

Access provided by:
Universidad Nacional de Colombia
(UNAL)
Sign Out

Browse

My Settings

Get Help

Advertisement

Back to Results | Next >

Conferences > 2018 10th International Confe...

Advertisement

Thai Lottery Number Reader App for Blind Lottery Ticket Sellers

<< Results | Next >

2

Nawin Somyat ; Songyot Nakariyakul

Author(s)

View All Authors

26
Full
Text Views

Export to
Collabratec

Alerts

Manage
Content Alerts

Add to Citation
Alerts

Related Articles

A Complete Optical Character Recognition Methodology for Historical Documents
2008 The Eighth IAPR International Workshop on Document Analysis Systems
Published: 2008

Automated read-write kit for blind using hidden Markov model and optical character recognition
2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)
Published: 2015

View All View More

See the top organizations patenting in technologies mentioned in this article

ORGANIZATION 4

ORGANIZATION 3

ORGANIZATION 2

ORGANIZATION 1

Click to Expand

Provided by: InnovationQ PLUS
POWERED BY IEEE AND IP.COM
A PATENT SEARCH AND ANALYTICS TOOL

Related Articles

A Complete Optical Character Recognition Methodology for Historical Documents
2008 The Eighth IAPR International Workshop on Document Analysis Systems
Published: 2008

Automated read-write kit for blind using hidden Markov model and optical character recognition
2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)
Published: 2015

View All View More

See the top organizations patenting in technologies mentioned in this article

ORGANIZATION 4

ORGANIZATION 3

ORGANIZATION 2

ORGANIZATION 1

Click to Expand

Provided by: InnovationQ PLUS
POWERED BY IEEE AND IP.COM
A PATENT SEARCH AND ANALYTICS TOOL

See the top organizations patenting in technologies mentioned in this article

ORGANIZATION 4

ORGANIZATION 3

ORGANIZATION 2

ORGANIZATION 1

Click to Expand

Provided by: InnovationQ PLUS
POWERED BY IEEE AND IP.COM
A PATENT SEARCH AND ANALYTICS TOOL

Advertisement

- Abstract
- Document Sections
- I. Introduction

Downl
PDF

Abstract: Many Thai people with visual impairments make a living by selling lottery

II. Lottery Number
Scanner
Techniques

tickets. To assist the blind in selling their tickets, we present a new Android app called Lotto... [View more](#)

III. The Lottotu App

Metadata

IV. Experimental
Results

Abstract:

Many Thai people with visual impairments make a living by selling lottery tickets. To assist the blind in selling their tickets, we present a new Android app called LottoTU for scanning and reading aloud Thai lottery numbers for the blind. Our system includes lottery number segmentation, thresholding using Otsu's method, digit recognition using the Tesseract engine, and text-to-speech synthesis. Since lottery ticket images can be taken under different illumination conditions and at various orientations and distances to the smartphone camera, we implement three different sets of threshold values and morphological structuring element sizes to detect lottery number locations. The segmented lottery number images are thresholded using Otsu's method and converted into text data using the Tesseract OCR engine. The lottery number text data are then read aloud by text-to-speech software. The experimental results show that our LottoTU app successfully scans and reads aloud lottery numbers with high accuracy.

V. Conclusion

Authors

Figures

References

Keywords

Metrics

Related Articles

Published in: 2018 10th International Conference on Knowledge and Smart Technology (KST)

Date of Conference: 31 Jan.-3 Feb. 2018 **INSPEC Accession Number:** 18023322

Date Added to IEEE Xplore: 09 August 2018 **DOI:** 10.1109/KST.2018.8426096

Publisher: IEEE

ISBN Information:

Conference Location: Chiang Mai, Thailand

Advertisement

Contents

SECTION I. Introduction

The official national lottery administered by the Government Lottery Office (GLO) of Thailand is vastly popular in Thailand. Nowadays, more than 70 million lottery tickets are pre-printed for the bi-weekly lottery, and 27% of them are allotted to several disability organizations [1]. These organizations distribute lottery tickets to local sellers with disabilities who make a living by selling them on the street and at local markets. It is estimated that thousands of these disabled retailers are blind or visually impaired. To know the lottery numbers they are selling, some blind lottery ticket sellers ask people with normal vision to help punch Braille numbers on their lottery tickets, while some memorize the last three digits of lottery numbers and where they are placed on the stalls. This is very inconvenient, and there is a need to help blind lottery ticket sellers perform their jobs more effectively.

