Braille Character Recognition Using Find Contour and Artificial Neural Network

Joko Subur
Department of Electrical Engineering
Hang Tuah University
Surabaya, Indonesia
joko.subur@hangtuah.ac.id

Tri Arief Sardjono
Department of Electrical Engineering
Institute of Technology Sepuluh Nopember
Surabaya, Indonesia
sardjono@elect-eng.its.ac.id

Ronny Mardiyanto
Department of Electrical Engineering
Instituteof Technology Sepuluh Nopember
Surabaya, Indonesia
ronny@elect-eng.its.ac.id

***Abstract*—Braille letters is characters designed for the blind, composed of six embossed points, arranged in a standard braille character. Braille letters is touched and read using fingers, therefore the sensitivity of the fingers is important. Those characters need to be memorized, so it is very difficult to be learned. The aim of this research is to create a braille characters 'recognition system and translate it to alpha-numeric text. Webcam camera is used to capture braille image from braille characters on the paper sheet. Cropping, grayscale, thresholding, erosion, and dilation techniques are used for image preprocessing. Then, find contour and artificial neural network method are used to recognize the braille characters. The system can recognize braille characters with 99% accuracy even when the braille image is tilted up to 1 degree.**

Keywords—Artificial neural network; braille characters; find contour; image processing; webcam

I. INTRODUCTION

Reading is one way to get information, but for those who are blind it will be difficult if to read the regular letters. Therefore, letters for the blind people was specially designed, named braille letters. Braille letters is composed of six points, which is three lines with two points. Six points can be arranged in such a way to create a variety of combinations. Usually, braille letters is read by touching the dot on the braille paper using fingers.

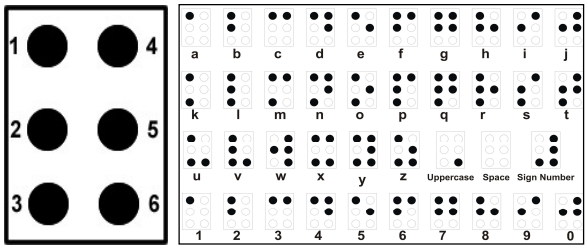

Fig. 1.Braille letters structure

To be able to read braille by feeling, a person needs to learn it. This is usually takes a long time, because the sensitivity of the hands should be trained. Beside that, the dots and the combinations that form a letter should be understood and memorized. Therefore, there are not many person able to read braille. It is necessary to research assistive technology to translatee braille into text to make it easier to read. Therefore, it will take a relatively shorter time compared to readit using finger.

Some researches to improve learning process forbraille reading in the computer system has been done using image processing[5]. Research to introduce braille image processing techniques has also been done. In the previous research, a scanner is used to take pictures of braille, then image processing is done and output is generated in text [6]. The scanner is also used as a tool to get braille pictures, as well as applying the method of neural network in the recognition process of braille[7]. The previous research also tried to take the picture with the camera phone. But the experiments were carried out on braille installed in public places, not on pieces of braille paper [2]. Find contour method is suitable to locate the dot in braille, but for recognition process needed one of method another [4].

Therefore, this research aimed to make a real time braille recognition system. The system can be real time because a webcam is used to get braille pictures and webcam is connected to a computer system. From pictures obtained braille, then do image processing, combine from find contour and artificial neural network method is used to recognize the letters. With so may make it easier for users to do the reading braille, without having to process through the scanner first and can increase the level accuracy of system.

II. METHOD

The steps for braille letters recognition used in this research is shown in Fig. 2, such as:

- Capturing braille letters image using webcam
- Image preprocessing
- Find coordinates x and y using find contour
- Segmentation area braille characters

