19th International Computer Science and Engineering Conference (ICSEC)
Chiang Mai, Thailand, 23-26 November, 2015

UCap: A Crowdsourcing Application for the Visually Impaired and Blind Persons on Android Smartphone

Apirak Hoonlor*, Srisupa Palakvangsa Na Ayudhya†, Sukritta Harnmetta‡, Suttichai Kitpanon§, Krisanat Khlaprasit¶
Faculty of ICT, Mahidol University, Bangkok, Thailand
Email: {apirak.hoo*, srisupa.pal†}@mahidol.ac.th,{sukritta.har‡, suttichai.kit§, krisanat.khl¶}@student.mahidol.ac.th }

***Abstract*—One of the visual challenge problems that the blind faces is the consumer product identification with contextual and description information problem. In order to increase their independence in food shopping and other product recognition, we implement an Android application called UCap to assist the blind in this visual challenge problem. In the visually impaired and blind persons mode, UCap is a camera-base mobile application that identifies the consumer product of a captured image using the UCap annotated image database. The UCap annotated image database is created using the crowdsourcing paradigm. In the sighted user mode, the user can use UCap to capture an image of a consumer product, add its description, and upload them to the UCap annotated image database. The sighted user can add more images of the existing products in the database. The seed database contains 3,950 annotated images. We used Infrastructure-as-a-Service (IaaS) on MS Azure cloud server for the initial system testing and evaluation. With exception of the lower-than-expected accuracy of image identification, the application received high praised from the visually impaired and blind persons. However, from the experiment and testing, the accuracy of UCap can be increased as more images are added to the database.**

Keywords: Mobile Application, Crowdsourcing, Camera-based Application

I. Introduction

According to the World Health Organization, there are an estimate of 285 millions visually impaired persons in the world [15]. Most of the time, these visually impaired persons can live in the society without any assistants. However, they do have visual challenges that requires assistants. A large scale study was conducted in [2] to identify the visual challenge problems for the visually impaired person. The visual challenge problems were summarized and grouped from 40,780 images related questions took and asked by 5,329 visually impaired users via a social mobile application VizWiz over a year. The study identified four groups of problems: identification, reading, description, and unanswerable. We tried to identify common problems that the blind and visually impaired persons in Thailand have in common with the study by interviewing 15 visually impaired persons. 8 persons were the visually impaired workers at Dialog in the Dark exhibition [3] in Bangkok, Thailand. The other were the students and staffs at Mahidol University and Rajchasuda College, Thailand. One common visual challenge problem that they have is the item identification with contextual and descriptive information when shopping. Specifically, the visually impaired people cannot see the unit's price, serving size, ingredient, and nutrients of the targeted items. As such, they need the assistants to read them the information on the food items' packages during the food shopping.

To help the visually impaired and blind persons be more independent, many researchers and developers have developed frameworks, software, and applications to assist them. One approach to solve this item identification visual challenge is inform of a camera-base mobile application such as LookTel [11] and TapTapSee [9]. One other approach is to utilize a crowdsourcing paradigm, such as the Be My Eyes [1]. Crowdsourcing paradigm is a powerful framework that distributes tasks to a group of human to solve [16]. It is applied in various applications solving different tasks such as LabelMe [13] and PlateMate [12]. Please see section II for additional details on these related applications.

For this work, we develop UCap: an Android application for identifying consumer products and informing the nutrition information to support visually impaired people in their grocery shopping. Our application is implemented using the ideas and frameworks from the camera-base mobile application and the crowdsourcing application. UCap has two modes: the visually impaired and blind user mode, and the sighted user mode. For the visually impaired and blind user mode, the user will be able to ask UCap to identify the consumer product via its image search function. We use the crowdsourcing paradigm to develop the sighted user mode. In this mode, the sighted user will capture the image of the consumer product, add the product's information, and then upload them to the database. As more images of the products are added, the application is getting better at its identification task. As a by product, this work produces a growing annotated consumer product images database, UCap annotated image database. For the rest of the document, the term visually impaired person implies both the visually impaired and the blind persons.

