

---

## ANALISIS PENGARUH CLAHE PADA KINERJA MOBILENETV3-SVM KLASIFIKASI AKSARA JAWA

Dela Ayu Putri Mayona<sup>1</sup>, Chrystia Aji Putra<sup>2\*</sup>, Hendra Maulana<sup>3</sup>

<sup>1,2</sup>Program Studi Informatika, Universitas Pembangunan Nasional "Veteran" Jawa Timur, <sup>3</sup>Program Studi  
Bisnis Digital, Universitas Pembangunan Nasional "Veteran" Jawa Timur

Jl.Raya Rungkut Madya, Gunung Anyar, Surabaya, Indonesia

<sup>1</sup> [22081010008@student.upnjatim.ac.id](mailto:22081010008@student.upnjatim.ac.id), <sup>2\*</sup> [ajiputra@upnjatim.ac.id](mailto:ajiputra@upnjatim.ac.id), <sup>3</sup> [hendra.maulana.if@upnjatim.ac.id](mailto:hendra.maulana.if@upnjatim.ac.id)

---

### Abstract

This study discusses the recognition of Javanese script as an effort to preserve cultural heritage in the digital technology era. The main challenge lies in the low image quality and the high visual similarity between characters, which can reduce classification accuracy. This research aims to analyze the performance of the MobileNetV3-SVM model and the effect of applying Contrast Limited Adaptive Histogram Equalization (CLAHE) preprocessing in improving recognition accuracy. The proposed method consists of preprocessing stages (CLAHE, resizing, and normalization), feature extraction using MobileNetV3Small, and classification using a Support Vector Machine (SVM) with a linear kernel. The dataset used comprises 2,500 images of Javanese script, divided into training and testing sets with an 80:20 ratio. The results show that the model with CLAHE achieves the highest accuracy of 98.4%, outperforming the model without CLAHE, which achieves 97.2%. The novelty of this study lies in the integration of CLAHE with the MobileNetV3-SVM architecture to enhance feature discrimination in visually similar Javanese characters. The findings imply that the proposed approach can serve as an effective solution for developing more accurate and adaptive AI-based systems for regional script recognition.

**Keywords :** *Javanese handwritten recognition, MobileNetV3Small, SVM linear kernel, CLAHE, image classification*

### Abstrak

Penelitian ini membahas pengenalan aksara Jawa sebagai upaya pelestarian budaya di era teknologi digital. Permasalahan utama terletak pada rendahnya kualitas citra dan kemiripan visual antar karakter yang dapat menurunkan akurasi klasifikasi. Penelitian ini bertujuan untuk menganalisis kinerja model MobileNetV3-SVM serta pengaruh penerapan preprocessing Contrast Limited Adaptive Histogram Equalization (CLAHE) dalam meningkatkan akurasi pengenalan. Metode yang digunakan meliputi tahap preprocessing (CLAHE, resize, dan normalisasi), ekstraksi fitur menggunakan MobileNetV3Small, serta klasifikasi menggunakan Support Vector Machine (SVM) dengan kernel linear. Dataset yang digunakan terdiri dari 2.500 citra aksara Jawa yang dibagi dengan rasio 80:20 untuk pelatihan dan pengujian. Hasil penelitian menunjukkan bahwa model dengan CLAHE mencapai akurasi tertinggi sebesar 98,4%, lebih baik dibandingkan tanpa CLAHE yang memperoleh 97,2%. Kebaruan penelitian ini terletak pada integrasi metode CLAHE dengan arsitektur MobileNetV3-SVM untuk meningkatkan diskriminasi fitur pada citra aksara Jawa yang memiliki kemiripan visual tinggi. Implikasi dari penelitian ini menunjukkan bahwa pendekatan yang diusulkan dapat menjadi solusi efektif dalam pengembangan sistem pengenalan aksara daerah berbasis kecerdasan buatan yang lebih akurat dan adaptif.

**Kata kunci :** *pengenalan tulisan aksara Jawa, MobileNetV3Small, SVM linear kernel, CLAHE, klasifikasi citra*

---



*Machine learning* merupakan bagian dari kecerdasan buatan yang memungkinkan sistem untuk mempelajari pola dari data secara otomatis tanpa perlu diprogram secara langsung. Proses pembelajaran ini melibatkan kemampuan sistem dalam mengenali pola, membentuk representasi, serta meningkatkan kinerja seiring bertambahnya data dan pengalaman. Penerapannya telah banyak digunakan dalam berbagai bidang, seperti deteksi penipuan, sistem rekomendasi, dan analisis data medis. Berdasarkan pendekatannya, *machine learning* diklasifikasikan menjadi *supervised learning*, *unsupervised learning*, dan *reinforcement learning* [8].

