

Perancangan Sistem Deteksi Wajah *Real-Time* Menggunakan *Convolutional Neural Network* pada Perangkat Komputer Desktop

Virgiawan Restu Pratama ^{1*}, Dani Sasmoko ², Toni Wijanarko Adi Putra ³

^{1,2} Program Studi Sistem Komputer, Fakultas Akademik, Universitas Sains dan Teknologi Komputer, Kota Semarang, Provinsi Jawa Tengah, Indonesia.

³ Program Studi Teknik Informatika, Fakultas Akademik, Universitas Sains dan Teknologi Komputer, Kota Semarang, Provinsi Jawa Tengah, Indonesia.

Email: kikomitsuhara@gmail.com ^{1*}, dani@stekom.ac.id ², t.wijanarko@gmail.com ³

Histori Artikel:

Dikirim 30 Juni 2026; Diterima dalam bentuk revisi 10 Juli 2026; Diterima 30 Juli 2026; Diterbitkan 10 September 2026. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STM IK Indonesia Banda Aceh.

Abstrak

Penelitian ini guna merancang serta mengimplementasikan sistem pendeteksi wajah real-time memakai Convolutional Neural Network (CNN) pada perangkat komputer desktop. Penelitian dilaksanakan dengan tahap pengumpulan data, pelatihan model, preprocessing, serta evaluasi kinerja sistem. Dataset yang dipakai terdiri 200 citra wajah yang diproses melalui penyesuaian ukuran citra jadi 224×224 piksel, normalisasi, kemudian pembagian data menjadi data pelatihan dan pengujian. Model dikembangkan menggunakan TensorFlow, Keras, dan OpenCV, kemudian dievaluasi memakai confusion matrix berdasarkan nilai precision, accuracy, recall, serta F1-score. Temuan penelitian yaitu model bisa mendeteksi dan mengenali wajah dengan baik, ditunjukkan oleh nilai accuracy berupa 97,22%, kemudian nilai recall, precision, serta F1-score berupa 97%. Penerapan Batch Normalization dan Dropout juga terbukti memajukan stabilitas pelatihan serta kecakapan penyamarataan model. Temuan dari penelitian ini menandakan metode CNN efektif diterapkan untuk sistem deteksi wajah real-time pada perangkat komputer desktop.

Kata Kunci: CNN; Deteksi Wajah; Real-Time; Komputer Desktop; Deep Learning.

Abstract

This research aims to design and implement a real-time face detection system use a Convolutional Neural Network (CNN) on a desktop computer. The research consisted of data preprocessing, collection, model training, and performance evaluation. The dataset contained 200 facial images, which were resized to 224×224 pixels, normalized, and divided into training and testing sets. The model was developed using TensorFlow, Keras, and OpenCV, and evaluated use confusion matrix based on precision, accuracy, recall, and F1-score. The research results is that the proposed model can detect and recognize faces effectively, as demonstrated by an accuracy value of 97.22%, along with recall, precision, and F1-score values of 97%. The implementation of Batch Normalization and Dropout improved training stability and enhanced the model's generalization capability. These findings indicate that the CNN -based approach is effective for real-time face detection on desktop computer systems.

Keyword: CNN; Face Detection; Real-Time; Desktop Computer; Deep Learning.

1. Pendahuluan

Teknologi kecerdasan buatan (AI) begitu berkembang pesat dibeberapa tahun belakangan, mengubah banyak aspek kehidupan secara drastis (Nabila *et al.*, 2021). Suatu bidang AI yang mengalami pertumbuhan signifikan ialah *computer vision*, berupa kemampuan komputer guna memahami serta menilai masukan visual berupa gambar atau video. Di antara berbagai penerapan tersebut, deteksi wajah (*face detection*) menjadi salah satu teknologi yang banyak dikembangkan karena perannya yang penting dalam memenuhi kebutuhan modern (Khana *et al.*, 2024). Deteksi wajah ialah langkah awal yang penting disistem pengenalan wajah, yang berguna menemukan maupun melokalisasi wajah manusia dalam foto ataupun video secara otomatis (Christyanto *et al.*, 2022). Teknologi ini telah banyak diterapkan pada berbagai sektor, seperti keamanan dan pengawasan, otentikasi biometrik, sistem presensi otomatis, hingga aplikasi media sosial. Kemampuan mendeteksi wajah secara cepat dan akurat menjadi faktor utama dalam mendukung aplikasi computer vision yang handal dan efisien (Husna *et al.*, 2022). Metode pendeteksi wajah tradisional misalnya Haar Cascade ataupun Histogram of Oriented Gradients (HOG) pernah banyak digunakan karena sederhana dan relatif cepat. Namun, metode klasik tersebut memiliki keterbatasan yang cukup signifikan, terutama ketika dihadapkan pada kondisi lingkungan yang kompleks seperti pencahayaan yang buruk, variasi pose wajah, latar belakang yang ramai, maupun adanya occlusion atau wajah yang sebagian tertutup (Karseno *et al.*, 2024). Hal ini menyebabkan akurasi deteksi wajah menurun drastis pada situasi dunia nyata yang tidak ideal. Maka daripada itu, dibutuhkan teknik adaptif dan robust guna memajukan performa sistem deteksi wajah (Ardiansyah *et al.*, 2025). Salah satu metode yang saat ini dianggap paling efektif dalam menyelesaikan kesulitan computer vision ialah CNN. CNN memiliki keunggulan mengambilnya fitur citra secara yang otomatis dan hierarkis tidak memerlukan proses feature engineering secara manual seperti pada teknik tradisional (Goyal *et al.*, 2022). Berbagai arsitektur CNN modern seperti MTCNN, YOLO, dan SSD telah terbukti mampu memberikan penampilan lebih bagus dihal pengakurasian maupun ketahanan terhadap variasi kondisi wajah dibandingkan metode klasik (Heri *et al.*, 2025). Hal tersebut membuat CNN selaku pendekatan utama dipengembangan sistem pendeteksi wajah masa kini (Khana *et al.*, 2024). Meskipun CNN menawarkan performa yang unggul, implementasi sistem deteksi wajah secara real-time masih menghadapi tantangan tersendiri. Sistem real-time menuntut kemampuan pemrosesan video secara cepat dengan kecepatan minimal 24–30 frame per second (fps) agar menghasilkan output yang responsif dan berjalan mulus (Hangaragi *et al.*, 2023).

