One-shot learning Batak Toba character recognition using siamese neural network

Yohanssen Pratama, Sarah Try Novelitha Nainggolan, Desy Isabel Nadya, Nova Yanti Naipospos
Department of Software Engineering Technology, Faculty of Vocational Studies, Institut Teknologi Del, Laguboti, Indonesia

Article Info

Article history:

Received Mar 25, 2021
Revised Aug 11, 2021
Accepted Feb 04, 2022

Keywords:

Batak
Character recognition
Convolutional neural network
One-shot learning
Siamese neural network

ABSTRACT

Siamese neural network (SINN) is an image processing model that compares the scores of two patterns. The SINN algorithm is a combination of the use of the double convolutional neural network (CNN) algorithm. By combined SINN with a one-shot learning algorithm, we can build an image model without requiring thousands of images for training. The test results from the SINN algorithm and one-shot learning show that this process was successful in matching the two data but was unable to produce labels from the data being tested. Because of this, the researcher decided to continue the implementation process using the CNN algorithm combined with single shot detection (SSD). By using a dataset of 5000, the recognition and translation of the Toba Batak script was successful. The percentage of average accuracy results from CNN and SSD in recognizing Toba Batak characters is 84.08% for single characters and 74.13% for mixed characters. While the percentage of average accuracy results for testing the breadth first search algorithm is 75.725%.

Corresponding Author:

Yohanssen Pratama
Department of Software Engineering Technology, Faculty of Vocational Studies, Institut Teknologi Del
Jl. Sisingamangaraja, Sitoluama, Laguboti 22381, Indonesia
Email: yohanssen.pratama@del.ac.id

1. INTRODUCTION

The state of Indonesia is a country that is well-known for its various ethnicities and religions. The number of ethnic groups in Indonesia has made the country of Indonesia has a variety of cultures and local wisdom. Local culture is the wealth and identity of a nation. Even in Indonesia, culture has been claimed so that it is not taken by other countries. Local culture is passed down from generation to generation so that its existence does not become extinct. In cultural inheritance there are two things that must be remembered. First, oral culture and secondly written culture. The Batak tribe is one of the tribes that has a written and oral cultural heritage. One of the fruits of the Batak culture is the Batak script.

Character (aksara) is one of the traditional cultural heritages. One script that needs special attention is the Batak script which is endangered due to limited sources of data and information. At first the Batak script was only understood by certain groups such as the leadership group, medical experts and traditional experts at its time. Evidence from the historical Batak heritage can be seen in the Lak-Lak book which contains the writings/letters written in Batak script. The lack of knowledge in reading the Batak script means that the readers cannot interpret the Batak heritage. Batak script should be preserved so as not to diminish the cultural value of the Batak tribe.

At this time Batak script learning is carried out self-taught or in elementary schools. Batak script writing can also be found in the form of pictures that can be found on the internet or in books that provide Batak script writing. Although the internet has Batak script images, the internet cannot translate a collection of sentences written in Batak script into the Batak language in the form of sentences. As a result, the image cannot be recognized by the computer to interpret the meaning of the script.

Along with the development of the times, especially in the field of technology, making people always try to create various innovative breakthroughs with the aim of helping to complete human work. Various innovation breakthroughs in the field of technology have also been applied in the field of culture. Technology based on computer vision. Computer vision is a technology used for processing an image or image obtained from a camera that is similar to the human visual system. This system has been widely used in various things such as checking image elements, shape detection, and face detection.

The technology that will be developed by the research team to help the public in reading Batak script is a technology that applies computer vision and machine learning, namely character classification using the deep learning siamese neural network with one-shot learning [1]. The purpose of this research is to identify Batak script and produce Batak characters and sentences into Batak language using the siamese neural network (SINN) algorithm [2]-[5]. With this application, it is hoped that the community will be able to understand the Batak script so that the Batak script can still be preserved.

2. RESEARCH METHOD

This chapter will be described the method to collect, preprocess, and process data that we used. For the collection method, we create the character using a handwritten and software-generated font like Microsoft Word, and for preprocessing we use some image processing methods such as image segmentation. Also, we discussed the siamese neural network that is used for character matching and classification.

2.1. Data collection method

Data collection was carried out to obtain the information needed to achieve the objectives of the study. For the primary data, we used the character generated by the Microsoft Word application, and we also used handwritten data or another source from the internet. Data collection was carried out divided into two, namely:

a) Primary data

The primary data used in the construction of the Batak script recognition system using the siamese neural network is a written image of the Batak script. Specifically, the data in this study uses a dataset which is divided into training data and testing data [6]-[8]. The training data that will be used is obtained from the Batak script in Microsoft Word which is then screenshot on a white background and black letters. The word document supports the Batak scripts (the font was created by Uli Kozok).

The testing data that will be used consists of the Batak script obtained from the Batak script in Microsoft Word and handwritten on paper and then photographed using a laptop camera with the paper parallel to the camera [9], [10]. These handwritten and software-generated character images will be used as the training and testing data. For the testing data, there will be several combinations of the same and different Batak script characters [11].

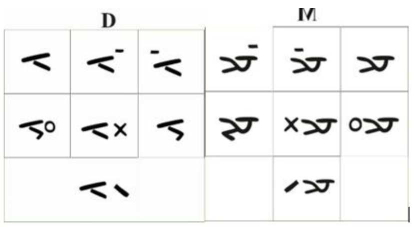


Figure 1. Dasaset folder

One training data will consist of one character while one testing data will consist of more than one character. For a testing character, we will use the sliding window for matching and get the similarity using the siamese network. The Batak script used is the Toba Batak script which has two categories of words, namely:

– Ina Ni Surat

Ina Ni Surat category of a letter is a main word or main word that has a sound ending /a/. In this data there are 19 fonts. So that the construction of the dataset in this letter category has 5 sample images for each letter.

