

Aplikasi Klasifikasi Huruf Hijaiyah Menggunakan Algoritma Convolutional Neural Network dan Random Forest

Frizsa Dias Puspitasari¹, Sri Anardani²

^{1,2}Universitas PGRI Madiun

email: frizsa_1905101051@mhs.unipma.ac.id

Abstract: This research aims to implement Convolutional Neural Network (CNN) and Random Forest (RF) algorithms in an Arabic letter classification system. The system is built using Python programming language, HTML, and CSS with the Flask framework through the Agile Development application development method. This approach allows developers to respond quickly and flexibly to changing requirements by following several stages of planning, analysis, design implementation, and evaluation. Comparing the CNN model with the RF model shows that CNN achieves a higher accuracy rate of 99.7%, while RF achieves around 74%. Overall, this research demonstrates impressive results regarding accuracy when implementing the CNN algorithm in a web-based application for Arabic letter recognition using the Agile Development methodology.

Keywords: Convolutional Neural Network, Random Forest, Arabic Letters

Abstrak: Penelitian ini bertujuan untuk menerapkan algoritma Convolutional Neural Network (CNN) dan Random Forest (RF) pada sistem klasifikasi huruf Arab. Sistem dibangun menggunakan bahasa pemrograman Python, HTML, CSS dengan framework Flask melalui metode pengembangan aplikasi website yang disebut Agile Development. Pendekatan ini memungkinkan pengembang untuk merespon dengan cepat dan fleksibel terhadap perubahan kebutuhan dengan mengikuti beberapa tahapan perencanaan, analisis, implementasi desain dan evaluasi. Membandingkan model CNN dengan model RF menunjukkan bahwa CNN memiliki tingkat akurasi yang lebih tinggi mencapai 99,7% sedangkan RF sekitar 74%. Secara keseluruhan penelitian ini menunjukkan hasil yang mengesankan terkait akurasi ketika mengimplementasikan algoritma CNN pada aplikasi berbasis web pengenalan huruf Arab melalui metodologi Agile Development.

Kata kunci: Convolutional Neural Network, Random Forest, Huruf Hijaiyah

Pendahuluan

Huruf hijaiyah merujuk pada huruf atau kata penyusun dalam Al-Qur'an. Menurut para ulama ahli Tajwid, jumlah huruf hijaiyah yang terdapat dalam Al-Qur'an mencapai 30, antara lain (ا) Alif, (ب) Ba, (ت) Ta, (ث) Tsa, (ج) Jim, (ح) Ha, (خ) Kha, (د) Dal, (ذ) Dzal, (ر) Ra, (ز) Zay, (س) Sin, (ش) Syin, (ص) Shod, (ض) Dhod, (ط) Tho, (ظ) Dzo, (ع) Ain, (غ) Ghoin, (ف) Fa, (ق) Qof, (ك) Kaf, (ل) Lam, (م) Mim, (ن) Nun, (و) Waw, (ه) Ha, (ي) Lam Alif, (ء) Hamzah, dan (يا) Ya (Abror Mufasssiril, 2021).

Klasifikasi adalah proses pengelompokan data atau analisis yang dilakukan berdasarkan karakteristik yang dimiliki oleh setiap objek. Tujuannya adalah untuk memberikan kategori tertentu untuk setiap objek yang diklasifikasikan (Widiastuti dkk, 2023).

Machine learning didasarkan pada gagasan bahwa sistem dapat belajar dari data, mengidentifikasi pola, dan membuat keputusan dengan sedikit campur tangan manusia (Chahal & Gulia, 2019).

CNN (Convolutional Neural Network) adalah tipe model *deep learning* untuk memproses data yang memiliki pola *grid*, seperti gambar, yang terinspirasi dari organisasi korteks visual hewan (Yamashita dkk, 2021). Convolutional Neural Networks (CNN) atau Jaringan Saraf Konvulsional adalah jenis jaringan saraf tiruan *feed-forward* yang memiliki struktur hierarkis. Dalam CNN, jaringan saraf ini dapat belajar representasi fitur internal dari data dan secara efektif menggeneralisasi fitur-fitur tersebut untuk menyelesaikan berbagai masalah gambar umum, seperti pengenalan objek dan tugas-tugas penglihatan komputer

lainnya. Selain digunakan untuk tugas pengenalan gambar, CNN juga telah menunjukkan hasil yang sangat baik dalam pemrosesan bahasa alami dan pengenalan suara, menjadikannya alat yang serbaguna untuk berbagai aplikasi di berbagai bidang (Raharjo, 2022).

Random Forest adalah model pembelajaran mesin yang menggunakan pembelajaran terawasi dan terdiri dari banyak pohon keputusan. Pohon keputusan bertindak sebagai klasifikasi dasar dalam hutan acak dan bekerja bersama sebagai sebuah kesatuan. Inilah sebabnya mengapa *Random Forest* kadang-kadang disebut sebagai *Classifier Ensembles* (Dutta dkk, 2020).