II. Literature Review and Related Work

The related work of our application can be categorized into two groups: the object recognition application, and the

978-1-4673-7825-3/15/$31.00

crowdsourcing application. TapTapSee [9], and LookTel [11] are the mobile camera applications that recognize objects for the visually impaired persons. Both applications can recognize everyday life object such as DVDs, consumer products, and bank notes. To use TapTapSee, the user has to hold the to-be-recognized object up close for taking its photo by double-tapping on the device screen. According to the review in [7], the TapTapSee application is accurate if the object is available in its extensive database. LookTel has a separate application for bank notes, and other objects. In addition, it can also scan the product's barcode. Be My Eye identifies the objects by the help of sighted people. This application needs the sighted volunteer helpers around the world, and supports many languages. Visually impaired person can request assistance via the application. The request will be forwarded randomly to a volunteer. The volunteer will receive a notification, a video call, and answer the question that visually impaired people asked. In regard to assisting the visually impaired persons, Be My Eye is then considered crowdsourcing. TapTapSee requires a subscription for its services with various packages, and LookTel costs $9.99. Be My Eye is currently the only free application. However, according to [8], the service time for each identification request is depended on the response time from the volunteers which may take longer than 2 mintues. We believe that the service time should be reduced as more volunteers use the application.

The crowdsourcing application exists in various areas ranging from public health to entertainment. The crowdsourcing related to our work is the image database construction via crowdsourcing. One standout application is LabelMe [13]. LabelMe is a web-based tool that allows human to identify the embedded objects in an image into predefined classes. The other food image annotation using crowdsourcing is PlateMate [12]. This work is slightly different in what we propose in a sense that we identify a consumer product, but PlateMate estimates the food intake based on the image of food on the plate. PlateMate tagged each food item in the image first. Then, it identifies each item, and estimates the food intake using Amazon Mechanic Turk [12]. The annotation quality from crowdsourcing is found to be promising and usable by [14]. [16] provides a lengthy survey on crowdsourcing system.

III. UCap Application

UCap is a crowdsourcing application that identifies consumer product images for the visually impaired persons. It is a camera-based mobile application. The application has two modes: the visually impaired user mode, and the sighted user mode as shown in Figure 1. UCap utilizes the crowd for consumer product identification. For visually impaired user, this application will provide an image searching function. The visually impaired user can tap on screen and this application will show the nutrition facts and food additives. The application allows Google TalkBack (GTB) [5], an eye-free-speech-endabled application, to read this information to the user. In the sighted user mode, the user can add a product's image along with its annotation and nutrient information. The sighted user can upload a new product image from the smartphone by taking a photo or selecting an image directly from gallery.

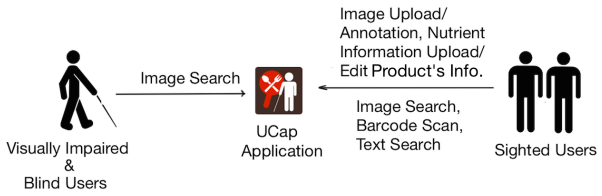


Fig. 1: Users and the functionality of UCap

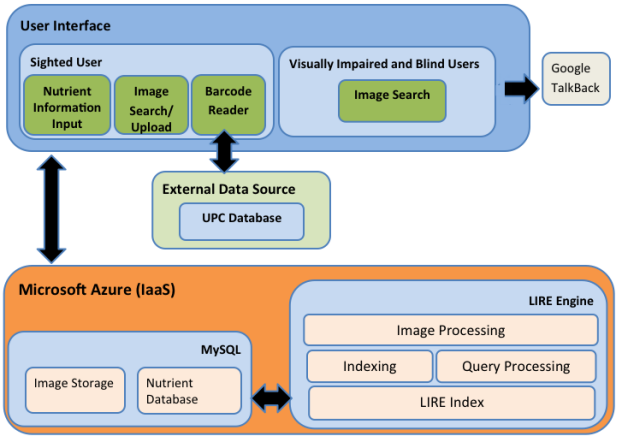


Fig. 2: System Architecture and Information Flow Overview.

