

APLIKASI KLASIFIKASI HURUF JAWA MENGGUNAKAN ALGORITMA *CONVOLUTIONAL NEURAL NETWORK* DAN *RANDOM FOREST*

Novan Windi Eko Purwanto

Program Studi Teknik Informatika, Universitas PGRI Madiun

email: novan_2005101020@mhs.unipma.ac.id

Abstract: *This study aims to develop a Javanese letter classification application using the Convolutional Neural Network (CNN) and Random Forest (RF) algorithms. This system is built using the Python programming language, HTML, and CSS, as well as the Flask framework, and is developed using the Agile Development method that allows rapid adjustment to changing needs. This application is designed to identify and classify Javanese letters, with a focus on model accuracy and efficiency. The CNN algorithm is applied to process Javanese letter images and produces an accuracy of 99.4%, while the RF model, which is used for regression analysis, achieves an accuracy of 42.2%. The evaluation results show that CNN is significantly more effective in classification than RF. This study concludes that the use of CNN in a web-based Javanese letter classification application with the Agile Development methodology provides very good results and can help users recognize and understand Javanese letters more accurately.*

Keywords: *Algorithm, CNN, Random Forest, Classification, Javanese Script*

Abstrak: Penelitian ini bertujuan untuk mengembangkan aplikasi klasifikasi huruf Jawa menggunakan algoritma Convolutional Neural Network (CNN) dan Random Forest (RF). Sistem ini dibangun dengan menggunakan bahasa pemrograman Python, HTML, dan CSS, serta framework Flask, dan dikembangkan dengan metode Agile Development yang memungkinkan penyesuaian cepat terhadap perubahan kebutuhan. Aplikasi ini dirancang untuk mengidentifikasi dan mengklasifikasikan huruf aksara Jawa, dengan fokus pada akurasi dan efisiensi model. Algoritma CNN diterapkan untuk memproses gambar huruf aksara Jawa dan menghasilkan akurasi 99,4%, sementara model RF, yang digunakan untuk analisis regresi, mencapai akurasi 42,2%. Hasil evaluasi menunjukkan bahwa CNN secara signifikan lebih efektif dalam klasifikasi dibandingkan RF. Penelitian ini menyimpulkan bahwa penggunaan CNN dalam aplikasi klasifikasi huruf Jawa berbasis web dengan metodologi Agile Development memberikan hasil yang sangat baik dan dapat membantu pengguna dalam mengenali dan memahami huruf aksara Jawa dengan lebih akurat.

Kata kunci: *Algoritma, CNN, Random Forest, Klasifikasi, Huruf Aksara Jawa*

Pendahuluan

. Di era modern ini, teknologi memainkan peran penting dalam memfasilitasi pekerjaan manusia di berbagai bidang. Saat ini, teknologi telah menjadi kebutuhan mendasar bagi manusia karena kemampuannya untuk membantu mengatur prioritas pekerjaan. Salah satu teknologi tersebut adalah Kecerdasan Buatan (AI). AI merupakan cabang ilmu komputer yang bertujuan menyelesaikan masalah-masalah yang biasanya berkaitan dengan kecerdasan manusia, seperti pembelajaran, penciptaan, dan pengenalan gambar. Dengan pemahaman yang baik tentang AI dalam hal klasifikasi huruf aksara Jawa, seseorang dapat dengan mudah membaca dan menulis aksara Jawa, yang sangat penting bagi suku Jawa dan bahasa yang digunakan di Indonesia, khususnya di pulau Jawa.

Namun, dalam praktiknya, pengenalan huruf aksara Jawa bisa menjadi sulit karena banyaknya variasi bentuk dan pola, serta perbedaan dalam pengucapannya. Saat ini, di Sekolah Dasar, siswa sering kesulitan membedakan dan menulis huruf aksara Jawa, terutama bagi mereka yang baru mulai belajar. Oleh karena itu, diperlukan metode yang dapat mempermudah proses belajar aksara Jawa bagi siswa, seperti metode Convolutional Neural Network dan Random Forest. Dengan menggunakan metode tersebut, pengenalan huruf aksara Jawa dapat dilakukan dengan lebih mudah, serta mempermudah identifikasi objek pada gambar.

Untuk mengatasi tantangan di sekolah dasar, penulis merekomendasikan penggunaan sistem klasifikasi berbasis web. Studi ini akan mengevaluasi akurasi kedua algoritma, Convolutional Neural Network (CNN) dan Random Forest (RF), dalam mengklasifikasikan huruf aksara Jawa dengan

menggunakan dataset huruf aksara Jawa yang tersedia. Tujuannya adalah untuk mengembangkan sistem yang dapat membantu siswa sekolah dasar dalam mempelajari huruf aksara Jawa. Hasil dari penelitian ini dapat bermanfaat untuk membuat aplikasi yang mendukung proses belajar huruf aksara Jawa bagi siswa sekolah dasar.

