

Implementation of Convolutional Neural Networks Algorithm for Javanese Handwriting Recognition

Penerapan Algoritma Convolutional Neural Networks untuk Pengenalan Tulisan Tangan Aksara Jawa

Lutfi Abdiansah¹, Sumarno², Ade Eviyanti³, Nuril Lutvi Azizah⁴

^{1,2,3,4}Program Studi Informatika, Fakultas Sains dan Teknologi,
Universitas Muhammadiyah Sidoarjo, Jawa Timur, Indonesia

E-Mail: ¹lutfiabdiansah18@gmail.com, ²sumarno@umsida.ac.id,
³adeeviyanti@umsida.ac.id, ⁴nurillutviazizah@umsida.ac.id

Abstract

Javanese script is a traditional writing system that was once widely used in East Java and Central Java, consisting of 20 main characters along with several additional attributes. However, its usage in daily life has significantly declined over time. This study aims to develop a Javanese script recognition system using a Convolutional Neural Network (CNN) as an effort to preserve it. The dataset utilized consists of 1,000 images of handwritten Javanese script, with 700 images allocated for training and 300 images for validation. The research process includes data collection, preprocessing, CNN architecture development, and model evaluation. The CNN architecture is designed to capture the key features of the script, including distinguishing visually similar characters. Evaluation results indicate strong performance, achieving 99,83% accuracy in recognizing input Javanese script, with consistent accuracy and loss graphs between training and validation data. These findings demonstrate the significant potential of CNN in Javanese script recognition, although further optimization is required to enhance accuracy and system efficiency for broader applications.

Keyword: Accuracy; Convolutional Neural Network; Deep Learning; Handwritten Recognition; Javanese Script

Abstrak

Aksara Jawa adalah sistem tulisan tradisional yang dulunya banyak digunakan di Jawa Timur dan Jawa Tengah, terdiri dari 20 huruf utama serta beberapa atribut tambahan. Namun, penggunaannya dalam kehidupan sehari-hari semakin berkurang. Penelitian ini bertujuan mengembangkan sistem pengenalan aksara Jawa menggunakan Convolutional Neural Network (CNN) sebagai upaya pelestarian. Dataset yang digunakan mencakup 1000 citra tulisan tangan aksara Jawa, dengan 700 citra untuk pelatihan dan 300 citra untuk validasi. Proses penelitian meliputi pengumpulan data, preprocessing, pembangunan arsitektur CNN, dan evaluasi model. Arsitektur CNN dirancang untuk menangkap fitur utama aksara, termasuk membedakan huruf yang memiliki kemiripan visual. Hasil evaluasi menunjukkan performa yang baik, dengan akurasi mencapai 99,83% pada pengenalan aksara yang diinputkan, serta grafik akurasi dan loss yang konsisten antara data pelatihan dan validasi. Temuan ini menunjukkan bahwa CNN memiliki potensi besar dalam pengenalan aksara Jawa, meskipun optimasi lebih lanjut tetap diperlukan untuk meningkatkan akurasi dan efisiensi sistem guna mendukung penerapannya secara lebih luas.

Kata Kunci: Aksara Jawa, Akurasi, Convolutional Neural Network, Deep Learning, Pengenalan Tulisan Tangan

1. PENDAHULUAN

Aksara Jawa adalah sistem tulisan abugida yang terdiri dari sekitar 20 hingga 33 aksara dasar, tergantung dari penggunaan bahasa yang bersangkutan. Seperti aksara Brahmi lainnya, setiap konsonan merepresentasikan satu suku kata dengan vokal inheren /a/ atau /ɔ/ yang dapat diubah dengan pemberian diakritik tertentu. Arah penulisan aksara Jawa adalah kiri ke kanan. Secara tradisional aksara ini ditulis tanpa spasi antarkata (*scriptio continua*) namun umum diselingi dengan sekelompok tanda baca yang bersifat dekoratif. [1]

Aksara Jawa hingga kini masih menjadi bagian dari pengajaran muatan lokal di DI Yogyakarta, Jawa Tengah, Jawa Timur, dan sebagian Jawa Barat [2]. Beberapa surat kabar dan majalah lokal memiliki kolom yang 2 menggunakan aksara Jawa, dan aksara Jawa dapat ditemukan pada papan nama tempat-tempat umum tertentu. Akan tetapi, banyak upaya kontemporer untuk menerapkan aksara Jawa hanya bersifat simbolik dan tidak fungsional sebagai contoh, publikasi berkala seperti majalah Kajawèn yang isi substansialnya menggunakan aksara Jawa [2]. Kebanyakan masyarakat Jawa hanya sadar akan keberadaan aksara Jawa dan mengenal beberapa huruf, tetapi jarang ada yang mampu membaca atau menulisnya secara substansial, sehingga sampai tahun 2019 tidak jarang ditemukan papan nama di tempat umum yang penulisan aksara Jawa memiliki banyak kesalahan dasar. Beberapa kendala dalam upaya revitalisasi penggunaan aksara Jawa termasuk perangkat elektronik yang sering kali mengalami kendala teknis untuk menampilkan aksara Jawa tanpa galat, sedikitnya instansi dengan kompetensi memadai yang dapat dikonsultasikan, dan kurangnya eksplorasi tipografi yang menarik bagi masyarakat. Meskipun begitu, upaya revitalisasi terus digeluti oleh sejumlah komunitas dan tokoh masyarakat yang aktif memperkenalkan kembali aksara Jawa dalam penggunaan sehari-hari, terutama dalam sarana digital [1].