– Anak Ni Surat

Anak Ni Surat is a letter that comes from this letter but the sound ending in /a/ is changed to /i/u/e/o. Anak Ni Surat in Batak script is a phonetic component inserted in Ina Ni Surat (diacritic mark) which serves to change the pronunciation of Ina Ni Surat. In the category there are 98 data for Batak script letters and each Anak Ni Surat has each 5 sample images.

In the convolutional neural network (CNN) process, the dataset used consists of train data generated by the pre-processing process and test data generated by the optical character recognition (OCR) process. The dataset is in the form of images of the Batak script as many as 749 letters consisting of 19 letter Ina Ni Surat and 98 Anak Ni Surat. The image produced by the pre-processing and OCR processes is an image that has passed the grayscale and threshold processes with a size of 105105. Each letter will be separated with each letter having a folder (ex: A, H, and M) and each folder contains the division of the font according to the change in the Anak Ni Surat which will have 7 types of letters and each letter has 5 sample images each respectively (Figure 1).

b) Secondary data

Secondary (supporting) data were obtained through literature reviews and exploration on the internet. Literature review is carried out to obtain theories, methods and concepts relevant to the research being carried out. The information obtained will be a reference in solving problems that are being researched by the research team.

2.2. Siamese neural network (SINN) modeling

The model is a reference in implementing the functions of an algorithm that will be used. The siamese neural network model that will be built aims to provide results from images or Batak characters through a comparison process [12]-[15]. This comparison will go through the process of whether the two images of the Batak script being compared are the same image or not (Figure 2).

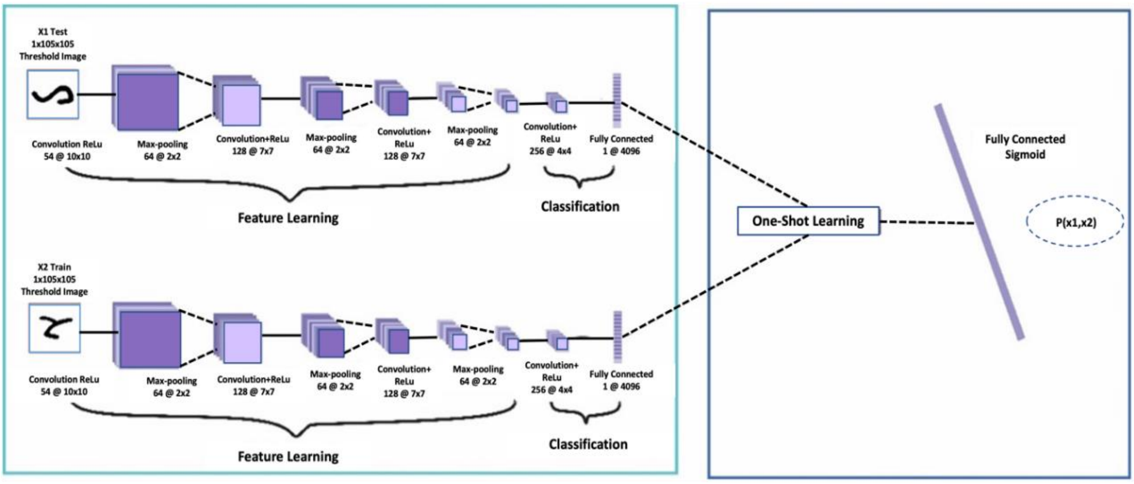


Figure 2. The SINN process provides results that used for one shot learning

The data to be processed by the SINN algorithm is an image with a size of 105×105 grayscale and has been threshold with reference to research conducted by Koch [5]. In order to get the appropriate data, before entering the SINN process, the image will first go through the grayscale process to produce a grayscale image, the thresholding process and the resize process to change the image size to 105×105. In the testing process, the data provided can consist of more than one character. While the SINN algorithm processes one character at a time. So that data consisting of more than one character will be divided by the segmentation process before entering the SINN process [16]. To fulfill the requirements of the data used in the SINN algorithm, the data will first go through a preprocessing process consisting of grayscale, thresholding and resize [17]. However, in the testing data that must be segmented, a segmentation stage will be added which is one of the character recognition stages and will use optical character recognition because the data used is in the form of images [18], [19]. There are three main stages in the process of introducing the Batak script, namely data input, CNN modeling and the application of one-shot learning.

2.2.1. Convolutional neural network modeling

1) Feature learning
   a) Convolutional layer

At the convolutional stage, there are two processes that each data must go through, namely convolution and rectified linear unit (ReLu). The term convolution refers to both the result function and and the process computing it, and to produce the feature map. We use rectified linear unit to activate the feature map. Below is described two processes in the convolutional layer:

– Convolution

At this stage the data convolution process will be carried out. Convolution is the process of repeatedly applying a function. This convolution process will perform the data extraction process to produce a transformation of the input image.

– ReLu

After carrying out the convolution process and producing a feature map, the next process is to activate the feature map. We use ReLu here with purpose that the model will be easier to train and achieved a better performance. The activation function will be used to change the value of the feature map generated from the convolution process. This process will be useful when the feature map value will be passed on to the next process. The image is a pixel containing a matrix, the number of pixels to be generated depends on the filter used. If a grayscale image must be defined first, the kernel size is different from the use of red, green and blue (RGB) which will automatically be divided into three parts, namely red, green, blue [20], [21].