Peneliti menemukan jawaban untuk masalah dengan menggunakan berbagai penelitian sebelumnya. Referensi-referensi tersebut berfungsi sebagai dasar untuk penulisan penelitian yang sedang dilakukan. Dalam penelitian pertama yang mengklasifikasikan gambar cuaca menggunakan CNN, SVM, dan KNN, ditemukan bahwa CNN menerima skor F1 tertinggi dengan akurasi 0,942, presisi 0,943, dan recall 0,942, masing-masing. Namun, CNN membutuhkan waktu eksekusi terpanjang, yaitu sekitar 458,49 detik, untuk mencapai kinerja terbaiknya (Farid Naufal, 2021).

Dalam penelitian kedua yang menggunakan metode CNN dan SVM untuk mengklasifikasikan buah menurut bentuknya, ditemukan bahwa CNN memiliki akurasi sebesar 96,87%, lebih tinggi dari SVM sebesar 93,09% (Kurniadi et al., 2021). Studi ketiga tentang analisis sentimen yang menggunakan teknik Random Forest dilakukan oleh (Adrian et al., 2021) dengan judul "Perbandingan Metode Klasifikasi Hutan Random dan SVM Pada Analisis Sentimen PSBB", menyatakan bahwa kemampuan model untuk memprediksi sentimen terkait PSBB dapat dikurangi oleh keragaman bahasa di Twitter; SVM dianggap lebih baik dalam mengidentifikasi tweet yang dikategorikan sebagai "Positif".

Pada penelitian yang dilakukan oleh (Nalatissifa et al., 2021) dengan judul "Perbandingan Kinerja Algoritma Klasifikasi *Naive Bayes*, *Support Vector Machine* (SVM), dan *Random forest* untuk Prediksi Ketidakhadiran di Tempat Kerja" menyatakan bahwa ketiga algoritma tersebut dapat memberikan akurasi, presisi, dan recall di atas 96%. Selain itu, dibandingkan dengan algoritma *Naive Bayes* dan SVM, algoritma *Random forest* memiliki tingkat akurasi, presisi, dan recall tertinggi, dengan nilai akurasi 99,38%, presisi 99,42%, dan recall 99,39%. Ini menunjukkan bahwa algoritma *Random forest* lebih cocok untuk memprediksi ketidakhadiran di tempat kerja. Dijelaskan bahwa, berdasarkan referensi penelitian sebelumnya, belum ada penelitian yang menggunakan algoritma *Convolutional Neural Network* dan *Random Forest* untuk mengklasifikasikan objek huruf aksara Jawa. Dengan mempertimbangkan pendahuluan yang diuraikan dalam penelitian ini, tujuan penelitian pertama adalah untuk mengetahui cara menggunakan algoritma *Random Forest* dan *Convolutional Neural Network* untuk mengklasifikasikan huruf aksara Jawa. Selanjutnya, tujuan kedua adalah untuk mengetahui bagaimana algoritma *Random Forest* dan *Convolutional Neural*. Salah satu aksara yang digunakan di Jawa dan sekitarnya adalah aksara hanacaraka. Aksara hanacaraka berasal dari lima aksara pertama dalam aksara Jawa, yaitu "hana caraka". Aksara Jawa terdiri dari dua puluh aksara, yaitu ha-na-ca-ra-ka-da-ta-sa-wa-la-pa-dha-ja-ya-nya-ma-ga-ba-tha-nga (Hidayat et al., 2020).

