Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 6, No. 2, May 2021, Pp. 117-126

Diagonal based feature extraction and backpropagation neural network in handwritten batak toba characters recognition

Elviawaty Muisa Zamzami*[1], Septi Hayanti[2], Erna Budhiarti Nababan[3]
Universitas Sumatera Utara, Indonesia[1,2,3]

Article Info

Keywords:
Character Recognition, Feature Extraction, Backpropagation Neural Network, Diagonal Based Feature Extraction

Article history:
Received: January 27, 2021
Accepted: February 16, 2021
Published: May 31, 2021

Cite:
Zamzami, E. M., Hayanti, S., & Nababan, E. B. (2021). Diagonal Based Feature Extraction and Backpropagation Neural Network in Handwritten Batak Toba Characters Recognition. Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control, 6(2). https://doi.org/10.22219/kinetik.v6i2.1212

*Corresponding author.
Elviawaty Muisa Zamzami
E-mail address:
elvi_zamzami@usu.ac.id

Abstract

Handwritten character recognition is considered a complex problem since one's handwritten character has its characteristics. Data used for this research was a photo of handwritten or scanned handwritten. In this research, Backpropagation Neural Network (BPNN) was used to recognize handwritten Batak Toba character, wherein preprocessing stage feature extraction was done using Diagonal Based Feature Extraction (DBFE) to obtain feature value. Furthermore, the feature value will be used as an input to BPNN. The total number of data used was190 data, where 114 data was used for the training process and another 76 data was used for testing. From the testing process carried out, the accuracy obtained was 87,19 %.

1. Introduction

Handwritten recognition is considered a complex problem since one's character has its characteristics. Computer technology is widely applied in various aspects of human activities to solve particular problems. Handwritten recognition using computer technology can be used to digitalization the cultural heritage form and information [1]. Many researchers research handwritten recognition relate to cultural heritage scripts. Indonesia has many cultural heritage scripts so that can be the problems or objects of research. There are researches on the recognition of Lampung characters [2][3], Sundanese scripts [4][5], Balinese scripts [6], and Lontara characters [7]. This paper discusses the Batak Toba script, that one of the classifications of the Batak script.

This paper discusses how to recognize the handwritten character of Batak Toba script using Backpropagation Neural Network *(*BPNN) and *Diagonal Based Feature Extraction (DBFE)* on the preprocessing stage. The character of the Batak Toba script has its uniqueness where the transcript is in semi-syllabic that consists of 19 letters, called as *ina ni surat* Figure 1.

Figure 1. The Ina Ni Surat [8]

In general, handwritten recognition can be done using online and offline methods [9][10][11]. In the online method, the two-dimensional coordinates of writing points are represented as a function of time, and the order of each line written is also stored in a real-time manner to recognize the characters written. On the other hand, the offline method automatically converts the writing on an image into characters that can be processed by computers and text processing applications [12].

Some previous research done regarding handwritten Batak Toba character recognition was done by Sitinjak [13]. They employed the Wavelet method in pre-processing stage and apply Backpropagation to classify the handwritten. The system got 100% of memory capability and 94.74% for data generalization capability that has not been trained with MSE 2.319937 x 10-8. Latin Handwriting Recognition was done by Khairunisa [14]. The zoning extraction method with an image size of 30x40 pixels divided into 48 zones with the size of each zone is 5x5 pixels. The extracted feature values are in the form of binary values, namely 0 and 1. The recognition rate obtained was 83.85%.

Furthermore, Putra [15] employed BPNN in handwritten number recognition using zoning and DBFE methods with an image size of 60x90 pixels divided into 54 zones each zone was in 10x10 pixels. The recognition rate obtained was 87%. In Pasaribu and Hasugian research [16] applied Combination of Data using Euclidean Distance (CoDED) method and statistical feature extraction and Elliptic Fourier to recognize handwritten Batak Toba, characters, however, the result was not satisfying. In the other research, Pasaribu and Hasugian [17] use the artificial neural network to remove the background noise of the handwritten script of Batak Toba. Romulus, et al [18] do research to convert the random Batak symbols in an image into Latin characters representation. Their research has a system accuracy between 42-96 % with a processing time is 1.9-34 seconds.

Afroge, et al [19] developed a Latin-character recognition using a feed-forward neural network that was intended to reduce the impact of noise in the image. The system was able to achieve 99% accuracy for numeric digits (0-9) and 97% and 96% accuracy for the uppercase letter (A-Z) and lowercase letter (a-z) respectively. The system achieved a 93% accuracy for alphanumeric characters. However, based on the test, the authors recognized that a specific type of noise contributed a significant decrement on the performance.

A solution in character recognition proposed by Zanwar, et al. [20] uses feature extraction before feeding the data into a backpropagation neural network system. Similar to the previously mentioned research, the data used in this test was handwritten English alphabets, with various shapes and pattern, selected from the Chars74K library. The research shows that by feeding the feature/pattern of the image (instead of using raw images), the system is capable to increase its performance to recognize the character.

