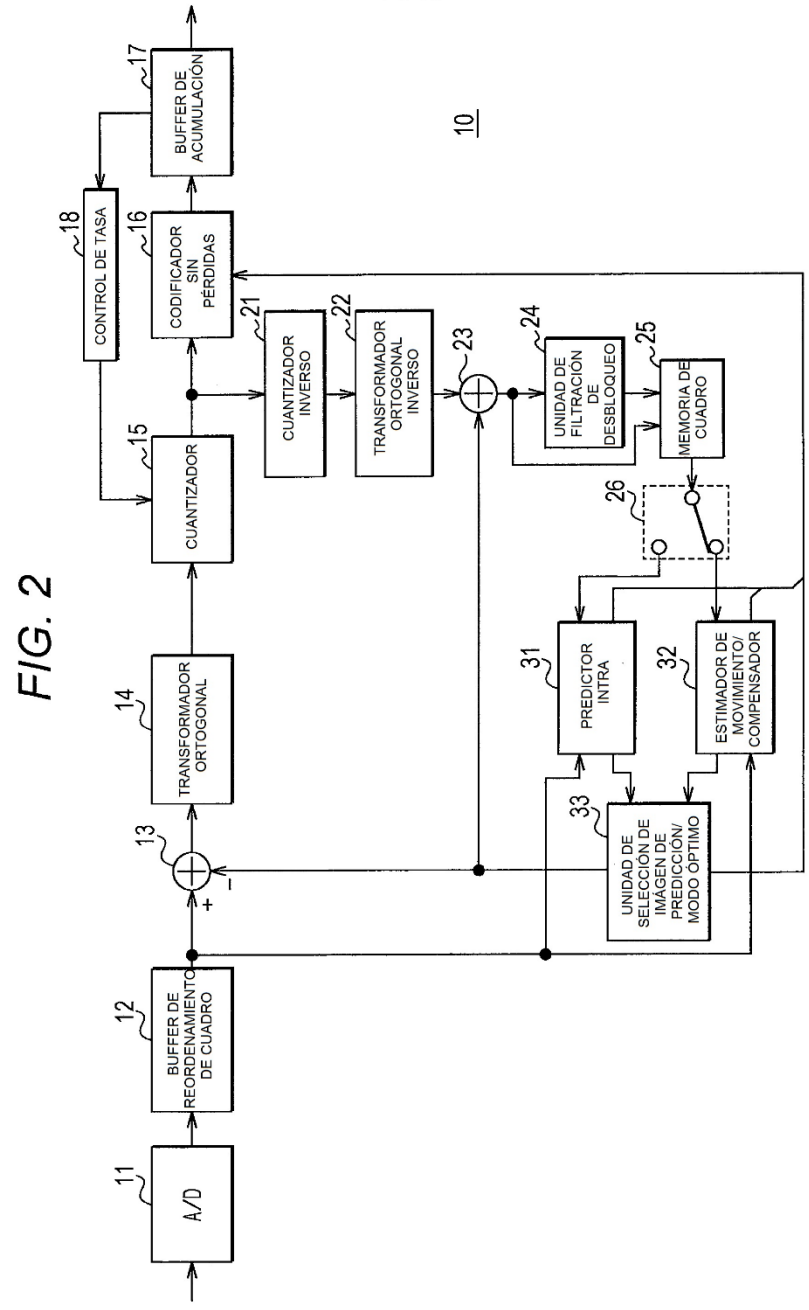

10
11
A/D
12
BUFFER DE REORDENAMIENTO DE CUADRO
13
14
TRANSFORMADOR ORTOGONAL
15
CUANTIZADOR
16
CODIFICADOR SIN PÉRDIDAS
17
BUFFER DE ACUMULACIÓN
18
CONTROL DE TASA
21
CUANTIZADOR INVERSO
22
TRANSFORMADOR ORTOGONAL INVERSO
23
24
UNIDAD DE FILTRACIÓN DE DESBLOQUEO
25
MEMORIA DE CUADRO
26
31
PREDICTOR INTRA
32
ESTIMADOR DE MOVIMIENTO/ COMPENSADOR
33
UNIDAD DE SELECCIÓN DE IMAGEN DE PREDICCIÓN/ MODO ÓPTIMO

FIG. 3

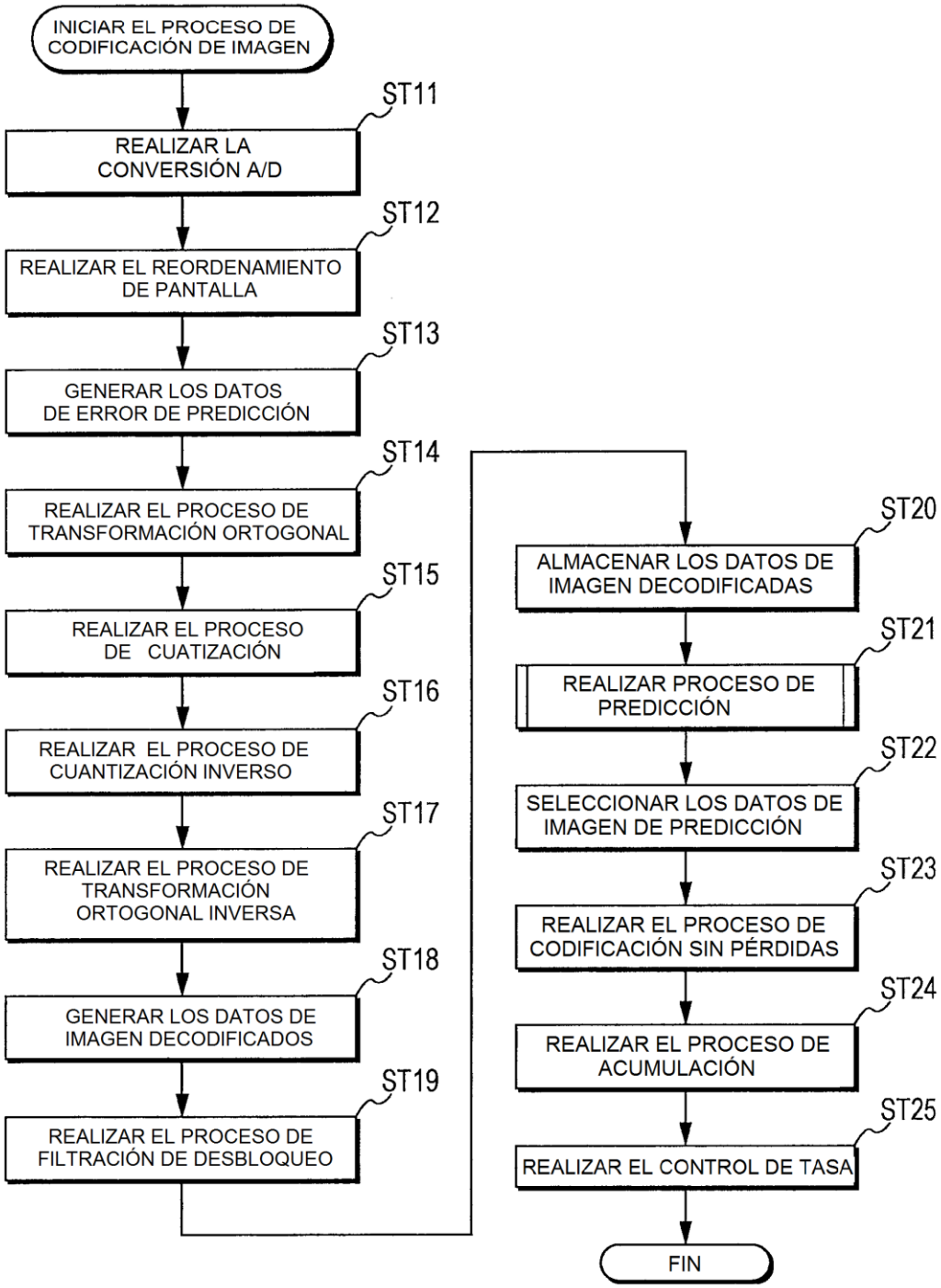

INICIAR EL PROCESO DE CODIFICACIÓN DE IMAGEN
ST11
REALIZAR LA CONVERSIÓN A/D
ST12
REALIZAR EL REORDENAMIENTO DE PANTALLA
ST13
GENERAR LOS DATOS DE ERROR DE PREDICCIÓN
ST14
REALIZAR EL PROCESO DE TRANSFORMACIÓN ORTOGONAL
ST15
REALIZAR EL PROCESO DE CUATIZACIÓN
ST16
REALIZAR EL PROCESO DE CUANTIZACIÓN INVERSO
ST17
REALIZAR EL PROCESO DE TRANSFORMACIÓN ORTOGONAL INVERSA
ST18
GENERAR LOS DATOS DE IMAGEN DECODIFICADOS
ST19
REALIZAR EL PROCESO DE FILTRACIÓN DE DESBLOQUEO
ST20
ALMACENAR LOS DATOS DE IMAGEN DECODIFICADAS
ST21
REALIZAR PROCESO DE PREDICCIÓN
ST22
SELECCIONAR LOS DATOS DE IMAGEN DE PREDICCIÓN
ST23
REALIZAR EL PROCESO DE CODIFICACIÓN SIN PÉRDIDAS
ST24
REALIZAR EL PROCESO DE ACUMULACIÓN
ST25
REALIZAR EL CONTROL DE TASA
FIN

FIG. 4

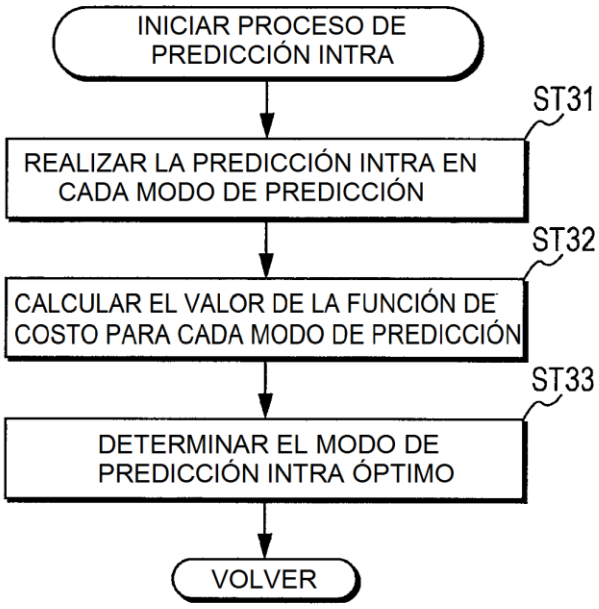

INICIAR PROCESO DE PREDICCIÓN INTRA
ST31
REALIZAR LA PREDICCIÓN INTRA EN CADA MODO DE PREDICCIÓN
ST32
CALCULAR EL VALOR DE LA FUNCIÓN DE COSTO PARA CADA MODO DE PREDICCIÓN
ST33
DETERMINAR EL MODO DE PREDICCIÓN INTRA ÓPTIMO
VOLVER

FIG. 5

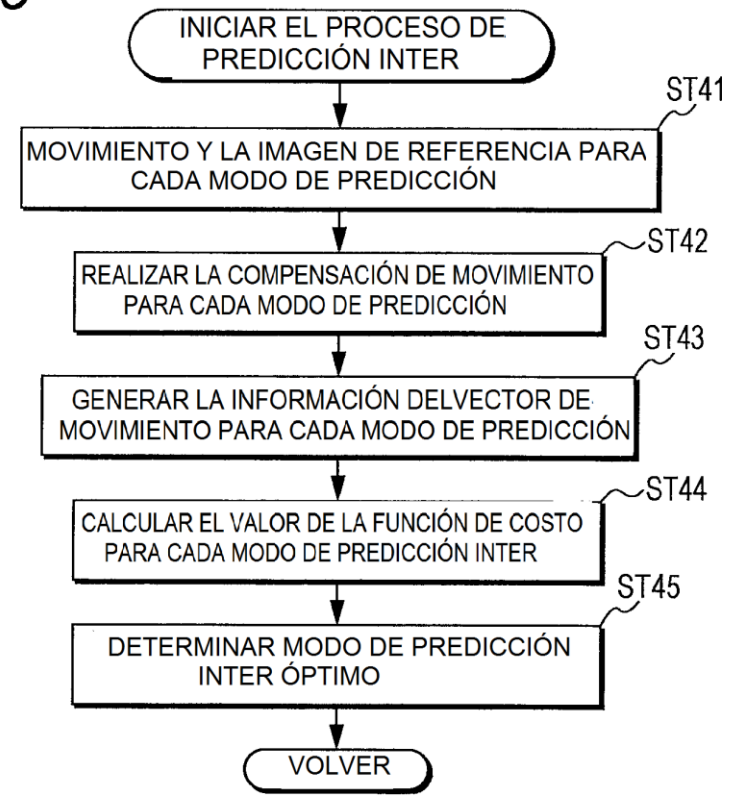

INICIAR EL PROCESO DE PREDICCIÓN INTER
ST41
MOVIMIENTO Y LA IMAGEN DE REFERENCIA PARA CADA MODO DE PREDICCIÓN
ST42
REALIZAR LA COMPENSACIÓN DE MOVIMIENTO PARA CADA MODO DE PREDICCIÓN
ST43
GENERAR LA INFORMACIÓN DELVECTOR DE MOVIMIENTO PARA CADA MODO DE PREDICCIÓN
ST44
CALCULAR EL VALOR DE LA FUNCIÓN DE COSTO PARA CADA MODO DE PREDICCIÓN INTER
ST45
DETERMINAR MODO DE PREDICCIÓN INTER ÓPTIMO
VOLVER