UCap composes of three parts: the consumer product identification, the crowdsourcing database, and the user interface. The image recognition technique and crowdsourcing paradigm are the cores of UCap for identifying a consumer product from its image. The product identification task is solved as a content based image retrieval task (CBIR). Specifically, the crowdsourcing paradigm is used to collect the consumer product images and their corresponding nutrient information. Then, the image index is constructed for the retrieval system.

A. System Architecture Overview

The system architecture can be viewed as two parts: the client on the Android smartphone device, and the server on Microsoft Azure (IaaS), as shown in Figure 2. The consumer product identification task, and the nutrient data retrieval task are solved on the server side. The client side acts as the interface for both the sighted users, and the visually impaired users. The consumer product identification task is solved as the image retrieval task in our application as discussed in Section III-C. The consumer product identification task is tied to the nutrient data retrieval task. Specifically, every time the image is indexed or retrieved, the nutrient data must be respectively indexed or retrieved. The feature values are extracted from the image by using Lucene Image Retrieval library (LIRE). Then, they are either used as query, or inserted as one of the indexes. The nutrient data is read for the blind via GTB. If the sighted user uses a barcode reader to retrieve the product information, the barcode query is sent to the Internet UPC database [4]. Then, the product's information is forwarded to the server.

B. User Interface

The interface on the client side is separated into the blind, and the sighted modes. The visually impaired users can only use the consumer product image identification service via the image search function. The application starts from the home page, shown in Figure 3a, where the user must hold for the blind mode. If the user is in the blind mode, the user is transfered from the home page to the camera. The user must take a photo of the consumer product using the camera on a mobile phone, and upload it to the server by performing the double tap on the Android smartphone. If the consumer product is found on the index, its name, nutrient information, and ingredient information are displayed in three tabs as shown in Figures 3b, 3c, and 3d respectively. The information on these three tabs can be read back to the users via GTB. The final tab is the search tab, in this tab the user can restart the process of identifying the next consumer product. UCap reports that the product is not found, if the retrieved product does not pass certain threshold. Then, the user is taken to the search tab.

For the sighted user, from the home page (Figure 3a), the user can hit anywhere on screen to enter sighted user mode. In the sighted user mode, the first page is the function selection page, shown in Figure 4a. The user can add the consumer product to the UCap index by either uploading an image from the photo library, or choosing the "Take a Photo" function. If the take photo function is selected, the user can take a photo and crop the image (shown in Figure 4b). The consumer product image is converted to query, and used to retrieve the top-four similar consumer products. For easy viewing on a $4-5$-inch screen size, only top-four consumer products are returned to the user as shown in Figure 4c. We believe that any image of size smaller $1.5 \times 1.5\ inch^2$ is too small to see.

If the queried product does not exist in the returned four products, the user can select "no match" option at the bottom of the page. Then, the user can enter the information by entering it directly via "custom" option, or using the barcode reader as seen in Figure 4d. For the former, the user can enter the name, the nutrient and the ingredient information, as seen in Figures 4e, 4f, and 4g. For the latter, UCap will call up the barcode reader , as seen in Figure 4h, and send the barcode query to the Internet UPC database. Then, for both cases, the consumer product image and its nutrient information are uploaded to the server.

If the queried product is in the returned four products, the user can click the image of the matched consumer product to view and edit the existing information. The nutrient and ingredient information on the server are retrieved and shown for editing via the custom option function, discussed earlier. If the user has the limited data plan, the user can simply store the product's image in the phone library. Then, via WiFi, the user can upload the image. Also, the user uses the "Search by Image" and "Search by Name" functions to check if the product exists in the UCap database. UCap only sends the minimum the text information in the Search by Name function.