- Recognize braille characters using ANN

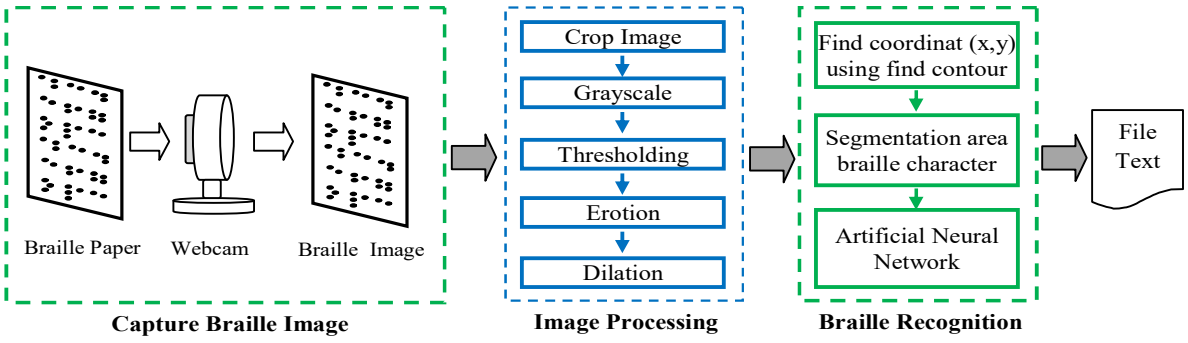

Fig. 2. System block diagram of braille letters recognition using find contour and artificial neural network method

A. Capturing Braille Image

To take braille images, braille paper is captured using a web camera with a resolution of 640 pixels by 480 pixels. The distance between the webcam and braille paper is ±15cm. Results from the capture process is generated as image files of *.jpeg type and has an RGB color scale. When the image capture braille, try lighting in the area of braille paper get enough light and constant. It will affect the quality of the image obtained .

B. Image Processing

This process is used to prepare the picture for the next process, to make it easier in the process of braille letters recognition.

B.1 Image Cropping

To get the required area only, the braille picture resulted from the capture process needs to cut, which eliminated the edge of the area that is not needed. The cutting process used cvSetImageROI command functions of OpenCV. The complete command is as follow:

cvSetImageROI(IplImage image,CvRectrect)*

The CvRectrect command determining the coordinates of the area to be cut. In this research, the coordinates of the cutting area is still determined manually.

B.2 Gray-scaling

Gray-scaling process is used to convert the original image that still has the RGB color scale into grayscale images. The next process is thresholding or change the grayscale image into a binary image as shown in Fig. 4. To do this,grayscale command functions of OpenCV is used. The complete command is as follow:

cvCvtColor(src ,bwsrc, CV_RGB2GRAY)

Where src is the original image that would be converted to grayscale, bwsrc is the image resulted from this process, CV_RGB2GRAY is the command from OpenCV.

B.3 Thresholding

Thresholding is the process to make the binary image. Binary image is an image that has only the color scale values of 0 and 255, or it could be regarded as black and white. In this process, adaptive thresholding is used. Adaptive thresholding is different with the usual thresholding in regards of the threshold value set. The usual threshold uses only one threshold value, while the adaptive threshold uses a range of value from minimum to maximum. If scanner image is the one being processed, then the general threshold can be used, but the webcam image has different intensity of the pixel values, therefore adaptive thresholding is used instead result of thresholding process as shown in Fig. 5. This thresholding process used functions of OpenCV command. The complete command is as follow:

cvAdaptiveThreshold(constCvArr src, CvArr* dst, double maxValue, int adaptive method=CV ADAPTIVE THRESH MEAN C, intthresholdType=CV THRESH BINARY, intblockSize, double param1)*

The maxValue is 255, adaptive method is implemented usingCV ADAPTIVE THRESH GAUSSIAN, CV THRESH BINARY is chose for threshold type, blocksize and param1 value is 57 and 5, respectively, but this value can be changed according to the processed image for a better result.

B.4 Erosion

Erosion is a process to change the pixel values inside to be equal with outside value. In this research, erosion process is used to minimize black area dot and remove noise from the braille image resulted from thresholding process. As shown in Fig. 5, there were black spots noise after the thresholding result. After the erosion process,the noise disappeared as shown in Fig. 6. In this process, CV_ERODE command of OpenCV is used. The complete command is as follow:

cvErode(constCvArr src, CvArr* dst, IplConvKernel* element=NULL, int iterations=1)*

Src is the image to be processed, dst is the resulted image, element value is NULL, and iterations is the

amount of looping in the process.