FIG. 6

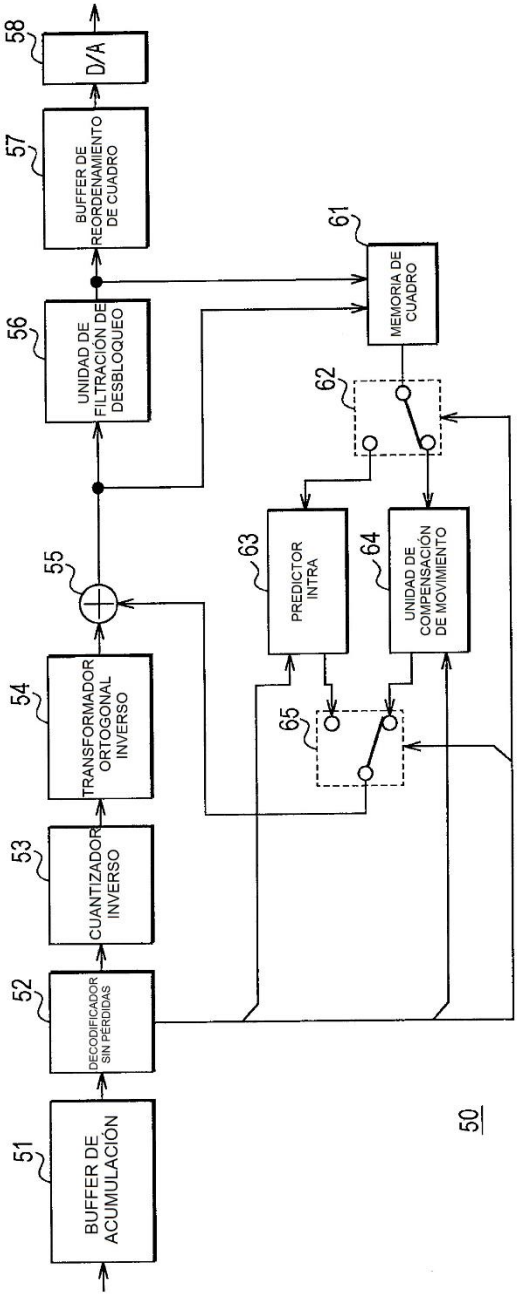

BUFFER DE ACUMULACIÓN
51
DECODIFICADOR SIN PÉRDIDAS
52
CUANTIZADOR INVERSO
53
TRANSFORMADOR ORTOGONAL INVERSO
54
55
UNIDAD DE FILTRACIÓN DE DESBLOQUEO
56
BUFFER DE REORDENAMIENTO DE CUADRO
57
D/A
58
MEMORIA DE CUADRO
61
62
PREDICTOR INTRA
63
UNIDAD DE COMPENSACIÓN DE MOVIMIENTO
64
65
50

FIG. 7

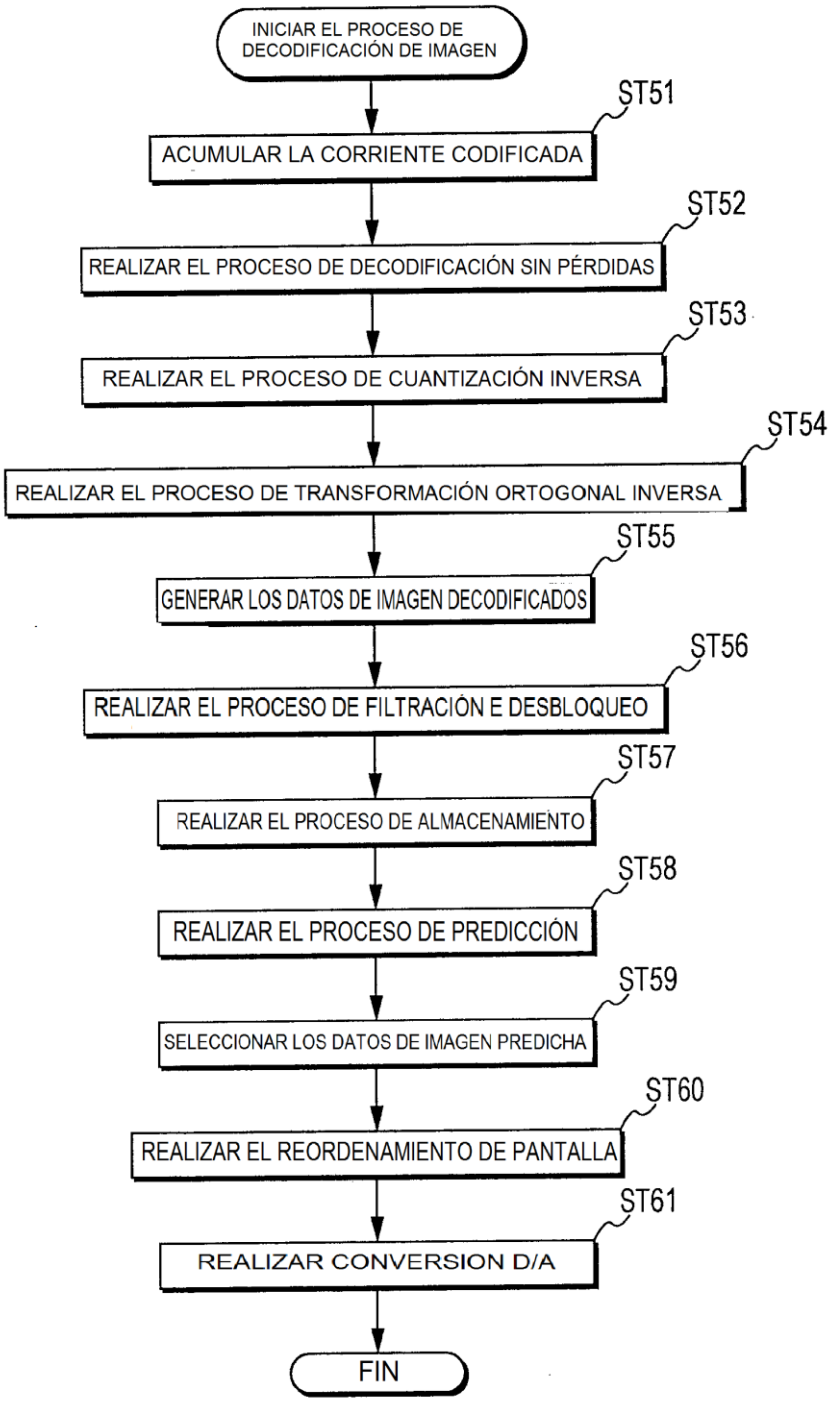

INICIAR EL PROCESO DE DECODIFICACIÓN DE IMAGEN
ST51
ACUMULAR LA CORRIENTE CODIFICADA
ST52
REALIZAR EL PROCESO DE DECODIFICACIÓN SIN PÉRDIDAS
ST53
REALIZAR EL PROCESO DE CUANTIZACIÓN INVERSA
ST54
REALIZAR EL PROCESO DE TRANSFORMACIÓN ORTOGONAL INVERSA
ST55
GENERAR LOS DATOS DE IMAGEN DECODIFICADOS
ST56
REALIZAR EL PROCESO DE FILTRACIÓN E DESBLOQUEO
ST57
REALIZAR EL PROCESO DE ALMACENAMIENTO
ST58
REALIZAR EL PROCESO DE PREDICCIÓN
ST59
SELECCIONAR LOS DATOS DE IMAGEN PREDICHA
ST60
REALIZAR EL REORDENAMIENTO DE PANTALLA
ST61
REALIZAR CONVERSION D/A
FIN

FIG. 8

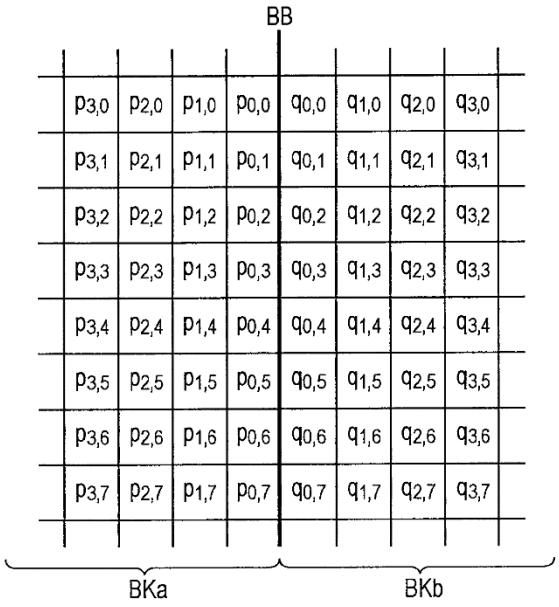

BB
p3,0 p2,0 p1,0 p0,0 q0,0 q1,0 q2,0 q3,0
p3,1 p2,1 p1,1 p0,1 q0,1 q1,1 q2,1 q3,1
p3,2 p2,2 p1,2 p0,2 q0,2 q1,2 q2,2 q3,2
p3,3 p2,3 p1,3 p0,3 q0,3 q1,3 q2,3 q3,3
p3,4 p2,4 p1,4 p0,4 q0,4 q1,4 q2,4 q3,4
p3,5 p2,5 p1,5 p0,5 q0,5 q1,5 q2,5 q3,5
p3,6 p2,6 p1,6 p0,6 q0,6 q1,6 q2,6 q3,6
p3,7 p2,7 p1,7 p0,7 q0,7 q1,7 q2,7 q3,7
BKa
BKb

FIG. 9

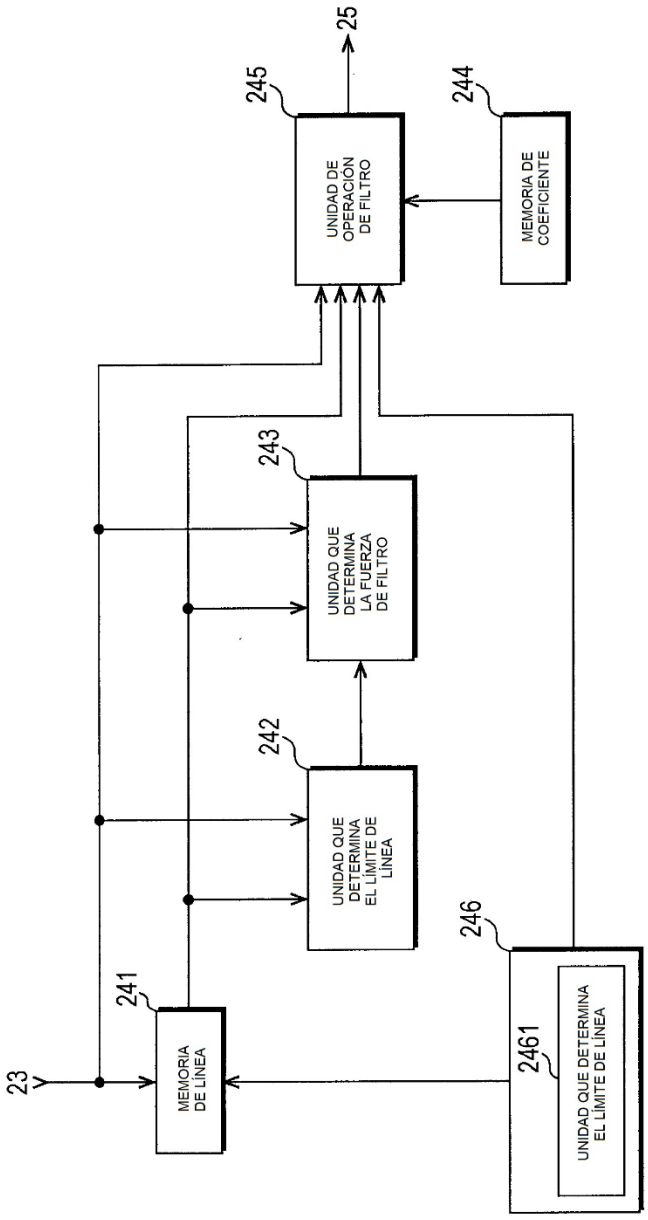

25
245
244
UNIDAD DE OPERACIÓN DE FILTRO
MEMORIA DE COEFICIENTE
243
UNIDAD QUE DETERMINA LA FUERZA DE FILTRO
242
UNIDAD QUE DETERMINA EL LÍMITE DE LÍNEA
246
241
2461
23
MEMORIA DE LÍNEA
UNIDAD QUE DETERMINA EL LÍMITE DE LÍNEA

FIG. 10

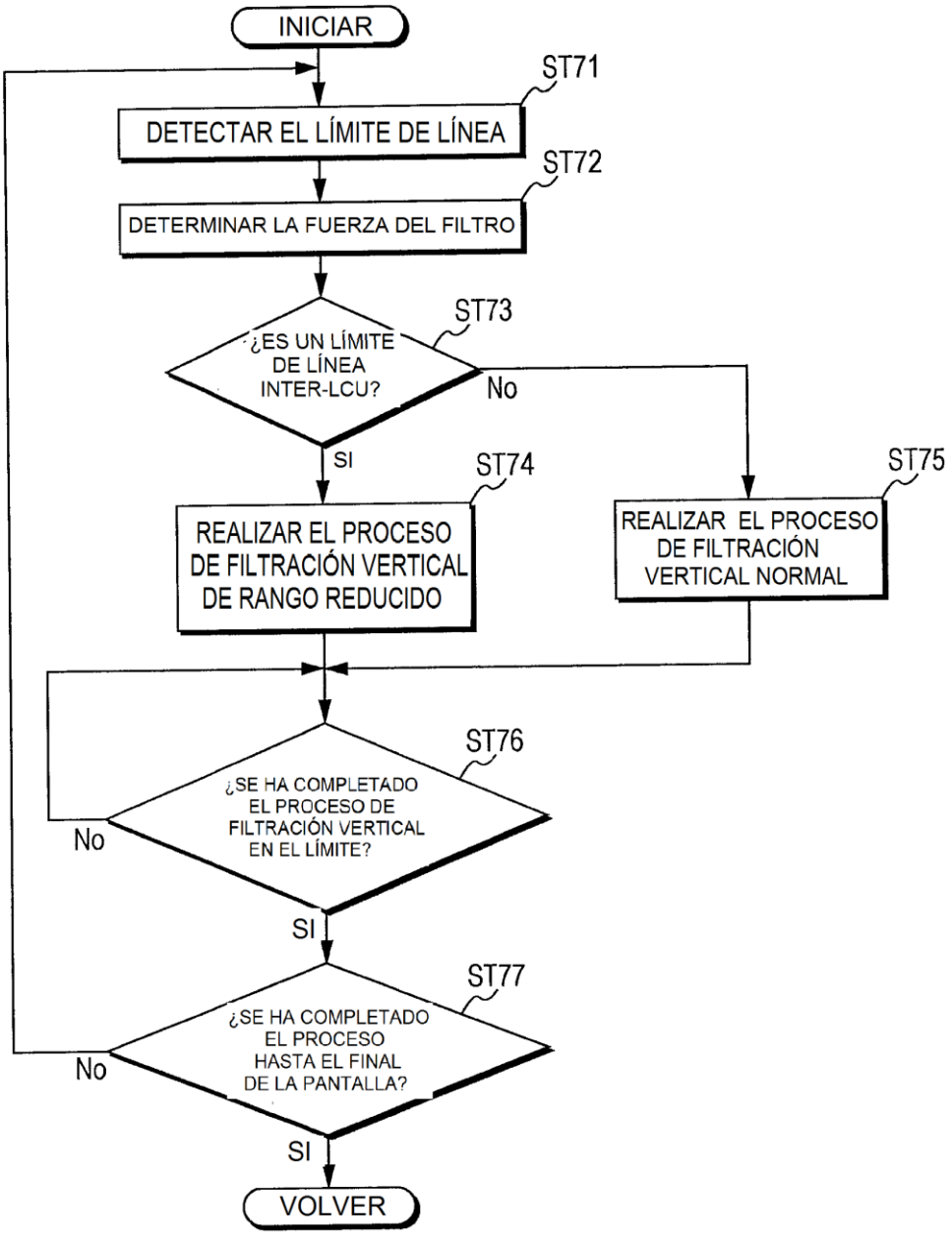

INICIAR
ST71
DETECTAR EL LÍMITE DE LÍNEA
ST72
DETERMINAR LA FUERZA DEL FILTRO
ST73
¿ES UN LÍMITE DE LÍNEA INTER-LCU?
No
SI
ST74
REALIZAR EL PROCESO DE FILTRACIÓN VERTICAL DE RANGO REDUCIDO
ST75
REALIZAR EL PROCESO DE FILTRACIÓN VERTICAL NORMAL
ST76
¿SE HA COMPLETADO EL PROCESO DE FILTRACIÓN VERTICAL EN EL LÍMITE?
No
SI
ST77
¿SE HA COMPLETADO EL PROCESO HASTA EL FINAL DE LA PANTALLA?
No
SI
VOLVER