Suryadibrata and Chandra [21] developed an Optical Character Recognition (OCR) system using a backpropagation neural network (BPNN) based as its method. The research was aimed to detect characters from printed documents. The first step of the system is to extract images of each character from the document to be fed to the neural network. This extraction process is done in 3 steps; line extraction, word extraction, and letter extraction). These images (of a single character) are then processed by the BPNN. The proposed system was able to achieve over 90% accuracy, based on the font used in the document eg: 94% for MS Arial Unicode and Times New Roman, 96.6% for Tahoma.

Applying the LVQ method to the Batak manuscript was done by Muchtar, et al [22]. In their recognition process, using the image in jpg format as input. Recognizing the Batak document (as an image), use preprocessing including grayscale, contrast, thresholding, and segmentation. Digitizing Batak manuscript using the LVQ method has an accuracy rate of 97.9%.

This paper discusses DBFE and BPNN implementation to recognize handwritten Batak Toba characters. Also, this paper shows the accuracy of the methods used in tests on the handwritten Batak Toba characters.

2. Research Method

The proposed method of recognizing the handwritten character of Batak Toba is depicted in Figure 2.

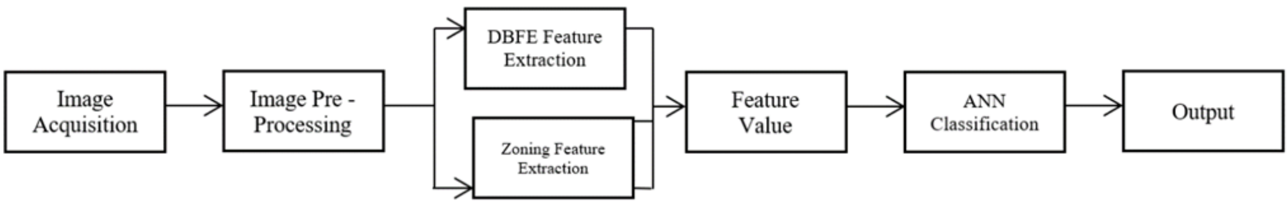

Figure 2. Batak Toba Handwritten Character Recognition

2.1 Image Acquisition

The input pass to the artificial neural network in the form of numeric data sets or binary data. Therefore, an image needs to be converted into a numeric data set or binary data before processing. The image acquisition can be using the scanner, digital camera, etc [23]. Sample data was collected from ten USU Information Technology students. Each student writes the Batak Toba characters from 'a' to 'u' as shown in Figure 1. The scanned image is saved in the Bmp format.

The total sample data is 190 handwritten Batak Toba characters. From the total data, 114 characters (6 patterns) are used for training data, another 76 (4 patterns) are used for testing.

2.2 Image Pre-processing

Image pre-processing is described in Figure 3. There are binarization, normalization, and thinning processes.

Figure 3. Pre-processing

a. Binarization

Image binarization is a process in preprocessing to improve the readability of images of handwritten characters [24]. Binarization is used to reduce unwanted information of the image and protect useful information [25]. This process will produce a clean black and white image from the gray level (grayscale) [26], or in other words, this method inverts the gray-level image to a bi-level image (binary image). At this stage, the average value of each RGB pixel will be taken and then checked, if the resulting value is less than the resulting threshold value then the pixel value is converted to black, otherwise, if it is greater than a constant value it will be converted to white. The binary matrix of this image is formed based on the black and white value in the image that has been obtained, if the image pixel at the coordinates (x, y) is black then the binary matrix value in row i and column j is 1, otherwise = 0. See Figure 4.

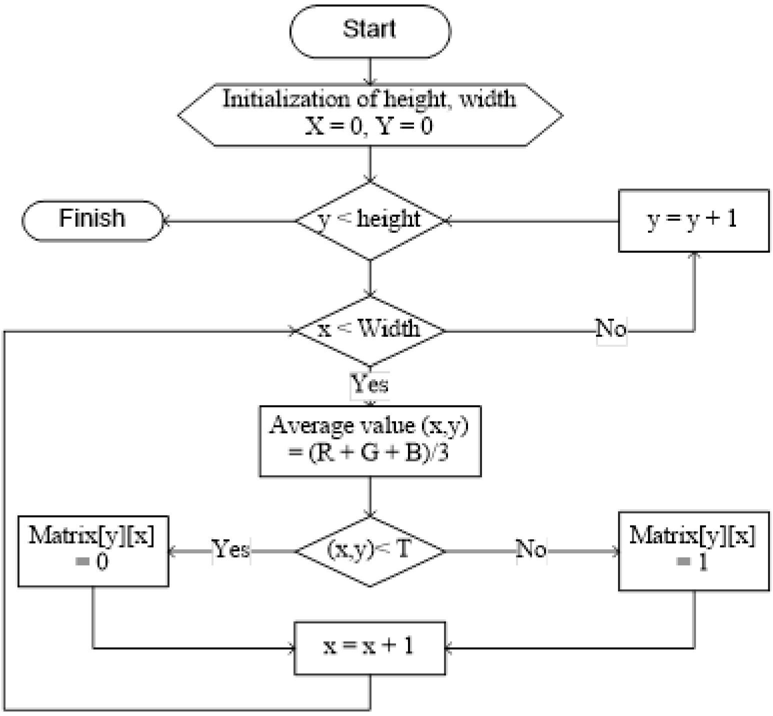

Figure 4. Flowchart of Binary Matrix Formation

b. Normalization

Crop images have different resolutions so they cannot be used as standard input for extracting. The image must be normalized, namely changing the image resolution to a resolution suitable for extracting, namely 60x90 pixels. An example of a normalized cropped image can be seen in Figure 5.