*Image enhancement* merupakan tahap penting dalam pengolahan citra digital yang bertujuan meningkatkan kualitas visual dengan menonjolkan fitur tertentu tanpa menambah informasi baru. Salah satu metode yang efektif adalah CLAHE, yaitu teknik *preprocessing* yang meningkatkan kontras citra secara lokal dengan membatasi amplifikasi kontras agar tidak memperbesar *noise*. Prosesnya dilakukan dengan membagi citra menjadi beberapa blok, kemudian histogram tiap blok dipotong sesuai ambang batas dan didistribusikan kembali sebelum dilakukan pemetaan nilai piksel menggunakan *Cumulative Distribution Function* (CDF) [9].

MobileNet merupakan arsitektur CNN ringan yang memanfaatkan *depthwise separable convolution* untuk mengurangi kompleksitas komputasi sehingga efisien pada perangkat dengan sumber daya terbatas [10]. Pengembangannya, yaitu MobileNetV2, menambahkan *inverted residual* dan *linear bottleneck* untuk meningkatkan efisiensi dan akurasi, sedangkan MobileNetV3 mengintegrasikan NAS, *Hard-Swish*, dan modul SE untuk menghasilkan performa yang lebih optimal. Dengan keunggulan tersebut, MobileNetV3 dipilih karena mampu memberikan keseimbangan terbaik antara akurasi dan efisiensi komputasi, serta cocok untuk implementasi pada sistem berbasis *web* dan *mobile* [11].

*SVM* merupakan algoritma *supervised learning* yang banyak digunakan untuk tugas klasifikasi dan regresi. Metode ini bekerja dengan menentukan *hyperplane* terbaik yang mampu memisahkan data dari kelas berbeda dengan margin maksimum. Keunggulan SVM terletak pada kemampuannya menangani data berdimensi tinggi melalui penggunaan *kernel trick*, seperti linear, RBF, dan sigmoid, yang memungkinkan transformasi data non-linear ke ruang berdimensi lebih tinggi agar lebih mudah dipisahkan [12].

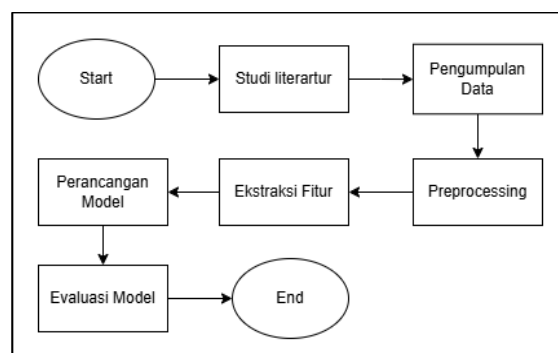
*Confusion matrix* merupakan salah satu metode evaluasi yang sering digunakan untuk

menilai kinerja model, khususnya dalam menghitung tingkat akurasi pada *data mining* atau sistem pendukung keputusan. Matriks ini terdiri dari empat komponen utama, yaitu *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN) [13].

## 2. METODOLOGI PENELITIAN

### 2.1. Skema Alur Penelitian

Pada bagian ini memuat metode yang digunakan pada penelitian ini. Penelitian ini terdiri dari beberapa tahapan utama, meliputi studi literatur, pengumpulan data, *preprocessing*, ekstraksi fitur, perancangan model, evaluasi model.



Gambar 2. Alur Penelitian

Tahapan penelitian yang ditunjukkan pada Gambar 2 dilakukan secara sistematis agar proses berjalan terstruktur dan menghasilkan *output* sesuai tujuan. Kegiatan diawali dengan studi literatur untuk memperoleh referensi dan pemahaman metode, dilanjutkan dengan pengumpulan dataset yang relevan. Tahap *preprocessing* dilakukan untuk menyiapkan data, kemudian ekstraksi fitur digunakan untuk mengambil karakteristik penting sebagai *input* model. Selanjutnya dilakukan perancangan model dengan menentukan algoritma dan parameter yang tepat, lalu dievaluasi menggunakan metrik tertentu untuk mengukur kinerjanya. Dengan alur tersebut, diharapkan diperoleh hasil yang akurat dan aplikatif sesuai tujuan penelitian.

### 2.2. Studi Literatur





Tahap studi literatur merupakan bagian penting dalam penelitian karena digunakan untuk mengkaji teori dasar serta pengetahuan yang telah ada terkait pengenalan aksara. Proses ini dilakukan dengan menelaah berbagai sumber

ilmiah, seperti jurnal, artikel akademik, dan penelitian terdahulu yang relevan dengan aksara Jawa, *machine learning*, *MobileNetV3*, dan SVM. Melalui studi literatur, peneliti dapat memahami konsep pengenalan aksara secara menyeluruh serta memperoleh gambaran mengenai penerapan SVM yang memanfaatkan fitur hasil ekstraksi dari *MobileNetV3* dalam penelitian. Tahapan ini menjadi sangat penting karena membantu peneliti memahami bagaimana suatu permasalahan telah dikaji sebelumnya oleh penelitian lain, sehingga dapat digunakan untuk mengidentifikasi kesenjangan (*research gap*) yang menjadi dasar pengembangan penelitian selanjutnya [14].

