

Implementation of Signature Recognition by Using Backpropagation

Rini Sovia^{1*}, Musli Yanto¹, and Widya Nursanty¹

¹Universitas Putra Indonesia YPTK
Padang, Sumatera Barat, Indonesia

*rini_sovia@upiypk.ac.id

Received 25 February 2016; Revised 23 March 2016; Accepted 20 June 2016

Abstract

Many things are required by all parties, especially in the process of recognition of one's identity, ranging from health care, maintenance of bank accounts, aviation services, immigration and others. Many ways of proving one's identity and the most popular one is using a signature. The signature is used as an identification system which serves to recognize a person's identity. Recognition process is still done manually by matching the signature by the person concerned. Therefore, the very need for a system that is able to analyze and identify the characteristics of the signature, so it can be used as an alternative to simplify the process of introducing people's signature. Artificial neural networks can be used as one of the solutions in identification of signatures. Artificial neural network is a branch of science of artificial intelligence that is capable of processing information with the performance of certain characteristics. Artificial neural networks have some method such as perceptron, Hopfield discrete, Adaline, Backpropagation, and Kohonen. In this paper, the artificial neural network with back propagation method is applied in the process of signature and pattern recognition which provided a solution that is able to analyze and recognize people's signature. Implementation of the application of neural networks in pattern recognition signature can further be applied to any computer that handles problems in the process of matching one's data.

Keywords: Neural network; Backpropagation; Identification; Signature.

1. Preliminary

The introduction of a signature is one field of pattern recognition has attracted many researches recently, where the application can be applied in various fields, especially in the field of security system, such as withdrawal of money in the bank [1]. Technology pattern signature included in the biometrics using natural human behavioral characteristics [2,3]. There are several methods of recognition of signatures, one of which is a method based on Artificial Neural Networks [4,5]. Neural Networks have been used widely in the field of pattern recognition and generally show the advantages compared to methods other learning, the nature of generalization and adaptability, as well as his trademark strength in performing non-linear mapping [7,8,9,10,11]. Nevertheless the introduction of the signature included most difficult problems in pattern recognition. This is because people's signatures can be similar but not identical. One of the Artificial Neural Networks methods that can be used to perform the pattern recognition method signature is Backpropagation [12]. This method is one of the learning algorithms to adjust the weights of a neural network with backward direction based on the error value in the learning process. The network will be trained continuously to obtain a minimum error and signature pattern recognition can be performed [13,14].

This paper presents an implementation of signature recognition by using backpropagation. There are two questions raised i.e. How to implement neural network method to perform the introduction of the signature? and How to design methods of Artificial Neural Networks in avoiding fraudulent

signatures?. Then, with the problems that have been described is the limitation on the core problems facing the data is input that will be used is the data in the form of images in the form of written signatures derived from the type of paper that will image it will be Artificial Neural Networks, so that the file format is shaped jpg, bmp, png, or gif. Then on each image signature is equipped with a photo of the owner of a signature that could later be used as well as proof of ownership of the signature. The introduction of last signature is done with 10 samples that will be trained using propagation. This study also presents a design of an application that can perform the signature recognition using the above proposed method.

The rest of this paper is organized as follows. Section 2 presents fundamental background. Section 3 presents the proposed method. Section 4 presents implementation, results obtained and following by discussion. Finally, the conclusion of this work is presented in section 5.

2. Fundamental Background

Software engineering is an engineering discipline that is concerned with all aspects of software production from the early stages of system specification through to maintaining the system after it has gone into use. In this definition, there are two key phrases:

1. *Engineering discipline* Engineers make things work. They apply theories, methods, and tools where these are appropriate. However, they use them selectively and always try to discover solutions to problems even when there are no applicable theories and methods. Engineers also recognize that they must work to organizational and financial constraints so they look for solutions within these constraints.
2. *All aspects of software production* Software engineering is not just concerned with the technical processes of software development. It also includes activities such as software project management and the development of tools, methods, and theories to support software production [15].

Software engineers must often take responsibility for user interface design as well as the design of the software to implement that interface. These different disciplines bring different perspectives to bear on designing the human-computer interface [16].

2.1. Unified Model Language (UML)

The Unified Modeling Language is a graphical notation for capturing requirements and expressing the design of a software system, providing a means for developing blueprints of a software system. UML assists software architects in getting the “big-picture” of a system by providing a balance between natural language (which is too imprecise) and code (which is too detailed). UML is also the industry-standard modeling language [17]. The Unified Modelling Language (UML) is one of the most exciting and useful tools in the world of system development. The UML is a visual modelling language that enables system builders to create blueprints that capture their visions in a standard, easy-to-understand way, and provides a mechanism to effectively share and communicate these visions each other. Communicating the vision is of utmost importance. Before the advance of the UML, system development was often a hit-or-miss proposition. System analyst would try to assess the needs of their clients generate a requirements analysis in some notation that the analyst understood (but not always the client), give that analysis to a programmer or team of programmer, and hope that the final product was the system the client wanted [18].