B.5 Dilation

Dilation process is the reverse of the process erosion, where the process is changing the pixel values outside equated with pixel values within. To easy understand, that the dilation process is process of adding value to a pixel towards the exit. In this research dilation process used to enlarge back black dot on image braille. As shown in Fig. 7 where the black dot dilation results back enlarged. To do this dilation process used functions of OpenCV command is CV_DILATE. This the complete command:

cvDilate(constCvArr src, CvArr* dst, IplConvKernel* element=NULL, int iterations=1)*

Src is image will be process, dst is result image from dilation process, element can be change NULL value and iterations is value how much looping this dilation process.

This is a image of the result image capture process until the dilation process .

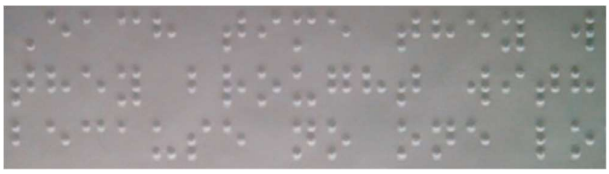

Fig. 3. Results capture braille image with a webcam

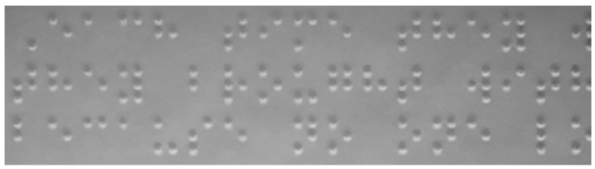

Fig. 4. Results grayscale process

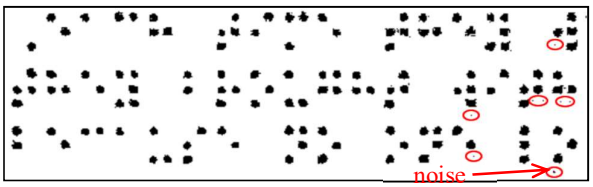

Fig. 5.Results thresholding process using adaptive threshold.

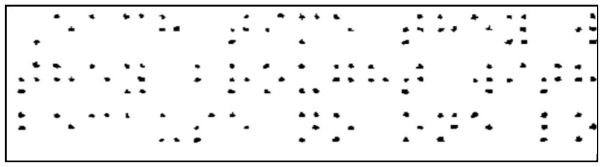

Fig. 6. Results erosion process (not visible noise)

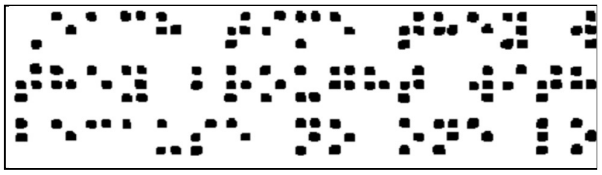

Fig. 7. Results dilation process (dot braille larger)

C.1 Find Center Coordinates Each Dot

Furthermore, the process find contour which aims to get the contour of black dots on the image dot braille, if the contour of the black dot for each point can be found then the position coordinates of each point of the black dot can be known and the number of points the black dot in the image can be calculated amount. To do this find contour process used functions of OpenCV command is cvFindContour. This the complete command:

cvFindContours(src image, dst image, &contours, sizeof(CvContour),CV_RETR_LIST)

In the process find this contour can be read also the number of members of each black dot braille. In one dot point there can be more than one member. The members of a collection of pixels that fall into the black dot area, and each pixel has coordinates x and y. This is example one of member's data from a black dot and value x and y coordinate:

(1). p->x: 185, p->y: 380
(2). p->x: 191, p->y: 382
(3). p->x: 189, p->y: 387
(4). p->x: 185, p->y: 386
(5). p->x: 183, p->y: 385
(6). p->x: 189, p->y: 380
(7). p->x: 191, p->y: 385
(8). p->x: 186, p->y: 387
(9). p->x: 184, p->y: 386
(10). p->x: 183, p->y: 382
Numbers of members: 10

From the above data it can be seen that the black dot is one point there were 10 members, or it can be assumed that there are 10 black pixels. Example of members of each black dot braille as shown in Fig. 8.

