Implementation of Real-Time Static Hand Gesture Recognition Using Artificial Neural Network

Lita Yusnita[1], Rosalina[2], Rusdianto Roestam[3], and RB Wahyu[4]
[1]Faculty of Computing, [2]President University [3]Bekasi, Indonesia
Email: [1]rosalina@president.ac.id,

***Abstract*—This paper implements static hand gesture recognition in recognizing the alphabetical sign from "A" to "Z", number from "0" to "9", and additional punctuation mark such as "Period", "Question Mark", and "Space" in Sistem Isyarat Bahasa Indonesia (SIBI). Hand gestures are obtained by evaluating the contour representation from image segmentation of the glove wore by user. Then, it is classified using Artificial Neural Network (ANN) based on the training model previously built from 100 images for each gesture. The accuracy rate of hand gesture translation is calculated to be 90%. Moreover, speech translation recognizes NATO phonetic letter as the speech input for translation.**

***Index Terms*—Static Hand Gesture, Artificial Neural Network, Speech Translation, SIBI**

I. Introduction

HUMAN naturally uses gesture to communicate and the advancement in information technology field has contributed a major influence to the way people communicate with each other. Exchanging information has always been the basic core to the study. For the people with speech and hearing impaired, a hand gesture in sign language is the most natural way to communicate with each other. Without having the ability to speak and hear like most of the people, they can interact well among each other. In Indonesia, Sistem Isyarat Bahasa Indonesia (SIBI) is the sign language officially approved by the government and currently used in educational curriculum for children in school [1]. Some difficulties in sign language communication may arise when the parties involved do not understand SIBI at all. The most common solution to this problem is by having another person as a translator to bridge the communication between them. However, alternative solution has to be provided because a translator, unlike a computer program, may not be available at any given time.

Many researches have been done during the past two decades in the area of vision based hand gesture recognition using artificial neural network to recognize American Sign Language (ASL) [2–5].

Human computer interaction is the essential base to provide the alternatives. In this work, there are some methods to deal with such as computer vision, machine learning, and speech recognition. Computer vision focuses on acquiring image with the support of image processing and extracts the essential data of the image. After that, there will be a classification process that compares and classifies the current gestures that users perform according to the training model. This process is based on machine learning and classification method used in this project is ANN. Last but not least, speech recognition field handles the input speech in form of NATO phonetic language given by the user to be translated in the respective sign language.

There are two possible inputs in this work which is hand gesture and speech recognition. As the region of interest is based on the color of the glove, the limitation may arise from the compulsories of the user to wear a glove for using the program. The other constraint is the hardware used because the result of recognition in translation will be directly proportional to the quality of the imaging and audio device that the user has. The lighting of the room where it takes place and the distance between user and webcam also determine the accuracy of the translation.

Two out of twenty-six letters in alphabet of SIBI, which are the letter "J" and "Z", require motion gesture. Meanwhile, this work only capture the static gesture. Hence, the particular motion signs will be altered to static. Additional hand gesture is also introduced in this paper such as gesture for punctuation mark like period, question mark, and space.

Received: XXXXXXXXX. 2, 2017; received in revised form: XXXXXX; accepted: XXXXXXXXXXXXXXX; available online: XXXXXXXXXXXXXXXXXXXX.

II. Literature Review

A. Sign Language

Practically, gestures can be restricted into static and dynamic. Static gestures are described in terms of hand shapes, while as dynamic gestures are generally described according to hand movements [6].

Sign language is a language that requires the combination of hand gesture, orientation, movement of the hands, arms, body, and facial to simultaneously express the thoughts of the speaker. Sign language recognition is done in three different categories: 1) Glove based analysis, 2) Device based analysis, and 3) Vision based analysis. Different countries are having different sign language, which is used by hearing impaired people for communication. In Indonesia, the sign language approved by the government for educational purpose is SIBI [1]. SIBI has been standardized according to grammar and word morphology. The root words have already had the sign to enrich the vocabulary [1]. This paper stresses upon translating SIBI.