Starting 1 September 2017, the GLO has printed a new lottery ticket format. Compare to the old format, the new one is single, not "ticket-pairs", and small. Fig. 1 shows an example of a new lottery ticket [2]. A six-digit number is clearly pre-printed on four yellowish text-line watermarks at the upper right of each ticket. The new format also includes a QR (quick response) code that contains the year, the unit, and the number of each lottery ticket.

Prior work [3], [4] on Thai lottery number reader devices for the blind are now discussed. Tangwannawit and Saetang [3] developed a mobile app for Android smartphones that used a camera to capture a lottery ticket image and employed the Tesseract optical character recognition (OCR) technique [5] to evaluate the image.

The detected lottery number was read aloud to the blind using the "TalkBack"

The detected lottery number was read aloud to the blind using the TTS mode. The main shortcoming of this app is that the user must place his/her smartphone and a lottery ticket into specified holders of a designed box before capturing the image. The lottery ticket image must be taken at a specific distance and position for error minimization before being segmented for lottery number recognition using the OCR engine. This may be inconvenient, for blind vendors must carry around the designed boxes to their stalls every day. Recently, a portable lottery number reader device for the blind named “NU Vision” [4] was introduced. The device mainly consists of a camera, a microprocessor board, a switch, and a speaker. To operate the device, a lottery ticket must be pressed against the scanner when the push button is pushed. The experimental results show that the lottery number could be read out with high accuracy. The device costs 2500 Baht (approximately \$80). However, it is too big to fit into a pocket.

This work presents a free Android app called “LottoTU” for scanning and reading out Thai lottery numbers. It is mainly designed for lottery ticket sellers with visual impairments. Using a mobile app for a smartphone to read out lotter numbers is attractive, since a smartphone is a very portable and versatile device which most people have these days. Unlike [3] , our proposed app does not require the users to place their phones and lottery tickets in boxes for scanning. Instead, the blind may hold a lottery ticket in one hand while using the other hand to capture an image with a smartphone camera. Thus, the blind users do not need to carry around boxes with them. Since lottery ticket images can be taken under varying illumination conditions and at different orientations and distances to the camera, we implement three different sets of threshold values and morphological structuring element sizes to detect lottery number locations. The segmented lottery number images are thresholded and converted into text data using the Tesseract OCR engine. The lottery number text data are read aloud by Android Text-to-Speech (TTS) API. The initial results show that LottoTU successfully scans and reads out lottery numbers with high accuracy.



Figure 1.
An example of a Thai government lottery ticket [2] . A six-digit number is pre-printed on four yellowish text watermarks at the upper right of each ticket.

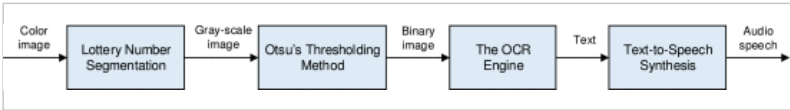


Figure 2.
Block diagram of the lottery number reader system.

The paper is organized as follows. In Section II , we discuss digital image processing techniques used to recognize lottery ticket numbers. Section III describes the proposed LottoTU app and its usage. Experimental results are shown in Section IV , and conclusions are advanced in Section V .

SECTION II.

Lottery Number Scanner Techniques

As noted earlier in Section I, the new lottery ticket format includes a QR code that contains the lottery number information. However, we feel that the QR code is very small and that it may be difficult for the blind to scan it. Therefore, we propose using various digital image processing techniques [6] – [7][8] [9] to detect a six-digit lottery number at the upper right of the ticket. Our problem definition is thus somewhat similar to automatic license plate recognition [10], [11]. Fig. 2 shows the block diagram of our lottery number reader system. It includes lottery number segmentation, thresholding using Otsu's method, digit recognition using the Tesseract OCR engine, and text-to-speech (TTS) synthesis.

