Cosine Similarity as a Distance Metric for Javanese Script Image Recognition Classification

Aji Priyambodo[1], Prihati[1], Danang[2], Farhan bin Mohamed[3]
[1]Institut Teknologi dan Bisnis Semarang, Semarang, Indonesia
[2]Universitas Sains dan Teknologi Komputer, Semarang, Indonesia
[3]Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia

Article Info

Article history:

Received June 05, 2024
Revised March 19, 2025
Accepted February 24, 2026

Keywords:

Cosine similarity;
Distance metric;
Image recognition;
Javanese script classification.

ABSTRACT

Javanese character (Hanacaraka) recognition presents significant challenges due to the intricate patterns and variations in character features. Addressing these issues is crucial for digitizing cultural heritage and supporting educational applications. This study aims to evaluate the effectiveness of cosine similarity as a distance metric for classifying Javanese characters, comparing its performance against traditional Euclidean and Manhattan distance metrics. The research used a feature-extraction technique based on the histogram of oriented gradients and evaluated cosine similarity across different classification models. Model performance was assessed using precision, recall, F1-score, and accuracy metrics. The results showed that cosine similarity, when combined with a support vector machine, achieved an accuracy of 99.84%, significantly outperforming other distance metrics. When applied to another classification model, cosine similarity improved accuracy to 90%, demonstrating its robustness in handling complex patterns. Parameter optimization was performed using a grid-based search, and model reliability was assessed through cross-validation. Compared with previous studies that primarily relied on deep learning, this research offers an alternative method that balances efficiency and accuracy while maintaining high interpretability. The findings establish a new benchmark for Javanese character recognition and highlight the potential of cosine similarity in broader applications. Future research can expand this study by incorporating more diverse feature extraction techniques, larger datasets, and hybrid approaches to further enhance recognition performance.

Corresponding Author:

Aji Priyambodo, +62 895-3924-99955,
Information Systems and Technology,
Semarang Institute of Technology and Business, Semarang, Indonesia,
Email: ajippro@students.undip.ac.id., ajipro@itbs.ac.id

How to Cite:
A. P. Priyambodo, P. Prihati, D. Danang, and Farhan bin Mohamed, "Cosine Similarity as a Distance Metric for Javanese Script Image Recognition Classification", *MATRIK: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, Vol. 25, No. 2, pp. 232-334, March, 2026.

1. INTRODUCTION

Natural language processing (NLP) technology in Indonesia has made significant progress in recent years. However, the recognition of Javanese script, known as Hanacaraka [1], remains underexplored despite its profound cultural and historical significance. Hanacaraka represents a distinctive aspect of Indonesia's heritage and holds considerable potential for computational research. Existing studies [1–8], have addressed various aspects of image-based character recognition, focusing on traditional or hybrid models. Nevertheless, limited attention has been paid to integrating advanced similarity metrics, such as cosine similarity, which could significantly improve the accuracy and efficiency of recognition systems. This gap highlights the need for innovative research to enhance Javanese script recognition while preserving Indonesia's cultural heritage through technology. Without accurate and efficient recognition systems, the use of Hanacaraka may continue to decline, especially in education and digital communication. By incorporating advanced similarity metrics, such as cosine similarity, this research can enhance the precision of recognition models, enabling broader applications in cultural preservation, historical documentation, and smart learning systems. Therefore, this study is crucial not only for advancing computational techniques but also for safeguarding Indonesia's rich linguistic and cultural heritage in the digital era.

Several studies have contributed to the development of character recognition systems, employing a range of techniques. For instance, approaches such as combining Local Binary Pattern (LBP) with Support Vector Machine (SVM) have achieved an accuracy of approximately 87.86% [8], hybrid convolutional neural networks (CNN)-Support Vector Machine (SVM) models have demonstrated recognition accuracy of up to 98.35% [4], and advanced preprocessing techniques have also shown moderate success, reaching accuracies 98% [5]. Other methods, such as Random Forest, reported accuracies of around 97.7% [2], the integration of K-Nearest Neighbor (KNN) with Local Binary Pattern (LBP) achieved approximately 90.5% [1], and deep convolutional neural networks (CNN) have further improved performance, with reported accuracies of up to 99.65% [3], have shown promising results on various datasets, including those involving Javanese script. Despite these advancements, there has been limited exploration of similarity metrics, such as cosine similarity, in this domain. Over the past five years, several studies have demonstrated the effectiveness of cosine similarity in tasks such as image classification and reconstruction [9–16] reporting accuracy improvements of up to 26-28% on datasets like Omniglot and MiniImageNet. These findings suggest significant potential for cosine similarity to address challenges in complex character recognition tasks.

Despite these advancements, critical gaps remain in the application of advanced similarity metrics, particularly cosine similarity, for Javanese script recognition. Most existing studies rely on traditional distance measures and hybrid models, without fully exploring the potential of similarity-based techniques to improve both accuracy and efficiency. Moreover, there is a lack of a comprehensive framework that integrates advanced preprocessing, feature extraction, and classification techniques optimized with cosine similarity. This research seeks to bridge these gaps by introducing cosine similarity as a core distance metric in Javanese script recognition. The novelty of this study lies in its ability to outperform traditional methods and systematically compare its results with six recent approaches: Local Binary Pattern (LBP) - Support Vector Machine (SVM), hybrid convolutional neural networks (CNN) - Support Vector Machine (SVM), preprocessing-based analysis, Random Forest, K-Nearest Neighbor (KNN) - Local Binary Pattern (LBP), and deep convolutional neural networks (CNN) models.