Flask merupakan kerangka kerja aplikasi *website* yang menggunakan bahasa pemrograman python dan menggunakan dependensi *Werkzeug* dan *Jinja2*. Dalam tahap pengemabangan sistem *hoax news detection* ini *flask* berfungsi sebagai pengontrol dalam pengembangan sistem *hoax news detection* (Arbain dkk, 2022).

Python adalah sebuah bahasa pemrograman yang berfokus pada teks sebagai bentuk komunikasi utama dengan *programmer*. Dalam *Python*, pengembang menulis kode menggunakan sejumlah kata-kata, singkatan, angka, dan simbol tertentu. Untuk memberikan instruksi kepada komputer, *programmer* mengetikkan kode menggunakan *keyboard* komputer (Santoso, 2022).

Bahasa Arab menggunakan alfabet yang disebut Hijaiyah, yang terdiri dari 30 huruf yang ditulis dari kanan ke kiri. Huruf-huruf ini memainkan peran penting dalam Islam karena digunakan untuk menulis ayat-ayat Al-Quran dan teks-teks keagamaan lainnya. Beberapa huruf memiliki bentuk yang mirip tetapi berbeda dengan titik atau tanda di atas atau di bawahnya, sehingga membutuhkan pengenalan yang cermat saat membaca tulisan Arab. Memahami karakter-karakter ini sangat penting untuk mempelajari budaya serta agama. Namun, hal ini dapat menjadi tantangan karena variasi bentuk dan perbedaan pengucapan. Untuk menghadapi permasalahan tersebut dapat digunakan metode klasifikasi gambar yang memiliki kemampuan untuk mengelompokkan objek berdasarkan pola.

Dalam menanggapi permasalahan yang dihadapi, peneliti melakukan studi dengan merujuk pada berbagai penelitian sebelumnya. Referensi-referensi dari berbagai sumber tersebut menjadi dasar tertulis bagi penelitian yang sedang dilakukan. Pada penelitian pertama mengenai penerapan beberapa metode termasuk SVM, KNN, dan CNN untuk mengklasifikasikan citra cuaca. Pada penelitian tersebut menyatakan bahwa CNN memiliki kinerja terbaik dalam mengklasifikasikan dataset cuaca dengan akurasi, presisi, *recall*, dan F1 score mencapai 0,942, 0,943, 0,942, dan 0,942 secara berturut-turut. Meski demikian, CNN memerlukan waktu eksekusi terlalu lama, yakni sekitar 458,49 detik untuk mencapai kinerja terbaik (Farid Naufal, 2021).

Pada penelitian kedua menggunakan metode CNN dan SVM untuk mengklasifikasikan buah berdasarkan bentuknya. Hasil penelitian ini menyatakan bahwa CNN berhasil mencapai akurasi sebesar 96,87%, lebih tinggi dibandingkan SVM yang hanya mencapai akurasi sebesar 93,09% (Kurniadi dkk, 2021). Penelitian ketiga yang berkaitan dengan CNN yaitu bertujuan untuk mengklasifikasikan lukisan karya *Van Gogh* menggunakan *Convolutional Neural Network-Support Vector Machine* (CNN-SVM). Dalam penelitian ini, beberapa metode ekstraksi fitur dan optimasi kernel diterapkan untuk menguji performa model yang dihasilkan. Hasil penelitian menunjukkan bahwa penggunaan ekstraksi fitur VGG-19 dengan kernel *linear* dan optimasi *grid* menghasilkan performa terbaik, dengan tingkat akurasi mencapai 93,00%, *precision* sebesar 87,00%, dan *recall* sebesar 94,00%. Model ini menjadi pilihan utama karena memberikan akurasi yang tinggi dalam mengklasifikasikan lukisan karya *Van Gogh*. Penggunaan model ResNet-50 dengan optimasi *random* menghasilkan akurasi sebesar 92,89%, sedangkan dengan optimasi *grid* hanya mencapai 90,28%. Kedua model ResNet-50 tersebut

menunjukkan performa yang lebih rendah dibandingkan dengan VGG-19 dengan optimasi *grid* (Yohannes dkk, 2021).

Penelitian terkait penggunaan metode *Random Forest* dalam analisis sentimen dilakukan oleh (Adrian dkk, 2021) dengan judul "Perbandingan Metode Klasifikasi *Random Forest* dan SVM Pada Analisis Sentimen PSBB" menyatakan bahwa bahasa yang beragam pada *Twitter* dapat menurunkan kemampuan model dalam memprediksi sentimen terkait PSBB. SVM dinilai lebih unggul dalam mengenali *tweet* yang dikategorikan sebagai "Positif".

Penelitian lain terkait *Random Forest* yakni membahas tentang penggunaan teknik *Random Forest* dalam mengklasifikasikan mahasiswa berdasarkan kinerja akademik mereka. Penelitian ini menunjukkan bahwa teknik *Random Forest* dapat digunakan secara efektif dalam mengklasifikasikan mahasiswa berdasarkan kinerja akademik mereka (Amutha & Priya, 2022). Penelitian lain membahas terkait teknik *Random Forest* dan aplikasinya dalam pembelajaran statistik. Penelitian ini menunjukkan bahwa teknik *Random Forest* dapat digunakan untuk mengestimasi tingkat kesalahan dengan lebih akurat dibandingkan dengan teknik *decision tree* (Schonlau & Zou, 2020).

