

# SiPuTiH: Model Convolutional Neural Network untuk Sistem Pengenalan Tulisan Tangan Hijaiyah

Saiful Nur Budiman <sup>1\*</sup>, Sri Lestanti <sup>2</sup>, Sandi Widya Permana <sup>3</sup>

<sup>1,2,3</sup> Program Studi Teknik Informatika, Universitas Islam Balitar, Blitar, Indonesia

[lestanti85@gmail.com](mailto:lestanti85@gmail.com) <sup>2</sup>, [sandiwpataa@gmail.com](mailto:sandiwpataa@gmail.com) <sup>3</sup>

Correspondence: [sync.saifulnb@gmail.com](mailto:sync.saifulnb@gmail.com) <sup>1</sup>

## ABSTRAK

Penelitian ini mengembangkan SiPuTiH (Sistem Pengenalan Tulisan Tangan Huruf Hijaiyah) menggunakan algoritma Convolutional Neural Network (CNN) untuk mengatasi tantangan variasi bentuk dan gaya tulisan tangan huruf Arab. Proses penelitian meliputi tahap akuisisi dan prapemrosesan dataset, perancangan arsitektur CNN, pelatihan model, dan evaluasi performa. Dataset terdiri dari 1.680 citra tulisan tangan yang mewakili 30 karakter hijaiyah, dibagi menjadi 80% data latih dan 20% data uji. Arsitektur CNN yang dirancang memiliki empat lapisan konvolusi dan pooling dengan total 6,8 juta parameter trainable. Hasil pengujian menunjukkan bahwa model SiPuTiH mencapai akurasi 99,7% dalam mengenali karakter hijaiyah, dengan satu-satunya kesalahan klasifikasi antara huruf 'ta' (ت) dan 'tsa' (ث) akibat kemiripan morfologis. Model ini kemudian diintegrasikan ke dalam aplikasi interaktif berbasis Streamlit yang menyediakan fitur pembelajaran huruf, kuis pengenalan, serta prediksi tulisan tangan secara real-time. SiPuTiH terbukti efektif tidak hanya sebagai sistem pengenalan citra, tetapi juga sebagai media edukasi digital yang interaktif dan mudah digunakan. Implementasi ini diharapkan dapat mendukung digitalisasi pembelajaran huruf hijaiyah serta menjadi dasar pengembangan sistem pengenalan teks Arab yang lebih kompleks di masa depan.

## Kata Kunci

CNN; Huruf Hijaiyah; Pengenalan Tulisan Tangan; Deep Learning; Edukasi Interaktif;

## ABSTRACT

*This research presents the development of SiPuTiH (Handwritten Hijaiyah Character Recognition System) using the Convolutional Neural Network (CNN) algorithm to address the challenges of handwriting variability in Arabic scripts. The methodology includes dataset acquisition and preprocessing, CNN architecture design, model training, and performance evaluation. The dataset consists of 1,680 handwritten images representing 30 Hijaiyah characters, divided into 80% training and 20% testing data. The proposed CNN architecture employs four convolutional and pooling layers with a total of 6.8 million trainable parameters. Experimental results show that SiPuTiH achieved a 99.7% accuracy rate in recognizing Hijaiyah characters, with only one misclassification between 'ta' (ت) and 'tsa' (ث) due to morphological similarity. The trained model was implemented in an interactive Streamlit-based application that includes learning modules, quizzes, and real-time handwriting prediction. SiPuTiH demonstrates high reliability not only as a handwriting recognition system but also as an engaging educational platform for learning Arabic letters. This study confirms the effectiveness of CNNs in handling the morphological complexity of Hijaiyah characters and contributes to the development of intelligent educational tools. Future work may explore larger datasets, transfer learning architectures, and contextual (word-level) recognition to enhance system scalability and performance.*

## Key Words

CNN; Hijaiyah Characters; Handwriting Recognition; Deep Learning; Interactive Education;

Received: 28<sup>th</sup> October 2025

Accepted: 27<sup>th</sup> November 2025

Published: 31<sup>st</sup> December 2025

Citation: -

10.46510/jami.v6i2.390

ISSN 2722-4414 (p)/ 2722-4406 (e)

84

<https://journal.akb.ac.id/>

## I. PENDAHULUAN

Pengenalan tulisan tangan (*Handwriting Recognition* – HWR) merupakan salah satu bidang riset fundamental dalam ranah ilmu komputer, khususnya dalam disiplin ilmu pengolahan citra dan kecerdasan buatan. Kemampuan sistem untuk menginterpretasi dan mengkonversi tulisan tangan manusia menjadi teks digital memiliki implikasi praktis yang sangat luas. Di era digital saat ini, HWR menjadi krusial untuk berbagai aplikasi, mulai dari digitalisasi dokumen bersejarah, formulir pendaftaran otomatis, sistem pengisian cek, hingga alat bantu belajar interaktif. Namun, kompleksitas yang melekat pada tulisan tangan manusia, yang dicirikan oleh variabilitas gaya penulisan antar individu, perbedaan ukuran dan ketebalan goresan, serta potensi adanya noise atau distorsi akibat media penulisan, telah menjadikan HWR sebagai tantangan yang signifikan (Gautam et al., 2022). Kondisi ini menuntut pengembangan metode yang tidak hanya akurat tetapi juga adaptif terhadap beragam kondisi input untuk mencapai performa yang optimal. Meskipun telah banyak kemajuan dicapai, terutama untuk aksara Latin, pengenalan tulisan tangan untuk aksara non-Latin, seperti aksara Arab atau Hijaiyah, masih menghadapi kendala yang lebih besar karena karakteristik intrinsik yang unik.

Dalam konteks pengenalan huruf Hijaiyah, tantangan yang dihadapi jauh lebih kompleks dibandingkan dengan aksara Latin. Huruf Hijaiyah memiliki struktur morfologis dan grafis yang khas; misalnya, banyak huruf yang memiliki bentuk dasar yang sama dan dibedakan oleh penempatan dan jumlah titik (*nuqath*) di atas, di bawah, atau di dalamnya (Angraheni et al., 2017). Selain itu, bentuk huruf Hijaiyah bersifat kontekstual, yang berarti bentuknya akan berubah tergantung pada posisinya dalam kata (awal, tengah, akhir, atau tunggal). Aspek kaligrafi Arab yang kaya juga menambah variasi yang signifikan, di mana satu huruf bisa dituliskan dalam berbagai gaya yang sangat artistik namun sulit dikenali oleh sistem otomatis. Semua karakteristik ini secara kolektif meningkatkan ambiguitas visual dan mempersulit proses ekstraksi fitur yang konsisten. Akibatnya, upaya digitalisasi manuskrip atau dokumen berbahasa Arab, serta pengembangan platform pembelajaran daring yang memungkinkan interaksi melalui tulisan tangan, sering kali terhambat oleh keterbatasan teknologi pengenalan yang ada, sehingga membutuhkan intervensi manual yang memakan waktu dan sumber daya.

