Medsinfo: An Android Application For Recognizing JFDA Approved Medication And Acquiring The Corresponding Arabic Leaflet

Rawan Ghnemat, Amer Mansour, Jaafar Abbadi, Rola Madain, Alia Madain

Abstract: This paper proposes a mobile application that targets the local Arabic speaking audience who use the local pharmacies in Jordan. It is aimed toward providing the ability for this specific audience to acquire the Arabic leaflets of the medication approved by Jordan food and drug administration. The application uses optical character recognition to recognize the enquired medication and provides a user-friendly interface to interact with the user and display required information in Arabic. We provide a small-scale experiment over 20 medication boxes. Every step in the process of identifying the medication is tested separately. The experimental results show that the application was able to identify the majority of medication boxes tested.

Index Terms: Mobile applications, Medication brand names, Non-English leaflet, Optical character recognition, OpenCV, Tesseract

1 INTRODUCTION

Optical character recognition (OCR) is used to convert scanned documents or digitally captured images of text into editable text. It is used as a type of data entry from printed paper data records, invoices, bank statements, business cards, passport papers, or any paperwork, it is a popular technique of digitizing printed texts to be edited, searched, stored and used electronically in applications such as machine translation, text-to-speech, and text mining. The proposed medication information application (MedsInfo) tests the use of OCR on the pharmaceutical box image to get the medication brand name and search for its corresponding information in Arabic. This is especially beneficial when it is troublesome or infeasible to insert the medication brand name manually or translate the medication information using translation software. A medication leaflet is expected to include certain details based on local medication labelling polices. It is supposed to give accurate information about the safety of the medication and how best to take it. For example, a typical medication leaflet will be organized as follows: First, a section to describe what the medication is (what the medication is intended for and which ingredients the medication contains) — secondly, an instructions section (how to use the medication). Thirdly, warnings section (when you should not use the medication and when caution is advised) and finally, a side effects section. In addition to health professionals, the targeted audience of leaflets comprises the general public and patients. Therefore, leaflets are supposed to be written in clear, easy-to-understand language. Our work proposes, MedsInfo, an application that will extract the brand name of the medication in an editable text format and compare it to a database of medications approved by Jordan food and drug administration (JFDA). Then MedsInfo returns the required leaflet in Arabic for the user to read. In addition to the need for a comprehensive database, there is a need to identify the medication first. We made a pilot study (small scale of 20 boxes) to evaluate the feasibility of identifying a certain medication by extracting its brand name from an image of its pharmaceutical box. The experimental results show that most cases chosen randomly are simple enough to be recognized correctly. In very few cases, the medication name is hard to recognize from the pharmaceutical box. Those cases happen when the background design and color intersect with the medication brand name. MedsInfo allows for manual input where the user can type the medication brand name. This paper is organized as follows: Section 2 gives a summary of related work regarding pharmaceutical applications and similar artificial intelligence problems; Section 3 discusses the implementation options. Section 4 discusses the first part of the small-scale study, where the application reads the medication brand name from an image. Section 5 shows two steps of searching for the medication leaflet and the case where no record is found of the specific medication brand name. Section 6 provides the results and a discussion of solved and unsolved problems, and finally, section 7 concludes the work done and provides direction of possible future work.

2 RELATED WORK

In this paper we introduce MedsInfo, a mobile pharmaceutical application that targets the patients as well as health care professionals. MedsInfo recognizes medication from its pharmaceutical box image and searches for its corresponding leaflet in Arabic. In case MedsInfo was not able to process the pharmaceutical box image properly, it is possible for the user to type in the medication brand name. Patients do search for information in leaflets, in a recent study undertaken to assess the people reading of medication leaflets in Malaysia, it is shown that from a total of 888 participants, 86.9% reported that they read their medication's label for the directions of usage and 84.3% for the dosage instruction. Nevertheless, it may be hard for patients to find relevant information, and this is especially true when there is a language barrier. In a recent study, the authors found that none of the over-the-counter medications manufactured by the Nepalese pharmaceutical industries adhered to the requirements for writing certain information in the Nepali language. We assume that it is not always feasible for the users to type in the medication name manually, and we made a small-scale test on the possibility of