The spatial size of the output of each image will be generated which is calculated based on:

$$(N - F + 2P) / S + 1 \quad (1)$$

Explanation:
N = image size (height = width)
F = filter
P = amount of padding
S = number of stride
If an image has a size of 5×5, a 3×3 filter that uses padding 1 and stride 2, it will generate as many map activations as: $(5 - 3 + 4)/2 + 1 = 4$. This process will be repeated with different filters to produce a collection of activation maps.

b) Pooling

In this process, pooling is carried out to reduce the value obtained in the image convolution results. The pooling process will divide the convolution results into several grids then take the largest value from each grid and form a new matrix containing the largest values of each previously divided grid. The type of pooling used is max pooling because the processed image will have a dark background obtained from the thresholding results. The pooling process only takes the largest value from each matrix and will not reduce the quality of the image that has been pooled. The pooling process is also called downsampling because it reduces the size of the matrix [14]. The pooling process runs by setting the number of filters as a reference in reducing data (Figure 3).

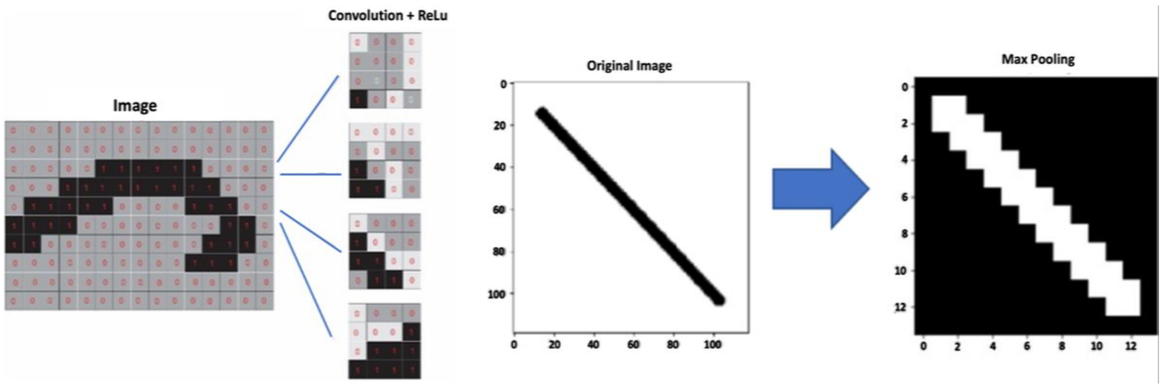


Figure 3. Convolutional + relu and max pooling process

2) Classification

a) Fully connected layer

The convolution and pooling processes cause the neurons of each image to be divided [22]. The last layer on CNN is the process of joining all neurons to reconnect. After the Batak script image is broken down into small grids, the combined neurons will be divided into several groups. This transformation process is called flatten. Dimensions that have been divided and combined are reconnected to connect all neurons into one dimension. A summary of the CNN process that will be carried out can be seen in the Table 1.

Table 1. CNN hyperparameter

Layer	Size	Parameters
Convolution	64×10×10	Stride = 1
Max pooling	54×2×2	Stride =1
Convolution	128×7×7	Stride = 1 Padding = 2
Max pooling	64×2×2	Stride =1
Convolution	128×4×4	Stride = 1 Padding = 1
Max pooling	64×2×2	Stride = 1
Convolution	256×4×4	Stride = 1 Padding = 1
Fully connected + feature vectors	4096	

2.2.2. One-shot learning

The application of one-shot learning is used to determine whether the input images and train data being compared are the same character or not [23]. This process will compare or calculate the results of the value resulting from the convolution process. In this process, a loss function will be implemented using the contrastive loss function. This loss function aims to predict the label value and the relative input distance generated by the input data (Figure 4). This process is used to conduct data training to obtain similarity scores between data points in binary form.

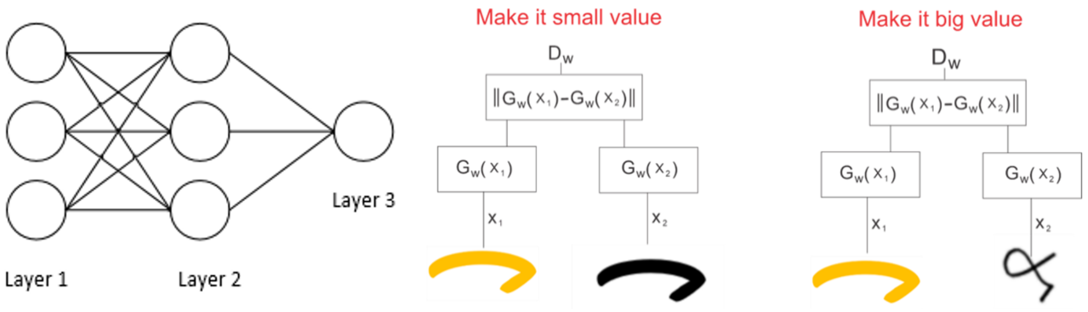


Figure 4. Fully connected layer and one-shot learning

In siamese neural network, to state whether the image is recognized by the system, it uses the contrastive loss function method. This use will compare the input image to all images and the smallest value is the same image as the input image. This comparison process uses the contrastive loss function as a calculation to find the closest distance between images. Figure illustration can be seen in Figure 5.

In the Figure 5, the red circle is an object that is not yet known which part of the Batak script is. While the other objects are a collection of Batak characters that are in accordance with their categories. The unknown object is between the characters "ga" and "ja". Based on the Figure 5, the closest and most number of results is the letter "Ga". It can be predicted that the object will be the letter "Ga", to ensure that the result is used a contrastive loss function (Table 2). In this result Ga = ga and has consistency in capital. After the calculation results are obtained, the contrastive loss function will determine the correct result regarding the meaning of the word by looking for the smallest value result [24].

