The Involvement of Local Binary Pattern to Improve the Accuracy of Multi Support Vector-Based Javanese Handwriting Character Recognition

Christy Atika Sari*[1], Wellia Shinta Sari[2], Viki Arri Shelomita[3], Mohammad Roni Kusuma[4], Silfi Andriana Puspa[5], Muhammad Bima Gusta[6]

[1,3,4,5,6]*Study Program in Informatics Engineering, Faculty of Computer Science, University of Dian Nuswantoro, Semarang, Indonesia*
[2]*Study Program in Information Systems, Faculty of Computer Science, University of Dian Nuswantoro, Semarang, Indonesia*
E-mail : christy.atika.sari@dsn.dinus.ac.id*[1], wellia.shinta@dsn.dinus.ac.id[2], viki.ari@gmail.com[3], 611202100020@mhs.dinus.ac.id[4], 611202100045@mhs.dinus.ac.id[5], 611202100024@mhs.dinus.ac.id[6]
*Corresponding author

Abstract - Indonesia is a country that is rich in cultural diversity. An example of one such variety is the Javanese language. The letters that are usually used in Javanese are non-Latin letters or are usually known as Javanese script. However, along with advances in technology, the Javanese language is increasingly being forgotten. In the past, the Javanese script was used as a subject in schools, aiming for Indonesian students to continue to gain knowledge about the Javanese script. The initial step in the introduction of the Javanese script starts with the preprocessing process by changing the image of the Javanese script from the RGB image to a grayscale image which is then performed feature extraction, where the feature extraction used in this script recognition is texture extraction with the Local Binary Pattern (LBP) algorithm. The results of this processing are obtained information that can be used as a parameter in the Multi Support Vector Machine (SVM) classification to predict Javanese script images. In this study using the LBP method with the Multi SVM Algorithm as a classification algorithm produces a high accuracy of 90% in the recognition of Javanese script, better than using only Multi SVM with an accuracy of 80%.

Keywords - Local Binary Pattern, Javanese character, Handwriting Script, Multi SVM

1. INTRODUCTION

Indonesia is a country that is rich in cultural diversity. An example of one such variety is the Javanese language. The letters that are usually used in Javanese are non-Latin letters or are usually known as Javanese script. However, along with advances in technology, the Javanese language is increasingly being forgotten. In the past, the Javanese script was used as a subject in schools, aiming for Indonesian students to continue to gain knowledge about the Javanese script [1], [2].

The initial step in the introduction of the Javanese script starts with the preprocessing process by changing the image of the Javanese script from the RGB image to a grayscale image which is then performed feature extraction, where the feature extraction used in this script recognition is texture extraction with the Local Binary Pattern (LBP) algorithm [3], [4]. The results

of this processing are obtained information that can be used as a parameter in the Multi SVM classification to predict Javanese script images. In this study using the LBP method with the Multi SVM Algorithm as a classification algorithm produces a fairly high accuracy of 90% so that it can be used as a system for recognizing Javanese script [5] .

Javanese letters or generally called Javanese script have their own level of difficulty to be learned. This is because it is associated with many similar curves and circles from letter to letter. The difficulties that occur in learning Javanese script will affect the interest of young people in learning Javanese letters [6], [7]. In addition, there are not many applications regarding the introduction of Javanese letters.

Image recognition process can be done by classification process either in real time or directly or indirectly. Image classification is usually done using 2 image processing techniques, namely unsupervised learning and supervised learning techniques. In supervised learning techniques, several algorithms that are usually and commonly used by researchers are SVM (Support Vector Machine) [8]–[10], Neural Network (NN) [11]–[17], K-Nearest Neighbor (KNN) [18]–[20], and others. The use of mind processing using supervised learning algorithms is usually used when researchers already have clear training data and variables to classify data. While the research object is in the form of images that are often used which can be static images or dynamic images. Static images are images obtained directly from available media such as books, the internet where images can be used directly to make detection easier. Meanwhile, dynamic images are images that are drawn or written by humans on a piece of paper or any media, for example, and the shape of the image so that one image from another will definitely be different and will not always be the same as a picture in a book or on the internet.