- *Rawan Ghnemat: holds a PhD in Computer Science, with distinction, from Le Havre University, France. Her research interests include Artificial Intelligence, Swarm Intelligence, Data Mining, Urban Dynamics Modeling and Multi-Agent Systems (MAS). She works as an associate professor at Princess Sumaya University for Technology, Amman, Jordan. She has several scientific publications and funded research projects. (e-mail:* r.ghnemat@psut.edu.jo)

recognizing a medication brand name from the pharmaceutical box image. The results are encouraging, as the majority of the pharmaceutical boxes used in the study where readable by the application. The work on recognizing a word on a pharmaceutical box is similar to an extent to a security mechanism that requires human users to recognizing a word from an image to make sure the user is not a machine. This mechanism is mainly used to secure websites from malicious bots, and it is called a Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA). CAPTCHA is a hard problem in artificial intelligence [3]. Text-based CAPTCHAs are the most widely deployed [4] [5]. Data reveals that CAPTCHAs are often difficult for humans, too [6] [7]. It is different from reading a medication name in that the design of medical boxes does not target challenging OCR programs. The medication name is meant to be clear on the box for many reasons, such as clarity for patients and marketing. Nevertheless, there is no standard design of pharmaceutical boxes, which means that there are challenging cases that are hard for an OCR to read, just like the case of a CAPTCHA. This is especially true when the design of the box intersects with the medication name.

3 IMPLEMENTATION DETAILS OF MEDSINFO

The mobile application is meant to be available in different operating systems, in our experiments and initial tests, we choose Android as the platform, as it is the most commonly used. Based on Gartner report of the fourth quarter of 2017, smartphone operating system share of Android is at 85.9%, iOS is at 14%, and all other operating systems together have 0.1% share of the market [8]. Android applications, in general, are small in size, an OCR application usually requires a considerable amount of space, and thus most storage and processing will be moved to the server-side to reduce the size of the application to a minimum. Moreover, the Android program would also provide for the user the ability to enter the medication brand name manually, not using the OCR provided. As the images sent or inquiries sent may be linked to a certain patient, the application should deploy some security and privacy measures that meet its security requirement [9], such as giving the users control over their information and ability to set privacy settings, additionally, the information collected for statistical reasons (if any) should be anonymized [10].

4 READING THE MEDICATION NAME FROM AN IMAGE

We have a few assumptions regarding the content of the input image. It is assumed that the face of the pharmaceutical box is either a square or a rectangle. In addition, the medication brand name is the biggest word in size on the pharmaceutical box. MedsInfo can read images from the camera or the images saved in memory. The users can take the image of the pharmaceutical box alone or with some background. It is assumed that the pharmaceutical box is the biggest shape with 4 vertices in the image sent by the user for processing. We use an open-source OCR software (Tesseract), but text detection solely based on Tesseract is restricted to various parameters of image and thus pre-processing is necessary [11]. For example, in case of some complex images, Tesseract gives higher accuracy if the input image is gray scale compared to color images [12]. Subsections from 4.1 to 4.4 are considered pre-processing in our case which comprises cropping, rotating, resizing, filtering and converting the image to binary.

4.1 Crop Pharmaceutical box when Necessary

If there is additional background information, the application separates the box from the background by cropping the image, otherwise this step is skipped. Figure 1a gives an example of an input image with background. As you may notice from the figure, the background has another object with edges, as shown in the upper left corner. Figure 1b shows the output of the cropping function for the same input image.

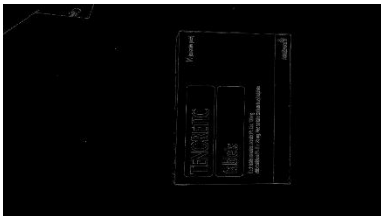
(a) Input image with background

(b) Cropped image

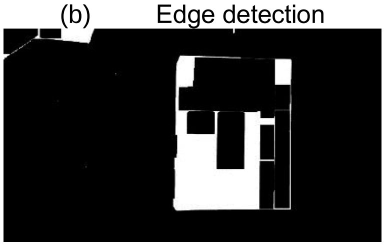
Fig. 1*. Input and output of the cropping function.*

This functionality is implemented by deploying a few functions from the OpenCV library. The process is shown in figure 2, after reading the input image, it is converted to a grey-scale image using cvtColor method. Afterwards, the image is blurred slightly using the GaussianBlur filter to reduce noise, as shown in figure 2a. The application detects the edges of every shape in the images regardless of the area and the shape type (triangle, rectangle, square, and circle), as shown in figure 2b. Then a closing morphological operation is applied to close the gaps in the outlines and make them clean and complete (see figure 2c).

(a) GaussianBlur filter



(b) Edge detection



(c) Closing morphological operation

***Fig. 2**. Process of cropping the input image*

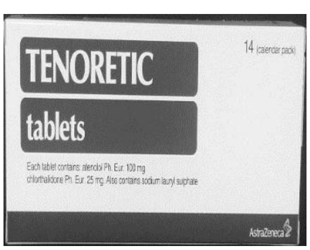
Finally, the outline is detected using functions that find contours, which results in the image shown in figure 1b. The application loops through every contour and compute its perimeter and then approximates it. The approximation of the contour is necessary because of the possible noise during image acquisition and/or shadows in the image which leads to incomplete shapes. Every approximated contour is validated to check if it has four vertices, then the rectangle (or square) with the maximum area is chosen.

