A Mobile System of Reading out Restaurant Menus for Blind People

Takehiro Sakai[1], Tetsuya Matsumoto[1], Yoshinori Takeuchi[2], Hiroaki Kudo[1] and Noboru Ohnishi[1]

[1]Graduate school of Information Science, Nagoya
sakai@ohnishi.m.is.nagoya-u.ac.jp
{matsumoto, kudo, ohnishi}@is.nagoya-u.ac.jp

[2]Faculty of Informatics, Daido University
ytake@daido-it.ac.jp

ABSTRACT

We propose a system that reads out character information in the surrounding environment and informs users via synthesized voice. The proposed system is implemented in a mobile device, such as a smart phone and a tablet PC, by using cloud computing. An image captured by the mobile device is processed by Character Recognition API (NTT docomo) in cloud computing and recognized words along with their positions in the image are sent to the mobile device. In the device, in case of restaurant menu, words are combined into food name and the correspondences between food name and its price are determined. Finally corresponded food with its price is outputted sequentially as synthesized voice. As a result of experiment, the correct correspondence rate is about 87% and the processing time is about 10 seconds. We also conducted a field test to investigate the usability by the visually impaired person. As a result, we confirmed that a totally blind user can manipulate the system and, take a photograph in a right angle and appropriate distance from a camera to a menu.

1. INTRODUCTION

Blind persons have various problems in their daily lives. These problems are divided into localization-and-movement and information acquisition. Information acquisition are divided into character information and diagram information. As for character information, braille was devised for blind people to use to read and write letters. Electronics devices, such as Optacon, Kurzweil reading machine, braille word processors, screen readers, document reading system using a scanner or a Tablet PC etc. have also been developed [2]-[6]. As for diagram information, our laboratory developed the system which supports recognition and expression of a figure [1].

Our laboratory also developed a notebook PC-based system [7], which helps blind people to get character information in their surrounding environment. This system needs a notebook PC and mobile a scanner. It is not realistic, however, for blind persons to carry a Note PC and a scanner in a daily life because a notebook PC is not inexpensive and bulky.

Nowadays mobile devices such as smart phones and tablet PCs are popular, and many people including blind persons use them in daily life. Especially in the United Kingdom and the USA, Blackberries and iPhones are everyday-life-tools for blind people and have a function of optical character recognition (can perform OCR) and read-aloud of signs, menus, and various kinds of text existing in the environment. SayText [9], KFNB Reader [10], Prizmo [11] are examples of similar application software.

These applications, however, are unavailable for Japanese texts. And, they need to read taking account of the correspondence between food/goods name and its price for restaurant menus and store leaflets. They sometimes include emphasized characters such as thick characters and large characters. So those OCR applications should emphasize these characters not just reading characters.

Nowadays, the data transmission speed is also accelerated quickly and cloud computing attracts more attention. By using cloud computing, even mobile devices with limited computation power can achieve high performance in accuracy and speed.

Therefore we planned to develop a mobile system by using cloud computing. In the system, we use Character Recognition API provided by NTT docomo. This API can perform extraction and recognition of words in an image. By using this, we have developed the system that can extract character information in the surrounding environment and inform users via synthesized voice.

In this paper, we focus restaurant menu as character information based on an interview which our laboratory conducted to the visually impaired person. The interview result shows that they want to know product name and price information, when purchasing goods.

So we chose a restaurant menu in which product (food) name and its price information should be related. In case of a menu, we have to combine words recognized by API in cloud computing into one food name, and also to relate food name with its price. These processing for combining should be done in the mobile device.

In the sections following this introduction, we describe a proposed system in Section 2, experiments and their results in Section 3, discussions in Section 4, and conclude this paper.

2. PROPOSED SYSTEM

2.1 System Overview

Figure 1 shows the overview of the proposed system. The system was implemented on the mobile device (SHARP, SH-06E, Android OS). An image captured by the device is sent to the cloud service. It extracts the character area in the image by using the character recognition API provided by docomo Developer support [8]. The outputs of this cloud computing are recognized words along with the coordinates of their enclosing rectangles in the image, and are sent to the mobile device.