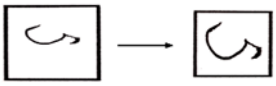

Figure 5. Normalization

c. Thinning

The next process is thinning. Thinning is one of the most used pre-processing that roles to analyze the image in handwritten character recognition [27]. The Batak Toba script handwritten objects contained in the image will be thinning until the thickness is only 1 pixel but does not change the important information and characteristics of the object. Through this management process, a framework will be obtained from the object of the Batak Tobascript handwriting. Recognition of objects such as letters or silhouettes will be easier by paying attention to the outline [28]. Examples of thinning images can be seen in Figure 6.

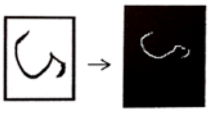

Figure 6. Image Thinning Result

2.3 Feature Extraction

In this study, feature extraction was carried out using a combination of zoning and diagonal based feature extraction methods. The two methods will be carried out sequentially, first zoning and secondly diagonal based feature extraction (DBFE). The feature values obtained from the two methods will be combined, namely connecting the feature values obtained from the zoning method with diagonal based feature extraction. The feature values of the two methods are stored in the same matrix. The feature values obtained from the zoning method are stored from index 0 to n and feature values obtained from the diagonal based feature extraction method are stored from index n + 1 to index m. The feature value matrix can be seen in Figure 7.

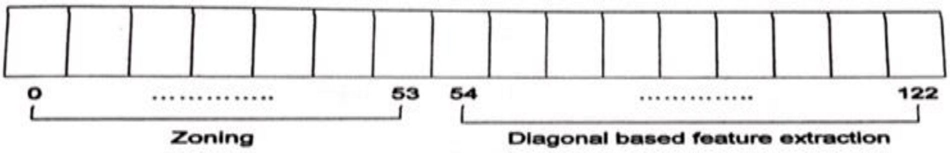

Figure 7. Feature Value Storage Matrix

After the image pre-processing stage is carried out, uniform sample data is obtained. The size of the sample data from the pre-processing image is 60x90 pixels. This measure follows the measurement used by Pradeep et. al [29]. In their research, each sample of data was divided into zones with a size of 10x10 pixels. From the division of the zones, 6 columns and 9 rows of zones are obtained. The number of zones is 54 zones. Each zone will be processed to get the feature value. The same zoning was also carried out in this study on the combination of extraction methods used. The feature extraction process using a combination of zoning and diagonal based feature extraction methods can be explained through the diagram in Figure 8.

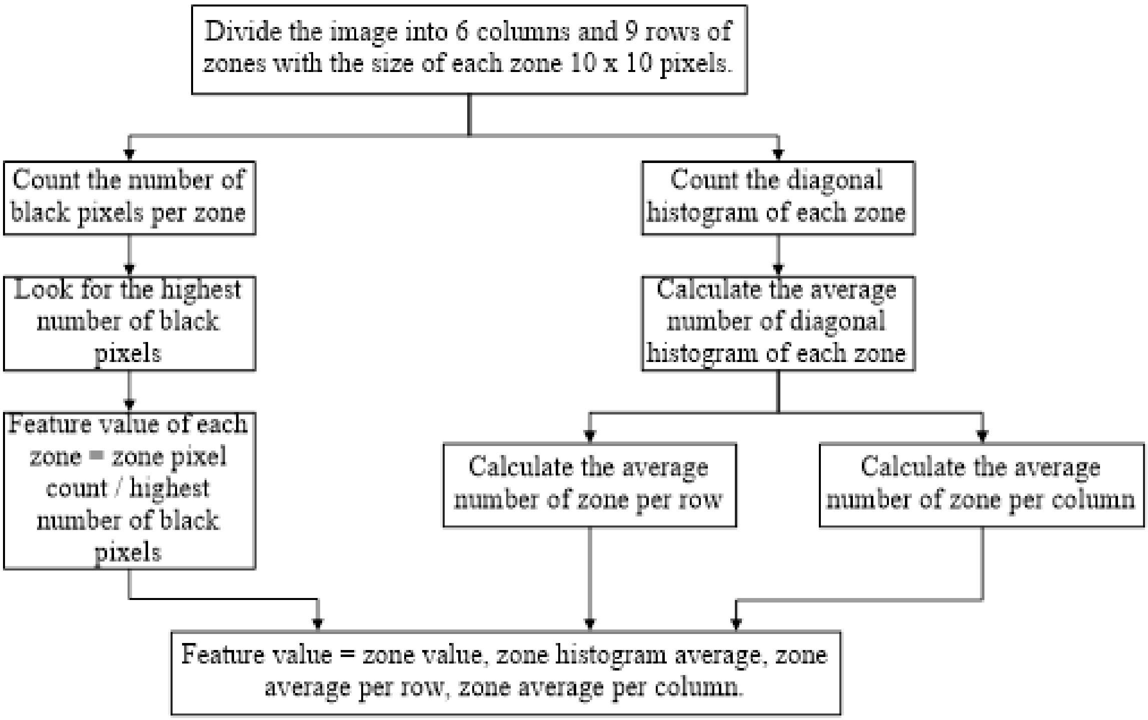

Figure 8. Feature Extraction Diagram [15]