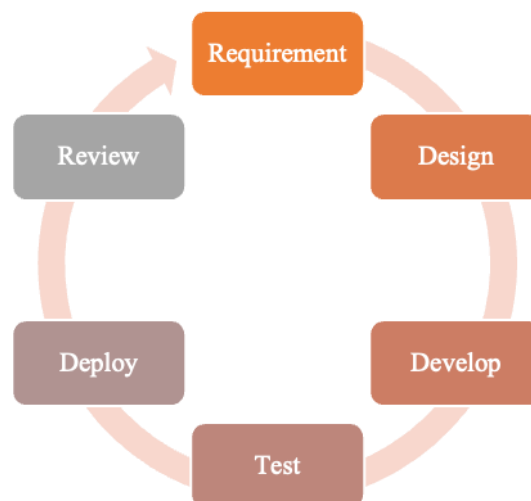
Terdapat beberapa studi keilmuan sebagai penunjang penelitian CNN dan *Random Forest* di antaranya mengenai klasifikasi adalah proses mengelompokkan data baru ke dalam kategori yang sudah ada sebelumnya dalam data mining untuk memprediksi kategori data yang belum diketahui (Hakim et al., 2022). *Machine learning* adalah disiplin ilmu dalam kecerdasan buatan yang menggunakan bahasa pemrograman untuk membuat komputer berperilaku cerdas seperti manusia (R.H. Zer et al., 2022). CNN atau *Convolutional Neural Network* adalah jenis jaringan saraf tiruan *feedforward* dengan struktur yang mendalam. CNN merupakan salah satu contoh algoritma *deep learning*. Keunggulannya terletak pada kemampuannya untuk mengidentifikasi informasi tersembunyi dari berbagai objek seperti gambar, suara, teks, dan lainnya, meskipun objek tersebut berada di posisi manapun dalam *input* (Arif Faizin et al., 2022). Sedangkan menurut Qotrunnada & Utomo, (2022) *Convolutional Neural Network* (CNN) adalah salah satu algoritma dalam *deep learning*. CNN digunakan untuk mengklasifikasikan gambar atau video, serta mendeteksi objek yang ada dalam gambar atau wilayah tertentu di dalam gambar. *Random Forest* jaringan saraf tiruan *convolutional* adalah metode yang digunakan dalam pengenalan dan pemrosesan gambar.. (Kholik, 2021). *Variable importance* (pentingnya variabel) merujuk pada ukuran yang digunakan untuk mengevaluasi seberapa signifikan suatu variabel dalam memprediksi variabel target dalam suatu model. (Lee et al., 2020). Flask adalah kerangka kerja aplikasi web berbasis Python dengan dependensi Tools dan Jinja2. (Arbain et al., 2022). *Python* adalah bahasa pemrograman yang dinamis dan berorientasi objek yang dapat digunakan untuk berbagai jenis pengembangan perangkat lunak. (Nugroho et al., 2020). *Scikit-learn* adalah sebuah modul *Python* yang mengintegrasikan

berbagai macam algoritma *machine learning* terkini untuk masalah *medium-scale supervised* dan *unsupervised*. (Baranwal et al., 2019). *Scikit-learn* juga menyediakan berbagai dataset yang siap digunakan, sehingga pengguna dapat fokus pada pengembangan algoritma daripada menghabiskan waktu untuk mengumpulkan dan membersihkan data. (Hackeling, 2014). *TensorFlow* adalah sebuah perangkat lunak kerangka kerja komputasi yang dikembangkan dan didukung secara *open source* oleh *Google*. (Ju et al., 2019). Sedangkan menurut Hikmatia A.E & Ihsan Zul (2021) *TensorFlow* adalah platform komputasi untuk membangun model pembelajaran mesin yang menyediakan berbagai *toolkit* untuk membuat model pada berbagai tingkat abstraksi. UML adalah bahasa visual yang digunakan untuk memodelkan sistem dan berkomunikasi dengannya melalui diagram dan teks pendukung. Beberapa jenis diagram yang termasuk dalam pemodelan UML termasuk diagram urutan, diagram use case, diagram kelas, dan diagram aktivitas. Syarif & Nugraha (2020). Pada tahap akhir penelitian dilakukan sebuah Pengujian fungsional menggunakan *black box testing* adalah metode pengujian perangkat lunak yang digunakan untuk menguji fungsionalitas perangkat lunak tanpa memperhatikan detail struktur internal kode atau program yang digunakan (Wicaksono, 2021).

Metode

Metode Pengembangan Sistem

Pendekatan pengembangan sistem agile digunakan dalam studi perancangan sistem. Pendekatan ini dimaksudkan untuk membuat pengembangan sistem lebih mudah untuk ditinjau dan diubah dibandingkan dengan pendekatan desain terstruktur. (Puspitasari & Anardani, 2023). Pada gambar 1 menunjukkan tahapan penelitian Agile Development untuk mengembangkan aplikasi klasifikasi huruf aksara Jawa.



Gambar 1. Konsep Agile Development (Hutauruk & Pakpahan, 2021)

Dalam tulisan ini, konsep Agile Development dijelaskan sebagai berikut.