Dapat diketahui bahwa berdasarkan referensi dari penelitian sebelumnya belum ada penelitian yang berkaitan dengan penerapan algoritma Convolutional Neural Network dan Random Forest yang digunakan sebagai klasifikasi pada objek huruf hijaiyah.

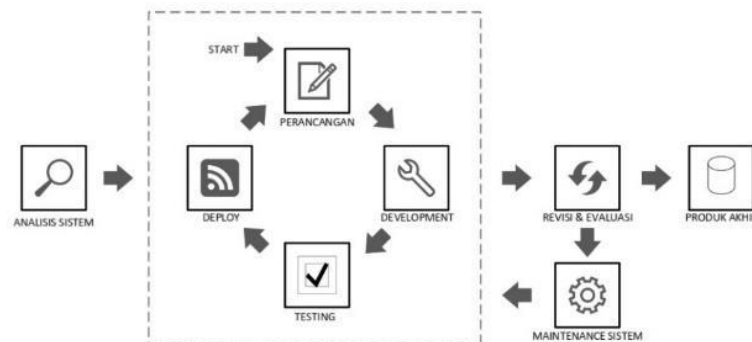
Klasifikasi citra menggunakan *Convolutional Neural Network* (CNN) dapat mempermudah dalam mengidentifikasi objek pada citra, terutama untuk siswa sekolah dasar yang sedang belajar huruf hijaiyah. Metode CNN memiliki kelebihan dibandingkan metode lain karena dapat mengekstrak fitur dari berbagai jenis input seperti gambar, suara, atau teks tanpa memperhatikan posisinya. Metode lain yang digunakan dalam klasifikasi gambar adalah *Random Forest*. Penelitian ini bertujuan untuk membandingkan akurasi dari dua algoritma, *Convolutional Neural Network* (CNN) dan *Random Forest*, dalam mengklasifikasikan huruf Arab. Peneliti akan menerapkan kedua algoritma tersebut pada sebuah dataset huruf Arab dan membandingkan hasil klasifikasinya untuk menentukan algoritma mana yang lebih akurat. Penelitian ini dapat bermanfaat untuk mengembangkan aplikasi yang mengajarkan siswa sekolah dasar dalam mengenali huruf Arab.

Metode

Metode Pengembangan Sistem

Penelitian ini menerapkan konsep *Agile Development* sebagai metode dalam penelitian perancangan sistem ini. Konsep ini bertujuan agar pengembangan sistem dapat ditinjau dan dilakukan revisi dengan lebih mudah. Pendekatan *Agile Development* memiliki tingkat keberhasilan pengembangan proyek yang lebih baik dibandingkan dengan metode desain terstruktur. Model *Agile* adalah suatu pendekatan dalam pengembangan sistem yang memungkinkan adaptasi dan perubahan dilakukan dengan fleksibilitas, memberikan kemampuan untuk menghadapi perubahan kebutuhan setiap saat (Nova dkk, 2022).

Tahapan penelitian *Agile Development* untuk pengembangan aplikasi klasifikasi huruf hijaiyah dapat dilihat pada Gambar 1.



Gambar 1. Metode *Agile Development*

Sumber: (Suhari dkk, 2022)

Berikut ini adalah tahapan penelitian yang berasal dari *Agile Development* untuk pengembangan aplikasi klasifikasi huruf hijaiyah.

1. Analisis Sistem

Analisis sistem adalah proses evaluasi terhadap pengguna untuk mendapatkan pemahaman mendalam tentang kebutuhan perangkat lunak yang dibutuhkan oleh mereka. Dalam analisis sistem, fokus diberikan pada pemahaman mendalam mengenai kebutuhan perangkat lunak dan perangkat keras yang diinginkan oleh pengguna.

2. Perancangan

Dalam perancangan ini, digunakan pemodelan UML (*Unified Modeling Language*). Pemodelan UML akan menghasilkan diagram *use case*, dan *sequence diagram*.

3. *Development* Aplikasi

Development aplikasi merupakan tahap implementasi dari perancangan aplikasi yang meliputi pengkodean dan pembuatan database. Proses pengimplementasian perancangan aplikasi ini menggunakan *framework*.

4. *Testing*

Pengujian sistem dilakukan menggunakan metode *black box testing*, yang bertujuan untuk menguji perangkat lunak secara keseluruhan untuk memastikan bahwa sistem berfungsi sebagaimana mestinya sesuai dengan kebutuhan fungsional yang telah ditentukan.