A. Lottery Number Segmentation

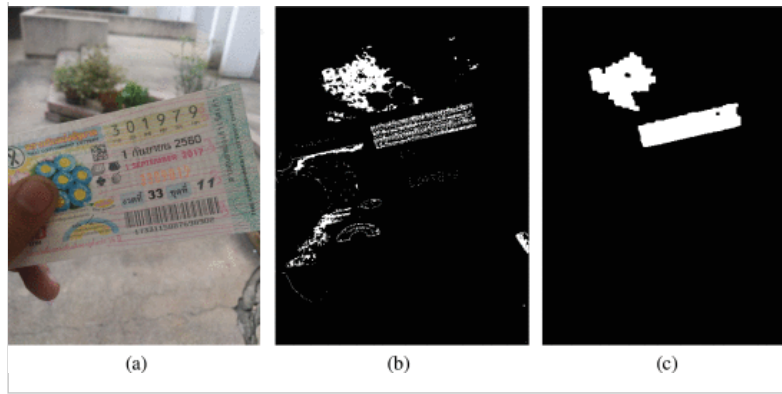
First, the color image captured by a smartphone camera is resized to one having the width of 400 pixels. Depending on the aspect ratio of the image, the resized image is approximately 534×400 pixels or 711×400 pixels. These image sizes are suitable for fast performance. The RGB color space is then converted to the HSV color space, since the HSV color model is less sensitive to color variations [12]. We detect the lottery number location by performing color segmentation to detect four yellow text-line watermarks. For a pixel at location $p = (x, y)$, let $H(x, y)$ be the normalized hue (H) value and $S(x, y)$ be the normalized saturation (S) value of that pixel. A binary mask image, M , is created that identifies pixels whose color is or similar to yellow ($H = 0.15$, $S = 0.27$). That is,

$$M(x, y) = \begin{cases} 1 & \text{If } (H(x, y) - 0.15)^2 + (S(x, y) - 0.27)^2 < T_i \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

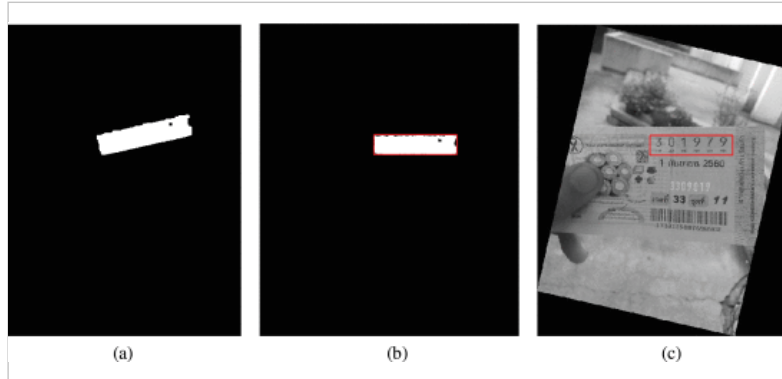
[View Source](#) 

where T_i is a user-defined threshold. The mask image, M , of the lottery ticket usually contains four separate lines indicating the location of the lottery number. Next, the mask image is morphologically processed by performing the closing followed by the opening with the structuring element of size $B_i \times B_i$. Blobs that contains fewer than 1500 pixels are removed to obtain the resultant mask image M_r . Since lottery ticket images can be taken under varying illumination conditions and at many distances, we propose using three different sets of user-defined parameters (T_i, B_i) ; $(T_1, B_1) = (0.008, 5)$ for weak thresholding, respectively. For a given image, strong thresholding produces a mask image containing fewer one-valued pixels than weak thresholding does. Thus, we start creating the mask image from strong to weak thresholding, respectively. If a lottery number is successfully detected using any of the three sets of parameters, the algorithm terminates. If all three sets of parameters do not detect a lottery number, we conclude that it is not an image of a lottery ticket.

