Makara Journal of Technology

Volume 17 | Issue 1 Article 6

4-1-2013

Recognition System of Indonesia Sign Language based on Sensor and Artificial Neural Network

Endang Supriyati
Informatics Engineering Department, Muria Kudus University, Gondangmanis Kudus 59352, Indonesia, esupriyati@gmail.com

Mohammad Iqbal
Electrical Engineering Department, Muria Kudus University, Gondangmanis Kudus 59352, Indonesia

Recommended Citation
Supriyati, Endang and Iqbal, Mohammad (2013) "Recognition System of Indonesia Sign Language based on Sensor and Artificial Neural Network," *Makara Journal of Technology*: Vol. 17: Iss. 1, Article 6.
DOI: 10.7454/mst.v17i1.1924
Available at: https://scholarhub.ui.ac.id/mjt/vol17/iss1/6

Makara Seri Teknologi, 2013, 17(1): 25-31
DOI: 10.7454/mst.v17i1.1924

Recognition System of Indonesia Sign Language based on Sensor and Artificial Neural Network

Endang Supriyati[1*] and Mohammad Iqbal[2]

1. Informatics Engineering Department, Muria Kudus University, Gondangmanis Kudus 59352, Indonesia
2. Electrical Engineering Department, Muria Kudus University, Gondangmanis Kudus 59352, Indonesia

[*]*E-mail: esupriyati@gmail.com*

Abstract

Sign language as a kind of gestures is one of the most natural ways of communication for most people in deaf community. The aim of the sign language recognition is to provide a translation for sign gestures into meaningful text or speech so that communication between deaf and hearing society can easily be made. In this research, the Indonesian sign language recognition system based on flex sensors and an accelerometer is developed. This recognition system uses a sensory glove to capture data. The sensor data that are processed into feature vector are the 5-fingers bending and the palm acceleration when performing the sign language. The most important part of the recognition system is a feature extraction. In this research, histogram is used as feature extraction. The extracted features are used as data training and data testing for Adaptive Neighborhood based Modified Backpropagation (ANMBP). The system is implemented and tested using a data set of 1000 samples of 50 Indonesia sign, 20 samples for each sign. Among these 500 data were used as the training data, and the remaining 500 data were used as the testing data. The system obtains the recognition rate of 91.60% in offline mode.

Abstrak

Sistem Pengenalan Bahasa Isyarat Indonesia Berbasis Sensor dan Jaringan Saraf Tiruan. Bahasa isyarat adalah salah satu cara yang paling alami dalam berkomunikasi bagi kaum tuna rungu. Tujuan pengenalan bahasa isyarat adalah untuk menerjemahkan bahasa isyarat ke dalam bentuk teks dan atau suara sehingga komunikasi antara kaum tuna rungu dengan masyarakat luas dapat terjalin. Penelitian ini mengembangkan sistem isyarat bahasa Indonesia berbasis sensor flex dan accelerometer. Sistem pengenalan ini menggunakan sarung tangan bersensor untuk mengumpulkan data. Data sensor yang diolah menjadi vektor ciri adalah data tekukan kelima jari tangan dan akselarasi telapak tangan selama melakukan gerak isyarat kata. Bagian terpenting pada sistem pengenalan adalah ekstraksi fitur. Proses ekstraksi fitur dalam penelitian ini menggunakan histogram. Hasil ekstraksi fitur digunakan sebagai data training dan data testing pada sistem pengenal menggunakan Adaptive Neighborhood based Modified Backpropagation (ANMBP). Sistem pengenalan bahasa isyarat ini diimplementasikan dan diuji coba dengan menggunakan 1000 data dari 50 gerak isyarat, dengan jumlah 20 data untuk tiap gerak isyarat. 500 data digunakan untuk data training dan 500 data untuk data testing. Hasil pengujian mendapatkan akurasi 91,60% dalam mode offline.

Keywords: accelerometer sensor, backpropagation, flex sensor, Indonesia sign language, neural network

1. Introduction

Indonesian sign language is the main communication tool for the deaf and the mute in Indonesia. The deaf and the muteuse writing or a translator to communicate with normal people. However, translators often have a high rate, as well as the risk of dependency and loss of privacy.