Metode pengenalan tulisan tangan secara historis telah berkembang dari pendekatan statistik tradisional menuju pendekatan *machine learning* modern. Pada awalnya, penelitian HWR banyak menggunakan metode *feature engineering* manual, di mana fitur-fitur seperti *stroke direction*, *endpoints*, atau *geometric moments* diekstraksi secara eksplisit dari citra karakter (Mawaddah & Suciati, 2020). Metode klasifikasi seperti Support Vector Machine (SVM), K-Nearest Neighbors (KNN), atau Hidden Markov Model (HMM) kemudian digunakan untuk mengklasifikasikan fitur-fitur tersebut. Meskipun pendekatan ini cukup berhasil untuk dataset yang terkontrol, performanya cenderung menurun drastis ketika dihadapkan pada variasi tulisan tangan dunia nyata yang sangat beragam (Miftahul Amri, 2022). Tantangan terbesar adalah bagaimana merancang fitur yang *robust* dan invarian terhadap skala, rotasi, dan gaya penulisan yang bervariasi, merupakan hal yang sangat sulit dan memerlukan keahlian domain yang mendalam. Keterbatasan ini mendorong para peneliti untuk mencari paradigma baru yang mampu mengekstraksi fitur secara otomatis dan belajar dari data secara hirarkis.

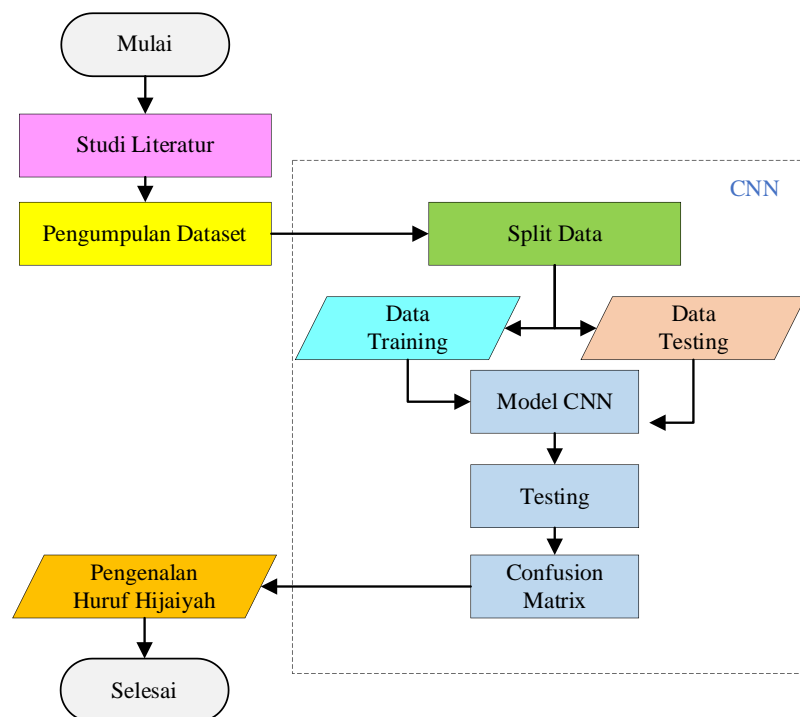
Munculnya *deep learning*, khususnya Convolutional Neural Network (CNN), telah merevolusi bidang pengolahan citra dan pengenalan pola secara fundamental (Dwiaji et al., 2024). CNN memiliki arsitektur yang terinspirasi dari sistem visual biologis, memungkinkan jaringan untuk secara otomatis mempelajari representasi fitur yang kaya dan hirarkis langsung dari data mentah (piksel citra) tanpa perlu *feature engineering* manual (Budiman et al., 2023). Lapisan konvolusi pada CNN secara efektif menangkap pola lokal, sedangkan lapisan *pooling* membantu mengurangi dimensi dan membuat representasi lebih invarian terhadap translasi kecil. Keunggulan CNN dalam mengatasi variasi visual, seperti yang sering ditemukan pada citra tulisan tangan, telah terbukti superior dalam berbagai tugas pengenalan objek, klasifikasi citra, dan deteksi wajah (Lia Farokhah, 2021). Efektivitas ini menjadikan CNN sebagai kandidat yang sangat menjanjikan untuk mengatasi kompleksitas pengenalan tulisan tangan (Handoko et al., 2024), termasuk aksara yang memiliki karakteristik unik seperti huruf Hijaiyah. Kemampuannya untuk belajar dari data dalam skala besar dan menemukan pola yang rumit telah membuka peluang baru untuk mencapai tingkat akurasi yang sebelumnya sulit dicapai.

Melihat potensi besar CNN dan urgensi kebutuhan akan sistem pengenalan tulisan tangan huruf Hijaiyah yang akurat, penelitian ini mengusulkan pengembangan Sistem Pengenalan Tulisan Tangan Hijaiyah (SiPuTiH) yang secara khusus memanfaatkan algoritma CNN. SiPuTiH dirancang untuk menyediakan solusi yang efisien dan akurat untuk identifikasi karakter huruf hijaiyah dari citra tulisan tangan. Sistem ini akan melalui serangkaian tahapan, dimulai dari akuisisi citra, prapemrosesan data (seperti normalisasi, binerisasi, dan segmentasi karakter), hingga tahap inti yaitu pelatihan dan evaluasi model CNN. Prapemrosesan yang efektif akan sangat penting untuk mempersiapkan data input agar sesuai dengan format yang dibutuhkan oleh CNN dan untuk mengurangi *noise* yang dapat mengganggu proses pembelajaran (Swasono et al., 2024). Pemanfaatan dataset tulisan tangan huruf Hijaiyah yang beragam dan representatif akan menjadi kunci untuk melatih model CNN agar mampu melakukan generalisasi dengan baik terhadap variasi tulisan tangan yang berbeda di dunia nyata.

## II. MATERIAL DAN METODE

Mengadopsi pendekatan kuantitatif dengan desain eksperimental untuk mengembangkan dan mengevaluasi Sistem Pengenalan Tulisan Tangan Hijaiyah (SiPuTiH) menggunakan Convolutional Neural Network (CNN). Metode ini dipilih karena memungkinkan pengujian hipotesis secara sistematis mengenai efektivitas CNN dalam klasifikasi citra tulisan tangan yang kompleks dan bervariasi. Tahapan penelitian akan meliputi akuisisi data, prapemrosesan data, desain dan implementasi arsitektur CNN, pelatihan model, serta evaluasi kinerja sistem. Seluruh proses akan dilakukan untuk memastikan replikabilitas dan validitas hasil. Fokus utama adalah pada pengukuran akurasi pengenalan karakter tulisan tangan huruf Hijaiyah, serta analisis terhadap faktor-faktor yang memengaruhi kinerja model, seperti variasi gaya penulisan, kualitas citra, dan kompleksitas arsitektur jaringan. Pendekatan ini akan memberikan landasan ilmiah yang kuat untuk membuktikan kapabilitas CNN dalam domain pengenalan tulisan tangan aksara non-Latin.

Pengenalan tulisan tangan huruf Hijaiyah menggunakan Convolutional Neural Network (CNN) dapat divisualisasikan melalui diagram alir penelitian yang terstruktur ditunjukkan pada Gambar 1. Tahap awal penelitian ini dimulai dengan studi literatur komprehensif. Pada fase ini, peneliti mengidentifikasi dan menganalisis berbagai riset terdahulu terkait pengenalan tulisan tangan, khususnya untuk aksara non-Latin seperti Hijaiyah, serta eksplorasi mendalam terhadap arsitektur dan kinerja model CNN dalam tugas klasifikasi citra. Studi literatur juga mencakup peninjauan metode pengumpulan dan prapemrosesan dataset tulisan tangan, teknik split data, serta evaluasi metrik kinerja model. Pemahaman mendalam dari studi literatur tersebut menjadi landasan kuat untuk perancangan sistem dan pemilihan parameter yang optimal pada tahap-tahap selanjutnya. Setelah studi literatur, langkah krusial berikutnya adalah pengumpulan dataset huruf Hijaiyah tulisan tangan. Dataset ini harus representatif, mencakup variasi gaya tulisan dari berbagai individu untuk memastikan generalisasi model yang baik. Setelah dataset terkumpul, dilakukan split data menjadi dua bagian utama: data training dan data testing. Pada penelitian ini, proporsi split yang digunakan adalah 80% untuk data training dan 20% untuk data testing, memastikan bahwa model dilatih pada sebagian besar data dan dievaluasi pada data yang belum pernah dilihat sebelumnya.