Perkembangan komputer sudah semakin pesat dimana komputer dapat digunakan untuk melakukan penyelesaian permasalahan yang kompleks, pengolahan gambar, dan pengenalan pola gambar. Teknologi komputer saat ini mulai dilatih untuk dapat mengenali huruf dalam sebuah objek atau pola dalam sebuah gambar [3]. Pengenalan tulisan tangan merupakan salah satu bentuk dari pengenalan pola, berbagai penelitian mengenai pengenalan huruf tulisan tangan terus dikembangkan dengan metode pembelajaran mendalam atau biasa disebut *deep learning*. Varian dari pembelajaran mesin yang berbasis jaringan saraf tiruan adalah pengertian konsep *deep learning* dengan banyak *hidden layers* yang memiliki 3 kemampuan untuk mempelajari representasi data dengan cara otomatis [4].

Perkembangan teknologi yang semakin canggih dapat memudahkan dalam berbagai aspek. Sebagaimana telah banyak *software* yang dapat mempermudah dalam berkomunikasi dan belajar berbagai bahasa. Dengan demikian dapat dibangun sebuah *software* pembelajaran aksara Jawa yang dapat mendeteksi benar tidaknya suatu tulisan tangan. Tulisan tangan merupakan suatu hal yang unik dari setiap orang. Pengenalan tulisan tangan menggunakan *deep learning* telah diimplementasikan pada tulisan tangan latin, Cina, Arab, Persia, dan Bangla [5]. Terdapat berbagai penelitian mengenai pendeteksi tulisan tangan menggunakan machine learning [6]. *Machine learning* merupakan bidang ilmu baru yang mana bidang ini menggunakan kumpulan data dan menggunakan data tersebut untuk mempelajari pola sehingga mendapatkan pengetahuan tertentu. Ketika proses pembelajaran tentang pola-pola yang ada semakin dalam, itu akan menjadi *deep learning*. *Deep learning* adalah proses pembelajaran *Artificial Neural Network* (ANN) di mana pola dipelajari lebih dalam untuk menghasilkan hasil yang lebih baik. *Deep learning* meniru kemampuan otak manusia untuk mengingat sesuatu [7]. Beberapa peneliti 3 menyebutkan bahwa pengenalan tulisan tangan menggunakan teknik *deep learning* telah menghasilkan akurasi yang lebih tinggi dari pada menggunakan teknik *Conventional Machine Learning* [4].

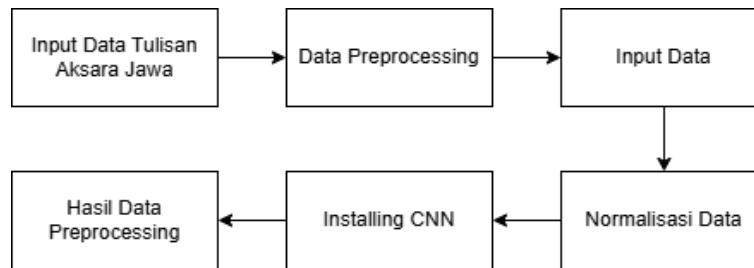
Penelitian dengan topik serupa dapat ditemukan seperti pada penelitian dengan judul “Javanese Scripted Hanacaraka Character Prediction With Resnet-18 Architecture” [8]. Tujuan dari penelitian adalah untuk mengenali tulisan dalam aksara Jawa menggunakan arsitektur Resnet-18. Hasil dari penelitian ini menunjukkan tingkat akurasi mencapai 98,01%. Selanjutnya ada penelitian Implementasi *Convolutional Neural Network* (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat [4]. Penelitian ini bertujuan untuk mengetahui apakah CNN dapat digunakan untuk klasifikasi batik tanah liat Sumatera Barat. Hasil dari penelitian ini hanya mencapai tingkat akurasi sebesar 62,5%. Serta ada penelitian dari Agus Mulyanto yang berjudul Penerapan CNN pada Pengenalan Aksara Lampung Berbasis *Optical Character Recognition* (OCR) [4]. Tujuan dari penelitian ini adalah mengidentifikasi objek tulisan tangan aksara Lampung dengan menggunakan teknologi *Optical Character Recognition* berbasis metode CNN. Hasil dari penelitian ini berdasarkan nilai training akurasi sebesar 57%.

Sehingga pada penelitian ini akan menggunakan *deep learning* sebagai teknik dalam pendeteksian tulisan tangan aksara Jawa. Penelitian ini dapat digunakan oleh seseorang yang ingin mempelajari huruf aksara Jawa secara mandiri. Keunggulan sistem dari implementasi CNN ini dapat mengenali 20 huruf aksara Jawa, mulai dari huruf ha sampai nga.

2. METODE PENELITIAN

Penelitian ini menggunakan CNN untuk mengenali tulisan tangan aksara Jawa, yang terdiri dari beberapa tahap, yaitu pengumpulan data, pra-pemrosesan data, pembangunan model CNN, dan evaluasi Model [7]. Tahap pertama adalah pengumpulan data untuk memperoleh dataset yang bervariasi. Setelah itu, pada tahap pra-pemrosesan data, dataset dibersihkan dan dipersiapkan. Pada tahap pembangunan model CNN, arsitektur jaringan syaraf tiruan dirancang dengan lapisan konvolusi dan pooling. Evaluasi model dilakukan dengan data pengujian yang fokus pada pengukuran akurasi. Penelitian ini menggabungkan metode CNN dengan dataset tulisan tangan aksara Jawa, menghasilkan sistem pengenalan karakter yang efektif. Hal ini

menyatukan aspek teknis dan linguistik untuk menciptakan sistem yang akurat [9]. Tahapan ini dapat dilihat pada gambar 1.