Progress in the field of pattern recognition systems promise to automate translation gesture, despite the many difficulties faced for implementation of such systems. Research in the field of sign language can be grouped into two categories: (i) based on computer vision [1-3], (ii) based on sensors [4-5]. Computer vision for gesture recognition is based on image processing and feature extraction techniques to extract

Makara Seri Teknologi, 2013, 17(1): 25-31
DOI: 10.7454/mst.v17i1.1924

and classify the hand gestures and body movements. While sensor-based gesture recognition use gloves and a motion sensor to detect hand gestures and movements.

Some research has been made on gesture recognition using Artificial Neural Networks [2,5], the Hidden Markov Model [3-4], statistical methods, and so on [6]. In this research, the learning process use a method of modified backpropagationwith adaptive learning rate, neighborhood random in the hidden layer and the adaptive weight which is the sum of linear error and non-linear error [7]. Indonesia sign language has conducted previous research [8], which used static sign words as dataset and used only a flex sensor. This research tried to recognize the gesture for dynamic sign wordby using flex sensors and accelerometer sensors to measure movement in the x, y, z.

The structure of this paper is the introduction in the first part, the sign language of Indonesia in the second part, the sensor and data acquisition in the third part, feature extraction in the fourth part, ANMBP (Adaptive Neighborhood-based Modified Backpropagation) in the fifth part. The sixth is the results and discussion and the seventh section is conclusion.

2. Experiment

Indonesia sign language. Indonesia Sign language is also known as SIBI (Sistem Isyarat Bahasa Indonesia). The fingers and hand movements form the main component of the signal. In most of the sign words, hand movements are highly varied and more dominant than the formation of the fingers. In Evita study [8], which only uses flex sensors, their identification accuracy dropped from 83.18% for recognition of static sign word, to 49.58% for recognition of dynamic sign word. In this research, the introduction is intended for sign words in Indonesian, with the addition of other types of sensors. The two main components of the information signal are measured with the use of the flex sensors and accelerometer sensors that are integrated in the form of gloves as a data acquisition tool. Figure 1 shows a block diagram of Indonesia sign language recognition system using ANMBP.

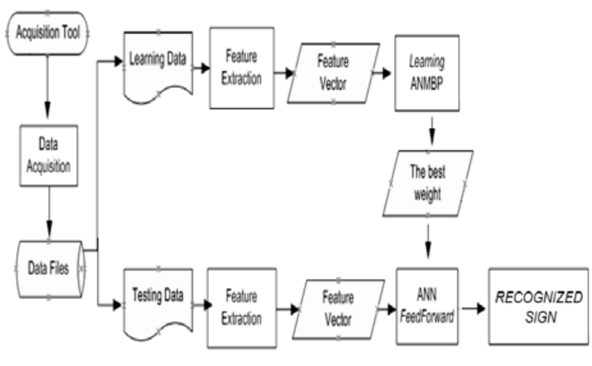


Figure 1. Diagram Block of Indonesia Sign Language Recognition System Using ANMBP

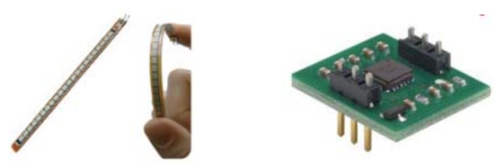
Sensor and data acquisition. The sensors used in this research are flex sensors and accelerometer sensor, as shown in Figure 2. Flex sensor is a type of sensor that changes its resistance when it is bent. In the normal condition (not bent), flex sensor has a value of 10 KΩ resistance. Resistance value will be even bigger up to 40 KΩ at maximum when its bent [9]. Because it has only one degree of freedom, each finger just uses a flex sensor. While the accelerometer sensor used is Hitachi H48C, the sensor module is made by Parallax. This sensor module is integrated with Analog to Digital Converter (ADC) and a voltage regulator. Accelerometer can detect inclination (tilt) and motion (acceleration) in three axes x, y, z with a maximum measurement range ±3 g (1 g = 9.81 m/s^2), non-linearity ±2% and sensitivity 333 mV / g at a voltage of 3 volt [10].