Fig. 3 shows an example of how the binary mask of a lottery number is created. Using strong thresholding parameters, Fig. 3(b) is the mask image M of a lottery ticket image shown in Fig. 3(a) obtained using $T_1 = 0.008$. Fig. 3(c) shows the result of morphological operations with $B_1 = 5$ on on Fig. 3(b). As seen, the gaps between four text-line watermarks are mostly closed by the closing, and many false alarms in Fig. 3(b) are removed by the opening. Small blobs with fewer than 1500 pixels are also omitted.

**Figure 3.**

An example of how a binary mask image is created (a) the color lottery ticket image (b) the binary mask image M using a threshold $\tau_1 = 0.008$ (c) The binary mask image M_r obtained after morphological processing using $B_1 = 5$.

**Figure 4.**

Lottery number segmentation of Fig. 3 (a) the candidate blob (b) the blob rotated for horizontal alignment with a red bounding box (c) the rotated gray-level lottery ticket image. The red rectangle indicates the region of the lottery number that is segmented.

Next, we analyze all blobs in the mask image M_r to verify whether it contains yellow watermarks at the upper right of the ticket. As seen in Fig. 1, the ratio of width to height of the yellow watermarks is 4.3:1. We assume that any blob whose ratio of major axis length to minor axis length is between 3.5:1 and 5.0:1 is a candidate of yellow watermarks. To correct the distortion in the image, the orientation of a candidate blob is calculated. With a known orientation of the blob, we rotate both the blob and the gray-scale image of the lottery ticket, so that they are parallel to the x-axis. The bounding box of the blob is used to segment the gray-scale lottery ticket image. Fig. 4 illustrates the above procedures applied to Fig. 3. Fig. 4(a) and 4(b) show the candidate blob of yellow watermarks before and after rotation, respectively. The red rectangle in Fig. 4(b) indicates the bounding box of the blob that is used to segment the rotated gray-scale image in Fig. 4(c). The segmented gray-scale image in Fig. 4(c) is used in the thresholding step using Otsu's method.

B. Thresholding Using Otsu's Method

Fig. 5(a) shows the enlarged version of the region that is segmented in Fig. 4(c). We find that the OCR engine can recognize characters very accurately when the input image is converted into binary and the character height is at least 20 pixels. Thus, we resize the segment region shown in Fig. 5(a) to one having the height of 120 pixels. This ensures that the height of each digit is at least 50 pixels. Otsu's method [6] is used to perform adaptive image thresholding. It selects a threshold value that maximizes the between-class variance of the two classes. Fig. 5(b) shows the thresholding result of Fig. 5(a) obtained using Otsu's method. Pixels whose gray-scale values are less than Otsu's threshold are set to white. Otherwise, they are black. As seen in Fig. 5(b), three-letter words under digits must be removed, so that the OCR software can translate the image into text correctly. They can be easily omitted, for there is a clear vertical gap between each digit and each word. We

omitted, for there is a clear vertical gap between each digit and each word. We calculate the sum of each row in Fig. 5(b) and find the valley (zero value) between two non-zero peaks of the sum vector. The rows below the valley row are set to zero (black). Fig. 5(c) shows the result after word elimination. This binary image is inputted to the Tesseract OCR Engine.

C. Digit Recognition using the Tesseract OCR Engine

The Tesseract engine [5] is one of the most accurate OCR engines and is freely available in many platforms. It can recognize many languages including Thai [13]. To improve the accuracy of the Tesseract software, the input images must be pre-processed. Any rotation or skew must be corrected, and the character height should be at least 20 pixels as noted earlier. In our work, we remove any blob containing fewer than 150 pixels, since it is assumed to be a false object. The binary image with a six-digit lottery number is translated into text data by the Tesseract OCR engine for further processing. Since the blind user may hold the lottery ticket upside down, LottoTU addresses this problem by analyzing both cases. If the lottery number is not successfully recognized by the OCR engine, we rotate the segmented image by 180 degrees and translate it by the Tesseract OCR engine again.