The primary objective of this research is to develop a robust framework for Javanese script recognition by leveraging cosine similarity and feature extraction techniques such as Histogram of Oriented Gradients (HOG). This involves enhancing the quality of Javanese script datasets, extracting discriminative features, and integrating advanced classification algorithms to achieve higher accuracy. The performance of this framework is evaluated systematically to ensure its effectiveness and reliability. By addressing the limitations of existing methods, this research aims to provide a novel and efficient solution for Javanese script recognition.

2. RESEARCH METHOD

The methodology implemented in this study is structured to systematically evaluate the effectiveness of cosine similarity in classifying Javanese script images. The research process comprises six primary stages: data collection, preprocessing, feature extraction, cosine similarity computation, classification, and evaluation. Each stage has been carefully designed to address the dataset's specific requirements and the study's objectives. The detailed steps are outlined below and visualized in Figure 1, a comprehensive flowchart of the methodology.

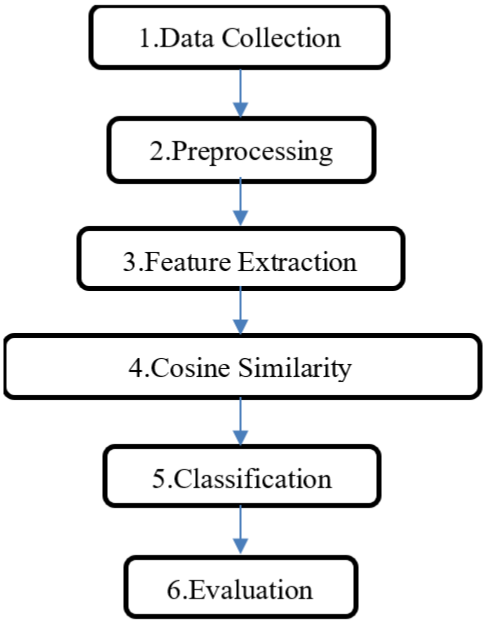

Figure 1. A comprehensive flowchart of the methodology

2.0.1. Data Collection

The study begins by collecting a dataset of Javanese script images. The dataset, sourced from Kaggle, includes 4,385 images representing 20 distinct classes of Javanese characters. These classes encompass a wide range of script patterns, including characters such as ba, ca, da, dha, ga, ha, ja, ka, la, ma, na, nga, nya, pa, ra, sa, ta, tha, wa, and ya. This diversity ensures a representative sample for the classification task. The dataset is uploaded to Google Drive for ease of access and integration into the Python programming environment. Google Colab is used as the primary development platform due to its support for GPU-based computations, which are crucial for efficiently processing large image datasets.

2.1. Preprocessing

Preprocessing is a critical step for standardizing the dataset and preparing it for feature extraction. Each image in the dataset undergoes the following transformations: Grayscale Conversion: To reduce computational complexity and focus on structural details, all images are converted to grayscale. This eliminates color information, which is not essential for classifying Javanese script. Resizing: Each image is resized to a fixed 64x64-pixel dimension. This uniform size ensures consistency across the dataset and facilitates efficient feature extraction. An example of a preprocessed image is depicted in Figure 2, illustrating the transformation from the original image to its grayscale and resized form. This step ensures that the input data is both compact and consistent, which is vital for the subsequent stages.

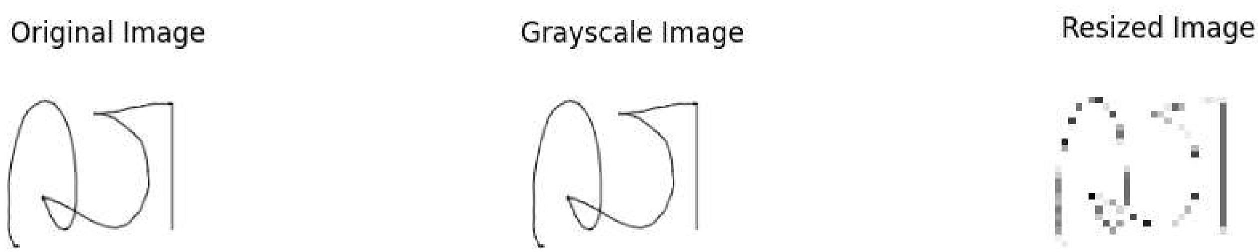

Figure 2. Example of transformed image

2.2. Feature Extraction

The next stage involves extracting distinguishing features from each image to capture the unique characteristics of Javanese script. The chosen method for feature extraction is the Histogram of Oriented Gradients (HOG) [17], a widely recognized technique for encoding shape and structure information in images. Histogram of Oriented Gradients (HOG) is particularly well-suited for character recognition tasks due to its ability to highlight gradient patterns and orientations. Steps in Histogram of Oriented Gradients (HOG) Feature Extraction: Grayscale Conversion: Ensures uniformity in pixel intensity values. Gradient Calculation: Image gradients are computed in the x and y directions using the Sobel operator, as defined in Equation (1): where I is the pixel intensity of the image.