1. Persyaratan: Tahapan ini bertujuan untuk menentukan bagaimana informasi kebutuhan pengguna terhadap sebuah sistem dibentuk.
2. Desain: Pada tahap ini, data yang telah diperoleh akan digunakan untuk merancang desain yang akan memberikan gambaran mendalam tentang apa yang harus dilakukan.
3. Perbaikan: Pada tahap ini, penulis menerjemahkan rancangan dan analisis sistem ke dalam bahasa pemrograman yang dapat difahami komputer dengan menggunakan kode pemrograman.
4. Pengujian: Setelah pengembangan sistem berhasil, langkah selanjutnya adalah melakukan verifikasi sistem melalui pemeriksaan dan pengujian untuk menemukan kesalahan.
5. Tahap Deployment: Tahap ini bertujuan untuk menyebarkan aplikasi pengembang.
6. Memeriksa tahapan akhir dari gagasan pengembangan cepat, yaitu tahapan.

Metode Klasifikasi

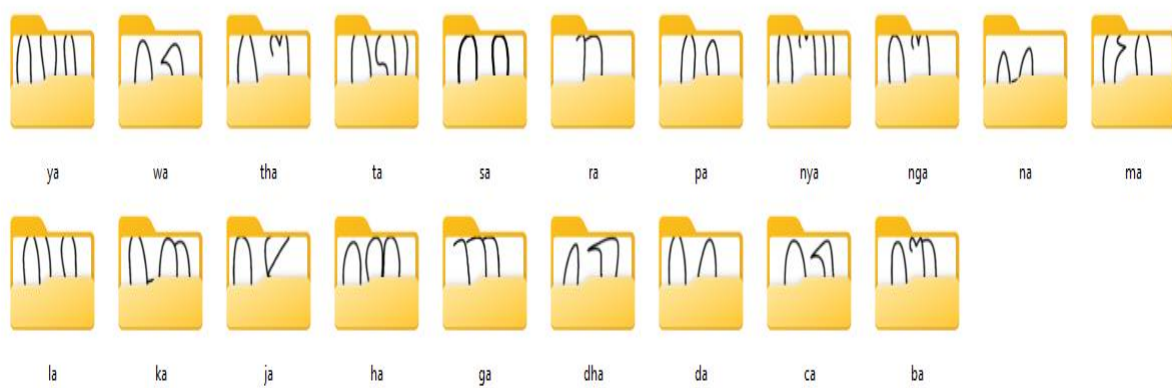
Dalam penelitian ini, dua pendekatan digunakan: Random Forest dan Convolutional Neural Network (CNN). CNN menangani data dalam bentuk grid, mirip dengan gambar. Jaringan ini bekerja dengan memecah gambar menjadi bagian-bagian kecil dan menganalisisnya untuk

menemukan pola pengenalan dan klasifikasi yang berguna. Metode ini memungkinkan CNN untuk mengenali dan mengklasifikasikan objek dalam gambar, memberikan dasar bagi banyak aplikasi kontemporer dalam pengenalan wajah, deteksi objek, dan analisis gambar.

Namun, Hutan acak digunakan untuk regresi dan klasifikasi dengan menggabungkan hasil dari berbagai pohon keputusan. Metode ini meningkatkan keakuratan prediksi dan mengurangi risiko overfitting, yang terjadi pada pohon keputusan tunggal. Untuk memulai proses pembentukan, beberapa subset data dibuat dari dataset awal menggunakan teknik sampling bootstrap, di mana data dipilih secara acak dengan penggantian. Dalam satu subset, beberapa poin data mungkin muncul beberapa kali, sementara yang lain mungkin tidak muncul sama sekali. Satu pohon keputusan dilatih dengan menggunakan setiap subset ini.

Datasets

Platform Kaggle yang dapat diakses secara terbuka adalah sumber data yang digunakan dalam penelitian ini. Dataset terdiri dari gambar dengan 20 label, masing-masing dengan 420 gambar, sehingga total 8.400 gambar digunakan dalam penelitian ini. Gambar 2 menunjukkan gambar tiap label pada dataset.



Gambar 2. Data citra yang telah diberi label

Hasil dan Pembahasan

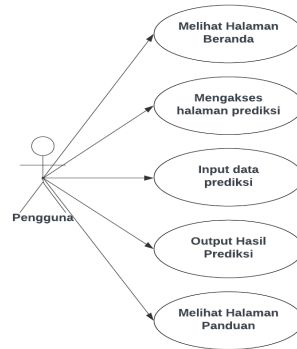
Penerapan Metode

Metode Convolutional Neural Network (CNN) dan Forest Random dapat digunakan untuk mengklasifikasikan huruf aksara Jawa. CNN sangat baik untuk pengenalan gambar. Setelah mengumpulkan gambar huruf aksara Jawa dan melakukan preprocessing seperti normalisasi dan pengubahan ukuran, CNN dibangun dengan beberapa lapisan konvolusi untuk mengekstraksi fitur dari gambar. Untuk klasifikasi, lapisan konvolusi terakhir digunakan. Data ini digunakan untuk melatih model, mengoptimalkannya, dan mengevaluasinya dengan metrik seperti akurasi. Setelah latihan, CNN dapat mengenali dan mengklasifikasikan huruf-huruf aksara Jawa dalam gambar yang dibuat.