There are two types of communication in sign language: the one that represents words and the one that represents alphabet letters. The first one is a dynamic gesture that hand, face, and body are taken into account in coordination to produce a word. The latter is mostly a static hand gesture to produce an alphabetical letter or called as finger-spelling. This type of sign language has the purpose to spell letter by letter to achieve more accurate intended word. This means using 26 different hand configurations to represent the letters of the alphabet. In addition to alphabetical letter, numerical numbers are also taken into account to finger-spelling. In every sign language communication, finger-spelling is generally combined with the word signing and is mainly used for spelling nouns (place names, people’s names, or objects’ names) or for spelling words [1].

B. HSV Color Space

HSV stands for Hue, Saturation, and Value. HSV is one of the most common cylindrical Red, Green, Blue (RGB) color model representation for digital image. In each cylinder, the angle around the central vertical axis corresponds to hue, the distance from the axis corresponds to saturation, and the distance along the axis corresponds to value or brightness. HSV is a transformation of RGB so there is mathematical conversion between the two colors is shown in Eq. (1) [7]

$$H = \begin{cases} \left(60\frac{G-B}{M-m}\right) \mod 360 & \text{if} \quad M = R \\ \left(60\frac{B-R}{M-m}\right) + 120 & \text{if} \quad M = G \\ \left(60\frac{R-G}{M-m}\right) + 240 & \text{if} \quad M = B \end{cases} \tag{1}$$

where $M = \max(R, G, B)$ and $m = \min(R, G, B)$.

TABLE I
HSV Color Range.

Color Name	RGB	HSV	Hue Range
Red	255, 0, 0	0°, 100%, 100%	0°–20°, 360°–380°
Blue	0, 0, 255	240°, 100%, 100%	220°–260°
Green	0, 255, 0	120°, 100%, 100%	100°–140°
Purple	255, 0, 255	300°, 100%, 100%	280°–300°

C. Color Detection

Color detection approach is used in order to segment the image based on color to divide the foreground from the background. Color based detection for segmentation may apply a certain lower range and upper range to thoroughly acquire the object of interest. The assignment of the range of the color they desire to segment is up to the user preference. Based on best practice, the lower range may be assigned to 20 Hue value lower and for the upper range may be assigned to 20 Hue value higher of the object’s HSV color. The HSV color and the range of some common colors are seen in Table I [8].

D. Image Processing

Morphological operations are the image processing operations that are used to remove structures or fill the holes of certain shape by a given structural element [9]. It only operates and processes binary images. There are two basic operations in morphological operations: dilation and erosion. Some morphological algorithms such as opening, closing, and top-hat are based on those two primitive operators. Dilation process adds pixels to the boundaries of objects in an image, while erosion process removes pixels on object boundaries to do erosion and dilation to the image. This research implements Emgu CV functions (`Erode()` and `Dilate()`) to perform those processed image by giving parameters such as input image, output image, structuring element, structuring element position, number of iteration, border type, and color.

E. Image Segmentation

Segmentation is the initial stage for any recognition process in which the acquired image is broken down into meaningful regions or segments. The segmentation process is only concerned with partitioning the image and not with what the regions represent. In the simplest case (binary images), only two regions exist, a foreground (object) region and a background region. In gray level images, several types of region or classes may exist within the image. For example, when a natural scene is segmented, regions of clouds, ground, buildings, and trees may exist [10]. The segmentation

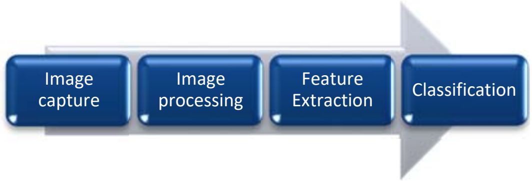


Fig. 1. Gesture recognition process.

process should be stopped when the objects of interest in an application have been isolated [11].

F. Gaussian Blur

A Gaussian blur or Gaussian smoothing process results in a blur image using the Gaussian function. The effect of the process is a smooth blur that resembles viewing the image through translucent screen. Gaussian blur also can be used to obtain smooth digital image instead of pixelated. Gaussian blur process has widely been used in pre-processing image before any operation to reduce image noise and detail [9].

