TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

Implementasi Convolutional Neural Network Untuk Pengenalan Tulisan Tangan Akasara Sunda Ngalangéna

Rahma Nur Azizah[*], Donny Avianto

Fakultas Sains & Teknologi, Prgram Studi Informatika, Universitas Teknologi Yogyakarta, Yogyakarta, Indonesia
Email: [1,*]rahma.azizah13@gmail.com, [2]donny@uty.ac.id
Email Penulis Korespondensi: rahma.azizah13@gmail.com

Abstrak–Upaya pelestarian aksara Sunda sebagai warisan budaya menghadapi tantangan di era digital, salah satunya adalah keterbatasan sumber daya untuk pengenalan pola. Penelitian ini bertujuan untuk mengembangkan model *Convolutional Neural Network* (CNN) kustom yang efektif untuk klasifikasi tulisan tangan aksara Sunda. Menghadapi kendala ketiadaan dataset publik, penelitian ini menggunakan dataset primer (Swaraksara Dataset) yang dibuat oleh penulis, terdiri dari 6.500 citra tulisan tangan yang terdistribusi merata dalam 13 kelas (kombinasi aksara "Na" dengan rarangkén). Metodologi yang diterapkan meliputi tahap prapemrosesan data secara komprehensif, mencakup konversi *grayscale*, penyeragaman ukuran 200x200 piksel, normalisasi, dan teknik augmentasi data untuk mencegah *overfitting*. Arsitektur CNN kustom dirancang dengan lima lapisan konvolusi (filter 32 hingga 512) dan optimizer Adam. Hasil eksperimen menunjukkan bahwa konfigurasi optimal dicapai dengan *learning rate* 0.001 dan pelatihan 50 *epoch*, menghasilkan performa model yang sangat tinggi. Pada evaluasi menggunakan data uji, model mencapai akurasi sebesar 99,54% dengan nilai *loss* 0,0175. Performa model yang optimal ini didorong dari kualitas dataset primer yang didukung oleh tahapan prapemrosesan citra yang komprehensif, sehingga menjamin input data yang bersih, seragam, dan bebas dari gangguan (*noise*) yang signifikan. Analisis *confusion matrix* dan kurva belajar juga mengonfirmasi kemampuan generalisasi model yang sangat baik tanpa indikasi *overfitting*. Model ini telah berhasil diimplementasikan dalam aplikasi web "Swaraksara" sebagai sistem pengenalan aksara Sunda.

Kata Kunci: *Convolutional Neural Network*; *Deep Learning*; Klasifikasi Citra; Pengenalan Pola; Aksara Sunda

Abstract–Efforts to preserve the Sundanese script as a cultural heritage face challenges in the digital era, one of which is the limited resources for pattern recognition. This research aims to develop an effective custom Convolutional Neural Network (CNN) model for the classification of handwritten Sundanese script. Facing the constraint of no available public dataset, this study utilizes a primary dataset (Swaraksara Dataset) created by the author, consisting of 6,500 handwritten images evenly distributed across 13 classes (combinations of the "Na" script with rarangkén). The methodology applied includes a comprehensive data preprocessing stage, covering grayscale conversion, resizing to 200x200 pixels, normalization, and data augmentation techniques to prevent overfitting. The custom CNN architecture was designed with five convolutional layers (filters 32 to 512) and the Adam optimizer. The experimental results show that the optimal configuration was achieved with a learning rate of 0.001 and 50 training *epoch*s, resulting in very high model performance. In the evaluation using test data, the model achieved an accuracy of 99.54% with a loss value of 0.0175. The optimal performance of this model is driven by the quality of the primary dataset supported by comprehensive image preprocessing stages, thus ensuring clean, uniform, and significantly noise-free data input. Analysis of the confusion matrix and learning curves also confirmed the model's excellent generalization ability with no indications of overfitting. This model has been successfully implemented in the "Swaraksara" web application as a Sundanese script recognition system.

Keywords: Convolutional Neural Network; Deep Learning; Image Classification; Pattern Recognition; Sundanese Script

1. PENDAHULUAN

Keragaman bahasa dapat dikategorikan berdasarkan media penyampaiannya, yaitu bahasa lisan dan bahasa tulisan. Salah satu bahasa yang memiliki bentuk tulisan tersendiri adalah bahasa Sunda. Di Indonesia, bahasa Sunda termasuk bahasa daerah dengan jumlah penutur terbanyak kedua setelah bahasa Jawa (Wulandari dkk., 2023). Setiap aksara merepresentasikan bunyi konsonan tertentu, dan ketika digabungkan dengan tanda vokal (rarangkén), membentuk satuan suku kata. Setiap huruf konsonan dalam aksara Sunda memiliki vokal bawaan /a/ yang dapat dimodifikasi melalui tanda diakritik untuk menghasilkan variasi bunyi vokal lainnya (Rosalina dkk., 2024). Seiring dengan perkembangan teknologi modern, penerapan teknologi pengenalan pola pada aksara Sunda menjadi langkah penting dalam upaya pelestarian dan penguatan nilai budaya tersebut. Meskipun demikian, pelestarian aksara tradisional ini masih menghadapi berbagai kendala, antara lain rendahnya tingkat pemahaman masyarakat serta keterbatasan sumber daya yang tersedia.

Convolutional Neural Network (CNN) tergolong sebagai algoritma jaringan saraf buatan yang memiliki kelebihan utama berupa tingkat akurasi yang luar biasa dalam proses pengklasifikasian (Hadhiwibowo dkk., 2024). CNN dikategorikan sebagai salah satu jenis Deep Neural Network (DNN) berkat kedalaman arsitektur jaringannya yang tinggi serta penerapan luas pada data citra. CNN merepresentasikan kelas khusus dalam *neural network* yang dirancang secara spesifik untuk memproses data berstruktur *grid-like*, seperti gambar. Metode ini dapat diaplikasikan pada berbagai tugas, termasuk pengenalan wajah, analisis dokumen, klasifikasi gambar, klasifikasi video, serta aplikasi serupa lainnya (Abdiansah dkk., 2025). CNN dipilih karena keunggulannya yang telah terbukti dalam pengenalan pola pada citra, terutama dalam mengatasi variasi visual yang kompleks seperti perbedaan gaya tulisan, ukuran, serta orientasi karakter huruf (Swasono dkk., 2024). Berbagai penelitian sebelumnya juga menunjukkan keberhasilan CNN dalam mengenali beragam jenis aksara, termasuk aksara Sunda. Penelitian ini menggunakan dataset berupa gambar tulisan tangan. Langkah awal adalah mengekstrak fitur dari gambar tersebut, diikuti dengan proses klasifikasi. Ketiadaan dataset publik tulisan tangan aksara sunda menjadi hambatan utama dalam penelitian berbasis pengenalan

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

pola. Oleh karena itu, penulis membuat dataset sendiri sesuai ruang lingkup penelitian, dengan keunggulan kualitas data yang terkontrol dan terdokumentasi secara jelas.

Kajian dengan topik serupa telah dilakukan pada penelitian-penelitian sebelumnya oleh (Rahmawati dkk., 2021) CNN diterapkan untuk mengenali aksara Sunda dari tulisan tangan dengan memanfaatkan dataset gabungan antara tulisan tangan penulis dan citra dari internet, yang terdiri atas 26 kategori huruf. Dataset tersebut dibagi menjadi 80% untuk pelatihan dan 20% untuk validasi. Tahap pra-pemrosesan meliputi pengubahan citra menjadi skala abu-abu, penyesuaian ukuran menjadi 32x32 piksel, normalisasi intensitas piksel melalui pembagian dengan 255, serta penerapan dropout guna mencegah *overfitting*. Model dilatih dengan optimizer Adam dan menghasilkan akurasi 94,44% pada data pelatihan serta 86,46% pada data validasi setelah menjalani 50 *epoch*. Penelitian serupa oleh (Aryanto dkk., 2023) mengimplementasikan CNN untuk pengenalan aksara Lontara Ende dengan memanfaatkan *image augmentation* guna mengatasi *overfitting*. Model yang dilatih selama 100 *epoch* menggunakan *batch size* 32 mencapai akurasi 96,88% pada data pelatihan dan 100% pada data validasi, mengindikasikan kemampuan generalisasi yang sangat baik.