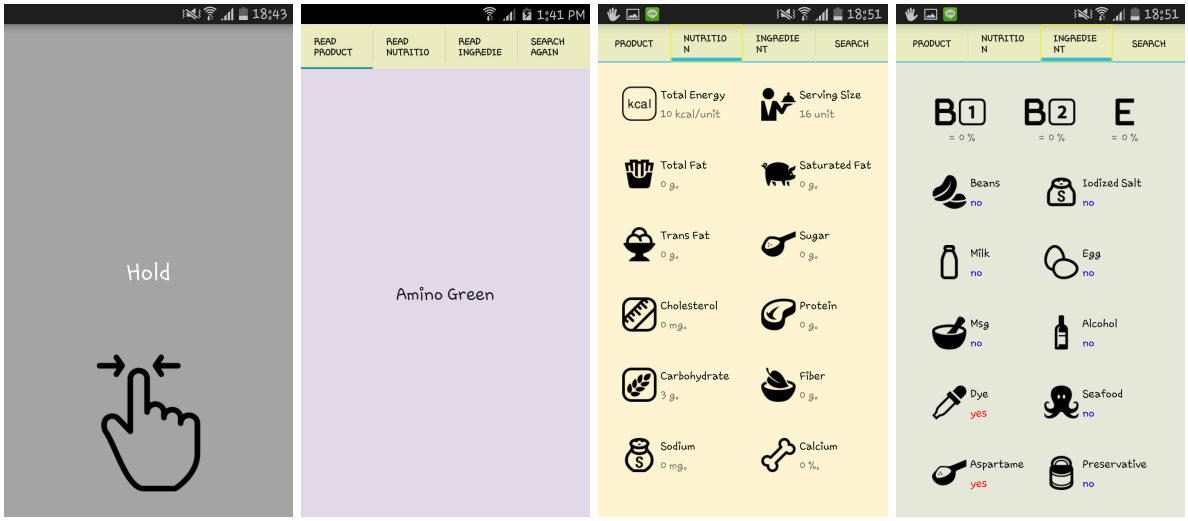
C. Consumer Product Identification

We define our consumer product identification task as "**given an image, identify its corresponding consumer product, and report its nutrient and ingredient information**". The scope of our work is the consumer products with nutrient and ingredient labels. In this first phase of data collection, we focused only on the products sold in supermarket due to the light setting environment. The product identification task is a challenging problem because there are numerous unique product packagings in the market. The reason for this is the variation in brands, packaging, serving size, and flavors. For example, we found that there are over 100 kinds of potato chips carried in a Thai supermarket. Also, different lighting conditions cause the images of the same product to be different. Adding to the complexity, based on our observation, the visually impaired users did not know if they took the picture of the front, the back, or the sides of the package. To be able to retrieve the right product of an image taken by the blind, we need multiple angles of the same product from various setting.

We solved the product identification task by using the CBIR concept. CBIR is an image search system optimized for a very large image database. The CBIR system contains three main parts: the index, the query, and the result ranking. For UCap's CBIR, the images are indexed by first converted them into the vectors of the predefined features, and indexed. For our query system, the searched image is converted into feature values according to the predefined features. Then, the values are used to find the similarity ranking of all images in the index. Finally, the top-k images of each feature are retrieved for product identification (see Section IV for more detail). Then, a product ranking is created from these top-k images. For the sighted user, the application returns the four-highest product ranking as the result. For the visually impaired users, our system returns the product with the highest product ranking. For UCap image retrieval system, we used Lucene Image Retrieval, a Java library for CBIR [10].