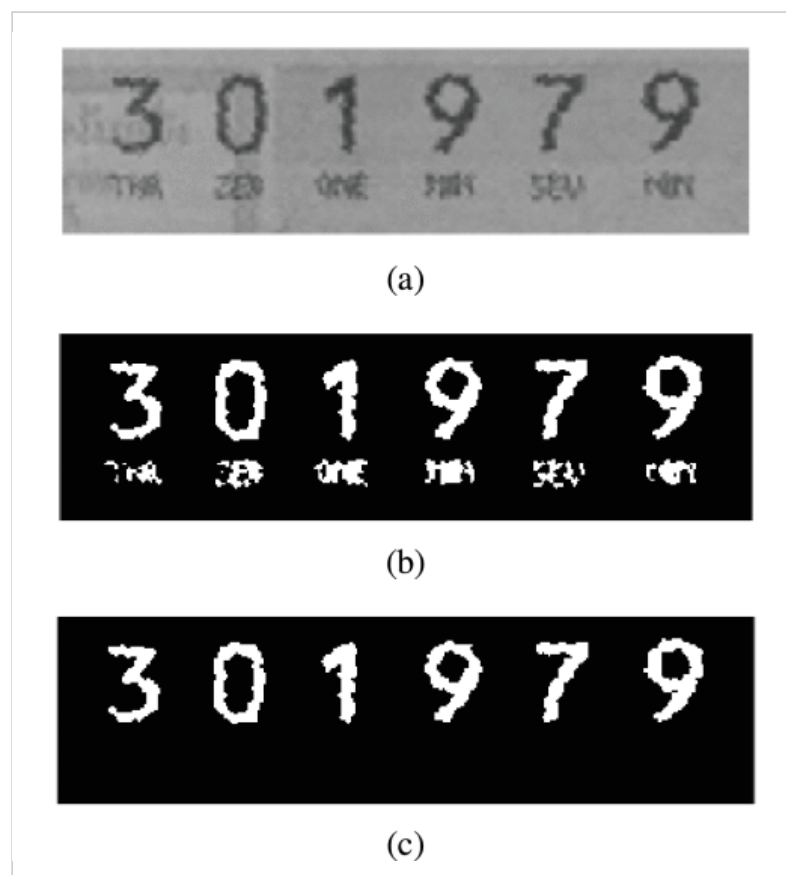


Figure 5.

The thresholding result using Otsu's method (a) gray-scale image (b) binary image result obtained using Otsu's method (c) binary image result after word elimination.

D. Text-to-speech (TTS) Synthesis

The TTS system enables apps to read content aloud. In our LottoTU app, we employ Android Text-to-Speech API [14], which provides Text-to-Speech service for the Android mobile operating system, to read out the lottery number text data. Android Text-to-Speech API is free and supports Thai language.

SECTION III.

The LottoTU App

In this section, we describe in detail the components of the LottoTU app that are designed to read out lottery numbers for the blind.

A. System Requirements

LottoTU runs on the Android smartphones version 4.4 or later. Google TalkBack is activated to help blind lottery ticket sellers interact with their phones. LottoTU requires approximately 75 MB for installation, and 1 GB of RAM is recommended. The primary camera of the smartphone is needed to capture an image of a lottery ticket.

B. Functional Descriptions

The usage of LottoTU is simple. When LottoTU is used, the blind hold a lottery ticket in one hand while using the other hand to capture the image with a smartphone camera. For best results, it is recommended that when the image is captured, the distance to a lottery ticket should be between three and eight inches. Fig. 6 displays how LottoTU is operated. The lottery ticket image may appear rotated and be taken under different illumination conditions. The blind users need to press a button to capture a lottery ticket image. If the lottery number is detected, the app reads aloud the lottery number for the blind. Otherwise, the app asks the users to re-capture the image.