2.2. Image Processing

Digital image processing is a discipline that studies matters related to image quality improvements (enhancing the contrast, color transformation, restoration of images), image transformation (rotation, translation, scaling, transformation, geometric), an election imagery characteristic (*feature image*) is optimal for analysis purposes, the process of withdrawal of information or description of the object or the introduction of objects contained in the image, compression or data reduction for the purpose of data storage, data transmission, and processing time data. In this study, carried out the first stage of image processing, the X-ray color conversion of RGB color to

grayscale. After that, each X-ray it will be through thresholding to take a detailed framework of the hand bone area (especially joints) needed in diagnosing osteoarthritis [19].

Image processing involves processing or altering an existing image in a desired manner. The first step is obtaining an image in a readable format. This is much easier today than five years back. The Internet and other sources provide countless images in standard formats. This chapter describes the TIFF and BMP file formats and presents source code that reads and writes images in these formats. Once the image is in a readable format, image processing software needs to read it so it can be processed and written back to a file. This chapter presents a set of routines that do the reading and writing in a manner that frees the image processing programming from the details [20].

2.2.1. Binary Image

The binary image is a digital image that has only two possible values: black and white. Pixel image is also referred to as a binary image of the B & W or monochrome image. It only takes 1 bit to represent the value of each pixel of the binary image. The binary image can be used to distinguish between red hue and the other regions. If the colors in the range of "redness", then the second command line above to convert the values range from 0 to 65535-1. Otherwise, zero if it is outside that range. Selection of a value below 5000 is due to the fact that the red color appears on both sides of 0° (or 360°) of the color circle.

2.2.2. Image Grayscale

Grayscale image is a digital image that has only one channel on each pixel value, in other indicates the level of intensity. The color owned is the color of black, gray-and-gray in white. The level here is a gray color with various levels of black to nearly white. The following grayscale image has a color depth of 8 bits (256 color combinations of gray) words the value of the fraction Red = Green = Blue [21].

2.3. Artificial Intelligence

Artificial intelligence (AI) is a field of computer science that explores computational models of problem solving, where the problems to be solved are of the complexity of problems solved by human beings. An AI researcher does not need to be a computer expert, because they might believe that computers can do things brains cannot do computationally. Most AI researchers are computer experts to some extent, even if they think digital computers and brains-as-computers compute things in different ways [22]. Artificial Intelligence techniques are used to solve relatively simple problems or complex problems that are internal to more complex systems. For example, the search techniques described in Chapter 4 are rarely used to provide a robot with the ability to find its way out of a maze, but are frequently used for much more prosaic problems [23]. Artificial intelligence (AI) applications are utilized to simulate human intelligence for either solving a problem or making a decision. AI provides the advantages of permanency, reliability, and cost-effectiveness while also addressing uncertainty and speed in either solving a problem or reaching a decision. The AI has been applied in such diverse realms as engineering, economics, linguistics, law, manufacturing and medicine, and for a variety of modeling, prediction, and decision support and control application [24].

2.4. Artificial Neural Networks

The neural network models are commonly used for processing classification problems. But signature verification differs from the general classification problem. The goal of the general classification problem is to choose one class from several classes, whereas the training data contain data from all classes. For our application all the training data are authentic signatures and we have no data for the second class fake signatures. This is the reason why the frequently used supervised learned neural network model such as multi-layer perceptron cannot be applied to the signature verification task [25]. Artificial Neural Network is a computational model, which is based on Biological Neural Network. Artificial Neural Network is often called as Neural Network (NN). To build artificial neural network, artificial neurons, also called as nodes, are interconnected. The

architecture of NN is very important for performing a particular computation. Some neurons are arranged to take inputs from outside environment. These neurons are not connected with each other, so the arrangement of these neurons is in a layer, called as Input layer. All the neurons of input layer are producing some output, which is the input to next layer. The architecture of NN can be of single layer or multilayer. In a single layer Neural Network, only one input layer and one output layer is there, while in multilayer neural network, there can be one or more hidden layer [26]. Several network architectures are often used in artificial neural networks, among others [27].

2.4.1. Network with a single layer

Network with a single layer only has one layer connection weights. Single layer network consists of units of *input* that receives signals from the outside world, and the units of *output* where we can read the *response* from the network (See Figure 1).

