ASTJ Journal

Applied Systems and Technologies Journal (ASTJ)

Recognition of Javanese Month Names Written in Javanese Script Using The Backpropagation Method and Wavelet Transform Feature Extraction

Kafiyanto Hanafi Yahya [1]*, Ahmad Kamsyakawuni [1], Abduh Riski [2]
*correspondence: khafi.hanafi@gmail.com
[1] Department of Mathematics, Faculty of Mathematics and Natural Sciences
Universitas Negeri Jember
Jember, Indonesia

Abstract- Javanese script is one of Indonesia's cultural heritages. It was once widely used as a means of communication, but as time has progressed, Javanese script has come to be used far less. This decline reflects a decrease in public interest in learning Javanese script. A pattern-recognition model is therefore needed to help a computer system recognize words written in Javanese script. Such a model can be built using the backpropagation neural network method combined with wavelet transform feature extraction. The training data consisted of 560 images collected by the author, divided into 14 classes, while the testing data consisted of handwritten samples from 5 different respondents. The data were first segmented to separate the individual characters; feature extraction was then performed using the wavelet transform, and the resulting features were evaluated with the backpropagation network to determine recognition accuracy. The results show that combining wavelet transform feature extraction with the backpropagation algorithm gives good performance, achieving the highest accuracy of 84.18%, correctly recognizing 181 of 215 test images.

Keywords: Javanese Script; Artificial Neural Network; Wavelet Transform; Word Recognition.

Article info: *submitted October 18, 2025, revised November 29, 2025, accepted December 8, 2025*

1. Introduction

Javanese script is one of Indonesia's cultural heritages. It differs from the Latin alphabet, which is commonly used in both spoken and written communication. The Latin alphabet is alphabetic in nature, meaning that vowels are required to support pronunciation, whereas Javanese script is syllabic, in that each character can be pronounced even when it stands alone. According to [3], the carakan (Javanese alphabet) used in Javanese spelling essentially consists of twenty basic syllabic characters. Learning Javanese script is highly valuable for gaining a fuller understanding of Javanese culture as a whole. Javanese script was once widely used as a means of communication, but as time has progressed it has been used far less. Today, ancient Javanese characters can be automatically converted into Latin-based Javanese transliteration using a computer system.

Computer systems are also used to solve complex problems, including image processing and pattern recognition. Image processing is a field concerned with processing images to improve their quality so that they can be more easily interpreted by humans or computers [6]. Pattern or object recognition in digital images has advanced rapidly alongside developments in computer technology. Character-recognition systems, for instance, have been widely developed for Latin, Arabic, Chinese, and Japanese scripts. Pattern recognition generally consists of two main steps: preprocessing and feature extraction. Preprocessing serves a variety of purposes, including resizing, noise reduction, grayscale conversion, smoothing, cropping, and binarization. In this study, the researchers used cropping, resizing, and binarization. For feature extraction, many methods are available, one of which is the wavelet transform.

A transform is a process that converts data or a signal into another form to make it easier to analyze. The Fourier transform, for example, converts a signal into a set of sine or cosine waves of different frequencies, whereas the wavelet transform converts a signal into various forms of a wavelet basis (mother wavelet) through different shifts and scales [10]. After feature extraction, the next step in pattern recognition is classifying the image data. Many methods can be used for classification, one of which is the artificial neural network (ANN). An artificial neural network is a pattern-recognition approach in which the network is a computing system designed to mimic the natural processes that occur in the biological neural networks of the human brain [4].

Artificial neural networks can be trained using several methods, one of which is backpropagation. This method aims to minimize the error occurring within the network while also training the network toward a balanced state. The algorithm consists of three training stages: forward propagation, backward propagation, and weight adjustment. Forward propagation proceeds from the input layer to the output layer, while backward propagation runs in the opposite direction, from the output layer back to the input layer. The third stage, weight adjustment, is carried out to reduce the error.