Namun, model CNN yang kompleks dengan jumlah parameter besar membutuhkan komputasi yang intensif sehingga sulit dijalankan secara optimal apabila perangkat keras tidak memadai. Kondisi ini menimbulkan permasalahan trade-off antara akurasi deteksi yang tinggi dan kecepatan pemrosesan yang dibutuhkan untuk aplikasi real-time (Ardiansyah *et al.*, 2025). Komputer desktop memiliki keunggulan dibandingkan perangkat mobile maupun embedded systems dalam implementasi deteksi wajah real-time. Desktop umumnya dilengkapi prosesor yang lebih kuat, memori besar, serta dukungan GPU yang dapat mempercepat proses training dan inference model CNN (Aminullah, 2021). Dengan kemampuan tersebut, desktop berpotensi mendukung sistem deteksi wajah yang lebih kompleks dan akurat. Namun, pemanfaatannya dalam penelitian masih belum optimal karena banyak studi lebih berfokus pada pengembangan untuk perangkat mobile (Ardiansyah *et al.*, 2025). Penelitian yang dilakukan oleh Anwar (2025), menggambarkan sistem pendeteksi wajah dengan arsitektur CNN yang dikemukakan bisa sampai akurasi 96,2%. Penelitian dari Ulhaq *et al.* (2023), menunjukkan bahwa pendekatan berbasis SSD mampu mendeteksi ekspresi dari wajah secara langsung dengan waktu pemrosesan 0,51 detik dan kecepatan 31 fps, sehingga efektif digunakan pada perangkat mobile dengan komputasi rendah. Riset yang dilakukan oleh (Marinda & Amin, 2023), menunjukkan bahwa sistem deteksi masker berbasis CNN mampu mendeteksi penggunaan masker secara real-time dengan akurasi 97,5% pada jarak 1–10 meter, meskipun faktor seperti pencahayaan dan halangan objek dapat mempengaruhi kinerjanya. Program Studi Sistem Komputer di STekom hingga saat ini belum memiliki sistem monitoring kehadiran berbasis teknologi deteksi wajah. Proses pencatatan kehadiran

mahasiswa di laboratorium komputer masih dilakukan secara manual melalui tanda tangan presensi atau pencatatan oleh dosen maupun asisten laboratorium. Mekanisme tersebut berpotensi menimbulkan berbagai permasalahan, seperti terjadinya penitipan absen, kesalahan dipencatatan, maupun keterlambatan proses rekap data. Selain itu, belum adanya sistem monitoring otomatis menyebabkan pengawasan aktivitas dan kehadiran mahasiswa di laboratorium belum berjalan secara optimal dan terintegrasi secara digital. Berdasarkan kondisi tersebut, direncanakan pengembangan sistem monitoring kehadiran berbasis deteksi wajah real-time pakai CNN yang akan diimplementasikan pada perangkat komputer desktop di laboratorium. Sistem ini akan memanfaatkan kamera yang terhubung dengan komputer untuk mendeteksi wajah mahasiswa secara otomatis saat memasuki ruangan, kemudian mencatat data absensi secara langsung ke dalam basis data. Penerapan sistem tersebut diharap bisa menambah akurasi, efisiensi, serta meminimalkan potensi kecurangan dalam presensi, sekaligus mendukung transformasi digital di lingkungan Laboratorium Program Studi Sistem Komputer STekom. Selain aspek algoritmik, implementasi sistem deteksi wajah real-time juga perlu memperhatikan efisiensi penggunaan sumber daya komputasi. Sistem yang berjalan terus-menerus dapat membebani CPU dan GPU secara intensif, menyebabkan konsumsi daya tinggi, overheating, hingga penurunan performa sistem (Christyanto *et al.*, 2022). Oleh karena itu, penggunaan arsitektur ringan, pemanfaatan GPU acceleration, multi-threading, maupun teknik kompresi model menjadi faktor penting agar sistem dapat berjalan stabil dan efisien dalam jangka panjang. Sesuai uraian tersebut, penelitian ini penting diadakan dikarenakan masih terdapat berbagai permasalahan dalam proses monitoring kehadiran di lingkungan Laboratorium Komputer Program Studi Sistem Komputer STekom. Penelitian ini untuk merancang maupun mengembangkan sistem pendeteksi wajah berbasis CNN yang mampu bekerja secara akurat, stabil, dan real-time pada perangkat komputer desktop, serta mengoptimalkan kecepatan pemrosesan dan efisiensi penggunaan sumber daya. Sistem yang dikembangkan diharapkan dapat mendukung proses pencatatan kehadiran secara otomatis sehingga meningkatkan efektivitas, akurasi, dan keandalan monitoring kehadiran.

2. Metode Penelitian

2.1 Jenis Penelitian

Studi ini berupa penelitian eksperimen serta pendekatan kuantitatif yang berguna dalam menguji kinerja model CNN guna deteksi wajah menggunakan Face Recognition Dataset yang diperoleh dari Kaggle. Tahapan penelitian yaitu pengumpulan data set, proses data, pelatihan model, pengujian, serta evaluasi penampilan model berdasarkan hasil klasifikasi.