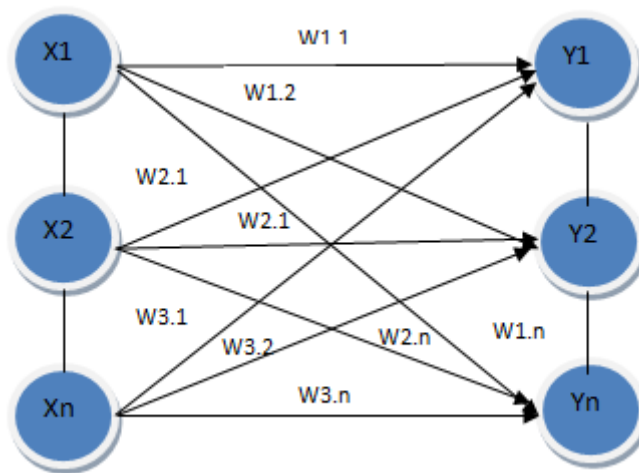


Figure 1: Single layer networks

2.4.2. Multilayer Network

Multilayer net is a network with one or more hidden layer. Multilayer net is in their ability to troubleshoot when compared with single-layer net, but the training may be more complicated (See Figure 2).

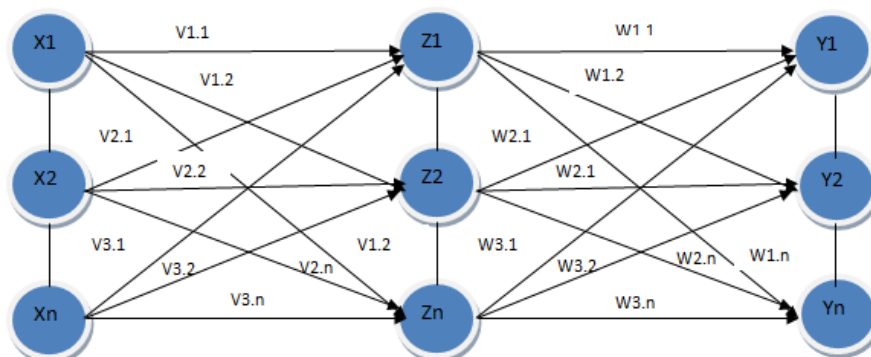


Figure 2: Network with many layers

2.4.3. Competitive Networks

In this kind of network, a set of neurons compete for the right to be active (See Figure 3).

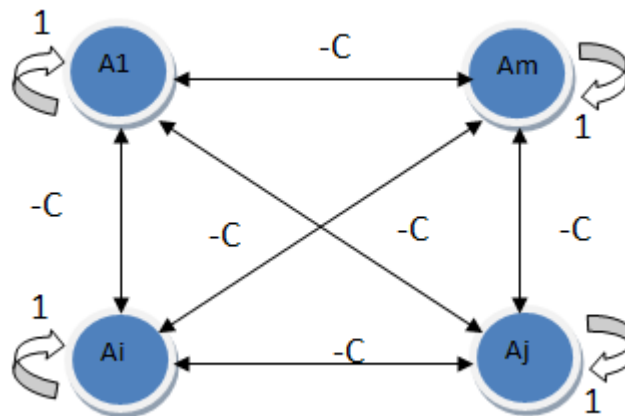


Figure 3: Competitive network layer

2.5. Method of Backpropagation

The back propagation algorithm cycles through two distinct passes, a forward pass followed by a backward pass through the layers of the network. The algorithm alternates between these passes several times as it scans the training data [26]. As with any model of *neural network* to another, propagation train the network to obtain a balance between the ability of the network to recognize patterns used during training as well as the network's ability to provide the correct response to the input pattern that is similar (but not equal) to the pattern used for training.

2.5.1. Architecture

Propagation neural network consists of many layers (multilayer neural network – See Figure 4):

- The input layer (1 piece). The input layer consists of neurons or units of inputs, starting from input 1 to input unit n .
- Hidden layer (minimum 1). Hidden layer consists of units hidden from unit 1 to unit hidden p .
- The output layer (1 piece). The output layer is composed of units of output from the output unit 1 to unit outputs m . The n, p, m respectively are arbitrary integers according to an artificial neural network architecture that is designed [27].

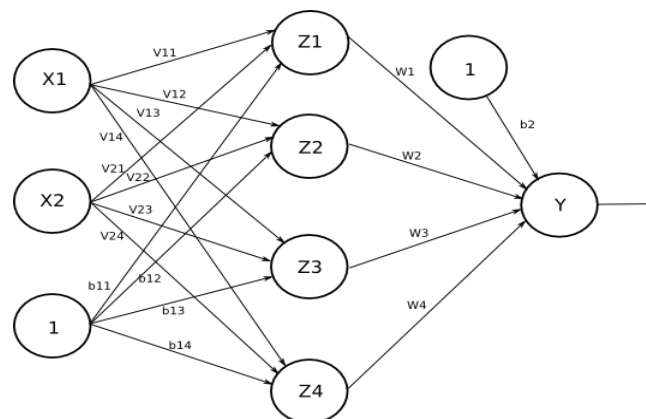


Figure 4: Architecture neural network backpropagation

2.5.2. Backpropagation

The overall goal of the algorithm is to attempt to find weights such that, for every input vector in the training set, the neural network yields an output vector closely matching the prescribed target vector.