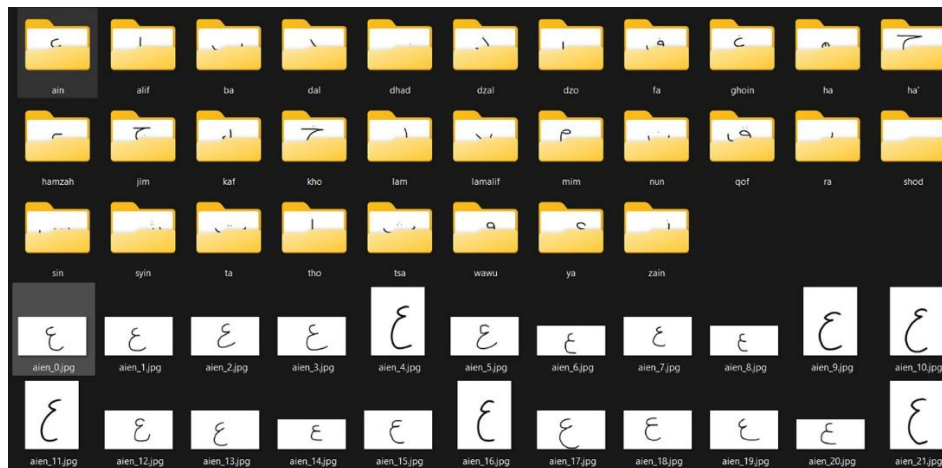


Gambar 1. Diagram Alir Penelitian

Selanjutnya, pembentukan model CNN menjadi inti dari penelitian ini. Arsitektur CNN yang dipilih terdiri dari lapisan konvolusi, aktivasi (misalnya ReLU), *max pooling*, dan lapisan *fully connected* (Dense) diakhiri dengan lapisan *softmax* untuk klasifikasi multi-kelas. Proses training melibatkan *forward propagation* dan *backpropagation* untuk memperbarui bobot model berdasarkan fungsi *loss* yang dipilih. Setelah model dilatih, tahap testing dilakukan menggunakan data testing yang terpisah. Pada tahap ini, model memprediksi kelas huruf Hijaiyah untuk setiap citra dalam dataset testing. Kinerja model kemudian dievaluasi secara kuantitatif, khususnya melalui akurasi pada confusion matrix. Confusion matrix memberikan gambaran detail mengenai kinerja klasifikasi model, menunjukkan jumlah prediksi benar dan salah untuk setiap kelas huruf Hijaiyah. Dari confusion matrix ini, dapat dihitung metrik-metrik lain seperti presisi, *recall*, dan F1-score, yang secara kolektif memberikan evaluasi komprehensif terhadap efektivitas model CNN dalam mengenali tulisan tangan huruf Hijaiyah.

## 2.1 Pengumpulan dan Pra-pemrosesan Data

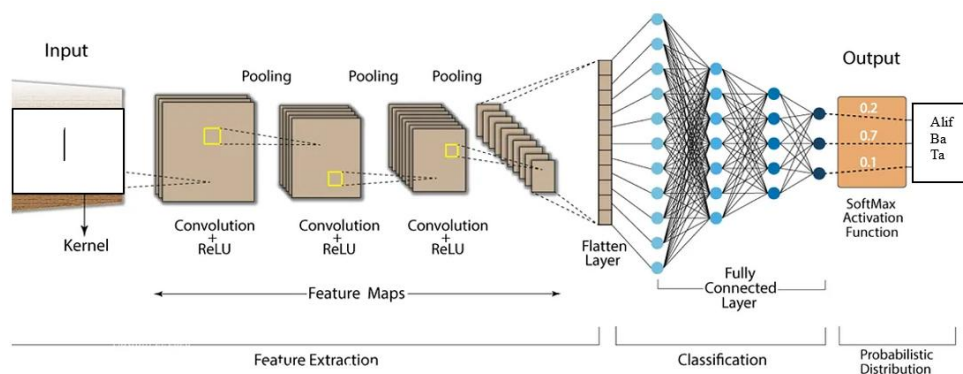
Tahap awal yang krusial dalam penelitian ini adalah pengumpulan dan prapemrosesan dataset citra tulisan tangan huruf Hijaiyah. Dataset akan dikumpulkan dari berbagai sumber, termasuk tulisan tangan asli dari responden yang berbeda (misalnya, siswa, guru, kaligrafer) untuk memastikan variasi gaya penulisan, usia, dan jenis kelamin. Setiap karakter huruf Hijaiyah, termasuk bentuk tunggal, awal, tengah, dan akhir, akan dikumpulkan secara terpisah. Pada Gambar 2 ditunjukkan contoh dari dataset yang digunakan di penelitian ini. Setelah akuisisi, citra akan melalui serangkaian proses prapemrosesan. Normalisasi ukuran diterapkan untuk menyeragamkan dimensi setiap citra karakter ke ukuran standar  $64 \times 64$  piksel guna memudahkan input ke CNN dan memastikan konsistensi. Selanjutnya binerisasi akan dilakukan untuk mengubah citra menjadi format hitam-putih, memisahkan *foreground* (tulisan) dari *background*. Segmentasi karakter otomatis atau semi-otomatis diterapkan untuk memastikan setiap citra hanya mengandung satu karakter huruf Hijaiyah yang ingin dikenali. Terakhir, teknik augmentasi data seperti rotasi kecil, translasi, scaling, dan shearing akan digunakan untuk memperkaya dataset dan meningkatkan *robustness* model terhadap variasi input.



Gambar 2. Contoh Dataset Hijaiyah

## 2.2 Desain Arsitektur CNN dan Pelatihan Model

Bagian inti dari penelitian ini adalah desain dan implementasi arsitektur Convolutional Neural Network (CNN). Beberapa arsitektur CNN yang populer dan terbukti efektif dalam tugas klasifikasi citra, seperti LeNet-5, AlexNet, VGG, atau ResNet, akan menjadi referensi awal. Eksperimen akan dilakukan untuk mengidentifikasi arsitektur paling optimal yang dapat menangani karakteristik unik huruf Hijaiyah. Arsitektur yang diusulkan akan terdiri dari beberapa lapisan konvolusi yang diikuti oleh lapisan *max pooling* untuk mengekstraksi fitur hirarkis dan mereduksi dimensi (Willyanto et al., 2021). Setelah lapisan konvolusi, akan terdapat satu atau lebih lapisan *fully connected* yang berfungsi sebagai klasifikator. Fungsi aktivasi ReLU (*Rectified Linear Unit*) digunakan pada lapisan konvolusi dan *fully connected* untuk mempercepat konvergensi dan mengatasi masalah *vanishing gradient*. Proses pelatihan model akan dilakukan menggunakan algoritma Adam optimizer dengan fungsi loss categorical cross-entropy, mengingat ini adalah masalah klasifikasi multi-kelas. Dataset akan dibagi menjadi data pelatihan, validasi, dan pengujian. *Hyperparameter* seperti *learning rate*, *batch size*, dan jumlah *epoch* akan diatur dan dioptimalkan melalui proses validasi silang (misalnya, *k-fold cross-validation*) atau pencarian *grid/random search* untuk menemukan konfigurasi yang menghasilkan kinerja terbaik. Bentuk dari arsitektur CNN yang digunakan pada penelitian ini ditunjukkan pada Gambar 3.