$$G_x = \frac{\partial I}{\partial x}, G_y = \frac{\partial I}{\partial v} \tag{1}$$

The gradient magnitude and orientation are essential components derived from the x and y gradients of an image. These gradients, calculated for each pixel, represent the rate of intensity change in horizontal and vertical directions, providing critical information about edges and boundaries. By combining the x and y gradients, the gradient magnitude measures the strength of the edge, indicating how sharp the intensity transition is at a given pixel. On the other hand, the gradient orientation specifies the direction of the edge, revealing the angle at which the intensity changes occur. Both magnitude and orientation are calculated using mathematical formulas, such as the one represented in Equation (2), to ensure precise edge detection and feature extraction in image processing tasks. These values play a significant role in applications like object recognition and image segmentation, where identifying and analyzing edges is a fundamental step:

$$\begin{aligned} Magnitudo &= \sqrt{G_x^2 + G_y^2} \\ Orientasi &= arctan(\frac{G_y}{G_x}) \end{aligned} \tag{2}$$

Cell Division: The image is divided into cells, typically of size 8×8 pixels. Histogram Creation: For each cell, a histogram of gradient orientations is created, with gradient magnitudes serving as weights. The orientations are grouped into bins (e.g., 9 bins for angles between 0^o and 180^o). Block Normalization: To ensure robustness against lighting and contrast variations, several cells are grouped into blocks (e.g., 2×2). Histograms within each block are normalized using L2 normalization, as defined in Equation (3): where ϵ is a small constant to prevent division by zero.

$$\text{Vektor Histogram Ter - normalisasi} = \frac{\text{Vektor Histogram}}{\sqrt{||\text{Vektor Histogram}||\frac{2}{2} + \varepsilon^2}} \tag{3}$$

Feature vector formation is a crucial step in image representation and classification. In this process, the normalized histograms from all image blocks are concatenated to form a comprehensive feature vector. This final feature vector serves as a condensed yet descriptive representation of the image, encapsulating vital characteristics such as texture, patterns, and structural details. By combining information from all blocks, the feature vector captures the local and global attributes necessary for effective analysis. Specifically, in the case of Javanese character recognition, these feature vectors encode the essential distinctions required to differentiate between various characters. This representation ensures that even subtle variations in structure and design are preserved, making it an integral component in tasks such as classification and pattern recognition. Ultimately, the feature vector provides the foundation for accurate and reliable recognition in advanced machine learning systems.

2.3. Cosine Similarity

Cosine similarity [18] is widely used as a primary metric for assessing feature vector similarity due to its effectiveness at capturing angular relationships. Unlike other metrics, cosine similarity focuses on vector orientation rather than magnitude, making it robust to variations in lighting or contrast that may otherwise affect results. This approach ensures that the similarity measurement remains consistent, even when the overall scale of the feature vectors changes. By analyzing the cosine of the angle between two vectors, this metric provides a precise evaluation of their directional alignment, which is crucial in distinguishing patterns or features. The mathematical calculation of cosine similarity, as outlined in Equation (4), provides a straightforward yet powerful means of comparing feature representations, ensuring reliable performance in tasks such as image classification and pattern recognition. Its robustness and simplicity make it an indispensable tool across many machine learning and data analysis applications.

$$\text{Cosine Distance} = 1 \text{ - Cosine Similarity} \tag{4}$$

For classification purposes, the cosine distance, defined as cosine distance, is used. This metric ranges from 0 to 2: 0: Identical vectors. 1: Orthogonal vectors. 2: Opposite vectors. The cosine similarity module is implemented to facilitate efficient comparison of feature vectors, enabling accurate classification of Javanese script images. The adoption of cosine similarity over traditional distance metrics (e.g., Euclidean) is justified by its specific advantages: Focus on Orientation [9] : Cosine similarity emphasizes the directional alignment of vectors, making it robust to variations in magnitude caused by lighting or contrast changes. Efficiency in High Dimensions [10, 19] : With high-dimensional Histogram of Oriented Gradients (HOG) features, cosine similarity offers computational efficiency and reliable comparisons. Robustness to Sparsity [20]. The metric effectively handles sparse data, which is common in feature extraction tasks involving gradients and orientations. By highlighting these benefits, the study underscores the suitability of cosine similarity for classifying Javanese script images.

2.4. Classification

The classification process is conducted using two powerful algorithms: K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) [21], both are adapted to incorporate cosine similarity for enhanced performance. The first step is to split the dataset into training (80%) and testing (20%) subsets to ensure balanced model evaluation. For the K-Nearest Neighbors (KNN) model, the number of neighbors (3, 5, 7, or 9) is optimized, and different distance metrics (cosine, Euclidean, and Manhattan) are compared to determine the most effective metric for classification. In the case of Support Vector Machine (SVM), a customized cosine kernel is implemented to measure the similarity between feature vectors, with additional hyperparameters such as C and gamma fine-tuned for optimal performance. To further refine the models, GridSearchCV [22] is used for hyperparameter optimization, leveraging 10-fold cross-validation to ensure robust, reliable evaluations. This comprehensive classification strategy ensures that both algorithms are tailored to deliver high accuracy and effective performance for the given task.