Data acquisition. This phase is intended to get data from sensors and then processed into a feature vector, as shown in Figure 3.

The data obtained from the sensors are as many as 8 kinds of data. The data is as follows: a) degree bend thumb, b) degree bend finger, c) the degree of bending the middle finger, d) the degree of bending the little finger, e) the degree of bending of the ring finger, f) hand movement to the axis x, y, z.

For the fifth finger grooves Data were normalized to equalize the value difference between the tolerance of used flex sensors. The value of bent fingers are normalized [11] on the range of values 0 to 20 by using equation (1) before being stored in the file dataset.

$$X_{norm} = \frac{X_i - X_{\min}}{X_{\min} + X_{\max}} * 20 \tag{1}$$



(a) Flex Sensor (b) accelerometer

Figure 2. Sensors Used

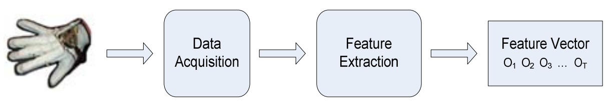


Figure 3. Diagram Block of Data Acquisition

Tabel 1. Sample Data for Sign Word 'Brother'

No	Flex sensor					Accelerometer			
	1	2	3	4	5	x	y	z	reff
1	3	2	3	4	3	133	119	136	127
2	3	2	3	4	3	134	111	139	127
3	3	2	3	4	3	132	112	138	127
4	3	2	3	4	3	138	119	142	127
5	3	2	3	4	3	156	129	158	128
6	3	2	3	4	3	152	127	154	127
7	3	2	3	3	3	145	127	150	127
8	3	2	3	4	3	140	129	141	127
9	3	2	3	4	3	142	135	143	127
10	3	2	3	4	3	147	143	147	127
11	3	2	3	4	3	144	147	146	127

The acceleration data are still raw counting data, taken from the 8-bit MSB (Most Significant Bit) of 12 bits of data to counter it. To get the actual value of acceleration requires special calculations described in the feature extraction. Sample data for sign word 'brother' is stored in the form of a text format file with defined extensioni.e. *.ibi, as shown in Table 1. The data stored is the data sequence, the length (number of lines) is different depending on the type and speed of movement sign word. In the example of Table 1, the length of the data signaling word 'brother' is 11.Column (1 ... 5) is the data from the sensor flex for the thumb to the little finger, while (6 ... 9) is the data from accelerometer sensor for axes X, Y, Z and references. Line (1 .. 11) shows the sequence of data captured, from the beginning until the completion of making sign word.

Indonesian sign words made by reference to the video of Indonesia sign language contained in [12] which isvisualization for the dictionary of Indonesia sign language.

Feature extraction. Feature extraction is performed to obtain quantities indicating the specificity of the data being processed. Feature extraction for Indonesia sign language recognition system uses a statistical approach that uses histogram for the bending of fingers data and palms data for acceleration. For acceleration palms data, before transformed into the form of a histogram quantization, process is carried out for the testing data and the learning data and done with the same feature extraction to obtain feature vector. Feature vector consists of the processed data values from the flex sensor and the accelerometer. The flex sensor data representing characteristic shape of the hand is calculated based on the frequency of occurrence with the histogram method. Expressed d is the number of sub-range, h_i is a column histogram $0 \le i \le d$ and n is the number of data. Here is a histogram equation used:

$$h_i = \sum_j^n \gamma_i(x_i) \forall_i 0 \le i \le d \qquad (2)$$

with

$$\gamma_i(x_j) = \begin{cases} 1 & if\ x_j = i \\ 0 & if\ x_j \neq i \end{cases}$$

Figure 4 shows the feature extraction process, as if a notation for the acceleration, then a_{x1} showed acceleration values for the x-axis to the first data (first line in the file), a_{y1} to the value of the y-axis acceleration of first data, a_{z1} to the value of the z-axis acceleration of first data, a_{x2} to the x-axis acceleration value of the second data, and so on.

If k is a notation for the value of quantization acceleration, then the acceleration k_{x1} shows quantitation values for the x-axis to the first data (first row), k_{y1} for quantitation values y axis acceleration of first data, k_{z1} for the value of quantization z-axis acceleration of first data, k_{x2} for quantitation values x-axis acceleration of second data, and so on. Figure 5 shows the histogram curve of your fingers.