- Step 1 Initialize the weights and set the learning rate and the stopping criteria.
- Step 2 Randomly choose an input and the corresponding target.
- Step 3 Compute the input to each layer and the output of the final layer.
- Step 4 Compute the sensitivity components.
- Step 5 Compute the gradient components and update the weights.
- Step 6 Check against the stopping criteria. Exit and return the weights or loop back to Step 2.

Artificial Neural Networks need to 'learn' first. How, in the network included a set of training examples called training set. Set the training is illustrated with a feature vector called input vector associated with an output target training. Training then carried out with the intention of making Neural Networks adapt to the characteristics of the examples in the training set by performing conversion late updating the weights in the network. Backpropagation algorithm can be divided into two parts:

Training algorithm

Training algorithm consists of three stages: the stage of advanced propagation, propagation phase reverse, and phase changes in weight.

Algorithm application

Algorithm application only uses feed-forward stage only [27]. Training algorithm to network with one hidden layer (with a binary sigmoid activation function) is given as follows:

- Step 0:** Initialize all weights with small random numbers.
- Step 1:** If the termination condition is not met, do steps 2-9.
- Step 2:** For each pair of training data, perform steps 3-8.

Phase I: Propagation forward

Step 3: Each input unit receives the signal and forward it to the hidden units on it.

Step 4: Calculate all output in a hidden unit ($j = 1, 2, \dots, p$).

$$z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}}$$

Step 5: Calculate all network output in the output unit ($k = 1, 2, \dots, m$).

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j w_{kj}$$

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}}$$

Phase II: Propagation backwards

Step 6: Calculate the error factor δ output unit based on the error in each unit of output ($k = 1, 2, \dots, m$).

$$\delta_k = (t_k - y_k)f'(y_{net_k}) = (t_k - y_k)y_k(1 - y_k)$$

An error unit will be used to change the weight of the layer beneath (step 7). Calculate the weight change rate (which will be used later to change the weights) with the accelerated pace of α .

$$\begin{aligned}\Delta w_{kj} &= \alpha \delta_k z_j ; \\ k &= 1, 2, \dots, m; \\ j &= 0, 1, \dots, p\end{aligned}$$

Step 7: Calculate the error factor δ hidden unit based on the error in each hidden unit ($j = 1, 2, \dots, p$).

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj}$$

Factors hidden units:

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j)$$

Calculate the weight change rate (which will be used later to change the weights)

$$\begin{aligned}\Delta v_{ji} &= \alpha \delta_j x_i \\ j &= 1, 2, \dots, p \\ i &= 0, 1, \dots, n\end{aligned}$$

Phase III: Changes in *weight*

Step 8: Calculate all the weight change

Weight change line leading to the unit output:

$$w_{kj}(\text{new}) = w_{kj}(\text{old}) + \Delta w_{kj}, \text{ for } k = 1, 2, \dots, m, \text{ and } j = 0, 1, \dots, p$$

Weight change line leading to the hidden units:

$$v_{ji}(\text{new}) = v_{ji}(\text{old}) + \Delta v_{ji}, \text{ for } j = 1, 2, \dots, p \text{ and } i = 0, 1, \dots, n$$

After training is completed, the network can be used for the introduction pattern. In this case, only the forward propagation (steps 4 and 5) are used to determine the output neural. The result activating units in the output layer is a decision of the neural network.

2.6. Signature

Signature is a special form of handwriting that contains special characters and additional forms are often used as proof of identity someone. Particularly, signature *verification* can be read, however many signatures that cannot be treaded. Therefore, signature of a person often changes anytime. This changed due to concerns on the position, size and signs of stress factors of signature. The factors on those changes are affected by time, age, habits and specific mental state.

3. Proposed Method

This section describes research steps. Figure 5 presents proposed research steps.