From Figure 8 the left side process is the zoning method and the right side process is the diagonal based feature extraction method. The two extraction methods are:

a. Zoning Extraction Method

The zoning method consists of three processes, namely:

1. Count the number of black pixels for each zone from Zi to Z54.
   For example, Z1 = 5, Z10 = 10 and Z15 3.
2. Determine the zone that has the highest number of black pixels.
   For example, from the example in step 1, the zone with the highest number of pixels is Z10, which is 10 pixels.
3. Calculate the feature value for each zone from ZI to Z54.
   Namely using the formula: The highest feature value Zn = Zn/Z where ≤ n ≤ 54.

From zoning extraction obtained 54 as the feature value that represents each zone.

b. Diagonal Based Feature Extraction Method

This method consists of 4 processes, namely:

1. Calculate the diagonal histogram for each zone. The diagonal histogram is the number of black pixels per diagonal. The diagonal histogram calculation for each zone is performed as shown in Figure 9.

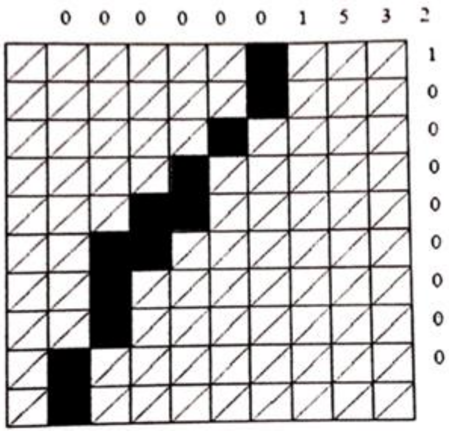

Figure 9. Histogram Diagonal Zone [29]

   The number of diagonal histograms for a zone is:
   Number of diagonal histograms = Zone length + Zone height - 1
   Many diagonal histograms = 10 + 10 -1 = 19
2. Calculate the average of the histograms for each zone.
   Feature value Zn = Average histogram = (Histl + .. + Hist19) / 19
   where 1 ≤ n ≤ 54
3. Calculate the zone average for each row, where the number of zones per row / 6
4. Calculate the zone average for each column, where the number of zones per column / 9.

From the diagonal based feature extraction method, it is found that 54 diagonal histogram average feature values for each zone, 9 zone average feature values for each row, and 6 zone average feature values for each column. The total feature value obtained from this method is 69 feature values. From zoning extraction obtained 54 as the feature value that represents each zone.

2.4 After Phase Feature Extraction

After the feature extraction stage is carried out, the next stage can be carried out, namely the classification using the backpropagation network. At this stage, the feature values obtained from the feature extraction stage are used as input for the backpropagation network input layer. The classification stage consists of two processes, namely the training process and the testing process. At the training stage, network training is carried out using the feature values obtained from the training data. The network must be trained first so that it can then be used. After the training stage, the backpropagation network can be used for the testing phase using the feature values obtained from the test data. Before training and testing can be carried out, the network must be designed first.

Based on the number of feature values obtained from the feature extraction stage, namely 123, the number of neurons in the backpropagation network input layer is 123 neurons. The number of neurons in the output layer is 19 Batak Toba characters. The output value of each number can be seen in Table 1.

Table 1. Output Value and Network Output Target

Output Value	Target
0000000000000000001	A
0000000000000000010	Ha
0000000000000000100	Ma
0000000000000001000	Na
0000000000000010000	Ra
0000000000000100000	Ta
0000000000001000000	Sa
0000000000010000000	Da
0000000000100000000	Ga
0000000001000000000	Ja
0000000010000000000	Ba
0000000100000000000	Nga
0000001000000000000	La
0000010000000000000	Pa
0000100000000000000	Nya
0001000000000000000	Wa
0010000000000000000	Ya
0100000000000000000	I
1000000000000000000	U

Based on the design of the artificial neural network built in Khairunisa [14] and Putra [15] studies, it can be seen in Figure 10.