There have been many studies on handwriting recognition based on various methods, such as research conducted by Susanto, et. Al [5], in his research on A High Performance of Local Binary Patterns on Classify Javanese Character Classification, with the Local Binary Pattern (LBP) algorithm for feature extraction and KNN as the classification algorithm, produces an accuracy of 82.5% in the parameters K = 3 and LBP [64 64]. Another study is the by [21] using Javanese script character recognition using the Convolutional Neural Network method as a classification algorithm on Javanese script images, resulting in an accuracy of 95.04%. Another study is the research by [22], regarding the introduction of Javanese script using the Area Based Feature Extraction and Support Vector Machine methods resulting in an accuracy of 90.84%. Several previous studies inspired this researcher to create a Javanese script recognition system. This study proposes the LBP and Multi SVM methods. SVM was chosen because it has a computation that has a fairly good level of accuracy. The use of the LBP method is also considered more effective in solving problems to recognize the texture of Javanese script.

Here, the LBP algorithm chosen as a feature extraction process in this study because LBP was first studied to be used for texture analysis. In this study, it is also necessary to analyze the form of the Javanese script writing so that with LBP the computer can recognize the Javanese script writing form and use Multi SVM.

2. RESEARCH METHOD

Overall in this research is to analyze and identify patterns of Javanese script and recognize whether the Javanese script is correct or not. The introduction of the Javanese script uses the LBP algorithm and to determine the accuracy level of the system's accuracy in recognizing the Javanese script using the Multi SVM algorithm to classify it. The initial method used is to input the data taken with a cellphone camera, then the image is pre-processed from the RGB image which is then converted into a grayscale image. Then the grayscale image is converted into a binary image, which is then carried out by a feature extraction process where

this feature extraction process uses the LBP algorithm. After carrying out the feature extraction process on the image and classification is carried out for the process of recognizing Javanese script letters using the Multi SVM algorithm. The hypothesis of the above explanation is that in this study it can recognize Javanese script writing based on the pattern or shape of the indentation of the Javanese script.

The process of recognizing Javanese characters by combining the LBP and Multi SVM algorithms, in which LBP is used as feature extraction and Multi SVM as a prediction or classification of Javanese script. This merging algorithm begins with the process of taking pictures using a cellphone. Then the RGB image from the photo is converted to greyscale, after which the image is resized to a size of 256x256 pixels. After that, the feature extraction process is carried out using LBP. The next process is the classification process using the Multi SVM algorithm. The flow of the research process is explained in Figure 1.
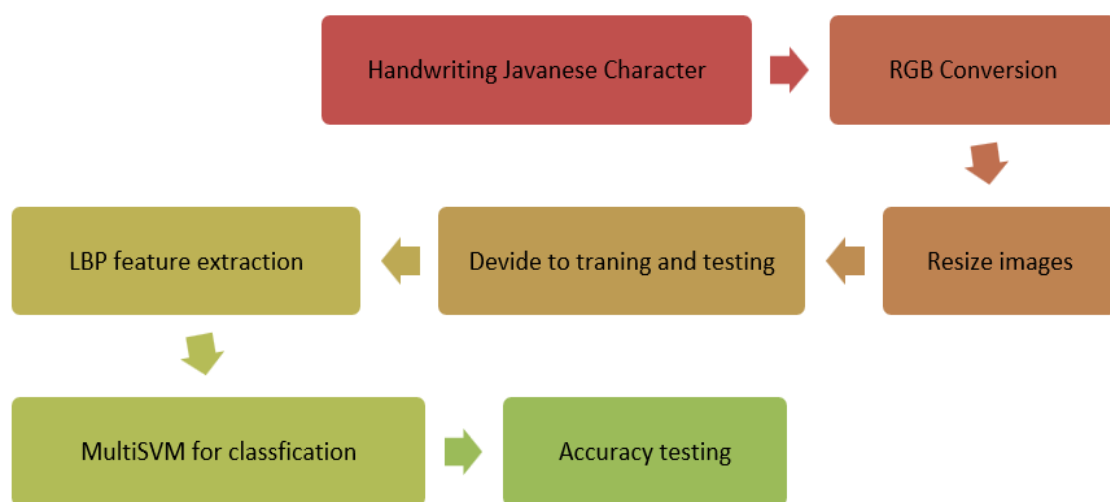
Figure 1. Proposed Method

2.1 Data

The data is in the form of images of Javanese script writing which consists of 20 Javanese script letters and 20 of each letter are written and then photographed using a mobile phone camera. groups of letters consisting of 30 images and 200 images are used as testing data.