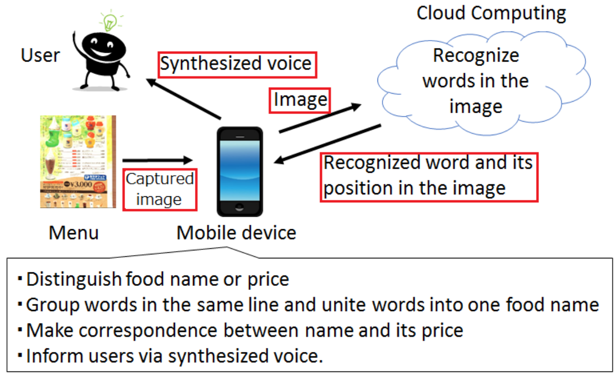
Cloud Computing
User
Synthesized voice
Recognize words in the image
Image
Captured image
Recognized word and its position in the image
Menu
Mobile device
・Distinguish food name or price
・Group words in the same line and unite words into one food name
・Make correspondence between name and its price
・Inform users via synthesized voice.

Figure 1. Overview of the proposed system.

Although character strings such as food name and price are composed of several words, the character recognition API recognizes only each word. Therefore in the mobile device, we have to distinguish each word whether it is price or a part of food name, and group split words into one food name or one price. Furthermore we have to determine correspondence between food name and its price. Finally, the system reads out the corresponded food name and its price by the speech synthesis engine.

2.2 Processing in Cloud

There are various services and software which provide OCR function. However, these general OCRs are limited to a simple image of characters onto a uniform background. In the real environment, characters are found on a complex background, such as signboards, merchandise packages, menus and so on. It is an important subject to extract characters in such complex images.

NTT docomo has developed high-precision OCR technology working even in such complex images. It just matches words in the image against the dictionary data of language. Therefore, even if character strings are composed of several words, the API recognizes only each word.

Docomo character recognition API processes an input image including character information and output recognized text and the coordinates of its enclosing rectangle. The size of an input image must be not larger than 4000 x 4000[pixels]. The time needed for processing is about 10 seconds.

2.3 Processing in Smart Phone

Here we explain the processing for grouping recognized words done in a smart phone. The words and their coordinates of the enclosing rectangles in the image are given to the smart phone by the cloud service. A word is determined to be a part of price if it includes the currency mark "￥" or the character representing money unit "円". **Figure 2** shows the result of recognition of the API. The yellow rectangles represent each of recognized words by the API. The light-blue one represents a part of price because it includes the currency mark "￥".

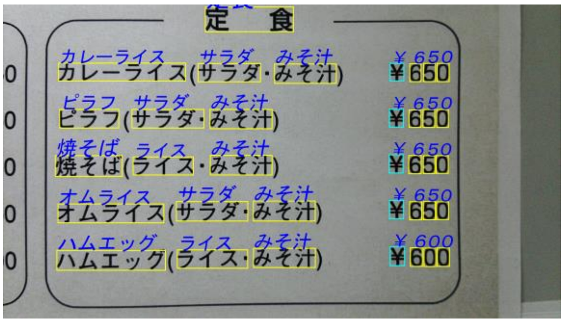
定 食
カレーライス(サラダ・みそ汁)
￥650
ピラフ(サラダ・みそ汁)
￥650
焼そば(ライス・みそ汁)
￥650
オムライス(サラダ・みそ汁)
￥650
ハムエッグ(ライス・みそ汁)
￥600

Figure 2. The result of recognition.

The API recognizes separately only words or digits composing food name or price. For example, as shown in the lower left of **Figure 3**, numerals representing price(650) are recognized every digit (6, 5, 0). So, in order to simplify the subsequent processing, the words in the image are grouped for every line. Grouping is performed by making the words on the same line into the same group. Then, the system combines the words that are included in the same food name or the same price. The words are combined if the distance between the words is less than a threshold value. **Figure 3** shows the images before and after combining words. In the left figure, the yellow rectangles represent each of recognized words by the API and the light-blue one the character representing money unit "円". In the right figure, the red rectangles represent the combined food name and price.

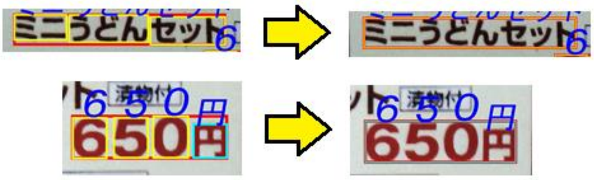

Figure 3. Before and after combining words.

Next it determines the correspondences between food name and price. It is assumed that the correspondence between food name and price is one-to-one correspondence. Since restaurant menus include various layouts, food name and its price sometimes may not be located on the same line (See **Figure 4**). In order to cope with such a problem, we select candidate pairs satisfying the following conditions:

1. Price is not located at the left of food name.
2. Price is not located above food name.
3. The character size of food name is large.