Gambar 1. Tahap Penelitian

Tahapan penelitian ini dapat dilihat pada Gambar 1. yang digunakan untuk mempermudah pelaksanaan penelitian.

2.1. Identifikasi Masalah

Penelitian ini dilakukan untuk mengatasi permasalahan dalam mengenali dan mengklasifikasikan tulisan tangan aksara Jawa dengan menerapkan algoritma CNN. Algoritma ini dipilih karena kemampuannya dalam menganalisis pola pada citra, sehingga diharapkan dapat mendeteksi dan mengenali aksara Jawa secara akurat [6]. Langkah ini bertujuan untuk mendukung pelestarian aksara Jawa melalui penerapan teknologi modern.

Selain itu, penelitian ini juga bertujuan mengevaluasi efektivitas algoritma CNN dalam mengenali pola tulisan tangan aksara Jawa. Evaluasi ini mencakup pengukuran tingkat akurasi algoritma, terutama dalam menghadapi variasi gaya tulisan dan kualitas citra. Hasil penelitian diharapkan dapat memberikan kontribusi pada pengembangan teknologi berbasis kecerdasan buatan untuk pelestarian budaya.

2.2. Analisis Data

Proses analisis data melibatkan beberapa tahap, termasuk pemeriksaan, pembersihan, transformasi, dan pemodelan data[8]. Tujuannya adalah untuk mengidentifikasi pola atau informasi yang berguna, yang mendukung pengambilan keputusan dan menghasilkan kesimpulan yang tepat. Melalui tahapan ini, analisis data memungkinkan pengolahan data kompleks menjadi lebih mudah dipahami dan relevan dengan konteks yang ada. Selain itu, hasil dari proses ini juga membantu dalam prediksi dan pemodelan data lebih lanjut, sehingga memungkinkan pengambilan keputusan yang strategis. Analisis data adalah proses iteratif yang memerlukan perhatian terhadap detail dan pemahaman mendalam mengenai data dan metode analisis, dengan tujuan akhir menghasilkan wawasan yang dapat digunakan dan mendukung keputusan yang lebih baik [10].

2.3. Convolutional Neural Network (CNN)

CNN adalah pengembangan dari *Multilayer Perceptron* (MLP) yang didesain untuk mengolah data dua dimensi [11]. CNN termasuk dalam jenis *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. Metode CNN merupakan suatu kelas pada *neural network* yang berspesialisasi dalam memproses data yang memiliki topologi seperti *grid*, misalnya gambar. Metode CNN ini dapat digunakan dalam pengenalan wajah, analisis dokumen, klasifikasi gambar, klasifikasi video, dan sebagainya. Hal yang menjadi ciri khusus dari CNN adalah input awal pada jaringannya secara eksplisit selalu diasumsikan sebagai citra digital, sehingga 14 parameter pada arsitektur jaringannya dapat lebih mudah disesuaikan dan lebih efisien [12].

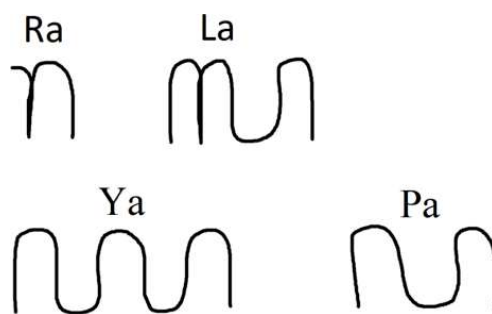
CNN merupakan teknologi yang dikombinasikan dengan jaringan syaraf buatan dan deep learning. CNN terinspirasi dari struktur sistem visual yang dikemukakan oleh Hubel dan Wiesel [13]. CNN merupakan jenis dari jaringan syaraf buatan multilapisan yang didesain khusus untuk memproses data dua dimensi [14]. CNN dapat mengurangi jumlah parameter pada jaringan untuk meningkatkan algoritma backpropagation sebagai jaringan propagasi maju. Pada algoritma CNN, area kecil yang juga disebut sebagai *area local sensing* diambil sebagai data input. Melalui propagasi maju, informasi melewati berbagai lapisan pada jaringan. Masing-masing lapisan terdiri dari kernel sehingga memperoleh beberapa fitur penting dari data yang diamati. Metode ini juga dapat memberikan bentuk bervariasi seperti *stretching* dan *rotation* [15]. Hubungan dari tingkatan lapisan konvolusi pada CNN sangat baik digunakan untuk pemrosesan gambar dan dapat meng- ekstrak karakteristik dari gambar. CNN telah diaplikasikan pada berbagai aspek, sebagai pengenalan wajah, analisis dokumen, deteksi suara dan pengenalan nomor plat kendaraan [11].

CNN merupakan model yang memiliki kemampuan dalam melakukan pelatihan pada ukuran data yang besar. Elleuch et al. menunjukkan dengan menggunakan *ReLU* dan *dropout* meningkatkan performa dari *deep neural network Dropout* dapat mengurangi *overfitting* dengan secara acak mengurangi unit koneksi saat

pelatihan [16], sehingga mencegah korelasi antara unit. Pada tahun 2015 Ioffe et al. memperkenalkan *batch normalization* yang merupakan teknik lain untuk membuat jaringan lebih kuat [13]. Untuk menghindari masalah bahwa neuron menerima masukan dari berbagai distribusi pada setiap laluan dan harus menyesuaikan bobotnya. Augmentasi data merupakan model-agnostic untuk metode pembelajaran *invariance*. Flipping dan pemotongan secara acak telah ditunjukkan pada augmentasi data untuk meningkatkan kekuatan jaringan. Metode lain yang digunakan untuk meningkatkan kekuatan jaringan CNN adalah menggunakan lapisan recurrent yang disusun oleh Liang et al [17]. Ini akan membuat jaringan lebih dalam dan membuat akurasi semakin tinggi dengan nilai parameter yang sama.