Several previous studies have applied the backpropagation algorithm to character-pattern recognition. [7] developed a Javanese script image-recognition application using backpropagation with wavelet transform as the initial image-processing step, achieving 97.857% accuracy on test images included in the training data and 45% accuracy on test images not included in the training data. [8] recognized handwritten Javanese characters using a backpropagation neural network, achieving 84.1% accuracy. Later, in 2018, [9] performed handwritten Javanese script pattern recognition with the backpropagation algorithm, achieving 83% accuracy. These results indicate that backpropagation is effective at reducing the error rate in object recognition. A further study by [2] applied a backpropagation neural network with wavelet features to Nglegena script, achieving 86.40% accuracy, while [5] carried out face recognition using backpropagation based on the discrete wavelet transform, achieving 90% accuracy. These findings, obtained after wavelet feature extraction, show that wavelet-based features yield high accuracy. Building on this, the present article applies the wavelet transform to the problem of recognizing words composed of several Javanese characters. The article aims to apply the wavelet transform to pattern recognition in digital images while also designing a backpropagation neural network architecture for this purpose.

2. Research Methods

This study was carried out using MATLAB 2015b. The research data were divided into two groups: training data and testing data. The training data consisted of images of the basic Javanese characters (Nglegena) together with their diacritical marks (sandhangan), while the testing data consisted of handwritten samples obtained from 5 respondents. A total of 560 training images were used, divided into 14 classes. The testing data consisted of 60 images, made up of 12 samples of Javanese month names written by 5 different respondents. These handwritten samples were used to evaluate the success of the segmentation and recognition processes.

The steps used for recognizing Javanese script words followed the flow described below.

1) Literature Review: Collecting and reviewing literature related to image processing, backpropagation, and wavelet transform feature extraction.

2) Data Acquisition: Collecting samples of the basic Javanese characters (Nglegena) together with their diacritics and rewriting them to match the test data, as well as recruiting respondents to obtain handwritten samples from several individuals.

3) Data Grouping: Before grouping, the data were first scanned, since they consisted of handwritten samples. The data were then grouped and organized into folders based on their type and label, i.e., training data and testing data.

4) Preprocessing: The preprocessing stage involved several steps: image cropping, binarization, segmentation, and resizing. Cropping was used to cut a specific part of the image to obtain the desired region, with the crop point determined at the center of the target area. Binarization was then applied to convert the grayscale image into a black-and-white image, making it possible to identify the object regions within the image. The final preprocessing step was segmentation, which aimed to split the image into its individual characters; this step was applied only to the test images. The segmented results were then automatically resized in pixels.

5) Feature Extraction: Feature extraction is the initial step in classifying an image. Good features are those with strong discriminative power, allowing pattern classification to be performed with high accuracy. Wavelet transform feature extraction works through image decomposition performed across the rows and columns of a two-dimensional array, corresponding respectively to the horizontal and vertical directions of the image. This produces the image's subband frequencies, whose components are generated by successively lowering the decomposition level — a process also known as the variable windowing technique.

6) Data Processing with the Backpropagation Algorithm: The backpropagation neural network was used during both training and testing to recognize Javanese script patterns. The test results were then stored in the program.

7) Result Analysis: The number of correctly recognized characters was examined, after which the accuracy was calculated and the best-performing result was selected.

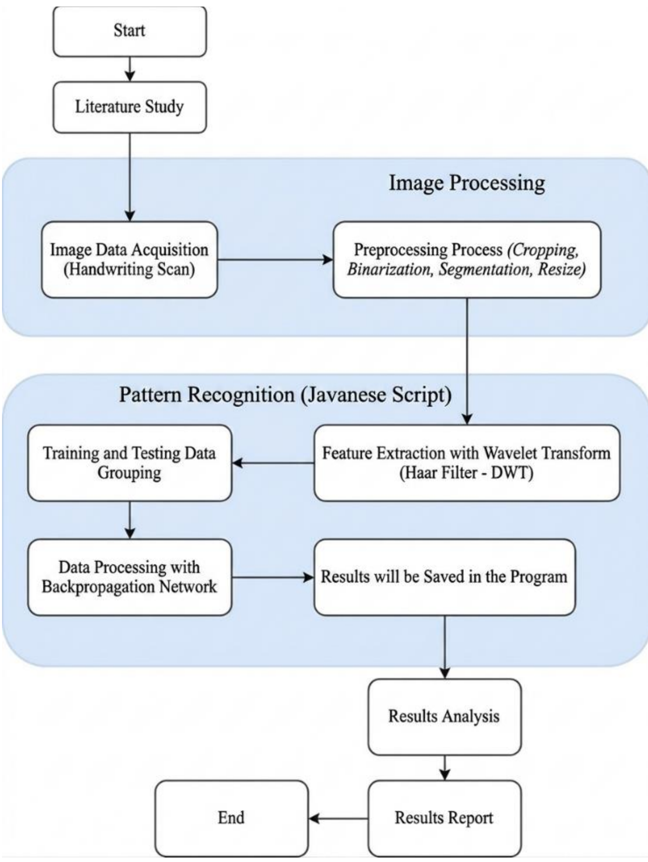

Figure 1. The research flow diagram

3. Result and Discussion

The results below explain how words consisting of several Javanese characters were recognized, using data on Javanese month names.