G. Artificial Neural Network

ANN is a method of processing information that models the biological nervous systems of human. Resembling the brain that have neurons connected by synapses, ANN is structured by a large number of connected processing elements (neurons) working synchronously to solve problems [12]. A model of a neural network maps sets the input data onto a set of expected output data and pass through one or more hidden layer(s). The process of classification in ANN includes forward propagation and back propagation. Forward propagation aims to find the output value by combining weights and input which is activated by a sigmoid function. Then, the output value compares the target values to get the margin of error. Back propagation plays its roles to make the error to be smaller by altering the weight value.

III. Research Method

A. Static Gesture Recognition

There are two basic approaches in static gesture recognition as described by Ref. [13]. They are

1) the top-down approach, and
2) the bottom-up approach.

The process of static hand gesture recognition is divided into four stages: image capturing, image processing, feature extraction, and classification as shown in Fig. 1.

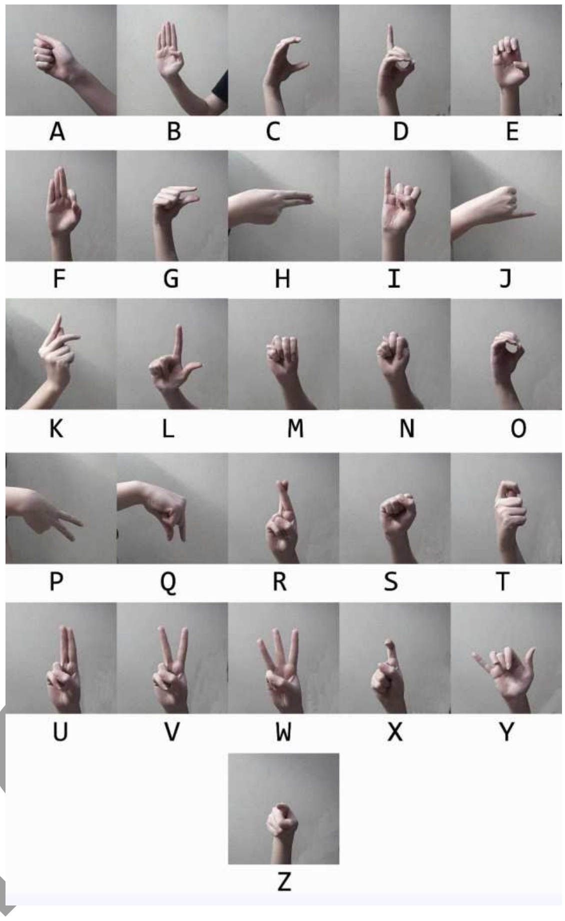


Fig. 2. Alphabet signs.

1) Image Capturing: The capturing is done using a single camera that is external or built-in to capture image in a real-time manner with a view of the person hand that performs the gestures. Each image frame is taken into the entire process flow for the whole time. For example, it can be seen in Figs. 2 and 3.

Image capturing can be done by different color space methods such as RGB, Gray-scale, and HSV. This work uses RGB color space model to capture the image. Then, the image is automatically set to a 20 × 20 resolution by having a total of 400 pixels for each image. Meanwhile, a set of training image is statistically set to be in total of 3900 images. This means each alphabet letter has 100 images and is stored in a one-dimensional array of image with the size of 3900.

2) Image Processing: Some preprocessing of the image is essential so that the desired information can be obtained from the current capture of webcam. First thing to be performed is to divide the background with the hand using a threshold process. A certain range should be explicitly defined according to the HSV color

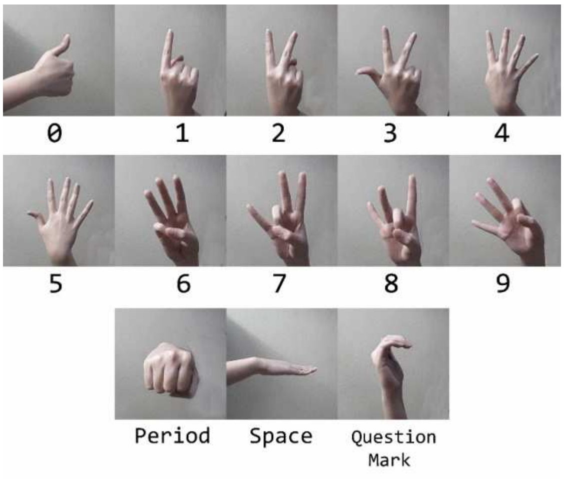


Fig. 3. Number and punctuation mark signs

of the detected object. Then, image is blurred using Gaussian blur. Finally, erosion and dilation processes are applied to the images.