2.4. Pengumpulan Data

Pengumpulan data adalah teknik standar untuk mengumpulkan, mengukur, dan menganalisis data. Tujuan utamanya adalah untuk memperoleh informasi yang dapat diandalkan untuk keputusan bisnis yang penting [12]. Dalam penelitian ini, data yang digunakan berupa tulisan aksara Jawa yang diperoleh dari 10 responden terampil, dengan total dataset sebanyak 1.000 data. Dataset ini dibagi menjadi data pelatihan sebanyak 700 data (70%) dan data pengujian sebanyak 300 data (30%). Setiap gambar aksara Jawa diberi label dan memiliki dimensi 180 x 180 piksel dengan warna RGB. Contoh data ini dapat dilihat pada Gambar 2.



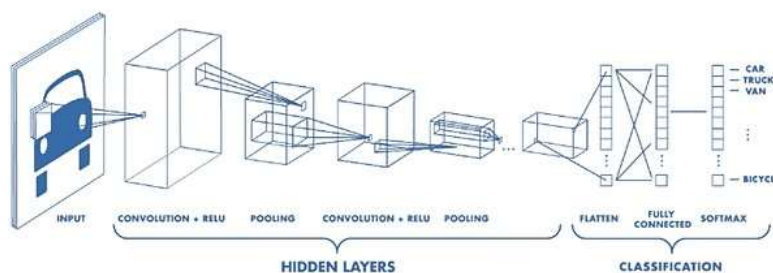
Gambar 2. Tulisan Tangan Aksara Jawa

2.5. Preprocessing Data

Preprocessing data merupakan tahap awal sebelum proses klasifikasi, yang bertujuan untuk membersihkan dan mengubah data menjadi format yang sesuai. Data yang diperoleh sering kali tidak lengkap atau mengandung masalah, sehingga perlu diproses agar siap digunakan [18]. Langkah pertama yang dilakukan adalah *image augmentation*, dengan menerapkan lapisan *Random Flip*, *Rotation*, dan *Zoom* untuk menambah variasi data. Ini membantu mengurangi *overfitting* dan meningkatkan kekuatan model. Selanjutnya, dilakukan *caching dataset* untuk mempercepat pelatihan dengan menyimpan data. Proses ini juga mengacak data, yang mendukung pembelajaran lebih cepat. *Caching dataset* membantu model untuk belajar dari variasi data, sehingga meningkatkan kualitas pelatihan. Kombinasi antara *preprocessing* dan *caching dataset* dapat meningkatkan efisiensi serta akurasi model selama pelatihan [19].

2.6. Pemodelan CNN

Pada tahap pemodelan, algoritma CNN digunakan untuk mengenali pola pada gambar tulisan tangan aksara Jawa. Proses ini dilakukan menggunakan *Google Colaboratory* dan bahasa pemrograman *Python*. Dataset dibagi menjadi 700 data pelatihan (70%) dan 300 data pengujian (30%) untuk mencegah *overfitting* dan menguji kemampuan model dalam generalisasi. Tahap awal melibatkan pendefinisian kelas dan pembuatan arsitektur model berdasarkan metode CNN, yang mencakup lapisan ekstraksi fitur konvolusional dan pooling, serta lapisan *fully connected*. Setelah arsitektur model selesai dibuat, dilakukan rangkuman model dan pelatihan dengan 150 *epoch*.



Gambar 3. Langkah-Langkah CNN

Langkah-langkah ini memungkinkan CNN untuk belajar dari data pelatihan dalam mengenali karakteristik tulisan tangan aksara Jawa. Proses dimulai dengan memasukkan input ke dalam model, kemudian dilanjutkan dengan tahap konvolusi, dilanjutkan pooling, dan diakhiri dengan tahap *flatten* sebelum proses klasifikasi. Model kemudian diuji menggunakan data pengujian untuk mengukur akurasi [20]. *Google Colab* digunakan untuk implementasi dengan pustaka *Python*, yang memastikan efisiensi dan kualitas hasil yang optimal. Dengan metode ini, diharapkan model CNN dapat mengenali dan mengklasifikasikan dengan akurat, memahami variasi gaya tulisan, serta menghasilkan hasil akurasi yang dapat diandalkan untuk aplikasi pengenalan tulisan tangan aksara Jawa [15].

2.7. Evaluasi Model

Pada proses klasifikasi, matriks data dianalisis menggunakan empat metrik, yaitu Benar Positif (BP), Benar Negatif (BN), Salah Positif (SP), dan Salah Negatif (SN), yang mencerminkan hasil klasifikasi oleh *classifier* [11]. Evaluasi kinerja *classifier* ditunjukkan dengan akurasi. Metrik BP mengindikasikan jumlah data positif yang berhasil diklasifikasikan dengan benar, sedangkan BN menunjukkan data negatif yang diklasifikasikan dengan benar. SP merujuk pada data negatif yang salah diklasifikasikan sebagai positif, dan SN menunjukkan data positif yang salah diklasifikasikan sebagai negatif. Dalam penelitian ini, evaluasi dilakukan dengan menggunakan parameter akurasi, yang mengukur sejauh mana klasifikasi dilakukan dengan benar secara keseluruhan. Matriks evaluasi ini mencerminkan seberapa tepat prediksi model terhadap kelas yang ada. Meskipun ini adalah metode umum untuk mengukur kinerja model, namun dapat terbatas, terutama pada data yang tidak seimbang, yang bisa menyebabkan bias dalam efektivitas model [17].