About the third condition, smaller characters than the average character size in a menu represent caption of the menu instead of food name in many cases (see **Figure 4**). Therefore, in order to prevent incorrect correspondence, food name with smaller character size is removed from the candidates of correspondence. Next, we select the pair with the shortest distance between food name and price (See **Figure 5**). The distance is calculated as Euclid one between the centers of gravity of enclosing rectangles for food name and price.

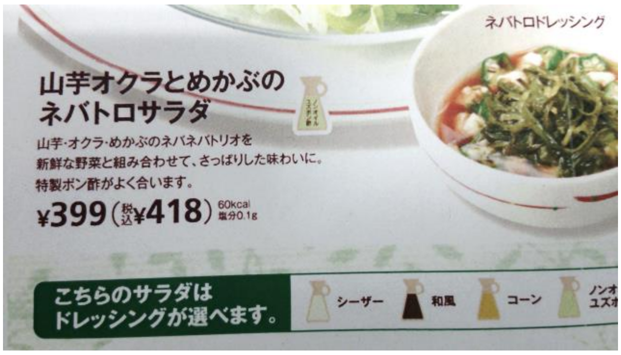

Figure 4. An example with a complicated layout. Food name and its price are not on the same line and its caption are between them.

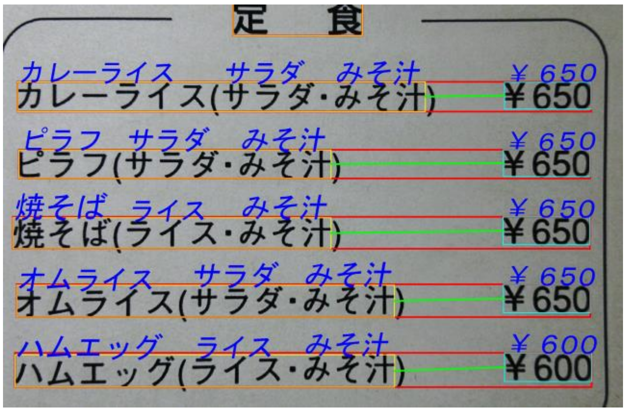
定 食
カレーライス サラダ みそ汁 ¥650
カレーライス(サラダ・みそ汁) ¥650
ピラフ サラダ みそ汁 ¥650
ピラフ(サラダ・みそ汁) ¥650
焼そば ライス みそ汁 ¥650
焼そば(ライス・みそ汁) ¥650
オムライス サラダ みそ汁 ¥650
オムライス(サラダ・みそ汁) ¥650
ハムエッグ ライス みそ汁 ¥600
ハムエッグ(ライス・みそ汁) ¥600

Figure 5. The result of matching. Green lines represent the correspondence between food name and price.

Finally, all the pairs of corresponded food name and price are read out by a voice synthesis engine sequentially from the left-top to the right-bottom in the image.

2.4 User Interface

What the user should do to start the system is just touching consecutively two times the home button of a smart phone. Then the user hears the announcement that” application has started.”

Next he or she takes a picture of a menu with the camera of the phone by touching the screen. Since the camera of a smart phone has autofocus function, a high quality image is captured. And the user hears the announcement of “in recognition”. After about 10 seconds, menu-reading starts. If the user wants to hear again, he or she touches the screen.

2.5 Photography in Blindness

When using the system, it is desirable for the user to keep a smartphone as small rotation around the camera axis and parallel to a menu as possible. If the angle of camera rotation is more than about 15 degrees, the correct correspondence of food name and price will not be on the horizontal lines, and this causes miss-correspondence. Furthermore, when taking a photograph, the user keeps a suitable distance between a camera and a menu.

So we propose the operation for blind users to successfully capture a menu image by using the system. First, he or she places a smartphone on a menu so that the smartphone edges are parallel to the edges of the menu. Next he or she lifts the smartphone right above to the height of about 10 cm as it is. Finally the user just touches a screen once.

By using this operation, we can acquire the good quality pictures even if we wear an eye mask. So, blind users also will be able to acquire pictures well.

3. EXPERIMENTS AND RESULTS

We conducted three experiments. First, we investigated how the recognition accuracy by the API is affected by the distance from a camera to a menu. Second, we investigated the accuracy of the correspondence of food name and its price. Third, we conducted a field test by visually impaired people, and evaluated the feasibility of the system.

3.1 Experiment 1: Recognition rate and character size

It is well known that the accuracy of the character recognition by the API changes with the character sizes in an image. So we investigated the recognition accuracy by changing character size, i.e. the distance between a camera and a menu.