### 2.3. Pengumpulan Data

Pada tahap pengumpulan data, penelitian ini memanfaatkan data sekunder yang diperoleh dari *public dataset* di Kaggle yang dipublikasikan oleh Phiard dengan judul *Aksara Jawa: Aksara Jawa Custom Dataset* [15]. Dataset tersebut terdiri dari 125 sampel untuk setiap aksara dengan total keseluruhan sebanyak 2.500 citra.

TABEL I. SAMPEL CITRA DATASET AKSARA JAWA

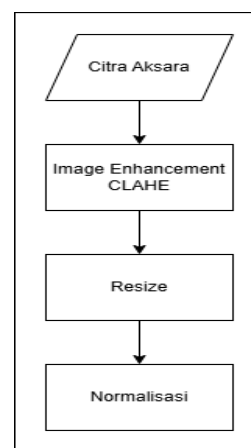
No.	Aksara Jawa	Citra Sampel
1.	HA	
2.	NA	
3.	CA	
4.	RA	

Tabel I menampilkan contoh data sekunder yang diambil dari *public dataset* Kaggle dengan jenis aksara yang sama, yaitu HA, NA, CA, dan RA. Gambar tersebut memperlihatkan karakteristik visual data yang beragam, mulai dari variasi bentuk tulisan, ketebalan garis, hingga tingkat kontras. Keberagaman ini penting untuk melatih model agar lebih robust dalam mengenali pola

aksara Jawa dan tidak mudah terpengaruh oleh perbedaan visual pada data.

### 2.4. Preprocessing

Tahap *preprocessing* merupakan proses awal yang bertujuan menyiapkan citra agar siap digunakan dalam ekstraksi fitur dan pelatihan model. Pada tahap ini, data mentah diolah untuk meningkatkan kualitas dan keseragaman, karena kondisi data sangat berpengaruh terhadap kemampuan model dalam mengekstraksi fitur dan menghasilkan klasifikasi yang akurat.



Gambar 3. Alur *Preprocessing*

Gambar 3 menggambarkan alur *preprocessing* pada citra aksara Jawa yang diawali dengan penyeimbangan jumlah data pada setiap kelas untuk menghindari bias model. Setelah itu, citra dimuat ke dalam sistem dan dilakukan *image enhancement* guna meningkatkan kualitas visual, khususnya dalam memperbaiki kontras antara objek aksara dan latar belakang. Proses ini penting karena dataset yang digunakan berupa citra *grayscale* dengan rentang intensitas 0–255, sehingga pengaturan distribusi piksel menjadi kunci dalam memperjelas struktur karakter.

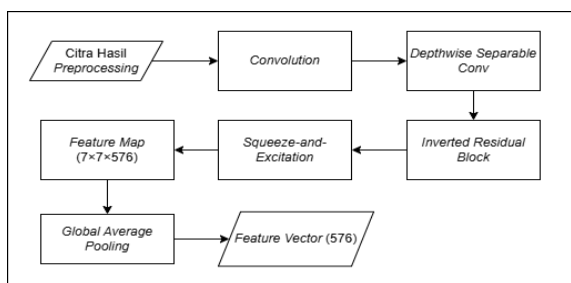
Sebagai tahap lanjutan, diterapkan metode CLAHE untuk meningkatkan kontras citra secara lokal. Teknik ini bekerja dengan membagi citra menjadi beberapa blok kecil, kemudian membatasi nilai frekuensi histogram menggunakan *clip limit* agar tidak memperbesar *noise*. Nilai frekuensi yang melebihi batas akan dipotong dan didistribusikan kembali, sehingga kualitas fitur tetap terjaga. Namun, pada kondisi tertentu di mana nilai histogram tidak melebihi batas, proses *clipping* tidak memberikan perubahan signifikan terhadap distribusi citra.

Tahap berikutnya adalah *resize* dan normalisasi citra. Proses *resize* dilakukan untuk menyeragamkan ukuran citra sesuai kebutuhan model, yaitu 224×224 piksel pada *MobileNetV3*,

sehingga data dapat diproses secara konsisten. Selanjutnya, normalisasi dilakukan dengan mengubah rentang nilai piksel menjadi 0-1 agar distribusi data lebih stabil. Proses ini membantu model dalam fokus mengenali pola dan bentuk aksara tanpa terpengaruh oleh variasi pencahayaan, sehingga menghasilkan ekstraksi fitur yang lebih optimal.

### 2.5. Ekstraksi Fitur

Ekstraksi fitur merupakan tahap penting dalam pengenalan citra yang mengubah citra menjadi representasi numerik (*feature vector*) agar dapat diproses oleh model. Pada penelitian ini, digunakan *MobileNetV3Small* karena efisien secara komputasi dan mampu menghasilkan fitur yang kuat, sehingga sesuai sebagai *feature extractor* sebelum klasifikasi menggunakan SVM.