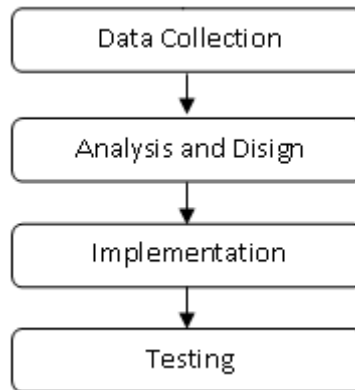


Figure 5: Research steps

3.1. Use Case Diagram

The use case diagram of this application can be seen in the Figure 6 as follow:

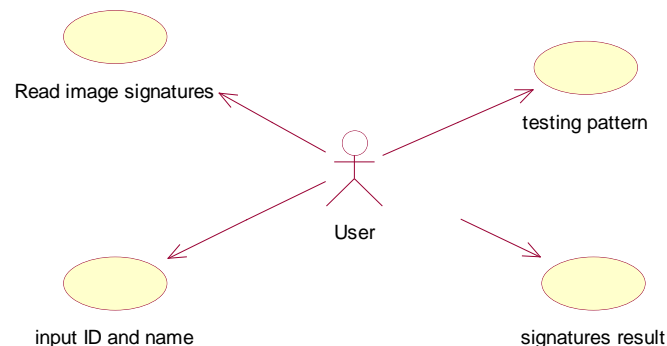


Figure 6: Use Case diagram

Use case diagram above illustrates how the actor, which in this case is the user, interact with the system described through the use case diagram.

3.2. Implementation

This stage aims to implement clicking the system design has been done in the previous stage in order to become a real system that can be used. This is done by clicking the draft code according to the method of implementation that been used. With the designed application system, the result can be operated and used optimally fit needs. In this paper, the implementation method used is Object-Oriented Programming using the Java programming language.

3.3. Testing

The testing process focuses on the internal logic of the software, ensuring that all statements have been tested, and the external functional, which directs the testing to find errors and ensure that the limited *input* will give the actual results that correspond to the required result. The program that has been built will be tested to determine whether the program is already running correctly and in

accordance with the design done. The program that has been built will be tested to compare the results of the application of Artificial Neural Networks in MATLAB version 6.1 results.

4. Results and Discussion

Analysis of the system is an early stage in the design of a system to be designed, at this stage will measure the performance of the designed system, the identification of the problems that exist and the measures to the needs of the design as expected. To analyze the system takes data from the system to the required analysis. Data are matters relating to the definition of data. The use of this propagation method is to improve the reliability and the ability to detect an object or a signature with a high recurrence rate. Signature recognition process conducted to assess the success of the character or pattern recognition based on the characteristics and classification character or a predetermined pattern. The recognition of this signature will be concluded whether a character or pattern determined already sufficiently reliable for recognition of signatures.

4.1. Data Analysis

In this study, the data used is signature. As for example, the input image of signature is given in Figure 7 as follow:



Figure 7: Example of input image signature

However, in the data analysis, the writer uses the example data taken just a simple signature squiggle. Writing signature paint or made in writing on a computer or a sheet of paper which is then removed with a scanner or digital camera. The collected data will be the training process and system testing. Both the training and testing data system used in the form of data signatures. Preliminary data taken from 10 sample signatures comprises of 6 signatures for training and 4 signatures for testing. The image of the handwriting process that has been established further processed by the floating process, because the process grayscale images should be done first. Grayscale process is performed to change the 24-bit RGB image into 8-bit grayscale image. Grayscale process will follow the following formula:

$$\text{GRAY} = (R + G + B) / 3$$

This process will produce a grayish image. After the grayscale process, then we perform the process of floating to differentiate text and background. If the specified threshold 140, then the pixel-1 were black because it has a grayscale value is smaller than the value of 140.

4.2. Feature Extraction

Feature extraction process in the training process aims to classify the data based on the input image and extract its value from the input image data as a key component in the process of beginning to use neural network training. In the process of signature recognition feature extraction section is only intended to extract value from an input image data has been segmented to Recognize the signature process. The introduction of a character or a pattern by using a neural network to recognize patterns method backpropagation signature is using 5×3 matrices. Here, an example of the process of floating and characteristic extraction process can be seen in Figure 8.

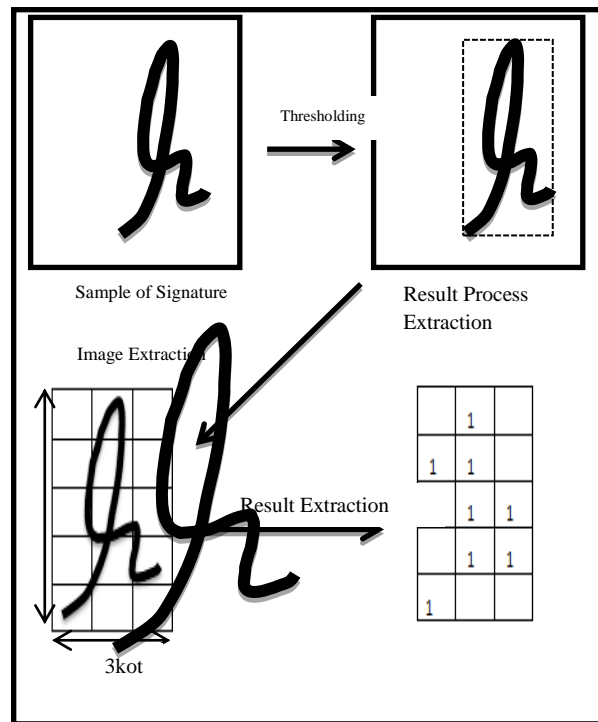


Figure 8:Process threshold and feature extraction process

4.3. Data input

Before the data processing is done, it needs to be processed in determining the input and the target or results to be obtained from the processing of data, which will facilitate the sharing of data and perform data processing to obtain results in line with what is expected. To be clearer about backpropagation training, consider the following example. For example, the extraction of characteristic signatures that will be trained is shown in Figure 9.