We used 14 images of the same menu (3840 × 2160 [pixels]). The system outputted the results within about 10 seconds after capturing. And we classified these images by the ratio of the average height of food name to the height of an image ([pixel] / [pixel]) contained in the image. We counted the number of recognized food name and price which coincided partially or wholly with true character strings for each group classified with the ratio. The result is shown in **Table 1**.

We find that recognition accuracy is bad in the ratio of a character height to an image height being less than 0.03. So, it is said that the ratio suitable to recognition is more than 0.03.

3.2 Experiment 2: Accuracy of the correspondence

Food name is related with its price by the method explained in 2.3. We investigated the accuracy of this correspondence and evaluated the validity of the method. We used 75 menu images of 10 restaurants captured by the smart phone under the camera distance of 5 to 10 cm. There are 368 pairs of food name and price in 75 images. **Table 2** shows the experiment result. Among 368 pairs, only 164 pairs are founded by the proposed method. The extraction rate of pairs is about 45%. This low rate is due to low rate of word recognition by API. Among 164 founded pairs of food name and price, the number of correct correspondence is 143 (87.2%) and that of miss-correspondence 21 (12.8%).

Character height / Image height	# of Image	# of food name and price	# of wholly coincidence	# of partial coincidence
0.015 to 0.020	1	42	0 (0%)	3 (7%)
0.020 to 0.025	3	107	31 (29%)	24 (22%)
0.025 to 0.030	2	60	12 (20%)	9 (15%)
0.030 to 0.035	3	48	18 (38%)	19 (37%)
0.035 to 0.040	4	60	33 (42%)	29 (36%)
0.040 to 0.045	1	18	8 (44%)	8 (44%)

Table 1. Comparison of the recognition accuracy by the ratio of the height of a character to that of an image.

# of found pairs	164(45%)
# of correct correspondence	143 (87%)
# of miss-correspondence	21 (13%)

Table 2. Accuracy of the correspondence between food name and price.

3.3 Experiment 3: Field test by visually impaired people

We conducted a field test to confirm that a visually impaired person also can use the system. Two subjects cooperated with the test. One subject is a male and the other is a female. And they are total blindness.

The procedure of the experiment is as follows. First, we explained them the operation of the system. The operation is a series of tasks from starting the system to taking a picture. Next, we asked them to practice the system operation one or two times. Then, they performed five trials of menu reading by the system. Every trial is a series of tasks from starting the system to taking a picture. The menu used for the experiment is the same one. After all trials, we interviewed them about the usability.

We show the result of those trials in **Table 3**. "# of price" and "# of food name" mean the number of all the price items and name items contained in the picture. "# of all the pairs of name and price" means the number of correct pairs of food names and prices in the picture (It contains the pair matched by the system, and the pair which was not matched). "# of correspondence in found pairs" and # of miss-correspondence in found pairs" mean the numbers of correct correspondence or incorrect correspondence as the result of matching.

	Subject 1	Subject 2
Sex	Male	Female
Vision	Total blindness	
# of images	5	
# of price	27	15
# of food name	34	17
# of all the pairs of name and price	20	12
# of found pairs	20	9
# of correspondence in found pairs	18	7
# of miss-correspondence in found pairs	2	2

Table 3. The result of the field test by visually impaired people's trial.

The number of all the pairs and the number of correspondence in **Table 3** explain that the combination of food names and prices are acquirable for visually impaired people using the system. An example of the images taken by the subject 1 is shown in **Figure 6**.

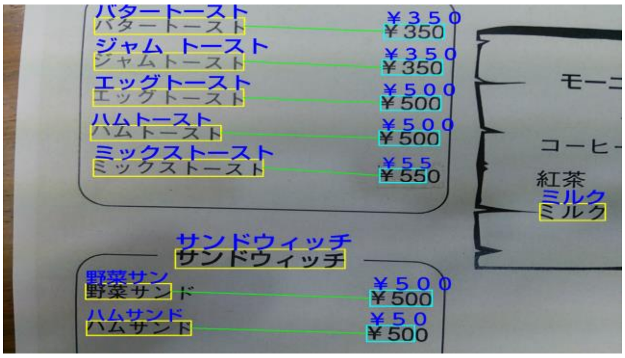

Figure 6. An example of successful results.

However, the number of price items is different from that of food name items, and there are fewer pairs of correct correspondence than the number of food name or price items. This means that either of price or food name exists in an image and the other doesn't exist in the image (See **Figure 7**). The reason is as follows. The layout of this menu is two columns. But the subjects were not informed in advance about the menu layout, and took a picture among two columns. In order to prevent this, the subject needs to understand the layout in advance. After they were taught the layout, they could take the picture more exactly than before.