FIG. 11

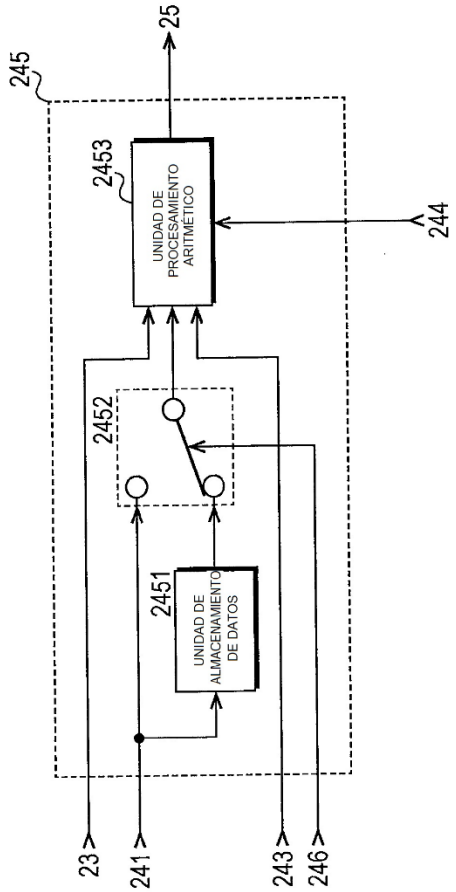

245
25
2453
UNIDAD DE PROCESAMIENTO ARITMÉTICO
244
2452
2451
UNIDAD DE ALMACENAMIENTO DE DATOS
23
241
243
246

FIG. 12

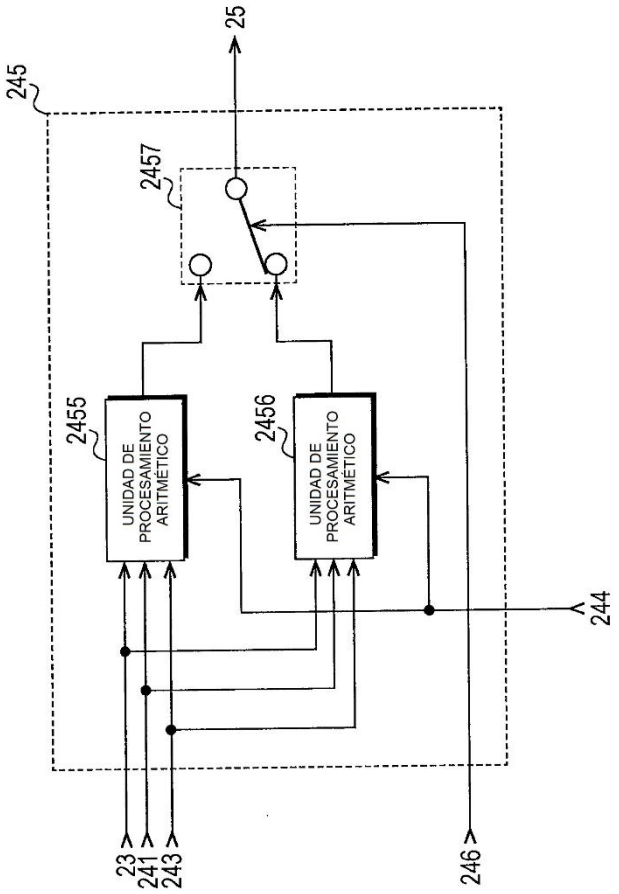

25
245
2457
2455
UNIDAD DE PROCESAMIENTO ARITMÉTICO
2456
UNIDAD DE PROCESAMIENTO ARITMÉTICO
244
23
241
243
246

FIG. 13

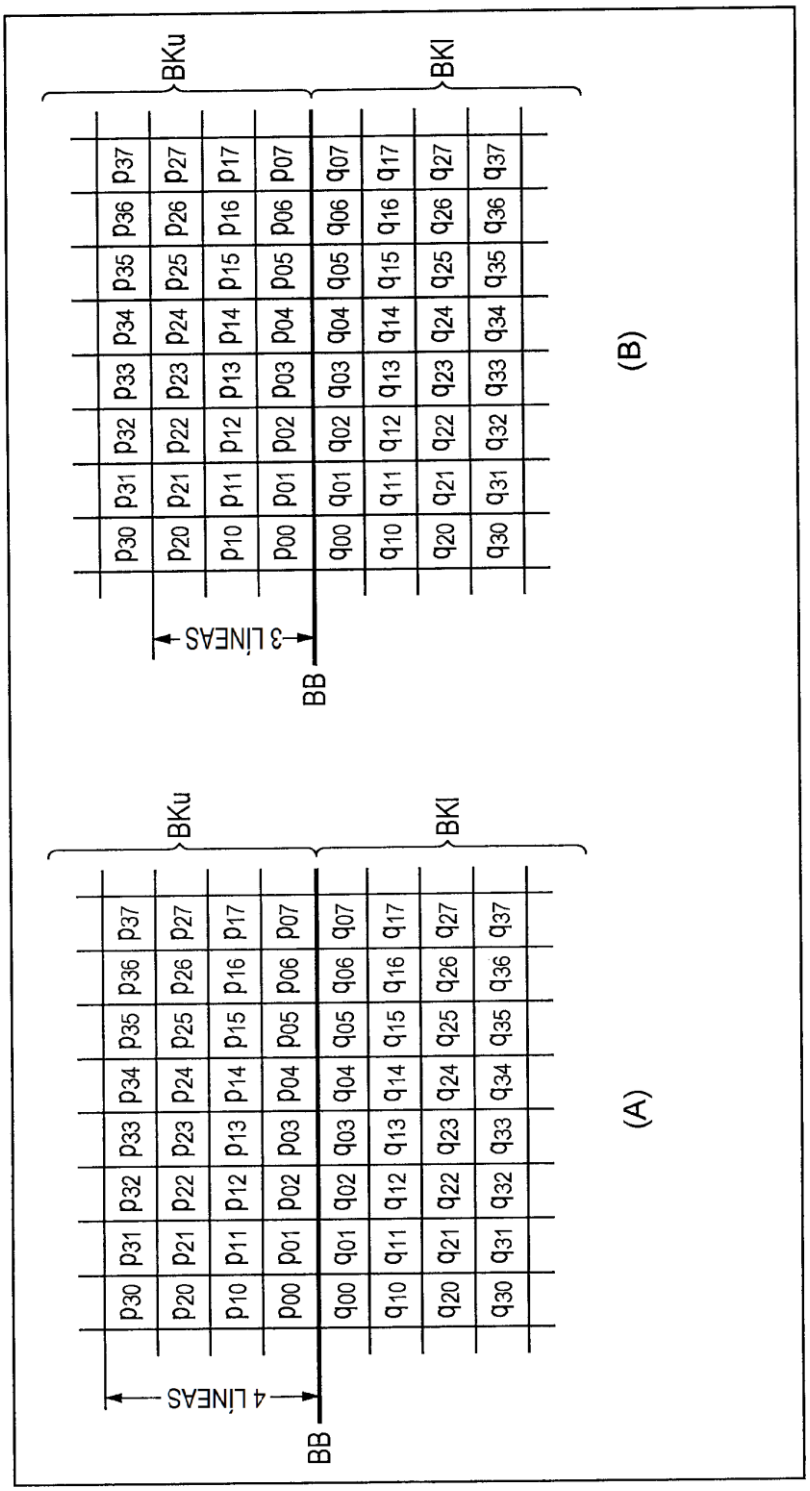


FIG. 14

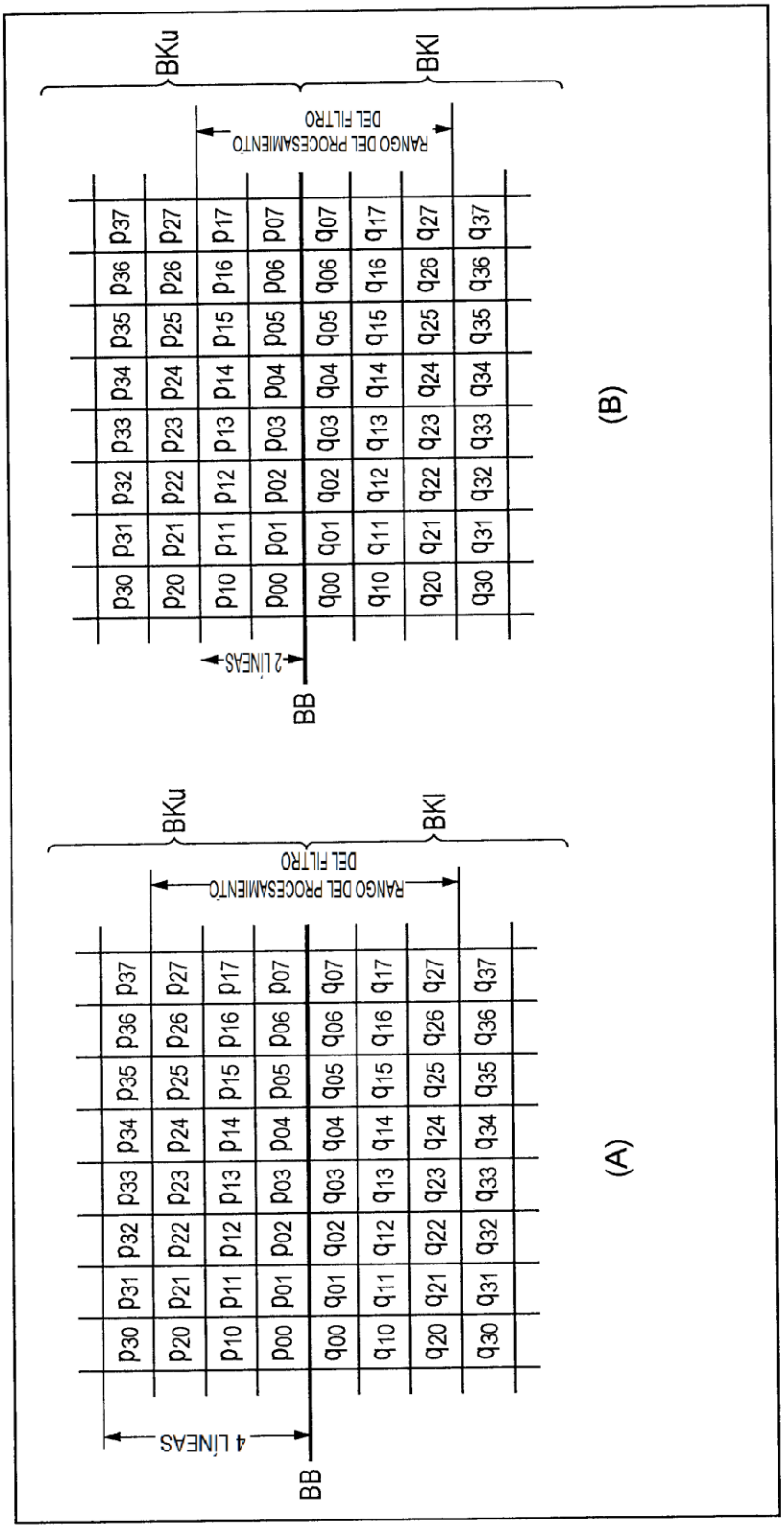

BKu
BKl
RANGO DEL PROCESAMIENTO DEL FILTRO
4 LÍNEAS
BB
(A)
BKu
BKl
RANGO DEL PROCESAMIENTO DEL FILTRO
2 LÍNEAS
BB
(B)