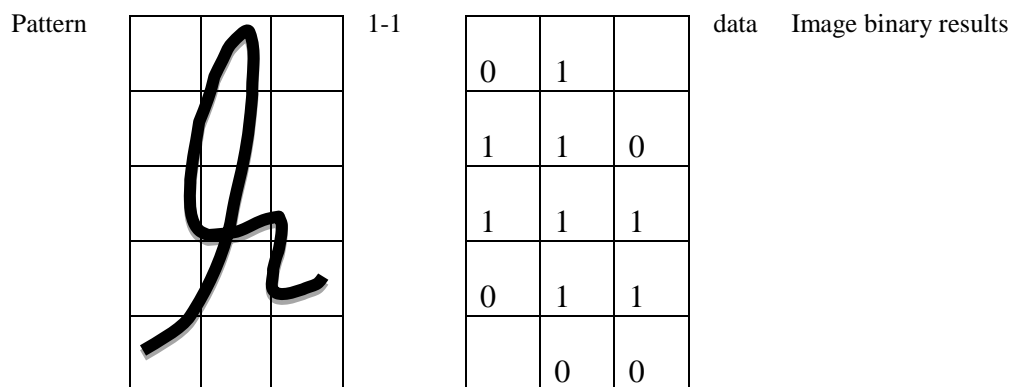


Figure9:Results of data extraction

4.4. Designing Network Architecture

On this issue Neural Network architecture that is used is a Neural Network with many layers (multilayer net – See Figure 10) with backpropagation algorithm, which consists of:

- a. The input layer is 15 (results of feature extraction in the form of 3x5 box)

- b. Hidden layer by the number of nodes specified by the user.
- c. The output layer (*output layer*) = 1
- d. *Learning Rate* (α) = 0.2

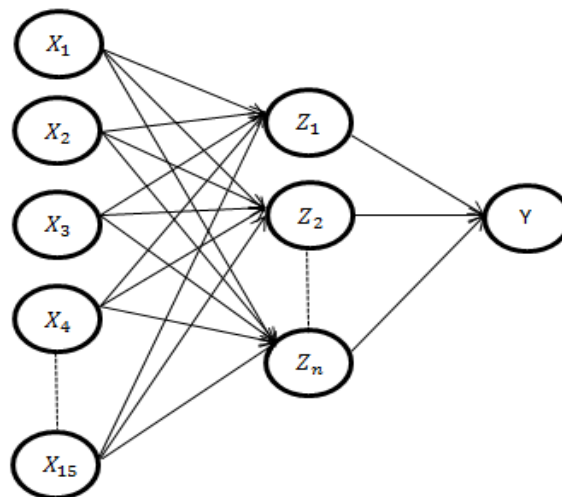


Figure 10: Neural network architecture network used

4.5. Training and Testing

The data will be processed using MATLAB with the architectural pattern 15-5-1, with 15 training data as input data exemplified signature with the signature code 01a as in Figure 11 below.

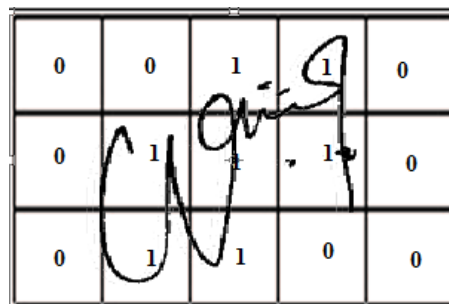


Figure 11: Signature 01a

To see the results issued by the network can use the following command.

```
>> [Y, Pf, Af, e, perf] = sim (net, p, [], [], t)
```

After looping do then found the minimum error or goal in the epoch 11 as in Figure 12 as follow.