Namun, Random Forest biasanya bekerja dengan data datar, sehingga perlu mengekstraksi fitur dari gambar aksara Jawa terlebih dahulu. Setelah fitur diekstraksi, data tersebut digunakan untuk melatih model Random Forest, yang dilatih dengan menentukan jumlah pohon dan kedalaman maksimal. Setelah dilatih dan divalidasi, model dapat digunakan untuk mengklasifikasikan huruf aksara Jawa berdasarkan fitur yang diekstraksi dari gambar baru.

Perancangan

Dalam desain arsitektur perangkat lunak, bahasa model unified (UML) semakin populer. Diagram use case adalah salah satu model UML yang paling umum digunakan saat mendesain aplikasi perangkat lunak. Diagram ini menunjukkan persyaratan fungsional aplikasi dan bagaimana pengguna berinteraksi dengan sistem dan entitas internal dan eksternal. Aplikasi klasifikasi huruf aksara Jawa bertujuan untuk mengkategorikan huruf-huruf aksara Jawa, seperti yang ditunjukkan pada Gambar 3.



Gambar 3. Use case pengguna

Pengguna dapat mengakses halaman beranda dan halaman prediksi, seperti yang ditunjukkan di atas. Di halaman prediksi, pengguna harus memasukkan huruf aksara Jawa yang akan diklasifikasikan, dan sistem akan menampilkan hasil prediksi. Pengguna juga dapat melihat halaman panduan untuk mempelajari cara menggunakan sistem. Tabel berikut menunjukkan contoh bagaimana sistem klasifikasi dapat digunakan.

Preprocessing

Untuk menghindari *overfitting* selama proses pelatihan, tahap *preprocessing* data citra huruf Aksara Jawa sangat penting untuk digunakan. Ini terjadi ketika model bekerja dengan baik saat dilatih tetapi tidak saat dihadapkan pada data baru. Pengaturan *augmentasi* data citra secara otomatis menggunakan kode sebagai berikut:

```

train_datagen = ImageDataGenerator(
    rescale=1./255,
    rotation_range=20,
    horizontal_flip=True,
    width_shift_range=0.2,
    height_shift_range=0.2,
    brightness_range=[0.5, 1.5],
    shear_range=0.2,
    zoom_range=0.2,
    fill_mode = 'nearest')
  
```

Berikut ini adalah penjelasan dari sumber kode di atas:

1. *Rescale* = `rescale=1./255` berfungsi untuk mengubah ukuran data piksel RGB menjadi rentang angka (0-1) untuk memudahkan proses pelatihan data.
2. *Rotation range* = `rotation_range=20` berfungsi untuk mengubah rotasi gambar secara acak, angka 40 menunjukkan besaran derajat rotasi terhadap citra.
3. *Horizontal Flip* = `horizontal_flip=True` berfungsi untuk membalik secara horizontal gambar secara acak dan di atur dengan nilai benar.
4. *Width Shift Range* = `width_shift_range=0.2` berfungsi untuk mengatur posisi gambar pada lebar gambar, angka 0.2 menunjukkan bahwa citra dapat secara acak berada maksimal 20% dari samping atau lebar awal citra.
5. *Height Shift Range* = `height_shift_range=0.2` berfungsi untuk mengatur posisi gambar pada tinggi gambar, angka 0.2 menunjukkan bahwa citra dapat secara acak berada maksimal 20% dari samping atau tinggi awal citra.
6. *Brightness Range* = `brightness_range=[0.5, 1.5]` berfungsi kecerahan gambar akan disesuaikan dalam rentang tersebut. Angka 0.5 mewakili penurunan kecerahan hingga setengah dari nilai aslinya, sedangkan angka 1.5 mewakili peningkatan kecerahan hingga 1.5 kali lipat dari nilai aslinya. Jadi, gambar dapat dibuat lebih gelap (hingga setengah kecerahan asli) atau lebih terang (hingga 1.5 kali kecerahan asli).
7. *Shear Range* = `shear_range=0.2` merupakan sudut geser dalam arah jarum jam dalam derajat.
8. *Zoom Range* = `zoom_range=0.2` berfungsi untuk membesarkan gambar secara acak, 0.2

menunjukkan intensitas pembesaran pada citra gambar.