Metode Penelitian

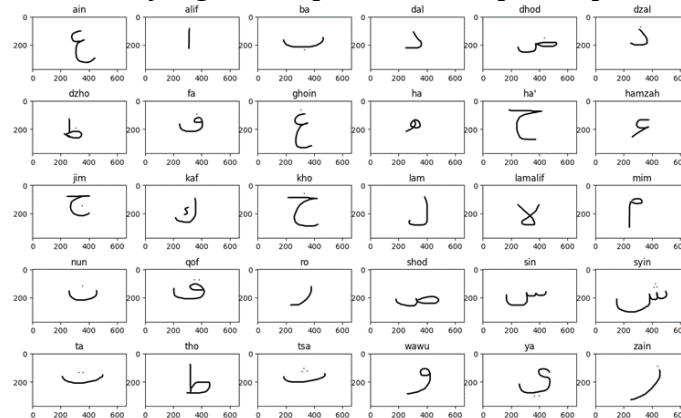
Pada penelitian ini menerapkan dua metode yaitu metode *Convolutional Neural Network* dan *Random Forest*. CNN merupakan sebuah algoritma *machine learning* yang merupakan pengembangan dari *Multilayer Perceptron* (MLP). Algoritma ini termasuk dalam kategori *Deep Learning* dan memiliki kemampuan untuk melakukan klasifikasi gambar maupun suara (Hasyim dkk, 2021). Arsitektur CNN didasarkan dari jaringan saraf biologi yang terdiri dari lapisan *convolutional*, *pooling*, dan *fully-connected*. Lapisan-lapisan inilah yang membentuk arsitektur dari CNN. Setiap *neuron* pada CNN diwakili dalam bentuk tiga dimensi, sehingga CNN sangat cocok digunakan dalam pemrosesan *input* berupa gambar (Abdulhakim dkk, 2021).

Sedangkan *Random Forest* adalah suatu modifikasi dari Pohon Keputusan (*Decision Tree*) yang telah banyak digunakan dalam klasifikasi citra (Ratnawati & Sulistyaningrum, 2020). *Random Forest* (RF) adalah suatu algoritma yang menggunakan metode pemisahan biner secara rekursif untuk mencapai simpul akhir dalam struktur pohon. Algoritma ini digunakan dalam pohon klasifikasi dan regresi. Salah satu keunggulan dari *Random Forest* adalah kemampuannya menghasilkan tingkat kesalahan yang relatif rendah, performa yang

baik dalam tugas klasifikasi, serta mampu mengatasi data pelatihan dalam jumlah besar dengan efisien. Selain itu, algoritma ini juga efektif dalam mengestimasi data yang hilang (Pamuji & Ramadhan, 2021).

Datasets

Datasets yang digunakan pada penelitian ini di dapatkan dari platform *github* yang dapat diakses secara terbuka. *Datasets* berupa citra dengan jumlah 30 label, dan setiap label terdapat 56 citra. Sehingga citra keseluruhan yang digunakan dalam penelitian ini adalah 1.680 citra. Berikut adalah Gambar 2 yang menampilkan citra tiap label pada datasets.



Gambar 2. Citra huruf hijaiyah dari setiap label

Hasil

Analisis Sistem

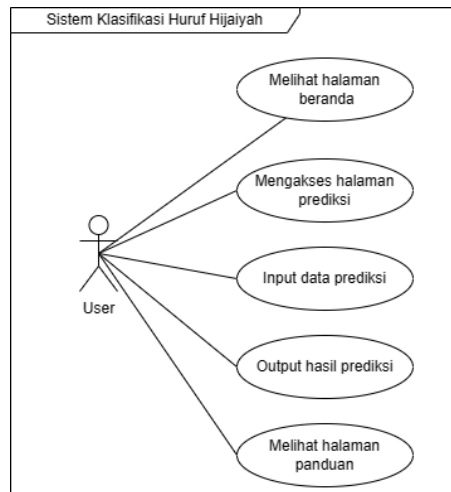
Analisis Sistem merupakan penjabaran dari sebuah sistem secara utuh menjadi beberapa komponen kecil dengan tujuan untuk mengidentifikasi permasalahan serta hambatan yang akan diperbaiki. Adapun perangkat lunak dapat dilihat pada Tabel 1.

Tabel 1. Perangkat lunak kebutuhan sistem

Spesifikasi	Keterangan
Sistem Operasi	Windows 11
Web Browser	Microsoft Edge dan Google Chrome
Text Editor	Jupyter Notebook
Framework Website	Flask
Framework Model	Keras
Bahasa Pemrograman	Python

Perancangan

Penggunaan *Unified Modeling Language* (UML) dalam desain arsitektur telah menjadi semakin populer. Salah satu model UML yang paling umum digunakan untuk mendesain aplikasi perangkat lunak adalah diagram use case. Diagram ini berfungsi sebagai representasi visual dari persyaratan fungsional aplikasi, yang menggambarkan interaksi antara pengguna, sistem, dan entitas eksternal. Contoh *use case* diagram adalah yang digunakan dalam perancangan aplikasi klasifikasi abjad Arab. Aplikasi ini dirancang untuk mengklasifikasikan huruf-huruf abjad Arab, khususnya huruf-huruf aksara Hijaiyah. *Use case* diagram untuk aplikasi ini dapat dilihat pada Gambar 3.