4.2 Rotate and Resize Image when Necessary

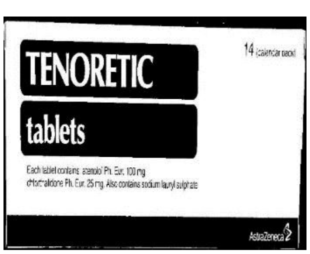
Using Python Image Library (PIL) and PyTessBaseApi class from Tesserocr module, the application checks if the image is in the correct orientation and rotates it if required. In addition, all images sent to the OCR are of the same size; in the experiments the size assumed is (1000x800) pixels.

4.3 Convert to a Binary Image

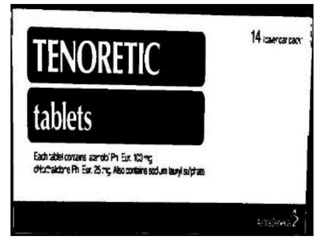
In our small-scale experiment, one threshold is used to convert the grey image to a binary image, where every pixel less than the threshold value is considered black, and each pixel equal to or greater than the value of the threshold is considered white, as shown in figure 3b.



(a) Gray scale image



(b) Binary image



(c) Filtered image

***Fig. 3.** Binarization and filtering*

There are many methods of calculating the threshold, the work in [13] categorize these methods into the following categories: histogram shape-based methods, clustering-based methods, entropy-based methods, object attribute-based methods, spatial methods, and local methods. Although the determination of a single global threshold for the image is often unsatisfactory [14], the threshold we used in our pilot study was sufficient. This may be due to the content of the pharmaceutical box is usually limited, and the information we are looking for in the box image is usually clear.

4.4 Image Filtering

Based on the data set used, all images were filtered effectively when a median filter was used at first. This filter will take the window size (WS) as an input parameter to create a window of size WS×WS, then it will move on all pixels, in every window it will take the median pixel value and apply it to all the pixels in that window. As the image is binary, the median value can be 0 or 255. After applying the median filter, there is more filtering required. Either the minimum or the maximum filter is used. Similar to the process used in applying the median filter, the minimum and the maximum filters will take WS as a parameter to create a window of size WS×WS and both will move on all pixels. The minimum filter will take the minimum pixel value in every window and apply it to all the pixels in that window whereas the maximum filter will take the maximum pixel value in every window and apply it to all the pixels in that window. As we can see from figure 3c, we have almost a clean image with a little noise, but if you look at the medication's name, there is no noise around it. In the next step, we will try to detect and crop the medication's brand name.

4.5 Return the Medication Name

The application crops the biggest word, as shown in Figure 4. The cropped image is sent to the OCR engine (Tesseract) to do the segmentation, normalization, and recognition phases. In [15], an overview of the Tesseract OCR engine was given. The OCR output is the text in the input image, and as we provided a pre-processed image to the OCR, only the medication brand name is in the input image.

***Fig. 4**. Final pre-processed image*

The results always require the user confirmation. If the returned name was wrong the user could enter the name manually. When the server receives the confirmation; which is a request to get the medication leaflet in Arabic, a query is sent to the database to get that leaflet; when the database returns the results, the server sends the response to the user at the frontend.

5 SEARCHING FOR THE LEAFLET

MedsInfo searches for the leaflet using the Soundex. The application generates the Soundex for the medication name in question. Soundex is a code (hash) for the medication built on its spelling/pronunciation; it is then compared to the soundecies stored in the database. If this query returned a result, it would be forwarded to the user (frontend). An example of the leaflet returned is shown in figure 5. If finding the leaflet is not straightforward, the LEVENSHTEIN function is used; it's a stored procedure to compute the words distance, every different character will add 1 to the result, if the difference is within two characters it is returned. The LEVENSHTEIN function is applied to the Soundex, this is because a Soundex is smaller than the word itself, and the LEVENSHTEIN function is slow. If the found leaflet is for a medication with a slightly different name the user will be notified and warned as the system will assume there might be an input error and approximations were made. Medication errors commonly involve drug names that look or sound alike [16]. This is why it is required that the user confirm that the system recognized the name correctly. If the system does not find the leaflet, the system searches for a leaflet of an equivalent medication. Equivalent medication has the same INN, which is also known as the chemical or generic name.