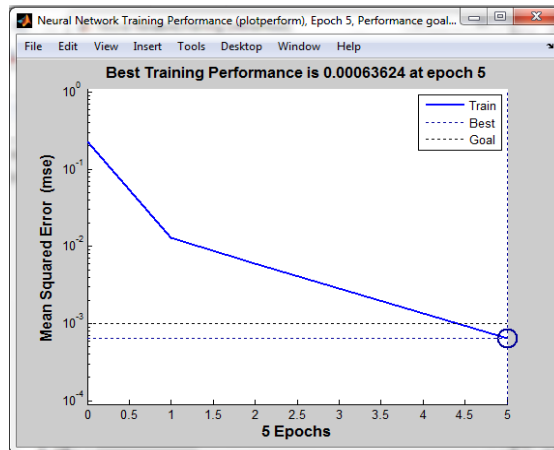


Figure 12: Results onsignature training 01a

In the test data will be conducted on the pattern of the signature with the signature code 02a. The steps of data processing are using the same MATLAB data training 0.2. The 15-5-1 with learning rate is the signature 02a which can be seen in Figure 13. Meanwhile, the chart goal against signature training 02a is presented in Figure 14.

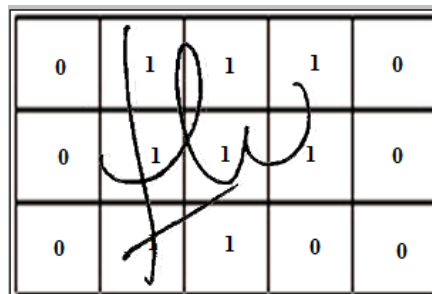


Figure 13:Signature 02a

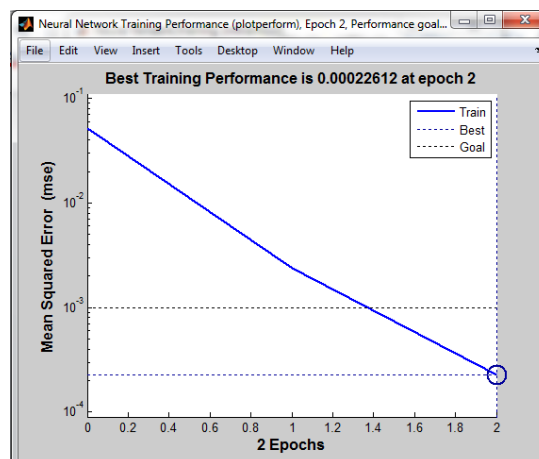


Figure 14:Chart goal against signature training 02a

Figure 15 as follow presents the results of the introduction of the signature with the pattern 15-5-1 of propagation method.

ID	Signature Code	Name	Learning Rates	Epoch	MSE	Target
01	01a	Widya Nursanty	0.01	106	0.00970	Valid
02	02a	Triana Welandi	0.01	101	0.000990	Valid
03	03a	Ningsih	0.01	83	0.000995	Valid
04	04a	Lini Astari	0.01	106	0.000970	Valid
05	05a	Teguh Kurnia	0.01	101	0.000990	Valid
06	06a	Fadhli Aziz	0.01	106	0.000970	Valid
07	07a	Alfikri Hardi	0.01	106	0.000970	Valid
08	08a	Rizki Ananda	0.01	106	0.000970	Valid
09	09a	Ceny Hartati	0.01	106	0.000970	Valid
010	010a	Halimah Tusa'diyah	0.01	99	0.000982	Valid

Figure 15: Result with pattern recognition signature 15-5-1

4.6. Implementation of Neural Network Application Program

As designed at the design stage, the display software is so executed as well as a display of training and recognition of signatures. *Input* the *id* owner and *id* picture will then appear the name, the target binary, photos on *id* identity and binary *inputs* and signature image on images, Display software soprogram run is shown in Figure 16 below.



Figure16: The display of signature recognition application

5. Conclusion

This paper studied implementation of signature recognition by using backpropagation of neural network. The artificial neural network with back propagation is applied as a method in the process of signature and pattern recognition to provide a solution that is best to analyze and recognize people's signature. Implementation of the application of neural networks in pattern recognition signature can further be applied to any computer that handles problems in the process of matching one's data. Neural network with backpropagation method that has been trained by people's signature image data can be used to identify the owner of the signature. The application comes with a photo of the owner of the signature that can be seen when entering the id the owner's identity, so as to reduce errors in fraudulent signatures. If the owner of the signature is different from the picture is not justified in doing custody signature. Hence, signature is something to do with the real owner.

Acknowledgement

The authors would like to thanks Universitas Putra Indonesia "YPTK" Padang for supporting this research.