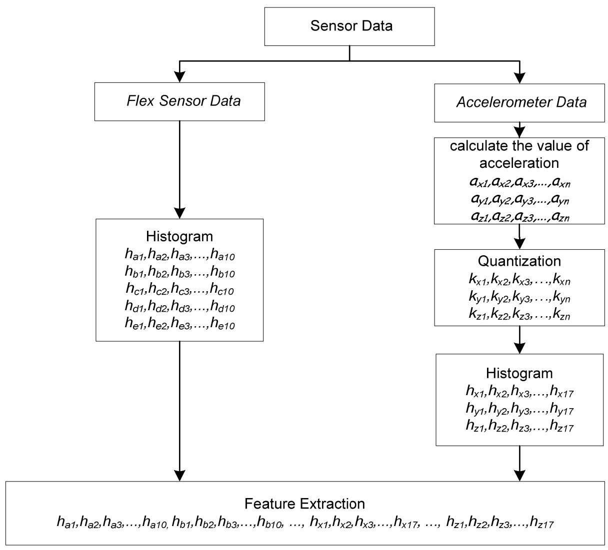


Figure 4. Feature Extraction Process

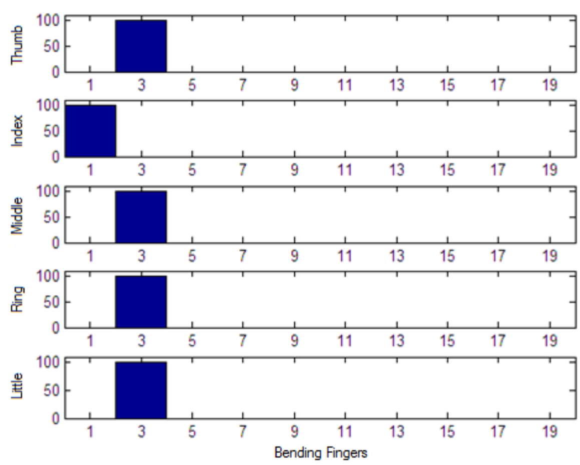


Figure 5. Histogram of Five Bending Fingers

Makara Seri Teknologi, 2013, 17(1): 25-31
DOI: 10.7454/mst.v17i1.1924

Table 2. Acceleration Quantitation

Acceleration (*a*)	Quantitation (**k**)
$a > 2$ g	16
g $< a \leq 2$ g	$10 < k \leq 15$
$0 < a \leq$ g	$1 < k \leq 10$
$a = 0$	$k = 0$
-g $\leq a < 0$	$-10 \leq k < 0$
-2g $\leq a <$ -g	$-15 \leq k < -10$
$a <$ -2 g	-16

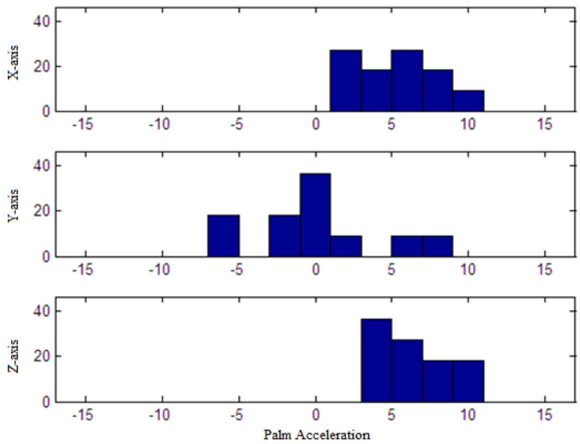


Figure 6. Histogram of Palm Acceleration

While for the data,accelerometer calculated acceleration as a true value, and used the equation (3) as specified in the datasheet [10], where

$$a = (c - reff) * 0.0022 * 16 \qquad (3)$$

A value of acceleration is relatively small to the value of the normalization of the curve of the finger, so do quantization [13-14]. A value of acceleration for each axis X, Y, and Z are based on Table 2 (1g = 9.81 m/s2). Figure 6 shows the histogram of palm acceleration.