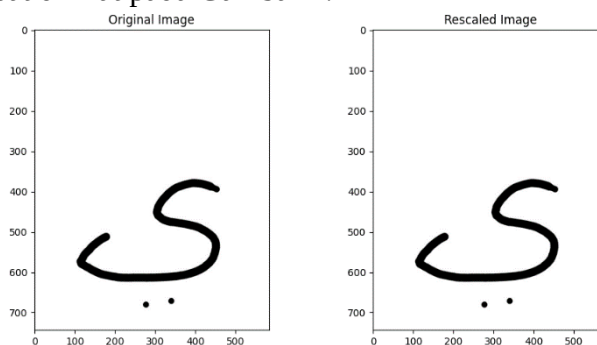


Gambar 3. Use case pengguna

Pada gambar diatas menyampaikan bahwa user dapat melihat halaman beranda serta mengakses halaman prediksi yang di dalamnya user perlu input huruf hijaiyah yang akan di klasifikasi kemudian sistem akan menampilkan output berupa hasil prediksi. Selanjutnya, user juga dapat melihat halaman panduan untuk mengetahui langkah-langkah dalam menggunakan sistem. Di bawah ini adalah tabel skenario use case yang terdapat dalam sistem klasifikasi.

Preprocessing

Tahap *preprocessing* data citra huruf Hijaiyah digunakan dalam proses pelatihan untuk mencegah terjadinya *overfitting*, yang dapat terjadi ketika model memiliki kinerja yang baik saat dilatih tetapi buruk saat dihadapkan pada data baru. Berikut adalah contoh hasil dari preprocessing yang dapat dilihat pada Gambar 4.



Gambar 4. Huruf hijaiyah sebelum dan sesudah proses *rescale*

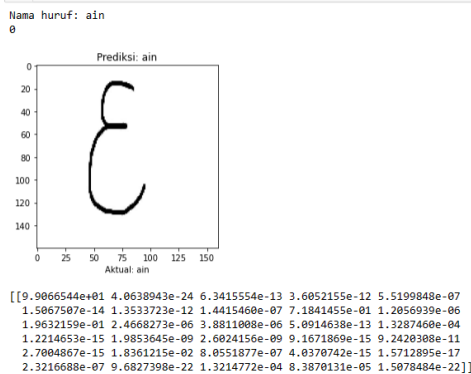
Pelatihan

Dalam klasifikasi huruf hijaiyah, untuk mencapai akurasi pengenalan yang tinggi, algoritma perlu dilatih dengan sekumpulan data pelatihan. Tujuannya adalah untuk mengidentifikasi karakteristik setiap gambar dan menandai *neuron* mana yang akan diaktifkan ketika gambar diklasifikasikan. Para peneliti menggunakan metode CNN untuk pelatihan, yang membutuhkan 105 epoch dan 4 jam untuk mencapai akurasi 99,7%. Di sisi lain, metode RF digunakan untuk melatih model RF untuk mempelajari batas-batas keputusan yang optimal antar kelas. Proses ini melibatkan *n_estimator* atau jumlah pohon keputusan. Akurasi yang diperoleh dari pelatihan model RF adalah 74%. Hasil terbaik dari setiap model disimpan dalam format h5.

Pengujian Model

Pengujian dataset merupakan langkah penting dalam proses pembelajaran mesin. Hal ini memungkinkan untuk mengevaluasi kinerja model dalam skenario dunia nyata, memastikan bahwa model tersebut dapat secara akurat memprediksi hasil dalam kondisi yang berbeda. Dataset yang digunakan dalam pengujian dipilih dengan cermat untuk mewakili beragam skenario, memastikan bahwa akurasi model tidak terbatas pada kumpulan data tertentu.

Model CNN dan RF biasanya digunakan dalam pembelajaran mesin karena kemampuannya untuk menganalisis kumpulan data yang kompleks dengan cepat. Model CNN sangat berguna dalam tugas pengenalan gambar dan klasifikasi, sedangkan model RF sering digunakan dalam analisis regresi. Kedua model tersebut memiliki kekuatan dan kelemahan yang unik, dan sangat penting untuk mengevaluasi kinerjanya dalam menguji set data untuk menentukan model mana yang paling cocok untuk tugas tertentu. Pada tahap pengujian dataset, hasil model yang telah dilatih dimasukkan untuk mengevaluasi kinerja model yang telah dilatih. Tahap pengujian ini sangat penting dalam menentukan keandalan dan akurasi model. Model CNN telah melalui proses pengujian, dan hasilnya dapat dilihat pada gambar 5.