***Fig. 5**. Arabic leaflet returned*

6 RESULTS AND DISCUSSION

Our small-scale study is conducted in order to evaluate feasibility of the application as it is not clear if the application can reliably return the right and accurate information to the user in two cases, first, when the user depends solely on the application to recognize the medication name from an image, secondly, when no Arabic leaflet is available. It is common for the pharmaceutical boxes to have the brand name clear for marketing reasons; this is the reason why we assumed to be able to recognize the name by an OCR after certain pre-processing. If the application is used, it would require a confirmation from the user for reliability. In case the exact medication leaflet is not found, the application searches for an equivalent medication leaflet and notifies the user of the difference such as a medication with the same International Nonproprietary Name (INN). We tested every function (functions/steps shown before) separately, then we integrated the functions (steps) and tested the system. There are two main challenges of detecting and cropping the pharmaceutical box, the functionality depends on the difference between the color of the background and the color of the text, plus it depends on the way the image was taken, especially the angle. Rotation errors may happen because of the pharmaceutical box design. Also, different languages, characters and logos may affect the orientation parameters. Converting the image to binary using one threshold is sufficient for our small-scale study purposes, the threshold was correct for all images, except one where the medication name has two parts, one of them is very similar to the box background color. The filtering approach of using two filters works for the random data set used, but the filtering phase can be further enhanced. Currently, detecting and cropping the medication name depends on the assumption that the medication brand name has the biggest font size. Finally, a factor that effects the output is the design of the box, some pharmaceutical boxes design intersects or covers the brand name text. In subsection 6.1, we will list some problems we faced in the testing phase, and we will discuss how they were solved. In subsection 6.2, we discuss the problem that was not solved by the current implementation of MedsInfo.

6.1 Solved Problems

1-Background DesignFigure 6a shows that there are three circles crossing the medication name. The method used to calculate the threshold and convert the image to a binary one helped in partial removal of the three circles, but it will leave some black dots. In the image filtering step, the three circles disappeared, as we can see in figure 6b.

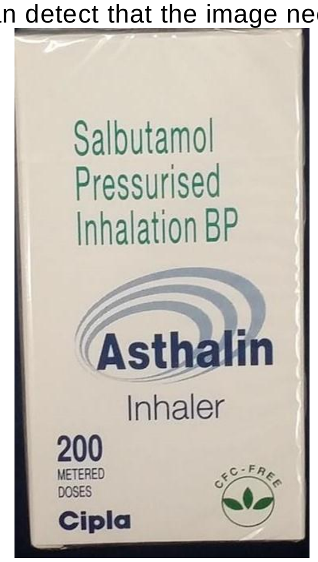
2-Colors

An example of this case is shown in figure 7a. We can notice from the figure that the medication brand name (SINGULAIR) has two colors and the difference between the colors is significant, so when we compute the threshold and convert the image to a binary image, the second part of the name will disappear, because the second part of the word has a color similar of that of the pharmaceutical box. This problem is solved using the Soundex function and LEVENSHTEIN function when searching for the medication in the database.

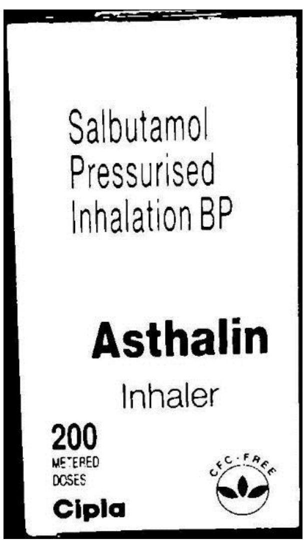
3-Orientation

Figure 8a shows a case where the pharmaceutical box orientation is tricky. This box seems to be in its correct orientation, but when the server checks if it needs to be rotated the answer will be true as the bottom half of the pharmaceutical box contains text with a different orientation.

Now we can notice that the pharmaceutical box part that contains the medication name is not in the correct orientation. So, after the image filtering step, and before sending the image to the OCR engine, the application checks again if the image needs to be rotated, because at that step the text on the right half will be gone, as shown in figure 8b. The system can detect that the image needs to be rotated.



(a) Input image after cropping



(b) After filtering

Fig. 6. *Background problem case*.

6.2 Challenging Problems

Challenging cases in our application are similar to the CAPTCHA concept discussed earlier. In some pharmaceutical boxes, the background intersects with the medication brand name in a way that it is hard to separate the letters from the background. This case is shown in figure 9.