9. *Fill Mode* = *fill_mode*='nearest' berfungsi untuk pengisian piksel yang kosong

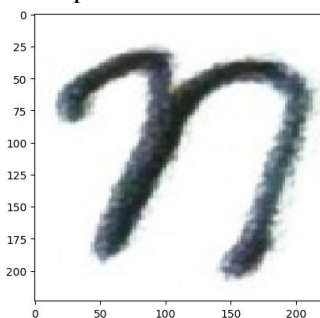
Pelatihan

Algoritma perlu dilatih dengan data pelatihan untuk mencapai akurasi tinggi dalam klasifikasi huruf Aksara Jawa. Tujuan dari penelitian ini adalah untuk mempelajari karakteristik setiap gambar dan menentukan neuron yang diaktifkan saat gambar diklasifikasikan. Metode CNN digunakan untuk pelatihan, yang memerlukan 105 epoch dan 2 jam untuk mencapai akurasi 99,4%. Di sisi lain, model RF dilatih untuk menentukan batas keputusan antar kelas dengan jumlah pohon keputusan atau *n_estimator*, dan mencapai akurasi 42,2%. Hasil terbaik dari setiap model disimpan dalam format h5.

Pengujian Model

Pengujian dataset adalah langkah penting dalam pembelajaran mesin karena memungkinkan evaluasi kinerja model dalam situasi nyata, memastikan akurasi prediksi dalam berbagai kondisi. Dataset diuji dengan hati-hati untuk mencerminkan berbagai situasi, sehingga model tidak hanya akurat untuk satu jenis data. Model CNN dan RF digunakan untuk menganalisis data kompleks, dengan CNN efektif untuk pengenalan gambar dan RF untuk analisis regresi. Evaluasi kinerja pada dataset uji membantu menentukan model terbaik untuk tugas tertentu, memberikan gambaran tentang performa model dalam aplikasi praktis.

The predicted class is: ra



Gambar 4. Pengujian model CNN

Perbandingan Hasil Uji

Dalam evaluasi kinerja model dalam penelitian ini, dilakukan pengukuran menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *f1-score* secara rata-rata.

Tabel 1. Perbandingan skor klasifikasi

	<i>Accuracy (%)</i>	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1-score (%)</i>
<i>Convolutional neural network</i>	07	07	07	07
<i>Random Forest</i>	42	40	42	40

Hasilnya, seperti yang ditunjukkan dalam Tabel 1, menunjukkan bahwa model CNN lebih akurat dalam klasifikasi data daripada model Random Forest. Hasil tes model CNN sebesar 99,4% dengan laporan klasifikasi sebesar 07%, sedangkan model Random Forest hanya mencapai 42%. Ini menunjukkan bahwa model CNN lebih akurat dalam klasifikasi data daripada model Random Forest.

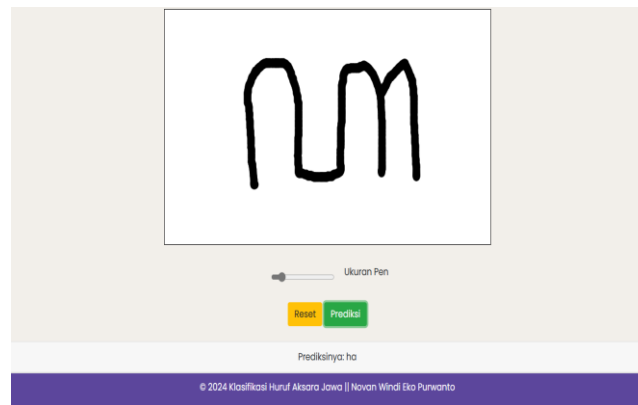
Development Aplikasi

Fokus utama implementasi sistem ini adalah pada pembuatan situs web yang mengumpulkan huruf aksara Jawa. Situs web ini dirancang dengan berorientasi pada pengguna dan memiliki antarmuka yang sederhana dan mudah digunakan, yang bertujuan untuk menyediakan platform agar pengguna dapat belajar huruf aksara Jawa dengan mudah. Gambar 5 menunjukkan cara mengimplementasikan halaman beranda.



Gambar 5. Implementasi halaman beranda

Halaman beranda situs web yang menggunakan sistem klasifikasi huruf aksara Jawa terdiri dari beberapa komponen kunci. Salah satunya adalah navbar, yang berfungsi sebagai menu navigasi untuk memudahkan pengguna berpindah antara halaman beranda, halaman prediksi, dan halaman panduan. Halaman prediksi, yang ditunjukkan pada gambar 6, adalah salah satu halaman utama yang dapat diakses melalui navbar tersebut.



