Arabic Braille Recognition and Transcription into Text and Voice

Saad D. Al-Shamma and Sami Fathi

Abstract— **This paper presents a system for a design and implementation of Optical Arabic Braille Recognition(OBR) with voice and text conversion . the implemented algorithm based on a comparison of Braille dot position extraction in each cell with the database generated for each Braille cell . Many digital image processing have been performed on the Braille scanned document like binary conversion ,edge detection ,holes filling and finally image filtering before dot extraction. The work in this paper also involved a unique decimal code generation for each Braille cell used as a base for word reconstruction with the corresponding voice and text conversion database . The implemented algorithm achieve expected result through letter and words recognition and transcription accuracy over 99% and average processing time around 32.6 sec per page . using matlab environmemt**

I. INTRODUCTION

The Braille system, devised in 1821 by Frenchman Louis Braille, is a method that is widely used by blind people to read and write. Each Braille character or "cell" is made up of six dot positions, arranged in a rectangle containing two columns of three dots each. A dot may be raised at any of the six positions to form sixtyfour combinations (including the combination in which no dots are raised). For reference purposes, a particular combination may be described by naming the positions where dots are raised, the positions being universally numbered 1 through 3 from top to bottom on the left, and 4 through 6 from top to bottom on the right. The Braille code system has become widely used by several communities because of its simplicity, comfortable for blinds to use it when read and write. Braille was applied or translated into several languages including Arabic language.
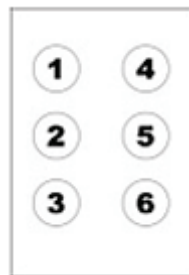

Fig. 1. Braille Cell

The dimensions of a Braille dot have been set according to the tactile resolution of the Fingertips of person. The horizontal and vertical distance between dots in a character, the distance between cells representing a word and the inter-line distance are also specified by the Library of Congress. Dot height is approximately 0.02 inches (0.5 mm); the horizontal and vertical spacing between dot centers within a Braille cell is approximately 0.1 inches (2.5 mm); the blank space between dots on adjacent cells is approximately 0.15 inches (3.75 mm) horizontally and 0.2 inches (5.0 mm) vertically. A standard Braille page is 11 inches by 11.5 inches and typically has a maximum of 40 to 43 Braille cells per line and 25 lines. Braille has been adapted to write many different languages including Arabic and is also used for musical and mathematical notation. Note that both Arabic and English Braille are read from left to right. For OBR, a Braille document is placed on a standard scanner and scanned by the OBR program, which either converts the document into text or saves it as a formatted Braille file. OBR works with single-side or double-sided Braille using a single scan. OBR offers many benefits to Braille users and those who work with them, facilitating communication, reducing storage space, and preserving out-of-print Braille texts. Everyone who works with blind people and does not read Braille will benefit from using the OBR. For example: parents, teachers, public organizations communicating with blind individuals, and computerized Braille libraries. All people in workplaces where Braille is used can read Braille easily by using an OBR. This paper looks at developing algorithms to recognize an image of embossed Arabic Braille

II. LITERATURE REVIEWS

An Optical Braille Recognition (OBR) system consists of several basic modules including: image acquisition, image processing, dot localization and segmentation, and finally dot recognition and conversion. Among issues that must be taken into consideration when implementing an OBR system are factors that negatively influence the identification process, such as lighting conditions, page placement in the scanner, and page movement [6]. Image acquisition is the first and most important step in any pattern recognition system. In OBR systems data is provided to the system in the form of images of Braille embossed pages. The process of acquiring these images digitally can be achieved by using a number of different equipments such as scanners or digital cameras, both of which have been used by developers and researchers [2-4, 7, 9-12]. Image preprocessing is an essential step during which errors that occurred while the images were taken are eliminated. Errors include noise, deformation, bad