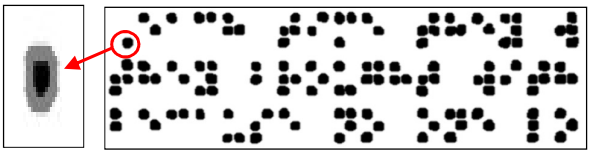

Fig. 8. Results find contour process

After the pixel coordinate data of each member of the black dot obtained, then the data is searched for the minimum value of p->x (minX), p->y (minY) and the maximum value of p->x (maxX), p->y (maxY). Of the minimum and maximum values can be determined the coordinates of the center point of the black dot. To find the value of the center point can be used Eq. (1) and Eq. (2) below:

$$\text{centerX} = \frac{\text{minX} + \text{maxX}}{2} \quad (1)$$

$$\text{centerY} = \frac{\text{minY} + \text{maxY}}{2} \quad (2)$$

Where centerX is center coordinate x each dot, minX is minimum value coordinate x, maxX is maximum value coordinate x, centerY is center coordinate y each dot, minY is minimum value coordinate y, maxY is maximum value coordinate y.

In Fig. 9 can be show the results of reading the center point coordinates of each black dot, the dot is marked in red color.

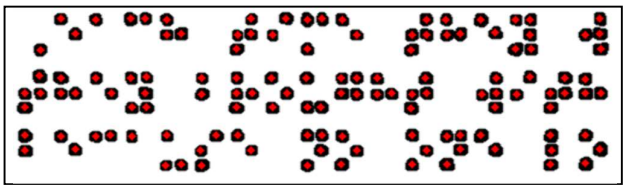

Fig. 9. Results of reading the center point coordinate each dot

After the center of the coordinate values of each point of the dot can be known, then determined the average middle in each row and column. Why is that, because in one row or in one column does not necessarily have the coordinates x and y are the same. So it is necessary to find the average value of the coordinate values. To search for the average value of x and y coordinates of each row and column used Eq. (3) and Eq. (4) below:

$$aX_k = \sum_{k=1}^{k}\sum_{n=1}^{n}(aX_k + centerX_n)/n \quad (3)$$

Where aX_k is average value x coordinates each column, center X_n is value centerX coordinate each column, k is column value, and n is much of dot each column.

From the calculation of Eq. (3) will be obtained average value of x-coordinate in each column. Furthermore, to obtain the average value of y-coordinates in each row is used the calculation of the Eq. (4). The results from this process can be observed in Fig. 10. In Fig. 10 can be show the results of the process of finding the average coordinates x and y in each row and column with a red line marked lines x and y coordinates in each row and column. With already known each coordinate it will facilitate the process of segmentation and recognition braille for next process.

$$aY_b = \sum_{b=1}^{b}\sum_{n=1}^{n}(aY_b + centerY_n)/n \quad (4)$$

Where $aY_b$is average value y-coordinates each row, $centerY_n$is value centerY coordinate each row, b is row value, and n is much of dot each row.

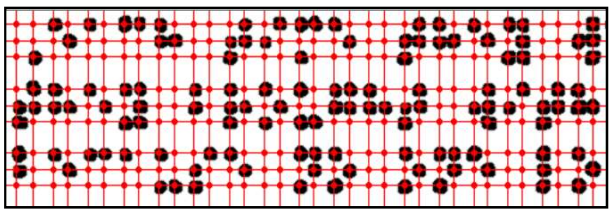

Fig. 10.Results of find average x-y coordinates each row and column

C.2 Segmentation Braille Each Character

Of the average coordinate data that have been obtained in each row and column are then used as reference segmentation process. The segmentation process designed to determine the area to read data for each of the braille course. Because of the wide range consists of 2 columns and 3 rows, then the determination area segmentation is based on the average value of x and y coordinates of each of the six points of the x and y coordinates. The calculations used to determine the area segmentation can be used Eq. (5) below:

$$ASeg_i = \sum_{b=1}^{b}\sum_{k=1}^{k}[i = i + 1]\left(\sum_{m=b}^{b+3}\sum_{n=k}^{k+2}([aX_n]\ [aY_m])\right) \quad (5)$$

Where ASeg is segmentation area each braille character, b is value of row, k is value of column, i counter of value of braille character, m is counter 3 row each braille character, n is counter 2 column each braille character, aX is average value x-coordinate, aY is average value y-coordinate.