Gambar 6. Implementasi halaman prediksi

Halaman prediksi adalah tampilan pertama yang dilihat pengguna saat memasuki sistem klasifikasi huruf aksara Jawa. Halaman ini memiliki navbar yang serupa dengan halaman beranda, namun di bagian bawah terdapat area kanvas tempat pengguna dapat menggambar huruf aksara Jawa yang ingin diklasifikasikan. Di bawah area kanvas, terdapat dua tombol yang menyediakan fungsi tambahan untuk aplikasi.

Hasil Pengujian Sistem

Pada tahap ini, aplikasi klasifikasi huruf aksara Jawa diuji menggunakan metode black box. Penelitian ini memfokuskan pada evaluasi fungsionalitas aplikasi yang telah dikembangkan. Tabel 2 menunjukkan hasil dari pengujian tersebut.

Tabel 2. Pengujian *black box* pada klasifikasi huruf aksara jawa

No	Menu	Hasil		Kesimpulan
		Normal	Error	
1.	Menu Beranda	✓		Normal
	Tombol Lebih Lanjut	✓		Normal
2.	Menu Prediksi	✓		Normal
	Input Huruf Aksara Jawa	✓		Normal
	Slider Ukuran Pen	✓		Normal
	Tombol Reset	✓		Normal
	Tombol Prediksi	✓		Normal
	Tampilan Hasil Prediksi	✓		Normal

Simpulan

Penelitian ini mengembangkan aplikasi klasifikasi huruf Jawa menggunakan algoritma Convolutional Neural Network (CNN) dan Random Forest (RF). Aplikasi ini dibangun dengan menggunakan bahasa pemrograman Python dan diimplementasikan di situs web menggunakan framework Flask. Evaluasi model dilakukan dengan laporan klasifikasi dan confusion matrix, yang menunjukkan bahwa CNN mencapai akurasi 99,4%, jauh lebih tinggi dibandingkan dengan RF yang hanya mencapai akurasi 42,2%. Selain itu, CNN juga menunjukkan skor precision, recall, dan f1 yang lebih baik daripada RF. Dengan demikian, metode CNN terbukti lebih efektif untuk klasifikasi huruf aksara Jawa. Aplikasi ini diharapkan dapat mempermudah pengguna dalam mengenali dan memahami huruf aksara Jawa dengan lebih akurat dan efisien.

Daftar Pustaka

- Adrian, M. R., Putra, M. P., Rafialdy, M. H., & Rakhmawati, N. A. (2021). Perbandingan Metode Klasifikasi Random Forest dan SVM Pada Analisis Sentimen PSBB. *Jurnal Informatika Upgris*, 7(1), 36–40. <https://doi.org/10.26877/jiu.v7i1.7099>
- Alwanda, M. R., Ramadhan, R. P. K., & Alamsyah, D. (2020). Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle. *Jurnal Algoritme*, 1(1), 45–56. <https://doi.org/10.35957/algoritme.v1i1.434>
- Arbain, A., Muhammad, M. A., Septiana, T., & Septama, H. D. (2022). Learning Hoax News Pada Local Dan Cloud Computing Deployment Menggunakan Google App Engine. *Jurnal Informatika Dan Teknik Elektro Terapan*, 10(3). <https://doi.org/10.23960/jitet.v10i3.2646>
- Baranwal, A., Bagwe, B. R., & M, V. (2019). Machine Learning in Python. *Journal of Machine Learning Research*, 12, 128–154. <https://doi.org/10.4018/978-1-5225-9902-9.ch008>
- Dutta, K. K., Sunny, S. A., Victor, A., Nathu, A. G., Ayman Habib, M., & Parashar, D. (2020). Kannada Alphabets Recognition Using Decision Tree and Random Forest Models. *Proceedings of the 3rd International Conference on Intelligent Sustainable Systems, ICISS 2020*, 534–541. <https://doi.org/10.1109/ICISS49785.2020.9315972>
- Faizin, A., Moh. Lutfi, & Achmyatari. (2022). Perbandingan Arsitektur Lenet Dan Googlenet Dalam Klasifikasi Diabetic Retinopathy Pada Citra Retina Fundus. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 6(1), 342–347. <https://doi.org/10.36040/jati.v6i1.4581>
- Farid Naufal, M. (2021). Analisis Perbandingan Algoritma SVM, KNN, dan CNN untuk Klasifikasi Citra Cuaca. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 8(2), 311–318. <https://doi.org/10.25126/jtiik.202184553>
- Hackeling, G. (2014). *Mastering Machine Learning with scikitlearn* (A. S. Roshni Banerjee (ed.); Vol. 2507, Issue 1). Packt Publishing Ltd.
- Hakim, L., Saefuddin, A., & Nisrina, S. (2022). Klasifikasi Varietas Unggul Padi Menggunakan Metode Bagging, Boosting, dan Extremely Randomized Trees. *Statistika*, 22(2), 127–132.
- Hidayat, D., Rachmiatie, A., & Rizkyana, D. (2020). Perilaku Komunikasi dalam Konteks Hanacaraka Masyarakat Kampung Cireunde. *Jurnal Komunikasi Universitas Garut: Hasil Pemikiran Dan Penelitian*, 6(2), 495–508. <https://journal.uniga.ac.id/index.php/JK/article/view/747>
- Hikmatia A.E, N., & Ihsan Zul, M. (2021). Aplikasi Penerjemah Bahasa Isyarat Indonesia menjadi Suara berbasis Android menggunakan Tensorflow. *Jurnal Komputer Terapan*, 7(Vol. 7 No. 1 (2021)), 74–83. <https://doi.org/10.35143/jkt.v7i1.4629>
- Hutauruk, A. C., & Pakpahan, A. F. (2021). Perancangan Sistem Informasi Organisasi Kemahasiswaan Berbasis Web pada Universitas Advent Indonesia Menggunakan Metode Agile Development (Studi Kasus: Universitas Advent Indonesia) Design of Web-Based Student Organization Information System at Adventist. *Cogito Smart Journal*, 7(2), 2021.
- Ju, Y., Wang, X., & Chen, X. (2019). Research on OMR recognition based on convolutional neural network tensorflow platform. *Proceedings - 2019 11th International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2019*, 688–691. <https://doi.org/10.1109/ICMTMA.2019.00157>
- Kholik, A. (2021). Klasifikasi Menggunakan Convolutional Neural Network (Cnn) Pada Tangkapan