2.5. Evaluation

The performance of the proposed methodology is assessed using standard evaluation metrics, including precision, recall, and F1-score, to ensure a comprehensive analysis of its effectiveness. Precision measures the proportion of true positives among predicted positives, while recall evaluates the proportion of true positives among actual positives, offering insights into the model's accuracy and sensitivity. The F1-score, a balanced metric that considers both precision and recall, provides a holistic measure of the model's performance [22]. These metrics collectively highlight the strengths and weaknesses of the proposed approach, enabling a detailed evaluation of how cosine similarity enhances classification accuracy. The methodology establishes a robust framework for Javanese script classification, systematically guiding the process from data collection to performance evaluation. The visual representations in Figures 1 and 2 further clarify the results, reinforcing the significance of cosine similarity as a powerful metric for pattern recognition. This approach demonstrates the potential of integrating advanced similarity measures to improve classification outcomes effectively.

3. RESULT AND ANALYSIS

This section outlines the study's results and provides an interpretation of the findings to assess the performance of cosine similarity as a distance metric for classifying Javanese characters (Hanacaraka). The analysis includes a detailed comparison of performance metrics such as precision, recall, and F1-score to highlight the strengths and limitations of the proposed approach. Key findings are presented to emphasize the importance of cosine similarity in improving classification accuracy, particularly for distinguishing complex character patterns. Additionally, the results are compared with previous studies to demonstrate the method's novelty and effectiveness, highlighting improvements in both accuracy and robustness. This comprehensive evaluation not only validates the proposed methodology but also demonstrates its potential as a significant contribution to the field of pattern recognition and character classification.

3.1. Overview of Method Performance

The primary goal of this study was to explore the effectiveness of cosine similarity in classifying Javanese script images. Results demonstrated that cosine similarity, when combined with Histogram of Oriented Gradients (HOG) feature extraction and classification models such as K-Nearest Neighbors (KNN) (Table 1) and Support Vector Machine (SVM) (Table 2), provided superior accuracy. The highest performance was achieved with cosine similarity and a Support Vector Machine (SVM), achieving an accuracy of 99.84%. This indicates that cosine similarity is particularly well-suited to high-dimensional feature spaces, as it captures directional similarity between vectors, which is critical for distinguishing subtle differences in character shapes and textures.

Table 1. KNN Classification Report

Javanese Script	precision	recall	f1-score
ba	0.88	0.90	0.89
ca	0.88	0.97	0.92
da	0.67	0.24	0.85
dha	0.92	0.92	0.92
ga	0.88	0.91	0.90
ja	0.97	0.97	0.97
ka	0.87	0.92	0.89
la	0.89	0.92	0.90
ma	0.95	0.91	0.93
na	0.87	0.93	0.90
nga	0.89	0.92	0.91
nya	0.95	0.87	0.91
pa	0.84	0.93	0.88
ra	0.92	0.95	0.93
sa	0.87	0.92	0.89
ta	0.91	0.92	0.89
tha	0.92	0.99	0.95
wa	0.91	0.86	0.88
ya	0.90	0.88	0.89
accuracy			0.90
macro avg	0.89	0.89	0.88
weighted avg	0.89	0.90	0.89

Table 2. SVM Classification Report

Javanese Script	precision	recall	f1-score
ba	1.000	1.000	1.000
ca	1.000	1.000	1.000
da	1.000	1.000	1.000
dha	1.000	1.000	1.000
ga	0.992	0.992	0.992
ja	1.000	1.000	1.000
ka	1.000	1.000	1.000
la	1.000	1.000	1.000
ma	1.000	1.000	1.000
na	1.000	1.000	1.000
nga	1.000	1.000	1.000
nya	1.000	0.987	0.994
pa	1.000	1.000	1.000
ra	0.991	0.991	0.991
sa	1.000	1.000	1.000
ta	1.000	1.000	1.000
tha	1.000	1.000	1.000
wa	1.000	1.000	1.000
ya	0.988	1.000	0.994
accuracy			0.998
macro avg	0.998	0.998	0.998
weighted avg	0.998	0.998	0.998

3.2. Key Findings from Classification Models

3.2.1. Performance of Cosine Similarity with KNN and SVM

Cosine similarity was evaluated alongside traditional distance metrics. such as Euclidean and Manhattan distances, to determine their effectiveness when used with K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) classifiers. The comparison aimed to highlight the advantages of cosine similarity in handling feature vectors for classification tasks. The results. summarized in Table 3, demonstrate the performance differences across these metrics. showcasing how cosine similarity consistently outperformed the others in terms of classification accuracy. This finding underscores the suitability of cosine similarity for tasks involving angular relationships. particularly in scenarios with varying magnitudes due to lighting or contrast. The analysis further reinforces the importance of selecting an appropriate distance metric to optimize classifier performance for specific applications.