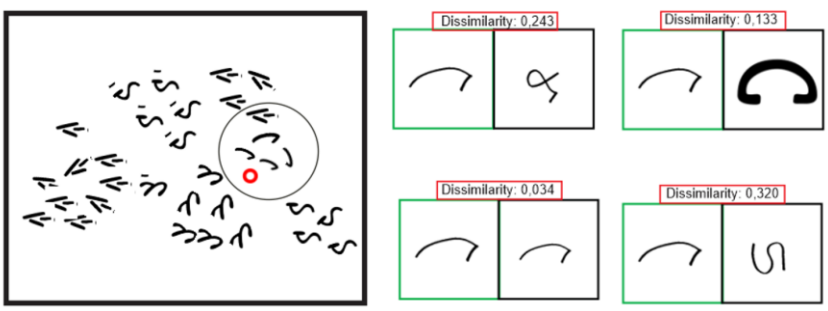


Figure 5. One shot learning illustration and example

Table 2. Contrastive loss function calculation

No	$G_w(X_1)$)	$G_w(X_2)$)	$\|G_w(X_1) - G_w(X_2))\|$
1	1.238	0.243	0.243
2	1.238	0.133	0.133
3	1.238	0.034	0.034
4	1.238	0.918	0.320

2.3. Preprocessing

Preprocessing is one of the steps in image processing. The preprocessing stage aims to improve the quality of the image given. The preprocessing stage will be used to process the train data in order to meet the data requirements that can be processed by the SINN algorithm. The processes required by the data train for preprocessing are grayscale, thresholding and resize (Figure 6) [25].

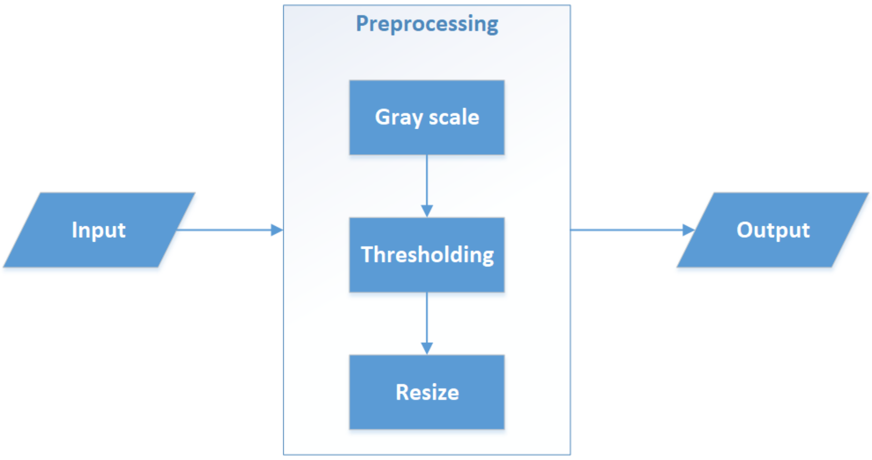


Figure 6. Preprocessing stage

a) Input is the raw data processing stage that will be used in the classification process. The raw data is 575 Batak script data in the form of images.
b) The next process is grayscale. Grayscale is converting an image in color to black and white which functions to remove noise in the data.
c) Thresholding will be done to adjust the color by applying values of 0 and 1 in order to separate the pattern from the background (Figure 7).
d) The next stage is the process of resizing the image to equalize the entire image size to 105×105 because not all data is the same size, so the resizing process is required. Example of the letter "Ga" in grayscale. The letter "Ga" initially has a size of 524×364 and then processed to 105×105 (Figure 8).

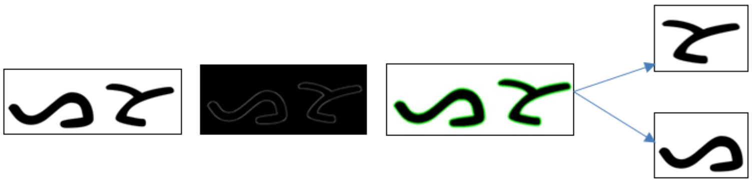


Figure 7. Preprocessing (grayscale, thresholding, and segmentation)

Figure 8. Resize result

2.4. Breadth first search (BFS) algorithm

When the input goes through the OCR process and finds the appropriate label from the SINN process, it will produce a label for each character combined in one word. If the input consists of more than one word, it is necessary to separate them with spaces. To divide the word, the BFS algorithm will be used and the BFS algorithm is dependent to the CNN outputs. The words that have been obtained will be searched and matched with the words in the Indonesian Batak dictionary written by J.P. Sarumpaet. The application of BFS in the word division process is:

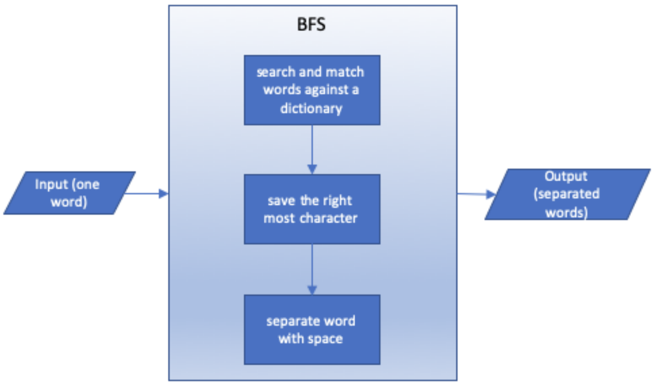


Figure 9. Breadth first search process

This section explains the application of the BFS algorithm to divide the words obtained from the results of one-shot learning with the reference to the Indonesian Batak dictionary. The scheme used can be seen in the following Figure 9. The explanation of the scheme is:

– Words that have been obtained from the results of one-shot learning will be searched and matched with the Batak dictionary. Matching is done by looking for a word in the dictionary with the first letter, last letter and the number of letters according to the word obtained.
– If the word is not found, the last letter will be taken so that the word will be reduced by one letter and searched again.
– After finding which word will be the rightmost word, it is put into one word and separated from the words that have not been matched.
– The process will be repeated for words that have not been completed to the last letter.
– All words that have been matched will be recombined, separated by a space.