Gambar 5. 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. Pada Tabel 1, terdapat dua kali proses pelatihan pada model CNN yang dilakukan dengan variasi nilai epoch. Tabel tersebut menyajikan nilai epoch (jumlah iterasi pelatihan) dan hasil akurasi yang diperoleh dari setiap proses pelatihan. Dalam proses pelatihan pertama, model CNN dilatih dengan 50 epoch. Pada proses pelatihan kedua, model CNN dilatih dengan 105 epoch. Pada kasus ini, kenaikan dari 50 epoch menjadi 105 epoch menghasilkan peningkatan akurasi dari 95% menjadi 100%.

Tabel 2. Nilai Akurasi berdasarkan epoch pada model CNN

Nilai Epoch	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
50	95	95	95	95
105	100	100	100	100

Tabel 2 mengilustrasikan hasil dari tiga putaran pelatihan menggunakan model Random Forest dengan nilai *n_estimator* yang bervariasi, yang ditetapkan pada 100, 150, dan 200. Tabel tersebut menampilkan nilai *n_estimator*, yang merepresentasikan jumlah pohon keputusan yang digunakan pada setiap putaran pelatihan, serta nilai akurasi yang diperoleh dari setiap sesi pelatihan.

Tabel 3. Nilai Akurasi berdasarkan $n_estimator$ pada model RF

Nilai $n_estimator$	<i>Accuracy</i> (%)	<i>Precision</i> (%)	<i>Recall</i> (%)	<i>F1-score</i> (%)
100	73	76	73	74
150	74	78	74	74
200	71	74	71	71

Dengan demikian, berdasarkan Tabel 3, dapat disimpulkan bahwa model CNN memiliki tingkat akurasi yang lebih tinggi dibandingkan dengan model *Random Forest*. Model CNN mencapai akurasi sebesar 100%, sedangkan model *Random Forest* hanya mencapai akurasi 74%. Hal ini menunjukkan bahwa model CNN mampu melakukan klasifikasi data dengan tingkat keakuratan yang lebih tinggi dibandingkan dengan model *Random Forest*. Dengan hasil tersebut, maka metode CNN diterapkan pada aplikasi klasifikasi huruf hijaiyah.

Tabel 4. Perbandingan skor klasifikasi

	<i>Accuracy</i> (%)	<i>Precision</i> (%)	<i>Recall</i> (%)	<i>F1-score</i> (%)
<i>Convolutional neural network</i>	100	100	100	100
<i>Random Forest</i>	74	78	74	74

Development Aplikasi

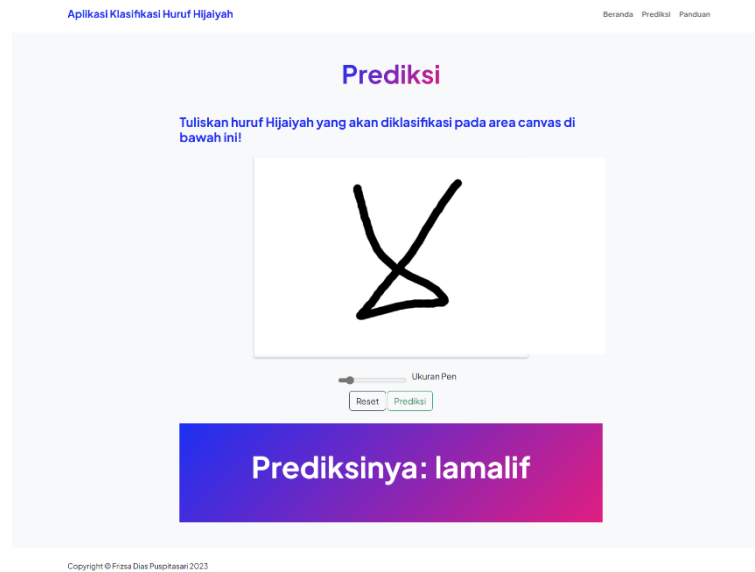
Dalam implementasi sistem ini, fokus utamanya adalah pada pengembangan situs web yang mengklasifikasikan alfabet Arab yang dikenal sebagai huruf Hijaiyah. Situs web ini dirancang dengan mempertimbangkan pengguna, dengan penekanan pada pembuatan antarmuka yang sederhana dan mudah digunakan. Tujuan dari sistem ini adalah untuk menyediakan platform di mana pengguna dapat dengan mudah mengakses dan mempelajari huruf-huruf Hijaiyah. Berikut adalah implementasi halaman beranda yang dapat dilihat pada gambar 6.



Gambar 6. Implementasi halaman beranda

Halaman beranda pada website sistem klasifikasi huruf hijaiyah memiliki beberapa komponen yang sangat penting. Salah satunya adalah *navbar*, yang berfungsi sebagai menu navigasi yang mempermudah pengguna untuk beralih antara halaman beranda, halaman

prediksi, dan halaman panduan. Selain itu terdapat halaman prediksi yang dapat dilihat pada gambar 7.