Table 1. Datasets labelling

Image Name	Text Label
BA_01	BA
BA_02	BA
…	…
CA_31	CA
CA_32	CA
…	…
DA_61	DA
DA_62	DA
…	…
DHA_91	DHA

Image Name	Text Label
DHA_92	DHA

Data collection was carried out by writing directly by hand on 800 Javanese script letters, Javanese script writing containing 600 Javanese script writing which was used for data training, and 200 Javanese script writing which was used for data testing. So that a total of 800 Javanese writings are handwritten using a black marker, written on A4 size HVS paper. Then the image is taken by taking a photo with the Samsung A51 cellphone camera, with the best lighting. After that, each picture is grouped by letter and given a name. Table 1 is an example of naming the data for training. Image acquisition is a method of collecting image datasets to be used in research. In this study the dataset was obtained from the handwriting of 20 Javanese script letters written on the media in the form of A4 white paper. After writing, each post will be taken or photographed using the Samsung A51 mobile phone camera, with the best lighting.
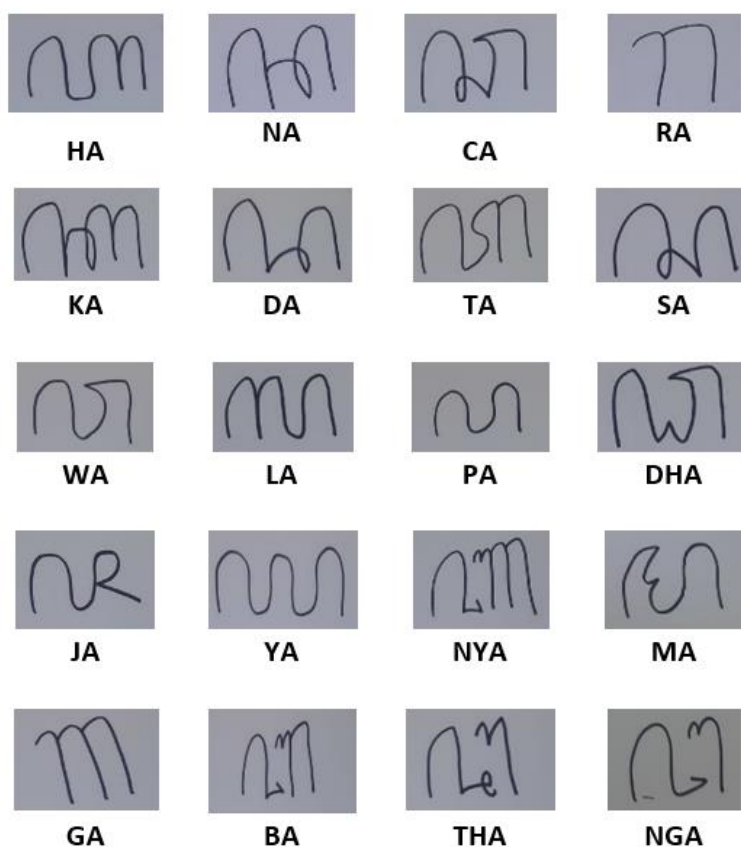
Figure 2. Javanese Handwriting Characters

2.2 RGB Conversion

After the dataset is obtained, the next step for testing this time is preprocessing, where the goal with image preprocessing before processing serves to improve digital image quality in order to achieve maximum results. In digital image preprocessing initially the sizes are still different so as to make it easier and to achieve maximum accuracy so it is necessary to change all image sizes to 256x256 size. To resize digital images, you can use the matlab imresize() function.

2.3 Local Binary Pattern (LBP)

Local Binary Pattern is an excellent method used for texture analysis which was first introduced by [7]. LBP uses 8 neighboring pixels and 1 central pixel which functions as a threshold value. The value of each neighboring pixel will be compared with the central pixel value, if the central value is greater than the neighboring pixel value then the neighboring pixel value is 0, otherwise the value is 1. All neighboring pixel values are combined from left to right, a binary number will be obtained which can be decilmalized which are also known as Local Binary Patterns (LBP) as shown in Figure 3.