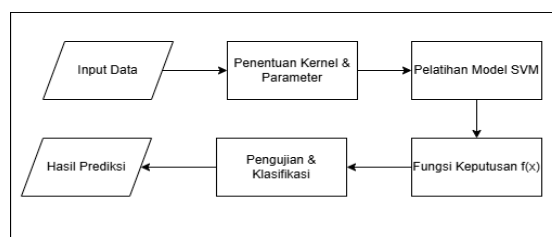


Gambar 4. Alur Ekstraksi Fitur MobileNetV3

Gambar 4 merupakan alur dari ekstraksi fitur. Pada tahap *convolution*, citra diproses untuk mengekstraksi pola lokal seperti tepi dan perubahan intensitas melalui pergeseran kernel dan perhitungan perkalian elemen. Hasilnya berupa *feature map* yang merepresentasikan respons terhadap pola tertentu. Selanjutnya, pada *depthwise convolution*, setiap *channel* diproses secara terpisah untuk meningkatkan efisiensi komputasi. Proses ini dilanjutkan dengan *inverted residual block* yang memperluas dimensi fitur melalui ekspansi, kemudian memproses dan mengembalikannya ke dimensi semula. Pada mekanisme *Squeeze-and-Excitation* (SE), dilakukan *Global Average Pooling* (GAP) untuk merangkum informasi tiap *channel* menjadi satu nilai representatif yang mencerminkan kontribusi fitur terhadap citra.

Hasil akhir ekstraksi fitur pada MobileNetV3 berupa *feature map* berukuran 7x7x576 yang menyimpan berbagai karakteristik visual seperti bentuk, tekstur, dan pola aksara. Setiap *channel* kemudian dirata-ratakan menggunakan GAP sehingga menghasilkan satu nilai per *channel*. Proses ini menghasilkan *feature vector* berdimensi 576 yang merepresentasikan citra secara ringkas dan digunakan sebagai *input* dalam proses klasifikasi menggunakan SVM.

### 2.6. Perancangan Model



Gambar 5. Perancangan Model SVM

Gambar 5 menunjukkan alur perancangan model SVM yang dilakukan setelah tahap ekstraksi fitur menggunakan *MobileNetV3*. Pada tahap ini, setiap citra aksara telah direpresentasikan dalam bentuk *feature vector*, bukan lagi berupa piksel mentah, sehingga lebih mudah diproses oleh algoritma klasifikasi. Vektor fitur tersebut kemudian digunakan sebagai input dalam proses pelatihan model SVM untuk menemukan *hyperplane* optimal yang mampu memisahkan data antar kelas dengan margin maksimum. Penelitian ini menggunakan kernel linear karena karakteristik data hasil ekstraksi MobileNetV3 cenderung sudah terpisah secara linear dalam ruang fitur berdimensi tinggi, sehingga kernel linear lebih efisien secara komputasi dibandingkan kernel non-linear seperti RBF.

Parameter utama yang digunakan dalam SVM adalah parameter regulasi *C*, yang berfungsi untuk mengontrol *trade-off* antara *margin* maksimum dan kesalahan klasifikasi. Nilai *C* yang lebih kecil akan menghasilkan *margin* yang lebih lebar namun toleran terhadap kesalahan, sedangkan nilai *C* yang lebih besar akan meminimalkan kesalahan klasifikasi tetapi berisiko *overfitting*. Dalam penelitian ini, dilakukan pengujian beberapa variasi nilai *C* untuk mendapatkan performa optimal.

### 2.7. Evaluasi Model

Tahap evaluasi performa model merupakan langkah penting untuk mengukur tingkat akurasi dan keandalan model dalam mengklasifikasikan citra. Model yang telah dilatih kemudian diuji menggunakan *confusion matrix* serta beberapa metrik evaluasi, dengan akurasi sebagai indikator utama. Selain itu, digunakan juga metrik lain seperti *precision*, *recall*, dan *F1-score* untuk memberikan gambaran performa yang lebih menyeluruh. Evaluasi dilakukan dengan membandingkan kinerja model pada berbagai pendekatan guna mengetahui metode yang paling optimal.

Selain itu, dilakukan validasi model menggunakan pembagian data *train-test split* dengan rasio 80:20. Untuk meningkatkan

keandalan hasil, pendekatan validasi ini memastikan bahwa model diuji pada data yang tidak pernah dilihat sebelumnya.

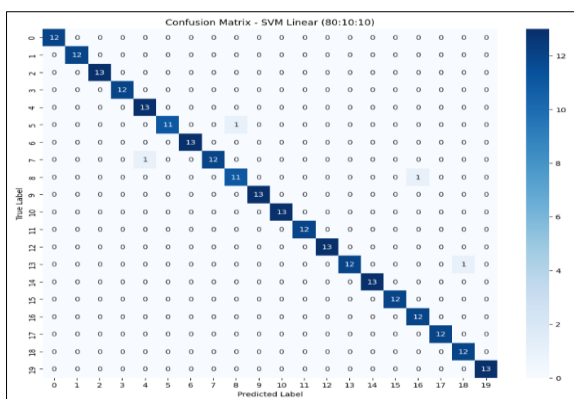
Potensi bias data juga diperhatikan, terutama terkait distribusi kelas dan variasi kualitas citra. Oleh karena itu, dilakukan penyeimbangan dataset serta *preprocessing* yang konsisten untuk meminimalkan bias dan meningkatkan kemampuan generalisasi model terhadap data baru.