FIG. 15

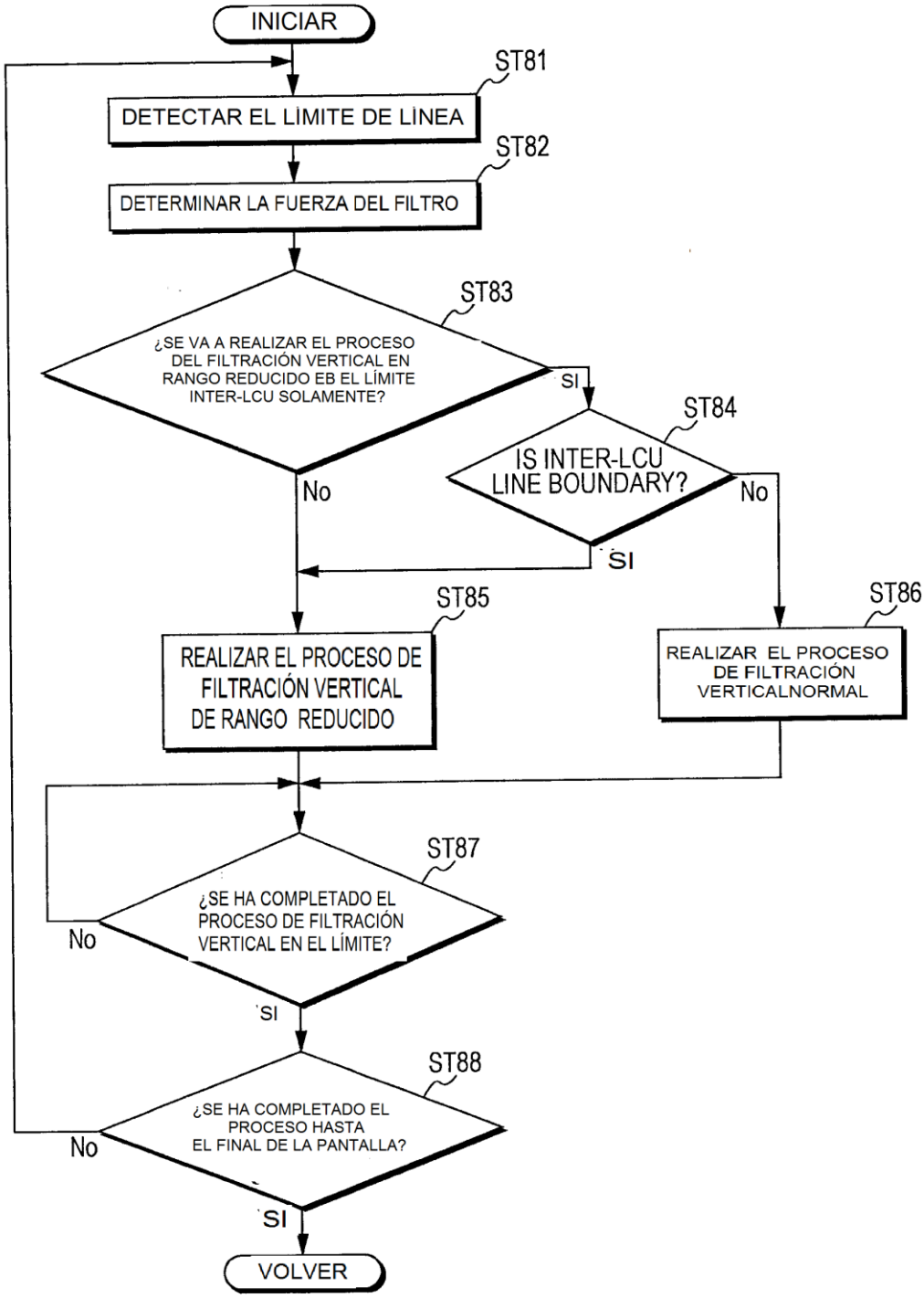

INICIAR
ST81
DETECTAR EL LÍMITE DE LÍNEA
ST82
DETERMINAR LA FUERZA DEL FILTRO
ST83
¿SE VA A REALIZAR EL PROCESO DEL FILTRACIÓN VERTICAL EN RANGO REDUCIDO EB EL LÍMITE INTER-LCU SOLAMENTE?
SI
No
ST84
IS INTER-LCU LINE BOUNDARY?
No
SI
ST85
REALIZAR EL PROCESO DE FILTRACIÓN VERTICAL DE RANGO REDUCIDO
ST86
REALIZAR EL PROCESO DE FILTRACIÓN VERTICALNORMAL
ST87
¿SE HA COMPLETADO EL PROCESO DE FILTRACIÓN VERTICAL EN EL LÍMITE?
No
SI
ST88
¿SE HA COMPLETADO EL PROCESO HASTA EL FINAL DE LA PANTALLA?
No
SI
VOLVER

FIG. 16

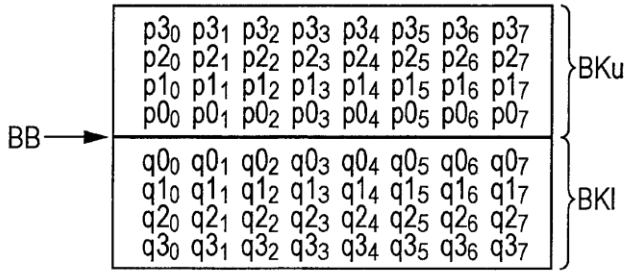

p3₀ p3₁ p3₂ p3₃ p3₄ p3₅ p3₆ p3₇
p2₀ p2₁ p2₂ p2₃ p2₄ p2₅ p2₆ p2₇
p1₀ p1₁ p1₂ p1₃ p1₄ p1₅ p1₆ p1₇
p0₀ p0₁ p0₂ p0₃ p0₄ p0₅ p0₆ p0₇
BKu
BB
q0₀ q0₁ q0₂ q0₃ q0₄ q0₅ q0₆ q0₇
q1₀ q1₁ q1₂ q1₃ q1₄ q1₅ q1₆ q1₇
q2₀ q2₁ q2₂ q2₃ q2₄ q2₅ q2₆ q2₇
q3₀ q3₁ q3₂ q3₃ q3₄ q3₅ q3₆ q3₇
BKl

FIG. 17

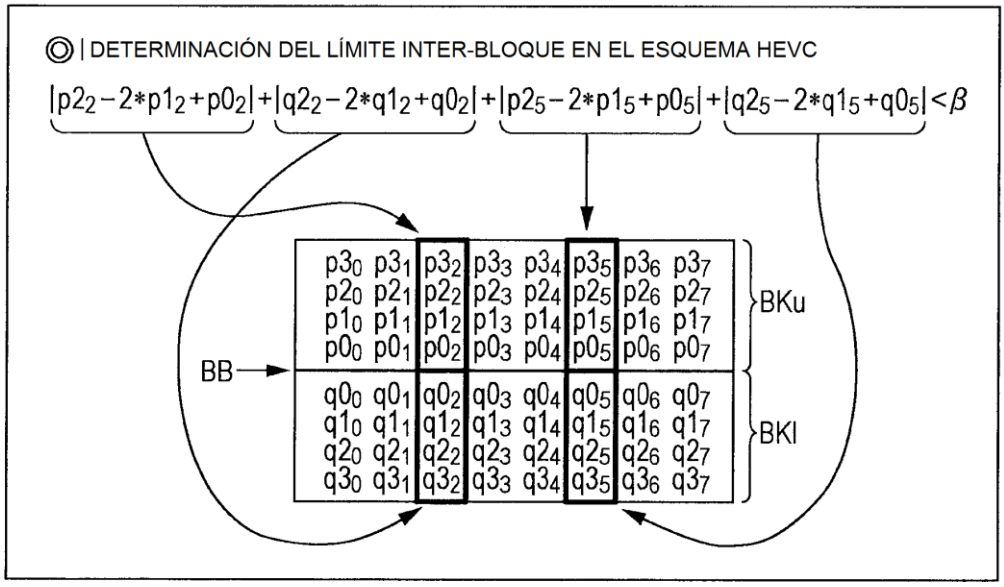

◎ | DETERMINACIÓN DEL LÍMITE INTER-BLOQUE EN EL ESQUEMA HEVC
|p2₂ − 2*p1₂ + p0₂| + |q2₂ − 2*q1₂ + q0₂| + |p2₅ − 2*p1₅ + p0₅| + |q2₅ − 2*q1₅ + q0₅| < β
p3₀ p3₁ p3₂ p3₃ p3₄ p3₅ p3₆ p3₇
p2₀ p2₁ p2₂ p2₃ p2₄ p2₅ p2₆ p2₇
p1₀ p1₁ p1₂ p1₃ p1₄ p1₅ p1₆ p1₇
p0₀ p0₁ p0₂ p0₃ p0₄ p0₅ p0₆ p0₇
BKu
BB
q0₀ q0₁ q0₂ q0₃ q0₄ q0₅ q0₆ q0₇
q1₀ q1₁ q1₂ q1₃ q1₄ q1₅ q1₆ q1₇
q2₀ q2₁ q2₂ q2₃ q2₄ q2₅ q2₆ q2₇
q3₀ q3₁ q3₂ q3₃ q3₄ q3₅ q3₆ q3₇
BKl

FIG. 18

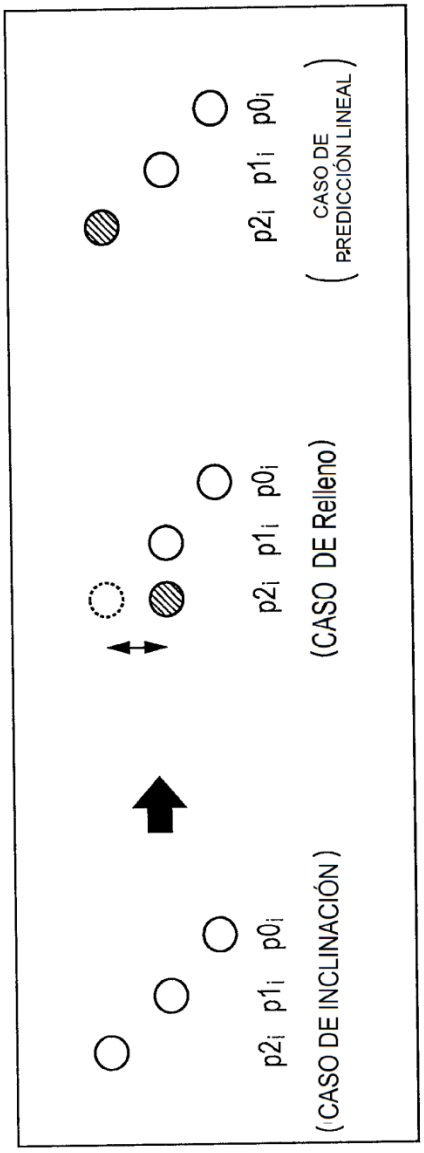

p0i
p1i
p2i
(CASO DE PREDICCIÓN LINEAL)
p0i
p1i
p2i
(CASO DE Relleno)
p0i
p1i
p2i
(CASO DE INCLINACIÓN)

FIG. 19

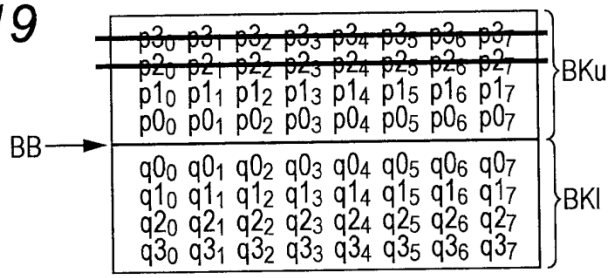
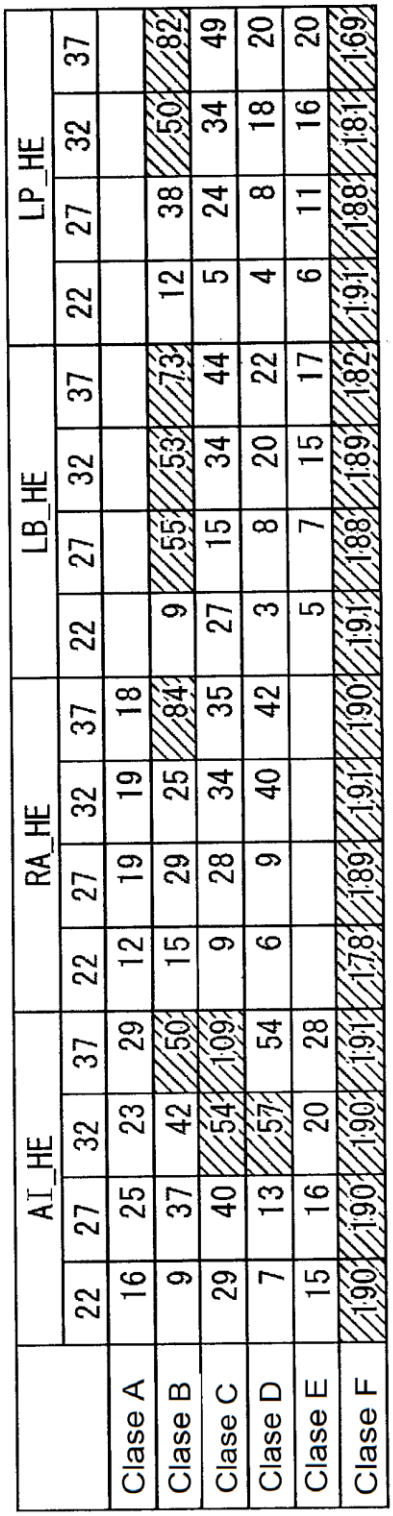

p3₀ p3₁ p3₂ p3₃ p3₄ p3₅ p3₆ p3₇
p2₀ p2₁ p2₂ p2₃ p2₄ p2₅ p2₆ p2₇
p1₀ p1₁ p1₂ p1₃ p1₄ p1₅ p1₆ p1₇
p0₀ p0₁ p0₂ p0₃ p0₄ p0₅ p0₆ p0₇
BKu
BB
q0₀ q0₁ q0₂ q0₃ q0₄ q0₅ q0₆ q0₇
q1₀ q1₁ q1₂ q1₃ q1₄ q1₅ q1₆ q1₇
q2₀ q2₁ q2₂ q2₃ q2₄ q2₅ q2₆ q2₇
q3₀ q3₁ q3₂ q3₃ q3₄ q3₅ q3₆ q3₇
BKl

FIG. 20



	AI_HE				RA_HE				LB_HE				LP_HE			
	22	27	32	37	22	27	32	37	22	27	32	37	22	27	32	37
Clase A	16	25	23	29	12	19	19	18								
Clase B	9	37	42	50	15	29	25	84	9	55	53	73	12	38	50	82
Clase C	29	40	54	109	9	28	34	35	27	15	34	44	5	24	34	49
Clase D	7	13	57	54	6	9	40	42	3	8	20	22	4	8	18	20
Clase E	15	16	20	28					5	7	15	17	6	11	16	20
Clase F	190	190	190	191	178	189	191	190	191	188	189	182	191	188	181	169