Gambar 3. Arsitektur CNN

### 2.3 Evaluasi Kinerja Sistem

Evaluasi kinerja SiPuTiH menjadi tahap krusial untuk mengukur efektivitas model CNN yang telah dibangun. Pengujian dilakukan pada dataset terpisah yang tidak digunakan selama pelatihan atau validasi untuk memastikan kemampuan generalisasi model. Metrik evaluasi utama yang akan digunakan meliputi (Gomes et al., 2023):

- Akurasi (*Accuracy*), yaitu proporsi prediksi benar dari total prediksi
- Presisi (*Precision*), yaitu rasio prediksi positif benar terhadap total prediksi positif
- Recall (*Sensitivitas*), yaitu rasio prediksi positif benar terhadap total kejadian positif aktual
- F1-Score, yang merupakan rata-rata harmonik dari presisi dan recall.

Selain itu, Confusion Maxtrix seperti yang ditunjukkan pada Gambar 4, dibangun untuk menganalisis kesalahan klasifikasi dan mengidentifikasi huruf-huruf Hijaiyah yang paling sering salah dikenali. Kesalahan Tipe I terjadi ketika model memprediksi sebuah observasi sebagai kelas Positif, padahal kondisi aktual dari observasi tersebut adalah Negatif. Kesalahan Tipe II merupakan kondisi sebaliknya, di mana model memprediksi sebuah observasi sebagai kelas Negatif, padahal kondisi aktual dari observasi tersebut adalah Positif. Analisis ini memberikan wawasan mendalam mengenai kekuatan dan kelemahan sistem, serta mengidentifikasi area yang memerlukan perbaikan lebih lanjut.

		Actual Values	
		1 (Postive)	0 (Negative)
Predicted Values	1 (Postive)	<b>TP</b> (True Positive)	<b>FP</b> (False Positive) <i>Type I Error</i>
	0 (Negative)	<b>FN</b> (False Negative) <i>Type II Error</i>	<b>TN</b> (True Negative)

Gambar 4. Confusion Matrix

## III. HASIL

### 3.1 Desain Arsitektur dan Implementasi Model

Implementasi model dilakukan dengan menggunakan *training* bahasa Python dan *library* TensorFlow/Keras. Output log dari proses tersebut memberikan data empiris mengenai arsitektur final dan dataset yang digunakan di penelitian ini. Berdasarkan log eksekusi, sistem berhasil mengidentifikasi 1680 *files* dan 30 *class*. Angka ini merupakan total data citra yang digunakan. Jumlah 30 kelas ini merupakan penyesuaian dari rencana awal di proposal, yang kini mencakup pemisahan antara huruf 'ha' (هـ) dan 'ha' (ح) sebagai kelas yang berbeda, serta kelas lainnya. Daftar lengkap 30 kelas yang berhasil dimuat adalah: ['ain', 'alif', 'ba', 'dal', 'dhad', 'dhal', 'dzo', 'fa', 'ghoin', 'ha', 'ha"', 'hamzah', 'jim', 'kaf', 'kho', 'lam', 'lamalif', 'mim', 'nun', 'qof', 'ra', 'shod', 'sin', 'syin', 'ta', 'tho', 'tsa', 'wawu', 'ya', 'zain']. Sesuai dengan metodologi, data ini dibagi untuk validasi, di mana log mencatat "Using 336 files for validation." Ini mengkonfirmasi alokasi 20% (336 dari 1680 file) untuk *validation split*, menyisakan 1.344 file untuk proses *training* yang sebenarnya.

Model *Convolutional Neural Network* (CNN) yang digunakan dalam penelitian ini dirancang dengan arsitektur berlapis (*multi-layered*) menggunakan pendekatan sekuensial pada framework TensorFlow–Keras. Arsitektur ini ditunjukkan pada Gambar 5, terdiri atas tiga blok utama konvolusi dan *pooling*, diikuti oleh lapisan flatten dan dense untuk klasifikasi akhir. Desain ini bertujuan untuk mengekstraksi fitur hierarkis dari citra tulisan tangan huruf hijaiyah secara bertahap, mulai dari fitur sederhana seperti tepi dan kurva, hingga fitur kompleks seperti bentuk khas setiap karakter. Secara umum, struktur model CNN SiPuTiH dapat dijelaskan sebagai berikut:

a) Lapisan Konvolusi Pertama (Conv2D)

Lapisan awal memiliki 32 filter berukuran kernel 3×3 dengan fungsi aktivasi ReLU. Lapisan ini berperan dalam mendeteksi fitur dasar seperti garis dan tepi dari citra input berukuran 150×150 piksel. Output shape yang dihasilkan adalah (148, 148, 32), menandakan terbentuknya 32 peta fitur.

- b) Lapisan MaxPooling2D Pertama  
Pooling dilakukan dengan ukuran kernel 2×2 untuk mereduksi dimensi spasial citra menjadi (74, 74, 32). Tahap ini membantu mengurangi kompleksitas komputasi dan mencegah overfitting dengan mempertahankan fitur penting saja.
- c) Lapisan Konvolusi Kedua (Conv2D\_1)  
Pada tahap ini digunakan 64 filter 3×3, menghasilkan output (72, 72, 64). Lapisan ini memperdalam pemahaman jaringan terhadap pola menengah seperti kurva dan titik yang menjadi ciri khas huruf hijaiyah.
- d) Lapisan MaxPooling2D Kedua  
Pooling kedua mereduksi ukuran fitur menjadi (36, 36, 64), menurunkan jumlah piksel namun mempertahankan esensi pola visual.
- e) Lapisan Konvolusi Ketiga (Conv2D\_2)  
Lapisan ini memperluas jumlah filter menjadi 128 dengan ukuran kernel 3×3, menghasilkan keluaran (34, 34, 128). Tujuannya adalah menangkap pola kompleks seperti kombinasi goresan huruf dan perbedaan nuqath (titik di atas/bawah huruf).
- f) Lapisan MaxPooling2D Ketiga  
Mereduksi fitur menjadi (17, 17, 128), menjaga keseimbangan antara detail fitur dan efisiensi komputasi.
- g) Lapisan Konvolusi Keempat (Conv2D\_3)  
Lapisan terdalam menggunakan 256 filter 3×3 untuk menghasilkan keluaran (15, 15, 256). Tahapan ini mengekstraksi fitur abstrak tingkat tinggi yang membantu jaringan membedakan huruf-huruf yang sangat mirip secara visual seperti ba, ta, dan tsa.
- h) Lapisan MaxPooling2D Keempat  
Pooling terakhir menghasilkan peta fitur berukuran (7, 7, 256) yang mewakili bentuk paling ringkas dari karakter tulisan tangan.
- i) Lapisan Flatten  
Fitur dua dimensi hasil ekstraksi diubah menjadi satu dimensi sebanyak 12.544 neuron untuk dapat diolah oleh lapisan klasifikasi berikutnya.
- j) Lapisan Fully Connected (Dense)  
Lapisan dense pertama terdiri dari 512 neuron dengan aktivasi ReLU, berfungsi menggabungkan fitur yang telah diekstraksi dan memperkuat generalisasi pola.
- k) Lapisan Output (Dense\_1)  
Lapisan terakhir memiliki 30 neuron dengan fungsi aktivasi softmax yang mewakili 29 huruf hijaiyah tunggal ditambah satu kelas khusus untuk karakter gabungan (lam-alif). Fungsi softmax digunakan untuk mengubah skor logit menjadi probabilitas klasifikasi.