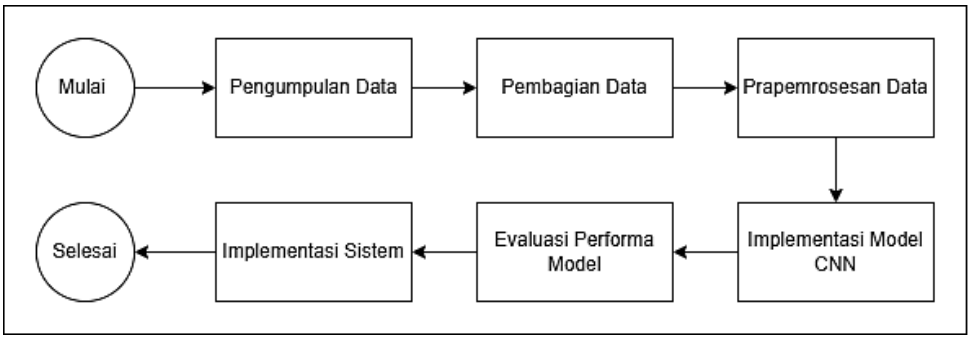
Selain itu penelitian pengenalan tulisan tangan bahasa korea menggunakan arsitektur *AlexNet* oleh (Adella dkk., 2023) menggunakan dataset sebanyak 1.000 citra, yang terbagi menjadi 800 sampel pelatihan dan 200 sampel pengujian. Model dievaluasi dengan tiga optimizer (Adam, SGD, dan RMSprop), di mana SGD mencatat akurasi tertinggi sebesar 78,5%, dengan nilai *precision, recall, dan F1-score* masing-masing berkisar antara 76–80%. Penelitian oleh (Arief dkk., 2024) mengimplementasi CNN Arsitektur *MobileNetV2* Untuk Klasifikasi Tulisan Aksara Jawa, menggunakan 2.644 citra dari Kaggle yang mencakup 20 kelas karakter. Proses prapemrosesan mencakup pengubahan ukuran (224×224 piksel), augmentasi citra (flip, rotasi, zoom), dan normalisasi nilai piksel. Arsitektur MobileNetV2 memanfaatkan *depthwise separable convolution* dan *inverted residual blocks* sehingga total parameter hanya 2.283.604, dengan 25.620 parameter yang dapat dilatih. Model dilatih menggunakan optimizer Adam selama 10 *epoch*, menghasilkan akurasi 87,41% tanpa *fine-tuning* dan 86,46% dengan *fine-tuning*. Hasil menunjukkan model efektif secara umum, namun mengalami kesulitan membedakan karakter dengan bentuk visual yang mirip.

Studi-studi sebelumnya telah membuktikan bahwa CNN efektif untuk pengenalan aksara, namun mayoritas penelitian tersebut, seperti yang dilakukan oleh (Adella dkk., 2023)dan (Arief dkk., 2024), sangat bergantung pada arsitektur transfer learning yang kompleks (AlexNet dan MobileNetV2). Penggunaan arsitektur yang terlalu dalam sering kali berlebihan (inefisien) dan rentan overfitting jika diterapkan pada dataset tulisan tangan aksara Sunda yang terbatas. Belum banyak penelitian yang secara spesifik mengeksplorasi optimasi arsitektur Custom CNN yang dibangun menyesuaikan karakteristik data tersebut. Selain itu, meskipun teknik regularisasi serta augmentasi data terbukti berhasil meningkatkan akurasi, tantangan tetap ada dalam mengenali aksara dengan variabilitas bentuk tulisan tangan yang tinggi. Oleh karena itu, penelitian ini akan menginvestigasi pemanfaatan custom layer CNN untuk mengoptimalkan efisiensi dan akurasi pengenalan aksara Sunda, dengan penekanan pada teknik augmentasi dan regularisasi yang lebih efektif.

Penelitian ini berfokus pada implementasi metode CNN guna mengenali pola aksara Sunda secara akurat, dengan penekanan khusus pada aksara Ngalagena. Selanjutnya, penelitian ini mengevaluasi performa model CNN tersebut dalam mengklasifikasikan karakter aksara Sunda untuk mengukur tingkat akurasinya, dengan menggunakan *dataset* tulisan tangan yang telah melalui tahap pengumpulan dan prapemrosesan. Penelitian ini mengembangkan model CNN custom untuk klasifikasi tulisan tangan aksara sunda, yang kemudian diimplementasikan ke dalam sebuah aplikasi web menggunakan Streamlit.

2. METODOLOGI PENELITIAN

2.1 Kerangka Dasar Penelitian



Gambar 1. Tahapan Penelitian

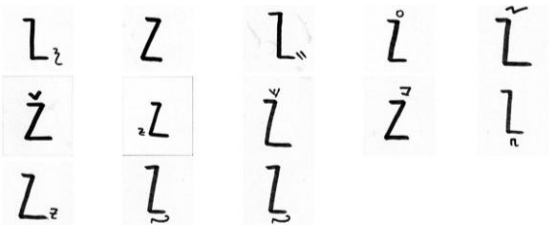
Tahapan penelitian yang diilustrasikan pada Gambar 1 diawali dengan Pengumpulan Data yang relevan, diikuti oleh Pembagian Data (latih, validasi, dan uji). Tahap krusial selanjutnya adalah prapemrosesan Data, yang mencakup teknik-teknik untuk mempersiapkan data dan menangani ketidakseimbangan data agar tidak menurunkan performa model (Putri dkk., 2024). Data yang telah diproses kemudian digunakan untuk Implementasi Model CNN. Kinerja model dievaluasi secara komprehensif melalui Evaluasi Performa Model menggunakan data uji (menganalisis *confusion matrix, precision, recall, dan f1-score*). Tahap terakhir adalah Implementasi Sistem, di mana model yang telah teruji diintegrasikan ke dalam aplikasi fungsional sebelum penelitian dinyatakan Selesai.

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

2.2 Tahapan Penelitian

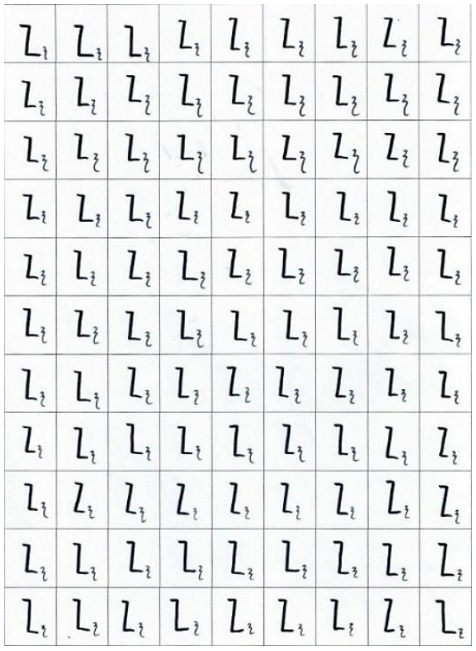
2.2.1 Pengumpulan Data

Tahap awal yang krusial dalam penelitian ini adalah pengumpulan data citra tulisan tangan aksara Sunda. Meskipun aksara Sunda Ngalagéna memiliki 20 karakter dasar, penelitian ini memfokuskan ruang lingkup pada variasi bentuk yang dihasilkan oleh satu aksara dasar, yaitu "Na", yang dikombinasikan dengan sistem vokalisasi atau rarangkén. Total terdapat 13 kelas yang digunakan dalam dataset ini, yang merepresentasikan kombinasi aksara "Na" dengan berbagai posisi rarangkén: 5 bentuk di atas aksara, 3 bentuk di bawah, 4 bentuk sejajar, serta 1 bentuk asli aksara "Na" (tanpa rarangkén). Rincian kelas tersebut meliputi variasi fonetik seperti *Na, Nah, Nang, Nar, Ne, Né, Neu, Ni, Nla, No, Nra, Nu, dan N (Pamaéh)*, sebagaimana diilustrasikan pada Tabel 1 dan Gambar 2.