Results of the segmentation process can be show in Fig. 11. Segmentation area of one braille character can already be marked with a red line. After segmentation process each area determined to do next letter recognition process braille character.

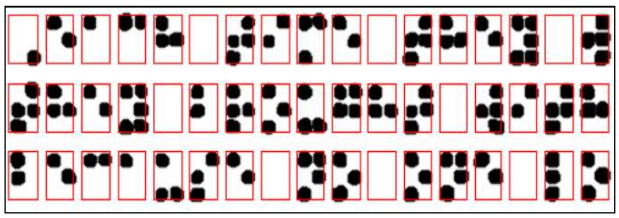

Fig. 11. Results segmentation process

Because it can be specified area of segmentation for each of the Braille, the segmentation is then performed again every area of the Braille. Segmentation is done by making a small segment as many as 40 areas of segmentation, which consists of 5 columns and 8 rows, as shown in Fig.12.

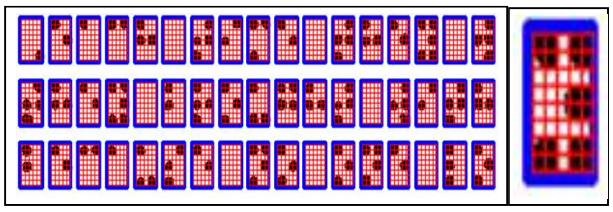

Fig. 12.Segmentation small each dot

TABLE 1. Output data process ANN

Output Data Process ANN Each Character															
No.	Char	Target Data ANN						No.	Char	Target Data ANN					
		Y5	Y4	Y3	Y2	Y1	Y0			Y5	Y4	Y3	Y2	Y1	Y0
0.	SPACE	0	0	0	0	0	0	17.	Q	0	1	0	0	0	1
1.	A	0	0	0	0	0	1	18.	R	0	1	0	0	1	0
2.	B	0	0	0	0	1	0	19.	S	0	1	0	0	1	1
3.	C	0	0	0	0	1	1	20.	T	0	1	0	1	0	0
4.	D	0	0	0	1	0	0	21.	U	0	1	0	1	0	1
5.	E	0	0	0	1	0	1	22.	V	0	1	0	1	1	0
6.	F	0	0	0	1	1	0	23.	W	0	1	0	1	1	1
7.	G	0	0	0	1	1	1	24.	X	0	1	1	0	0	0
8.	H	0	0	1	0	0	0	25.	Y	0	1	1	0	0	1
9.	I	0	0	1	0	0	1	26.	Z	0	1	1	0	1	0
10.	J	0	0	1	0	1	0	27.	Capital	0	1	1	0	1	1
11.	K	0	0	1	0	1	1	28.	Number	0	1	1	1	0	0
12.	L	0	0	1	1	0	0	29.	“ . ”	0	1	1	1	0	1
13.	M	0	0	1	1	0	1	30.	“ ? “	0	1	1	1	1	0
14.	N	0	0	1	1	1	0	31.	“ , ”	0	1	1	1	1	1
15.	O	0	0	1	1	1	1	32.	“ ! “	1	0	0	0	0	0
16.	P	0	1	0	0	0	0	33.	“ – “	1	0	0	0	0	1

From 40 area small segmentation will be read value of pixels. Where in this area is value betwen 0 (black) or 255 (white), each small area segmentation will be get 1 data input. So, result in this process will be 40 get data. This data will be as data input for artificial neural network process.

C.3 Artificial Neural Network

Artificial Neural Network (ANN) is a mathematical model in the form of a collection of units connected in parallel which resembles a neural network in the human brain. So can be used then ANN must be learning first. Learning process by providing the data input of the data patterns is included with the desired target output value.
In this research made five kinds of data patterns at each of the braille characters . Five kinds of patterns includes the data pattern with a tilt of: -1 degrees, -0.5 degrees, 0 degrees, 0.5 degrees and 1 degree. Each one data pattern consist of 40 data input.