2.5. System overview

At this time Batak script learning is carried out self-taught or in elementary schools. With this problem, the research team will create a system that is able to help translate the Batak script. The system will be developed by a desktop-based research team and will change the Batak script taken using a camera. The system will identify the Batak script on the inputted image and convert it to the alphabet to form a word. The results of the Batak script reading performed by the system will be displayed on the monitor. The flow of the system can be seen in Figure 10.

The test data will pass the OCR stage and then classified with training data to be compared and produce values with one-shot learning. The results will be labeled with letters into one word and then the words will be separated by spaces using the BFS algorithm. The design of the system display can be seen in the Figure 11. On the left will be shown the image taken from the camera. After pressing the translate button, the translation of the image will be displayed on the right.

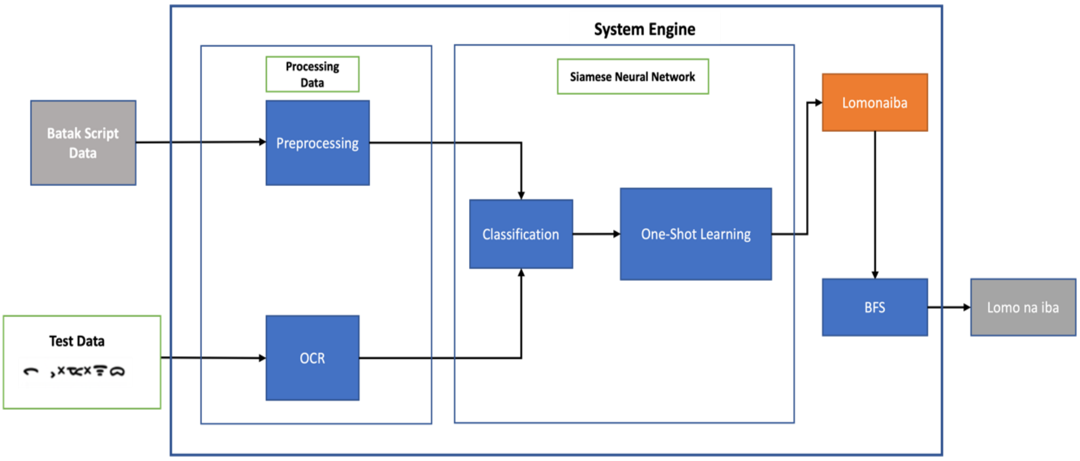


Figure 10. System flow diagram

Figure 11. System user interface

2.6. Batak translator architectural design

The system architecture design can be seen in Figure 12. The user will take a picture in the form of a sentence in the Batak script using a laptop camera as a capture device. Then the results of the image will be translated into the Batak language using a Batak translator.

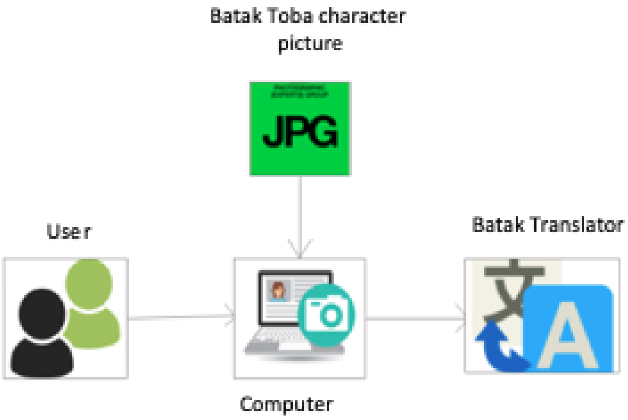


Figure 12. System architecture

3. RESULTS AND ANALYSIS

This chapter describes the results obtained from the implementation and discussion of the introduction of the Toba Batak script using the siamese neural network. The results shown are the results of preprocessing, namely grayscale, thresholding, resize, segmentation and the results of one-shot learning. The activity data obtained is divided into 2 types of datasets, namely training data and testing data.

3.1. Siamese neural network results

The results of using the siamese neural network obtained during the siamese convolutional neural network algorithm research are that the Siamese algorithm is not capable of predicting the input image. Siamese is only capable of calculating the difference between two pictures for which you want to see the similarities. The image that can be seen the difference is the image that has been carried out by the previous train process. Siamese neural network will provide a value from the calculation of the contrastive loss function and the final result if the value given is 0 then the image is different and the value is 1 if the image is the same.

During the implementation process, the researcher found that the siamese neural network did not focus on character recognition or on the classification of data provided during the training period. Siamese neural network can perform the recognition or similarity value search process not because the system recognizes the type of letter or type of classification, but this system is designed to understand handwriting patterns. When testing an image with a line pattern which is the result of the thresholding process, it can give a tensor value whether the image is the same or different from the requirements that the image must be 105×105 in size and has gone through the thresholding process.