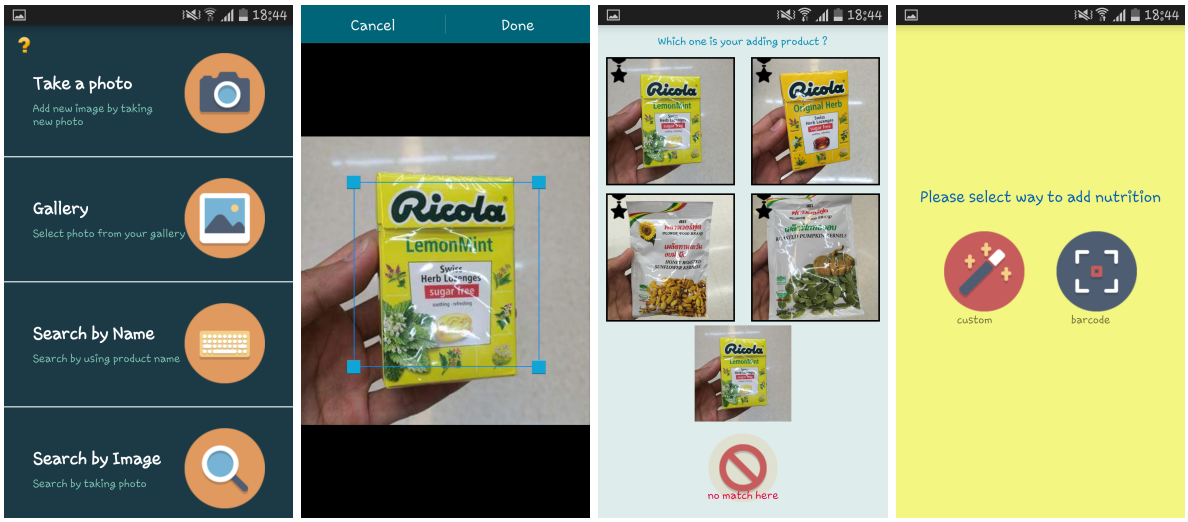
D. UCap Database, Seeding Data, and Crowdsourcing Consumer Product Image Annotation

The UCap database has three components: the image file storage, the nutrient database, and the image index. Once the image is uploaded, its features are extracted and indexed by LIRE. The image file is kept separately in the storage where it is retrieved only to display for the sighted user. The product's name, ingredients, and nutrient information are stored in the nutrient database. Each product is given a unique identification number by the system, and used as the primary key in the nutrient database. The path for the image file, and the product ID are kept along with the LIRE index. Tesco Lotus Thailand kindly allowed us to take the photos of its carried consumer products. From which, we create the seeding data by taking $3,950$ images of the 296 products. As such, we only have one lighting and background condition for all images. To identify the product by an image taken by the blind, as mentioned above, we need multiple images for each product. Hence, we