2.2 Objek Penelitian

Penelitian ini berfokus pada model CNN yang dirancang untuk deteksi wajah secara real-time pada komputer desktop. Model tersebut dilatih menggunakan dataset “Face Detection Photos and Labels” yang tersedia di platform Kaggle pada tautan <https://www.kaggle.com/datasets/trainingdatapro/face-detection-photos-and-labels>. Dataset ini diakses dan diunduh pada bulan Mei 2026 sebagai sumber data sekunder untuk penelitian. Dataset tersebut sangat tepat untuk proses pelatihan dan evaluasi model CNN dalam mengenali posisi wajah pada citra karena memuat gambar wajah yang disertai anotasi atau label. Variasi dalam hal latar belakang, pencahayaan, posisi (sudut pengambilan gambar), dan ekspresi wajah merupakan karakteristik dari 200 gambar wajah dalam dataset yang digunakan untuk penelitian ini. Variasi fitur ini menghasilkan representasi data yang lebih beragam, sehingga memungkinkan model untuk mempelajari berbagai skenario wajah yang mungkin ditemui selama proses deteksi secara *real-time*. Setiap gambar dalam dataset menjalani beberapa tahap persiapan sebagai data penelitian sebelum digunakan untuk pelatihan model. Karena berasal dari berbagai sumber, foto-foto dalam dataset awal memiliki ukuran yang beragam. Untuk menjaga konsistensi dimensi agar sesuai dengan ukuran input model CNN, semua foto awalnya diperkecil menjadi 224×224 piksel. Berguna untuk meningkatkan

stabilitas proses pelatihan model, nilai piksel distandardisasi ke rentang 0–1 selain diperkecil ukurannya. Untuk memastikan reproduksibilitas temuan penelitian, dataset dibagi menjadi 150 foto (75%) untuk pelatihan dan 50 foto (25%) untuk pengujian menggunakan pendekatan `*train-test split*` dengan ``random_state`` bernilai 42.

2.3 Teknik Pengumpulan Data

Teknik pengumpulan data dilaksanakan lewat studi literatur serta penggunaan *dataset* sekunder. Studi literatur digunakan untuk memperoleh referensi mengenai *deep learning*, deteksi wajah, CNN, *TensorFlow*, Keras, dan *OpenCV*. Data penelitian diperoleh dari *Face Recognition Dataset* yang tersedia di Kaggle, kemudian dilakukan *preprocessing* berupa transformasi dimensi citra jadi 224×224 piksel, menormalisasi nilai piksel, kemudian pembagian data menjadi data pelatihan dan pengujian.

2.4 Analisis Data

Analisis data dilakukan dengan mengevaluasi performa model CNN pakai data pengujian. Evaluasi dilakukan melalui *Confusion* Matriks beserta menaksir nilai *precision*, *accuracy*, *recall*, maupun *F1-score*. Selain itu, perkembangan angka *loss* serta *accuracy* disaat proses latihan juga dianalisis guna menilai kapabilitas model saat pelajari pola data dan melakukan generalisasi pada data yang tidak pernah digunakan.

2.5 Pengujian *Confusion Matrix*

Pengujian dilakukan menggunakan *Confusion Matrix* dalam menilai penampilan model CNN guna mengenali citra wajah. Evaluasi dilakukan sesuai metrik *precision*, *accuracy*, *recall*, maupun *F1-score*. Nilai *akurasi* dipakai saat menilai taraf akurat prediksi model secara totalitas, *precision* guna mengetahui keakuratan model dalam mengklasifikasikan data positif, *recall* dalam mengetahui kapabilitas model dalam mengenali semua data positif yang sebenarnya, sedangkan *F1-score* dipakai menilai keseimbangan diantara nilai *precision* maupun *recall*. Hasil evaluasi dari empat metrik dipakai untuk mengevaluasi penampilan serta tingkat keandalan model CNN dalam deteksi wajah.

3. Hasil dan Pembahasan

3.1 Hasil

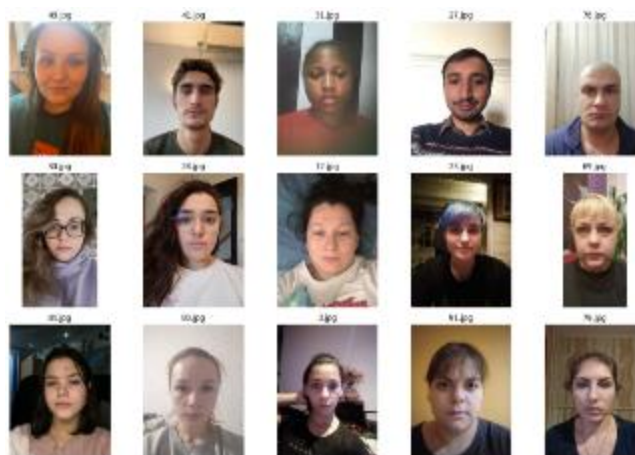
3.1.1 Deskripsi Data Penelitian

Penelitian ini dilaksanakan dengan mengimplementasikan metode *Convolutional Neural Network* atau disingkat CNN untuk deteksi dan pengenalan wajah menggunakan *Face Recognition Dataset* yang diperoleh dari platform Kaggle. Dataset terdiri atas citra wajah dengan variasi ekspresi, sudut pengambilan gambar, dan kondisi pencahayaan yang dipakai berupa data pelatihan serta pengujian model. Tahap penelitian terdiri pengumpulan data set, proses, pelatihan, serta evaluasi performa model. Proses pengolahan data dilakukan pakai bahasa pemrograman Python beserta mengoptimalkan pustaka *OpenCV* dipengolahan citra, *NumPy* untuk manipulasi data, serta *TensorFlow* dan *Keras* dalam menciptakan serta melatih model CNN. Selanjutnya, model dievaluasi berdasarkan kemampuannya dalam mendeteksi dan mengenali wajah menggunakan data pengujian.