FIG. 21

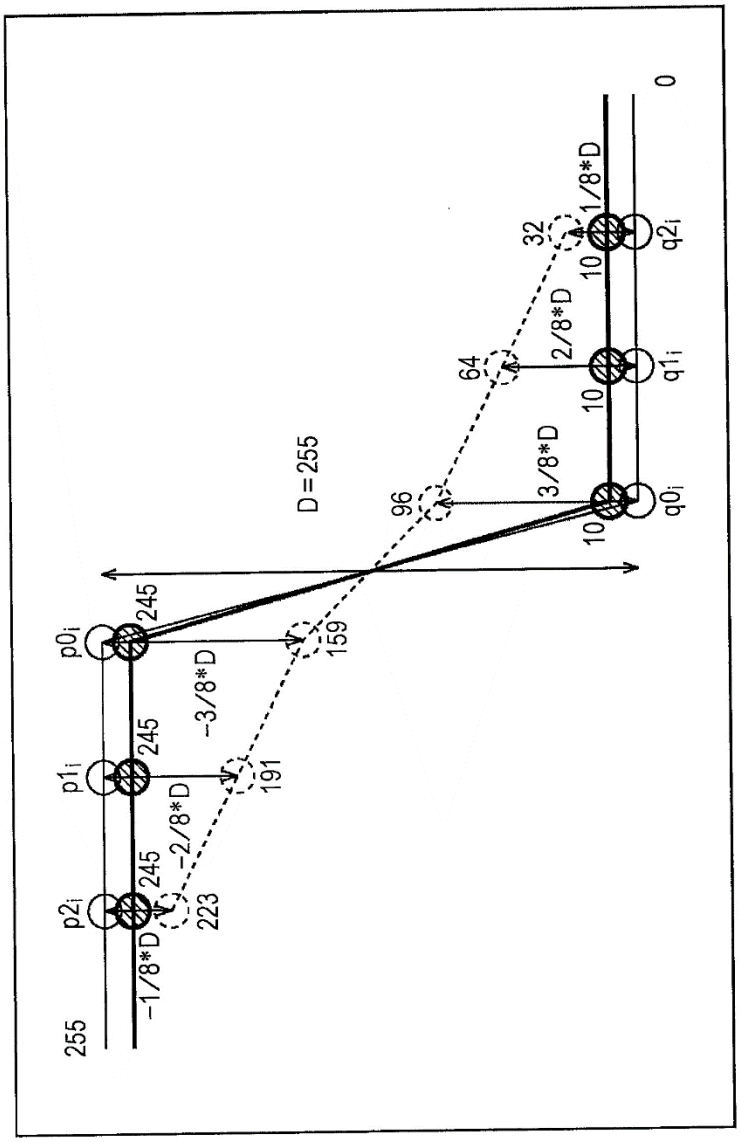

255
p2i
p1i
p0i
245
245
245
-1/8*D
-2/8*D
-3/8*D
223
191
159
D=255
96
64
32
3/8*D
2/8*D
1/8*D
10
10
10
q0i
q1i
q2i
0

FIG. 22

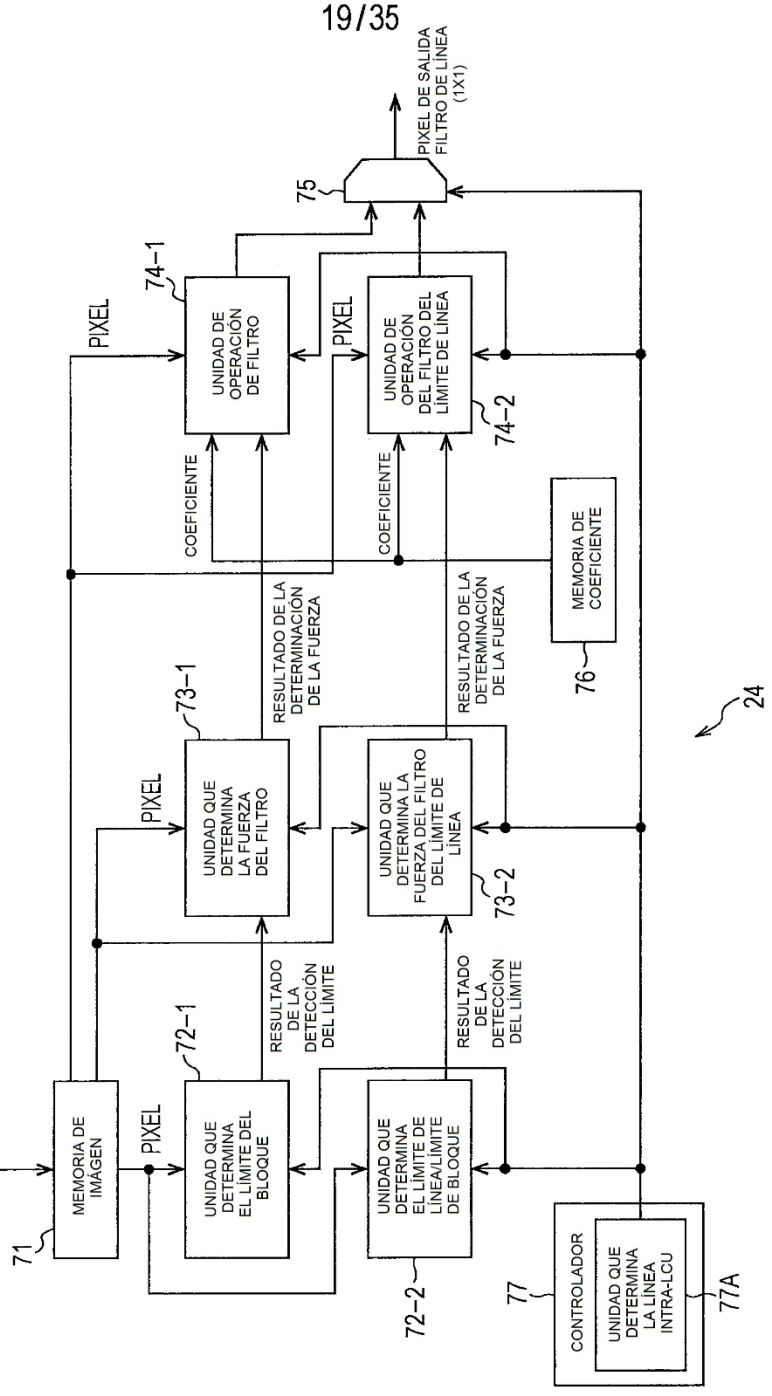

71
MEMORIA DE IMÁGEN
PIXEL
72-1
UNIDAD QUE DETERMINA EL LÍMITE DEL BLOQUE
RESULTADO DE LA DETECCIÓN DEL LÍMITE
72-2
UNIDAD QUE DETERMINA EL LÍMITE DE LÍNEA/LÍMITE DE BLOQUE
RESULTADO DE LA DETECCIÓN DEL LÍMITE
73-1
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
73-2
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO DEL LÍMITE DE LÍNEA
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
74-1
UNIDAD DE OPERACIÓN DE FILTRO
COEFICIENTE
74-2
UNIDAD DE OPERACIÓN DEL FILTRO DEL LÍMITE DE LÍNEA
COEFICIENTE
76
MEMORIA DE COEFICIENTE
75
PIXEL DE SALIDA FILTRO DE LÍNEA (1X1)
77
CONTROLADOR
UNIDAD QUE DETERMINA LA LÍNEA INTRA-LCU
77A
24

FIG. 23

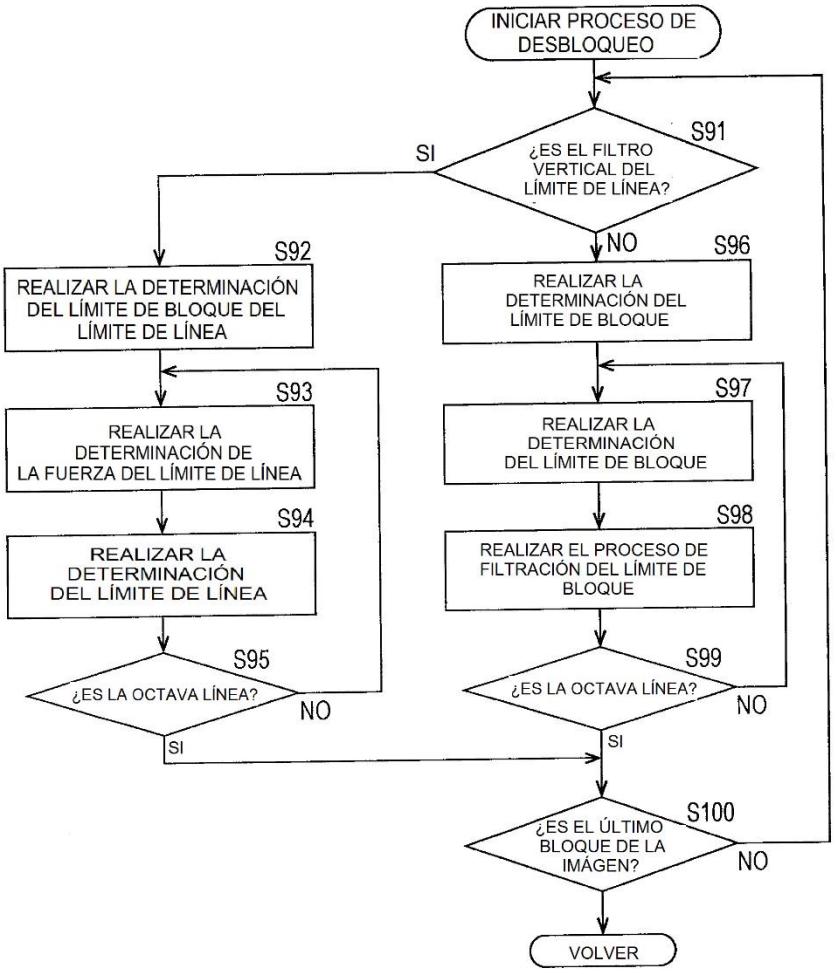

INICIAR PROCESO DE DESBLOQUEO
S91
¿ES EL FILTRO VERTICAL DEL LÍMITE DE LÍNEA?
SI
NO
S92
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE BLOQUE DEL LÍMITE DE LÍNEA
S93
REALIZAR LA DETERMINACIÓN DE LA FUERZA DEL LÍMITE DE LÍNEA
S94
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE LÍNEA
S95
¿ES LA OCTAVA LÍNEA?
NO
SI
S96
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE BLOQUE
S97
REALIZAR LA DETERMINACIÓN DEL LÍMITE DE BLOQUE
S98
REALIZAR EL PROCESO DE FILTRACIÓN DEL LÍMITE DE BLOQUE
S99
¿ES LA OCTAVA LÍNEA?
NO
SI
S100
¿ES EL ÚLTIMO BLOQUE DE LA IMÁGEN?
NO
VOLVER

FIG. 24

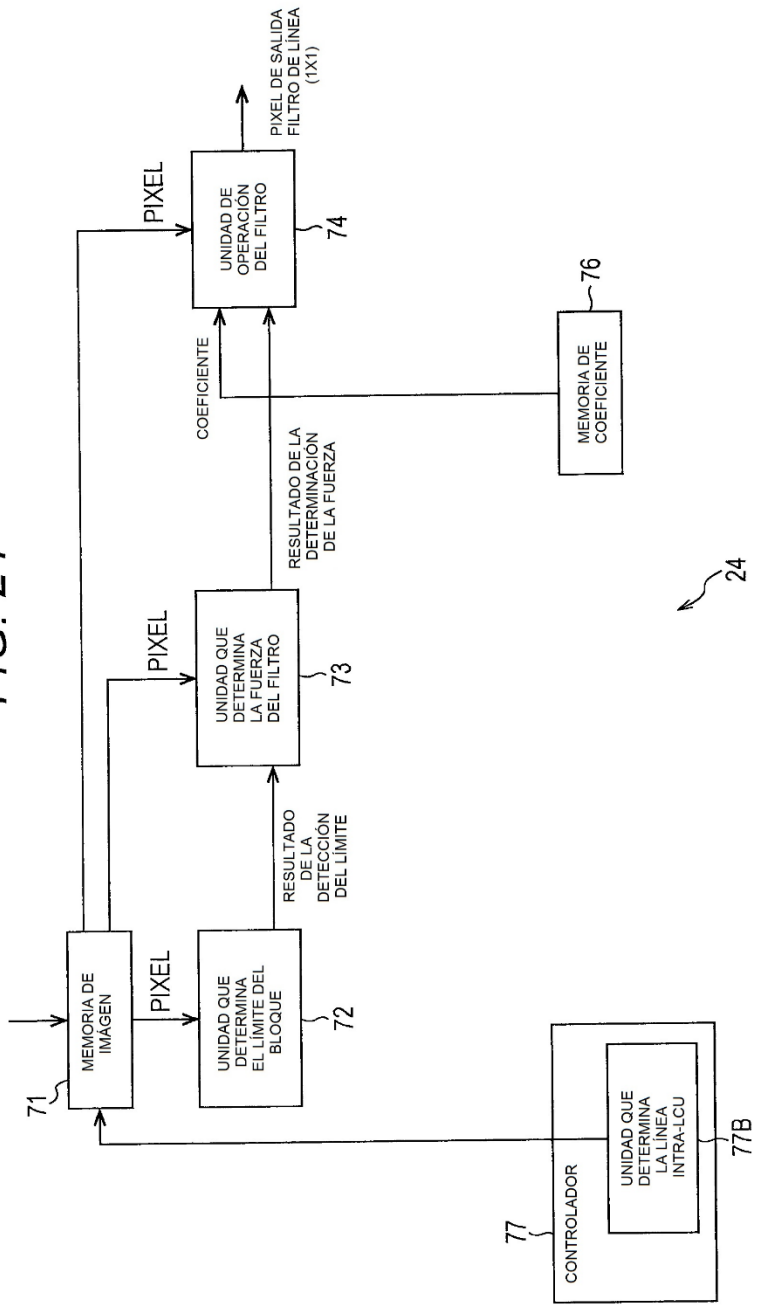

MEMORIA DE IMAGEN
71
PIXEL
UNIDAD QUE DETERMINA EL LÍMITE DEL BLOQUE
72
RESULTADO DE LA DETECCIÓN DEL LÍMITE
PIXEL
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO
73
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
PIXEL
UNIDAD DE OPERACIÓN DEL FILTRO
74
PIXEL DE SALIDA FILTRO DE LÍNEA (1X1)
COEFICIENTE
MEMORIA DE COEFICIENTE
76
24
CONTROLADOR
77
UNIDAD QUE DETERMINA LA LÍNEA INTRA-LCU
77B