illumination or blurring. Image preprocessing can be used for image enhancement by reducing noise, sharpening images, or rotating a skewed page. The algorithms used differ from one system to another depending on the classification approach followed by researchers and developers. An early effort was presented by Mennens et al. in their work [4]. The authors addressed the problem of false shadows in the image caused by the fact that Braille pages are never perfectly flat due to the tension in the paper's surface, by subtracting a locally averaged image from the original. The authors rotate skewed images by using the deviation over a vertical projection of the image. Hermida's et al. [3] system employed thresholding before the image is passed on to the Braille dot extraction module. Their algorithm converts a digital image of a scanned Braille page into one consisting mainly of black and white spots denoting the dots. The thresholds used were adaptively calculated from the histogram of the input image. In 1995, Hentzschel and Blenkhorn [2] presented a system for optical Braille recognition based on twin shadows approach, which subtracts two images of the same Braille page, where each image was taken under different illumination conditions. This helps eliminate blemish and noise in images caused by the texture of the paper used. Unlike Blenkhorn's work [1] presented in 1994 that solely discussed the classification process, his work in association with Hentzschel in the following year addressed the different modules of a pattern recognition system, including image processing. In [2], the image-processing module encompassed a variety of routines, each serving a different and crucial purpose as using random-noise reduction filter that was necessary to avoid undesired emphasis of noise. In [8], preprocessing consists of two sub operations: noise filtering and edge enhancement. Noise filtering is achieved via a low-pass special Gaussian filter. Edge detection is achieved using convolution Sobel kernels. The approach adopted by Mennens et al. in [5] for extracting Braille dots is based on several assumptions, one of which is that a single dot is represented by two gray level intensities, a light area right above a dark one. Also their system is designed to recognize a double-sided Braille page. This approach may produce false core regions if two dots are vertically neighbors. The localization and extraction algorithm developed by Hermida et al. [3] takes a thresholded image consisting of couples of white and black spots, where each couple denotes a single Braille dot. Though this method is easy and quick, it suffers from that: if some legitimate points are lost, false ones are produced. The dot localization and extraction technique used in [2] consists of two steps; the first is image registration and the second is character segmentation. The final step of dot extraction is normalization, where each Braille character is represented by a 2x3 matrix. Each bit in the matrix denotes a dot in Braille cell. Deciding whether a bit value is either 0 or 1 is based on a threshold used to test 1/6 of the cell area. In 2004, Wong et al. [7] proposed an OBR system that is capable of recognizing a single sided Braille page in addition to preserving the format of the original document in the produced text file. The algorithm processes the image one row at a

time reducing the computation time significantly. The dot detection module incorporated in the OBR system proposed by Oyama et al. [6] in 1997 is designed to detect both recto and verso Braille dots; that is detecting dots on both sides of the page. This was possible due to the difference in light reflectance between recto and verso dots. A hardware circuit configuration that corresponds to the equations and operates in a similar manner was given as well. Mennens et al. [4] adopted Binary Braille cell sets as basis for their classifier, which is grouping the dots and representing each dot by a bit position. The authors did not elaborate on the comparison method used to recognize a Braille cell as a certain letter or digit. The classifier presented by Hermida et al[3] takes as an input the image produced from the dot extraction module, where characters are represented as a group of dots, each dot is in turn represented by a single bit, with 1 or 0 values. Using this representation, the Braille-to-ASCII conversion is accomplished. The system proposed in [1] is based on finite state approach that operates with a finite number of states that perceives the correct state, in addition to its ability to perform both left and right context checking using matching algorithms. This feature is very important in determining characters proceeding wildcards. One of the greatest advantages of this system is that it is designed in a way to perform the conversion from Braille to any of the natural languages depending on the tables provided containing the conversion rules. The work in [2] did not elaborate on the techniques used for converting an extracted Braille cell into natural language characters. The recognition module in [7] is designed to work with thresholded images resulting from the halfcharacter detection module. The classification process is carried out using a probabilistic neural network. For interpretation purposes, authors in [8] determine centriod distances between each dot and its four possible neighbors. Dots are then grouped into cells. Based on the boundary coordinates, information and illumination characteristics, two standard templates were then constructed to represent the frontface dots and back-face dots. To the best of our knowledge, there is only one commercial OBR software. It is a Windows-based software that allows reading single and double-sided Braille documents with a standard scanner [13]. Unfortunately, it does not support the Arabic language and there is no explanation of the used techniques. In 2007 Abdul- Malik Al-Salman et al aimed to build fully functional Optical Arabic Braille Recognition system. It had two main tasks, first is to recognize printed Braille cells, and second is to convert them to regular text. It comprised the following stages. First, an image of a Braille document page is obtained using a flatbed scanner. Second, the image is converted to a gray color. Following that any white or black frames are cropped. The image is then thresholded so that only three classes of regions exist: dark, light and background. In case of having a tilted paper then de-skewing is applied using a binary search algorithm. Having labeled each of the different types of regions, an initial identification of Braille dots is performed. Finally, Braille cells are then recognized [14]. The system has been tested with a wide