3.1.2 Pengolahan Data

Pengolahan data diawali dengan penggunaan *Face Recognition Dataset* yang diperoleh dari Kaggle dan terdiri dari 200 citra wajah dengan variasi ekspresi, sudut pengambilan gambar, dan kondisi pencahayaan. Seluruh citra dibaca menggunakan *OpenCV*, kemudian dilakukan *preprocessing* berupa transformasi dimensi jadi 224×224 piksel kemudian normal angka piksel ke rentang 0–1 agar data punya bagian yang sama beserta bisa diproses secara optimal oleh CNN. Setelah proses *preprocessing*, data set terbagi jadi data pelatihan serta pengujian pakai fungsi `train_test_split()` serta proporsi 75%:25%,

sehingga diperoleh 150 citra sebagai data pelatihan dan 50 citra sebagai data uji. Pembagian data dilaksanakan pakai pengukuran *random state* = 42 guna memastikan temuan penelitian dapat direproduksi secara konsisten. Data pelatihan dipakai dalam melatih *CNN*, selain itu data pengujian dipakai guna mengevaluasi kapabilitas model saat mengenali wajah yang tidak pernah diajari sebelumnya. Setelah seluruh tahapan pengolahan data selesai, data siap digunakan pada proses pelatihan dan pengujian model. Gambar 1 menunjukkan beberapa sampel citra wajah yang telah melalui proses *preprocessing* dengan ukuran 224×224 piksel. Hasil tersebut memperlihatkan bahwa karakteristik penting wajah, seperti bentuk wajah, mata, hidung, dan mulut, tetap dipertahankan meskipun terdapat variasi ekspresi, posisi wajah, dan pencahayaan. Keceragaman ukuran citra ini membantu model *CNN* dalam mengekstraksi fitur wajah secara lebih konsisten sehingga dapat meningkatkan performa deteksi dan pengenalan wajah secara *real-time*.



Gambar 1. Hasil Preprocessing Data berupa Perubahan Ukuran Citra Jadi 224×224 Pixel

3.1.3 Implementasi Metode CNN

Implementasi metode *CNN* dilakukan untuk membangun model yang mampu mempelajari karakteristik wajah dari set data yang sudah lewat tahap *preprocess*. Model dikembangkan pakai TensorFlow dan Keras pada bahasa pemrograman Python dengan citra berukuran 224×224 piksel sebagai masukan. Arsitektur model terdiri atas empat lapisan *Conv2D* yang masing-masing menggunakan 32, 64, 128, dan 256 *filter*, disertai empat lapisan *MaxPooling2D* untuk mengurangi dimensi *feature map*. Setelah proses ekstraksi fitur, data diteruskan ke lapisan *GlobalAveragePooling2D* dan dua lapisan *Dense* sebelum menghasilkan prediksi koordinat *bounding box* wajah melalui lapisan keluaran dengan fungsi aktivasi *sigmoid*. Ringkasan arsitektur model ditampilkan pada Tabel 1. Model memiliki empat lapisan *Conv2D*, empat lapisan *MaxPooling2D*, satu lapisan *GlobalAveragePooling2D*, dan tiga lapisan *Dense* dengan total 488.800 *trainable parameters*. Lapisan konvolusi berfungsi mengambil fitur wajah secara berangsur, dimulai difitur sederhana sampai fitur yang lebih lengkap, selain itu bagian *pooling* bantu meminimalisir kompleksitas komputasi serta risiko *overfitting*. Struktur tersebut dirancang agar mampu memberikan keseimbangan antara kemampuan ekstraksi fitur dan efisiensi komputasi dalam proses deteksi wajah *real-time*.

Tabel 1. Ringkasan Arsitektur Model CNN

No	Layer	Output Shape	Jumlah Parameter
1	Conv2D	(None, 224, 224, 32)	320
2	MaxPooling2D	(None, 112, 112, 32)	0
3	Conv2D	(None, 112, 112, 64)	18.496
4	MaxPooling2D	(None, 56, 56, 64)	0
5	Conv2D	(None, 56, 56, 128)	73.856

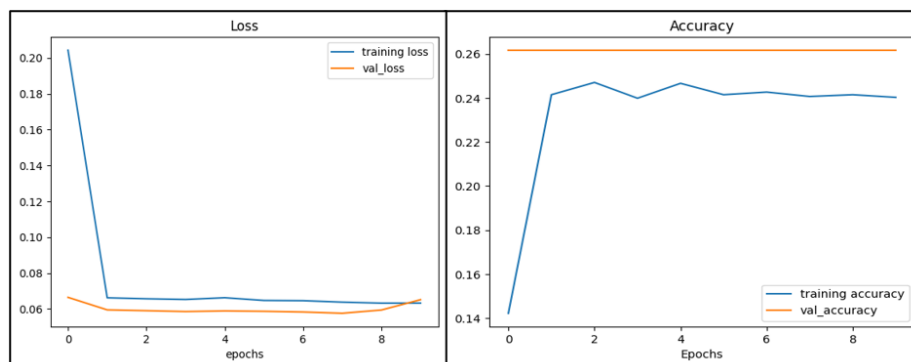
6	MaxPooling2D	(None, 28, 28, 128)	0
7	Conv2D	(None, 28, 28, 256)	295.168
8	MaxPooling2D	(None, 14, 14, 256)	0
9	GlobalAveragePooling2D	(None, 256)	0
10	Dense	(None, 256)	65.792
11	Dense	(None, 64)	16.448
12	Dense (Output)	(None, 288)	18.720
Total			488.800

Setelah model dibangun, dilakukan konfigurasi pelatihan memakai guna *optimizer Adam*, *loss binary_crossentropy*, serta metrik penilaian akurasi. Proses pelatihan dijalankan sepanjang 10 *epoch* pakai data pelatihan serta data validasi, serta dilengkapi *callback* untuk memantau hasil prediksi *bounding box* pada setiap *epoch*. Konfigurasi ini bertujuan agar proses pembelajaran berlangsung secara stabil sekaligus memudahkan pengamatan terhadap perkembangan kemampuan model selama pelatihan.