Gambar 2. Contoh Data Aksara Sunda 13 Kelas

Pengumpulan data dilakukan secara manual oleh penulis (data primer). Untuk menjaga konsistensi ukuran dan mempermudah proses segmentasi, penulisan dilakukan di atas lembar kerja (*worksheet*) yang telah diberi format kotak-kotak (*grid*). Penulis menuliskan satu jenis karakter berulang kali dalam kolom-kolom yang tersedia untuk menangkap variasi tarikan garis tulisan tangan yang alami. Gambar 3 memperlihatkan contoh lembar pengumpulan data mentah untuk kelas "N" (kombinasi Na + Pamaéh) sebelum diproses lebih lanjut. Keseluruhan dataset yang terkumpul berjumlah 6.500 citra, dengan distribusi yang seimbang (*balanced*) sebanyak 500 citra untuk setiap kelas. Dataset ini bersifat terbuka dan dapat diakses melalui repositori GitHub: https://github.com/rhmanr/Dataset-Aksara-Sunda.



Gambar 3. Contoh Lembar Pengumpulan Data Mentah (*Raw Data*) untuk Kelas "N"

2.2.2 Pembagian Data

Dataset yang telah dikumpulkan selanjutnya menjalani tahap prapemrosesan sebelum dibagi menjadi tiga himpunan: data latih (70%), data validasi (20%), dan data uji (10%) dari total sampel dengan 13 kelas. Pembagian data bertujuan pokok untuk mengukur kemampuan model dalam menghadapi data baru yang tidak terlibat dalam fase pelatihan (Putra dkk., 2024).

Tabel 1. Pembagian Data

Class	Training	Validasi	Test
N	350	100	50
Na	350	100	50
Nah	350	100	50

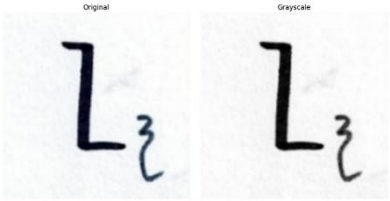
TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

Class	Training	Validasi	Test
Nang	350	100	50
Nar	350	100	50
Ne	350	100	50
Né	350	100	50
Neu	350	100	50
Ni	350	100	50
Nla	350	100	50
No	350	100	50
Nra	350	100	50
Nu	350	100	50
Total	4550	1300	650

Tabel 1 menyajikan rincian distribusi dataset yang dimanfaatkan dalam penelitian ini. Dataset dibagi menjadi tiga bagian utama dengan proporsi 70% untuk data latih (*training*), 20% untuk data validasi (*validation*), dan 10% untuk data uji (*test*). Dari total keseluruhan 6.500 sampel data, 4.550 sampel didedikasikan untuk proses pelatihan model, 1.300 sampel untuk validasi, dan 650 sampel untuk pengujian. Seperti yang dirinci pada tabel, pembagian ini diterapkan secara *stratified* (berlapis) dan seimbang (*balanced*) di ke-13 kelas yang diteliti. Setiap kelas secara konsisten menyumbang 350 sampel untuk data latih, 100 sampel untuk data validasi, dan 50 sampel untuk data uji, guna memastikan bahwa model dilatih dan dievaluasi pada distribusi kelas yang proporsional.

2.2.3 Pra-pemrosesan

Pra-pemrosesan data merupakan tahap krusial dalam analisis data yang memerlukan perhatian khusus. Tahap ini mencakup serangkaian langkah untuk mengolah data mentah menjadi format yang lebih bersih, terstruktur, dan siap digunakan untuk analisis selanjutnya (Rewina dkk., 2024). Dataset yang telah dibagi selanjutnya menjalani tahap pre-processing untuk meningkatkan kualitas data dan mempermudah pemrosesan oleh model.

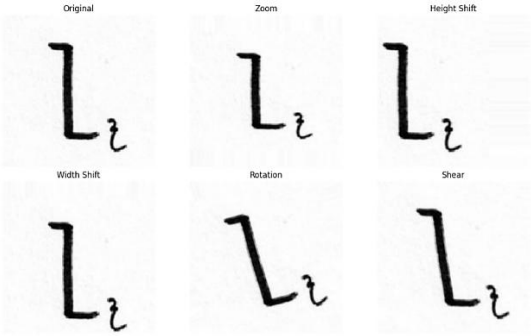


Gambar 4. Citra RGB ke *Grayscale*

Langkah awal pra-pemrosesan adalah konversi citra aksara Sunda ke format *grayscale*. Konversi ini mengurangi kompleksitas data karena setiap piksel hanya memiliki satu saluranan intensitas cahaya, berbeda dengan citra berwarna (Wahid & Ujianto, 2024). Pendekatan ini memungkinkan model CNN lebih fokus pada fitur bentuk dan tekstur aksara—yang merupakan elemen krusial dalam proses pengenalan pola—sekaligus menyederhanakan representasi citra, mempercepat waktu pelatihan, dan memfasilitasi implementasi model secara keseluruhan.

Setelah konversi *grayscale*, tahapan dilanjutkan dengan penyeragaman ukuran (*resizing*). Tahap *resize* melibatkan penyesuaian dimensi citra, di mana citra asli dengan ukuran yang bervariasi diubah skalanya menjadi citra dengan dimensi tetap (Khumaidi & Nurpadilah, 2024). Pada penelitian, ini setiap citra diubah dimensinya menjadi 200 X 200 piksel. Standardisasi ukuran input ini esensial untuk memastikan konsistensi data yang akan diproses oleh arsitektur model. Tahap berikutnya adalah normalisasi intensitas piksel (*rescaling*), proses ini berfungsi untuk menormalisasi intensitas piksel dengan mentransformasikan nilai RGB dari rentang [0, 255] menjadi rentang [0.0, 1.0] (Rodiah dkk., 2025). Secara matematis, proses ini dapat dilihat pada persamaan (1):

$$I^I = \frac{1}{255} \tag{1}$$



Gambar 5. Hasil Augmentasi

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

Terakhir, untuk meningkatkan generalisasi model dan memitigasi risiko *overfitting*, diterapkan teknik augmentasi data. Augmentasi data merupakan teknik *oversampling* yang digunakan untuk memperbanyak jumlah sampel data, dan sering diterapkan guna meningkatkan kinerja model klasifikasi (Nur Azizah dkk., 2023). Proses ini memperkaya dataset latih dengan menghasilkan variasi citra baru melalui serangkaian transformasi geometris secara acak. Transformasi yang diaplikasikan meliputi rotasi (*rotation*), pergeseran lebar (width shift), pergeseran tinggi (*height shift*), shear, dan zoom seperti yang diilustrasikan pada Gambar 5. Dalam pengaplikasian transformasi ini, fill mode '*nearest*' digunakan untuk mengisi nilai piksel yang baru terbentuk di luar batas citra asli.

2.2.4 Implementasi Model CNN

CNN merupakan salah satu varian jaringan saraf tiruan yang dirancang secara spesifik untuk mengolah data citra, dengan efisiensi tinggi dalam tugas klasifikasi, identifikasi, serta pengenalan pola pada gambar (Septian Nugraha dkk., 2025). Pada penelitian ini, model arsitektur CNN dirancang seperti yang ditunjukkan pada Tabel 2. Arsitektur ini terdiri atas lima blok konvolusi (Conv2D) dengan jumlah filter yang bertambah secara bertahap dari 32 hingga 512, menggunakan kernel 3×3 dan aktivasi ReLU untuk mengekstraksi fitur secara hierarkis. Setiap blok konvolusi diikuti lapisan MaxPooling2D (2×2) untuk menurunkan dimensi spasial sekaligus mengendalikan *overfitting*. Output dari rangkaian konvolusi diratakan melalui lapisan *Flatten*, kemudian diproses oleh lapisan Dense berukuran 512 unit dengan aktivasi ReLU guna membentuk representasi fitur tingkat tinggi. Lapisan akhir Dense dengan aktivasi softmax menghasilkan distribusi probabilitas untuk klasifikasi multikelas. Model dikompilasi dengan optimizer Adam, fungsi kerugian *sparse categorical crossentropy*, dan metrik akurasi, dirancang untuk mengoptimalkan keseimbangan antara kemampuan ekstraksi fitur dan efisiensi komputasi pada dataset gambar yang telah diprapemrosesan.