--- Model Summary (Log Arsitektur) ---  
Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 128)	73,856
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 128)	0
conv2d_3 (Conv2D)	(None, 15, 15, 256)	295,168
max_pooling2d_3 (MaxPooling2D)	(None, 7, 7, 256)	0
flatten (Flatten)	(None, 12544)	0
dense (Dense)	(None, 512)	6,423,040
dense_1 (Dense)	(None, 30)	15,390

Total params: 6,826,846 (26.04 MB)  
Trainable params: 6,826,846 (26.04 MB)  
Non-trainable params: 0 (0.00 B)

Gambar 5. Model Arsitektur CNN

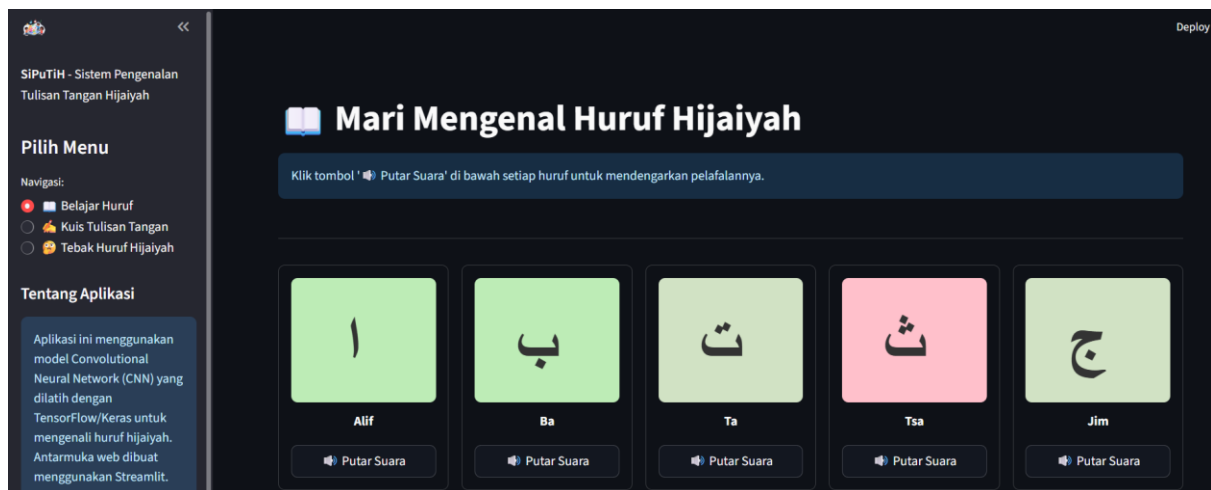


Total parameter model berjumlah 6.820.846 parameter trainable, dengan ukuran memori sekitar 26,04 MB, menunjukkan bahwa model ini memiliki kapasitas pembelajaran yang cukup besar untuk mengenali kompleksitas bentuk huruf hijaiyah tanpa kehilangan efisiensi komputasi. Arsitektur ini diimplementasikan menggunakan TensorFlow 2.x dengan optimizer Adam, fungsi *loss categorical\_crossentropy*, dan batch size sebesar 32. Proses pelatihan dilakukan selama 50 epoch dengan pembagian data training dan testing sebesar 80:20. Implementasi model ini kemudian diintegrasikan ke dalam aplikasi edukatif interaktif berbasis Streamlit, yang memungkinkan pengguna melakukan prediksi tulisan tangan secara langsung melalui kanvas digital dan mendengarkan pelafalan huruf melalui *Google Text-to-Speech* (gTTS).

### 3.2 Implementasi Sistem Aplikasi (SiPuTiH)

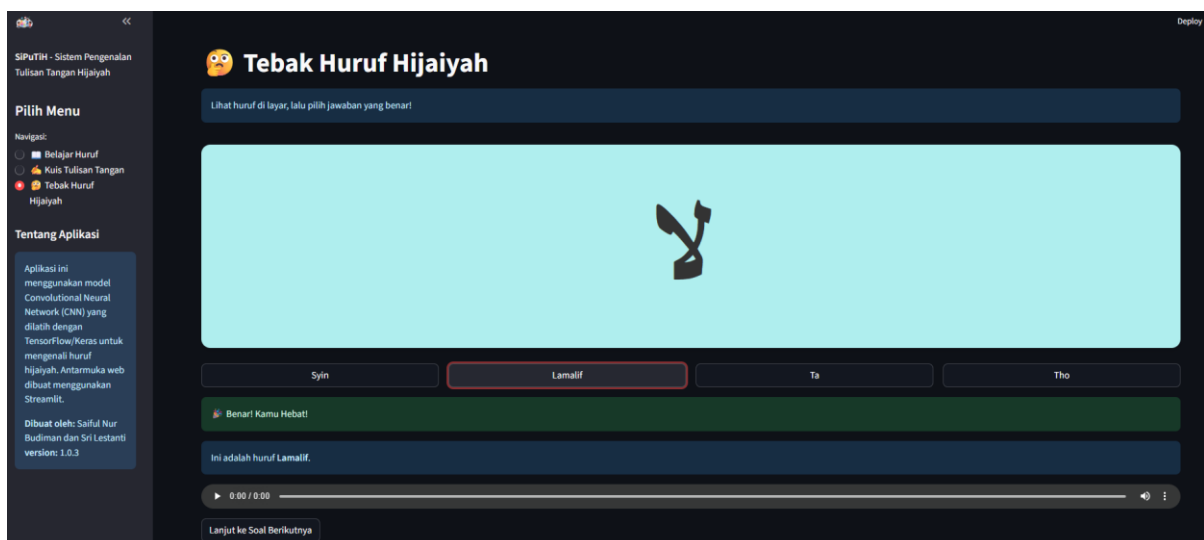
Hasil penelitian tidak hanya berupa model .h5, tetapi juga sebuah aplikasi fungsional yang disebut SiPuTiH. Aplikasi ini dibangun menggunakan framework Streamlit, seperti yang terlihat pada app.py, dan dirancang sebagai alat bantu edukasi interaktif. Aplikasi ini memiliki tiga menu utama:

1. Menu Belajar Huruf ditunjukkan pada Gambar 6. Fitur ini berfungsi sebagai kamus digital interaktif. Halaman ini menampilkan ke-30 huruf hijaiyah dalam format kartu. Setiap kartu menampilkan bentuk huruf Arab dan nama latinnya (misal, 'a' untuk 'alif'). Fitur utama di sini adalah tombol "Putar Suara" yang memanfaatkan *Google Text-to-Speech* (gTTS) untuk menghasilkan dan memutar pelafalan audio (Bahasa Arab) dari huruf tersebut. Ini membantu pengguna dalam pengenalan visual sekaligus auditori.