Table 3. Summary of the Results

Metric	Classifier	Precision	Recall	F1-Score	Accuracy
Cosine	KNN	0.89	0.89	0.9	0.9
Cosine	SVM	0.9984	0.9984	0.9984	0.9984
Euclidean	KNN	0.89	0.88	0.89	0.88
Manhattan	KNN	0.89	0.88	0.89	0.88

3.2.2. Parameter Optimization

Using GridSearchCV, the optimal parameters for both K-Nearest Neighbors (KNN) and Support Vector Machine (SVM) models were successfully identified to enhance classification accuracy. For the K-Nearest Neighbors (KNN) model, the best performance was achieved with k=3 while utilizing cosine similarity as the distance metric, as detailed in Table 4. This result highlights the effectiveness of cosine similarity in capturing the relationships between data points in the given feature space. Similarly. The Support Vector Machine (SVM) model achieved superior accuracy with the cosine kernel. surpassing the performance of both the radial basis function (RBF) and linear kernels. Table 5 provides a comprehensive comparison of the kernel methods. showcasing the robustness of the cosine kernel for the dataset used in this study. These findings emphasize the importance of selecting appropriate parameters and similarity measures in optimizing machine learning models for complex datasets.

Table 4. KNN Optimization Parameters

Metric	K	Score
Cosine	3	0.611
Cosine	5	0.332
Cosine	7	0.303
Cosine	9	0.311

Table 5. SVM Optimization Parameters

Kernel	C	Gamma	Score
Cosine	10	auto	0.937
RBF	10	auto	0.890
Linear	10	auto	0.891

3.2.3. Cross-Validation Results

The models were rigorously evaluated using 10-fold cross-validation to ensure their reliability and consistency. This method provided a robust framework for assessing the generalizability of the models across different subsets of the data. Notably. Cosine similarity proved highly effective as a metric. maintaining consistent performance across all validation folds. Tables 6 and 7 illustrate this stability, demonstrating its ability to handle the inherent variations in the Javanese character dataset. These results underscore the robustness of cosine similarity. highlighting its suitability for tasks that require precision and adaptability in diverse feature spaces.

Table 6. KNN 10-fold Cross Validation Confusion Matrix

Actual / Prediction	ba	ca	da	dha	ga	Ja	ka	la	ma	na	nga	nya	pa	ra	sa	ta	tha	wa	ya
ba	**213**	0	0	0	2	0	0	4	0	4	0	2	0	0	0	6	2	2	2
ca	2	**229**	0	0	0	0	0	0	0	0	0	0	2	2	0	0	2	0	0
da	6	10	**40**	6	2	2	6	6	6	14	12	0	18	4	12	4	12	8	2
dha	2	4	0	**213**	0	0	6	0	0	0	0	0	0	0	2	0	2	0	2
ga	2	0	0	2	**214**	2	0	2	0	2	2	0	2	4	2	0	0	0	0
ja	2	0	0	0	2	**228**	0	0	0	2	0	0	0	0	0	0	0	0	0
ka	2	2	0	0	0	0	**219**	4	0	0	0	0	0	0	0	0	0	0	2
la	2	0	0	0	2	0	2	**216**	0	0	0	2	2	2	2	0	0	0	4
ma	0	2	2	0	0	0	0	0	**215**	4	0	2	0	0	2	0	0	4	0
na	0	2	2	2	2	0	2	2	2	**221**	0	2	0	0	0	0	0	2	0
nga	0	2	2	0	0	0	2	4	2	0	**219**	0	2	2	0	2	0	0	0
nya	0	2	2	2	4	0	4	4	0	0	2	**203**	0	0	0	2	0	0	2
pa	0	2	0	2	0	0	0	2	0	0	2	0	**215**	2	0	0	0	0	0
ra	2	0	0	0	2	0	0	0	0	0	0	4	2	**213**	0	0	2	0	2
sa	0	0	2	2	2	2	2	0	2	2	0	0	2	0	**217**	0	0	2	2
ta	2	0	2	0	2	0	4	0	0	0	2	0	4	0	0	**210**	0	2	0
tha	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	**229**	0	0
wa	4	2	4	0	6	0	2	0	0	0	6	0	0	2	4	2	2	**203**	0
ya	2	4	4	2	4	0	4	0	2	0	0	2	2	0	2	0	0	0	**209**

Table 7. SVM 10-fold Cross Validation Confusion Matrix

Actual / Prediction	ba	ca	da	dha	ga	Ja	ka	la	ma	na	nga	nya	pa	ra	sa	ta	tha	wa	ya
ba	**237**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ca	0	**237**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
da	0	0	**170**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
dha	0	0	0	**231**	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0
ga	0	0	0	0	**234**	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ja	0	0	0	0	0	**234**	0	0	0	0	0	0	0	0	0	0	0	0	0
ka	0	0	0	0	0	0	**237**	0	0	0	0	0	0	0	0	0	0	0	0
la	0	0	0	0	0	0	0	**234**	0	0	0	0	0	0	0	0	0	0	0
ma	0	0	0	0	0	0	0	0	**237**	0	0	0	0	0	0	0	0	0	0
na	0	0	0	0	0	0	0	0	0	**237**	0	0	0	0	0	0	0	0	0
nga	0	0	0	0	0	0	0	0	0	0	**231**	0	0	0	0	0	0	0	0
nya	0	0	0	0	0	0	0	0	0	0	0	**231**	0	0	0	0	0	0	0
pa	0	0	0	0	0	0	0	0	0	0	0	0	**225**	0	0	0	0	0	0
ra	0	0	0	0	0	0	0	0	0	0	0	0	0	**237**	0	0	0	0	0
sa	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**237**	0	0	0	0
ta	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**228**	0	0	0
tha	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**231**	0	0
wa	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**237**	0
ya	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**237**