Tabel 2. Arsitektur CNN

Jenis Layer	Output Shape	Parameter
Input Layer	(None, 200, 200, 1)	-
Conv2D) + Relu	(None, 198, 198, 32)	320
MaxPooling2D	(None, 99, 99, 32)	-
Conv2D + Relu	(None, 97, 97, 64)	18,496
MaxPooling2D	(None, 48, 48, 64)	-
Conv2D + Relu	(None, 46, 46, 128)	73,856
MaxPooling2D	(None, 23, 23, 128)	-
Conv2D + Relu	(None, 21, 21, 256)	295,168
MaxPooling2D	(None, 10, 10, 256)	-
Conv2D + Relu	(None, 8, 8, 512)	1,180,160
MaxPooling2D	(None, 4, 4, 512)	-
Flatten	(None, 8192)	-
Dense + Relu	(None, 512)	4,194,816
Dense + Softmax	(None, 13)	6,669

2.2.5 Evaluasi Performa Model

Evaluasi performa model klasifikasi dapat dilakukan dengan berbagai cara, salah satunya menggunakan *confusion matrix* yang memberikan gambaran hubungan antara prediksi dan label sebenarnya. *Confusion matrix* adalah matriks evaluasi yang menyajikan distribusi hasil klasifikasi dalam format tabel, mencakup prediksi benar (*true*) dan prediksi salah (*false*), sehingga memungkinkan analisis mendalam terhadap akurasi sistem klasifikasi serta identifikasi pola kesalahan klasifikasi (Nurhidayat & Dewi, 2023).

Penghitungan metrik evaluasi seperti akurasi, presisi, dan *recall* dapat diperoleh dari formulasi matematis seperti yang ditunjukkan pada persamaan (2) hingga (5):

$$Accuracy = \frac{TN+TP}{TP+FP+FN+TN} \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

$$F-Score = \frac{2*(Recall*Precision)}{Recall+Precision} \quad (5)$$

Penjelasan istilah dalam persamaan evaluasi adalah sebagai berikut:
a. *True Positive* (TP): data yang secara akurat diklasifikasikan sebagai benar oleh model.
b. *True Negative* (TN): data yang secara akurat diklasifikasikan sebagai salah oleh model.
c. *False Positive* (FP): data yang sebenarnya salah tetapi diklasifikasikan sebagai benar oleh model.
d. *False Negative* (FN): data yang sebenarnya benar tetapi diklasifikasikan sebagai salah oleh model.

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

2.2.6 Implementasi Sistem

Sistem pengenalan tulisan tangan aksara Sunda ini dikembangkan dalam bentuk aplikasi berbasis web dengan memanfaatkan bahasa pemrograman Python serta framework Streamlit, yang bersifat *open-source* dan berbasis pada Python, diciptakan guna memudahkan para pengembang dalam merancang aplikasi web interaktif yang berfokus pada domain sains data dan pembelajaran mesin (Toscana dkk., 2024). Antarmuka pengguna sistem ini, bernama Swaraksara, dilengkapi dengan berbagai menu utama meliputi Beranda, Klasifikasi Gambar, Pengenalan *Real-time*, dan Dashboard. Menu-menu tersebut dirancang untuk memfasilitasi akses pengguna secara mudah tanpa memerlukan instalasi perangkat lunak tambahan pada perangkat mereka.

3. HASIL DAN PEMBAHASAN

3.1 Eksperimen dan Optimasi Model

Pada tahapan ini, proses pelatihan dan evaluasi model CNN dilaksanakan dengan menerapkan metode *early stopping* serta evaluasi akhir terhadap data uji. Tabel 4 dan 5 menyajikan hasil eksperimen dari berbagai *hyperparameter* yang diuji coba, di mana setiap pengujian menggunakan arsitektur CNN sebagaimana ditunjukkan dalam Tabel.

Tabel 3. Hasil Pelatihan dan Validasi Model

Epoch	*Learning rate*	Accuracy *Train*	*Loss Train*	Accuracy Validation	*Loss* Validation
20	0.01	7.32%	2.5666	7.69%	2.5654
	0.001	93.38%	0.1848	99.38%	0.0234
	0.0001	93.16%	0.1904	98.08%	0.0701
	0.00001	92.55%	0.2075	97.92%	0.0684
50	0.01	6.73%	2.5665	7.69%	2.5651
	0.001	94.90%	0.1476	99.54%	0.0123
	0.0001	91.32%	0.2373	98.31%	0.0462
	0.00001	68.92%	0.8808	79.62%	0.5493
100	0.01	7.14%	2.5667	7.69%	2.5653
	0.001	94.59%	0.1399	99.62%	0.0162
	0.0001	89.76%	0.2854	97.00%	0.0878
	0.00001	70.92%	0.8255	83.85%	0.4664

Tabel 3 menyajikan metrik kinerja yang terperinci dari fase pelatihan dan validasi model CNN pada berbagai konfigurasi *hyperparameter*. Tabel ini memaparkan nilai akurasi dan *loss* untuk *dataset* pelatihan (*Accuracy Train*, *Loss Train*) serta dataset validasi (*Accuracy Validation*, *Loss Validation*) pada setiap skenario pengujian yang dibedakan berdasarkan jumlah *epoch* (20, 50, 100) dan *learning rate* (0.01 hingga 0.00001). Data ini esensial untuk menganalisis dinamika proses pembelajaran model, mengidentifikasi kombinasi *hyperparameter* yang konvergen, serta mendeteksi fenomena *overfitting* yang ditandai dengan adanya disparitas signifikan antara metrik kinerja pelatihan dan validasi.

Tabel 4. Hasil Pengujian Model

Epoch	*Learning rate*	Accuracy *Test*	*Loss Test*
20	0.01	7.69%	2.5650
	0.001	98.92%	0.0378
	0.0001	98.31%	0.0607
	0.00001	99.38%	0.0466
50	0.01	7.69%	2.5651
	0.001	99.54%	0.0175
	0.0001	98.62%	0.0526
	0.00001	83.54%	0.5489
100	0.01	7.69%	2.5650
	0.001	99.23%	0.0274
	0.0001	97.85%	0.0828
	0.00001	82.92%	0.5167

Tabel 4 menyajikan evaluasi final atas kinerja model, yang diukur menggunakan dataset pengujian (*test* dataset) yang independen. Metrik *Accuracy Test* dan *Loss Test* merepresentasikan tolok ukur objektif terhadap kemampuan generalisasi model pada data yang sama sekali baru dan belum pernah ditemui sebelumnya. Hasil ini merupakan indikator utama keberhasilan model setelah proses pelatihan dan validasi selesai. Analisis pada tabel ini menunjukkan akurasi pengujian tertinggi yang dicapai untuk setiap kelompok *epoch*: pada 20 *epoch*, akurasi puncak sebesar 99,38% diperoleh dengan *learning rate* 0.00001; pada 50 *epoch*, akurasi tertinggi mencapai 99,54% dengan *learning rate* 0.001; dan pada 100 *epoch*, akurasi maksimal yang tercatat adalah 99,23% dengan *learning rate* 0.001.

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

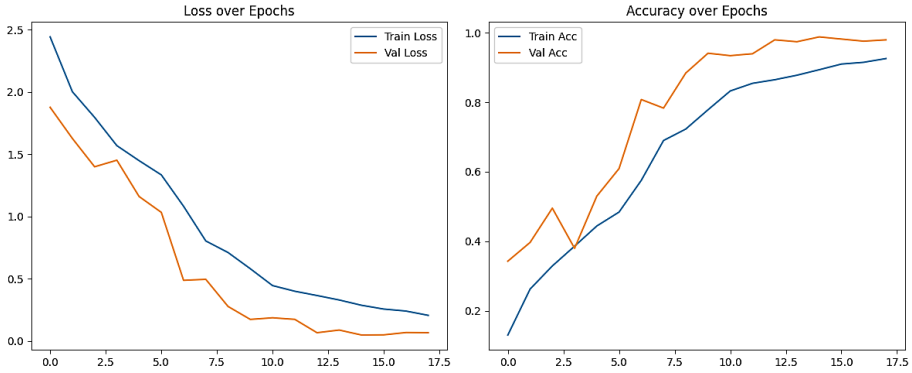
3.2 Analisis Kinerja Model Terbaik

Berikut ini adalah rincian performa yang dicapai model, menunjukkan perbandingan akurasi dan loss antara data training, validasi, dan test.