- Layar Halaman Instagram. *Jdmsi*, 2(2), 10–20.
- Kurniadi, B. W., Prasetyo, H., Ahmad, G. L., Aditya Wibisono, B., & Sandya Prasvita, D. (2021). Analisis Perbandingan Algoritma SVM dan CNN untuk Klasifikasi Buah. *Seminar Nasional Mahasiswa Ilmu Komputer Dan Aplikasinya (SENAMIKA) Jakarta-Indonesia, September*, 1–11.
- Lee, H., Kim, J., Jung, S., Kim, M., Kim, B., & Kim, S. (2020). Variable importance measures based on ensemble learning methods for convective storm tracking. *2020 Joint 11th International Conference on Soft Computing and Intelligent Systems and 21st International Symposium on Advanced Intelligent Systems, SCIS-ISIS 2020*.
<https://doi.org/10.1109/SCISISIS50064.2020.9322692>
- Nalatissifa, H., Gata, W., Diantika, S., & Nisa, K. (2021). Perbandingan Kinerja Algoritma Klasifikasi Naive Bayes, Support Vector Machine (SVM), dan Random Forest untuk Prediksi Ketidakhadiran di Tempat Kerja. *Jurnal Informatika Universitas Pamulang*, 5(4), 578.
<https://doi.org/10.32493/informatika.v5i4.7575>
- Nugroho, P. A., Fenriana, I., & Arijanto, R. (2020). Implementasi Deep Learning Menggunakan Convolutional Neural Network (CNN) Pada Ekspresi Manusia. *Jurnal Algor, Vol 2*(No 1), 12–21.
- Puspitasari, F. D., & Anardani, S. (2023). Aplikasi Klasifikasi Huruf Hijaiyah Menggunakan Algoritma Convolutional Neural Network dan Random Forest. *Seminar Nasional Teknologi Informasi Dan Komunikasi*, 67–78.
- Qotrunnada, F. M., & Utomo, P. H. (2022). Metode Convolutional Neural Network untuk Klasifikasi Wajah Bermasker. *Prisma*, 5, 799–807.
- R.H. Zer, P. P. P. A. N. . F. I., Hayadi, B. H., & Damanik, A. R. (2022). Pendekatan Machine Learning Menggunakan Algoritma C4.5 Berbasis Pso Dalam Analisa Pemahaman Pemrograman Website. *Jurnal Informatika Dan Teknik Elektro Terapan*, 10(3).
<https://doi.org/10.23960/jitet.v10i3.2700>
- Syarif, M., & Nugraha, W. (2020). Pemodelan Diagram UML Sistem Pembayaran Tunai Pada Transaksi E-Commerce. *Jurnal Teknik Informatika Kaputama (JTIK)*, 4(1), 70 halaman.
- Wicaksono, S. R. (2021). *Blackbox Testing Teori dan Studi Kasus* (S. R. Wicaksono (ed.); 1st ed., Issue Juli). CV. Seribu Bintang. <https://doi.org/10.5281/zenodo.7659674>