variety of A4 scanned Braille documents, both single and double sided, written in the Arabic language and scanned with different scanners. Overall, on single sided and double-sided documents 99% of the dots are correctly recognized. The tests included different variations of Braille documents; skewed, reversed or worn-out [14].

III. METHODOLOGY

The work has been developed under MATLAB environment and divided into two stages: 1-Arabic Braille letters recognition and transcription 2-Arabic Braille words recognition and transcription Both sections are passing through several stages including image capturing or scanning, converting the Image to Gray Level, image Thresholding (converting it to binary), de-skewing the image, edging the Braille dots, filling Braille dots, opening the image by removing all micro scale objects, cropping the cell frames, cropping the dot frames from cell frames, generating binary equivalent of activated Braille dots, generating equivalent decimal Braille code, apply matching algorithm to Braille frames, get equivalent voice and text file of matched Braille cell
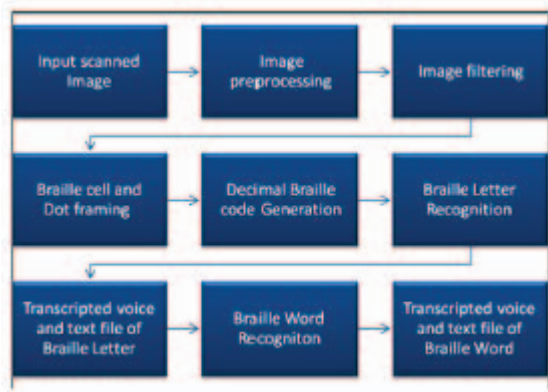

Fig. 2.   Block diagram of the implemented system

A. Braille document scanning

Braille document is a page that written using Braille alphabets. In this project the design based on standard specification of Arabic Braille writing, so to make sure that the Braille page which used in implementation will meet the standards, the research team contacted Sudanese rehabilitation center for blinds in Khartoum to meet some experts in Arabic Braille writing and to get some Braille documents written in Braille with standard specifications. The test image has scanned by using HP Scan jet djf2200 scanner with horizontal and vertical resolution 200 dpi, bit depth 24, dimension 1621X2248 and the image store in JPEG format

B. Image preprocessing

1) Gray scale and binary conversion: Gray scale and binary conversion using rgb2gray and im2bw matlab functions
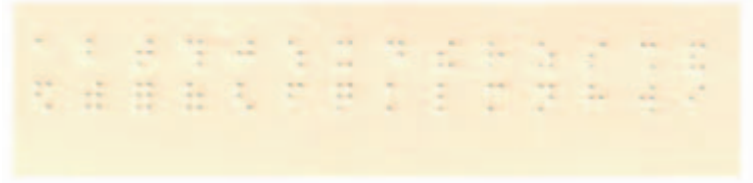

Fig. 3.   scanned test image

2) edge detection: canny method is used to finds edges by looking for local maxima of the gradient of input image. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds, to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This method is therefore less likely than the others to be fooled by noise, and more likely to detect true weak edges. The edged dots are not clearly identified, so there is another step implemented to increase the contrast of Braille dots. As mentioned above edged image contains edges of dots but does not include whole dots, so it must fill edged dots to clearly recognize Braille dots.

3) filling: filling dots function imfill Matlab function is used to fills holes in the binary image. A hole is a set of background pixels that cannot be reached by filling in the background from the edge of the image.