Tabel 2. Hasil Pelatihan Model CNN

<i>Epoch</i>	<i>Loss</i>	<i>Accuracy</i>	<i>Validation Loss</i>	<i>Validation Accuracy</i>
1	0,2043	0,1422	0,0664	0,2616
2	0,0662	0,2414	0,0595	0,2616
3	0,0657	0,2470	0,0590	0,2616
4	0,0653	0,2398	0,0585	0,2616
5	0,0662	0,2466	0,0589	0,2616
6	0,0647	0,2414	0,0587	0,2616
7	0,0646	0,2426	0,0583	0,2616
8	0,0638	0,2406	0,0576	0,2616
9	0,0632	0,2414	0,0594	0,2616
10	0,0633	0,2402	0,0652	0,2616

Hasil pelatihan model disajikan pada Tabel 2. Nilai *loss* mengalami penurunan 0,2043 diepoch pertama jadi 0,0633 diepoch ke-10, menandakan model semakin bisa mempelajari pola pada data pelatihan. Nilai *accuracy* meningkat dari 0,1422 menjadi sekitar 0,24, sedangkan *validation loss* tetap stabil pada kisaran 0,0576–0,0664 dengan *validation accuracy* sebesar 0,2616. Temuan tersebut mengartikan model punya performa relatif konsisten didata validasi, sehingga perkembangan nilai *loss* dan *accuracy* selanjutnya divisualisasikan dalam bentuk grafik untuk mempermudah analisis performa model semasa proses pelatihan.

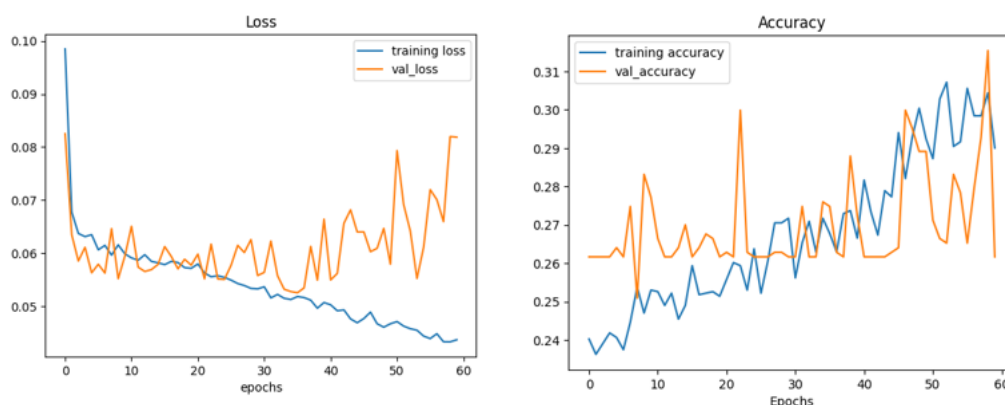


Gambar 2. Visualisasi Performa Model CNN Berdasarkan Nilai Loss dan Akurasi Data Pelatihan serta Data Validasi

Berdasarkan Gambar 2, grafik *loss* menunjukkan penurunan yang signifikan di awal proses pelatihan, dari 0,204 di *epoch* pertama jadi 0,063 di *epoch* terakhir. Sementara itu, *validation loss* cenderung stabil pada kisaran 0,057–0,065, yang mengart model punya performa konsisten saat diuji menggunakan data validasi. Keterdekatan diantara kurva *training loss* serta validasi *loss* mengindikasikan bahwa model tidak merasakan *overfitting* yang signifikansi semasa pelatihan. Grafik *accuracy* menunjukkan peningkatan nilai *training accuracy* dari sekitar 14,22% pada *epoch* pertama menjadi sekitar 24% dan kemudian relatif stabil hingga akhir pelatihan, sedangkan *validation accuracy* tetap berada pada kisaran 26,16%. Hasil tersebut menunjukkan bahwa model telah mampu mempelajari karakteristik dasar dari data wajah, meskipun tingkat akurasi masih terbatas akibat jumlah *dataset* yang relatif sedikit, yaitu 200 citra wajah. Secara keseluruhan, stabilnya nilai *loss* dan *accuracy* pada beberapa *epoch* terakhir mengartikan model sudah melampaui kondisi konvergen serta bisa mempertahankan performa yang konsisten selama pelatihan.

3.1.4 Hasil Pelatihan Model

Setelah dilakukan pengembangan arsitektur CNN dengan menambahkan lapisan *Batch Normalization* dan *Dropout*, proses pelatihan model kembali dilakukan untuk mengevaluasi pengaruh modifikasi tersebut terhadap performa model. Pelatihan dilaksanakan sepanjang 60 *epoch* beserta memakai data pelatihan serta validasi sama di model sebelumnya. Temuan pelatihan kemudian divisualisasikan dalam bentuk grafik *loss* dan *accuracy* untuk melihat perkembangan kemampuan model selama proses pembelajaran.



Gambar 3. Grafik Perkembangan *Loss* dan *Accuracy* Model CNN Setelah Penambahan *Batch Normalization* dan *Dropout*

Berdasarkan Gambar 3, nilai *training loss* menunjukkan tren penurunan yang konsisten disekitar 0,10 saat awal pelatihan jadi disekitar 0,04 di *epoch* terakhir, yang mengindikasikan model semakin bisa mengartikan pola serta karakter pada *dataset* wajah. Selain itu, *validasi loss* mengalami perubahan pada kisaran 0,05 hingga 0,08, namun selisih antara kedua kurva masih berada dalam batas yang wajar. Kondisi ini menafsirkan model punya kapabilitas general yang cukup baik serta tidak merasakan *overfitting* yang signifikansi semasa proses pelatihan. Grafik *training accuracy* menunjukkan peningkatan bertahap dari sekitar 24% menjadi lebih dari 30%, sedangkan *validation accuracy* meningkat dari kisaran 26% hingga mencapai lebih dari 31% meskipun masih mengalami sedikit fluktuasi. Peningkatan tersebut menunjukkan bahwa penambahan lapisan *Batch Normalization* dan *Dropout* mampu meningkatkan kemampuan model dalam mempelajari fitur wajah sekaligus memperbaiki performa pada data validasi. Dibandingkan model sebelumnya yang hanya mencapai *validation accuracy* sekitar 26%, model hasil pengembangan menunjukkan performa yang lebih baik sehingga modifikasi arsitektur memberikan kontribusi positif terhadap peningkatan kinerja CNN dalam mendeteksi wajah.