ANMBP (Adaptive neighborhood based modified backpropagation). Backpropagation is an algorithm in artificial neural networks that are widely used to solve various kinds of problems, one of which is classified. Problems that occur in backpropagationare the convergence of the old and stuck in a local minimum. Many algorithms are proposed to solve the problem of one ANMBP method [7]. ANMBP method uses a modified backpropagation algorithm [15], neighborhood structure in the hidden layer [16] and adaptive learning rate [17]. In ANMBP, algorithm to find the value of net use is as follows:

$$u_j^s = \sum_{i=1}^{n} w_{ji}^s y_i^{s-1} \qquad (4)$$

By using binary sigmoid activation function, the output is:

$$f(u_j^s) = \frac{1}{(1+e^{-u_j^s})} = y_j^s \qquad (5)$$

With n indicating the number of neurons and the weight of the neuron to the i of the layer (s-1) to neuron j from layer to s.In the algorithm used ANMBP $Ep$is the sum of the squares of linear and non-linear error of the output.

$$E_p = \sum_{j=1}^{n} \frac{1}{2}(e_{1j}^s)^2 + \sum_{j=1}^{n} \frac{1}{2}\lambda(e_{2j}^s)^2 \qquad (6)$$

where λ is a weigh coefficient, linear error $e1$ and nonlinear error $e2$ obtained from:

$$e_{1j}^s = d_j^s - y_j^s \qquad (7)$$

$$e_{2j}^s = ld_j^s - u_j^s \qquad (8)$$

$$ld_j^s = f^{-1}(d_j^s) \qquad (9)$$

where d is the expected output and y is the output. So the change in weights in the output layer is:

$$\Delta w_{ji}^s = -\mu \frac{\partial E}{\partial W_{ji}} \qquad (10)$$

$$\Delta w_{ji}^s = \mu e_{1j}^s \frac{\partial y_j^s}{\partial W_{ji}^s} + \mu\lambda e_{2j}^s \frac{\partial u_j^s}{\partial W_{ji}^s}$$

$$\Delta w_{ji}^s = \mu e_{1j}^s \frac{\partial y_j^s}{\partial u_j^s}\frac{\partial u_j^s}{\partial W_{ji}^s} + \mu\lambda e_{2j}^s y_i^{s-1}$$

$$\Delta w_{ji}^s = \mu e_{1j}^s f'(u_j^s) y_i^{s-1} + \mu\lambda e_{2j}^s y_i^{s-1} \qquad (11)$$

Error linear and non-linear in the hidden layer (L) is:

$$e_{1j}^L = \sum_{r=1}^{n_{L+1}} f'(u_r^{L+1}) e_{ir}^{L+1} w_{rj}^{L+1} \qquad (12)$$

$$e_{2j}^L = f^1(u_j^L)) \sum_{r=1}^{n_{L+1}} e_{2r}^{L+1} w_{rj}^{L+1} \qquad (13)$$

So the changes in the hidden layer weights are:

$$\Delta w_{ji}^L = \mu e_{1j}^L y_i^{L-1} f'(u_j^L) + \mu\lambda e_{2j}^L y_i^{L-1} \qquad (14)$$

Learning parameters μ and $\mu\lambda$ are replaced with adaptive parameter [17], namely:

$$\eta' = \frac{\mu \| \tilde{y} \|^2}{\| J_p^T \tilde{y} \|^2 + \varepsilon} \qquad (15)$$

$$\mu' = \frac{\mu \| \tilde{y} \|^2}{\| J_p^T \tilde{y} \|^2 + \varepsilon} \qquad (16)$$

with $\tilde{y} = d_j^s - y_j^s = \beta_{1_j}^s$ (17)

and $J_p^T = \dfrac{\partial y_p}{\partial w_{ji}}$ (18)