### 3. HASIL DAN PEMBAHASAN

Setelah proses perancangan dan pelatihan selesai, dilakukan pengujian untuk membandingkan kinerja dua model, yaitu *MobileNetV3-SVM* tanpa *preprocessing* tambahan dan model yang menggunakan teknik CLAHE. Pengujian ini bertujuan menilai perbedaan akurasi dalam mengenali aksara Jawa, dengan hasil ditampilkan melalui prediksi pada data uji beserta visualisasi klasifikasi yang benar maupun salah. Kesalahan prediksi dianalisis untuk memahami kelemahan model dan menjadi dasar perbaikan di masa depan. Selain itu, perbandingan ini juga digunakan untuk mengetahui sejauh mana CLAHE mampu meningkatkan kualitas fitur serta kestabilan model terhadap variasi pencahayaan dan kualitas citra.

#### 3.1. Model Menggunakan Preprocessing CLAHE

Berdasarkan hasil pengujian, konfigurasi terbaik untuk klasifikasi aksara Jawa diperoleh pada pembagian data 80:20 dengan kernel Linear dan nilai C sebesar 0,1 tanpa penggunaan parameter gamma. Kombinasi ini terbukti memberikan akurasi tertinggi serta menunjukkan kinerja model yang paling stabil dan efisien dibandingkan konfigurasi lainnya.



Gambar 6. Confusion Matrix Model Menggunakan Preprocessing CLAHE

Gambar 6 menunjukkan *confusion matrix* hasil klasifikasi SVM dengan kernel linear pada 20

kelas aksara Jawa, di mana sebagian besar nilai berada pada diagonal utama yang menandakan prediksi model sudah tepat. Banyak kelas seperti ba, ca, da, dha, ga, ja, ma, na, nga, nya, ra, sa, ta, tha, wa, dan ya berhasil dikenali tanpa kesalahan, menunjukkan bahwa fitur yang diekstraksi mampu merepresentasikan karakter aksara dengan baik. Meskipun demikian, terdapat sedikit kesalahan pada beberapa kelas, seperti ha yang diprediksi sebagai ka, ka sebagai ga, la sebagai ta, dan pa sebagai wa, yang kemungkinan disebabkan oleh kemiripan pola visual antar aksara. Namun, jumlah kesalahan yang sangat kecil ini tidak signifikan, sehingga secara keseluruhan model memiliki performa klasifikasi yang sangat baik.

```

C Terbaik: 0.1

=== Classification Report (Testing Data) ===
precision recall f1-score support
0 1.00 1.00 1.00 12
1 1.00 1.00 1.00 12
2 1.00 1.00 1.00 13
3 1.00 1.00 1.00 12
4 0.93 1.00 0.96 13
5 1.00 0.92 0.96 12
6 1.00 1.00 1.00 13
7 1.00 0.92 0.96 13
8 0.92 0.92 0.92 12
9 1.00 1.00 1.00 13
10 1.00 1.00 1.00 13
11 1.00 1.00 1.00 12
12 1.00 1.00 1.00 13
13 1.00 0.92 0.96 13
14 1.00 1.00 1.00 13
15 1.00 1.00 1.00 12
16 0.92 1.00 0.96 12
17 1.00 1.00 1.00 12
18 0.92 1.00 0.96 12
19 1.00 1.00 1.00 13

accuracy 0.98 250
macro avg 0.98 0.98 0.98 250
weighted avg 0.98 0.98 0.98 250
    
```

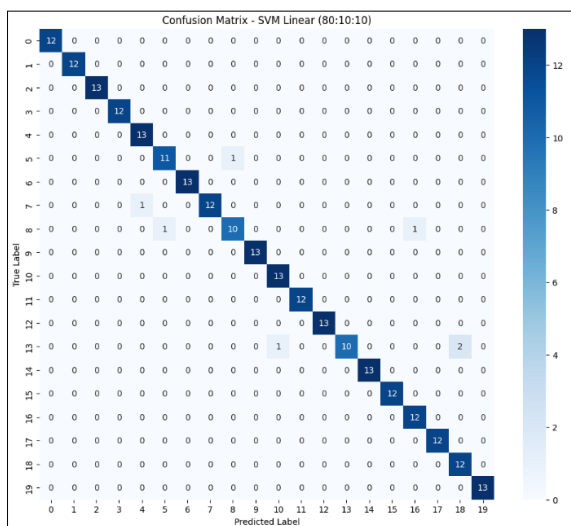
Gambar 7. Classification Report Model Menggunakan Preprocessing CLAHE

Gambar 7 menunjukkan hasil evaluasi model yang memiliki performa sangat baik dengan akurasi keseluruhan mencapai 98,4% menggunakan parameter C = 0,1. Sebagian besar kelas berhasil memperoleh nilai sempurna, khususnya pada metrik *f1-score*, yang menandakan keseimbangan antara *precision* dan *recall* serta kemampuan model dalam membedakan antar kelas dengan sangat baik. Nilai *macro average* dan *weighted average* yang sama-sama tinggi juga menunjukkan distribusi performa yang merata pada seluruh kelas. Meskipun terdapat sedikit penurunan pada beberapa kelas dengan nilai di kisaran 0,92–0,93 akibat kemiripan pola visual, hal tersebut tidak signifikan. Secara keseluruhan, model menunjukkan kinerja yang sangat optimal dengan

tingkat kesalahan yang rendah dan reliabilitas yang tinggi dalam mengenali aksara Jawa.