3) Feature Extraction: After the separation of the object with a certain color from its surrounding contours, the system needs to find the image to analyze the objects further. There may be more than one contour existing in the image so it is important to get the largest contour. To get the largest contour, we need to iterate through the size of the available contours and obtain it using a simple algorithm of finding the largest number of area.

4) Classification: Neural network cannot accept the data fed to the input layer in form of images. Therefore, the post-processed images have to be normalized first by changing the image representation to binary. Pixel that is white is converted to zero, and black is converted to one. The input layer for the neural network is proportional to the number of pixel of each image. In this program, the image is automatically set to a 20×20 resolution having a total of 400 pixels for each image. Therefore, the input layer to the network is set to have 400 neurons. Data class for this classification, or the number of neurons in output layer, are the total of 39 representing the total of 10 numbers, 26 alphabet letters, and 3 punctuation marks. The number of hidden layers is set to be the means between input and output layers. The weight and all the information needed are obtained from the XML which is previously generated from the training process. The network performs prediction through the Emgu CV function Predict() and returns a matrix containing 39 values from index zero that represents alphabet “A” until the index 25 for alphabet “Z”.

The best prediction of an image is the largest number among all the 39 values. To display the output, the existing textbox named txtTranslation that is filled with the corresponding output. It can be number, alphabet letter, or punctuation mark. For the alphabet letter, index of the largest value is added by 55 so that it matches the ASCII letter value in order to be converted to a char.

Training the neural network model requires data that are fed into input layer along with its classes (output). Input data in this program are in form of images. As mentioned previously, image cannot be an acceptable input for the neural network. Therefore, normalization of the image has to be done. The initialization of the network layer size in the input is the same as number of pixel in each image that is 400, number of neuron is output layer that is 39, and hidden layer have 220 neurons from the half of the addition of input and output layer.

A set of training images is statistically set to be in total of 3900 images. This means each alphabet letter has 100 images and are stored in a one-dimensional array of image with size of 3900. Training images already have a fixed naming format and they are obtained from user defined folder path in txtFolderPath to populate array of image named imgArray. All images in imgArray is converted to binary and converted to a matrix called inputData. Another matrix, outputData is as the indicator of output class. For instance, the number “0” of alphabet is having the outputData of index zero with value “1”, and the rest of the value is set to “0”. The letter “B” of alphabet is having the outputData of index 11 with value “1”, and the rest of the value is set to “0”, and so on until the last index of 39 which defined punctuation mark.

After all the input and output data has been completely stored in corresponding array, the setting of the neural network is saved to a temporary storage in XML form. Then, the training process starts with the function called Train() and save the information of current training in XML form.

B. Speech Translation

1) Create a choice of words: This work is able to listen and recognize the words that they have listened to. A set of words assigned in the form of *Choices* are the one that the system will recognize. Those are *alpha, bravo, charlie, delta, echo, foxtrot, golf, hotel, india, juliett, kilo, lima, mike, november, oscar, papa, quebec, romeo, sierra, tango, uniform, victor, whiskey, xray, yankee, zulu*

2) Create and load the grammar object: This step is to embed available word choices to a set of grammar

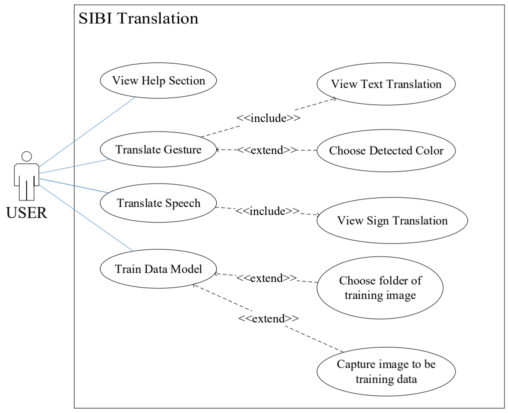


Fig. 4. The use-case diagram.