Substitute (17) and (18) into the equation (15) and (16) so that the adaptive learning parameter becomes:

$$\eta' = \frac{\mu \| \beta_{1j} \|^2}{\| \frac{\partial y_p}{\partial w_{ji}} \beta_{1j} \|^2 + \varepsilon} \quad (19)$$

$$\eta' = \frac{\mu \| \beta_{1j} \|^2}{\| \frac{\partial y_p \partial u_{ji}}{\partial u_j \partial w_{ji}} \beta_{1j} \|^2 + \varepsilon} \quad (20)$$

$$\eta' = \frac{\mu \| e_{1j} \|^2}{\| f'(u_j) y_i^{s-1} e_{1j} \|^2 + \varepsilon} \quad (21)$$

$$\mu' = \frac{\lambda \| e_{1j} \|^2}{\| f'(u_j) y_i^{s-1} e_{1j} \|^2 + \varepsilon} \quad (22)$$

Where μ, λ is a small positive constant value and constant ε with a small positive value to ensure error towards instability when 0. The selection of the value of μ, λ and $\varepsilon$is conducted heuristics to obtain optimal results.So the change in weights in the output layer and a hidden layer is:

$$\Delta w_{ji}^s = \eta' e_{1j}^s f'(u_j^s) y_i^{s-1} + \mu' e_{2j}^s y_i^{s-1} \quad (23)$$

With the change in weight, a new weight is calculated by the equation (24)

$$w(t+1) = w(t) + \Delta w(t) \quad (24)$$

Where t is the iteration. Mean Squared Error (MSE) [18] of the network is calculated from the network is calculated by the mean of the squares of non-linear error.

$$MSE = \frac{1}{p}\frac{1}{N} \sum_{p=1}^{p} \sum_{j=1}^{N} \left(d_{jp}^s - y_{jp}^s \right)^2 \quad (25)$$

With d = the expected target, y = output ANN, p = the amount of training data, N = number of output ANN. Neighborhood in ANMBP approach is to minimize the time and weight during training. Different Neighborhood is formed randomly in the hidden layer

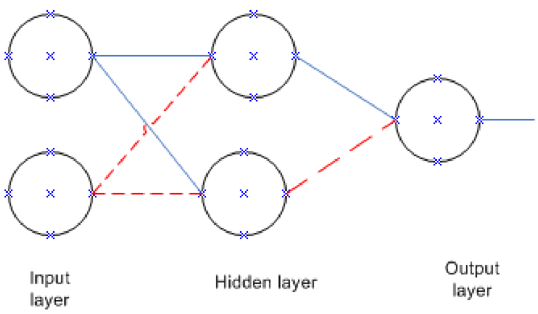


Figure 7. Structure of the Network

with the same number of neurons. Therefore, it causes the minimum number of operations and required memory.

ANMBP algorithms for multiclass classification of Indonesia sign language are as follows: 1) Initialization: set the network structure, initial weight, define neighbourhood, 2) Choose a random neighbourhood, 3) Determine the pattern of learning feed forward, 4) For each unit/node in the hidden layer, compute:net using the equation (4), output using the equation (5), 5) For unit/node in the output layer, compute:net using the equation (4), output using the equation (5), calculate linear error (7) and non-linear error (8), change the weights with (21), (22), (23) for the selected neighbourhood, 6) For the hidden layer nodes, compute: calculate linear error (12) and non-linear error (13). change the weights with (21), (22), (23) for the selected neighbourhood, 7) Update weights, node updates the weights on the output layer and the hidden layer node using the equation (23) for the selected neighborhood, 8. Repeat steps 3-7 for the entire pattern, 9. Evaluation of network error with new weight, 10. Stop when the desired condition is reached, if not repeat steps 2-10.

Figure 7 shows the structure of the network, the dotted lines indicating the unit/node in the hidden layer are chosen randomly. In the process forward (forward propagation), across units/nodes at hidden layer are used, whereas the backward (backward propagation) units/nodes use only a portion of the randomly selected. Therefore, the system is able to accelerate the convergence.

3. Results and Discussion

To perform the test, the data taken consists of 50 gesture classes; each word has 20 data samples, so there are a total of 1000 datasets. From the 1000 sample data, 500 data are taken for each class as a testing data and 500 remained as learning data. The word is a word taken move. Words do not only require finger gestures but also requires moving the hands or other parts of the hand. Figure 8, is a example of the sign word 'brother'. The devices used for computation using Acer Aspire 3810T with Intel specifications Core2Solo 1.4 GHz FSB 800MHz, 2GB DDR3 RAM and Intel GMA 4500MHD Graphic Card, while the tool in use is the Delphi 7.0 version to process data acquisition and Matlab version 7.0 for computing the recognition process.