3.1.5 Hasil Pengujian

Sesudah proses pelatihan model berakhir dilakukan, tahap selanjutnya ialah pengujian memakai data *testing* yang berasal dari *Face Recognition Dataset*. Pengujian ini bertujuan yg guna menilai kapabilitas model CNN saat mendeteksi wajah dicitra yang tidak pernah dipakai semasa proses pelatihan. Dengan menggunakan data yang berbeda dari data pelatihan, hasil pengujian dapat memberikan gambaran mengenai kapabilitas general model dalam mengetahui wajah dikondisi yang baru. Gambar 4 menunjukkan beberapa contoh hasil pengujian model terhadap data uji. Pada setiap citra, model berhasil mendeteksi area wajah dengan memberikan *bounding box* yang mengelilingi objek wajah. Hasil tersebut menunjukkan bahwa model mampu mengidentifikasi lokasi wajah pada berbagai variasi karakteristik citra, seperti perbedaan jenis kelamin, ekspresi wajah, pencahayaan, posisi wajah, serta latar belakang gambar. Keberhasilan deteksi pada berbagai kondisi tersebut menunjukkan bahwa proses pelatihan yang telah dilakukan mampu menghasilkan model dengan kemampuan ekstraksi fitur yang baik. Secara umum, hasil pengujian menunjukkan bahwa model CNN dapat mendeteksi objek wajah secara konsisten pada sampel data uji yang digunakan dalam penelitian ini. Keberhasilan model dalam memberikan *bounding box* pada setiap citra menjadi indikasi bahwa proses deteksi telah berjalan sesuai dengan tujuan penelitian. Hasil pengujian ini selanjutnya dievaluasi lebih lanjut memakai *Confusion Matrix* guna mengetahui taraf akurasi, *recall*, *precision*, maupun *F1-score* sebagai ukuran kuantitatif terhadap performa model dalam mengklasifikasikan data.



Gambar 4. Hasil Pengujian Model CNN Data Uji

Berdasarkan proses pengujian yang telah dilakukan, ringkasan hasil deteksi wajah oleh model CNN bisa diketahui di Tabel 3.

Tabel 3. Hasil Pengujian Model CNN pada Sampel Data Uji

No	Sampel Data Uji	Epoch Model	Hasil Deteksi	Keterangan
1		60	Wajah berhasil dideteksi	Berhasil

2		60	Wajah berhasil dideteksi	Berhasil
3		60	Wajah berhasil dideteksi	Berhasil
4		60	Wajah berhasil dideteksi	Berhasil
5		60	Wajah berhasil dideteksi	Berhasil
6		60	Wajah berhasil dideteksi	Berhasil
7		60	Wajah berhasil dideteksi	Berhasil
8		60	Wajah berhasil dideteksi	Berhasil

9		60	Wajah berhasil dideteksi	Berhasil
10		60	Wajah berhasil dideteksi	Berhasil

Berdasarkan Tabel 3, model CNN berhasil mendeteksi wajah pada seluruh sampel data uji yang digunakan dalam penelitian ini. Seluruh gambar yang diuji menunjukkan hasil deteksi yang berhasil, ditandai dengan munculnya *bounding box* pada area wajah sesuai dengan objek yang terdapat pada citra. Hal tersebut menunjukkan bahwa model mampu mengenali lokasi wajah secara konsisten setelah melalui proses pelatihan sebanyak 60 *epoch*. Keberhasilan deteksi pada sepuluh sampel data uji mengindikasikan bahwa model punya kapabilitas yang baik saat melakukan ekstraksi fitur wajah diberbagai karakteristik. Variasi kondisi pada dataset, seperti perbedaan ekspresi wajah, posisi objek, pencahayaan, maupun latar belakang, tidak menghambat model dalam mendeteksi keberadaan wajah. Dengan demikian, proses pelatihan yang telah dilakukan mampu menghasilkan model yang cukup stabil dalam mengenali pola-pola penting pada citra wajah. Meskipun seluruh sampel pada pengujian awal menunjukkan hasil deteksi yang berhasil, evaluasi performa model tidak hanya didasarkan pada hasil visual tersebut. Oleh karena itu, diperlukan pengukuran secara kuantitatif pakai *Confusion Matrix* guna menilai tataraf akurasi, *recall*, *precision*, maupun *F1-score* dari model CNN. Hasil evaluasi tersebut akan memberi gambaran komprehensif tentang tingkat keberhasilan model dalam mengklasifikasikan data disetiap kelas.

3.1.6 Evaluasi Model Menggunakan *Confusion Matrix*

Tingkat keberhasilan model saat mengklasifikasi data disetiap kelas, dilakukan evaluasi menggunakan *Confusion Matrix* yang menunjukkan hubungan diantara label aktual serta label hasil dari prediksi model.

Tabel 4. Hasil *Confusion Matrix*

Parameter	Hasil
Akurasi	97,22%
<i>Precision</i>	97%
<i>Recal</i>	97%
F1-Score	97%

Berdasarkan tabel diatas, separuh besar data sukses diklasifikasi dengan benar oleh model *CNN*, yang diperliy oleh dominannya angka diagonal utama *confusion* matriks. Kelas 1, 3, 4, 5, dan 7 memiliki tingkat klasifikasi yang sangat baik karena hampir seluruh data pada kelas tersebut dikenali dengan benar. Secara keseluruhan, model memperoleh akurasi sebesar 97,22%, yang menunjukkan kemampuan sangat baik dalam mengklasifikasikan data wajah berdasarkan 360 sampel pengujian. Selain akurasi tinggi, model juga memperoleh nilai *recall*, *precision*, maupun *f1-score* rerata berupa 0,97 *dimacro average* ataupun *weighted average*. Hasil tersebut mengartikan model punya stabilitas yang baik diantara ketepatan prediksi serta kesanggupan mengenali setiap kelas. Meski masih ada sebagian