FIG. 25

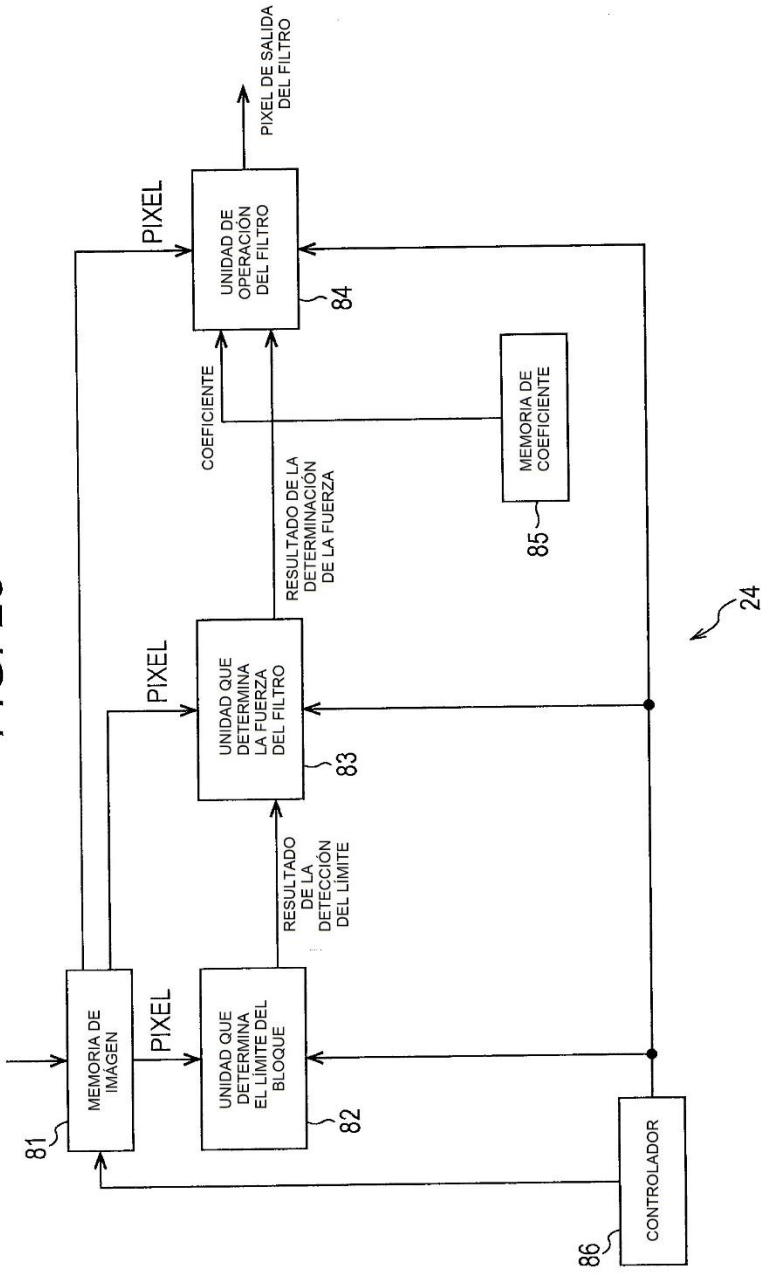

PIXEL DE SALIDA DEL FILTRO
PIXEL
UNIDAD DE OPERACIÓN DEL FILTRO
84
COEFICIENTE
MEMORIA DE COEFICIENTE
85
RESULTADO DE LA DETERMINACIÓN DE LA FUERZA
PIXEL
UNIDAD QUE DETERMINA LA FUERZA DEL FILTRO
83
RESULTADO DE LA DETECCIÓN DEL LÍMITE
24
MEMORIA DE IMAGEN
PIXEL
UNIDAD QUE DETERMINA EL LÍMITE DEL BLOQUE
82
81
CONTROLADOR
86

FIG. 26

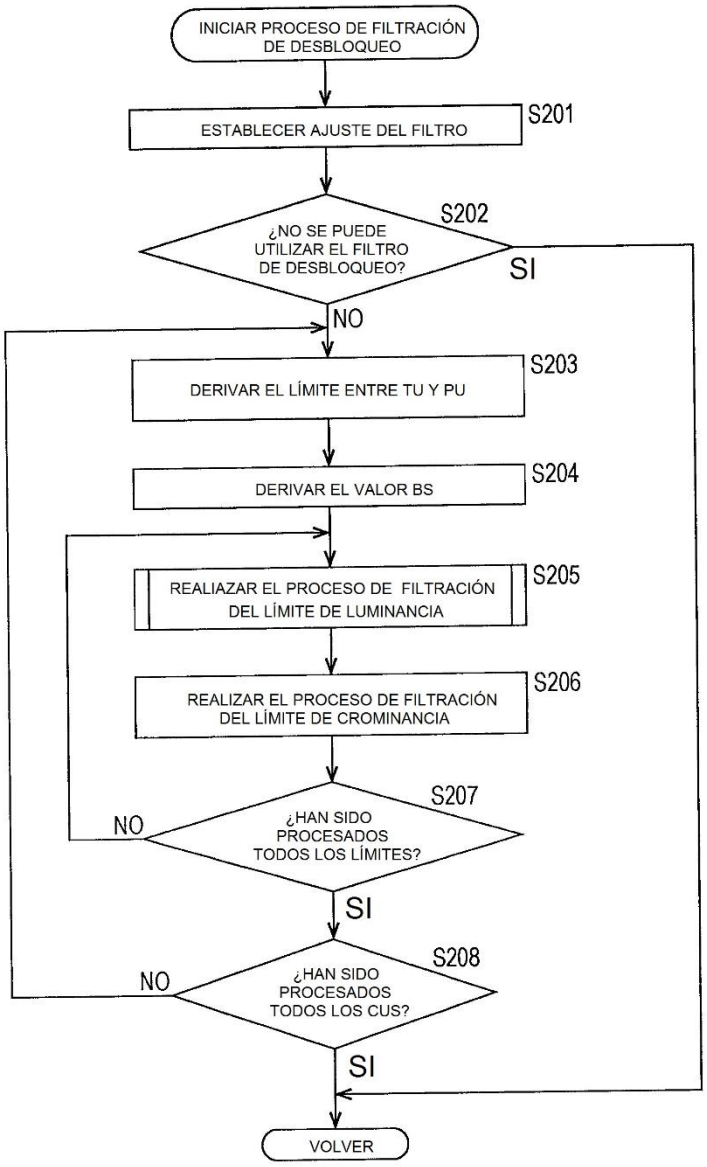

INICIAR PROCESO DE FILTRACIÓN DE DESBLOQUEO
ESTABLECER AJUSTE DEL FILTRO
S201
¿NO SE PUEDE UTILIZAR EL FILTRO DE DESBLOQUEO?
S202
SI
NO
DERIVAR EL LÍMITE ENTRE TU Y PU
S203
DERIVAR EL VALOR BS
S204
REALIAZAR EL PROCESO DE FILTRACIÓN DEL LÍMITE DE LUMINANCIA
S205
REALIZAR EL PROCESO DE FILTRACIÓN DEL LÍMITE DE CROMINANCIA
S206
¿HAN SIDO PROCESADOS TODOS LOS LÍMITES?
S207
NO
SI
¿HAN SIDO PROCESADOS TODOS LOS CUS?
S208
NO
SI
VOLVER

FIG. 27

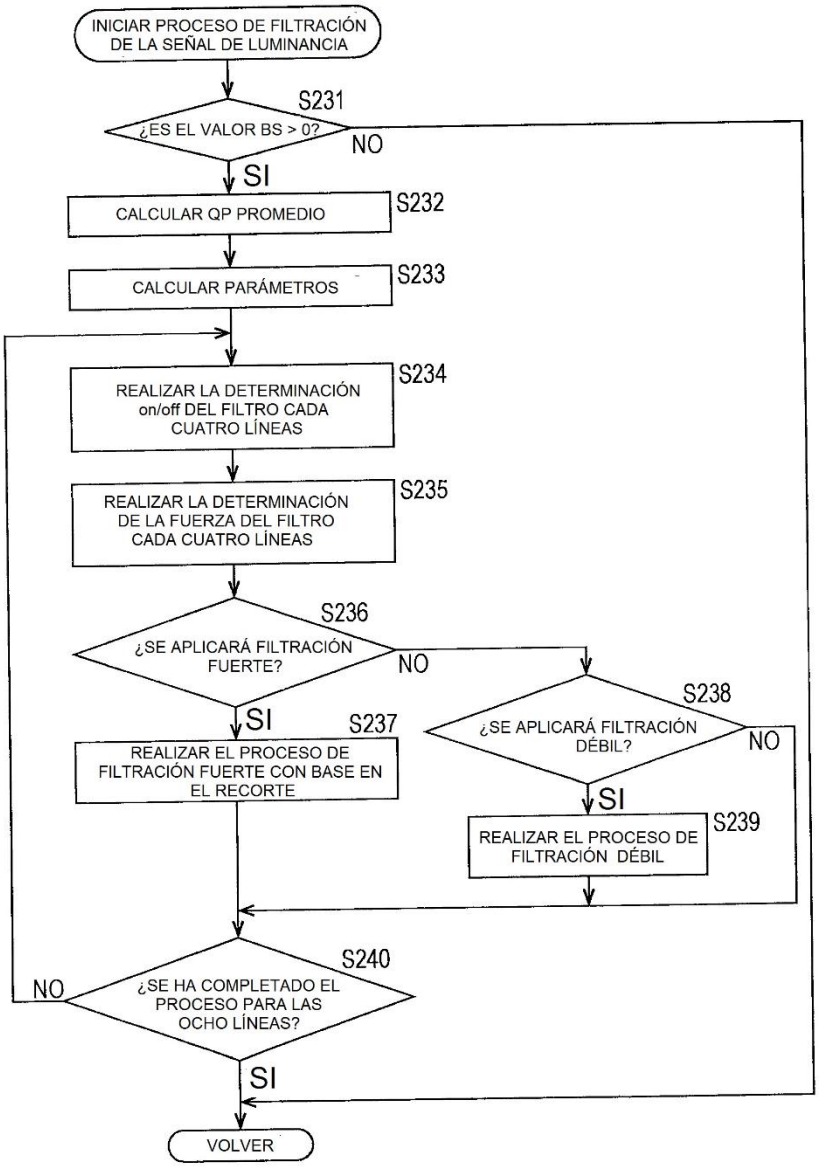

INICIAR PROCESO DE FILTRACIÓN DE LA SEÑAL DE LUMINANCIA
S231
¿ES EL VALOR BS > 0?
NO
SI
CALCULAR QP PROMEDIO
S232
CALCULAR PARÁMETROS
S233
REALIZAR LA DETERMINACIÓN on/off DEL FILTRO CADA CUATRO LÍNEAS
S234
REALIZAR LA DETERMINACIÓN DE LA FUERZA DEL FILTRO CADA CUATRO LÍNEAS
S235
S236
¿SE APLICARÁ FILTRACIÓN FUERTE?
NO
SI
S237
REALIZAR EL PROCESO DE FILTRACIÓN FUERTE CON BASE EN EL RECORTE
S238
¿SE APLICARÁ FILTRACIÓN DÉBIL?
NO
SI
S239
REALIZAR EL PROCESO DE FILTRACIÓN DÉBIL
S240
NO
¿SE HA COMPLETADO EL PROCESO PARA LAS OCHO LÍNEAS?
SI
VOLVER