Table 3. Siamese neural network result

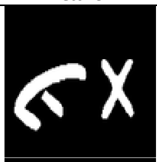
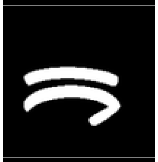
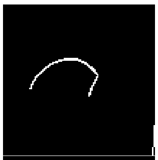
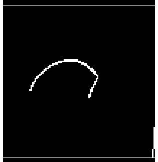
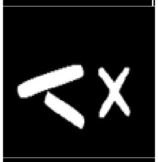
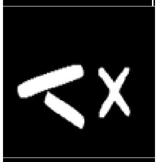
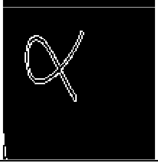
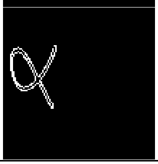
Picture 1	Picture 2	Tensor value	Final score	State
		10.9938	0	Not match
		20.679	0	Not match
		0.0278	1	Match
		0.0278	1	Not match
		0.167	0	Match

Table 3 is the result of a comparison of the same and different images with the Batak script and non-Batak script categories. The tensor value would be the indicator of whether the input image (Figure 1) matches the reference image (Figure 2). Higher the tensor value, the mismatch between images also larger.

3.2. Breadth first search results

The results of using the Breadth first search algorithm are currently unable to take input from the Siamese process due to the problem that Siamese cannot predict words or cannot read if more than 1 character. Therefore, the researcher input from the ".txt" file which contains Toba Batak language sentences that do not have spaces and will do word matching with the dictionary and word separation with spaces. For the accuracy of word separation, the researcher uses:

$$Accuracy = \frac{correct\ words}{total\ tested\ words} \times 100\% \tag{2}$$

In Table 4 the average percentage of Toba Batak word separation is 75.725%. The results show 100% accuracy as the highest accuracy result and 53.84% accuracy as the lowest accuracy result. From these results it can be concluded that the code for the breadth first search algorithm can be used for the word separation process.

Table 4. BFS testing result

No	Original word	BFS result	Correct words	Wrong words	Total tested words	Accuracy (%)
1	aekgodangtuaeklautdosnirohasibaennasaut	aek godang tu aek laut dosni ro hasi baen na saut	9	2	11	81.81%
2	bornginibulanmansaitorangdohotbintangsaimangirdopihuboanmaibanamardalanitutopinitaoholansoaraniaekdotarbegeidohotsipatsoaranihirikdiasarna	borngin i bulan mansai torang dohot bintang sai mangirdopi hu boan mai bana mardalani tu topi ni tao holan soara ni aek do tarbegei dohot sipata soara ni hirik di asar na	28	4	32	87.5%
8	rambanaposonasotubuanlatahalaknaposo	ramba na poso na so tubuan lata halak na poso na so umboto hata	14	-	14	100%
						Accuracy = 75.725%

3.3. Convolutional neural network

This section describes the results obtained from the CNN algorithm evaluation and system testing. We will evaluate the CNN algorithm accuracy when doing the character recognition and also the system translation performance. The system performance will be tested in this part.

3.3.1. Convolutional neural network algorithm evaluation

The results obtained from testing the CNN algorithm using the SSD method will be stored in a “.txt” file. To calculate the accuracy, the researcher uses the:

$$Accuracy = \frac{correct\ word}{total\ tested\ words} \times 100\% \tag{3}$$

The results of the evaluation and accuracy of the CNN algorithm with the single shot detection method are divided into two, namely 12 single characters and 8 images consisting of mixed characters which can be seen in Table 5 and Table 6. Evaluation results of the CNN algorithm the test results were carried out on 12 Toba Batak characters. Single with each character consisting of 10 data can be seen in Table 5. From the results of these tests the researcher presents them in the following graph:

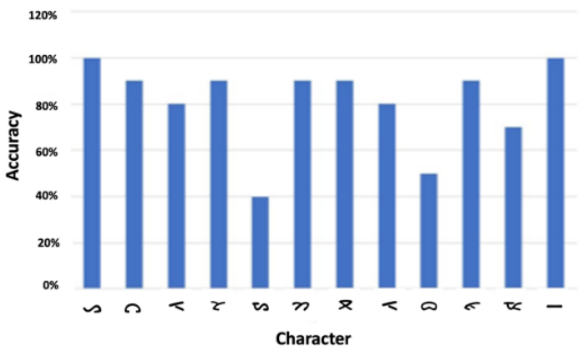


Figure 13. Percentage of Batak Toba script evaluation (single data)

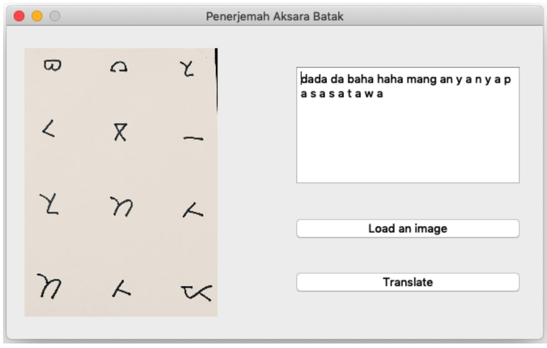
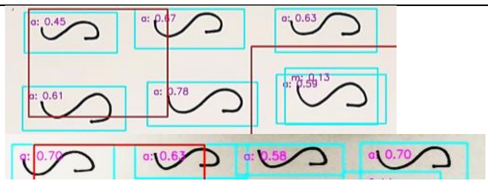
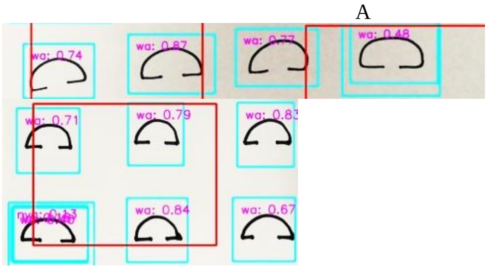
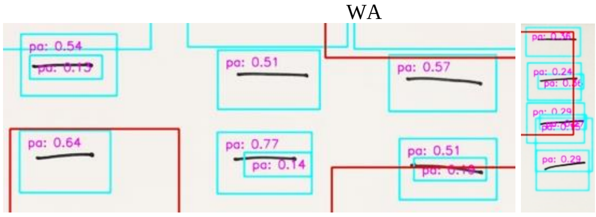