3.2.4. Advantages of Cosine Similarity

The application of cosine similarity offered several distinct advantages, contributing to its effectiveness in the classification process. Unlike Euclidean and Manhattan distances. Cosine similarity prioritizes the angle between vectors rather than their magnitude. allowing it to remain invariant to scale differences. This direction-based approach proves particularly advantageous for datasets where magnitude variations may not correlate with meaningful distinctions. Moreover. Cosine similarity is well-suited for handling high-dimensional feature vectors. such as those derived from Histogram of Oriented Gradients (HOG). enabling it to capture intricate details in character patterns with greater precision. Another notable benefit is its resilience to noise and scale variations, as it emphasizes the directional relationship between data points rather than absolute distances. These attributes collectively make cosine similarity a robust choice for complex pattern recognition tasks. especially when working with detailed and diverse datasets.

3.3. Comparison with Previous Studies

The performance of the proposed method was systematically benchmarked against prior research on Javanese character recognition to assess its effectiveness and relevance. This comparative analysis highlights the advancements and improvements achieved through the new approach. Table 8 presents a comprehensive overview of the comparison. including methods. accuracy. dataset sizes. and preprocessing techniques. ensuring a transparent and valid evaluation. As shown in Table 4, the proposed method achieves significant improvements. particularly with an accuracy of 99.84% when using cosine similarity with the Support Vector Machine (SVM) classifier. This outcome demonstrates the robustness of the proposed method in addressing challenges such as intricate patterns and variations in the Javanese character set. It is essential to note that previous studies employed varying dataset sizes and preprocessing techniques. The proposed method was evaluated using a dataset of 4,385 images, with Histogram of Oriented Gradients (HOG) used for feature extraction. This ensures a fair comparison, as consistent preprocessing and texture-based feature extraction provide reliable insights into the method's effectiveness.

For instance, Putri et al. [4], who achieved 98.35% accuracy, utilized a significantly larger dataset with dropout techniques to prevent overfitting. While Anggraeny et al. [5] applied noise reduction and skeletonization on 2,634 samples, achieving 98% similarly. Rasyidi et al. [2] used Random Forest with Histogram of Oriented Gradients (HOG) on a dataset of 6,000 images, achieving 97.7% accuracy. In contrast. The proposed approach integrates cosine similarity as a distance metric. optimizing texture feature discrimination. and demonstrating superior accuracy without relying on extensive computational resources or complex architectures. These findings emphasize the novelty and practicality of the proposed method. especially when compared to deep learning-based approaches like those by Susanto et al. [3], who achieved 99.65% accuracy using deep convolutional neural networks (DCNN) with data augmentation. The proposed method relies on cosine similarity. a computationally efficient metric. offers a simpler yet highly effective alternative for Javanese character recognition tasks.

Table 8. Highlights the Comparison

Research	Method	Accuracy	Dataset Size	Preprocessing
Katili dkk. [8]	*Local Binary Pattern (LBP) + Information Gain + Support Vector Machine (SVM)*	87.86%	160	LBP.Information Gain
Putri dkk. [4]	*Convolutional Neural Networks (CNN) + Support Vector Machine (SVM)*	98.35%	15600	Dropout Technique
Anggraeny dkk. [5]	Dilatation + skeletonization + noise reduction + ANN	98%	2634	Dilatation + skeletonization + noise reduction +
Rasyidi dkk. [2]	*Random Forest + HOG*	97.7%	6000	HOG
Susanto dkk. [1]	*K-Nearest Neighbor + Linear Binary Pattern (LBP)*	90.5%	2470	LBP
Susanto dkk [3]	*Deep Convolutional Neural Network (DCNN)*	99.65%	3360	Data Augmentation
This study	*Cosine Similarity + KNN*	90%	4385	HOG
This study	***Cosine Similarity + SVM***	**99.84%**	**4385**	**HOG**

3.4. Significance of Findings

The high accuracy achieved by the proposed method underscores its potential for a wide range of practical applications. particularly in areas where precision and efficiency are critical. One notable use is in digital preservation. where automating the recognition of Javanese script can play a crucial role in preserving and digitizing cultural heritage for future generations. Additionally, this method can be integrated into educational tools. providing interactive learning applications to teach and assess Javanese script writing effectively. Its computational efficiency also makes it an ideal solution for deployment in low-resource environments. where access to advanced hardware may be limited. These applications highlight the versatility of the proposed method. demonstrating its value beyond academic research and into real-world implementations that benefit diverse communities.