Network structure of Indonesia sign language recognition system is 101-100-50. Bias is added to the input layer and hidden layer. Activation function used is sigmoid binary. Single hidden layer is used in this study. The number of inputs into the ANN learning is 101 consisting of 5x10 and 3x17,where 5 x 10 is the amount of flex sensor 5 and 10 is the number of occurrence range of values. However,

Makara Seri Teknologi, 2013, 17(1): 25-31
DOI: 10.7454/mst.v17i1.1924

3 x 17 is 3 axis x, y, z from the accelerometer and 17 is the number of the appearance of the value range. The hidden layer using 100 units, while the output layer is of 50 units. Evaluation is done by confusion matrix [19]. Instrument performance measurement system is the recognition accuracy.

$$accuracy = \frac{number\ of\ correct\ prediction}{number\ of\ testing\ data} \qquad (26)$$

To achieve optimal results selecting constants μ, λ and ε in adaptive learning rate need to be considered. The selection is done by heuristic value. In this research, a constant value of μ is 0.4, λ is 0.00001, ε is 0.1, bias is 1, and maximum epoch is 200.

Figure 9 shows the MSE using ANMBP algorithm. Table 3 shows the results of experiments with ANMBP, testing with training data showed an accuracy of 100%, thus the system managed to recognize all the training patterns. Testing data shows 91.60% accuracy because

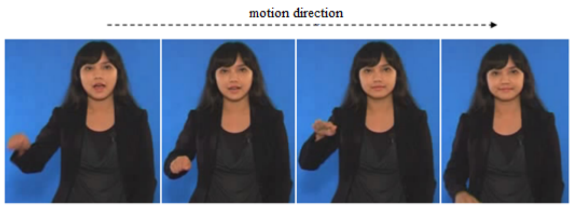


Figure 8. Sign Word 'Brother'

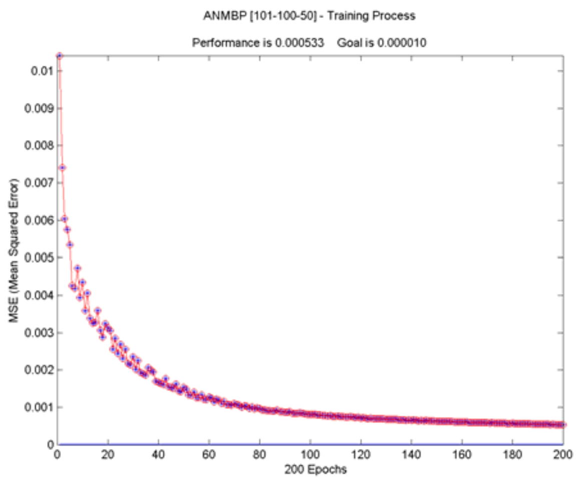


Figure 9. MSE Curve Using ANMBP

Table 3. Experiment Result Using ANMBP

Description	Value
Accuracy for testing training data	100%
Accuracy for testing using testing data	91.60%
Performance (MSE)	0.000533
Training duration	27 minutes

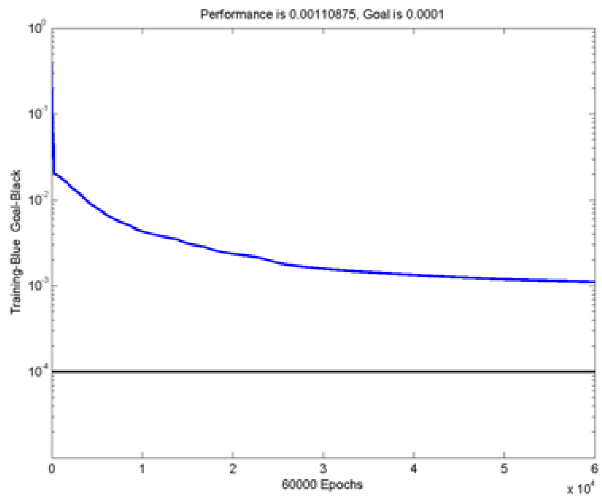


Figure 10. MSE Curve Using GDM

Table 4. Experiment Result Using GDM

Description	Value
Accuracy for testing training data	96%
Accuracy for testing using testing data	90.80%
Performance (MSE)	0.011
Training duration	90 menit