### 3.2. Model Tanpa Preprocessing CLAHE

Evaluasi model tanpa preprocessing CLAHE digunakan untuk mengukur efektivitas model *MobileNetV3-SVM*, dilakukan pengujian komparatif antara model standar yang hanya menggunakan *resize* dan normalisasi tanpa preprocessing CLAHE. Pengujian ini menggunakan konfigurasi yang sama dengan hasil optimal sebelumnya, yaitu kernel Linear dengan parameter C sebesar 0,1 serta pembagian data 80:20. Standarisasi ini dilakukan agar evaluasi berlangsung adil, sehingga perbedaan performa yang dihasilkan dapat menunjukkan pengaruh penggunaan CLAHE terhadap akurasi model.



Gambar 8. Confusion Matrix Model Tanpa Preprocessing CLAHE

Berdasarkan Gambar 8, confusion matrix model *MobileNetV3-SVM* tanpa preprocessing CLAHE dengan kernel Linear dan pembagian data 80:10:10 menunjukkan performa yang sangat baik, ditandai dengan dominasi nilai pada diagonal utama yang menunjukkan prediksi akurat. Sebagian besar kelas berhasil diklasifikasikan dengan sempurna tanpa kesalahan, seperti ba, ca, da, dha, ga, ja, ma, na, nya, ra, sa, ta, tha, wa, dan ya. Meskipun demikian, masih terdapat sedikit kesalahan pada beberapa kelas dengan kemiripan visual, seperti ha yang diprediksi sebagai la, ka sebagai ga, serta la dan pa yang mengalami beberapa misklasifikasi. Namun, secara keseluruhan model tetap menunjukkan tingkat keandalan yang tinggi dalam mengenali pola aksara Jawa.

C Terbaik: 0.1

=== Classification Report (Testing Data) ===

	precision	recall	f1-score	support
0	1.00	1.00	1.00	12
1	1.00	1.00	1.00	12
2	1.00	1.00	1.00	13
3	1.00	1.00	1.00	12
4	0.93	1.00	0.96	13
5	0.92	0.92	0.92	12
6	1.00	1.00	1.00	13
7	1.00	0.92	0.96	13
8	0.91	0.83	0.87	12
9	1.00	1.00	1.00	13
10	0.93	1.00	0.96	13
11	1.00	1.00	1.00	12
12	1.00	1.00	1.00	13
13	1.00	0.77	0.87	13
14	1.00	1.00	1.00	13
15	1.00	1.00	1.00	12
16	0.92	1.00	0.96	12
17	1.00	1.00	1.00	12
18	0.86	1.00	0.92	12
19	1.00	1.00	1.00	13
accuracy			0.97	250
macro avg	0.97	0.97	0.97	250
weighted avg	0.97	0.97	0.97	250

Gambar 9. Classification Report Model Tanpa Preprocessing CLAHE

Berdasarkan Gambar 9, laporan klasifikasi menunjukkan bahwa model memiliki performa yang sangat baik dengan akurasi keseluruhan sebesar 97,2% pada 250 data uji, menggunakan parameter  $C = 0,1$  yang memberikan keseimbangan optimal antara margin dan toleransi kesalahan. Sebagian besar kelas berhasil mencapai nilai sempurna pada *precision*, *recall*, dan *F1-score*, menandakan kemampuan model yang tinggi dalam membedakan karakter aksara. Meskipun demikian, terdapat sedikit penurunan performa pada beberapa kelas, seperti indeks 13 dengan *recall* terendah dan indeks 8 dengan *F1-score* yang lebih rendah akibat kemiripan pola visual, serta indikasi *over-prediction* pada kelas tertentu. Namun secara keseluruhan, model tetap menunjukkan kinerja yang solid dan andal, meskipun masih memerlukan sedikit perbaikan pada beberapa kelas tertentu.

### 3.3. Analisis Performa Kedua Skema Model

Bagian ini menjelaskan perbandingan kinerja antara dua skenario model, yaitu *MobileNetV3-SVM* dengan penerapan preprocessing CLAHE dan model yang sama tanpa menggunakan CLAHE.