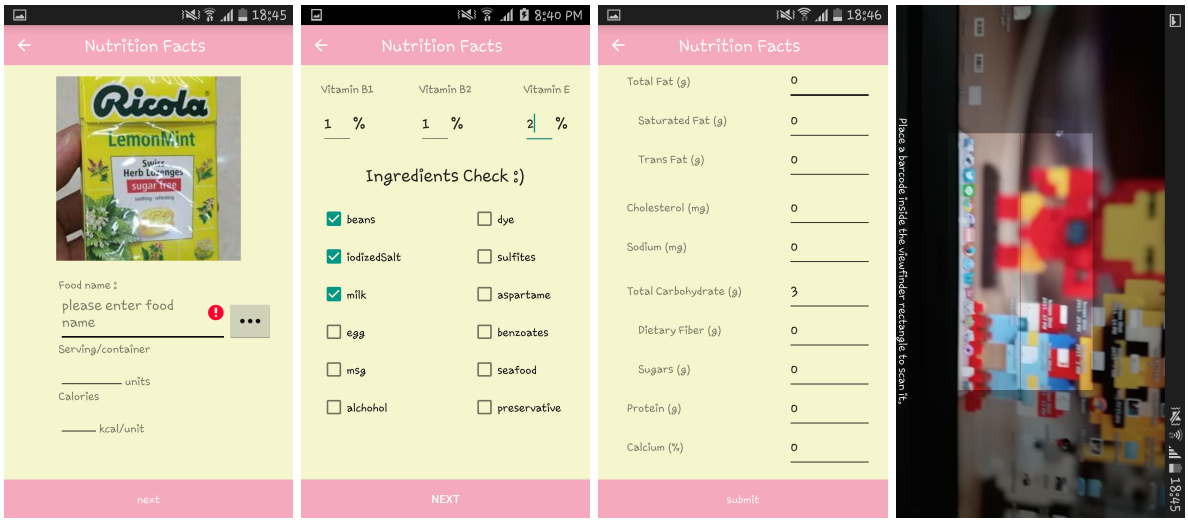


(a) Home Page (b) Product's Name (c) Nutrient Information (d) Ingredient Information

Fig. 3: The screen shot of the user interface for the visually impaired users.



(a) Function Selection (b) Image Capture (c) Product Retrieval (d) Method to Add Information



(e) Enter Product Name (f) Enter Ingredients (g) Enter Nutrient (h) Barcode Reader

Fig. 4: The screen shot of the user interface for the sighted user.

need a very large image database to accurately identify the large number of products in the market using as a camera-base application. We use the crowdsourcing framework to construct this database. In our system, the sighted users are the image creators, and the image annotators. As the crowdsourcer, the user takes an image of a consumer product, adds its annotation, and uploads them to the system.

IV. Implementation and Configuration

The mobile prototype is implemented and tested on Samsung Galaxy S5 with 5.1 inches display, 2GB RAM, and ran on Android OS version 4.4.2. We use Android-sdk for its API in our implementation. All the communications between the server and the client are sent and handled using JSON and PHP codes. The UCap database management system runs on Apache 2.4.10, MySQL 5.6.21, and FileZilla FTP Server 0.9.41. Finally, we use LIRE 0.9.4 beta 2 version. In order to increase the scalability, all environment is setup and run on Microsoft Azure (IaaS).

To use LIRE for CBIR, the list of features to represent the image, and the post-processing of the search results must be defined. Ideally, the image retrieved for the blind should be 100% correct, while using as little features as possible. The smaller the number of features is the smaller the index, hence a shorter time and smaller space to manage. LIRE can extract the following 10 features for image indexing such as Auto Color Correlogram (ACC), Color Histogram, and JCD, (see [10] for full list). For LIRE, when an image is searched, the top-k similar images of each feature are retrieved. LIRE retrieved the lists of images, but we need the list of product's information. Therefore, we have to construct a product ranking from these lists of images.

For our work, we choose the features for indexing using forward selection with wrapper approach (see [6] for further information). For this feature selection experiment, we use our seed dataset. We randomly select 30 products and took their images for validation. A single image is retrieved from LIRE index of $3,950$ images. If the image retrieved is of the same product queried, we count that as 1 correctly identified query. The best set of features has the highest correctly identified queries. ACC and JCD are found as the best set of features.

Intuitively, given an image of a searched product, we want the system to retrieve the correct product. As such, we have to construct the product ranking list from the lists of top-k images found from ACC and JCD features. Therefore, the product ranking is the result of this post-processing step. For post-processing, the product's name of each image from the lists of top-k images are retrieved. Then, each unique product is given a score based on a scoring function. We experiment on using the sum of reciprocal ranking, and the total recall rate as the possible scoring function. We use the 30 searched images from the first experiment, setting ACC and JCD as features. We fix the number of retrieved top-k images to 20 images. Both of them achieve 100% accuracy. For this experiment, because of the small number of images per product in our database, LIRE retrieves most images of the queried products in the database in top rankings on both lists. Since the recall rate, and the sum of reciprocal ranking post-processed the same list, they performed equally well. For our prototype, the total recall rate is used because the product with small number of images can still have a higher total recall rate than those of the similar product with larger number of images.