Next, we interviewed them about the usability. Both of them answered that the operation of the system itself was possible. They, however, also answered that it is difficult to take a suitable picture for the system because of being not able to grasp the menu layout. And one of them answered that it is hard to grasp the place of a home button for starting the system.

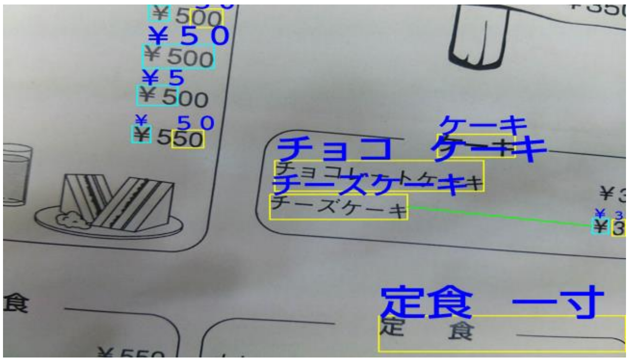

Figure 7. An example of failed results.

4. DISCUSSIONS

First we discuss the accuracy of character recognition by the API. From the experiment 1, it is said that the API can correctly recognize characters whose height is more than 120 pixels in an image with a size of 3840 longitudinal pixels and 2160 lateral pixels. The result of the experiment 2 is that among 368 food pairs, only 164 pairs are correctly recognized by the API. The recognition accuracy is 45%, and especially that of price is lower. So, we expect the API to be improved or we have to develop an OCR software for extracting characters in an image.

Next, we discuss the accuracy of processing for corresponding food and its price in a mobile terminal. Among 164 pairs, 143 pairs (87%) are correctly corresponded (the experiment 2). This seems good performance.

The third is usability. When taking picture, users need to keep a camera about 10 cm above a menu. In this situation, they can’t take the whole of a menu but capture only its part. The probability that such images contain both food name and price is low. An idea for solving this problem is that after capturing the whole of a menu, the system analyzes the layout of the menu and extracts a subimage containing both of food name and price. Another point of the usability is to incorporate a function for preventing miss-touch because of the blindness.

Finally, another improvement is as follows. To prevent miss-correspondence between food name and price, the rotation angle of the image should be less than ±15°. Although the proposed operation for photographing can minimize the rotation, an automatic function for correcting image rotation is desirable. This will be realized by using a gyro sensor in a smart phone.

Now, there are various text information in the surrounding environment except restaurant menus. We asked blind persons what kind of text information they want to understand. The answers are display labels with price and consume-by date or use-by date, washing labels and signboards in the outdoor etc. So we are also going to make the system available for them.

5. CONCLUSION

We have proposed the mobile system of reading out restaurant menus for blind people. We implemented the system on a smart phone and obtained satisfactory performance by the experiments. We also conducted the field test to confirm that a visually impaired person can also use the system. And we confirmed the feasibility of the system.

A future subject is to develop a method which can capture the whole of a menu and extract a partial image which contains both of food name and price with character size suitable to OCR by considering its layout. Another one is to extend the system to process other character information.

REFERENCES

[1] H. Minagawa, N. Ohnishi, N. Sugie, “Tactile-Audio Diagram for Blind Persons”, IEEE Transactions on Rehabilitaion Engineering, vol. 4, no. 4, December 1996.

[2] L.Kay, “Electronics aids for blind persons: An interdisciplinary subject.“ IEE Proc., vol. 131. pt.A.pp. 559-576. July 1984.

[3] Thatcher, Jim., “Screen Reader/2—programmed access to the GUI”, Springer Berlin Heidelberg, 1994.

[4] S. Morley, “Window concepts: An introductory guide for visually disabled users.” GUIB Consortium, 1995.

[5] Amedia, The document reader "Yomube-Smile", http://www.amedia.co.jp/product/ys/

[6] Kochi System Development Corporate, The document reader "Pashat-Reader", http://aok-net.com/news/pashatnews.html.

[7] N. Ohnishi, T. Matsumoto, H. Kudo , Y. Takeuchi, “A System Helping Blind People to Get Character Information in Their Surrounding Environment”, the proceeding of ASSETS 2013.

[8] NTT Docomo, “Character Recognition API”, https://dev.smt.docomo.ne.jp/?p=docs.api.page&api_docs_id=9&lang=1

[9] Haave Oy, “SayText”, http://www.docscannerapp.com/saytext/

[10] K–NFB Reading Technology Inc., “KNFB Reader”, http://www.knfbreader.com/

[11] Creaceed SPRL, “Prizmo”, http://www.creaceed.com/prizmo