TABEL II. HASIL EVALUASI PERFORMA KEDUA SKEMA MODEL

Metode	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Dengan CLAHE	98,4%	98,49%	98,4%	98,39%
Tanpa CLAHE	97,2%	97,36%	97,2%	97,14%

Berdasarkan hasil pengujian pada Tabel II, model *MobileNetV3-SVM* menunjukkan performa yang sangat konsisten dengan akurasi mencapai 98,4% pada penggunaan parameter optimal  $C = 0,1$ . Sebagian besar kelas aksara berhasil dikenali dengan sempurna, ditunjukkan oleh nilai *precision*, *recall*, dan *F1-score* yang mencapai 1,00 pada beberapa kategori seperti ba, ca, da, dha, ja, ma, nga, nya, ra, sa, tha, dan ya. Hal ini menunjukkan bahwa fitur yang dihasilkan memiliki karakteristik yang kuat dan mampu membedakan setiap aksara dengan sangat baik pada data uji sebanyak 250 sampel. Selain itu, penggunaan teknik *preprocessing* CLAHE terbukti memberikan peningkatan kinerja, dengan akurasi mencapai 98,4% dibandingkan tanpa CLAHE sebesar 97,2%. Secara kuantitatif, peningkatan akurasi sebesar 1,2% menunjukkan bahwa *preprocessing* memiliki kontribusi signifikan dalam meningkatkan kualitas sistem klasifikasi.

Secara konseptual, hasil ini sejalan dengan teori bahwa kualitas *input* sangat mempengaruhi hasil ekstraksi fitur dalam *deep learning*. CLAHE bekerja dengan meningkatkan kontras lokal tanpa memperbesar *noise* secara berlebihan, sehingga menghasilkan distribusi piksel yang lebih merata. Hal ini memungkinkan model untuk lebih mudah membedakan pola antar kelas, terutama pada dataset dengan variasi pencahayaan dan kualitas citra yang tidak seragam.

Jika dibandingkan dengan penelitian terdahulu, hasil penelitian ini menunjukkan peningkatan yang lebih tinggi dibandingkan metode berbasis LBP yang hanya mencapai akurasi sekitar 90% [5]. Hal ini menunjukkan bahwa pendekatan berbasis *deep feature extraction* lebih unggul dalam menangkap kompleksitas pola aksara Jawa. Selain itu, hasil yang diperoleh juga mendekati penelitian Shi et al. (2024) yang mencapai akurasi 98,75% menggunakan kombinasi *MobileNetV3-SVM* [6]. Perbedaan kecil ini dapat disebabkan oleh karakteristik dataset yang berbeda, di mana aksara Jawa memiliki tingkat kemiripan visual yang lebih tinggi dibandingkan dataset umum.

Secara keseluruhan, dapat disimpulkan bahwa peningkatan performa tidak hanya disebabkan oleh penggunaan model yang kuat, tetapi juga oleh kualitas data *input* yang telah ditingkatkan melalui *preprocessing*. Dengan demikian, CLAHE berperan penting dalam meningkatkan kemampuan model dalam mengekstraksi fitur yang lebih diskriminatif, sehingga menghasilkan klasifikasi yang lebih akurat dan stabil.

#### 4. Kesimpulan dan Saran

Berdasarkan hasil penelitian, dapat disimpulkan bahwa integrasi *MobileNetV3* sebagai *feature extractor* dan *SVM* sebagai *classifier* mampu menghasilkan sistem pengenalan aksara Jawa yang akurat dan efisien. Penerapan *preprocessing* CLAHE terbukti berperan penting dalam meningkatkan kualitas citra, khususnya dalam memperjelas kontras dan struktur karakter, sehingga menghasilkan fitur yang lebih diskriminatif. Hal ini menunjukkan bahwa kualitas data *input* memiliki pengaruh signifikan terhadap keberhasilan proses ekstraksi fitur dan kinerja model klasifikasi. Secara ilmiah, temuan ini menegaskan bahwa kombinasi teknik *image enhancement* dan *deep feature extraction* merupakan pendekatan yang efektif dalam menangani permasalahan klasifikasi citra dengan tingkat kemiripan visual yang tinggi, seperti pada aksara tradisional.

Selain itu, hasil penelitian ini memberikan implikasi praktis bahwa pendekatan *MobileNetV3-SVM* yang dioptimalkan dengan CLAHE berpotensi untuk diimplementasikan pada sistem pengenalan aksara berbasis *web* atau *mobile* yang ringan namun tetap memiliki akurasi tinggi. Dengan demikian, penelitian ini tidak hanya berkontribusi pada aspek metodologis, tetapi juga pada pengembangan teknologi pelestarian budaya berbasis kecerdasan buatan.

Adapun saran untuk penelitian selanjutnya adalah: (1) melakukan eksplorasi teknik *preprocessing* lain seperti *Gaussian filtering* atau *adaptive thresholding* untuk membandingkan efektivitas peningkatan kualitas citra, (2) menerapkan data *augmentation* seperti rotasi, translasi, dan variasi intensitas guna meningkatkan keragaman dataset dan kemampuan generalisasi model, (3) menguji penggunaan kernel non-linear pada *SVM* atau membandingkannya dengan algoritma klasifikasi lain seperti *Random Forest* atau *Neural Network*, serta (4) melakukan validasi yang lebih kuat seperti *k-fold cross-validation* untuk memastikan konsistensi performa model. Selain itu, penelitian selanjutnya juga dapat mengembangkan sistem ke dalam bentuk aplikasi *real-time* untuk menguji performa model pada kondisi data yang lebih variatif di lingkungan nyata.