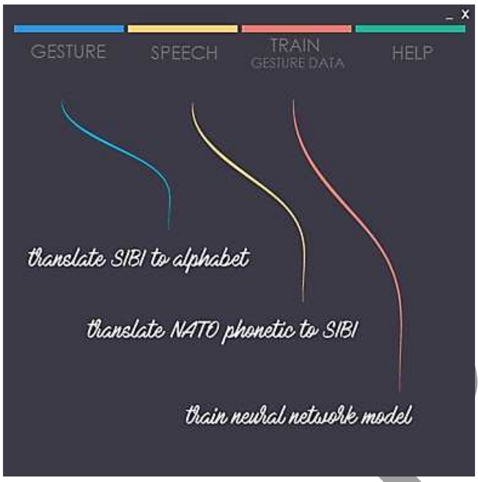

GESTURE
SPEECH
TRAIN
GESTURE DATA
HELP
_ X
translate SIBI to alphabet
translate NATO phonetic to SIBI
train neural network model

Fig. 5. Main menu.

to maximize the efficiency of the recognition. The system will utilize the set of words to be detected by the speech recognition engine through the available grammar.

IV. Results and Discussion

A. Implementation Result

The use case diagram for SIBI translation is shown in Fig. 4. The user can perform four type of main features, which are translate SIBI gesture to text, translate speech to SIBI, train gesture data, and view help section. In translating gesture, user can view text translation, and in the same time, it can choose the color they desired to be detected by the system. In training gesture data model, users start the training right away by browsing folder where they store the training images, or users can prepare the training data first by capturing image to be the training data. The graphical user interface is shown in Fig. 5 until Fig. 9.

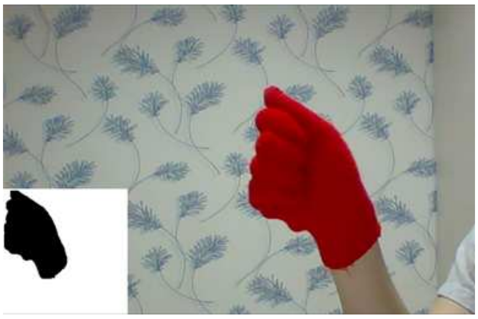


Fig. 6. Webcam capture display.

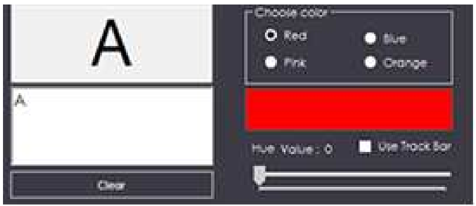
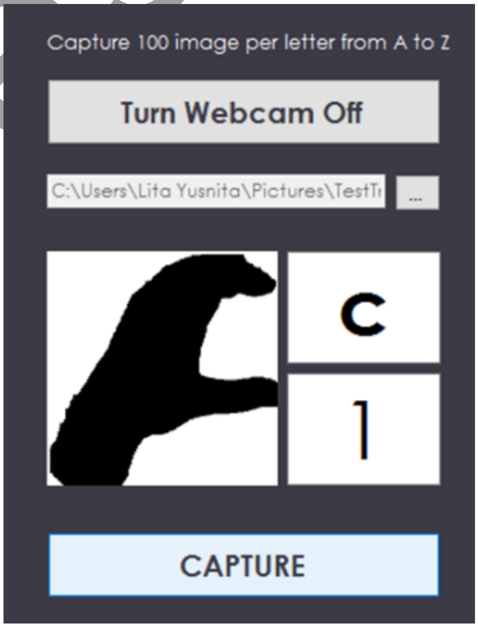


Fig. 7. Translated letter and detected color option

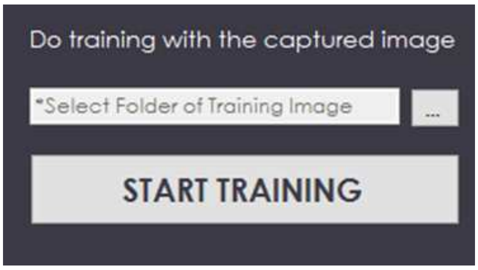


C:\Users\Lita Yusnita\Pictures\TestTr

Fig. 8. Capture training image

When users click Gesture in the tab menu and press the start button, Fig. 6 and Fig. 7 will be shown. The capture of webcam will be displayed in a box along with the small box on the lower left corner containing the cropping of hand detection by a certain color. In this case, it is red color.