For the value k in the top-k ranking, if the value k is too big, we loss processing time. If the value k is small then we may not retrieve the images of the searched product. We set k empirically by using the queries in the previous experiment for this experiment. We increase the size of k by one at a time starting from 1, until all the product images were identified correctly. The k value found during the experiment is 15.

We want to know how many images of the same product are needed for UCap to correctly identify the searched product. We perform the following experiment. We select 10 different products. We ask 3 sighted participants to take image of the products, and upload them along with the annotated information to our system. For each product, the participants have to upload 5 different images, and each image must be taken at various angle and orientation. They take the photo in the same testing room. One additional image is taken to be the searched image. We performed leave-one-out cross validation on this data set. For each fold, one product is selected as newly inserted product. All images of 9 other products are first indexed. Then, we add 2, 4, 8, and 15 images of the searched product one at a time. For the 2, 4, 8, and 15 product's images in the database, we found that the average across 10-fold of the total number of product's image retrieved on both list are 3.9, 7.1, 12.8, and 17.5 images respectively. Because k is 15, there are 30 images in total from both lists. This experiment indicates that we need at least 15 images of each product in the database to guarantee at least half of the images retrieved are of the searched product. We also notice that the retrieved images of the queried item are of similar background, and lighting. As such, it is crucial that the database contains images taken in as many environment setting as possible.

V. User Testing

We test the application on two groups of users: 10 participants of the sighted users, and 15 participants of the visually impaired users. For the first group, the participants are the university students. For the second group, the participants are from Ratchasuda College, Dialog in the dark exhibition, and the association of the blind. The evaluation is conducted via two surveys, observations during testing, and the interview. The first survey is given before application testing. This survey collects personal information and shopping behavior. After testing, the second survey and the interview are conducted in order to collect feedback and suggestion. From the first survey, all participants have smartphones. The visually impaired users have used at least one mobile application for the blind before, and often shopped for products at the supermarket. Less than 35% pay attention to the nutrient and ingredient information. For all the sighted users, they have never used applications for the blind before. The survey questions are related to the

TABLE I: Survey Results from the Visually Impaired Users

Questions	Ave. Score on the 1 - 5 scales
Application is easy to use and access	4.33
Show Enough Nutrient Information	3.8
Show Enough Ingredient Information	3.8
Product Identification Accuracy	1.93

TABLE II: Survey Results from the Sighted Users

Questions	Ave. Score on the 1 - 5 scales
Application is easy to use and access	4.8
Amount of Information Enter	4.2
Look and Feel of Interface Design	4.6
Respond Time	4.8
Product Identification Accuracy	2.4

Fig. 5: Sample images taken by the visually impaired testers.

application usages. The participants are asked to rate the application from 1 to 5 scales, where 5 is the very high satisfactory level, 3 is the satisfactory level (acceptable), and 1 implies very low satisfactory level. Both groups of users are asked to test the application on the same set of consumer products. The results are summarized in Tables I and II.

From Table I, the visually impaired users generally have very high satisfaction level for UCap, especially the ease of using the search function. The intrigue fact is that before knowing about this application, only 33.33% of the participants use nutrients or ingredients to help their decision making during shopping. After the testing, 80% of them are interested in learning more about the nutrient and ingredient information before buying a consumer product. As for the below satisfactory level in product identification accuracy, the reasons are two folds. First, the images taken are off focus such that only partial parts of the product are taken. Second, the images have different background and lighting than those in the database. The differences cause the ACC and JCD values to be different from the image of the same product taken at the supermarket, resulting in lower accuracy. Figure 5 shows a few images taken by the visually impaired testers. Note that the satisfactory level in accuracy of the sighted testers is higher than that of the visually impaired testers. This is because the sighted testers take better qualities of images.