The basic LBP has limitations, namely the size of a 3x3 neighboring pixel cannot capture large-scale texture domain features. Therefore, to deal with large-scale textures, LBP is developed using larger local neighboring pixels. The first step that can be taken in developing a local neighborhood pixel size is to define a T texture in the neighborhood of a P + 1 (P > 0) gray-scale image as illustrated in (1), where gc is the neighboring local pixel center value. gp(p=0, . . . , P-1) is the grayscale pixel value P with the same space as the radius of the circle R (R> 0) will form a circular symmetric neighbor local pixel. The figure below illustrates neighboring local pixels with different P and R values, (P,R) shows P neighboring pixels with a circle radius of R. Neighboring pixel values that do not exactly fall on the pixel will be calculated using bilinear interpolation [23].

$$T = t(gc, g0, . . . , gp\text{-}1) \tag{1}$$
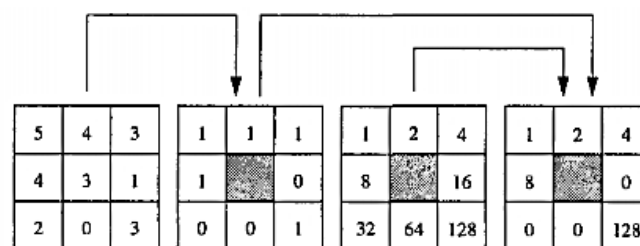
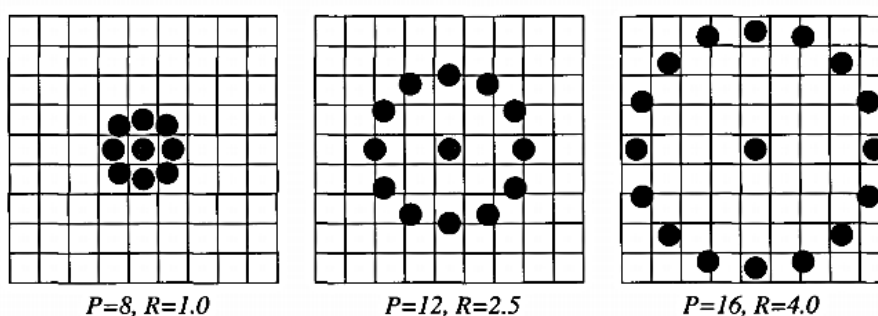$$LBP = 1+2+4+8+128 = 143$$

Figure 3. LBP conversion to biner value

Figure 4. Center of radius of LBP

A 2P binomial is assigned to each pixel s(gp-gc) to convert differences in neighboring pixels into LBP codes. The generated code will be used as the texture characteristic value of an image around (xc, yc) as illustrated in (2).

$$LBPP, R(xc, yc) = \sum_{P=0}^{P-1} s(g_p - g_c) 2^p \tag{2}$$

Where g_p is the gray value of the neighboring pixel and g_c is the gray value of the center pixel and P circle neighborhood pixel with radius R, and the function s(x) is defined as in (3).

$$s(x) = \begin{cases} 1 \ if \ x > 0 \\ 0 \ if \ x < 0 \end{cases} \tag{3}$$

Based on the definition above, it can be concluded that LBP is an invariant for grayscale image transformation which maintains the intensity order of each neighboring pixel. The calculated histogram LBP in an area can be used as a texture descriptor. The LBP operator (P,R) produces 2p different output values, according to the 2p binary pattern formed by P pixels in the neighbor set. So it can be seen that the binary contains more information than the other binaries. Therefore, it is possible to use a 2p binary pattern subset to be able to describe the texture of an image. The pattern a uniform pattern with the notation $LBP_{(P,R)}^{U2}$. There are 2 bitwise transitions from 0 to 1 or vice versa in a circular binary which can be said to be a uniform local binary. For example, 00000000(0 transitions) and 00110000 (2 transitions) are called uniforms because there are 2 bitwise transitions whereas 10001011 (4 transitions) 10101101 (6 transitions) are not uniforms. From the results of research conducted by Ojala et al, the uniforms pattern dominates 90% of the overall pattern in the (8,1) neighborhood and 70% dominates the (16,2) neighborhood in the image texture.

2.4 *Support Vector Machine (SVM)*

SVM is a learning system which uses parameters in the form of linear functions to predict or classify. SVM uses a feature space with high dimensions, and uses a learning algorithm based on the implementation optimization theory of statistical learning [7], [9], [22]. SVM is a classification method that is currently widely used and developed by many studies. SVM was sparked from learning theory using statistical numbers and promises to provide classification or prediction results compared to other classification methods. SVM can work very well in high-dimensional data sets. SVM itself uses a kernel technique which must map the original data from its original dimension to another dimension with a relatively higher dimension. In the Artificial Neural Network (ANN) method, during the training process all training data will be thoroughly studied.