When the users click the Training Gesture Data in the tab menu, Fig. 8 and Fig. 9 will be displayed in the screen. A webcam will capture the hand of user



Do training with the captured image

*Select Folder of Training Image

...

START TRAINING

Fig. 9. Train gesture data menu

TABLE II
OUTPUT FOR ADEQUATE LIGHTING WITH APPROXIMATE DISTANCE OF 50 CM.

No.	Gesture	Output		No.	Gesture	Output	
		T	F			T	F
1	A	10	0	21	U	9	1
2	B	9	1	22	V	10	0
3	C	10	0	23	W	10	0
4	D	10	0	24	X	6	4
5	E	8	2	25	Y	10	0
6	F	7	3	26	Z	7	3
7	G	10	0	27	0	10	0
8	H	10	0	28	1	8	2
9	I	10	0	29	2	9	1
10	J	10	0	30	3	10	0
11	K	10	0	31	4	10	0
12	L	10	0	32	5	10	0
13	M	8	2	33	6	5	5
14	N	9	1	34	7	9	1
15	O	8	2	35	8	10	0
16	P	10	0	36	9	8	2
17	Q	10	0	37	'.'	9	1
18	R	10	0	38	'?'	6	4
19	S	9	1	39	Space	9	1
20	T	8	2				
					Total	351	39

that is being performed in front of webcam. Then, it will save the image to the specified folder by the users. Descriptions of current letter and current number of letter is displayed on the right side. In order to train neural network with the training image that users just capture, users can direct the path to the folder previously saved and click the start button.

B. Testing Result

Testing output is done by calculating the accuracy rate over a sample of ten images of each gesture. Testing is with adequate lighting in distance of approximately 50 cm. Calculation of the average of output testing data from Table II shows that the application has 90% of accuracy rate on detecting gestures. Although it has optimal distance of approximately 50 cm, the lighting can also affect the performance and the accuracy of the data. Having inadequate lighting impacts, the accuracy by the insufficient color value of the glove can be segmented.

TABLE III
OUTPUT FOR ADEQUATE LIGHTING WITH APPROXIMATE DISTANCE OF 100 CM.

No.	Gesture	Output		No.	Gesture	Output	
		T	F			T	F
1	A	10	0	21	U	4	6
2	B	9	1	22	V	7	3
3	C	7	3	23	W	8	2
4	D	6	4	24	X	2	8
5	E	6	4	25	Y	8	2
6	F	6	4	26	Z	5	5
7	G	8	2	27	0	9	1
8	H	10	0	28	1	4	6
9	I	9	1	29	2	7	3
10	J	9	1	30	3	8	2
11	K	9	1	31	4	8	2
12	L	8	2	32	5	8	2
13	M	3	7	33	6	2	8
14	N	2	8	34	7	6	4
15	O	2	8	35	8	8	2
16	P	8	2	36	9	7	3
17	Q	9	1	37	'.'	5	5
18	R	4	6	38	'?'	2	8
19	S	8	2	39	Space	8	2
20	T	4	6				
					Total	253	137

TABLE IV
OUTPUT FOR INADEQUATE LIGHTING WITH APPROXIMATE DISTANCE OF 50 CM.