While the results of this study are highly promising, there are some limitations that warrant further investigation to enhance the method's applicability and robustness. One notable constraint is the dataset size. which was limited to 4.385 images; expanding the dataset with more diverse samples could provide stronger validation of the method's effectiveness. Additionally. The current approach relies exclusively on Histogram of Oriented Gradients (HOG) features, and future research could explore integrating other feature extraction techniques. such as Local Binary Patterns (LBP) or deep learning-based features. to improve generalization and performance. Real-world testing on handwritten or degraded scripts is another area for future work. as it would offer valuable insights into the method's applicability in practical scenarios. Despite these limitations. this study emphasizes the strengths of cosine similarity. contributing a novel and efficient perspective to the field of Javanese character recognition. The proposed method demonstrates remarkable accuracy and robustness. making it a compelling solution for high-dimensional classification tasks.

4. CONCLUSION

This study successfully addressed the gap in leveraging cosine similarity for Javanese script classification. By integrating cosine similarity with Histogram of Oriented Gradients (HOG) based feature extraction and classification models such as K-Nearest Neighbors (KNN) and Support Vector Machine (SVM). The proposed method demonstrated superior accuracy. achieving 99.84% with Support Vector Machine (SVM). These findings underscore cosine similarity's effectiveness in handling high-dimensional and texture-based data. offering a computationally efficient alternative to more complex neural network models.

The implications extend beyond character recognition. suggesting potential applications in digital preservation. education. and low-resource environments. The simplicity and robustness of the proposed framework make it particularly valuable for tasks requiring efficient and accurate pattern recognition. For future work. expanding the dataset. incorporating additional feature extraction techniques. and testing the method on more challenging datasets. such as handwritten or degraded scripts. are recommended. These directions could enhance the method's applicability and further validate its robustness in real-world scenarios. This research makes a novel contribution to computational linguistic and cultural heritage preservation. advancing the state of the art in Javanese script recognition.

5. ACKNOWLEDGEMENTS

We extend our heartfelt gratitude to the Directorate General of Higher Education. Research. and Technology for their invaluable support in funding this research. Specifically. we acknowledge the Directorate of Research. Technology. and Community Service (DRTPM) of Kemendikbudristek. whose commitment to advancing academic and community-driven initiatives made this study possible. The funding was provided through the Proposal for Research and Community Service program. demonstrating their dedication to fostering innovation and scholarly excellence. This support for the 2024 funding year has been instrumental in enabling us to pursue meaningful contributions in our field. We deeply appreciate their trust and investment in research that bridges academic inquiry with real-world applications.

6. DECLARATIONS

AUTHOR CONTIBUTION

The success of this research was made possible through the collaborative efforts of the team members, each contributing significantly to its various stages. Aji Priyambodo took the lead in preparing the research proposal. coordinating the implementation of the study. developing the classification model, and writing both the final report and the publication. Prihati, Danang, and Farhan bin Mohamed provided valuable assistance throughout the process. contributing to the preparation of the research proposal. supporting the development of the classification model, and aiding in the writing of the report and publication. This teamwork ensured that the research was conducted effectively. With each member bringing their expertise to achieve the project's goals. The combined efforts reflect a shared commitment to producing high-quality research outcomes that advance the field.

FUNDING STATEMENT

This research was supported by the Directorate of Research. Technology. and Community Service (DRTPM) of Kemendikbudristek.

COMPETING INTEREST

The authors declare that they have no conflict of interest.

REFERENCES

[1] A. Susanto, I. U. W. Mulyono, C. A. Sari, E. H. Rachmawanto, and R. R. Ali, "Javanese Character Recognition Based on K-Nearest Neighbor and Linear Binary Pattern Features," *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, vol. 7, no. 3, August 2022, https://doi.org/10.22219/kinetik.v7i3.1491.

[2] M. A. Rasyidi, T. Bariyah, Y. I. Riskajaya, and A. D. Septyani, "Classification of handwritten Javanese script using random forest algorithm," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 3, pp. 1308–1315, Jun. 2021, https://doi.org/10.11591/eei.v10i3.3036.

[3] A. Susanto, I. U. Wahyu Mulyono, C. Atika Sari, E. Hari Rachmawanto, D. R. Ignatius Moses Setiadi, and M. K. Sarker, "Handwritten Javanese script recognition method based 12-layers deep convolutional neural network and data augmentation," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 12, no. 3, p. 1448, Sep. 2023, https://doi.org/10.11591/ijai.v12.i3.pp1448-1458.

[4] D. U. K. Putri, D. N. Pratomo, and A. 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, p. 346, Apr. 2023, https://doi.org/10.12928/telkomnika.v21i2.24266.

[5] F. T. Anggraeny, Y. V. Via, and R. Mumpuni, "Image preprocessing analysis in handwritten Javanese character recognition," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 2, pp. 860–867, Apr. 2023, https://doi.org/10.11591/eei.v12i2.4172.

[6] A. Susanto, C. A. Sari, E. H. Rachmawanto, I. U. W. Mulyono, and N. Mohd Yaacob, "A Comparative Study of Javanese Script Classification with GoogleNet, DenseNet, ResNet, VGG16 and VGG19," *Scientific Journal of Informatics*, vol. 11, no. 1, pp. 31–40, Jan. 2024, https://doi.org/10.15294/sji.v11i1.47305.