Figure 14. System test result

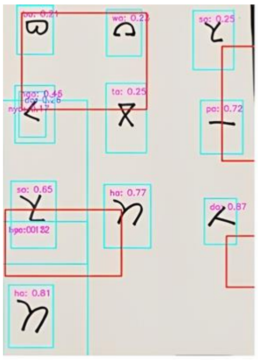
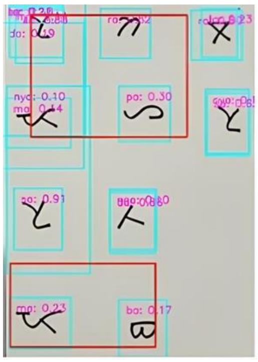
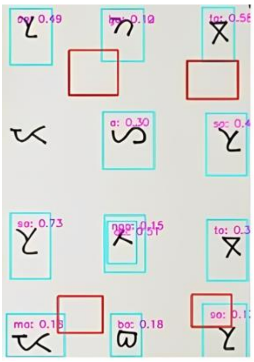
Table 5. Evaluation results of the CNN algorithm (single character)

No	Testing Result	Correct word	Total tested words	Accuracy (%)
1	 A	10	10	100%
2	 WA	9	10	90%
12	 PA	10	10	100%
				Average accuracy = 73.3 %

On the diagram Figure 13 shows that the average percentage of successful Toba Batak script recognition is 84.08%. The results show that letters A and Pa get 100% accuracy as the result of the highest accuracy and letters Ya get 40% accuracy as the lowest accuracy result. The letter Ya got the lowest accuracy because there are two parts of characters and sometimes it counts as two characters instead one.

The test results of 8 images consisting of 10 and 12 characters of the mixed Toba Batak script can be seen in Table 6. From the test results, the average percentage of success in the recognition of the Toba Batak script is 74.13%. The results show that data 1 gets 91% accuracy as the highest accuracy result and data 3 gets 50% accuracy as the lowest accuracy result.

Table 6. Evaluation results of the CNN algorithm (multiple or mixed character)

No	Testing result	Correct word	Total tested words	Accuracy (%)
1	BA WA SA NGA TA PA SA HA DA HA	9	10	91%
				
3	SA HA TA MA A SA SA DA MA BA	5	10	50%
				
8	SA HA TA MA A SA SA HA TA MA BA SA	9	12	75%
				
				Average accuracy = 74.13 %

3.3.2. System testing

The last test performed was to test the whole process by applying the CNN algorithm and combined it with the BFS algorithm which can be seen in Figure 14. The load an image button is used to select an image and is displayed on the left. The translate button will carry out the function to identify the Batak Toba script object from the image and then save the result in a ".txt" file. The file will be processed with the BFS algorithm to separate each Batak Toba word and the output will be displayed on the system in the upper right box.

4. CONCLUSION

Siamese neural network and one-shot learning algorithms can be used in the matching process between two images to determine whether the two images are the same or different images. The use of SINN with the implementation of CNN could be used to predict similarity by seeing the distance between two compared images. So in this experiment, the recognition and translation of the Batak Toba script were successfully carried out using single shot detection by implementing the CNN architecture.

We also utilized the BFS algorithm, so it can check every word generated at previous detection by using CNN. This process will receive input in the form of a ".txt" file containing the translated text. The system will check each word and process it to suit the Batak language dictionary using the BFS algorithm. After the process is done, a ".txt" file will be generated containing Batak language sentences that have been adjusted to the Batak language dictionary. There is the detection of repeated objects in the image using single shot detection which affects the detection results. For future works, we hope could prevent the multicount object and try to utilize SINN to predict the character by seeing the distance after compared it to all closest characters.

ACKNOWLEDGEMENTS

This work was supported and funding by the research institutions and community service of Institut Teknologi Del.

REFERENCES

[1] B. K. Barakat, R. Alasam, and J. El-Sana, "Word spotting using convolutional siamese network," in *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, 2018, pp. 229-234, doi: 10.1109/DAS.2018.67.

[2] A. M. M. O. Chacko and P. M. Dhanya, "A comparative study of different feature extraction techniques for offline malayalam character recognition," *Computational Intelligence in Data Mining*, 2014, vol. 2, pp. 9–18, doi: 10.1007/978-81-322-2208-8_2.

[3] B. Gosselin, "Multilayer perceptrons combination applied to handwritten character recognition," *Neural Processing Letters*, vol. 3, pp. 3–10, 1996, doi: 10.1007/bf00417783.

[4] Li F. -Fei, Fergus, and Perona, "A bayesian approach to unsupervised one-shot learning of object categories," *Proc. Ninth IEEE International Conference on Computer Vision*, 2003, vol. 2, pp. 1134-1141, doi: 10.1109/ICCV.2003.1238476.

[5] G. Korch, "Siamese neural networks for one-shot image recognition," M. S. Thesis, Department of Computer Science, University of Toronto, 2015. [Online]. Available: http://www.cs.toronto.edu/~gkoch/files/msc-thesis.pdf

[6] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701-1708, doi: 10.1109/CVPR.2014.220.