#### 5. UCAPAN TERIMA KASIH

Penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada dosen pembimbing I dan dosen pembimbing II atas bimbingan, arahan, serta dukungan yang diberikan selama proses penelitian hingga penyusunan artikel ini. Kontribusi yang diberikan

berupa masukan ilmiah, koreksi, serta motivasi yang sangat membantu dalam menyempurnakan penelitian ini. Selain itu, penulis juga mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan, baik secara langsung maupun tidak langsung, sehingga penelitian ini dapat diselesaikan dengan baik.

#### Daftar Pustaka:

- [1] A. Ali, A. Ali, and S. Mallaiah, "Intelligent handwritten recognition using hybrid CNN architectures based-SVM classifier with dropout," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 6, pp. 3294–3300, 2022, doi: 10.1016/j.jksuci.2021.01.012.
- [2] B. Jiang *et al.*, "Research on facial expression recognition algorithm based on improved MobileNetV3," *EURASIP J. Image Video Process.*, 2024, doi: 10.1186/s13640-024-00638-z.
- [3] S. Ahlawat and A. Choudhary, "ScienceDirect ScienceDirect Hybrid CNN-SVM Classifier for Handwritten Digit Recognition Hybrid CNN-SVM Classifier for Handwritten Digit Recognition," *PROCS*, vol. 167, no. 2019, pp. 2554–2560, 2020, doi: 10.1016/j.procs.2020.03.309.
- [4] L. N. Soni and A. A. Waoo, "Face Detection Under Low-Light and Low-Resolution Conditions Using Contrast-Limited Adaptive Histogram Equalization and a Modified Convolutional Neural Network," vol. 14, no. 32, pp. 2620–2631, 2025.
- [5] C. A. Sari, W. S. Sari, V. A. Shelomita, M. R. Kusuma, S. A. Puspa, and M. Bima Gusta, "The Involvement of Local Binary Pattern to Improve the Accuracy of Multi Support Vector-Based Javanese Handwriting Character Recognition," 2023.
- [6] G. X. Shi, Y. N. Wang, Z. F. Yang, Y. Q. Guo, and Z. W. Zhang, "Wildfire Identification Based on an Improved MobileNetV3-Small Model," *Forests*, vol. 15, no. 11, Nov. 2024, doi: 10.3390/f15111975.
- [7] R. Dijaya, "Buku Ajar Pengolahan Citra Digital," 2023.
- [8] D. Lizard, S. Dimara, B. Rahmat, and H. Maulana, "Identifikasi Penyakit Daun Padi Dengan Metode Transfer Learning Dengan MobileNet - Support Vector Machine," vol. 9, no. 3, pp. 5413–5420, 2025.
- [9] F. Putra *et al.*, "Enhancing Visibility in Low-Illumination Street Images Using HE, AHE, and CLAHE Techniques," vol. 9, no. July, pp. 261–270, 2025.
- [10] H. P. Tulili, A. Septiarini, and Hamdani, "Klasifikasi Penyakit Tanaman Daun Padi Menggunakan Metode Deep Learning dengan Teknik Transfer Learning MobileNet," vol. 8, no. 2, pp. 365–375, 2025, doi: <https://doi.org/10.36595/jire.v8i2.1737>.
- [11] A. Hussain, B. Barua, A. Osman, R. Abozariba, and A. T. Asyhari, "Performance of MobileNetV3 Transfer Learning on Handheld Device-based Real-Time Tree Species Identification."
- [12] S. Ernawati and R. Wati, "Evaluasi Performa Kernel SVM dalam Analisis Sentimen Review Aplikasi ChatGPT Menggunakan Hyperparameter dan VADER Lexicon," 2024.
- [13] A. N. Royana, Y. V. Via, C. A. Putra, J. Raya, R. Madya, and A. Music, "Evaluasi Kinerja LightGBM dan CatBoost untuk Prediksi Churn Berdasarkan Dataset Pelanggan Layanan Streaming Musik," vol. 9, no. 4, pp. 6977–6982, 2025.
- [14] E. Dwi, X. Anindhya, M. D. Paramita, A. P. Sari, S. Informatika, and G. Anyar, "Optimasi Lokasi Pembangunan Rumah Sakit di Kecamatan Ngrayun Kabupaten Ponorogo dengan K-Means," vol. 8, no. 2, pp. 199–205, 2025, doi: <https://doi.org/10.36595/jire.v8i2.1646>.
- [15] Phiard, "Aksara Jawa | Kaggle", [Online]. Available: <https://www.kaggle.com/datasets/phiard/aksara-jawa>