Tabel 5. Perbandingan Performa Model pada *Epoch*

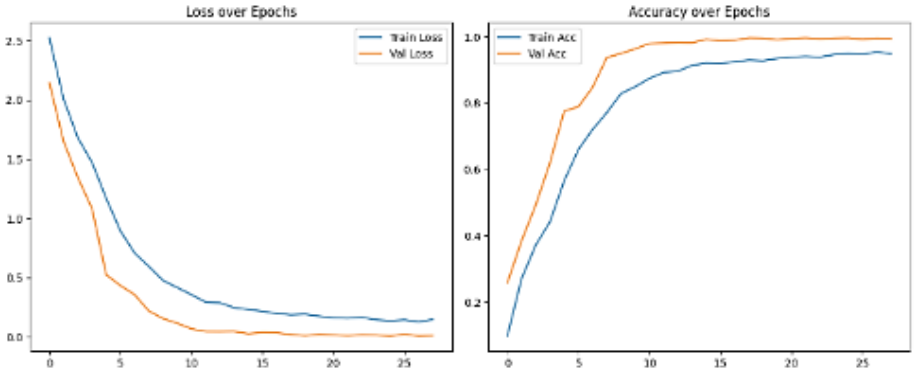
Epoch	*nning*	Akurasi *Validation*	Accuracy *Test*	*Loss Trainning*	*Loss Validation*	*Loss Test*
20	92.55%	97.92%	99.38%	0.2075	0.0684	0.0466
50	94.90%	99.54%	99.54%	0.1476	0.0123	0.0175
100	94.59%	99.62%	99.23%	0.1399	0.0162	0.0274

Tabel 5 menyajikan rangkuman metrik performa akhir dari tiga skenario pelatihan model yang berbeda, yang kesemuanya berhasil dioptimalkan menggunakan mekanisme *Early Stopping* dengan *patience* 3. Pada skenario pertama (baris 20 *epoch*), pelatihan yang diatur untuk 20 *epoch* berhenti di *epoch* ke-18 (menyimpan bobot terbaik dari *epoch* ke-15), menghasilkan *Final Validation Accuracy* 97.92% dan *Test Accuracy* 99.38%. Selanjutnya, skenario kedua (baris 50 *epoch*) menunjukkan proses yang diatur untuk 50 *epoch* namun berhenti di *epoch* ke-28 (menyimpan bobot dari *epoch* ke-25), yang mencapai *Final Validation Accuracy* 99.54% dan *Test Accuracy* 99.54% yang konsisten. Terakhir, skenario ketiga (baris 100 *epoch*) merangkum pelatihan yang diatur untuk 100 *epoch* tetapi berhenti di *epoch* ke-22 (menyimpan bobot dari *epoch* ke-19), yang mencapai *Final Validation Accuracy* 99.62% dan *Test Accuracy* 99.23%. Dalam ketiga kasus tersebut, fakta bahwa *Validation Accuracy* secara konsisten lebih tinggi daripada *Training Accuracy*, dan *Validation Loss* lebih rendah dari *Training Loss*, menjadi konfirmasi kuat bahwa model-model tersebut tidak mengalami *overfitting* dan memiliki kemampuan generalisasi yang sangat baik pada data uji yang sepenuhnya baru.



Gambar 6. Kurva Belajar Model Terbaik *Epoch* 20

Gambar 6 menampilkan kurva *loss* dan akurasi yang menunjukkan bahwa model berhasil dilatih dengan performa sangat baik. Pelatihan ini awalnya diatur untuk 20 epoch namun berhenti secara otomatis pada *epoch* ke-18 karena penerapan mekanisme *Early Stopping* yang memantau *validation accuracy* dengan *patience* 3. Analisis grafik menunjukkan *validation accuracy* (garis *oranye*) mencapai performa puncaknya (sekitar 98%) pada *epoch* ke-15, dan karena tidak ada peningkatan performa yang melampaui nilai ini selama tiga epoch berikutnya (16, 17, 18), pelatihan dihentikan.

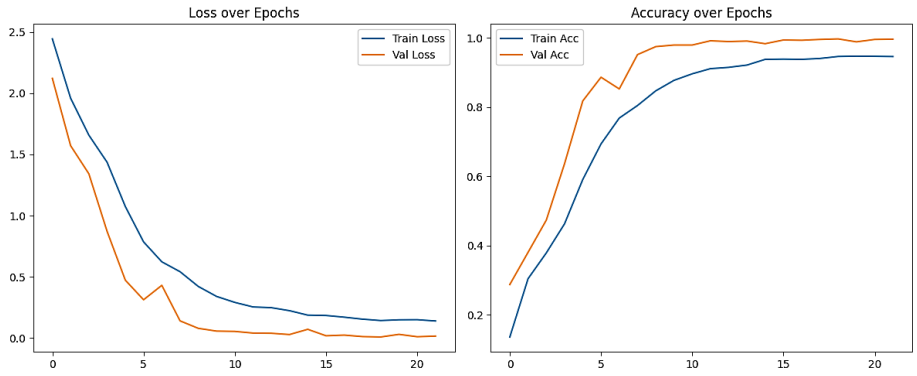


Gambar 7. Kurva Belajar Model Terbaik *Epoch* 50

Gambar 7 menyajikan kurva belajar (learning curves) model, yang memetakan metrik loss dan akurasi pada data latih dan data validasi di setiap *epoch*. Meskipun model dirancang untuk berlatih hingga 50 *epoch*, sebuah mekanisme EarlyStopping diimplementasikan dengan kriteria patience (batas toleransi) selama 3 *epoch* untuk mencegah overfitting dan meningkatkan efisiensi. Seperti yang ditunjukkan pada grafik, pelatihan berhenti secara otomatis pada *epoch* ke-28.

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

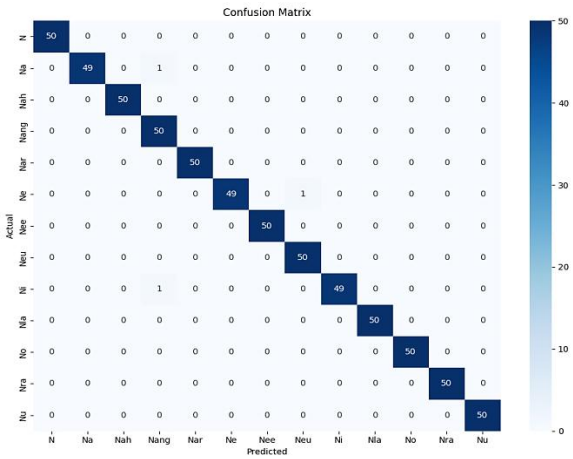
Ini mengindikasikan bahwa metrik yang dipantau (kemungkinan besar *validation loss*) tidak menunjukkan perbaikan (penurunan) yang signifikan selama 3 *epoch* berturut-turut



Gambar 8. Kurva Belajar Model Terbaik *Epoch* 100

Gambar 8 menampilkan kurva *loss* dan akurasi dari proses pelatihan model yang awalnya dikonfigurasi untuk berjalan selama 100 *epoch*. Pelatihan berhenti secara otomatis setelah *epoch* ke-22 (terlihat pada grafik sebagai *epoch* 0 hingga 21) karena penerapan mekanisme *Early Stopping* dengan *patience* 3, yang memantau performa model pada data validasi. Pemberhentian ini mengindikasikan bahwa performa optimal model (seperti *validation accuracy* tertinggi atau *validation loss* terendah) dicapai pada *epoch* ke-18, dan karena tidak ada peningkatan performa yang melampaui capaian tersebut selama tiga *epoch* berturut-turut (*epoch* 19, 20, dan 21), pelatihan dihentikan untuk menyimpan bobot model terbaik. Kurva menunjukkan model belajar dengan sangat efektif, namun hal yang paling menonjol adalah *Validation Loss* (garis *oranye*) secara konsisten berada di bawah *Training Loss* (biru) dan *Validation Accuracy* (*oranye*) berada di atas *Training Accuracy* (biru).