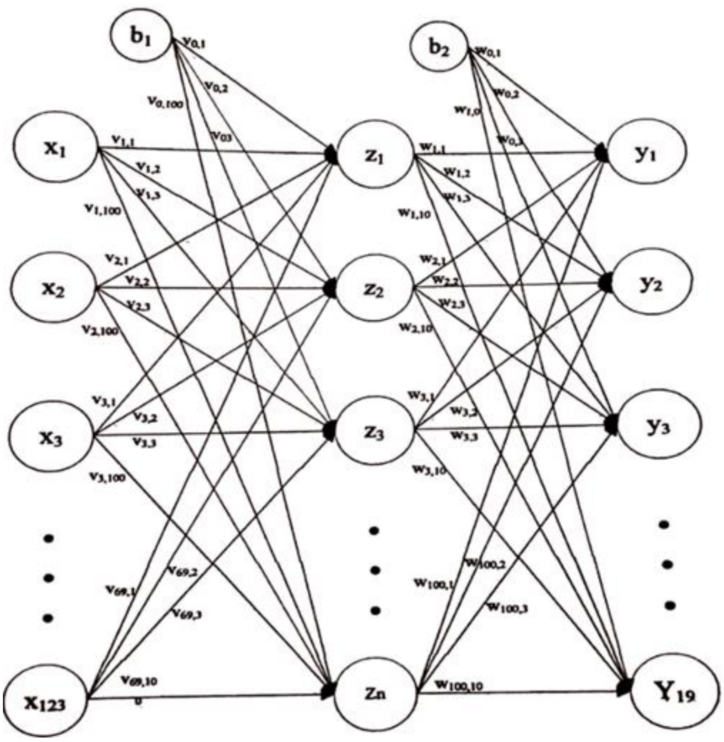

Figure 10. Artificial Neural Network

Based on Figure 10, the details of the artificial neural network architecture design used in this research are:

1. 1 input layer. The input layer consists of input units starting from input unit 1 to input unit i, where *i* = 123.
2. 1 hidden layer. The hidden layer consists of hidden units 1 to hidden units consisting of hidden units j, where *j* = 96.
3. 1 output layer. The output layer consists of output units ranging from output units 1 to output units k, where *k* = 19.

4. xl to x123 is the input layer units, y*i* to y19 are the output layer units, and z1 to z96 are hidden layer units.
5. is bias to the hidden layer and by is bias to the output layer.
6. v0j and wok are the bias weights for the j-*th* hidden unit and the k-*th* hidden unit of output, respectively. viy is the connection weight between the i-*th* unit of the input layer and the j-*th* unit of the hidden layer, while wjk is the weight of the connection between the j-*th* unit of the hidden layer and the k-unit of the output layer.

a. Training Process

The training process is carried out to train the network to recognize the handwriting of the Batak Toba script, whether trained or not. Each training is carried out using predetermined parameters. Backpropagation network training consists of three stages, namely feed-forward, error propagation, and correction of weight values. The three stages are continued until the network output error is less than the error tolerance or the maximum epoch is reached. The last weight obtained from the training process is used in the testing process.

b. Testing Process

The testing process is testing the network that has been trained to recognize unused test data at the network training stage. Using the weights obtained after the training process, the network is expected to be able to recognize writing using the feed-forward stage which is used to find the hidden layer and output layer output values.

3. Results and Discussion

The training process on BPNN aims to train the network the handwritten characters of Batak Toba so that the network could recognize other characters that have not been trained yet. In the testing stage, will be used 76 characters. The result of the training and testing were done, is shown in Table 2.

Table 2. Result of Training on Handwritten Characters using BPNN

Input	Learning Rate	Momentum	Epoch	Output	Compatibility (%)
A	0.9	0.1	10000	Ya	74.97
A2	0.5	0.2	8000	Ya	73.92
A3	0.4	0.3	1000	A	92.96
A4	0.3	0.4	4000	A	93.10
A5	0.2	0.5	2000	A	93.44
A6	0.1	0.7	1000	A	93.57
Ha	0.9	0.1	10000	Ha	79.97
Ha2	0.5	0.2	8000	Ha	82.92
Ha3	0.4	0.3	1000	Ha	84.95
Ha4	0.3	0.4	4000	Ha	84.09
Ha5	0.2	0.5	2000	Ha	80.40
Ha6	0.1	0.7	1000	Ha	88.57
Ma	0.9	0.1	10000	Ma	92.97
Ma2	0.5	0.2	8000	Ma	92.92
Ma3	0.4	0.3	1000	Ma	92.96
Ma4	0.3	0.4	4000	Ma	93.10
Ma5	0.2	0.5	2000	Ma	93.47
Ma6	0.1	0.7	1000	Ma	93.60
Na	0.9	0.1	10000	Na	89.97
Na2	0.5	0.2	8000	Na	89.92
Na3	0.4	0.3	1000	Na	81.96
Na4	0.3	0.4	4000	Na	80.11
Na5	0.2	0.5	2000	Na	85.40
Na6	0.1	0.7	1000	Na	84.61
Ra	0.9	0.1	10000	Ra	82.97
Ra2	0.5	0.2	8000	Ra	83.92
Ra3	0.4	0.3	1000	Ra	83.96
Ra4	0.3	0.4	4000	Ra	86.08
Ra5	0.2	0.5	2000	Ra	89.43
Ra6	0.1	0.7	1000	Ra	89.56
⋮	⋮	⋮	⋮	⋮	⋮
Da	0.9	0.1	10000	Da	93.97
Da2	0.5	0.2	8000	Da	87.92
Da3	0.4	0.3	1000	Da	93.96
Da4	0.3	0.4	4000	Da	93.10
Da5	0.2	0.5	2000	Da	84.53
Da6	0.1	0.7	1000	Da	80.54
Ga	0.9	0.1	10000	Ga	79.98