kesalahan klasifikasi, terutama pada kelas 8 dan 9 akibat kemiripan karakteristik fitur, jumlahnya relatif kecil sehingga bisa diartikan jikalau model *CNN* yang dikembangkan punya penampilan yang baik dan layak digunakan untuk deteksi serta pengenalan wajah berdasarkan *dataset* yang dipakai dipenelitian ini. Sesuai temuan penelitian, metode *CNN* mampu dipakai guna mendeteksi maupun mengenali wajah pada *dataset* yang digunakan. Tahap pengolahan data meliputi *preprocessing* berupa transformasi dimensi citra jadi 224×224 pixel, normalisasi, kemudian pembagian data jadi data pelatihan kemudian pengujian. Tahap tersebut mendapatkan data yang seragam hing bisa diproses secara baik oleh model selama proses pelatihan. Arsitektur model terdiri berbagai lapisan *MaxPooling2D*, *Conv2D*, *GlobalAveragePooling2D*, kemudian *Dense* yang berfungsi mengekstraksi fitur wajah secara bertahap. Selama proses pelatihan, nilai *loss* mengalami penurunan sedangkan nilai *accuracy* meningkat seiring bertambahnya *epoch*, yang menandakan model bisa mempelajari karakter wajah dengan baik. Temuan ini mengartikan bahwa arsitektur *CNN* yang digunakan efektif saat melakukan ekstraksi fitur dan klasifikasi wajah.

Penerapan *Batch Normalization* dan *Dropout* memberikan pengaruh positif terhadap performa model. *Batch Normalization* membantu menstabilkan proses pembelajaran melalui normalisasi keluaran pada setiap lapisan, sedangkan *Dropout* meminimalisir akibat dari *overfitting* beserta menonaktifkan sebagian neuron secara acak semasa pelatihan. Kombinasi kedua teknik tersebut menghasilkan peningkatan *accuracy* dan penurunan *loss* dibandingkan model awal. Temuan penilaian memakai *confusion* matriks serta *classification report* menandakan model mendapat akurasi sebanyak 97,22%, serta nilai *recall*, *precision*, serta *F1-score* sebesar 97%. Sebagian data berhasil diklasifikasikan dengan benar hingga menunjukkan model mampu membedakan karakteristik antarwajah secara akurat. Meski masih ada sebagian kesalahan klasifikasi dikelas tertentu, jumlahnya termasuk kecil dan tidak memengaruhi performa model secara keseluruhan. Jika dibandingkan dengan penelitian Satwikayana *et al.* (2021) dan Maulana (2025) yang memperoleh akurasi sebesar 92% pada sistem presensi model pengenalan wajah, penelitian ini menandakan performa yang jauh lebih baik dengan akurat sampai 97,22%. Peningkatan tersebut diduga dipengaruhi oleh penggunaan arsitektur *CNN* yang lebih dalam serta penerapan *Batch Normalization* dan *Dropout* yang mampu meningkatkan kemampuan generalisasi model. Di sisi lain, penelitian Christyanto *et al.* (2022) melaporkan akurasi sebesar 100% menggunakan model *FaceNet*, sedangkan Ulhaq *et al.* (2023) memperoleh *training accuracy* sebesar 74% pada deteksi bentuk wajah menggunakan *CNN*. Perbedaan hasil tersebut dapat dipengaruhi oleh variasi *dataset*, jumlah sampel, arsitektur model, dan lingkungan pengujian yang digunakan. Secara keseluruhan, hasil penelitian ini mengartikan metode *CNN* bisa memberi performa yang kompetitif dan efektif dalam mendeteksi serta mengenali wajah secara *real-time*.

3.2 Pembahasan

Pembahasan hasil penelitian menunjukkan bahwa model *CNN* yang dikembangkan mampu melakukan deteksi dan pengenalan wajah secara efektif dengan tingkat akurasi mencapai 97,22%. Hasil ini sejalan dengan penelitian yang dilakukan oleh Anwar (2025), yang melaporkan akurasi hingga 96,2% dalam sistem deteksi wajah berbasis *CNN*. Peningkatan performa ini juga didukung oleh penerapan teknik regularisasi seperti *Batch Normalization* dan *Dropout*, yang terbukti mampu meningkatkan stabilitas pelatihan dan kemampuan generalisasi model, sebagaimana juga dibuktikan oleh penelitian Ulhaq *et al.* (2023), yang menunjukkan bahwa teknik tersebut mampu memperbaiki performa model dalam proses pelatihan dan pengujian. Selain itu, hasil visualisasi grafik *loss* dan akurasi selama proses pelatihan menunjukkan tren penurunan *loss* dan peningkatan akurasi yang konsisten, menandakan bahwa model mampu belajar pola fitur wajah secara bertahap dan stabil. Keberhasilan ini menunjukkan bahwa arsitektur *CNN* yang dirancang cukup efisien dalam mengekstraksi fitur penting dari citra wajah dengan variasi kondisi pencahayaan, pose, dan ekspresi, yang merupakan tantangan utama dalam deteksi wajah di dunia nyata (Khana *et al.*, 2024). Meski demikian, performa model masih terbatas karena jumlah *dataset* yang relatif kecil, sehingga diharapkan ke depannya penggunaan *dataset* yang lebih besar dan beragam dapat meningkatkan akurasi dan ketahanan sistem. Hasil pengujian terhadap data uji juga menunjukkan bahwa model

mampu mendeteksi wajah secara konsisten dan akurat, yang dibuktikan dengan keberhasilan mendeteksi seluruh sampel data uji dengan bounding box yang tepat. Temuan ini sejalan dengan penelitian Husna *et al.* (2022), yang menyatakan bahwa model CNN memiliki potensi besar dalam aplikasi deteksi wajah secara real-time, asalkan didukung oleh dataset yang cukup dan arsitektur yang optimal. Secara keseluruhan, hasil penelitian ini memperkuat argumen bahwa CNN merupakan pendekatan yang sangat cocok untuk pengembangan sistem deteksi wajah otomatis yang akurat dan efisien pada perangkat desktop, dan mampu bersaing dengan metode-metode lain yang ada saat ini.