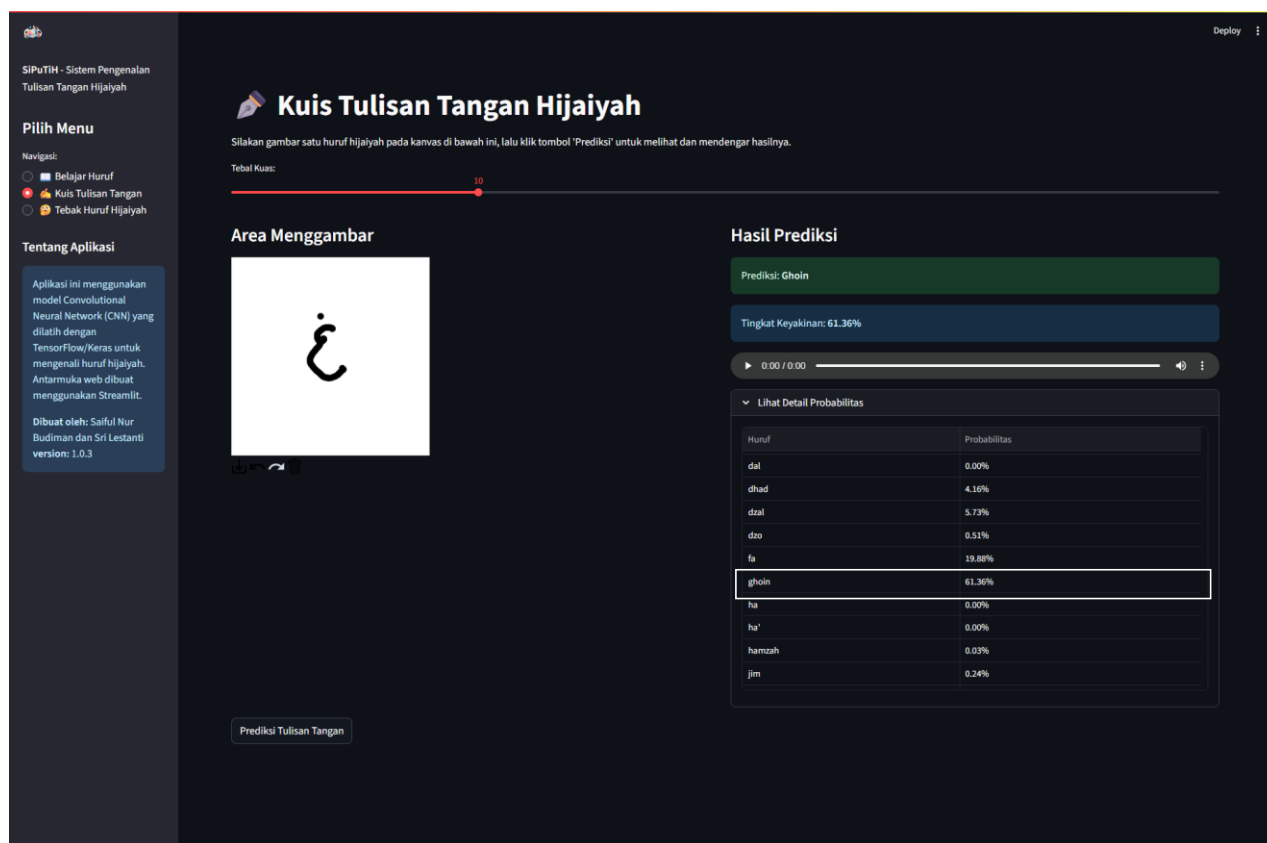
Gambar 6. Menu Belajar Huruf

2. Menu Tebak Huruf Hijaiyah ditunjukkan pada Gambar 7. Menu ini adalah kuis gamifikasi sederhana untuk menguji pengenalan visual. Sistem akan menampilkan satu huruf hijaiyah secara acak. Pengguna kemudian disajikan dengan empat pilihan jawaban (nama latin huruf), di mana hanya satu yang benar. Ini memperkuat proses belajar melalui pengujian aktif.



Gambar 7. Tebak Huruf Hijaiyah

3. Menu Kuis Tulisan Tangan ditunjukkan pada Gambar 8. Ini adalah fitur inti yang mengimplementasikan model CNN (hijaiyah\_model.h5). Halaman ini menyediakan kanvas digital (st\_canvas) berukuran 300x300 piksel tempat pengguna dapat menggambar huruf hijaiyah menggunakan mouse atau layar sentuh.



Gambar 8. Prediksi Tulisan Tangan Hijaiyah

Gambar 8 mengilustrasikan proses inferensi *end-to-end* dari sistem yang diusulkan. Dalam skenario ini, seorang pengguna telah menuliskan sebuah karakter pada "Area Menggambar" digital (st\_canvas). Karakter input ini secara visual dapat diidentifikasi sebagai huruf 'Gho' (غ). Ketika pengguna mengeksekusi tombol "Prediksi Tulisan Tangan", serangkaian proses backend dipicu:

1. Akuisisi Input: Citra dari kanvas, dalam format array RGBA, diambil sebagai input mentah.
2. Preprocessing: Citra input segera melalui pipeline preprocessing yang identik dengan yang digunakan selama pelatihan model. Proses ini mencakup konversi ke format RGB, pengubahan ukuran (resize) ke dimensi input model (150x150 piksel), dan normalisasi nilai piksel (pembagian dengan 255.0) untuk menghasilkan tensor yang valid.
3. Inferensi Model: Tensor yang telah diproses diumpankan ke model hijaiyah\_model.h5 yang telah dimuat, yang kemudian melakukan forward pass melalui arsitektur CNN (dijelaskan di 3.1).
4. Generasi Output: Model mengembalikan sebuah vektor probabilitas dari lapisan SoftMax, yang merepresentasikan tingkat keyakinan model untuk setiap 30 kelas hijaiyah.

Pada bagian "Hasil Prediksi" (Gambar 8), sistem menampilkan bahwa model dengan benar mengklasifikasikan citra input sebagai "Prediksi: Ghoin", yang merupakan sebuah True Positive. Skor keyakinan (*confidence score*) untuk prediksi ini adalah 61.36%. Selain menampilkan hasil visual, aplikasi juga menyediakan umpan balik auditori melalui pemutaran audio pelafalan huruf tersebut, yang memperkuat aspek edukasi dari sistem. Analisis yang lebih mendalam dapat ditinjau pada bagian "Lihat Detail Probabilitas". Bagian ini mengekspos distribusi probabilitas penuh dari lapisan output model, yang memberikan wawasan kritis terhadap "proses berpikir" model:

- Prediksi Utama: 'ghoin' (غ) menerima probabilitas tertinggi (61.36%), yang menunjukkan klasifikasi yang benar.
- Prediksi Alternatif: Sangat relevan secara ilmiah, model memberikan probabilitas signifikan kedua kepada huruf 'fa' (ف) sebesar 19.86%.