7 Conclusions

It is important to be able to identify medication automatically and reliably in order to lower medication usage errors in different applications, we presented a mobile pharmaceutical application for patients and health professionals and tested the use of optical character recognition in recognizing medication. We applied the android application to 20 images of pharmaceutical boxes and the application was able to recognize the majority of medication brand names successfully. Future work includes enhancing the database and achieving better accuracy of recognizing medication automatically by considering multiple identifiers. Also, finding ways of medications identification over a larger area and a larger domain where other countries and other types of packaging will be considered.

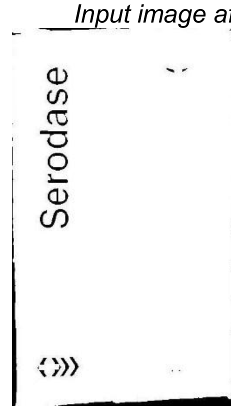
(a) Input image after cropping

(b) After converting to binary

Fig. 7. *Colors problem case*

.

(a) Input image after cropping



(b) After rotation and filtering

Fig. 8. *Orientation problem case.*

Fig. 9. *Buscopan pharmaceutical box.*

REFERENCES

[1] O. T. Dawood, M. A. Hassali, F. Saleem and I. R. Ibrahim, "Assessment of self-reporting reading of medicine's labels and the resources of information about medicines in general public in Malaysia," Pharmacology Research \& Perspectives, vol. 6, p. e00387, 2018.

[2] R. S. Poudel, S. Shrestha, S. Thapa, B. K. Poudel and M. Chhetri, "Assessment of primary labeling of medicines manufactured by Nepalese pharmaceutical industries," Journal of Pharmaceutical Policy and Practice, vol. 11, p. 13, 07 6 2018.

[3] L. Ahn, M. Blum, N. J. Hopper and J. Langford, "CAPTCHA: Using Hard AI Problems for Security," in Advances in Cryptology --- EUROCRYPT 2003, Berlin, 2003.

[4] J. Yan and A. S. El Ahmad, "A Low-cost Attack on a Microsoft Captcha," in Proceedings of the 15th ACM Conference on Computer and Communications Security, New York, NY, USA, 2008.

[5] H. Gao, M. Tang, Y. Liu, P. Zhang and X. Liu, "Research on the Security of Microsoft's Two-Layer Captcha," IEEE Transactions on Information Forensics and Security, vol. 12, pp. 1671-1685, 7 2017.

[6] E. Bursztein, S. Bethard, C. Fabry, J. C. Mitchell and D. Jurafsky, "How Good Are Humans at Solving CAPTCHAs? A Large Scale Evaluation," in 2010 IEEE Symposium on Security and Privacy, 2010.

[7] C. A. Fidas, A. G. Voyiatzis and N. M. Avouris, "On the Necessity of User-friendly CAPTCHA," in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 2011.

[8] "Gartner Says Worldwide Sales of Smartphones Recorded First Ever Decline During the Fourth Quarter of 2017," 2 2018. [Online]. Available: https://www.gartner.com/newsroom/id/3859963.

[9] A. L. Abu Dalhoum, A. Madain and H. Hiary, "Digital image scrambling based on elementary cellular automata," Multimedia Tools and Applications, vol. 75, no. 24, p. 17019–17034, 2016.

[10] J. Alqatawna, A. Madain, A. M. Al-Zoubi and R. Al-Sayyed, "Online Social Networks Security: Threats, Attacks, and Future Directions," in Social Media Shaping e-Publishing and Academia, N. Taha, R. Al-Sayyed, J. Alqatawna and A. Rodan, Eds., Cham, : Springer International Publishing, 2017, pp. 121-132.

[11] R. R. Palekar, S. U. Parab, D. P. Parikh and V. N. Kamble, "Real time license plate detection using openCV and tesseract," in 2017 International Conference on Communication and Signal Processing (ICCSP), 2017.

[12] C. Patel, A. Patel and D. Patel, "Optical Character Recognition by Open Source OCR Tool Tesseract: A Case Study," International Journal of Computer Applications, vol. 55, pp. 50-56, 2012.

[13] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," Journal of Electronic Imaging, vol. 13, pp. 13-20, 2004.

[14] J. Kittler, J. Illingworth and J. Föglein, "Threshold selection based on a simple image statistic," Computer Vision, Graphics, and Image Processing, vol. 30, pp. 125-147, 1985.

[15] R. Smith, "An Overview of the Tesseract OCR Engine," in Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), 2007.

[16] R. Filik, K. Purdy, A. Gale and D. Gerrett, "Labeling of Medicines and Patient Safety: Evaluating Methods of Reducing Drug Name Confusion," Human Factors, vol. 48, pp. 39-47, 2006.