4. Kesimpulan

Sesuai temuan penelitian yang sudah dilakukan, bisa ditarik kesimpulan bahwa metode *Convolutional Neural Network (CNN)* berhasil diimplementasikan dalam mendeteksi dan mengenali wajah memakai dataset *Face Recognition Dataset* yang diperoleh di Kaggle. Proses penelitian meliputi tahap *preprocess* data, pembangunan model *CNN*, pelatihan, kemudian penilaian kinerja memakai metrik *accuracy*, *recall*, *precision*, maupun *F1-score*. Temuan pengujian menandakan bahwa model yang sudah dioptimasi beserta penambahan *Batch Normalization* dan *Dropout* mampu meningkatkan performa klasifikasi dengan mendapat angka *accuracy* 97,22%, *precision* yaitu 97%, *recall* yaitu 97%, dan *F1-score* sebesar 97%. Temuan tersebut menandakan bahwa model punya kapabilitas yang sangat baik dalam mempelajari karakteristik wajah dan melakukan klasifikasi secara akurat. Dengan demikian, metode *CNN* terbukti efektif untuk digunakan dalam sistem deteksi dan pengenalan wajah serta mampu memberikan performa yang kompetitif dibandingkan penelitian-penelitian terdahulu yang menggunakan pendekatan serupa. Saran untuk penelitian selanjutnya adalah menggunakan dataset yang lebih besar dan beragam agar kemampuan *generalization* model meningkat, mengembangkan arsitektur *CNN* dengan membandingkannya terhadap model lain seperti *ResNet*, *VGGNet*, atau *MobileNet* untuk memperoleh performa yang lebih baik, serta mengimplementasikan model pada aplikasi nyata, seperti sistem *presensi*, keamanan, atau pengenalan wajah secara *real-time*.

5. Daftar Pustaka

- Adliansyah, F. N., Holilah, H., & Krisdianto, N. (2025). Sistem absensi berbasis face recognition dengan model Inception-ResNet. *RIGGS: Journal of Artificial Intelligence and Digital Business*, 4(4), 2111–2118. <https://doi.org/10.31004/riggs.v4i4.3545>.
- Aminullah, M. (2021). Alat deteksi masker dengan metode convolutional neural network untuk tunanetra pada era new normal. *SinarFe7*, 1–10.
- Anwar, K. (2025). Sistem deteksi wajah berbasis deep learning menggunakan convolutional neural network (CNN). *Journal of Computer Science and Information Technology*, 1(2), 46–52. <https://doi.org/10.70716/jocsit.v1i2.258>.
- Ardiansyah, I., Huda, F. Al, & Yudistira, N. (2025). Pengembangan aplikasi mobile klasifikasi ekspresi wajah manusia pada platform Android menggunakan arsitektur MobileNet. *Jurnal Pengembangan Teknologi dan Ilmu Komputer*, 9(1), 1–10.
- Christyanto, N. E., Jonemaro, E. M. A., & Yudistira, N. (2022). Pengembangan aplikasi Android presensi kehadiran real-time menggunakan pengenalan wajah dengan model Facenet. *Jurnal Pengembangan Teknologi dan Ilmu Komputer*, 6(10), 4839–4847.

- Goyal, H., Sidana, K., Singh, C., Jain, A., & Jindal, S. (2022). A real-time face mask detection system using convolutional neural network. *Multimedia Tools and Applications*, 81(11), 14999–15015. <https://doi.org/10.1007/s11042-022-12166-x>.
- Hangaragi, S., Singh, T., & N, N. (2023). Face detection and recognition using face mesh and deep neural network. *Procedia Computer Science*, 218, 741–749. <https://doi.org/10.1016/j.procs.2023.01.054>.
- Heri, H., Hartomi, Z. H., Ordila, R., & Irawan, Y. (2025). Integration of machine learning models random forest and XGBoost for credit card fraud detection in a Python Flask-based application. *Jurnal Teknologi dan Open Source*, 8(2), 811–825. <https://doi.org/10.36378/jtos.v8i2.4821>.
- Husna, I. N., Ulum, M., Saputro, A. K., Haryanto, D., Laksono, D. T., & Purnamasari, D. N. (2022). Rancang bangun sistem deteksi dan perhitungan jumlah orang menggunakan metode convolutional neural network (CNN). *SinarFe7*, 1–10.
- Karseno, D., Yuhandri, & Ramadhanu, A. (2024). Penerapan algoritma Haar cascade classifier dan computer neural network sebagai presensi karyawan. *Jurnal Kometika*, 11(4), 398–408. <https://doi.org/10.35134/komtekinfo.v12i1.565>.
- Khana, R., Saputra, A. E., & Sobirin, M. (2024). Implementasi sistem presensi pada kegiatan seminar dengan pengenalan wajah menggunakan metode YOLO v5. *Jurnal Kajian Teknik Elektro*, 9(1), 45–57. <https://doi.org/10.52447/jkte.v9i1.7631>.
- Marinda, D. E., & Amin, I. H. Al. (2023). Implementasi metode convolutional neural network untuk deteksi penggunaan masker secara real-time. *Jurnal Teknik Informatika UNIKA Santo Thomas*, 8(1), 81–91.
- Maulana, H. K. (2025). Penerapan arsitektur CNN-EfficientNetB2 dengan transfer learning pada klasifikasi gambar tokoh wayang kulit. *Jurnal Informatika dan Teknik Elektro Terapan*, 13(1), 1–12. <https://doi.org/10.23960/jitet.v13i1.5626>.
- Nabila, M., Idmayanti, R., & Rahmayuni, I. (2021). Deteksi wajah bermasker menggunakan webcam dan AWS EC2 berbasis Raspberry Pi. *JITSI: Jurnal Ilmiah Teknologi Sistem Informasi*, 2(4), 124–133. <https://doi.org/10.62527/jitsi.2.4.54>.
- Satwikayana, S., Wibowo, S. A., & Vendyansyah, N. (2021). Sistem presensi mahasiswa otomatis pada Zoom meeting menggunakan face recognition dengan metode convolutional neural network berbasis web. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 5(2), 785–793.
- Ulhaq, M. R. D., Zaidan, M. A., & Firdaus, D. (2023). Pengenalan ekspresi wajah secara real-time menggunakan metode SSD MobileNet berbasis Android. *Journal of Technology and Informatics (JoTI)*, 5(1), 48–52. <https://doi.org/10.37802/joti.v5i1.387>.