3. HASIL DAN PEMBAHASAN

Proses analisis CNN dalam penelitian ini dilaksanakan secara bertahap, yang mencakup tahap pengumpulan data, pra-pemrosesan data, pemodelan CNN, dan evaluasi.

3.1 Dataset

Langkah pertama dalam pengembangan sistem pengenalan aksara Jawa adalah pengumpulan data. Data yang dikumpulkan berupa citra (gambar) tulisan tangan dari responden yang menggunakan aksara Jawa. Proses ini melibatkan pengambilan gambar atau pemindaian tulisan tangan menggunakan perangkat yang sesuai untuk memastikan kualitas gambar yang optimal. Citra-citra tersebut kemudian akan digunakan sebagai input dalam sistem pengenalan aksara. Mengumpulkan data yang representatif dan berkualitas tinggi merupakan langkah awal yang krusial untuk memastikan keberhasilan sistem secara keseluruhan, pada Gambar 4. Menampilkan dataset aksara Jawa.



Gambar 4. Dataset Aksara Jawa

Dalam proyek pengenalan aksara Jawa menggunakan CNN, data dikumpulkan dari 10 responden yang menggunakan aplikasi paint di komputer untuk membuat 1000 gambar tulisan tangan aksara Jawa. Dataset ini mencakup 20 huruf aksara Jawa yang berbeda, dengan setiap huruf terwakili secara proporsional. Untuk memastikan model CNN dapat belajar dengan efektif dan menghasilkan prediksi yang akurat, dataset dibagi menjadi dua subset: data latih dan data validasi. Data latih terdiri dari 35 citra per huruf dalam 20 folder, sehingga totalnya ada 700 citra. Sedangkan data validasi terdiri dari 15 citra per huruf dalam 20 folder, dengan total 300 citra. Kumpulan dataset Aksara Jawa dapat ditunjukkan pada gambar 5.

Dataset yang telah dikumpulkan di *Google Drive* dimasukkan ke dalam *Google Colab* menggunakan kode *Python* agar dapat diproses dalam program penelitian ini. Berikut adalah ilustrasi dari kode *Python* yang digunakan untuk mengimpor dataset ke *Google Colab*.

Tahap kedua dalam penelitian ini adalah pra-pemrosesan data, yang melibatkan serangkaian langkah untuk memastikan data siap digunakan dalam pelatihan model. Pertama, data yang telah dikumpulkan diimpor ke dalam sistem menggunakan *Python* dengan pustaka *TensorFlow/Keras*, yang dirancang untuk memproses

dataset berbasis gambar. Dataset dihubungkan dari *Google Drive* ke *Google Colab* agar dapat diakses, dan label kelas didefinisikan sesuai dengan yang terdapat dalam dataset. Langkah selanjutnya adalah normalisasi data, yaitu dengan mengubah nilai piksel citra dari rentang 0-255 menjadi 0-1. Tujuan normalisasi ini adalah untuk meningkatkan kinerja model dengan menjaga konsistensi skala data, sehingga fitur yang diekstraksi lebih mudah dikelola selama pelatihan. Selain itu, pra-pemrosesan mencakup transformasi citra menggunakan teknik augmentasi gambar, seperti *Random Flip*, *Random Rotation*, dan *Random Zoom*. Teknik ini diterapkan untuk meningkatkan variasi data dan mencegah *overfitting*. Setelah augmentasi, citra di-*rescale* menjadi ukuran 64x64x1. Ukuran yang lebih kecil ini membantu mempercepat proses pelatihan tanpa mengurangi kualitas data secara signifikan [21].



Gambar 5. Dataset Aksara Jawa

Terakhir, dilakukan *caching dataset*, yaitu menyimpan data yang telah diproses ke dalam sistem. Proses ini memungkinkan data diacak secara otomatis setiap kali dijalankan ulang, sehingga struktur data selalu bervariasi, yang dapat meningkatkan kualitas pelatihan model. Tabel 1. Menampilkan tahapan *pre-processing* data yang dilakukan.

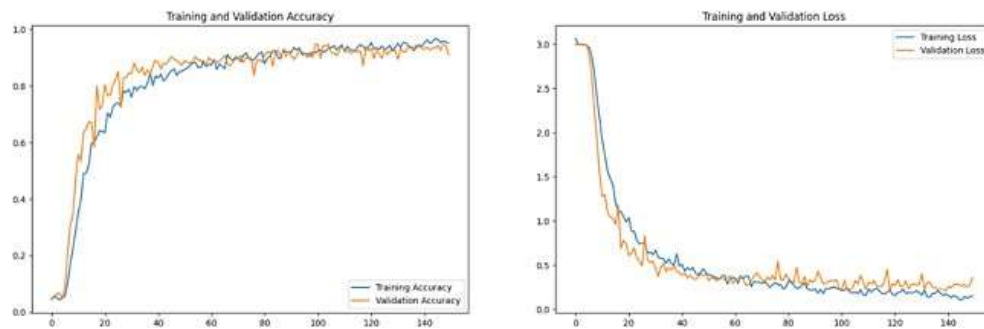
Tabel 1. Tahap *pre-processing* data

Layer (type)	Output Shape	Param #
random_flip (RandomFlip)	(None, 180, 180, 3)	0
random_rotation (RandomRotation)	(None, 180, 180, 3)	0
random_zoom (RandomZoom)	(None, 180, 180, 3)	0
rescaling (Rescaling)	(None, 180, 180, 3)	0
conv2d (Conv2D)	(None, 180, 180, 32)	896
max_pooling2d (MaxPooling2D)	(None, 90, 90, 32)	0
conv2d_1 (Conv2D)	(None, 90, 90, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 45, 45, 64)	0
conv2d_2 (Conv2D)	(None, 45, 45, 64)	36,928
max_pooling2d_2 (MaxPooling2D)	(None, 22, 22, 64)	0
dropout (Dropout)	(None, 22, 22, 64)	0
flatten (Flatten)	(None, 30976)	0
dense (Dense)	(None, 128)	3,965,056
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 128)	16,512
dropout_2 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 64)	8,256
dropout_3 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 20)	1,300