1. Data Collection

The data used consisted of images of Javanese script (Javanese month names). The images were captured using a smartphone camera with the help of a scanning application. Examples of the captured images are shown in Figure 2.

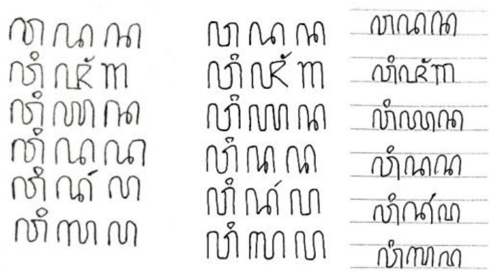

Figure 2. Sample images captured with a smartphone

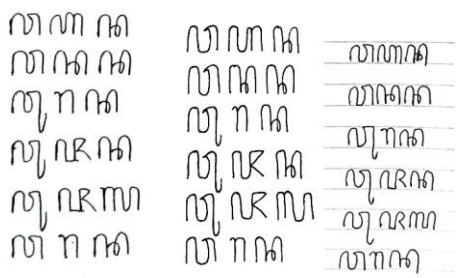

Figure 3. Sample images captured with a smartphone

The image preprocessing stage aims to improve image quality and to adapt the images to the needs of the study. The preprocessing steps carried out were as follows.

1) Cropping

Image cropping was performed manually using the Paint application. This step is essential for focusing on the target image, since a single captured image often contains many different Javanese script expressions. Cropping was used to separate these expressions from one another. An example of a cropped image is shown in Figure 3.

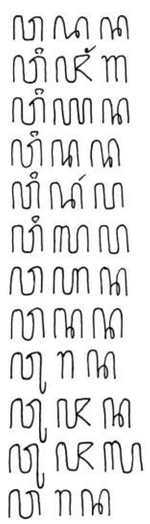

Figure 4. Result of the image cropping process

2) Segmentation

Segmentation was performed by implementing a segmentation algorithm in MATLAB. After determining the binarized values of the image, the regionprops function available in MATLAB was used to define the pixel boundaries of each segment. The separated images were then uniformly resized to 45×45 pixels. The segmentation results are shown in Figure 5.

Figure 5. Result of the segmentation process

2. Feature Extraction Calculation

Before classification, the preprocessed images undergo feature extraction using the wavelet transform with a Haar filter. The transform was performed with two levels of decomposition (level-2 decomposition). The example below shows a matrix taken from an image of the character wa, simplified into an 8×8 matrix formed from pixel rows 1–8 and pixel columns 9–16, in order to simplify the calculation.

255	255	252	0	9	1	235	255
243	0	0	15	6	0	26	245
23	5	0	0	0	0	10	0
0	4	11	1	1	3	3	16
0	0	15	241	255	253	0	0
13	0	241	255	255	239	12	4
0	248	255	238	249	255	0	0
255	255	245	255	255	250	12	1

One method for computing the Haar wavelet transform of n samples is as follows [1]:

- Compute the average of each pair of samples (n/2 averages).
- Compute the difference between each average and the corresponding sample (n/2 differences).
- Write the averages into the first half of the array.
- Write the differences into the second half of the array.
- Repeat the process on the first half of the array, halving the array size at each repetition.

In the first step, the average value is computed for each pair of pixels (samples) in the image. The average of each pair of columns is computed first — for example, column 1 plus column 2 divided by two, and so on through column n — with the results placed on the left side, shown by the pixels highlighted in red in the original data. On the right side, column 1 minus column 2 divided by two is computed. The results of this first-stage pairwise averaging are shown below.

255	126	5	245	0	126	4	-10
121.5	7.5	3	135.5	121.5	-7.5	3	-109.5
14	0	0	5	9	0	0	5
2	6	2	9.5	-2	5	-1	-6.5
0	128	254	0	0	-113	1	0
6.5	248	247	8	6.5	-7	8	4
124	246.5	252	0	-124	8.5	-6	0
255	250	252.5	6.5	0	-5	2.5	5.5

Note: blue indicates the approximation coefficients; red indicates the detail coefficients.