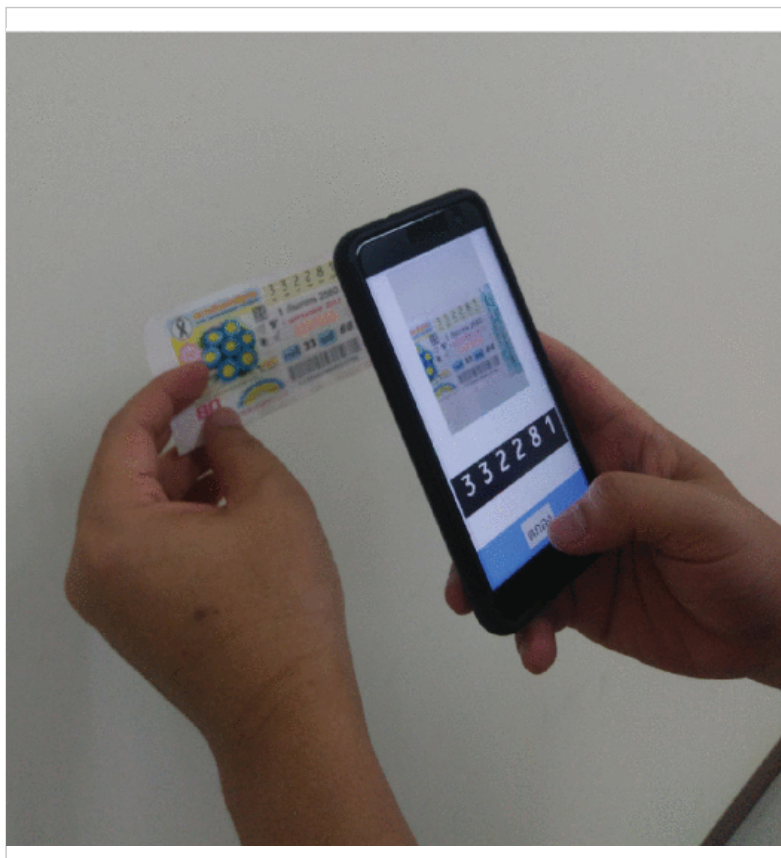


Figure 6.
One of the authors demonstrates the LottoTU app.

SECTION IV. Experimental Results

We test our LottoTU app on a data set of 60 lottery ticket images. The data set

we test our LOTTO10 app on a data set of 60 lottery ticket images. The data set contains images of ten lottery tickets, each taken at six locations (under different illumination conditions, orientations and distances to the smartphone camera). Table I lists these conditions for our experiment settings. Fig. 7 displays four images in the data set. Fig. 7(a) and 7(b) were taken indoors, while Fig. 7(c) and 7(d) were captured outdoors. As seen, the lottery ticket images appear in different sizes, orientations, and illumination conditions.

TABLE I Experiment Settings for Captured Images

No.	Location	Illumination	Orientation	Distance to the camera
1	Indoor (outside an office)	Low	-180°, -60°, -45°, -30°, 0°, 15°, 45°, 60°	From 3 to 8 inches
2	Indoor (inside an office)	High		
3	Indoor (in front of a whiteboard)	Medium		
4	Outdoor (in front of a building)	Medium		
5	Outdoor (under a tree)	Low		
6	Outdoor (on a sidewalk)	High		