[7] D. Wu, F. Zhu, and L. Shao, "One shot learning gesture recognition from RGBD images," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 7-12, doi: 10.1109/CVPRW.2012.6239179.

[8] N. I. A. Sabri and S. Setumin, "One-shot learning for facial sketch recognition using the siamese convolutional neural network," in *2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, 2021, pp. 307-312, doi: 10.1109/ISCAIE51753.2021.9431773.

[9] A. Ullah, K. Muhammad, K. Haydarov, I. U. Haq, M. Lee, and S. W. Baik, "One-shot learning for surveillance anomaly recognition using siamese 3D CNN," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1-8, doi: 10.1109/IJCNN48605.2020.9207595.

[10] V. A. Krishna, A. Reddy, and D. Nagajyothi, "Signature recognition using siamese neural networks," in *2021 IEEE International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, 2021, pp. 1-4, doi: 10.1109/ICMNWC52512.2021.9688430.

[11] N. Dlamini and T. L. V. Zyl, "Author identification from handwritten characters using siamese CNN," in *2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, 2019, pp. 1-6, doi: 10.1109/IMITEC45504.2019.9015897.

[12] S. Wang, X. Lv, R. Li, C. Yu, and J. Dong, "Characters verification via siamese convolutional neural network," in *2018 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, 2018, pp. 417-420, doi: 10.1109/SPAC46244.2018.8965605.

[13] G. Sokar, E. E. Hemayed, and M. Rehan, "A generic OCR using deep siamese convolution neural networks," in *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2018, pp. 1238-1244, doi: 10.1109/IEMCON.2018.8614784.

[14] D. Chauhan, D. Verma, and A. Aggarwal, "Hand-written characters recognition using siamese network design," *2022 1st International Conference on Informatics (ICI)*, 2022, pp. 66-70, doi: 10.1109/ICI53355.2022.9786908.

[15] A. Mittur, A. R. Shankar, and A. Narasimhan, "One-shot approach for multilingual classification of indic scripts," in *2022 International Conference on Innovative Trends in Information Technology (ICITIIT)*, 2022, pp. 1-6, doi: 10.1109/ICITIIT54346.2022.9744238.

[16] W. Xiao and D. Wu, "An improved siamese network model for handwritten signature verification," in *2021 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, 2021, pp. 1-6, doi: 10.1109/ICNSC52481.2021.9702190.

[17] C. Adak, S. Marinai, B. B. Chaudhuri, and M. Blumenstein, "Offline bengali writer verification by PDF-CNN and siamese net," in *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, 2018, pp. 381-386, doi: 10.1109/DAS.2018.33.

[18] M. Kassis, J. Nassour, and J. El-Sana, "Alignment of historical handwritten manuscripts using siamese neural network," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 2017, pp. 293-298, doi: 10.1109/ICDAR.2017.56.

[19] L. Meng, N. Kamitoku, X. Kong, and K. Yamazaki, "Deep learning based ancient literature recognition and preservation," in *2019 58th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, 2019, pp. 473-476, doi: 10.23919/SICE.2019.8860070.

[20] C. N. R. Kumar and S. Srinath, "A novel and efficient algorithm to recognize any universally accepted braille characters: a case with kannada language," in *2014 Fifth International Conference on Signal and Image Processing*, 2014, pp. 292-296, doi: 10.1109/ICSIP.2014.52.

[21] C. -H. Xue and X. -F. Jin, "Characters recognition of korean historical document base on data augmentation," in *2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*, 2020, pp. 2304-2308, doi: 10.1109/ICMCCE51767.2020.00498.

[22] M. A. Souibgui, A. Fornés, Y. Kessentini, and C. Tudor, "a few-shot learning approach for historical ciphered manuscript recognition," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 5413-5420, doi: 10.1109/ICPR48806.2021.9413255.

[23] C. Li, Y. Taniguchi, M. Lu, and S. Konomi, "Few-shot font style transfer between different languages," in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 433-442, doi: 10.1109/WACV48630.2021.00048.

[24] 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 *2015 International Conference on Quality in Research (QiR)*, 2015, pp. 47-50, doi: 10.1109/QiR.2015.7374893.

[25] M. Panggabean and L. A. Rønningen, "Character recognition of the Batak Toba alphabet using signatures and simplified chain code," *2009 IEEE International Conference on Signal and Image Processing Applications*, 2009, pp. 215-220, doi: 10.1109/ICSIPA.2009.5478614.

BIOGRAPHIES OF AUTHORS

Yohanssen Pratama Current Faculty Members & Researcher in Institut Teknologi Del. 4+ years experience specializing in back-end/infrastructure, analytical tools development and computer programming. Teach academic and vocational subjects to undergraduates and also pursue my own research to contribute to the wider research activities of my department. He can be contacted at email: yohanssen.pratama@del.ac.id.

Sarah Try Novelitha Nainggolan Currently, work as Business Analyst at Astra Graphia Information Technology (AGIT). Have experience at image processing using neural network, web & desktop developing, UX Design. She can be contacted at email: if416030@students.del.ac.id.

Desy Isabel Nadya Currently, work as Assistant Manager at Horas Family Home. Have experience at image processing using neural network, web and desktop developing, fullstack developer. She can be contacted at email: desy.isabel97@gmail.com.

Nova Yanti Naipospos Currently, work as a Technical Writer at PT. Bank Rakyat Indonesia. Have experience at image processing using neural network, web, and desktop developing. She can be contacted at email: if416012@students.del.ac.id.