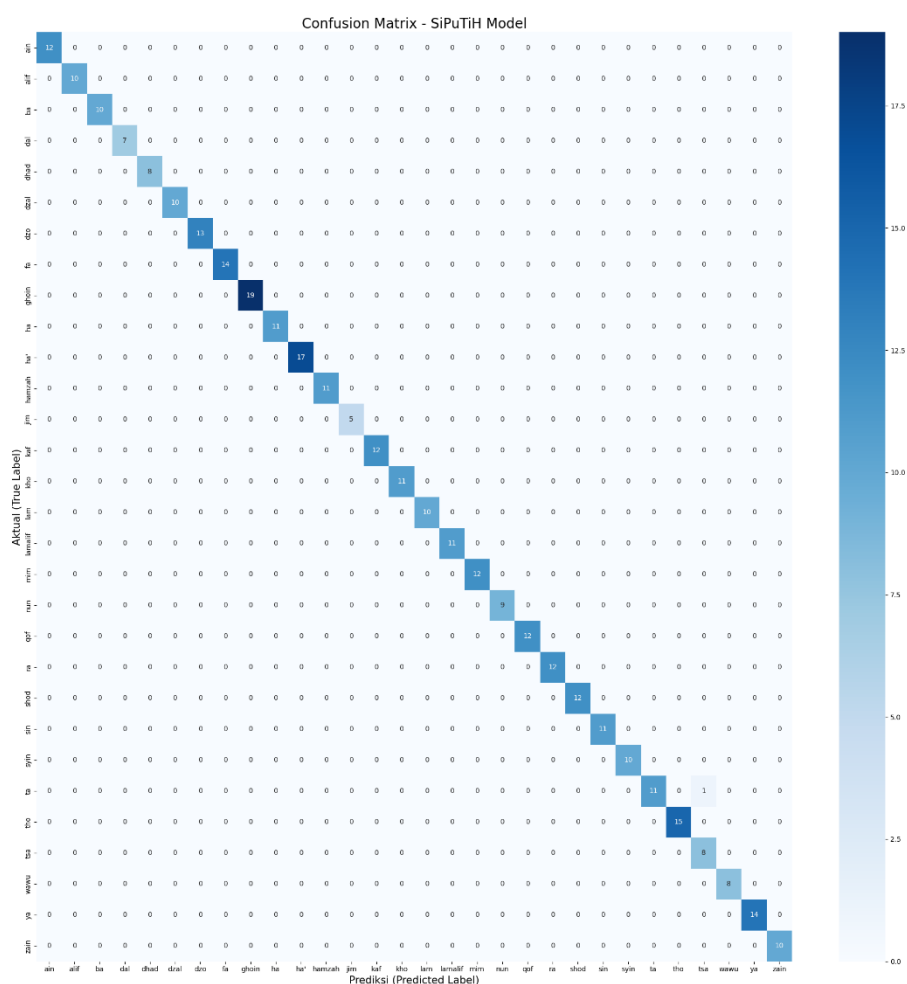


- Prediksi Lain: Huruf-huruf lain yang secara morfologis tidak mirip, seperti 'dal' (0.00%) dan 'jim' (0.24%), menerima probabilitas yang dapat diabaikan.

Distribusi probabilitas ini bukan sebuah kelemahan, melainkan bukti bahwa *convolutional base* dari model telah berhasil mempelajari fitur-fitur morfologis yang relevan. Model mengidentifikasi kemiripan struktural antara 'Ghoín' (غ) dan 'Fa' (ف), yang keduanya memiliki struktur "kepala" bulat dan satu nuqath (titik) di atas. Fakta bahwa model mempertimbangkan 'Fa' sebagai alternatif kedua yang paling mungkin dan bukan huruf acak lainnya menunjukkan bahwa model tidak sekadar menghafal (*overfitting*), melainkan telah berhasil melakukan generalisasi dan ekstraksi fitur yang bermakna.

### 3.3 Evaluasi Kinerja Model (Performa SiPuTiH)

Secara visual, Gambar 8 menunjukkan diagonal utama (dari kiri atas ke kanan bawah) yang solid dan nyaris sempurna. Setiap kotak di diagonal ini merepresentasikan True Positives (TP), atau prediksi yang benar (misalnya, 'alif' diprediksi sebagai 'alif'). Hal yang paling signifikan adalah hampir tidak adanya nilai di luar diagonal utama. Hanya terdapat satu sel di luar diagonal yang bernilai '1', yang menunjukkan satu-satunya titik kebingungan model. Sel ini terletak pada baris 'ta' (Aktual) dan kolom 'tsa' (Prediksi), yang secara visual mengonfirmasi bahwa model salah mengklasifikasikan satu huruf 'ta' sebagai 'tsa'. Hal ini juga bisa dilihat secara tabel kebenaran pada Tabel 1b.



**Gambar 9. Confusion Matrix Hijaiyah**

Model SiPuTiH mencapai akurasi keseluruhan sebesar 0.997 atau 99.7%. Angka ini didapat dari 336 total sampel data uji (support). Akurasi yang sangat tinggi ini mengindikasikan bahwa model berhasil mengklasifikasikan 334 dari 336 sampel dengan benar, dan hanya membuat 2 kesalahan. Lebih lanjut, nilai Weighted Avg (rata-rata terbobot) untuk F1-Score adalah 0.997047 (99.7%). Nilai yang nyaris identik dengan akurasi ini menunjukkan bahwa performa model sangat seimbang dan konsisten di seluruh 30 kelas, tanpa bias terhadap kelas tertentu. Detail keseluruhan hasil klasifikasinya ditunjukkan pada Tabel 1a. Untuk membedah satu-satunya kesalahan yang terlihat pada Gambar 9, dapat dirujuk pada Tabel 1b. Tabel ini menghitung secara pasti nilai True Positive (TP), False Positive (FP), False Negative (FN), dan True Negative (TN) untuk setiap kelas. Fokus analisis pada dua kelas yang diidentifikasi memiliki kesalahan pada tabel 1b tersebut yaitu 'ta' dan 'tsa'.

Kelas 'ta' memiliki TP = 11 dan FN = 1. Ini berarti, dari total 12 sampel 'ta' (11 TP + 1 FN), model berhasil mengenali 11 sampel dengan benar. Namun, model gagal mengenali (FN) 1 (satu) sampel 'ta'. Kesalahan Tipe II (*False Negative*) ini adalah penyebab nilai *Recall* (kemampuan menemukan) untuk 'ta' turun menjadi 0.916667.

Kelas 'tsa' memiliki TP = 8 dan FP = 1. Ini berarti, model memprediksi 'tsa' sebanyak 9 kali (8 TP + 1 FP). Dari 9 prediksi tersebut, 8 benar, namun 1 prediksi adalah *False Positive* (FP). Kesalahan Tipe I (*False Positive*) ini adalah penyebab nilai *Precision* (ketepatan prediksi) untuk 'tsa' turun menjadi 0.888889.

Ketika kedua analisis ini digabungkan, sebuah kesimpulan yang sangat jelas muncul. False Negative (FN) pada kelas 'ta' dan False Positive (FP) pada kelas 'tsa' bukanlah dua kesalahan yang berbeda, keduanya adalah satu kejadian yang sama. Secara spesifik, dari 336 sampel data uji, model CNN hanya membuat satu kesalahan tunggal: yaitu salah mengklasifikasikan satu citra 'ta' (ت) sebagai 'tsa' (ث). Kesalahan ini dapat dijelaskan secara ilmiah dan sesuai dengan tantangan utama penelitian ini, yaitu kemiripan morfologis (*nuqath*). Model mengalami kebingungan antara huruf 'ta' (dua titik) dan 'tsa' (tiga titik). Fakta bahwa akurasi 99.7% dengan hanya satu kesalahan yang dapat diprediksi secara logis, menegaskan bahwa arsitektur CNN yang diusulkan sangat efektif dan memiliki daya generalisasi bagus sekali.