No.	Gesture	Output		No.	Gesture	Output	
		T	F			T	F
1	A	10	0	21	U	9	1
2	B	9	1	22	V	9	1
3	C	7	3	23	W	7	3
4	D	3	7	24	X	1	9
5	E	5	5	25	Y	7	3
6	F	2	8	26	Z	4	6
7	G	8	2	27	0	8	2
8	H	9	1	28	1	5	5
9	I	9	1	29	2	8	2
10	J	7	3	30	3	8	2
11	K	9	1	31	4	9	1
12	L	7	3	32	5	9	1
13	M	4	6	33	6	1	9
14	N	7	3	34	7	6	4
15	O	3	7	35	8	5	5
16	P	8	2	36	9	6	4
17	Q	9	1	37	'.'	8	2
18	R	3	7	38	'?'	1	9
19	S	6	4	39	Space	7	3
20	T	7	3				
					Total	250	140

V. CONCLUSION

The main goal of this research has been successfully achieved. Gesture translation works best in case users who are a speaker of sign language will like to interact with people who do not know any sign language. Speech translation is useful for the non-speakers of sign language that wants to recognize the corresponding hand sign.

Meanwhile, ANN is the classification method in this application with the sample image of 100 for every gesture and the iteration of 90000 times per training data.

Room condition such as lighting may play a role in predicting the result as poor lighting. The light which is too bright or too dim will result in inaccurate segmentation of the hand thus inaccurate prediction of the gesture. Another aspect of inaccuracy may come from the peripheral used by the user such as low quality of web camera or low quality of microphone.

VI. Future Work

Several things can be accounted as an improvement in the near future to deeply enhance the use of this research.

1) Use skin color as the color for segmentation for users hand. It is more practical and effective for the user not to wear additional tools to use this project. One challenge is to divide the palm between the arm of the user
2) Make training process be more flexible, not just fixed value of training data and be more user-friendly.
3) Translate a dynamic hand gesture for words in sign language. This project solves the problem in static fingerspelling but not with the word gesture where it is based on the words instead of the spelling. Word based gesture is more complicated because there are a lot of things to be taken into account such as two hands movement, head movement, and sometimes body movement.

References

[1] D. R. Kurnia and T. Slamet, “Menormalkan yang dianggap ”tidak normal” (studi kasus: Penertiban bahasa isyarat tunarungu di slb malang),” *Indonesian Journal of Disability Studies (IJDS)*, vol. 3, no. 1, pp. 34–43, 2016.

[2] N. B. Bhoyar and M. Bartere, “Neural network based static hand gesture recognition,” *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, vol. 3, no. 2, Feb. 2014.

[3] P. Chaudhary and H. S. Ryait, “Neural network based static sign gesture recognition system,” *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, no. 2, pp. 3066–3072, Feb. 2014.

[4] T. Ahmed, “A neural network based real time hand gesture recognition system,” *International journal of computer applications*, vol. 59, no. 4, Dec. 2012.

[5] T.-N. Nguyen, H.-H. Huynh, and J. Meunier, “Static hand gesture recognition using artificial neural network,” *Journal of Image and Graphics*, vol. 1, no. 1, pp. 34–38, Mar. 2013.

[6] C. C. Chang, J. J. Chen, W. K. Tai, and C. C. Han, “New approach for static gesture recognition,” *Journal of Information Science and Engineering*, vol. 22, no. 5, pp. 1047–1057, 2006.

[7] M. K. Agoston, *Computer Graphics and Geometric Modeling*. Springer Science & Business Media, 2005, vol. 1.

[8] G. Amir. (2008, Nov.) Rapid application development model (rad model). [Online]. Available: http://www.testingexcellence.com/rapid-application-development-rad/

[9] R. C. Gonzalez, “Digital image processing: Pearson education india,” 2009.

[10] G. J. Awcock and R. Thomas, *Applied Image Processing*. McGraw–Hill, Inc., 1995.

[11] R. C. Gonzalez and R. E. Woods, “Digital image processing,” 2002.

[12] M. Alex. (2016, Dec.) Deep learning basics: Neural networkds, backpropagation, and stochastic gradient descent. [Online]. Available: http://alexminnaar.com/deep-learning-basics-neural-networks-backpropagation-and-sto html

[13] H. Zhou, D. J. Lin, and T. S. Huang, “Static hand gesture recognition based on local orientation histogram feature distribution model,” in *Proceedings of Computer Vision and Pattern Recognition Workshop, 2004 (CVPRW’04)*, vol. 10, IEEE. Washington, DC, USA: IEEE Computer Society, 2004, p. 161.