the data tests different training data. Performance (MSE) is 0.000533 and training duration is 27 minutes. The next testing is the steepest descent algorithm (Gradient descent) with momentum. This function is provided by matlab as TRAINGDM function. Network structure on algorithm Gradient Descent with Momentum (GDM) is 101-100-50, which are 101 units in the input layer, the hidden layer of 100 units and 50 units in the output layer. Activation function used is the binary sigmoid activation function. Epoch destination is 60 000.

Figure 10 shows the MSE algorithm GDM. Table 5 shows the results of the fifth experiment, testing the training data showed an accuracy of 96%, thus the system managed to identify almost all the training patterns. Testing data shows 90.80% accuracy to recognize system testing. Performance (MSE) is 0.0011 and the training time is of 90 minutes.

4. Conclusions

This research developed a sign language recognition system Indonesia based on flex sensor and accelerometer. Greatest accuracy 91.60% is obtained by using the histogram feature extraction, and methods of training with ANMBP. Epoch on ANMBP algorithm reaches 200 epoch to 91.60% accuracy with training duration of 27 minutes, While the GDM algorithm reach 60000 epoch and 90.80% accuracy with training duration of 90 minutes.

Acknowledgement

This study received funding from the Directorate General of Higher Education through Beasiswa Program Pascasarjana (BPPS) for the academic year 2009/2010 until 2010/2011.

References

[1] Q. Munib, M. Habeeb, B. Takruri, H.A Al-Malik, Expert Sistems Appl. 32 (2007) 24.
[2] Y.-H. Lee, C.-Y. Tsai, Expert Sistems Appl. 36 (2009) 1123.
[3] M. AL-Rousan, K. Assaleh, A. Tala'a, Appl. Soft Comput. 9 (2009) 990.
[4] W. Gao, G. Fanga, D. Zhaoa, Y. Chen, Pattern Recognit. 37 (2004) 2389.
[5] C. Oz, M.C. Leu, Neuro Comput. 70 (2007) 2891.
[6] M.H. Yang, N. Ahuja, IEEE Conference on Computer Vision and Pattern Recognition, Fort Collins, Colorado, 1 (1999) 466.
[7] T. Kathirvalakumar, S.J. Subavathi, Neuro Comput. 72 (2009) 3915.
[8] E.T. Sekar, Tesis Magister ITB, Bandung, Indonesia, 2001.
[9] Spectrasymbol, Flex Sensor, *http://spectrasymbol.com*, 2012.
[10] Parallax Inc., Hitachi® H48C 3-Axis Accelerometer Module, *http://www.parallax.com*, 2007.
[11] S. Abid, F. Fnaiech, M. Najim, IEEE Trans. Neural Netw. 12 (2001) 424.
[12] A. Toledo, M. Pinzolas, J.J. Ibarrola, G. Lera, IEEE Trans. Neural Netw. 16 (2005) 988.
[13] L. Behera, S. Kumar, A. Patnaik, IEEE Trans. Neural Netw. 17 (2006) 1116.
[14] PT. Telekomunikasi Indonesia, Tbk., i-CHAT, *http://app.i-chat.web.id*, 2010.
[15] H.P. Mauridhi, K. Agus, Supervised Neural Network dan Aplikasinya, Penerbit Graha Ilmu, Yogyakarta, 2006, p.176.
[16] A. Akl, Thesis, Department of Electrical and Computer Engineering, University of Toronto, 2010.
[17] J. Liua, L. Zhong, J. Wickramasuriya, V. Vasudevan, Pervasive and Mobile Comput. 5 (2009) 657.
[18] R. Kohavi, F. Provost, Glossary of Terms, Mach. Learn. 30/2-3 (1998) 271.
[19] M.M. Fernandez, Fundamentals on Estimation Theory, *http://lmi.bwh.harvard.edu/papers/pdfs/2004/martin-fernandezCOURSE04b.pdf*, access on 5-Peb-2010.