Ga2	0.5	0.2	8000	Ga	79.92
Ga3	0.4	0.3	1000	Ga	79.99
Ga4	0.3	0.4	4000	Ga	80.10
Ga5	0.2	0.5	2000	Ga	80.45
Ga6	0.1	0.7	1000	Ga	80.60
Ja	0.9	0.1	10000	Ja	79.98
Ja2	0.5	0.2	8000	Ja	75.92
Ja3	0.4	0.3	1000	Ja	76.96
Ja4	0.3	0.4	4000	Ja	75.09
Ja5	0.2	0.5	2000	Ja	80.47
Ja6	0.1	0.7	1000	Ja	81.55
⋮	⋮	⋮	⋮	⋮	⋮
Nga3	0.4	0.3	1000	Nga	78.96
Nga4	0.3	0.4	4000	Nga	79.10
Nga5	0.2	0.5	2000	Nga	79.46
Nga6	0.1	0.7	1000	Nga	79.58
La	0.9	0.1	10000	La	78.98
La2	0.5	0.2	8000	La	76.92
La3	0.4	0.3	1000	La	89.02
La4	0.3	0.4	4000	La	76.10
La5	0.2	0.5	2000	La	80.44
La6	0.1	0.7	1000	La	81.55
Pa	0.9	0.1	10000	Pa	78.89
Pa2	0.5	0.2	8000	Pa	92.92
Pa3	0.4	0.3	1000	Pa	78.96
⋮	⋮	⋮	⋮	⋮	⋮
Ya4	0.3	0.4	4000	Ya	83.09
Ya5	0.2	0.5	2000	Ya	85.45
Ya6	0.1	0.7	1000	Ya	81.57
I	0.9	0.1	10000	I	75.98
I2	0.5	0.2	8000	I	76.92
I3	0.4	0.3	1000	I	79.96
I4	0.3	0.4	4000	I	81.11
I5	0.2	0.5	2000	I	73.47
I6	0.1	0.7	1000	I	75.55
U	0.9	0.1	10000	U	82.98
U2	0.5	0.2	8000	U	76.93
U3	0.4	0.3	1000	U	72.97
U4	0.3	0.4	4000	U	77.09
U5	0.2	0.5	2000	U	77.45
U6	0.1	0.7	1000	U	77.54

After doing training, the next stage is to test the network using 76 characters with the value of learning rate is set to 0.1 a momentum value is 0.7 and Epoch is 1000. The result of the testing process is described in Table 3.

Table 3. Result of Testing on Handwritten Characters using BPNN

Input	Learning Rate	Momentum	Epoch	Output	Compatibility (%)
A7	0.1	07	1000	A	99.45
A8	0.1	07	1000	A	99.44
A9	0.1	07	1000	A	99.42
A10	0.1	07	1000	A	94.05
Ha7	0.1	07	1000	Ha	83.08
Ha8	0.1	07	1000	Ha	76.16
Ha9	0.1	07	1000	Ha	84.25
Ha10	0.1	07	1000	Ha	86.02
Ma7	0.1	07	1000	Ma	94.37
Ma8	0.1	07	1000	Ma	94.03
Ma9	0.1	07	1000	Ma	93.99
Ma10	0.1	07	1000	Ma	94.03
⋮	⋮	⋮	⋮	⋮	⋮
Ra7	0.1	07	1000	Ra	85.45
Ra8	0.1	07	1000	Ra	85.06
Ra9	0.1	07	1000	Ra	88.09
Ra10	0.1	07	1000	Ra	87.99

Da7	0.1	07	1000	Da	74.43
Da8	0.1	07	1000	Da	84.22
Da9	0.1	07	1000	Da	88.10
Da10	0.1	07	1000	Da	83.18
Nga7	0.1	07	1000	Nga	84.75
Nga8	0.1	07	1000	Nga	94.72
Nga9	0.1	07	1000	Nga	94.67
Nga10	0.1	07	1000	Nga	88.75
⋮	⋮	⋮	⋮	⋮	⋮
Ya3	0.1	07	1000	Ra	83.96
Ya4	0.1	07	1000	Ra	86.08
Ya5	0.1	07	1000	Ra	89.43
Ya6	0.1	07	1000	Ra	89.56
⋮	0.1	07	1000	⋮	⋮
I8	0.1	07	1000	I	80.71
I9	0.1	07	1000	I	80.80
I10	0.1	07	1000	I	82.70
U7	0.1	07	1000	U	83.98
U8	0.1	07	1000	U	82.67
U9	0.1	07	1000	U	82.71
U10	0.1	07	1000	U	82.73

From Table 2 we see that network recognition using BPNN combine with the zoning method and DBFE able to recognize handwritten Batak Toba characters with an average accuracy of 86.37%. Though there are some letters recognized incorrectly, such as A characters recognized as Ya. This is because the writing character of each person is not the same. Meanwhile, from Table 3, we get a test result that can recognize handwritten Batak Toba characters with average accuracy is 87.19, with a learning rate of 0.1, the momentum of 0.7, and epoch 1000.