Tahap ketiga adalah pemodelan dan pengujian menggunakan CNN. CNN merupakan jenis jaringan saraf yang efektif untuk pengenalan citra, terdiri dari beberapa lapisan konvolusional yang mengekstrak fitur dari citra [8]. Lapisan konvolusional ini berfungsi untuk mengekstrak fitur dari citra dengan menerapkan filter (kernel) pada citra input untuk mendeteksi fitur seperti tepi, tekstur, dan pola yang lebih kompleks seiring bertambahnya kedalaman lapisan. Lapisan pooling, atau dikenal juga sebagai lapisan sub-sampling, berfungsi untuk mengurangi dimensi fitur yang diekstrak guna mengurangi jumlah parameter dan komputasi dalam jaringan. Ini juga membantu dalam menciptakan representasi fitur yang lebih tahan terhadap pergeseran dan

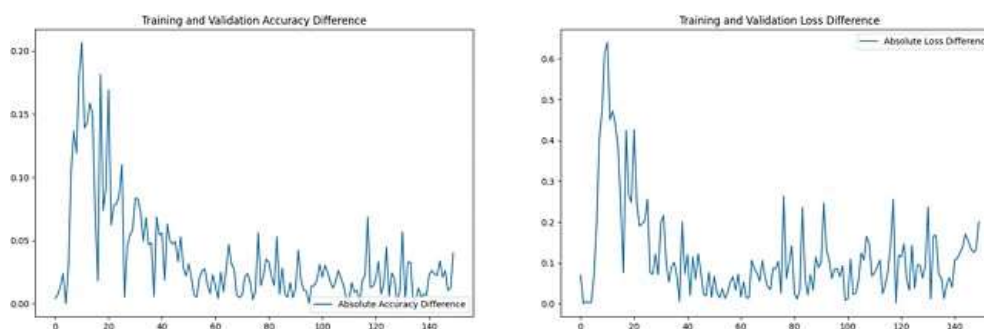
distorsi. Terakhir, lapisan flatten meratakan output menjadi vektor 1D yang kemudian dimasukkan ke dalam lapisan *fully connected* atau *dense layer* dalam jaringan saraf [22].

Proses pemodelan dengan algoritma CNN dalam penelitian ini dilakukan menggunakan *Google Colab* dan bahasa pemrograman *Python*. Sebelum memulai, dataset dibagi menjadi dua bagian: data pelatihan dan data pengujian dengan komposisi 80:20. Sebanyak 80% data digunakan untuk melatih model, sementara 20% sisanya digunakan untuk menguji kinerja model dan menghasilkan prediksi. Langkah pertama dalam pemodelan adalah mendefinisikan kelas yang akan digunakan. Selanjutnya, dibuat arsitektur model berdasarkan metode CNN, yang terdiri dari dua bagian utama: *feature extraction layer* dan *fully connected layer*. *Feature extraction layer* berfungsi untuk mengekstraksi fitur dari citra input, melibatkan proses pada lapisan konvolusional dan lapisan pooling. Sementara itu, *fully connected layer* bertugas untuk mengklasifikasikan hasil ekstraksi fitur ke dalam kelas tertentu, melalui tahapan *flatten layer* dan *dense layer*. Arsitektur ini dapat dimodifikasi sesuai dengan kebutuhan penelitian. Setelah arsitektur model selesai, langkah berikutnya adalah membuat summary model untuk memastikan bahwa model siap untuk melatih data pelatihan dan mengenali pola sesuai label kelas. Proses pelatihan dilakukan dengan 150 *epoch*, dan hasil performa model disimpan dalam variabel history. Dengan tahapan ini, diharapkan model mengenali data secara optimal [23].



Gambar 5. Visualisasi Performa CNN

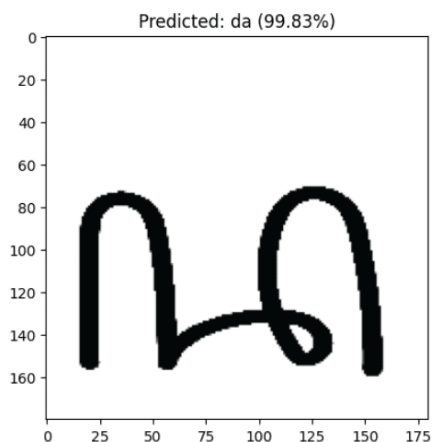
Hasil visualisasi performa CNN yang ditunjukkan pada Gambar 5. menggambarkan akurasi dan *loss* pada data latih dan validasi selama 150 *epoch* pelatihan. Grafik akurasi menunjukkan peningkatan kinerja model seiring berjalannya epoch, di mana akurasi pada data latih dan validasi stabil pada nilai tinggi antara epoch ke-100 hingga ke-150, yang menandakan bahwa model berhasil mengenali pola aksara Jawa dengan baik. Sementara itu, grafik *loss* menunjukkan penurunan nilai *loss* secara konsisten pada kedua data latih dan validasi, yang menunjukkan bahwa model berhasil meminimalkan kesalahan prediksi. Konsistensi tren antara data latih dan validasi juga mengindikasikan bahwa model tidak mengalami *overfitting* dan memiliki kemampuan generalisasi yang baik.