Gambar 7. Implementasi halaman prediksi

Halaman prediksi merupakan halaman pertama yang akan diakses oleh pengguna jika masuk ke sistem klasifikasi huruf hijaiyah. Pada halaman ini juga terdapat navbar sama halnya dengan halaman beranda. Hal yang membedakan halaman ini dengan halaman lain adalah terdapat area canvas yang digunakan pengguna untuk input huruf hijaiyah yang akan di klasifikasi. Bagian bawah canvas terdapat slider yang berfungsi untuk mengatur ukuran pen yang akan digunakan untuk input huruf hijaiyah. Kemudian di bagian bawah terdapat dua button yaitu button reset yang berfungsi untuk menghapus goresan pen yang sebelumnya sudah dilakukan. Button kedua yaitu button prediksi yang berfungsi untuk menampilkan hasil prediksi jika input huruf hijaiyah sudah dimasukkan.

Testing

Pada tahapan ini pengujian aplikasi klasifikasi huruf hijaiyah menggunakan metode *black box* yang berfokus pada fungsionalitas dari aplikasi yang telah dibuat. Berikut adalah hasil pengujian yang dapat dilihat pada Tabel 5.

Tabel 5. Pengujian *black box* pada klasifikasi huruf hijaiyah

No	Menu	Hasil		Kesimpulan
		Normal	Error	
1	Menu Beranda	✓		Normal
	Tombol Mulai Prediksi	✓		Normal
2	Menu Prediksi	✓		Normal
	Input Huruf Hijaiyah	✓		Normal
	Slider Ukuran Pen	✓		Normal
	Tombol Reset	✓		Normal
	Tombol Prediksi	✓		Normal
	Tampilan Hasil Prediksi	✓		Normal
3	Menu Panduan	✓		Normal

Pembahasan

Pembangunan dan perancangan aplikasi klasifikasi huruf hijaiyah berbasis web ini bertujuan untuk mempermudah siswa dalam mengenal huruf hijaiyah. Selain itu, juga dapat menjadi media alternatif untuk belajar mengenal huruf hijaiyah dengan cara yang menyenangkan. Aplikasi ini dibangun berbasis website dengan menerapkan metode CNN. *World Wide Web (WWW)* adalah suatu sistem yang memungkinkan akses informasi dalam internet dan dikenal juga sebagai web. Untuk beroperasi, web menggunakan protokol bernama *HTTP (Hypertext Transfer Protocol)* yang berjalan di atas protokol *TCP/IP*. Melalui konsep *hypertext*, pengguna dapat dengan mudah mengakses dokumen-dokumen lain dengan cara mengklik teks-teks khusus yang ditandai dengan garis bawah (Ahmad Martani dkk, 2022). *Convolutional Neural Network (CNN)* merupakan sebuah pengembangan dari *Multilayer Perceptron (MLP)* yang secara khusus dirancang untuk memproses dan mengolah data berdimensi dua (Ihsan, 2021). Aplikasi ini menerapkan *framework flask* dalam proses pembangunannya. *Flask* adalah sebuah web *framework* yang dibuat menggunakan bahasa pemrograman *Python* dan termasuk dalam kategori *microframework*. Fungsinya adalah sebagai kerangka kerja untuk membangun aplikasi web dan mengatur tampilan dari sebuah situs web (Nugroho dkk, 2020).

Hasil pengujian pada aplikasi klasifikasi huruf hijaiyah ini menerapkan metode *blackbox testing*. Berdasarkan pengujian setiap fungsi dari aplikasi dapat dijalankan sesuai dengan yang dibutuhkan. Keterbatasan aplikasi klasifikasi huruf hijaiyah ini adalah aplikasi hanya dapat mengklasifikasikan huruf hijaiyah yang sudah dilakukan proses pelatihan sebelumnya, sehingga tidak dapat mengklasifikasikan huruf hijaiyah yang menyambung. Implikasi dari aplikasi ini yaitu dapat memberikan hasil prediksi dari huruf hijaiyah yang dimasukkan oleh pengguna.

Simpulan

Penelitian ini bertujuan untuk mengklasifikasikan huruf hijaiyah dengan menggunakan *Convolutional Neural Network (CNN)* dan *Random Forest (RF)*. *Python* digunakan untuk membuat model dan diaplikasikan dalam sebuah situs web menggunakan *framework Flask*. Evaluasi dilakukan menggunakan *confusion matrix* dan laporan klasifikasi untuk menguji akurasi dan efisiensi kedua model. Hasil evaluasi menunjukkan bahwa model CNN memiliki akurasi yang lebih tinggi dibandingkan model RF dengan akurasi 99.7% dan 74% secara berturut-turut. Model CNN juga memberikan nilai *precision*, *recall*, dan *f1-score* yang lebih baik dibandingkan model RF. Oleh karena itu, penggunaan metode CNN dalam klasifikasi huruf hijaiyah merupakan pilihan yang efisien dan memberikan hasil yang baik. Aplikasi klasifikasi huruf hijaiyah ini diharapkan dapat membantu pengguna dalam memahami dan mengenali huruf hijaiyah dengan lebih baik.