FIG. 28

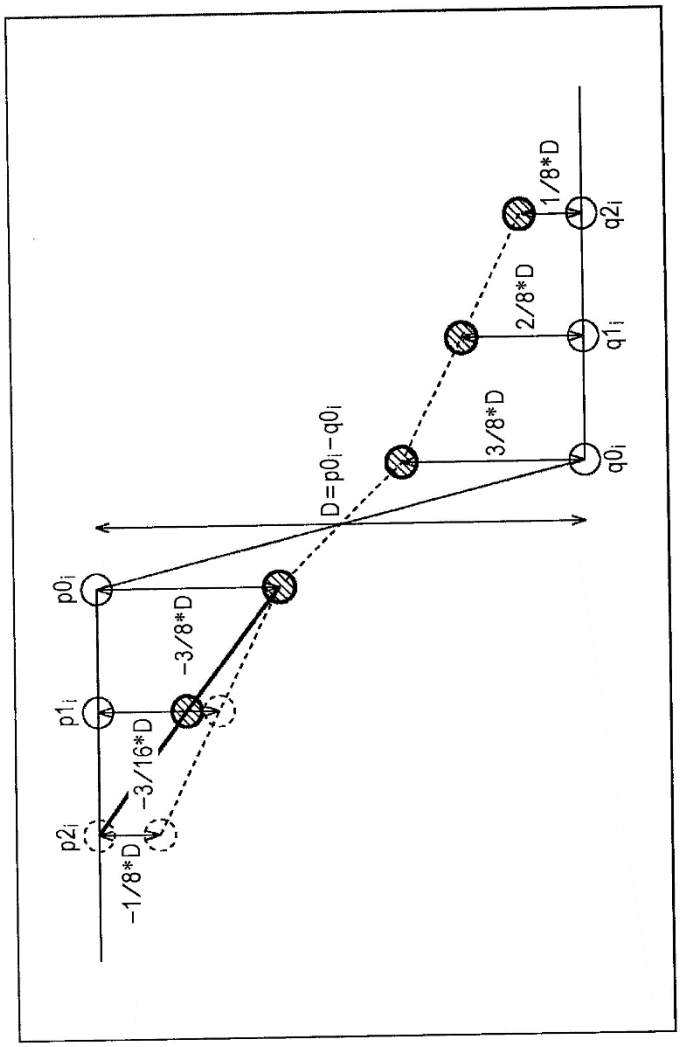


FIG. 29

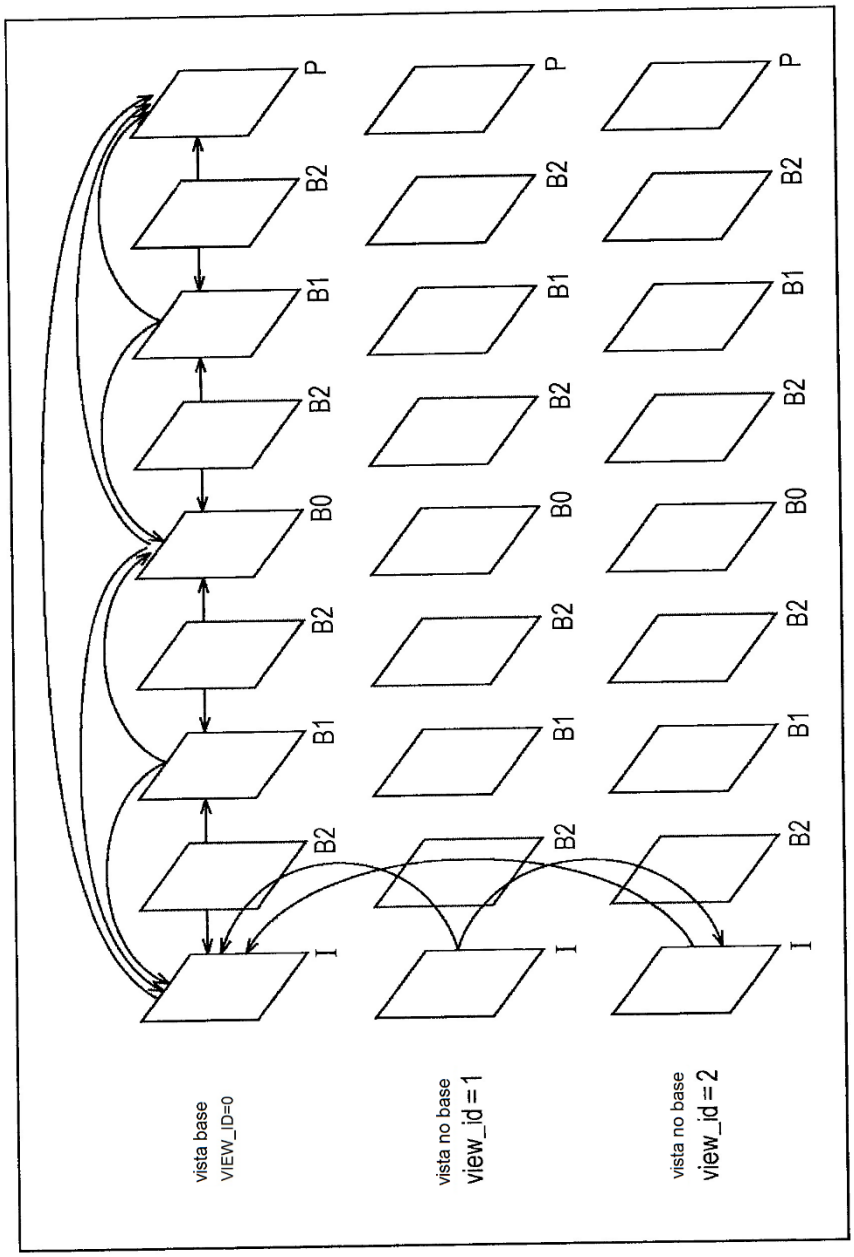

vista base
VIEW_ID=0
vista no base
view_id = 1
vista no base
view_id = 2
I
B2
B1
B2
B0
B2
B1
B2
P

FIG. 30

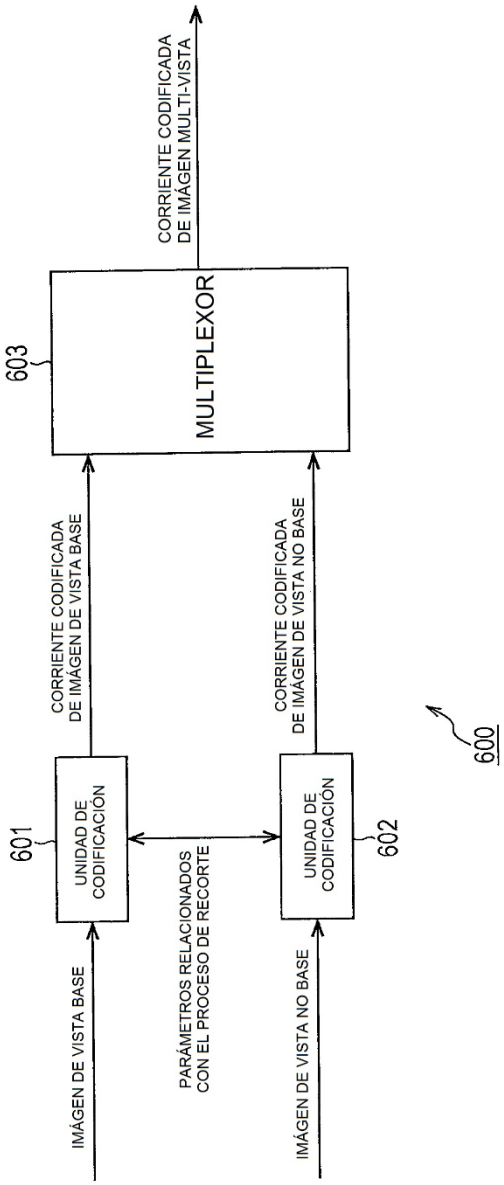

600
601
UNIDAD DE CODIFICACIÓN
602
UNIDAD DE CODIFICACIÓN
603
MULTIPLEXOR
IMÁGEN DE VISTA BASE
IMÁGEN DE VISTA NO BASE
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
CORRIENTE CODIFICADA DE IMÁGEN DE VISTA BASE
CORRIENTE CODIFICADA DE IMÁGEN DE VISTA NO BASE
CORRIENTE CODIFICADA DE IMÁGEN MULTI-VISTA

FIG. 31

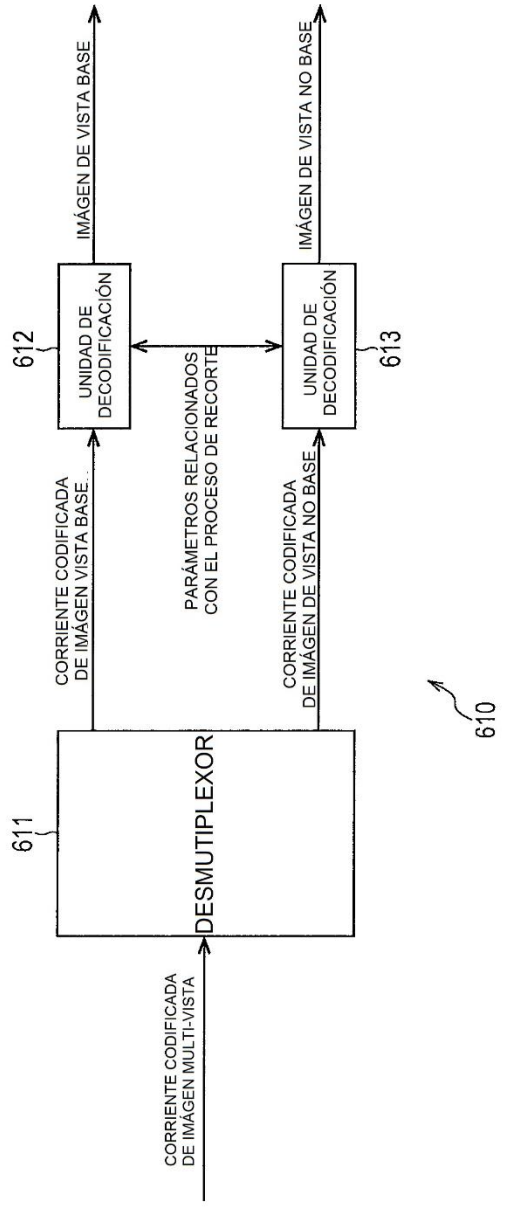

CORRIENTE CODIFICADA DE IMÁGEN MULTI-VISTA
611
DESMUTIPLEXOR
CORRIENTE CODIFICADA DE IMÁGEN VISTA BASE
CORRIENTE CODIFICADA DE IMÁGEN DE VISTA NO BASE
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
612
UNIDAD DE DECODIFICACIÓN
613
UNIDAD DE DECODIFICACIÓN
IMÁGEN DE VISTA BASE
IMÁGEN DE VISTA NO BASE
610

FIG. 32

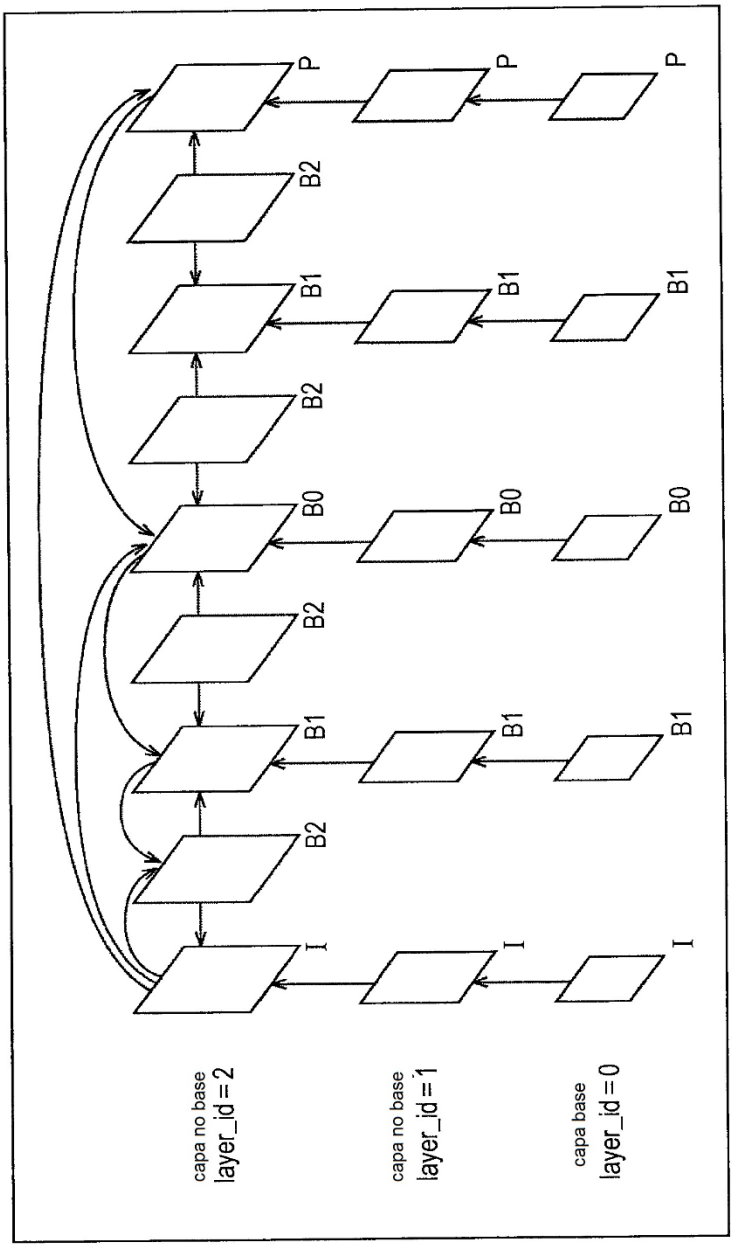

capa no base
layer_id = 2
capa no base
layer_id = 1
capa base
layer_id = 0
I
B2
B1
B2
B0
B2
B1
B2
P

FIG. 33

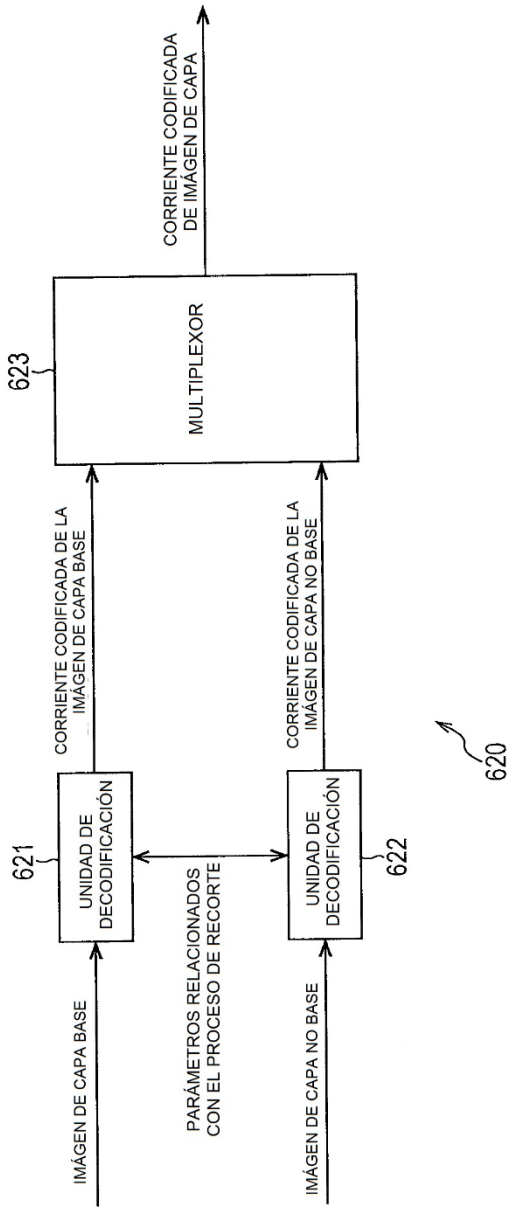

620
623
MULTIPLEXOR
CORRIENTE CODIFICADA DE IMÁGEN DE CAPA
CORRIENTE CODIFICADA DE LA IMÁGEN DE CAPA BASE
CORRIENTE CODIFICADA DE LA IMÁGEN DE CAPA NO BASE
621
UNIDAD DE DECODIFICACIÓN
622
UNIDAD DE DECODIFICACIÓN
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
IMÁGEN DE CAPA BASE
IMÁGEN DE CAPA NO BASE