Gambar 6. Visualisasi Perbandingan Performa CNN

Visualisasi performa CNN yang ditampilkan pada Gambar 6. memperlihatkan perbedaan absolut antara akurasi dan *loss* pada data latih dan validasi. Grafik di sebelah kiri menunjukkan perbedaan akurasi yang awalnya tinggi, namun menurun seiring bertambahnya epoch, meskipun masih terdapat fluktuasi. Grafik di sebelah kanan memperlihatkan perbedaan *loss* yang juga mengalami penurunan, namun tetap fluktuatif. Tren peningkatan perbedaan ini menunjukkan bahwa model telah cukup baik dan tidak mengalami *overfitting*, di mana model mampu belajar dengan spesifik pada data latih dan juga menggeneralisasi dengan baik pada data validasi. Hal ini menunjukkan perlunya penyesuaian teknik regularisasi atau pengaturan hyperparameter untuk meningkatkan kemampuan generalisasi model.

Tahap terakhir adalah evaluasi model dengan menggunakan akurasi sebagai metrik untuk menilai sejauh mana model dapat mengenali dan mengklasifikasikan citra tulisan tangan aksara Jawa. Evaluasi ini bertujuan untuk memastikan bahwa model menghasilkan hasil yang akurat dan relevan berdasarkan data input yang diberikan [10].



Gambar 7. Contoh Hasil Prediksi

Gambar 7. memperlihatkan hasil prediksi dari model CNN dalam mengenali huruf aksara Jawa "ya". Model memprediksi huruf tersebut dengan akurasi mencapai 99,83%, yang menunjukkan bahwa model sangat yakin bahwa gambar input adalah huruf "ya". Untuk menilai akurasi dan kinerja pengenalan aksara Jawa, model dilatih menggunakan dataset yang berisi gambar-gambar huruf aksara Jawa. Hasil prediksi kemudian dibandingkan dengan label asli. Tingginya akurasi ini menunjukkan bahwa model CNN telah dilatih dengan baik dan mampu mengenali karakter aksara Jawa dengan kesalahan yang minim, sehingga memberikan keyakinan tinggi terhadap kemampuannya dalam aplikasi dunia nyata.

4. KESIMPULAN

Penelitian ini menghasilkan model yang dapat mengenali tulisan tangan aksara Jawa dengan baik, serta menganalisis kinerja pengenalan tulisan tangan berdasarkan variasi jumlah iterasi menggunakan algoritma CNN. Hasil pengujian menunjukkan bahwa performa jaringan saraf dipengaruhi oleh jumlah *epoch*, *optimizer* yang digunakan, dan ukuran *batch*. Dalam penelitian ini, digunakan 150 *epoch*, *optimizer* Adam, dan ukuran *batch* 32. Akurasi model meningkat dari iterasi pertama hingga iterasi ke-20, namun dari iterasi ke-100 hingga ke-150, akurasi stabil tanpa perubahan yang signifikan. Pada rentang iterasi 100-150, akurasi tetap stabil dengan hasil evaluasi menunjukkan tingkat keberhasilan pengenalan sebesar 99,83%. Untuk penelitian selanjutnya, disarankan untuk mencoba variasi nilai *epoch*, *optimizer*, dan ukuran *batch* yang berbeda, serta mempertimbangkan algoritma pengenalan gambar selain CNN. Selain itu, model berbasis CNN yang lebih canggih, seperti *transfer learning* atau *recurrent neural network*, bisa digunakan untuk mencapai akurasi yang lebih optimal.

REFERENSI

- [1] S. D. Hartiyani, A. Prayogo, and E. Erlinawati, "Aplikasi Multimedia Pembelajaran Aksara Jawa Berbasis Android Untuk Siswa Sekolah Dasar," *EDUSAINTEK J. Pendidikan, Sains dan Teknol.*, vol. 10, no. 2, pp. 679–693, 2023, doi: 10.47668/edusaintek.v10i2.805.
- [2] D. Fakhruddin, A. Sachari, and N. Haswanto, "Pengembangan Desain Informasi dan Pembelajaran Aksara Jawa melalui Media Website," *ANDHARUPA J. Desain Komun. Vis. Multimed.*, vol. 5, no. 01, pp. 1–23, 2019, doi: 10.33633/andharupa.v5i01.1990.
- [3] M. Muslih and E. H. Rachmawanto, "Convolutional Neural Network (Cnn) Untuk Klasifikasi Citra Penyakit Diabetes Retinopathy," *SKANIKA*, vol. 5, no. 2, pp. 167–176, 2022, doi: 10.36080/skanika.v5i2.2945.
- [4] K. Azmi, S. Defit, and S. Sumijan, "Implementasi Convolutional Neural Network (CNN) Untuk Klasifikasi Batik Tanah Liat Sumatera Barat," *J. Unitek*, vol. 16, no. 1, pp. 28–40, 2023, doi: 10.52072/unitek.v16i1.504.
- [5] R. Kusumaningrum, I. Z. Nisa, R. P. Nawangsari, and A. Wibowo, "Sentiment analysis of Indonesian hotel reviews: from classical machine learning to deep learning," *Int. J. Adv. Intell. Informatics*, vol. 7, no. 3, pp. 292–303, 2021, doi: 10.26555/ijain.v7i3.737.