Daftar Pustaka

- Abdulhakim, R., Carudin, & Arif Dermawan, B. (2021). Analisis dan Penerapan Algoritma Convolutional Neural Network untuk Klasifikasi Kendaraan Prioritas. *Jurnal Sains Dan Informatika*, 7(2), 135–144. <https://doi.org/10.34128/jsi.v7i2.335>
- Abror Mufassiril, N. (2021). Pattern Recognition Tulisan Tangan Huruf Hijaiyah Menggunakan Metode Convolutional Neural Network (CNN). *Jurnal Informasi Dan Komputer*, 9(2), 6.
- 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>
- Ahmad Martani, Saripuddin M, & Nurul Ikhsan. (2022). Rancang Bangun Website Company

- Profile Berbasis Framework Bootstrap dan Framework Codeigniter Pada Yayasan Khalifah Cendekia Mandiri. *Jurnal Multidisiplin Madani*, 2(6), 2895–2912. <https://doi.org/10.55927/mudima.v2i6.510>
- 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>
- Amutha, P., & Priya, R. (2022). An optimized random forest for multi class classification to classify the students using machine learning approaches. *International Journal of Health Sciences*, 6(May), 47310–47324. <https://doi.org/10.53730/ijhs.v6ns7.13116>
- 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>
- Chahal, A., & Gulia, P. (2019). Machine learning and deep learning. *International Journal of Innovative Technology and Exploring Engineering*, 8(12), 4910–4914. <https://doi.org/10.35940/ijitee.L3550.1081219>
- 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>
- 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>
- Hasyim, F., Malik, K., & Rizal, F. (2021). Implementasi Algoritma Convolutional Neural Networks (CNN) Untuk Klasifikasi Batik. *Jurnal Kecerdasan Buatan, Komputasi Dan Teknologi Informasi*, X(X), 40–47. <https://ejournal.unuja.ac.id/index.php/core>
- Ihsan, C. N. (2021). Klasifikasi Data Radar Menggunakan Algoritma Convolutional Neural Network (CNN). *DoubleClick: Journal of Computer and Information Technology*, 4(2), 115. <https://doi.org/10.25273/doubleclick.v4i2.8188>
- Khairani, F., Kurnia, A., Aidi, M. N., & Pramana, S. (2022). Predictions of Indonesia Economic Phenomena Based on Online News Using Random Forest. *Sinkron*, 7(2), 532–540. <https://doi.org/10.33395/sinkron.v7i2.11401>
- 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.
- Nova, S. H., Widodo, A. P., & Warsito, B. (2022). Analisis Metode Agile pada Pengembangan Sistem Informasi Berbasis Website: Systematic Literature Review. *Techno.Com*, 21(1), 139–148. <https://doi.org/10.33633/tc.v21i1.5659>
- 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.
- Pamuji, F. Y., & Ramadhan, V. P. (2021). Komparasi Algoritma Random Forest dan Decision Tree untuk Memprediksi Keberhasilan Immunotherapy. *Jurnal Teknologi Dan Manajemen Informatika*, 7(1), 46–50. <https://doi.org/10.26905/jtmi.v7i1.5982>
- Raharjo, B. (2022). *Deep Learning dengan Python* (M. M. T. Dr. Mars Caroline Wibowo. S.T. (ed.)). Yayasan Prima Agus Teknik.

- Ratnawati, L., & Sulistyaningrum, D. R. (2020). Penerapan Random Forest untuk Mengukur Tingkat Keparahan Penyakit pada Daun Apel. *Jurnal Sains Dan Seni ITS*, 8(2). <https://doi.org/10.12962/j23373520.v8i2.48517>
- Santoso, J. T. (2022). *Proyek Coding dengan Python* (M. K. D. Muhammad Sholikan (ed.)). Yayasan Prima Agus Teknik.
- Schonlau, M., & Zou, R. Y. (2020). The random forest algorithm for statistical learning. *Stata Journal*, 20(1), 3–29. <https://doi.org/10.1177/1536867X20909688>
- Suhari, S., Faqih, A., & Basysyar, F. M. (2022). Sistem Informasi Kepegawaian Menggunakan Metode Agile Development di CV. Angkasa Raya. *Jurnal Teknologi Dan Informasi*, 12(1), 30–45. <https://doi.org/10.34010/jati.v12i1.6622>
- Widiastuti, N., Hermawan, A., & Avianto, D. (2023). Komparasi Algoritma Klasifikasi Datamining untuk Prediksi Minat Pencari Kerja. *Jurnal Teknoinfo*, 17(1), 219–227.
- Yamashita, R., Nishio, M., Do, G. K. R., & Togashi, K. (2021). Convolutional Neural Networks: An Overview and Its Applications in Pattern Recognition. *Smart Innovation, Systems and Technologies*, 195, 21–30. https://doi.org/10.1007/978-981-15-7078-0_3
- Yohannes, Y., Udjulawa, D., & Febbiola, F. (2021). Klasifikasi Lukisan Karya Van Gogh Menggunakan Convolutional Neural Network-Support Vector Machine. *Jurnal Teknik Informatika Dan Sistem Informasi*, 7(1), 192–205. <https://doi.org/10.28932/jutisi.v7i1.3399>