In the previous section, there are researchers [16] have the result was not satisfying using Combination of Data using Euclidean Distance (CoDED) method and statistical feature extraction and Elliptic Fourier to recognize handwritten Batak Toba, in this paper recognize it using DBFE and BPNN has resulted with compatibility > 74.43 %. The other method, LVQ used to recognize the Batak document as an image [22] has an accuracy rate better than the results of this research cause there are some letters recognized incorrectly.

According based on previous research, where the research conducted by Khairunisa [14] was about the recognition of continuous letters, and the research conducted by Putra [15] was about handwriting recognition in numerical form, but both still base their research on handwritten recognition which has a high degree of variation. Their researches help to test the architecture and network parameters it uses for the Batak Toba script handwriting recognition.

Using DBFE and BPNN methods to recognize handwritten numerical that conducted by Putra [15] obtained an accuracy rate of 87%. The result almost the same as the result of this research that recognizes handwritten Batak Toba characters using DBFE and BPNN methods was 87.19%.

4. Conclusion

Batak Toba characters are unique in that the script is semi-syllabic (consist of 19 letters) which ends with sound /a/ except for the letters i and u. The zoning method is one of the simplest and most popular feature extraction method combining with DBFE which perform well yielding compare to the system employing conventional horizontal and vertical methods of feature extraction, see [29][30] strengthen the information in the image. From the results of the testing carried out, it can be concluded that by combining BPNN with zoning and DBFE methods, the network handwritten of Batak Toba characters, with the highest recognize rate is 99.45%. Experimental results obtained demonstrated the effectiveness of this system.

References

[1] A. K. M. S. Azad Rabby, S. Haque, S. Abujar, and S. A. Hossain, “Ekushnet: Using convolutional neural network for Bangla handwritten recognition,” in *Procedia Computer Science*, Jan. 2018, vol. 143, pp. 603–610. https://doi.org/10.1016/j.procs.2018.10.437

[2] A. Junaidi, S. Vajda, and G. A. Fink, “Lampung - A new handwritten character benchmark: Database, labeling and recognition,” in *ACM International Conference Proceeding Series*, 2011, Pp. 1. https://doi.org/10.1145/2034617.2034632

[3] H. Fitriawan, Ariyanto, and H. Setiawan, “Neural Networks for Lampung Characters Handwritten Recognition,” in *Proceedings - 6th International Conference on Computer and Communication Engineering: Innovative Technologies to Serve Humanity, ICCCE 2016*, Dec. 2016, pp. 485–488. https://doi.org/10.1109/ICCCE.2016.107

[4] M. Suryani, E. Paulus, S. Hadi, U. A. Darsa, and J. C. Burie, “The Handwritten Sundanese Palm Leaf Manuscript Dataset from 15th Century,” in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, Jul. 2017, vol. 1, pp. 796–800. https://doi.org/10.1109/ICDAR.2017.135