2.5 *Testing Method*

In this study, we use confusion matrix calculation, because the confusion matrix can evaluate the prediction results of the Multi SVM algorithm more specifically because it has parameters True Positive (TP), True Negative (TN), False Negative (FN), False Positive (FP) as shown in (4) and Table 2. Thus, the results of the accuracy of the method used will be obtained so that it can determine the percentage of validation of a person's handwriting pattern recognition.

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FP} \ x \ 100\% \tag{4}$$

Table 2. Confusion Matrix

Confusion Matrix		Prediction Class		
		False	True	Total
Original Class	False	TN	FP	Total Class
	True	FN	TP	Total Class

Confusion Matrix		Prediction Class		
		False	True	Total
	Total	Prediction Class	Prediction Class	Grand Total

3. RESULTS AND DISCUSSION

The Javanese script that is written has characteristics, namely the curvature of each corner so that it distinguishes between one letter and another, but even though it has different characteristics, there are several letters that have almost the same characteristics as CA, SA, TA. Where the writing of these characters has similarities so that it confuses the system and humans. Therefore, there is a need for confirmation in the writing of Javanese script letters assisted by the preprocessing process to be able to better recognized by the system later as shown in Figure 5.
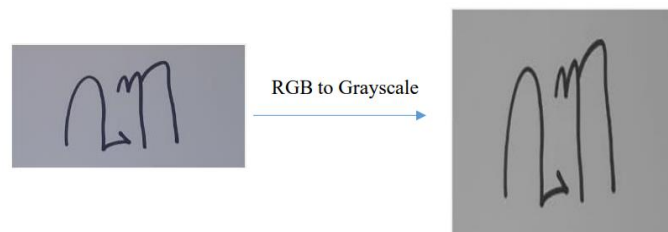

Figure 5. Convert RGB to Grayscale

After the image is resized, the image conversion will continue where the initial image is still in the form of an RGB image which will be converted to a grayscale image using the Matlab function, namely rgb2gray(), while the reason for converting to a grayscale image is to make it easier in the next process, namely the LBP Feature Extraction process because LBP can only process 2 images, namely binary images and grayscale images, convert RGB images into grayscale images.

```
img = imread(full_name);
matrix = [256 256];
img_resize = imresize(img,matrix);
Img_grey = rgb2gray(img_resize);
Img_Biner = im2biner(Img_grey,0.3);
```

Variant LBP can process images with larger sizes because variant LBP divides an image into several zones, to divide these zones in MATLAB using cellsize. Each image that has been preprocessed and added with contrast will be divided into several zones with cell sizes of 32, 64, and 128. After dividing the image into several zones, each zone will produce extraction or information which will later be combined into 1 matrix of overall extraction results from an image, the extraction results of each divided zone, the results of LBP extraction in the form of a histogram, the histogram results are used for parameters predict the Multi Support Vector Machine. In this study, testing an image with cellsize 32,64,128 to be able to produce extraction values from LBP using the MATLAB function as follows :

```
cellsize32 = [32 32];
cellsize64 = [64 64];
cellsize128 = [128 128];
lbpFeatures_32_latih=extractLBPFeatures(c,'CellSize',cellsize32,
'Normalization','No ne');
```

```
lbpFeatures_64_latih=extractLBPFeatures(c,'CellSize',cellsize64,
'Normalization','No ne');
lbpFeatures_128_latih=extractLBPFeaturesc,'CellSize',cellsize128
,'Normalization','N one');
training32(n,:)=lbpFeatures_32_latih;
training64(n,:)=lbpFeatures_64_latih;
training128(n,:)=lbpFeatures_128_latih;
```

In order to obtain predictive results from Multi SVM based on data training has been prepared, before the image is tested, the image will also be processed preprocessing and through the LBP extraction process so that it can produce data in the form of information which will be used for data parameter testing of the Multi SVM algorithm. After all the training data is processed and accommodated in the Multi SVM training data variable, the next step is testing using the test data that has been prepared. All test data will go through a process starting from preprocessing to LBP extraction, then digital images will be recognized or predicted using custom functions. Then the results of multiSvm () will be accommodated in a variable, namely out which is used as a parameter to predict the image of the Javanese script. When the find() function is called with the out parameter and the testing data label, it will display the predicted results from the Multi SVM algorithm.

```
disp('Prediksi32');
[out] = multiSvm(training32,group,sample32);
[a,~] = find(out==group_test);
```