Analisis kurva pelatihan pada Gambar 6, 7, dan 8 menunjukkan efektivitas penerapan mekanisme *Early Stopping* dengan *patience* 3 untuk mengoptimalkan pelatihan. Secara khusus, analisis pada Gambar 6 dan 8 menyoroti kesenjangan (gap) yang ideal, di mana *Validation Loss* secara konsisten lebih rendah daripada *Training Loss* dan *Validation Accuracy* lebih tinggi, yang menjadi konfirmasi kuat bahwa model tidak mengalami *overfitting* dan mampu menggeneralisasi data baru dengan sangat baik



Gambar 9. *Confussion Matrix*

Gambar 9 menyajikan *confussion matrix* yang merinci performa klasifikasi model terhadap 650 sampel uji. Data uji ini terdistribusi secara merata ke dalam 13 kelas aksara Sunda (varian awalan "N"), dengan masing-masing kelas terdiri dari 50 sampel. Matriks tersebut mengindikasikan performa yang sangat tinggi, di mana mayoritas prediksi terkonsentrasi pada diagonal utama (merepresentasikan true positives). Teridentifikasi hanya tiga *instans misclassification* dari total 650 sampel: satu sampel kelas 'Na' salah diklasifikasi sebagai 'Nang', satu sampel 'Ne' sebagai 'Nee (Né)', dan satu sampel 'Ni' sebagai 'Nang'. Dengan demikian, model mencapai tingkat akurasi global (*overall accuracy*) sebesar 99,54% (647/650). Hasil ini mengonfirmasi efektivitas model, sekaligus menyoroti adanya kerentanan kesalahan klasifikasi minor antar kelas yang memiliki kemiripan visual tinggi. Evaluasi model pada data uji menunjukkan performa yang sangat tinggi. Model berhasil mencapai akurasi testing sebesar 99,54%. Nilai ini diperoleh dari 647 klasifikasi yang benar dan hanya 3 instans misclassification dari total 650 sampel data uji. Tingkat keberhasilan ini mengonfirmasi kemampuan generalisasi model yang sangat baik.

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

3.3 Implementasi Sistem

Swaraksara merupakan sistem klasifikasi pola tulisan tangan aksara Sunda yang dibangun dengan memanfaatkan arsitektur Convolutional Neural Network (CNN) sebagai solusi digital dalam upaya pelestarian aksara Sunda yang semakin jarang digunakan. Sistem ini menyediakan empat menu utama dengan fungsi spesifik, yaitu menu Beranda yang berfungsi sebagai halaman awal berisi gambaran umum dan panduan penggunaan aplikasi, menu Klasifikasi Gambar yang memungkinkan pengguna mengunggah foto tulisan tangan aksara Sunda untuk diproses dan diidentifikasi oleh model CNN, menu Pengenalan Real-time yang memanfaatkan kamera perangkat untuk melakukan deteksi aksara secara langsung dan instan, serta menu Dashboard yang menyajikan visualisasi data hasil klasifikasi, statistik penggunaan, dan informasi performa model untuk keperluan evaluasi dan pengembangan sistem lebih lanjut.

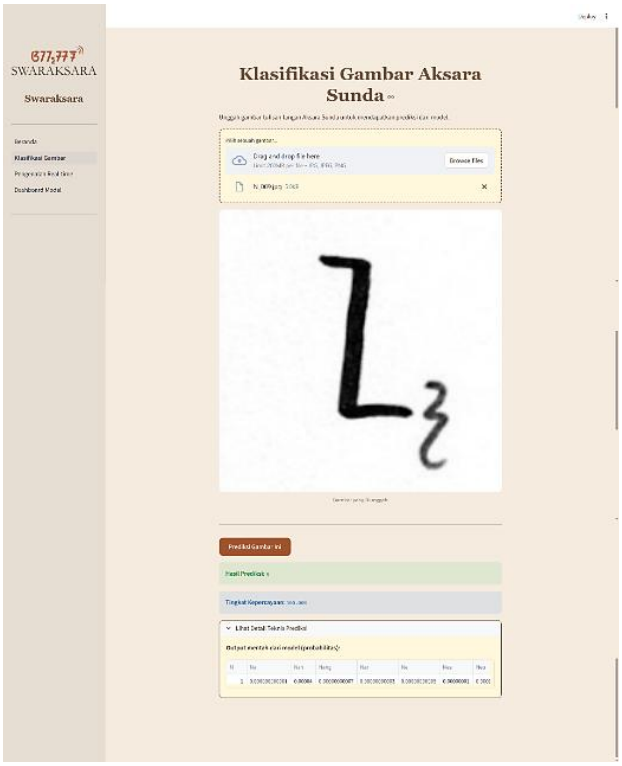
Gambar 10. Menu Beranda

Gambar 10 merepresentasikan antarmuka pengguna utama dari aplikasi Swaraksara. Halaman ini dirancang sebagai gerbang interaksi utama yang menggabungkan estetika budaya dengan fungsionalitas teknis. Secara visual, penerapan skema warna bernuansa krem dan cokelat tidak hanya memberikan kenyamanan visual, tetapi juga merefleksikan nuansa naskah kuno, selaras dengan objek penelitian yaitu Aksara Sunda. Dari sisi fungsionalitas, tata letak aplikasi mengadopsi struktur navigasi *sidebar* statis di sisi kiri untuk menjamin efisiensi aksesibilitas pengguna (*user accessibility*). Menu ini memfasilitasi transisi yang mulus antar modul komputasi utama, yaitu: Klasifikasi Gambar untuk pemrosesan citra statis, Pengenalan Real-time yang memanfaatkan input kamera secara langsung, serta Dashboard Model yang menyediakan transparansi terkait performa dan arsitektur model. Sebagai halaman muka, antarmuka ini juga secara eksplisit menginformasikan kepada pengguna bahwa sistem ini dibangun di atas arsitektur *Deep Learning* berbasis *Convolutional Neural Network* (CNN).

Gambar 11. Menu Klasifikasi Gambar

Antarmuka pengguna (UI) untuk modul Klasifikasi Gambar pada aplikasi Swaraksara dirancang secara khusus untuk memfasilitasi pengujian dan demonstrasi model CNN pengenalan aksara Sunda. Gambar 12 menyajikan visualisasi tata letak antarmuka ini, yang menyoroti halaman khusus (diaktifkan melalui *sidebar*) dengan area input terpusat yang fungsional. Area ini memungkinkan pengguna mengunggah file citra tulisan tangan Aksara Sunda melalui mekanisme *drag-and-drop* atau dialog "Browse files", didukung oleh petunjuk visual yang jelas.

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703



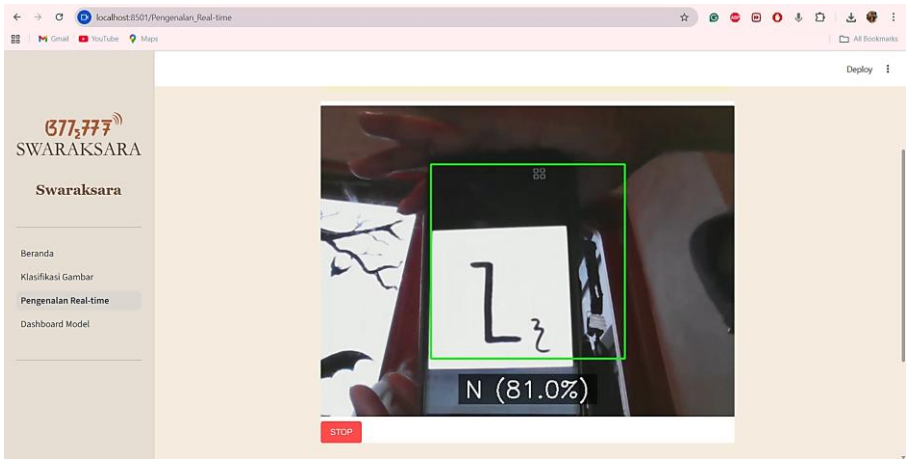
Gambar 12. Hasil Klasifikasi Gambar

Melengkapi desain tersebut, Gambar 12 mendemonstrasikan alur fungsional penuh dari antarmuka ini saat digunakan. Proses dimulai saat pengguna mengunggah citra (contoh: N_009.jpg) dan menekan tombol "Prediksi Gambar Ini" untuk memicu inferensi model. Aplikasi kemudian secara dinamis menampilkan hasil prediksi (contoh: N) beserta tingkat kepercayaannya (contoh: 100.00%), dan juga menyediakan bagian "Lihat Detail Teknis Prediksi" yang dapat diperluas untuk analisis mendalam terhadap output mentah atau vektor probabilitas dari lapisan *softmax* model.