Tabel 1. Hasil Klasifikasi (a), Confusion Matrix (b)

	precision	recall	f1-score	support
ain	1	1	1	12
alif	1	1	1	10
ba	1	1	1	10
dal	1	1	1	7
dhad	1	1	1	8
dzal	1	1	1	10
dzo	1	1	1	13
fa	1	1	1	14
ghoin	1	1	1	19
ha	1	1	1	11
ha'	1	1	1	17
hamzah	1	1	1	11
jim	1	1	1	5
kaf	1	1	1	12
kho	1	1	1	11
lam	1	1	1	10
lamalif	1	1	1	11
mim	1	1	1	12
nun	1	1	1	9
qof	1	1	1	12
ra	1	1	1	12
shod	1	1	1	12
sin	1	1	1	11
syin	1	1	1	10
ta	1	0.916667	0.956522	12
tho	1	1	1	15
tsa	0.888889	1	0.941176	8
wawu	1	1	1	8
ya	1	1	1	14
zain	1	1	1	10
accuracy	0.997024	0.997024	0.997024	0.997024
macro avg	0.996296	0.997222	0.99659	336
weighted avg	0.997354	0.997024	0.997047	336

(a)

Kelas	TP	FP	FN	TN
ain	12	0	0	324
alif	10	0	0	326
ba	10	0	0	326
dal	7	0	0	329
dhad	8	0	0	328
dzal	10	0	0	326
dzo	13	0	0	323
fa	14	0	0	322
ghoin	19	0	0	317
ha	11	0	0	325
ha'	17	0	0	319
hamzah	11	0	0	325
jim	5	0	0	331
kaf	12	0	0	324
kho	11	0	0	325
lam	10	0	0	326
lamalif	11	0	0	325
mim	12	0	0	324
nun	9	0	0	327
qof	12	0	0	324
ra	12	0	0	324
shod	12	0	0	324
sin	11	0	0	325
syin	10	0	0	326
ta	11	0	1	324
tho	15	0	0	321
tsa	8	1	0	327
wawu	8	0	0	328
ya	14	0	0	322
zain	10	0	0	326
Overall (Sum)	335	1	1	9743

(b)

#### IV. KESIMPULAN

Penelitian ini berhasil mengembangkan SiPuTiH (Sistem Pengenalan Tulisan Tangan Huruf Hijaiyah) berbasis Convolutional Neural Network (CNN) yang mampu mengenali 30 karakter hijaiyah dengan tingkat akurasi sangat tinggi, yaitu 99,7%. Hasil ini menunjukkan bahwa arsitektur CNN yang dirancang telah efektif dalam mengekstraksi fitur morfologis kompleks huruf hijaiyah, seperti perbedaan jumlah titik (*nuqath*) dan bentuk dasar huruf yang serupa. Kesalahan klasifikasi hanya terjadi pada satu pasangan huruf yang memiliki kemiripan struktur visual tinggi, yaitu antara 'ta' (ت) dan 'tsa' (ث). Implementasi model dalam bentuk aplikasi edukatif berbasis Streamlit memperkuat aspek fungsional penelitian ini. Aplikasi SiPuTiH tidak hanya mampu melakukan prediksi tulisan tangan secara real-time, tetapi juga dilengkapi fitur pembelajaran visual, auditori (melalui *Google Text-to-Speech*), dan kuis interaktif. Hal ini menjadikan sistem ini tidak sekadar alat pengenalan citra, tetapi juga sarana pembelajaran digital yang menarik dan aplikatif bagi siswa dan guru. Secara keseluruhan, penelitian ini membuktikan bahwa CNN merupakan pendekatan yang sangat efektif untuk pengenalan tulisan

tangan huruf hijaiyah. Kedepannya, pengembangan dapat diarahkan pada ekspansi dataset yang lebih besar, integrasi model *Transfer Learning* seperti VGG atau ResNet, serta pengenalan konteks kata (*word-level recognition*) untuk meningkatkan kemampuan sistem dalam membaca teks Arab tulis tangan secara menyeluruh.

## V. DAFTAR PUSTAKA

- Angraheni, N. R., Efendi, R., & Purwandari, E. P. (2017). Pengenalan Tulisan Tangan Huruf Hijaiyah Sambung Menggunakan Algoritma Template Matching Correlation. *Rekursif: Jurnal Informatika*, 5(1). <https://doi.org/https://doi.org/10.33369/rekursif.v5i1.2551>
- Budiman, S. N., Lestanti, S., & Yuana, H. (2023). *Klasifikasi Alfabet Sistem Isyarat Bahasa Indonesia (SIBI) Menggunakan Computer Vision dan Deep Learning*. Penerbit NEM.
- Dwiaji, A. Z., Junianto, B., Haswanto, S. P., & Yusnadi, M. R. (2024). Literature Review: Penggunaan Convolutional Neural Networks Untuk Klasifikasi Citra Tumor Otak. *Buletin Ilmiah Ilmu Komputer Dan Multimedia (BIKMA)*, 2(2), 491–496. <https://www.jurnalmahasiswa.com/index.php/biikma/article/view/1632>
- Gautam, R., Sinha, A., Mahmood, H. R., Singh, N., Ahmed, S., Rathore, N., Bansal, H., & Raza, M. S. (2022). Enhancing Handwritten Alphabet Prediction with Real-time IoT Sensor Integration in Machine Learning for Image. *Journal of Smart Internet of Things*, 2022(1), 53–64. <https://doi.org/10.2478/jsiot-2022-0004>
- Gomes, R., Schmith, J., Figueiredo, R., Freitas, S., Machado, G., Romanini, J., & Carrard, V. (2023). Use of Artificial Intelligence in the Classification of Elementary Oral Lesions from Clinical Images. *International Journal of Environmental Research and Public Health*, 20(5), 3894. <https://doi.org/10.3390/ijerph20053894>
- Handoko, A. A., Rosid, M. A., & Indahyanti, U. (2024). Implementasi Convolutional Neural Network (CNN) Untuk Pengenalan Tulisan Tangan Aksara Bima. *SMATIKA JURNAL*, 14(01), 96–110. <https://doi.org/10.32664/smatika.v14i01.1196>
- Lia Farokhah. (2021). Perbandingan Metode Deteksi Wajah Menggunakan OpenCV Haar Cascade, OpenCV Single Shot Multibox Detector (SSD) dan DLib CNN. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(3), 609–614. <https://doi.org/10.29207/resti.v5i3.3125>
- Mawaddah, S., & Suciati, N. (2020). Pengenalan Karakter Tulisan Tangan Menggunakan Ekstraksi Fitur Bentuk Berbasis Chain Code. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 7(4), 683–692. <https://doi.org/10.25126/jtiik.2020742022>
- Miftahul Amri, M. (2022). Studi Banding Implementasi Metode Hidden Markov Model dalam Pengenalan Tulisan Tangan. *Jurnal Genesis Indonesia*, 1(01), 42–54. <https://doi.org/10.56741/jgi.v1i01.26>
- Swasono, N. E., Himamunanto, A. R., & Budiati, H. (2024). Pengenalan Karakter Huruf Pada Gambar Tulisan Tangan Menggunakan Algoritma Convolutional Neural Network dan K-Means Clustering. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 4(4), 1646–1656. <https://doi.org/10.57152/malcom.v4i4.1451>
- Willyanto, A., Alamsyah, D., & Irsyad, H. (2021). Identifikasi Tulisan Tangan Aksara Jepang Hiragana Menggunakan Metode CNN Arsitektur VGG-16. *Algoritme Jurnal Mahasiswa Teknik Informatika*, 2(1), 1–11. <https://doi.org/https://doi.org/10.35957/algoritme.v2i1.1450>