References

- [1] Maltoni, D., Maio, D., Jain, A. and Prabhakar, S., 2009. *Handbook of fingerprint recognition*. Springer Science & Business Media.
- [2] Jain, A., Flynn, P. and Ross, A.A. eds., 2007. *Handbook of biometrics*. Springer Science & Business Media.
- [3] Tisse, C.L., Martin, L., Torres, L. and Robert, M., 2002, May. Person identification technique using human iris recognition. In *Proc. Vision Interface* (pp. 294-299).
- [4] Shah, H., Herawan, T., Ghazali, R., Naseem, R., Aziz, M.A. and Abawajy, J.H., 2014, November. An Improved Gbest Guided Artificial Bee Colony (IGGABC) Algorithm for Classification and Prediction Tasks. In *International Conference on Neural Information Processing* (pp. 559-569). Springer International Publishing.
- [5] Bakar, S.Z.A., Ghazali, R., Ismail, L.H., Herawan, T. and Lasisi, A., 2014. Implementation of Modified Cuckoo Search Algorithm on Functional Link Neural Network for Climate Change Prediction via Temperature and Ozone Data. In *Recent Advances on Soft Computing and Data Mining* (pp. 239-247). Springer International Publishing.
- [6] Shah, H., Herawan, T., Naseem, R. and Ghazali, R., 2014, October. Hybrid guided artificial bee colony algorithm for numerical function optimization. In *International Conference in Swarm Intelligence* (pp. 197-206). Springer International Publishing.
- [7] Shah, H., Ghazali, R., Nawi, N.M., Deris, M.M. and Herawan, T., 2013. Global artificial bee colony-Levenberg-Marquardt (GABC-LM) algorithm for classification. *International Journal of Applied Evolutionary Computation* (IJAEC), 4(3), pp.58-74.
- [8] Chiroma, H., Abdul-Kareem, S., Muaz, S.A., Khan, A., Sari, E.N. and Herawan, T., 2014, October. Neural Network Intelligent Learning Algorithm for Inter-related Energy Products Applications. In *International Conference in Swarm Intelligence* (pp. 284-293). Springer International Publishing.
- [9] Chiroma, H., Abdulkareem, S. and Herawan, T., 2015. Evolutionary Neural Network model for West Texas Intermediate crude oil price prediction. *Applied Energy*, 142, pp.266-273.
- [10] Nawi, N.M., Khan, A., Rehman, M.Z., Chiroma, H. and Herawan, T., 2015. Weight optimization in recurrent neural networks with hybrid metaheuristic Cuckoo search techniques for data classification. *Mathematical Problems in Engineering*, 2015.
- [11] Abubakar, A.I., Khan, A., Nawi, N.M., Rehman, M.Z., Wah, T.Y., Chiroma, H. and Herawan, T., 2016. Studying the Effect of Training Levenberg Marquardt Neural Network by Using Hybrid Meta-Heuristic Algorithms. *Journal of Computational and Theoretical Nanoscience*, 13(1), pp.450-460.
- [12] Nawi, N.M., Rehman, M.Z., Aziz, M.A., Herawan, T. and Abawajy, J.H., 2014, November. An Accelerated Particle Swarm Optimization Based Levenberg Marquardt Back Propagation Algorithm.

- In *International Conference on Neural Information Processing* (pp. 245-253). Springer International Publishing.
- [13] Riedmiller, M. and Braun, H., 1993. A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In *Neural Networks, 1993., IEEE International Conference On* (pp. 586-591). IEEE.
 - [14] Hecht-Nielsen, R., 1989, June. Theory of the backpropagation neural network. In *Neural Networks, 1989. IJCNN., International Joint Conference on* (pp. 593-605). IEEE.
 - [15] Sommerville, Ian. (2011). *Software Engineering Ninth Edition*. United States of America: Addison-Wesley.
 - [16] Bell, D. (2005). *Software Engineering For Students A Programming Approach*. England: Addison-Wesley
 - [17] Ram, K. (2002). *Short UML Reference*. University of Nebraska – Lincoln.
 - [18] Schuller, J. (2004). *Sams Teach Yourself UML in 24 Hours*. Sams Publishing The United States of America.
 - [19] Pratiwi, D., Santika, D.D. and Pardamean, B., 2013. *An application of backpropagation artificial neural network method for measuring the severity of Osteoarthritis*. arXiv preprint arXiv:1309.7522.
 - [20] Phillips, D. (2000), *age Processing in C*, R & D Publications
 - [21] Putra, D. (2010). *Pengolahan Citra Digital*. Yogyakarta: Ani Offset.
 - [22] McDermott, D. (2007). *Artificial Intelligence and Consciousness*. Yale University.
 - [23] Coppin, B. (2004), *Artificial Intelligence Illuminated*. Jones and Bartlett Publishers.
 - [24] Chowdhury, M. (2012). *Advantages and Limitations of Artificial Intelligence*. Clemson University.
 - [25] Mautner, P. (2010). *Segmentation EEG with Neural Network*. University of West Bohemia
 - [26] Ganatra, A. (2011). *Initial Classification Through Back Propagation In a Neural Network Following Optimization Through GA to Evaluate the Fitness of an Algorithm*. International Journal of Computer Science & Information Technology.
 - [27] Puspitaningrum, D. (2006). *Pengantar Jaringan Syaraf Tiruan*. Yogyakarta: Andi Offset.