VI. Conclusion and Future Work

This paper has proposed UCap, a crowdsourcing application for the visually impaired persons on android smartphone. UCap is a camera-based application that identifies a consumer product of a given image. The visually impaired users can take the photo using the smartphone camera and ask UCap to identify it. We have solved the consumer product image identification as the CBIR task. The CBIR system has been implemented using LIRE. As for the consumer product image database, we have used the crowdsourcing concept to take their pictures, add nutrient and ingredient information, and upload them to our system. We have created the seed database by annotating $3,950$ images of 296 products. From the user testing, both the visually impaired and sighted users have found that the application is easy to use, and has the sufficient amount of nutrient and ingredient information. Both groups have rated the accuracy of the consumer product identification task at low satisfactory level. However, our experiment and observation have indicated that UCap will have a higher accuracy once more annotated images in various environments are uploaded.

For future work, we will add the gamification concept to attract the sighted users to provide and review the product's information. We plan to add additional seed images from the other stores to increase variety in background and lighting images. Finally, we are looking for a lighting-independent feature to reduce the miss identification.

References

[1] Be My Eyes, "Be My Eyes Lend your eyes to the blind", Online available at http://www.bemyeyes.org.
[2] Erin Brandy, Meredith R. Morris, Yu Zhong, Samuel White, and Jeffrey P. Bigham, "Visual Challenges in the Everyday Lives of Blind People", In Proceeding of the CHI 2013, Apr. 27–May 2, 2013, Paris, France.
[3] Dialogue Social Enterprise GmbH, "Dialogue in the Dark Exhibition", Online available at http://www.dialogue-in-the-dark.com.
[4] Robert C. Fugina, "Internet UPC Database", Online available at http://www.upcdatabase.com
[5] Google, "eyes-free Speech Enabled Eyes-Free Android Application", Online available at https://code.google.com/p/eyes-free/
[6] Isabelle Guyon, and Andre Elisseeff, "An Introduction to Variable and Feature Selection", Machine Learning Research, vol 3, pp. 1157–1182, 2003.
[7] Bill Holton, "A Review of the TapTapSee, CamFind, and Talking Goggles Object Identification Apps for the iPhone", AFB AccessWorld Magazine, vol 14:7, July 2013.
[8] Bill Holton, "A Review of the Be My Eyes Remote Sighted Helper App for Apple iOS", AFB AccessWorld Magazine, vol 16:2, Feb. 2012.
[9] Image Searcher, "TapTapSee - Blind and Visually Impaired Camera", Online available at http://www.taptapseeapp.com
[10] Mathias Lux, and Oge Marques, "LIRE: Lucene Image Retrieval", Online available at http://www.lire-project.net
[11] NantWorks LLC, "LookTel - Instant Recognition Apps for Persons with Low Vision or Blindness", Online available at http://www.looktel.com
[12] Jon Noronha, Eric Hysen, Haoqi Zhang, and Krzysztof Z. Gajos, "PlateMate: Crowdsourcing Nutrition Analysis from Food Photographs", In Proceeding of the 24th ACM UIST Symposium 2011, Oct. 16–19, 2011, Santa Barbara, CA, USA.
[13] Bryan C. Russell, Antonio Torralba, Kevin P. Murphy, and Willian T. Freeman, "LabelMe: A Database and Web-Based Tool for Image Annotation", Int. Journal of Computer Vision, vol 77:1-3, pp. 157–173, May 2008.
[14] Sirion Vittayakorn, and James Hays "Quality Assessment for Crowdsourced Object Annotations", In Proceeding of the 22nd British Machine Vision Conference, pp. 109.1–109.11, BMVA Press, Sept. 2011.
[15] World Health Organization, "Visual impairment and blindness", Online available at http://www.who.int/mediacentre/factsheets/fs282/en/
[16] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung, "A Survey of Crowdsourcing Systems", In Proceeding of 2011 IEEE International Conference on Social Computing, Oct. 9–11, 2011, Boston, USA.