Gambar 13. Menu Pengenalan *Real-time*

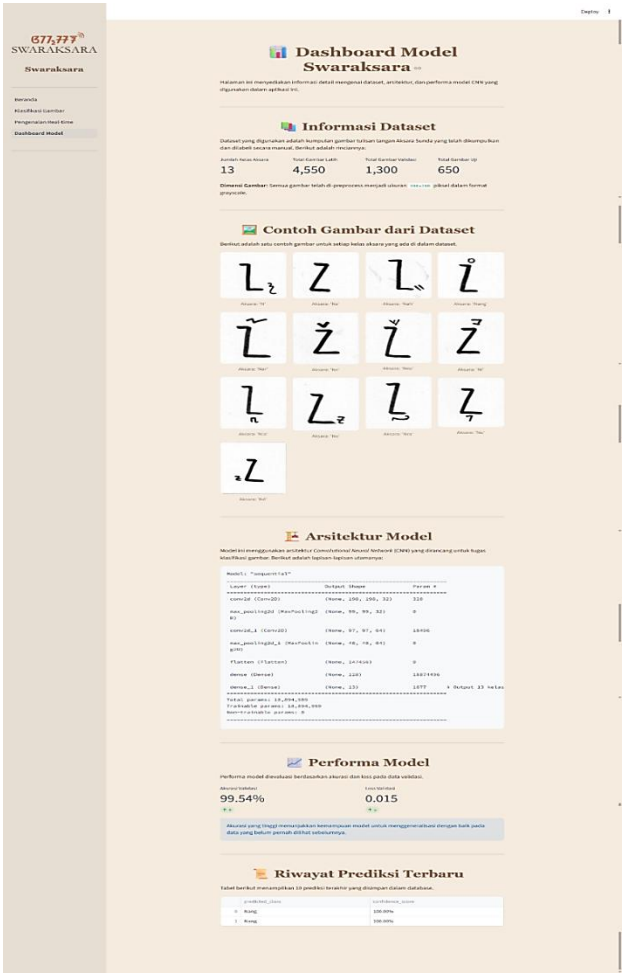
Gambar 13 menunjukkan antarmuka pengguna (UI) halaman "Pengenalan *Real-time*" pada kondisi awal sebelum proses inferensi diaktifkan. Halaman ini dirancang untuk memfasilitasi klasifikasi Aksara Sunda secara langsung melalui beberapa komponen kunci. Di bawah judul halaman, terdapat instruksi yang jelas: "Arahkan kamera ke tulisan tangan Aksara Sunda" untuk memberikan panduan kepada pengguna mengenai tujuan fitur ini. Komponen interaktif utama meliputi frame video (yang awalnya berupa placeholder), tombol "*START*" berwarna merah sebagai *call-to-action* (CTA) untuk menginisiasi akses kamera, serta opsi "*SELECT DEVICE*" yang memungkinkan pengguna memilih sumber kamera spesifik apabila terdapat lebih dari satu perangkat yang terhubung (misalnya, webcam internal laptop dan webcam eksternal). Untuk mengoptimalkan akurasi pengenalan, aplikasi menyediakan panduan operasional dalam kotak peringatan berwarna kuning yang menginstruksikan pengguna untuk "Pastikan pencahayaan cukup dan posisi aksara berada di tengah frame." Selain itu, terdapat catatan di bagian bawah yang menegaskan bahwa fitur ini bersifat eksperimental. Pemberitahuan ini berfungsi untuk mengatur ekspektasi pengguna dengan menekankan bahwa kinerja dan akurasi prediksi bergantung pada faktor-faktor eksternal seperti kualitas kamera, kondisi pencahayaan lingkungan, dan kejelasan tulisan tangan.

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703



Gambar 14. Hasil Pengenalan *Real-time*

Gambar 14 mendemonstrasikan fungsionalitas utama halaman "Pengenalan *Real-time*" pada aplikasi Swaraksara yang dirancang sebagai bagian terintegrasi dari antarmuka pengguna. Halaman ini memanfaatkan webcam untuk menangkap video dan melakukan inferensi model secara langsung (*real-time*). Pada contoh yang ditampilkan, kamera diarahkan ke layar *smartphone* yang menampilkan tulisan tangan Aksara Sunda, menunjukkan fleksibilitas sistem dalam mengenali aksara dari berbagai sumber, baik media fisik maupun digital sekunder. Proses inferensi menggabungkan dua tugas utama: deteksi objek dan klasifikasi. Model melakukan lokalisasi dengan menggambar kotak pembatas (*bounding box*) berwarna hijau di sekitar aksara yang terdeteksi, dan secara bersamaan menampilkan hasil klasifikasi di bawahnya, yaitu karakter "N" dengan skor kepercayaan (*confidence score*) sebesar 81,0%. Penampilan skor kepercayaan ini memberikan transparansi kepada pengguna mengenai tingkat keyakinan model terhadap prediksinya. Sebagai kontrol pengguna, tersedia tombol "*STOP*" yang memungkinkan penghentian aliran video dan proses pengenalan setiap saat.



Gambar 15. Menu *Dashboard*

TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

Gambar 15 menunjukkan *Dashboard* Model "Swaraksara" yang menyediakan tinjauan komprehensif mengenai model CNN yang digunakan, berfungsi sebagai alat pelaporan tanpa kemampuan untuk melakukan pelatihan ulang. Dasbor ini terdiri dari lima segmen utama yang saling melengkapi. Pertama, bagian Informasi Dataset menyajikan ringkasan statistik yang menunjukkan bahwa model dilatih menggunakan 13 kelas simbol Aksara Sunda dengan total 4.550 gambar latih, 1.300 gambar validasi, dan 650 gambar uji. Segmen Contoh Gambar dari Dataset menampilkan 13 sampel citra yang merepresentasikan setiap kelas yang dipelajari model. Bagian Arsitektur Model memvisualisasikan struktur CNN dalam bentuk ringkasan teks yang memperlihatkan lapisan-lapisan kunci seperti *Conv2D, MaxPooling2D, Flatten, dan Dense*, dengan total 10.694.305 parameter yang menunjukkan kompleksitas arsitektur. Bagian Performa Model mengkuantifikasi kinerja dengan akurasi 99,54% dan nilai loss 0,015. Terakhir, bagian Riwayat Prediksi Terbaru menampilkan log inferensi dalam bentuk tabel yang mencakup gambar yang diunggah, hasil prediksi model, dan waktu kejadian.

