IOP Conf. Series: Materials Science and Engineering **434** (2018) 012034 doi:10.1088/1757-899X/434/1/012034

Java character recognition using Hopfield network

W Widodo*, R A Priambodo and B P Adhi

Department of Informatics Education, Universitas Negeri Jakarta, Jl.Rawamangun Muka, Jakarta, Indonesia

*widodo@unj.ac.id

Abstract. Java character is a legacy in Indonesian culture. It is a character to express Java language for Java people. Java characters have high level of its complexity because it consists of three types, basic characters (carakan), numbers, and complementary characters (sandhangan). This research performed an experiment to recognize Java character using Hopfield network. Firstly, Java character image will be transformed into bitmap format, then we process the data using Hopfield network as an associative memory to recognize. The experimental result denoted that average accuracy is 88%. It is better than some other neural networks method used by other researchers.

1. Introduction

Java character is a character to express Java language which is used by Java people. It is a legacy of Java culture, but many Java people abandoned it. Recently, in information technology era, we can digitize this character so that the young generation of Java people will understand this culture legacy.

Unfortunately, study on Java character recognition is very rare. We found some studies related to Java character recognition. A study which is conducted by Budhi and Adipranata compared its recognition between evolutionary neural network and combination of chi2 and backpropagation [1]. The study result showed that combination of chi2 and backpropagation outperformed with 70% accuracy. Then, study concerned with java character is feature extraction [2]. This study focused on how to extract feature to obtain simple closed curve, straight lines and curve. To reduce the pixels they used skeletonizing. Research using associative memory compared with evolutionary neural network and counterpropagation has been conducted [3]. Its experiment lead to evolutionary neural network as the highest accuracy. Budhi tried to compare all his work to recognize Java character by using neural networks [4] and the best accuracy is combination of chi2 and backpropagation. For all type of character its accuracy is 73.71% and for basic character 65.75% respectively. The more advanced of this study is to develop Java word processor [5]. As mentioned above, the performed study did not try to use Hopfield network yet. It is a good method for recognizing traditional characters.

The rest of this paper is organized as follows. In section 2 we describe about Java characters. In section 3 we discuss some theories of Hopfield network. Section 4 we describe the methodology. Section 5 shows the experimental result, then paper ends in section 6 with conclusions.

2. Java characters

Java characters is derived character from Brahmi character. It was long time used in many Indonesia region, not only in Java but also Makassar, Melayu, Sunda, Bali, Sasak. Java characters have the basic characters called carakan [6] as shown in figure 1.

Figure 1. Basic Java characters (*carakan*) [6].

Java characters also have its own numbers which is shown in figure 2 in ascending order and *sandhangan* as a complementary character which is shown in figure 3. While s*andhangan* is a Java complementary characters and it can be used to express a vowel to complement *carakan*.

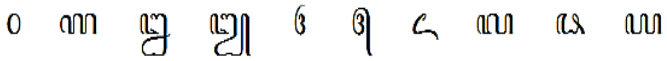

Figure 2. Number Java characters [6].

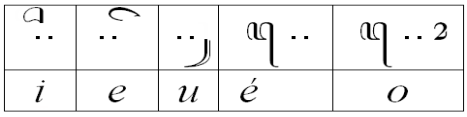

Figure 3. Number Java characters (*sandhangan*).

3. Hopfield network

In this section we will describe Hopfield network, associative memory, Hebbian Learning, and Hopfield network as an associative memory.

3.1. Hopfield

Hopfield Network is a method in Neural Network which uses combination of feed-forward and feedback. When a feedback is entering its network, stability of it cannot be guaranteed. By using Hopfield network, it must consider how to control network stability. Although Hopfield is a single layer but its structure as a feedback model could make it behave like a multilayer [7]. It is obviously seen in figure 4.

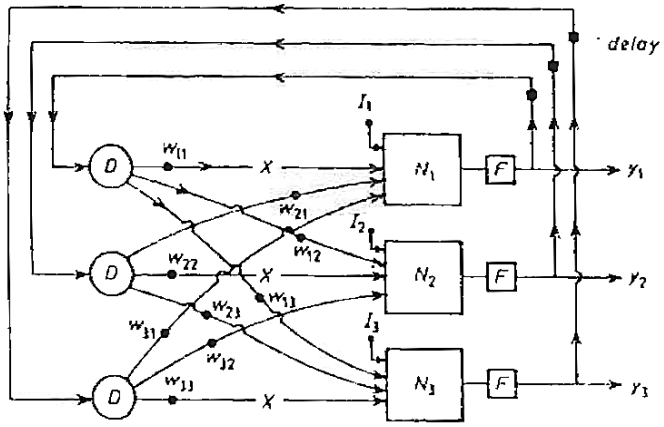

Figure 4. Structure of Hopfield network [7].

3.2. Associative memory

Associative memory is distributed memory that alike brain and learning by association [8]. Associative memory can map input - which concerns with its value – to a stored information in memory. We can implement associative memory by using recurrent networks and choosing proper weight to have a stable output. As a recurrent network, this stable output become an input [9].

3.3. Hebbian learning

A Hebbian learning rule is a learning rule in neural networks that determines the weight of connection between two units whether it should be increased or decreased to the product of their activation. Hebbian learning uses learning rate to activate neurons simultaneously to strengthen them [10]. Four properties of Hebbian learning are Time-dependent mechanism, Local mechanism, Interactive mechanism and Conjunctional mechanism [8].