FIG. 34

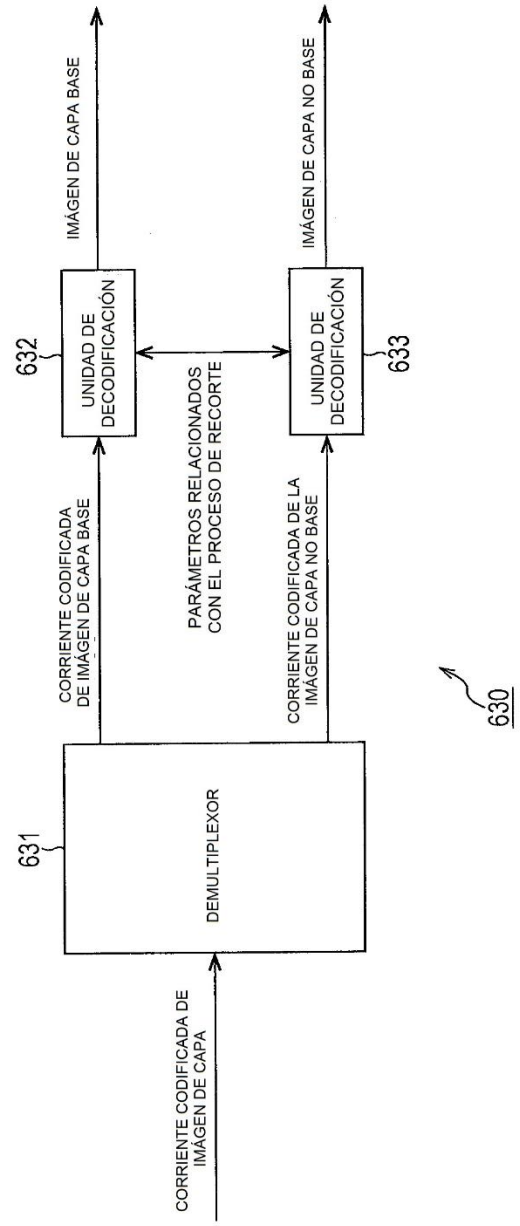

CORRIENTE CODIFICADA DE IMÁGEN DE CAPA
631
DEMULTIPLEXOR
CORRIENTE CODIFICADA DE IMÁGEN DE CAPA BASE
CORRIENTE CODIFICADA DE LA IMÁGEN DE CAPA NO BASE
PARÁMETROS RELACIONADOS CON EL PROCESO DE RECORTE
632
UNIDAD DE DECODIFICACIÓN
633
UNIDAD DE DECODIFICACIÓN
IMÁGEN DE CAPA BASE
IMÁGEN DE CAPA NO BASE
630

FIG. 35

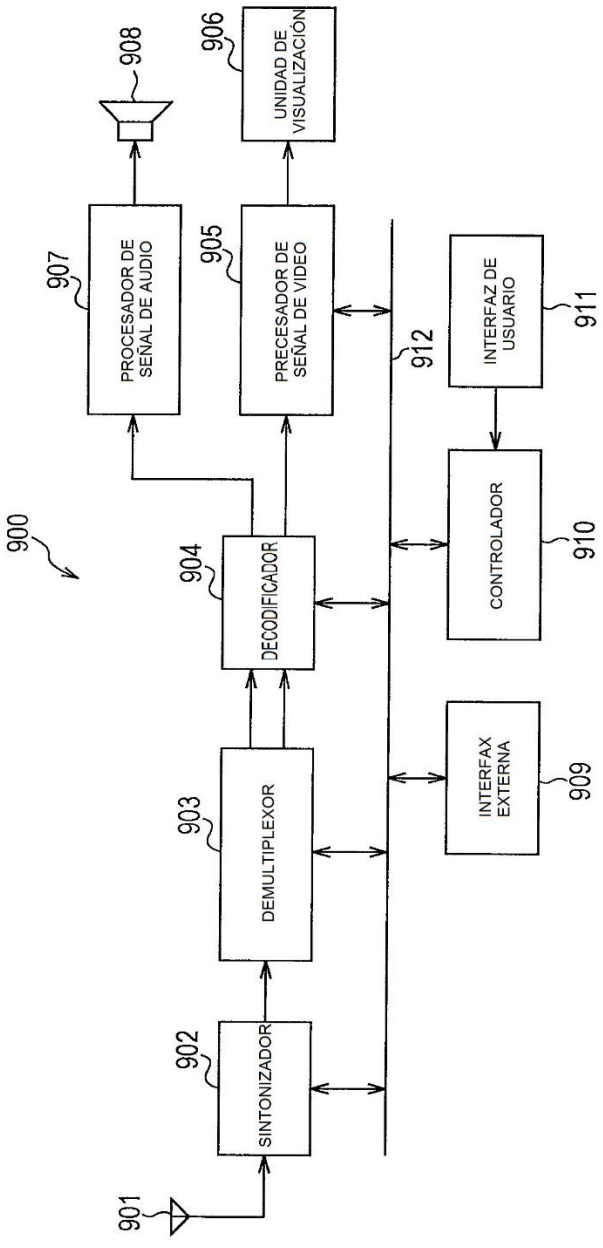

900
901
902
SINTONIZADOR
903
DEMULTIPLEXOR
904
DECODIFICADOR
905
PRECESADOR DE SEÑAL DE VIDEO
906
UNIDAD DE VISUALIZACIÓN
907
PROCESADOR DE SEÑAL DE AUDIO
908
909
INTERFAX EXTERNA
910
CONTROLADOR
911
INTERFAZ DE USUARIO
912

FIG. 36

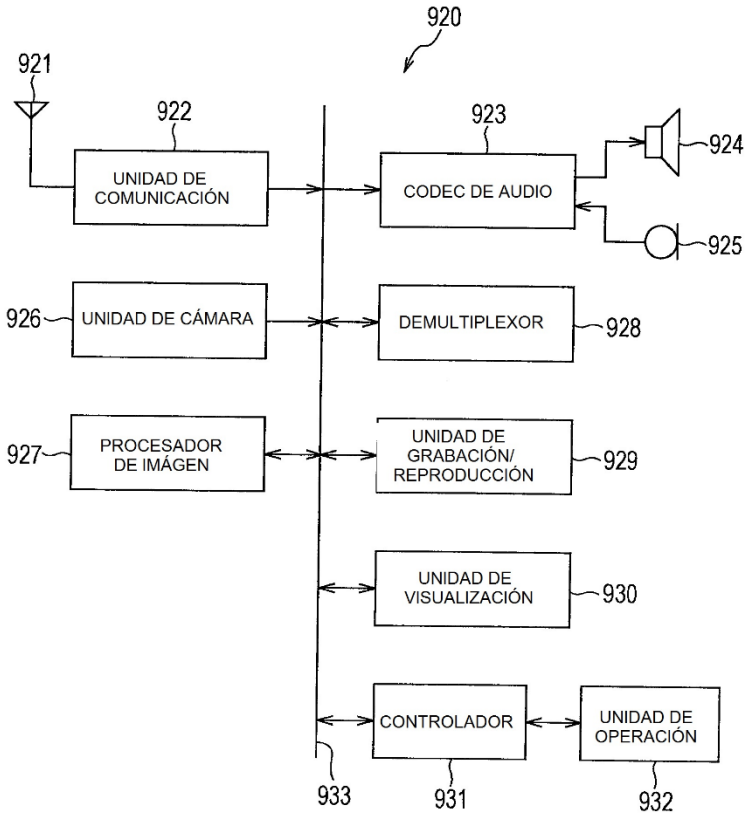

920
921
922
923
924
925
926
927
928
929
930
931
932
933
UNIDAD DE COMUNICACIÓN
CODEC DE AUDIO
UNIDAD DE CÁMARA
DEMULTIPLEXOR
PROCESADOR DE IMÁGEN
UNIDAD DE GRABACIÓN/ REPRODUCCIÓN
UNIDAD DE VISUALIZACIÓN
CONTROLADOR
UNIDAD DE OPERACIÓN

FIG. 37

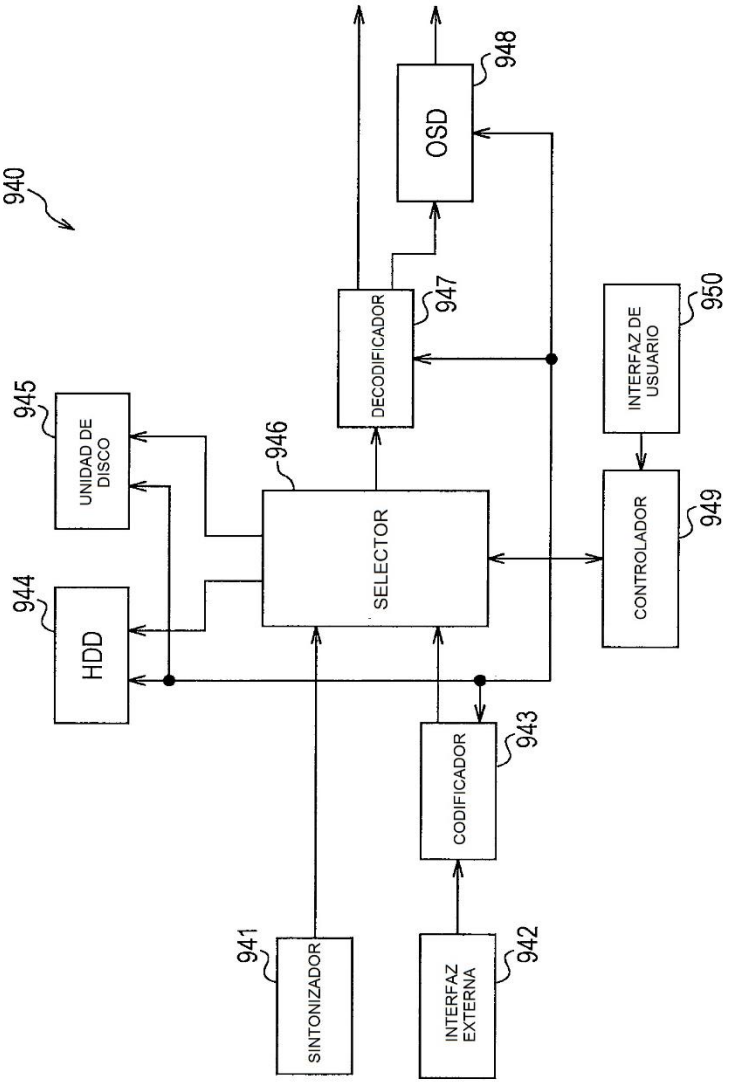
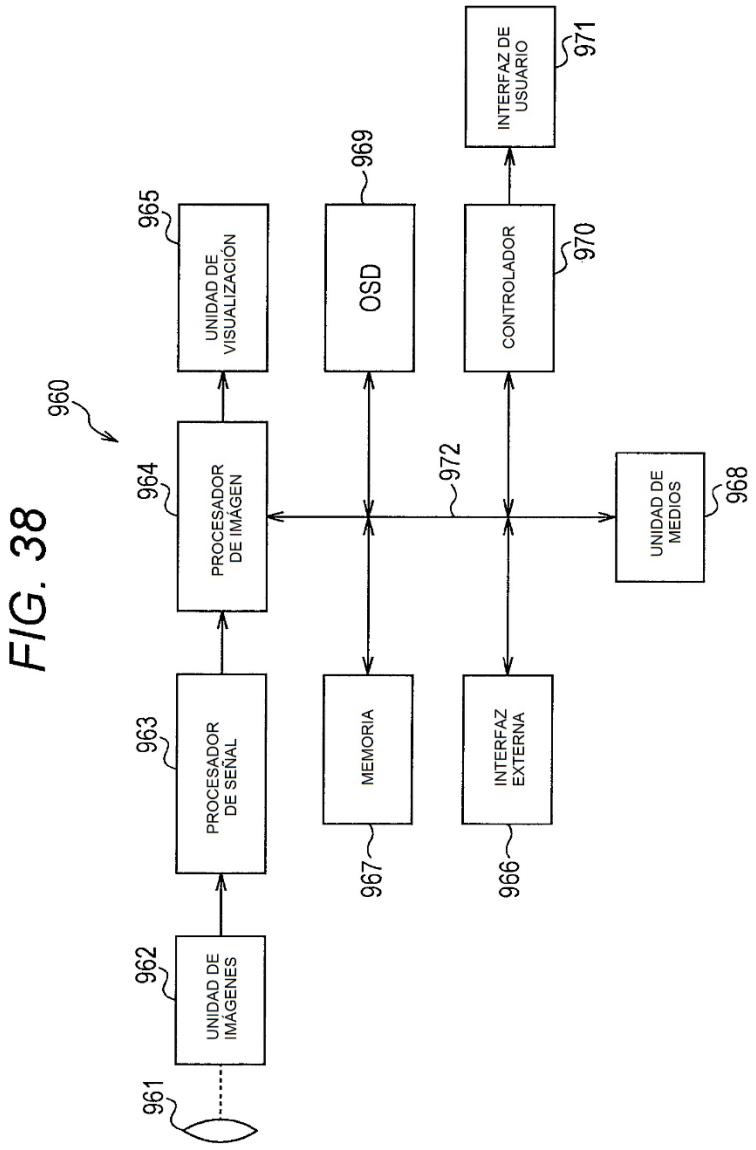

940
941
SINTONIZADOR
942
INTERFAZ EXTERNA
943
CODIFICADOR
944
HDD
945
UNIDAD DE DISCO
946
SELECTOR
947
DECODIFICADOR
948
OSD
949
CONTROLADOR
950
INTERFAZ DE USUARIO

FIG. 38


960
961
962
UNIDAD DE IMÁGENES
963
PROCESADOR DE SEÑAL
964
PROCESADOR DE IMAGEN
965
UNIDAD DE VISUALIZACIÓN
966
INTERFAZ EXTERNA
967
MEMORIA
968
UNIDAD DE MEDIOS
969
OSD
970
CONTROLADOR
971
INTERFAZ DE USUARIO
972