In the second step, the previous procedure is repeated, but the pairwise averaging is now performed from top to bottom along each column. The results of this second-stage pairwise averaging are shown below.

188.25	66.75	4	190.25	60.75	59.25	3.5	-59.75
8	3	1	7.25	3.5	2.5	-0.5	0.5
3.25	188	250.5	4	3.25	-60	4.5	2
189.5	248.5	252.25	3.25	-62	1.75	-1.75	2.75
66.75	59.25	1	54.75	-60.75	66.75	3.5	49.75
6	-3	-1	-2.25	5.5	-2.5	0.5	5.75
-3.25	-60	3.5	-4	-3.25	-53	-3.5	-2
-65.5	-3.5	-0.25	-3.25	-62	6.75	-4.25	-2.75

Note: blue (LL) denotes the approximation component; yellow (HL) denotes the horizontal detail component; green (LH) denotes the vertical detail component; red (HH) denotes the diagonal detail component.

In the third step, to continue the compression/decomposition process, a submatrix of size (rows/2) × (columns/2) is extracted from the initial matrix. This submatrix corresponds to the approximation matrix, the result of Haar-based compression. The resulting compressed matrix for the image of the character wa is shown below.

188.25	66.75	4	190.25
8	3	1	7.25
3.25	188	250.5	4
189.5	248.5	252.25	3.25

Next, to perform the level-2 decomposition, the same pairwise-averaging procedure described above is applied again. The result of the level-2 decomposition for the image of wa is shown below.

66.75	50.625	63.25	-48.125
157.3125	127.5	-60.9375	123.875
60.75	46.5	29.125	-45
-61.6875	-0.25	-31.4375	-0.625

The approximation component of the level-2 decomposition for the image of wa is shown below.

66.75	50.625
157.3125	127.5

3. *Pattern Identification/Recognition*

Pattern identification was carried out using a backpropagation neural network. The parameters used to construct the network were as follows:

- Input layer: 576 nodes, derived from the wavelet transform feature extraction process.
- Hidden layer: 14 different configurations were tested, as shown in Table I.
- Output layer: one output layer.
- Target: 14 target classes — wa, da, na, i, ja, ng, ga, ya, r, pa, la, ha, wu, ra.
- Activation function: three MATLAB functions were used — tansig (bipolar sigmoid), logsig (binary sigmoid), and purelin (linear/constant function).
- Training method: Backpropagation Conjugate Gradient with Powell–Beale restarts ('traincgb') and Backpropagation Gradient Descent with Momentum and Adaptive Learning Rate ('traingdx').
- Learning rate: 0.5.
- Epochs: 10,000.
- Maximum validation failures: 10 (the maximum number of allowable validation failures, used when validation is included as an optional criterion; training stops once the number of failures exceeds this maximum; default = 5).

• Performance goal: 1×10^{-10} (the MSE threshold at which training stops; training halts once the MSE falls below this threshold or the maximum number of epochs is reached)

4. *Program Implementation*

The entire process was carried out using MATLAB R2015b. After testing different parameter combinations, as summarized in Table I, several key parameters were identified for finding the optimal solution. The highest recognition rate was obtained using the tansig activation function with 3 hidden layers, 50 neurons per layer, and the traincgb training method (Backpropagation Conjugate Gradient with Powell–Beale restarts), correctly recognizing 181 of 215 image segments

Table 1. Recognition accuracy by network configuration

Hidden Layers	Neurons per Layer	Activation Function	No. of Segments	Training Method	No. Recognized	Recognition Rate
1	30	Logsig	215	Traincgb	162	75.34%
1	50	Logsig	215	Traincgb	165	76.74%
2	30, 50	Logsig	215	Traincgb	163	75.81%
2	50, 50	Logsig	215	Traincgb	175	81.39%
3	40, 30, 30	Logsig	215	Traincgb	165	76.74%
3	20, 30, 50	Logsig	215	Traincgb	146	67.90%
3	50, 50, 50	Logsig	215	Traincgb	174	80.93%
4	25, 25, 25, 25	Logsig	215	Traincgb	165	76.74%
4	50, 50, 50, 50	Logsig	215	Traincgb	170	79.06%
4	90, 90, 90, 90	Logsig	215	Traincgb	160	74.41%
3	50, 50, 50	Purelin	215	Traincgb	91	42.32%
2	50, 50	Tansig	215	Traincgb	172	80.00%
3	50, 50, 50	Tansig	215	Traincgb	181	84.18%
3	50, 50, 50	Tansig	215	Traingdx	107	49.76%
Average						72.95%