C. Image filtering

The filtering function removes from a binary image all connected components (objects) that have fewer than specified number of pixels, producing another binary image that filtered from micro objects. Experimentally found that most suitable pixel number in the image is 10 pixel, that's mean all objects have less or equal pixels, were removed as micro objects and it did not treaded as dots. By using matlab function bwareaopen

D. Braille Cells and Dots Framing

The framing process to determine the cells, words and lines in the scanned image based on standard dimensions of Braille documents as shown in fig. 4 , the cell frame size is known and experimentally recorded as 95X80 pixels in resolution of 200 dpi. Also the dot size recorded as 20X15 pixels in resolution of 200 dpi.

E. Decimal Braille Code Generation

This stage is a core stage of the system and it was done by testing each dot in Braille cell, if it active the position of this dot take digit one and if inactivated the position of this dot take digit zero. The recognizing process of active or inactive dots was depend on taking summation of dot frame, if the summation was been ones digits, that means this dot is activated, else that dot will be inactive.
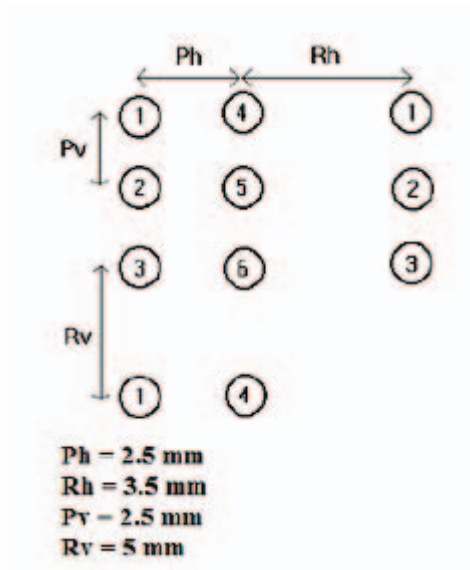As shown in Fig 5

Ph = 2.5 mm
Rh = 3.5 mm
Pv = 2.5 mm
Rv = 5 mm

Fig. 4.   Braille mesh.Standard distances

Dot position   : 6 5 4 3 2 1
Binary number : 0 0 1 1 0 1
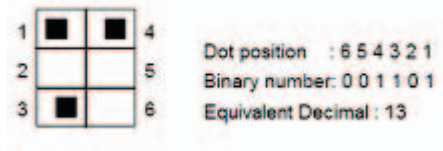Equivalent Decimal : 13

Fig. 5.   Decima; Braille code generation

F. Braille letter recognition and transcription

In this stage of the project the Braille letter was recognized using matching algorithm to match each of the input decimal Braille code from an input processed image with codes of each Arabic letter. After recognition process implemented the recognized letter transcript into equivalent text and voice file through addressing process to run matched equivalent files from stored addressed database.

G. Braille word recognition and transcription

This is the last stage in implementation, the word recognition process flow through letter recognition and fill Braille decimal array of word, to apply matching process with stored addressed text and voices files of words, then to run equivalent files from addressed word database

IV. RESULTS

The implemented method has been tested with a variety of scanned Braille documents written using standard Arabic Braille and it has accredited by Sudanese rehabilitation centre for blinds in Sudan. Documents were scanned using a commercially available HP Scan jet djf2200 scanner with 200 dpi resolution. The processing was performed on a PC with an Intel core 2 duo CPU, 2GB RAM, under MATLAB implementation environment. The CPU time taken for one spaced page to be processed is averaged at around 32.6s. As shown in fig 6 for one sample of Braille word recognition.
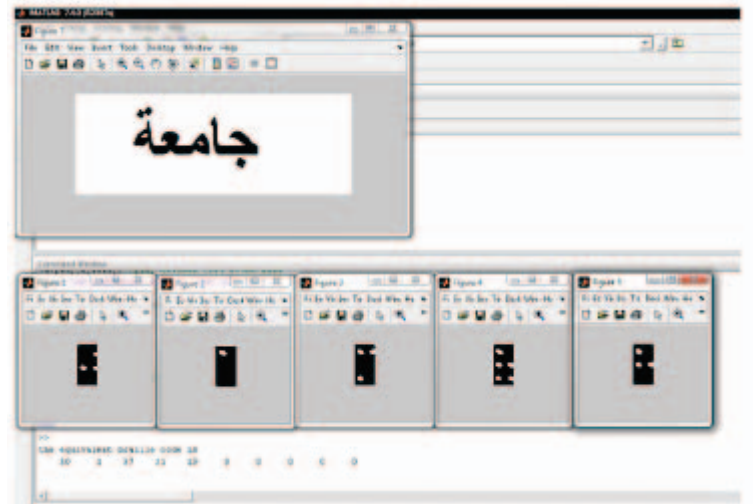