[5] H. Salsabila, E. Rachmawati, and F. Sthevanie, "Sundanese Aksara Recognition Using Histogram of Oriented Gradients," in *2019 2nd International Seminar on Research of Information Technology and Intelligent Systems, ISRITI 2019*, Dec. 2019, pp. 253–258. https://doi.org/10.1109/ISRITI48646.2019.9034589
[6] M. W. A. Kesiman, S. Prum, J. C. Burie, and J. M. Ogier, "Study on feature extraction methods for character recognition of Balinese script on palm leaf manuscript images," in *Proceedings - International Conference on Pattern Recognition*, Jan. 2016, vol. 0, pp. 4017–4022. https://doi.org/10.1109/ICPR.2016.7900262
[7] A. Hidayat, I. Nurtanio, and Z. Tahir, "Segmentation and recognition of handwritten Lontara characters using convolutional neural network," in *2019 International Conference on Information and Communications Technology, ICOIACT 2019*, Jul. 2019, pp. 157–161. https://doi.org/10.1109/ICOIACT46704.2019.8938445
[8] N. T. B. Pasaribu and M. J. Hasugian, "Feature Extraction Comparison in Handwriting Recognition of Batak Toba Alphabet," *IJITEE (International J. Inf. Technol. Electr. Eng.*, vol. 1, no. 3, p. 86, Jan. 2018. https://doi.org/10.22146/ijitee.31969
[9] A. Priya, S. Mishra, S. Raj, S. Mandal, and S. Datta, "Online and offline character recognition: A survey," in *International Conference on Communication and Signal Processing, ICCSP 2016*, Nov. 2016, pp. 967–970. https://doi.org/10.1109/ICCSP.2016.7754291
[10] X. Y. Zhang, Y. Bengio, and C. L. Liu, "Online and offline handwritten Chinese character recognition: A comprehensive study and new benchmark," *Pattern Recognit.*, vol. 61, pp. 348–360, Jan. 2017. https://doi.org/10.1016/j.patcog.2016.08.005
[11] C. L. Liu, F. Yin, D. H. Wang, and Q. F. Wang, "Online and offline handwritten Chinese character recognition: Benchmarking on new databases," *Pattern Recognit.*, vol. 46, no. 1, pp. 155–162, Jan. 2013. https://doi.org/10.1016/j.patcog.2012.06.021
[12] N. Arica and F. T. Yarman-Vural, "An overview of character recognition focused on off-line handwriting," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 31, no. 2. pp. 216–233, May 2001. https://doi.org/10.1109/5326.941845
[13] S. Sitinjak, "Pengenalan Tulisan Tangan Aksara Batak Toba Menggunakan Backpropagation," Universitas Atma Jaya, Yogyakarta, 2012.
[14] Khairunisa, "Pengenalan Tulisan tangan Latin Bersambung Menggunakan Jaringan Saraf Tiruan Propagasi Balik," 2012.
[15] N. Putra, "Peningkatan Fiture Jaringan Propagasi Balik pada Pengenalan Angka Tulisan Tangan Menggunakan Metode Zoning dan Diagonal Base Feature Extraction," *Dunia Teknol. Inf.*, vol. I, no. 1, 2012.
[16] N. Theresia Br Pasaribu and M. Jimmy Hasugian, "Pengenalan Tulisan Tangan Ina ni surat Aksara Batak Toba," 2015.
[17] N. T. B. Pasaribu and M. J. Hasugian, "Noise removal on Batak Toba handwritten script using Artificial Neural Network," in *Proceedings - 2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering, ICITACEE 2016*, Apr. 2017, pp. 373–376. https://doi.org/10.1109/ICITACEE.2016.7892474
[18] P. Romulus, Y. Maraden, P. D. Purnamasari, and A. A. P. Ratna, "An analysis of optical character recognition implementation for ancient Batak characters using K-nearest neighbors principle," in *14th International Conference on QiR (Quality in Research), QiR 2015 - In conjunction with 4th Asian Symposium on Material Processing, ASMP 2015 and International Conference in Saving Energy in Refrigeration and Air Conditioning, ICSERA 2015*, Jan. 2016, pp. 47–50. https://doi.org/10.1109/QiR.2015.7374893
[19] S. Afroge, B. Ahmed, and F. Mahmud, "Optical character recognition using back propagation neural network," Mar. 2017. https://doi.org/10.1109/ICECTE.2016.7879615
[20] S. R. Zanwar, A. S. Narote, and S. P. Narote, "English Character Recognition Using Robust Back Propagation Neural Network," in *Communications in Computer and Information Science*, Dec. 2019, vol. 1037, pp. 216–227. https://doi.org/10.1007/978-981-13-9187-3_20
[21] D. P. Chandra and A. Suryadibrata, "Implementasi Jaringan Saraf Tiruan Backpropagation untuk Pengenalan Karakter pada Dokumen Tercetak," *Ultim. Comput.*, vol. XI, no. 2, Pp. 81, 2019.
[22] M. A. Muchtar *et al.*, "Digitization of Batak Manuscripts Using Methods Learning Vector Quantization (LVQ)," in *IOP Conference Series: Materials Science and Engineering*, May 2020, vol. 851, no. 1, Pp. 012066. https://doi.org/10.1088/1757-899X/851/1/012066
[23] S. Vijayprasath, "A Simple Feature Extraction Method for Analysis of Hand Written Characters," *J. Phys. Conf. Ser.*, vol. 1717, no. 1, p. 012066, Jan. 2021. https://doi.org/10.1088/1742-6596/1717/1/012066
[24] F. Westphal, N. Lavesson, and H. Grahn, "Document image binarization using recurrent neural networks," in *Proceedings - 13th IAPR International Workshop on Document Analysis Systems, DAS 2018*, Jun. 2018, pp. 263–268. https://doi.org/10.1109/DAS.2018.71
[25] A. Choudhary, R. Rishi, and S. Ahlawat, "Off-line Handwritten Character Recognition Using Features Extracted from Binarization Technique," *AASRI Procedia*, vol. 4, pp. 306–312, Jan. 2013. https://doi.org/10.1016/j.aasri.2013.10.045
[26] P. Puneet and N. Garg, "Binarization Techniques used for Grey Scale Images," *Int. J. Comput. Appl.*, vol. 71, no. 1, pp. 8–11, Jun. 2013. https://doi.org/10.5120/12320-8533
[27] L. Ben Boudaoud, A. Sider, and A. Tari, "A new thinning algorithm for binary images," Aug. 2015. https://doi.org/10.1109/CEIT.2015.7233099
[28] D. Phillips, *Image Processing in C Second Edition*. R & D Publications, 1994.
[29] J. Pradeep, E. Srinivasan, and S. Himavathi, "Diagonal Based Feature Extraction for Handwritten Alphabets Recognition System Using Neural Network," *Int. J. Comput. Sci. Inf. Technol.*, vol. 3, no. 1, 2011. https://doi.org/10.5121/ijcsit.2011.3103
[30] M. A. Firmansyah, K. N. Ramadhani, and A. Arifianto, "Pengenalan Angka Tulisan Tangan Menggunakan Diagonal Feature Extraction dan Artificial Neural Network Multilayer Perceptron," *Indones. J. Comput.*, vol. 3, no. 1, pp. 65, May 2018. https://doi.org/10.21108/INDOJC.2018.3.1.214