Table 3. Comparison of accuracy using MultiSVM and MultiSVM-LBP

Cellsize	Accuracy (%)		
	LBP	MultiSVM	MultiSVM-LBP
32	62	73.5	77
64	69	82	90
128	65	79	88.5
Average	65.33	78.16	85.16

In Table 3, the highest accuracy was obtained from Cellsize 64, which was 90%, while the lowest accuracy occurred in Cellsize 32, which was 77%. From the results of the research conducted, regarding handwriting pattern recognition by combining the LBP (Local Binary Pattern) extraction algorithm and for the classification algorithm, namely Multi SVM so that it trains the system to be able to recognize and predict the types of Javanese script optimally. Starting from the initial character image where the image has a different size, then it is uniformized to a size of 256x256, then the RGB image is processed into a grayscale image and then into binary. Binary images are then processed using the LBP algorithm using cells size 32, 64, and 128 in the hope of getting extraction results or information in an image, where the results of these values can be used for parameter values to process Multi SVM classification. In testing using the LBP and Multi SVM methods with 800 datasets (600 data training and 200 data testing) with 20 different fonts, cellssize = 64 produces the highest accuracy rate of 90% compared to other cellsizes.

4. CONCLUSION

LBP with Cellsize 64 produces an accuracy rate of 90% and Cellsize 128 has an accuracy rate of 88.5% while Cellsize 32 has an accuracy rate of 77%. The combination of the LBP

algorithm with the Multi SVM Algorithm as a classification algorithm produces a fairly high accuracy of 90% so that it can be used as a system for recognizing Javanese script. To improve the results and refinement in this study, there are several suggestions that can be used in further research, namely using a combination of other algorithms to produce maximum accuracy. In future research, you can try using additional threshold algorithms, such as canny or others, in order to increase the level of accuracy. Our algorithm still has deficiencies with results that are not optimal, so if this research is developed further, researchers may need to try several kernels in SVM and Multi SVM to analyze the results of preprocessing-based processing.

5. ACKNOWLEDGEMENT

REFERENCES

[1] I. Prihandi, I. Ranggadara, S. Dwiasnati, Y. S. Sari, and Suhendra, "Implementation of Backpropagation Method for Identified Javanese Scripts," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, 2020. doi: 10.1088/1742-6596/1477/3/032020.

[2] I. Ferdiansyah Katili, F. Dyah Esabella, and A. Luthfiarta, "Pattern Recognition Of Javanese Letter Using Template Matching Correlation Method," *Journal of Applied Intelligent System*, vol. 3, no. 2, pp. 49–56, 2018.

[3] A. Susanto, C. Atika Sari, I. U. W. Mulyono, and M. Doheir, "Histogram of Gradient in K-Nearest Neighbor for Javanese Alphabet Classification," *Scientific Journal of Informatics*, vol. 8, no. 2, pp. 289–296, Nov. 2021, doi: 10.15294/sji.v8i2.30788.

[4] G. A. Robby, A. Tandra, I. Susanto, J. Harefa, and A. Chowanda, "Implementation of Optical Character Recognition using Tesseract with the Javanese Script Target in Android Application," *Procedia Comput Sci*, vol. 157, pp. 499–505, 2019, doi: 10.1016/j.procs.2019.09.006.

[5] A. Susanto, D. Sinaga, C. A. Sari, E. H. Rachmawanto, D. Rosal, and I. M. Setiadi, "A High Performace of Local Binary Pattern on Classify Javanese Character Classification," *Scientific Journal of Informatics*, vol. 5, no. 1, pp. 2407–7658, 2018, [Online]. Available: http://journal.unnes.ac.id/nju/index.php/sji

[6] L. D. Krisnawati and A. W. Mahastama, "Building classifier models for on-off javanese character recognition," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Dec. 2019. doi: 10.1145/3366030.3366050.

[7] N. Irsalinda, S. Surono, I. Dwi, and R. Sary, "Pattern Recognition using Multiclass Support Vector Machine Method with Local Binary Pattern as Feature Extraction," 2022. [Online]. Available: https://doi.org/11.26554/sti.2222.7.3.269-274

[8] M. Daniel, J. Raharjo, and K. Usman, "Iris-based Image Processing for Cholesterol Level Detection using Gray Level Co-Occurrence Matrix and Support Vector Machine," *Engineering Journal*, vol. 24, no. 5, pp. 135–144, Sep. 2020, doi: 10.4186/ej.2020.24.5.135.