Fig. 6.   the recognized and transcript Braille word

V. CONCLUSION AND DISCUSSION

1) : The results obtained were promising during experiments performed on single sided computer embossed documents, with more than 99% accuracy achieved. As a whole, the approach described here shows the feasibility of a cost-effective, fast and easy method to detect dots in Braille documents. It does not require expensive or complicated hardware. It uses a flatbed scanner, which can be shared with other applications. Robustness to cope with low quality scans and defective documents is built-in at different levels, from the initial Threshold through to the flexible text and voice file generation.

2) : This project represents a major importance to the blinds through its focusing purpose on rehabilitation the blind persons, so there are some recommendations to help those who want to develop this work -To deepen the results obtained, more experiments and supports to apply this new method for implemented Arabic Braille recognition system is needed. - Enhancing system capabilities to be real time application. -Integrating advance systems and applications by using a bigger range of learning dataset.

3) : a more advanced image processing could be used for automatic process for all the manually setted parameters (conversion to binary image threshold , cell framing and line framing)

REFERENCES

[1] Blenkhorn, P.,"A System for Converting Braille into Print" *IEEE transactions on rehabilitation engineering,* Vol. 3, No. 2, June 1995.
[2] Hentzschel, T. W., and P. Blenkhorn, "An Optical Reading Systems for Embossed Braille Characters using a Twin Shadows Approach",*Journal of Microcomputer Applications*, pp. 341-345. 1995.
[3] Hermida, X. F., et al, "A Braille O.C.R. for Blind People", *Proceedings of ICSPAT-96.* Boston (U.S.A.). October, 1996.
[4] Mennens, J., et al,"Optical Recognition of Braille Writing", *IEEE, 1993.* pp. 428-431.
[5] Mennens, J., et al, "Optical Recognition of Braille Writing Using Standard Equipment",*IEEE transactions of rehabilitation engineering,* Vol. 2, No. 4, December 1994.

[6] Oyama, Y., T. Tajima, and H. Koga, "Character Recognition of Mixed Convex- Concave Braille Points and Legibility of Deteriorated Braille Points",*System and Computer in Japan,* Vol. 28, No. 2, 1997.

[7] Wong, L., W. Abdulla, and S. Hussmann, "A Software Algorithm Prototype for Optical Recognition of Embossed Braille", *the 17th conference of the International Conference in Pattern Recognition*, Cambridge, UK, 23- 26 August 2004.

[8] C. Ng and V. Lau, "Regular feature extraction for recognition of Braille",*ICCIMA'99,* 1999.

[9] Dias, "A portable device for optically recognizing Braille-Part II: Software development", *7th Australian & Neazlan Intelligent Information Systems Conf.*, pp. 18-21, Perth, W. Australia, Nov 2001.

[10] T. Gomez, et al,"AIR-Coupled Ultrasonic Scanner for Braille",*IEEE ultrasonic Symposium,* pp. 591-594, 2001.

[11] T. Yoshida, A. Ohya and S. Yuta, "Braille Block Detection for Autonomous Mobile Robot Navigation", *Proc. Of the 2000 IEEE/RSJ intl. Conf. on Intelligent Robots and Systems,* pp. 633-638, 2000.

[12] J. Dubus, et al,"Image Processing Techniques to Perform an Autonomous System to Translate Relief Braille into Black-Ink, Called: Lectobraille",*IEEE Engineering in Medicine & Biology Society 10th annual Inter. Conf.*, 1988.

[13] Optical Braille Recognition System, version 3.5, User Manual, October 2000.

[14] AbdulMalik Al-Salman et al,"An Arabic Optical Braille Recognition System",*ICTA'07,* Hammamet, Tunisia, April, 2007.