The segmentation process performed well, successfully splitting the 60 images into 215 segments corresponding to the number of characters present. However, a limitation of this segmentation approach lies in the thickness and separation of the strokes. For example, a character with a separate diacritical mark, such as wi, is split into two distinct images: the base character wa and the diacritic i. The patterns across the images also vary considerably in complexity. The large number of training trials was intended to identify a network architecture capable of recognizing the pattern of each image used.

As shown in Table I, the best-performing network architecture used the tansig activation function with 3 layers, 50 neurons per layer, and traincgb training, outperforming all other configurations. The best test result achieved a recognition rate of 84.18%, correctly identifying 181 of 215 images. This result was obtained through a series of trials, beginning with variations in the number of hidden layers and the number of neurons per hidden layer. Table I summarizes the 14 trials conducted: after identifying a configuration with improved accuracy, subsequent trials varied the activation function, and finally the training method. The weight and bias values used in this study were assigned automatically by MATLAB, so the weights and biases differed across each trial.

4. Conclusion

Based on the results and discussion, the following conclusions can be drawn:

- The filter used for initial image processing was the Haar filter, one of the filters within the Discrete Wavelet Transform (DWT). Decomposition was limited to level 2.
- The best backpropagation network architecture obtained consisted of 3 layers with 50 neurons per layer, using the tansig (bipolar sigmoid) activation function, a learning rate of 0.5, and the traincgb training method.
- The highest accuracy achieved with the backpropagation algorithm was 84.18%, with 181 of 215 test images correctly recognized, 15 images unrecognized, and 19 images misclassified.
- Combining wavelet transform feature extraction with the backpropagation algorithm still has certain limitations, namely characters that fail to be recognized. Possible causes include handwriting quality, image-capture quality, the quality of the character-separation (segmentation) process, and the feature-extraction process itself.

References

[1] Arora, S., Brar, Y. S., Kumar, S. 2014. Haar Wavelet Transform For Solution Of Image Retrieval.International Journal of Advanced and Mathematical Science, vol 5, pp 27-31

[2] Arum, N. B. W. 2016. Pengenalan Pola Aksara Jawa Nglegena berbasis Wavelet dan Jaringan Syaraf Tiruan Backpropagation. Skripsi. Yogyakarta: Universitas Gadjah Mada.

[3] Darusuprapta, Harjana, H. S. U., Nursatwika, Subalidinata, Hadiatmadja, S. 2002. Pedoman Penulisan Aksara Jawa. Yogyakarta: Yayasan Pustaka Nusatama.

[4] Fausett, L. 1994. Fundamentals of Neural Networks Architectures, Algorithms, and Applications. London: Prentice Hall, Inc.

[5] Firdausi, A. I. 2019. Pengenalan Wajah dengan Algoritma Backpropagation Berdasarkan Transformasi Wavelet Diskrit. Skripsi. Jember: Universitas Jember.

[6] Sabri. A. 2016. Pengolahan Citra. Depok: Universitas Gunadarma.

[7] Weisling, Daphne, E. J. 2011. Pembangunan Aplikasi Pengenalan Citra Aksara Jawa Menggunakan Metode Backpropagation Dengan Wavelet Sebagai Pemrosesan Awal Citra. Skripsi. Yogyakarta. Universitas Atma Jaya Yogyakarta.

[8] Wibowo, A. 2012. Pengenalan Huruf Jawa Tulisan Tangan Menggunakan Jaringan Saraf Tiruan Perambatan Balik Dengan Fuzzy Feature Extraction. Skripsi. Semarang: Universitas Diponegoro.

[9] Wibowo, A. T. 2018. Pengenalan Pola Tulisan Tangan Aksara Jawa dengan Algoritma Backpropagation. Skripsi. Yogyakarta: Universitas Sanata Dharma

[10] Wijaya, I. G. P. S. dan Bulkis K. 2004. Pengenalan Citra Sidik Jari Berbasis Transformasi Wavelet dan Jaringan Syaraf Tiruan. Universitas Mataram: Departemen Teknik Elektro. 46-52.