[9] D. U. K. Putri, D. N. Pratomo, and Azhari, "Hybrid convolutional neural networks-support vector machine classifier with dropout for Javanese character recognition," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 21, no. 2, pp. 346–353, Apr. 2023, doi: 10.12928/TELKOMNIKA.v21i2.24266.

[10] F. Damayanti, Y. K. Suprapto, and E. M. Yuniarno, "Segmentation of Javanese Character in Ancient Manuscript using Connected Component Labeling," in *CENIM 2020 - Proceeding: International Conference on Computer Engineering, Network, and Intelligent Multimedia 2020*, Institute of Electrical and Electronics Engineers Inc., Nov. 2020, pp. 412–417. doi: 10.1109/CENIM51130.2020.9297954.

[11] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh, and B. Yoon, "Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN).," *Sensors (Basel)*, vol. 20, no. 12, pp. 1–18, Jun. 2020, doi: 10.3390/s20123344.

[12] A. K. Nugroho, I. Permadi, and M. Faturrahim, "Improvement Of Image Quality Using Convolutional Neural Networks Method," *Scientific Journal of Informatics*, vol. 9, no. 1, pp. 95–103, May 2022, doi: 10.15294/sji.v9i1.30892.

[13] Hassan M. Najadat, Ahmad A. Alshboul, and Abdullah F. Alabed, "Arabic Handwritten Characters Recognition usingConvolutional Neural Network," in *10th International Conference on Information and Communication Systems (ICICS*, 2019, pp. 147–151.

[14] I. Khandokar, M. Hasan, F. Ernawan, S. Islam, and M. N. Kabir, "Handwritten character recognition using convolutional neural network," *J Phys Conf Ser*, vol. 1918, no. 4, p. 042152, Jun. 2021, doi: 10.1088/1742-6596/1918/4/042152.

[15] T. Q. Vinh, L. H. Duy, and N. T. Nhan, "Vietnamese handwritten character recognition using convolutional neural network," *IAES International Journal of Artificial Intelligence*, vol. 9, no. 2, pp. 276–283, Jun. 2020, doi: 10.11591/ijai.v9i2.pp276-283.

[16] R. Parthiban, R. Ezhilarasi, and D. Saravanan, "Optical Character Recognition for English Handwritten Text Using Recurrent Neural Network," in *2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*, IEEE, Jul. 2020, pp. 1–5. doi: 10.1109/ICSCAN49426.2020.9262379.

[17] N. B. Muppalaneni, "Handwritten Telugu Compound Character Prediction using Convolutional Neural Network," in *International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, 2020, pp. 1–4. doi: 10.21227/phg6-m127.

[18] A. Susanto, C. Atika Sari, I. U. W. Mulyono, and M. Doheir, "Histogram of Gradient in K-Nearest Neighbor for Javanese Alphabet Classification," *Scientific Journal of Informatics*, vol. 8, no. 2, pp. 289–296, Nov. 2021, doi: 10.15294/sji.v8i2.30788.

[19] D. Sutaji and R. Dijaya, "Classification of Milk Fish Quality using Fuzzy K-Nearest Neighbor Method Based on Form Descriptor and Co-Occurrence Matrix," *J Phys Conf Ser*, vol. 1179, no. 1, p. 012021, Jul. 2019, doi: 10.1088/1742-6596/1179/1/012021.

[20] S. Winiarti, F. I. Indikawati, A. Oktaviana, and H. Yuliansyah, "Consumable Fish Classification Using k-Nearest Neighbor," in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, May 2020. doi: 10.1088/1757-899X/821/1/012039.

[21] C. K. Dewa, A. L. Fadhilah, and A. Afiahayati, "Convolutional Neural Networks for Handwritten Javanese Character Recognition," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 12, no. 1, p. 83, Jan. 2018, doi: 10.22146/ijccs.31144.

[22] Y. Sugianela and N. Suciati, "Javanese Document Image Recognition Using Multiclass Support Vector Machine," *Communication & Information Technology) Journal*, vol. 13, no. 1, pp. 25–30, 2019.