3.4. Hopfield network as an associative memory

There are four steps in Hopfield network [10]:

a. Using Hebb learning, set of input $x_1, x_2, \ldots, x_m$ are contributed from N-dimensional fundamental memory. Synaptic weight can be obtained from formula (1) bellow:

$$W = \frac{1}{N}\sum_{r=1}^{M} x_r x_r^T - \frac{M}{N} I \quad (1)$$

b. Initialize xp vector for testing. The formula (2) is shown as follow:

$$x_i(0) = x_{ip}, \quad (2)$$

$$i = 1,2,\ldots,N$$

where $x_i(0)$ is i^{th} neuron when $n=0$, x_{ip} is i^{th} element from vector x_p, and N is number of neurons.

c. Do iteration with asynchronously adjustment using rule as shown below:

$$x_i(n+1) = hsgn\left(\sum_{j=1}^{N} w_{ij} x_j(n)\right), \quad i = 1,2,\ldots,N$$

$$hsgn(v_i(n+1)) \begin{cases} 1, & v_i(n+1) > 0 \\ v_i(n), & v_i(n+1) = 0 \\ -1 & v_i(n+1) < 0 \end{cases}$$

$$v_i(n+1) = \sum_{j=1}^{N} w_{ij} x_j(n)$$

d. Result, when vector x in stable condition.

4. Methodology

We used Hopfield network for recognizing Java characters. First, we collected the data. Then, the Java characters transformed from TTF format into *bitmap* format (BMP). The characters are combined in two combination and each character is put noises. Then we performed Hopfield network to obtain pattern. We used Hebbian learning to construct Hopfield network as an associative memory.

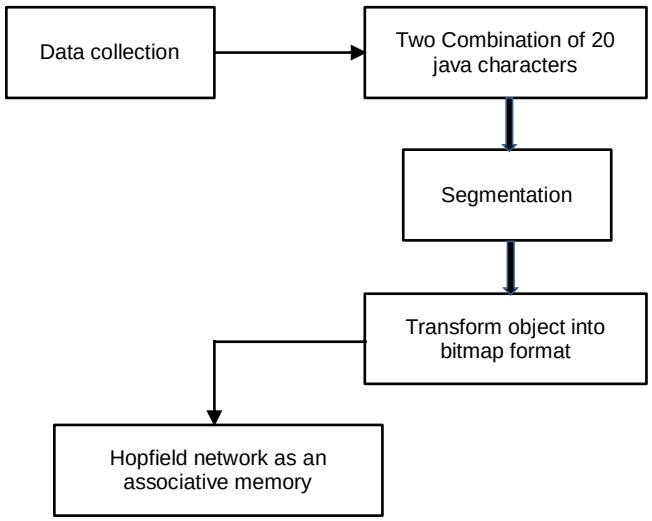

Figure 5. The methodology using Hopfield network.

5. Experiment result

We used combination of two Java characters for testing the performance of Hopfield network. From the characters combination, we obtained 190 combinations and put noise into them. Matlab function, *imnoise*, is employed to get noise *gaussian* is used also.

Table 1. Experiment result.

Noise	Result
0	100%
0.4	100%
0.6	99%
0.8	98%
1	96%
1.5	92%
2	87%
2.5	83%
3	81%
3.5	79%
4	78%
4.5	76%
5	74%

The experimental results obviously seen from table 1 that without noise Hopfield reached outstanding performance in recognizing Java characters. When it is put noise, its accuracy decreased as noise set to be increased. By using matlab embedded function *imnoise*, we put maximum noise at 5 since at that point the character is getting more blurred and hard to be seen obviously.

The average accuracy of our experiment was 88%. It outperformed some research which was conducted by researchers as being mentioned in section 1.

6. Conclusion

In this research we conducted an experimental research to recognize Java characters. Java characters have high level of complexity since its character consists of three types which is basic characters (*carakan*), numbers, and complementary characters (*sandhangan*). We used one of a neural network method, Hopfield network and Hebbian learning to perform this research. Its average accuracy is 88%

and it is denoted that this work outperformed some research which was conducted by some other researchers which used other neural network methods.

References

[1] G S Budhi and R Adipranata 2014 Java Character Recognition using Evolutionary Neural Network and Combination of Chi2 and Backpropagation *Neural Network International Journal of Applied Engineering Research* **9**(22) pp.15395-15406
[2] R Adipranata, Liliana, M Indrawijaya and G S Budhi 2015 Feature Extraction for Java Character Recognition In : Intan R. Chi C.H., Palit H, Santoso L. (eds) Intelligence in the Era of Big Data *ICSIIT Communication in Computer and Information Science* **516** (Springer, Berlin)
[3] G S Budhi and R Adipranata 2014 Comparison of Bidirectional Associative Memory, Counterpropagation and Evolutionary Neural Network for Java Characters Recognition. *International Conference of Advanced Informatics: Concepts Theory and Applications (ICAICTA)*
[4] G S Budhi and R Adipranata 2015 Handwritten Javanese Character Recognition using Several Artificial Neural Network Methods *Journal of ICT Research and Applications* **8**(3)
[5] R Adipranata, G S Budhi and R Thedjakusuma 2012 Java Characters Word Processing *International Conference on Soft Computing, Intelligent System and Information Technology*
[6] Gregorius Satia Budhi and Rudy Adipranata 2014 Java Characters Recognition using Evolutionary Neural Network and Combination of Chi2 and Backpropagation Neural Network *International Journal of Applied Engineering Research* **9**(22)
[7] D Graupe 2013 Principles of Artificial Neural Networks. 3rd edition *World Scientific Publishing*
[8] S Haykin 2009 *Neural Networks and Learning Machines 3rd edition* (Pearson)
[9] T Munakata 2008 *Fundamentals of the New Artificial Intelligence 2nd edition* (Springer, Verlaag, London)
[10] S Lynch 2004 *Dynamical Systems with Application using MATLAB* (New York: Springer Science Bussiness Media)