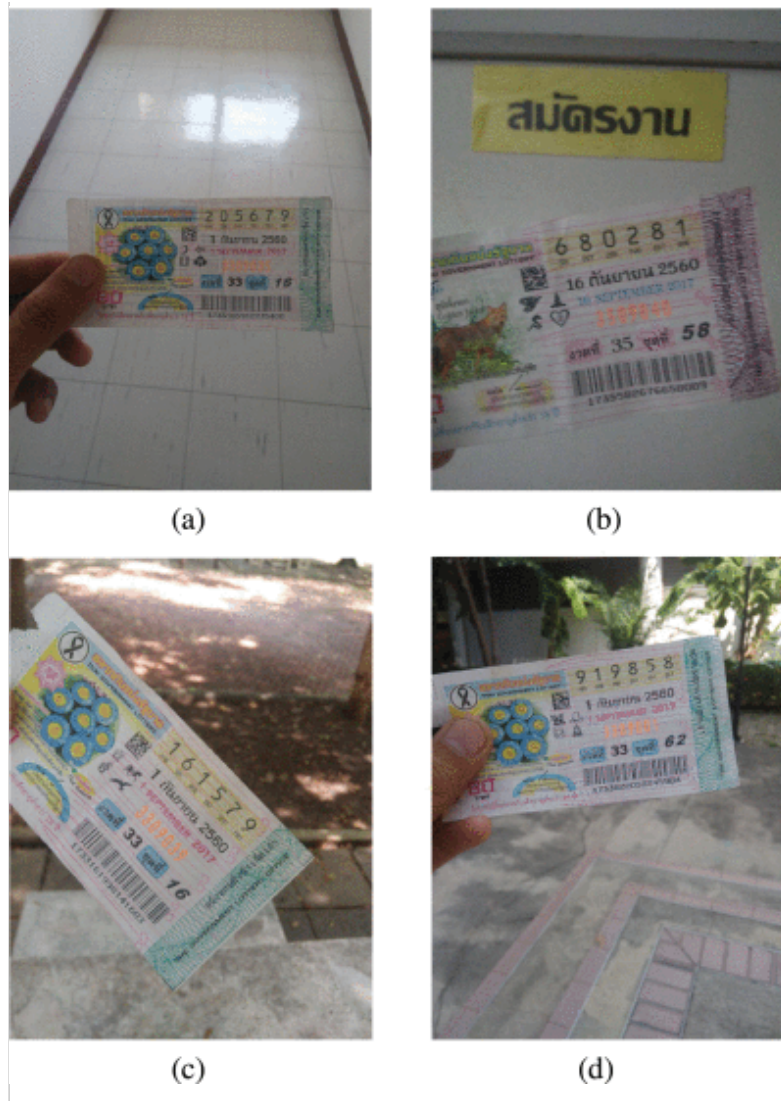


Figure 7.
Four of 60 lottery ticket images in the data set.

The experimental results are promising, as the LottoTU app correctly reads out lottery numbers from 59 out of 60 test images (including those shown in Fig. 7). In the lottery number segmentation step, 33, 22, and 4 images are segmented successfully using the parameter sets (T_1, B_1) , (T_2, B_2) , and (T_3, B_3) , respectively. This clearly shows that using only one set of parameters is not adequate for lottery number segmentation. One of 60 test images could not be read out because the lottery ticket appears very small and blurry, and the yellow watermarks are not detected. In our experiments using an Android smartphone version 5.1.1 with 3 GB of RAM, after the user captures the image, the LottoTU app takes approximately four seconds to detect and read aloud the six-digit lottery number.

For best results, we recommend that when the LottoTU app is used to capture an image, the distance between the camera and a lottery ticket should be between three and eight inches. Otherwise, the lottery ticket image would appear too large (fewer than six digits are captured) or too small (yellow watermarks are not detected). In addition, LottoTU fails to detect the lottery number in the dark.

SECTION V. Conclusion

In this work, we present a new Android app called LottoTU for scanning and

In this work, we present a new Android app called LOTTO-TU for scanning and reading aloud Thai lottery numbers for blind lottery ticket sellers. We employ three different threshold values and morphological structuring element sizes to perform lottery number segmentation. The segmented image is thresholded using Otsu's method to obtain the binary image of a six-digit lottery number. Next, the binary image is converted to text data by the Tesseract OCR engine and then read aloud by Android Text-to-Speech API. The experimental results show that LottoTU successfully scans and reads aloud lottery numbers with high accuracy from images taken under varying illumination conditions and at different orientations and distances to the camera. Compared to prior work, our proposed app is easy for the blind to use and does not require an additional hardware or a box to operate with.



Authors
Figures
References
Keywords
Metrics

IEEE Account

Profile Information

Purchase Details

Need Help?

Other

A not-for-profit organization, IEEE is the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity.
© Copyright 2018 IEEE - All rights reserved. Use of this web site signifies your agreement to the terms and conditions.

US & Canada: +1 800 678 4333
Worldwide: +1 732 981 0060

IEEE Account	Purchase Details	Profile Information	Need Help?
» Change Username/Password	» Payment Options	» Communications Preferences	» US & Canada: +1 800 678 4333
» Update Address	» Order History	» Profession and Education	» Worldwide: +1 732 981 0060
	» View Purchased Documents	» Technical Interests	» Contact & Support

About IEEE *Xplore* | Contact Us | Help | Accessibility | Terms of Use | Nondiscrimination Policy | Sitemap | Privacy & Opting Out of Cookies

A not-for-profit organization, IEEE is the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity.
© Copyright 2018 IEEE - All rights reserved. Use of this web site signifies your agreement to the terms and conditions.