In this ANN 40 data input used, 2 hidden layers and 5 neurons in ouput layer used. Data input get from value of 40 little segmentation areas. In hidden layer1 80 neurons used and 50 neurons used in hidden layer2. In the output layer 6 neurons used, because will purpose can be make combination binary 6 bits. Where binary 6 bits combination will be represent 31 sum of alphabets A to Z, space, sign number, dot, coma and question sign etc. Structure topology ANN can be shown in fig.13.

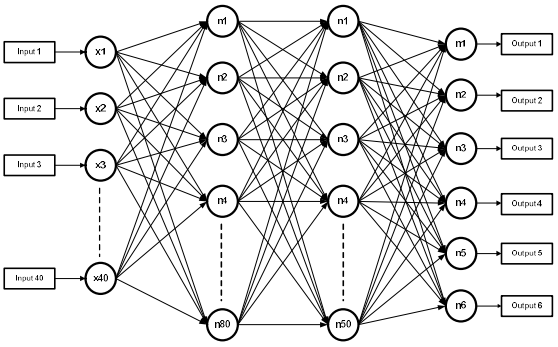

Fig. 13 Structure topology of ANN in used

III. Results

To determine the level of accuracy of the system in recognizing braille characters, it is tested by performing the introduction of braille on 10 different images data. The test is done with a variety of degree of picture tilting. The degree ranging from -1.5 degrees to 1.5 degrees. Test results are shown in Table 2 below.

TABLE 2.The data of experiment to recognize braille character

Degree of Tilting	All Data				
	Actually Character	Read Character	Mising Character	Error (%)	Accuracy (%)
-1,5	1229	9	1220	99,27 %	0,73 %
-1,25	1229	665	564	45,89 %	54,11 %
-1	1229	1220	9	0,74 %	99,26 %
-0,75	1229	1217	12	0,98 %	99,02 %
-0,5	1229	1225	4	0,33 %	99,67 %
-0,25	1229	1224	5	0,41 %	99,59 %
0º	1229	1226	3	0,25 %	99,75 %
0,25º	1229	1225	4	0,33 %	99,67 %
0,5º	1229	1225	4	0,33 %	99,67 %
0,75º	1229	1214	15	1,23 %	98,77 %
1º	1229	1221	8	0,66 %	99,34 %
1,25º	1229	731	498	40,53 %	59,47 %
1,5	1229	24	1205	98,05 %	1,95 %

IV. Conclusions

In this research, the braille recognition system using find contour and artificial neural network method is realized. From the experimental results, accuracy level of 99% can be achieved by this sytem on the tilted the image of -1 degrees to 1 degrees. The level of accuracy began to decrease when the image is tilted more than 1 degree, and the system is unable to recognize the image at all when the image is tilted at 1.5 degrees. Therefore, combine find contour and artificial neural network method in braille character recognition system, can increase the level of accuracy.

V. References

[1] Jie Li andXiaoguang Yan, “Optical Braille Character Recognition with Support-Vector Machine Classifier,” International Conference on Computer Application and System Modeling (ICCASM), 2010.

[2] Nambaand Zhang, ”Cellular Neural Network for Associative Memory and Its Application to Braille Image Recognition,” International Joint Conference on Neural Networks, BC, Canada, 2006.

[3] Shreekanth. T andUdayashankara. V, ”A Review on Software Algorithms for Optical Recognition of Embossed Braille Characters,” International Journal of Computer Applications (0975-8887), volume 81-No.3, 2013.

[4] Subur. J, Sardjono. T.A,Mardiyanto.R ”Braille Character Recognition Using Find Contour Method,” The 5th International Conference on Electrical Engineering and Informatics (ICEEI), 2015.

[5] Wajid. M, Abdullah, Farooq. M, "Imprinted Braille-Character Pattern Recognition using Image Processing Techniques," International Conference on Image Information Processing, 2011.

[6] Wong. L, Abdulla. W dan Hussmann.S, “A Software Algorithm Prototype for Optical Recognition of Embossed Braille,” IEEE-2004. In: 17th Conference of the International Conference in Pattern Recognition, Cambridge, UK, pp. 23–26.

[7] Zhang. S, and Yoshino. K, “A Braille Recognition System by the Mobile Phone with Embedded Camera,” 0-7695-2882-1/07, 2007, IEEE

[8] OpenCV Reference Manual, v2.2, Desember 2010