[7] M. F. Naufal, J. Siswantoro, and J. T. Soebroto, "Transliterating Javanese Script Images to Roman Script using Convolutional Neural Network with Transfer Learning," *JOIV : International Journal on Informatics Visualization*, vol. 8, no. 3, p. 1460, Sep. 2024, https://doi.org/10.62527/joiv.8.3.2566.

[8] I. F. Katili, M. A. Soeleman, and R. A. Pramunendar, "Character Recognition of Handwriting of Javanese Character Image using Information Gain Based on the Comparison of Classification Method," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 1, pp. 193–200, Feb. 2023, https://doi.org/10.29207/resti.v7i1.4488.

[9] N. Xie, M. Xiong, F. Wei, T. Zhang, Z. Yang, and W. Yu, "CA-LOSS: A Cosine Affinity Loss for Imbalanced SAR Ship Classification," in *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*. Athens, Greece: IEEE, Jul. 2024, pp. 9070–9074, https://doi.org/10.1109/IGARSS53475.2024.10640754.

[10] X. Qiao, S. K. Roy, and W. Huang, "3-D Sharpened Cosine Similarity Operation for Hyperspectral Image Classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 1114–1125, November, 2024, https://doi.org/10.1109/JSTARS.2023.3337112.

[11] Z. Wang, J. Chen, and J. Hu, "Multi-View Cosine Similarity Learning with Application to Face Verification," *Mathematics*, vol. 10, no. 11, p. 1800, May 2022, https://doi.org/10.3390/math10111800.

[12] S. Palanisamy and J. Periyasamy, "Interval-Valued Intuitionistic Fuzzy Cosine Similarity Measures for Real World Problem Solving," *International Journal of Robotics and Control Systems*, vol. 4, no. 2, pp. 655–677, May 2024, https://doi.org/10.31763/ijrcs.v4i2.1251.

[13] K. Endo, M. Tanaka, and M. Okutomi, "CNN-Based Classification of Degraded Images Without Sacrificing Clean Images," *IEEE Access*, vol. 9, pp. 116 094–116 104, August, 2021, https://doi.org/10.1109/ACCESS.2021.3105957.

[14] R. Wang, T. Chen, P. Yao, S. Liu, I. Rajapakse, and A. O. Hero, "ASK: Adversarial Soft k-Nearest Neighbor Attack and Defense," *IEEE Access*, vol. 10, pp. 103 074–103 088, September, 2022, https://doi.org/10.1109/ACCESS.2022.3209243.

[15] Q.-H. Nguyen, C. Q. Nguyen, D. D. Le, and H. H. Pham, "Enhancing Few-Shot Image Classification With Cosine Transformer," *IEEE Access*, vol. 11, pp. 79 659–79 672, July, 2023, https://doi.org/10.1109/ACCESS.2023.3298299.

[16] M. Shulgin and I. Makarov, "Scalable Zero-Shot Logo Recognition," *IEEE Access*, vol. 11, pp. 142 702–142 710, December, 2023, https://doi.org/10.1109/ACCESS.2023.3342721.

[17] A. Naseer, H. A. Alzahrani, N. A. Almujally, K. A. Nowaiser, N. A. Mudawi, A. Algarni, and J. Park, "Efficient Multi-Object Recognition Using GMM Segmentation Feature Fusion Approach," *IEEE Access*, vol. 12, pp. 37 165–37 178, March, 2024, https://doi.org/10.1109/ACCESS.2024.3372190.

[18] M. Mohammadi, M. Eftekhari, and A. Hassani, "Image-Text Connection: Exploring the Expansion of the Diversity Within Joint Feature Space Similarity Scores," *IEEE Access*, vol. 11, pp. 123 209–123 222, October, 2023, https://doi.org/10.1109/ACCESS.2023.3327339.

[19] X. Li, H. Belcher, and H. Wang, "Robust Spherical Laplacian Embedding," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 4, pp. 7594–7605, Apr. 2025, https://doi.org/10.1109/TNNLS.2024.3401010.

[20] H. Wang, "Clustering statistical method of high dimensional sparse data based on fuzzy data," *Journal of Physics: Conference Series*, vol. 2791, no. 1, p. 012060, Jul. 2024, https://doi.org/10.1088/1742-6596/2791/1/012060.

[21] I. T. Ahmed, B. T. Hammad, and N. Jamil, "A Comparative Performance Analysis of Malware Detection Algorithms Based on Various Texture Features and Classifiers," *IEEE Access*, vol. 12, pp. 11 500–11 519, January, 2024, https://doi.org/10.1109/ACCESS.2024.3354959.

[22] R. Rofik, R. A. Hakim, J. Unjung, B. Prasetiyo, and M. A. Muslim, "Optimization of SVM and Gradient Boosting Models Using GridSearchCV in Detecting Fake Job Postings," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 23, no. 2, pp. 419–430, Mar. 2024, https://doi.org/10.30812/matrik.v23i2.3566.