4. KESIMPULAN

Penelitian ini telah berhasil mengembangkan dan mengimplementasikan model *Convolutional Neural Network* (CNN) kustom untuk klasifikasi tulisan tangan aksara Sunda. Dengan memanfaatkan dataset primer yang dibuat penulis (Swaraksara Dataset) sebanyak 6.500 citra untuk 13 kelas, model ini menunjukkan performa yang sangat tinggi setelah melalui serangkaian tahapan prapemrosesan, termasuk konversi *grayscale*, resizing 200x200 piksel, normalisasi, dan augmentasi data. Arsitektur CNN kustom yang terdiri dari lima lapisan konvolusi terbukti efektif dalam mengekstraksi fitur-fitur kompleks dari citra tulisan tangan. Melalui eksperimen hyperparameter, ditemukan bahwa konfigurasi optimal dicapai menggunakan optimizer Adam dengan *learning rate* 0.001 dan pelatihan selama 50 *epoch*. Hasil evaluasi akhir pada data uji menunjukkan pencapaian yang sangat memuaskan, dengan tingkat akurasi mencapai 99,54% dan nilai *loss* serendah 0,0175. Analisis *confusion matrix* lebih lanjut mengonfirmasi performa model, di mana hanya terjadi 3 kesalahan klasifikasi dari total 650 sampel data uji. Sebagai bukti konsep dan implementasi praktis, model ini telah diintegrasikan ke dalam aplikasi web "Swaraksara" berbasis Streamlit. Meskipun demikian, penelitian ini masih memiliki beberapa keterbatasan yang menjadi celah untuk pengembangan selanjutnya. Pertama, lingkup klasifikasi saat ini baru terbatas pada 13 kelas aksara swara dan belum mencakup keseluruhan sistem aksara Sunda (seperti ngalagena dan rarangken). Kedua, fitur pengenalan real-time yang disajikan masih bersifat eksperimental dan kinerjanya sangat dipengaruhi oleh kondisi pencahayaan serta resolusi kamera pengguna. Oleh karena itu, pengembangan selanjutnya disarankan untuk memperluas dataset dengan variasi karakter yang lebih kompleks dan mengimplementasikan model pada platform *mobile* untuk pengujian ketahanan model di kondisi pencahayaan nyata yang lebih beragam.

REFERENCES

Abdiansah, L., Sumarno, S., Eviyanti, A., & Azizah, N. L. (2025). Penerapan Algoritma Convolutional Neural Networks untuk Pengenalan Tulisan Tangan Aksara Jawa. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *5*(2), 496–504. https://doi.org/10.57152/malcom.v5i2.1814

Adella, V. G., Rusbandi, & Devella, S. (2023). Pengenalan Tulisan Tangan Bahasa Korea Menggunakan Convolutional Neural Network Arsitektur ALEXNET. *JIKI (Jurnal Ilmu Komputer dan Informatika)*, *4(1)*, 1–7. https://doi.org/https://doi.org/10.24127/jiki.v4i1.3311

Arief, N. S., Putra, W. A., Septhian, D., & Pratama, R. (2024). *Implementasi Cnn Arsitektur Mobilenetv2 Untuk Klasifikasi Tulisan Aksara Jawa* (Vol. 3). https://doi.org/https://doi.org/10.29407/z1cbjw36

Aryanto, R., Alfan Rosid, M., & Busono, S. (2023). Penerapan Deep Learning untuk Pengenalan Tulisan Tangan Bahasa Aksara Lota Ende dengan Menggunakan Metode Convolutional Neural Networks. *Jurnal Informasi dan Teknologi*, *5*(1), 258–264. https://doi.org/10.37034/jidt.v5i1.313

Hadhiwibowo, A., Asri, S. R., & Dinata, R. A. (2024). Penerapan Convolutional Neural Network dengan Arsitektur Mobilenetv2 Pada Aplikasi Penerjemah dan Pembelajaran Bahasa Isyarat. *TIN: Terapan Informatika Nusantara*, *4*(8), 518–523. https://doi.org/10.47065/tin.v4i8.4879

Khumaidi, A., & Nurpadilah, A. (2024). Klasifikasi Molting Kepiting Soka Menggunakan Algoritma Convolusional Neural Network. *KOMPUTA: Jurnal Ilmiah Komputer dan Informatika*, *13*(2), 43. https://doi.org/https://doi.org/10.34010/komputa.v13i2.13984

Nur Azizah, A., Falach Asy'ari, M., Wisma Dwi Prastya, I., & Purwitasari, D. (2023). Easy Data Augmentation untuk Data yang Imbalance pada Konsultasi Kesehatan Daring. *Jurnal Teknologi Informasi dan Ilmu Komputer*, *10*(5), 1095–1104. https://doi.org/10.25126/jtiik.2023107082

Nurhidayat, R., & Dewi, K. E. (2023). *Penerapan Algoritma K-Nearest Neighbor Dan Fitur Ekstraksi N-Gram Dalam Analisis Sentimen Berbasis Aspek*. *12*(1), 91–100. https://doi.org/https://doi.org/10.34010/komputa.v12i1.9458

Putra, F., Tahiyat, H. F., Ihsan, R. M., Rahmaddeni, R., & Efrizoni, L. (2024). Penerapan Algoritma K-Nearest Neighbor Menggunakan Wrapper Sebagai Preprocessing untuk Penentuan Keterangan Berat Badan Manusia. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *4*(1), 273–281. https://doi.org/10.57152/malcom.v4i1.1085


TIN: Terapan Informatika Nusantara
Vol 6, No 7, December 2025, page 1037-1049
ISSN 2722-7987 (Media Online)
Website https://ejurnal.seminar-id.com/index.php/tin
DOI 10.47065/tin.v6i7.8703

Putri, C. N., Qornain, W. D., Bamahri, F., Yuliastuti, G. E., & Kurniawan, M. (2024). Klasifikasi Jenis Jerawat pada Data Citra Jerawat Wajah Menggunakan Convolutional Neural Network. *TIN: Terapan Informatika Nusantara*, *5*(2), 172–181. https://doi.org/10.47065/tin.v5i2.5231

Rahmawati, S. N., Hidayat, E. W., & Mubarok, H. (2021). Implementasi Deep Learning Pada Pengenalan Aksara Sunda Menggunakan Metode Convolutional Neural Network. *INSERT: Information System and Emerging Technology Journal*, *2*(1), 46. https://doi.org/https://doi.org/10.23887/insert.v2i1.37405

Rewina, A. E., Sulistyowati, S., Kurniawan, M., N, M. D., & Yunanda, S. F. (2024). Penerapan Metode CNN (Convolutional Neural Network) dalam Mengklasifikasi Uang Kertas dan Uang Logam. *TIN: Terapan Informatika Nusantara*, *4*(12), 778–785. https://doi.org/10.47065/tin.v4i12.5128

Rodiah, Susetianingtias, D. T., & Patriya, E. (2025). Model Convolutional Neural Network (CNN) Custom Sequential untuk Klasifikasi Citra Ikan Hias Carassius Auratus pada Industri Akuakultur. *Decode: Jurnal Pendidikan Teknologi Informasi*, *5*(3), 813–825. https://doi.org/10.51454/decode.v5i3.1229

Rosalina, Afriliana, N., Utomo, W. H., & Sahuri, G. (2024). Deep learning utilization in Sundanese script recognition for cultural preservation. *Indonesian Journal of Electrical Engineering and Computer Science*, *36*(3), 1759–1768. https://doi.org/10.11591/ijeecs.v36.i3.pp1759-1768

Septian Nugraha, M., Dewi Ariessanti, H., & Akbar, H. (2025). Komparasi Model CNN untuk Klasifikasi Citra Pakaian Adat Tradisional Indonesia. *JIM: Jurnal Ilmu Multidisplin*, *4*(4), 2404–2416. https://doi.org/https://doi.org/10.38035/jim.v4i4.1245

Swasono, N. E., Himamunanto, A. R., & Budiati, H. (2024). Pengenalan Karakter Huruf Pada Gambar Tulisan Tangan Menggunakan Algoritma Convolutional Neural Network dan K-Means Clustering. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *4*(4), 1646–1656. https://doi.org/10.57152/malcom.v4i4.1451

Toscana, A. Z., Setianingsih, C., & Paryasto, M. W. (2024). Integrasi Streamlit pada Aplikasi Berbasis Web dengan Algoritma YOLO V8 dan Teknologi Drone untuk Identifikasi Jenis dan Estimasi Tinggi Pohon. *e-Proceeding of Engineering*, 1828–1831. https://www.kdnuggets.com/2019/10/write-web-

Wahid, M. H., & Ujianto, E. I. H. (2024). Efficient Pattern Recognition of Sundanese Script Variants Using CNN. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *8*(6), 808–818. https://doi.org/10.29207/resti.v8i6.6122

Wulandari, L. S., Rosalina, E., & Shomami, A. (2023). Adaptive Learning of Sundanese Script Based on Android Games in The Digital Era. *P2M STKIP Siliwangi*, *10*(1), 25–39. https://doi.org/https://doi.org/10.22460/p2m.v10i1.3747