- [6] A. Mulyanto, E. Susanti, F. Rossi, W. Wajiran, and R. I. Borman, "Penerapan Convolutional Neural Network (CNN) pada Pengenalan Aksara Lampung Berbasis Optical Character Recognition (OCR)," *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 1, p. 52, 2021, doi: 10.26418/jp.v7i1.44133.
- [7] R. Aryanto, M. Alfian Rosid, and S. Busono, "Penerapan Deep Learning untuk Pengenalan Tulisan Tangan Bahasa Aksara Lota Ende dengan Menggunakan Metode Convolutional Neural Networks," *J. Inf. dan Teknol.*, vol. 5, no. 1, pp. 258–264, 2023, doi: 10.37034/jidt.v5i1.313.
- [8] E. D. B. Sudewo, M. K. Biddinika, and A. Fadlil, "Javanese Script Hanacaraka Character Prediction With Resnet-18 Architecture," *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. 10, no. 2, pp. 363–370, 2024, doi: 10.33330/jurteks.v10i2.3017.
- [9] M. A. S. M. Afendi and M. Yusoff, "A sound event detection based on hybrid convolution neural network and random forest," *IAES Int. J. Artif. Intell.*, vol. 11, no. 1, pp. 121–128, 2022, doi: 10.11591/ijai.v11i1.pp121-128.
- [10] A. Kirana, H. Hikmayanti, and J. Indra, "Pengenalan Pola Aksara Sunda dengan Metode Convolutional Neural Network," *Sci. Student J. Information, Technol. Sci.*, vol. 1, no. 2, pp. 95–100, 2020, [Online]. Available: <http://journal.ubpkarawang.ac.id/mahasiswa/index.php/ssj/article/download/19/15>
- [11] D. S. Wita and D. Y. Liliana, "Klasifikasi Identitas Dengan Citra Telapak Tangan Menggunakan Convolutional Neural Network (CNN)," *J. Rekayasa Teknol. Inf.*, vol. 6, no. 1, p. 1, 2022, doi: 10.30872/jurti.v6i1.7100.
- [12] E. H. Rachmawanto and P. N. Andono, "Deteksi Karakter Hiragana Menggunakan Metode Convolutional Neural Network," *J. Nas. Pendidik. Tek. Inform.*, vol. 11, no. 3, pp. 183–191, 2022, doi: 10.23887/janapati.v11i3.50144.
- [13] W. Setiawan, "Perbandingan Arsitektur Convolutional Neural Network Untuk Klasifikasi Fundus," *J. Simantec*, vol. 7, no. 2, pp. 48–53, 2020, doi: 10.21107/simantec.v7i2.6551.
- [14] I. Gusti, A. Gede, and A. Kadyanan, "Pengembangan Aplikasi Deep Learning untuk Identifikasi Kain Endek Bali," *J. Ilmu Komput.*, vol. 14, no. 1, pp. 32–39, 2022.
- [15] I. Akil and I. Chaidir, "Deteksi Karakter Huruf Arab Dengan Menggunakan Convolutional Neural Network," *INTI Nusa Mandiri*, vol. 15, no. 2, pp. 183–188, 2021, doi: 10.33480/inti.v15i2.2179.
- [16] R. N. S. Amriza and D. Supriyadi, "Komparasi Metode Machine Learning dan Deep Learning untuk Deteksi Emosi pada Text di Sosial Media," *J. Penelit. Ilmu dan Tek. Komput.*, vol. 13, no. 2, pp. 130–139, 2021.
- [17] I. Arifin, R. F. Haidi, and M. Dzalhaqi, "Penerapan Computer Vision Menggunakan Metode Deep Learning pada Perspektif Generasi Ulul Albab," *J. Teknol. Terpadu*, vol. 7, no. 2, pp. 98–107, 2021, doi: 10.54914/jtt.v7i2.436.
- [18] Efanntyo and A. R. Mitra, "Perancangan Aplikasi Sistem Pengenalan Wajah Dengan Metode Convolutional Neural Network (CNN) Untuk Pencatatan Kehadiran Karyawan," *J. Instrumentasi dan Teknol. Inform.*, vol. 3, no. 1, pp. 1–11, 2021.
- [19] N. P. Wulandari and D. Fitriana, "Analisa Perbandingan Algoritma CNN Dan MLP Dalam Mendeteksi Penyakit COVID-19 Pada Citra X-Ray Paru," *J. Sains, Apl. Komputasi dan Teknol. Inf.*, vol. 3, no. 2, p. 44, 2021.
- [20] D. J. P. Manajang, S. R. U. . Sompie, and A. Jacobus, "Implementasi Framework Tensorflow Object Detection Dalam Mengklasifikasi Jenis Kendaraan Bermotor," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 4, pp. 1821–1831, 2021, doi: 10.35957/jatisi.v8i4.1269.
- [21] M. Ihsan, R. K. Niswatin, and D. Swanjaya, "Deteksi Ekspresi Wajah Menggunakan Tensorflow," *Joutica*, vol. 6, no. 1, p. 428, 2021, doi: 10.30736/jti.v6i1.554.
- [22] N. F. Hasan, K. Kusriani, and H. Al Fatta, "Analisis Arsitektur Jaringan Syaraf Tiruan Untuk Peramalan Penjualan Air Minum Dalam Kemasan," *J. Rekayasa Teknol. Inf.*, vol. 3, no. 1, p. 1, 2019, doi: 10.30872/jurti.v3i1.2290.
- [23] S. Prihatiningsih, N. Shafiy M, F. Andriani, and N. Nugraha, "Analisa Performa Pengenalan Tulisan Tangan Angka Berdasarkan Jumlah Iterasi Menggunakan Metode Convolutional Neural Network," *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 1, pp. 58–66, 2019, doi: 10.35760/tr.2019.v24i1.1934.