

IMPLEMENTASI SISTEM PRESENSI BERBASIS *FACE RECOGNITION* MENGGUNAKAN MODEL PRA-LATIH *FACENET* DAN *COSINE SIMILARITY* DENGAN PENDEKATAN *WATERFALL* PADA WEBSITE PRAKTIKUM SISTEM INFORMASI

Raihan Daiva Geralda, Akhmad Irsyad, Muhammad Labib Jundillah

Sistem Informasi, Universitas Mulawarman

Jl. Kuaro, Gn. Kelua, Kec. Samarinda Ulu, Kota Samarinda,

Kalimantan Timur 75119, 089620019496

daivageralda831@gmail.com

ABSTRAK

Universitas berperan penting dalam mengembangkan potensi mahasiswa melalui kegiatan akademik, salah satunya praktikum yang menuntut keterlibatan aktif mahasiswa. Presensi menjadi komponen penting dalam memastikan kehadiran dan partisipasi praktikan, namun metode presensi yang masih menggabungkan pencatatan manual dan digital di Program Studi Sistem Informasi Universitas Mulawarman menimbulkan berbagai permasalahan, seperti potensi kesalahan pencatatan, ketergantungan pada asisten laboratorium, serta durasi presensi yang relatif lama. Penelitian ini bertujuan mengimplementasikan sistem presensi berbasis *face recognition* menggunakan model pra-latih *FaceNet* dan metrik *Cosine Similarity* untuk meningkatkan efisiensi dan akurasi pencatatan kehadiran. Metode pengembangan sistem yang digunakan adalah *Waterfall* yang mencakup analisis kebutuhan, perancangan, implementasi, pengujian, dan evaluasi. Sistem dikembangkan dalam bentuk REST API menggunakan FastAPI sebagai *backend* dengan integrasi *FaceNet* untuk ekstraksi *embedding* wajah dan *Cosine Similarity* dengan nilai *threshold* 0,6 sebagai batas keputusan kecocokan wajah. Data penelitian berupa citra wajah praktikan yang diperoleh melalui proses registrasi dan verifikasi menggunakan *webcam*. Hasil penelitian menunjukkan sistem mampu melakukan verifikasi wajah secara cepat dan akurat, dengan pengujian *Blackbox* menunjukkan seluruh fungsi berjalan sesuai kebutuhan serta peningkatan efisiensi waktu presensi lebih dari 80% dibandingkan metode konvensional.

Kata kunci : *presensi, pengenalan wajah, sistem presensi, sistem informasi, praktikum, visi komputer*

1. PENDAHULUAN

Perkembangan teknologi informasi mendorong transformasi proses pencatatan kehadiran di lingkungan pendidikan. Sistem presensi manual atau digital berbasis input pengguna masih rentan terhadap kesalahan, manipulasi data, dan kurang efisien, terutama pada kegiatan praktikum yang menuntut akurasi tinggi. Teknologi biometrik berbasis pengenalan wajah menjadi solusi karena mampu mengidentifikasi individu secara otomatis dan unik, sementara model pra-latih *FaceNet* dapat mengekstraksi fitur wajah dalam bentuk *embedding* dengan tingkat akurasi tinggi pada berbagai kondisi. [1]. Oleh karena itu, penerapan teknologi ini berpotensi meningkatkan keandalan dan efisiensi sistem presensi secara signifikan.

Proses pencocokan identitas menggunakan metrik *Cosine Similarity* juga terbukti efektif dalam meningkatkan akurasi verifikasi wajah, sehingga integrasi teknologi ini berpotensi meningkatkan efisiensi serta meminimalkan *human error* dan manipulasi data pada sistem presensi.

Permasalahan dalam penelitian ini berfokus pada perancangan dan implementasi sistem presensi berbasis pengenalan wajah yang terintegrasi dengan *website* praktikum. Selain itu, penerapan model pra-latih *FaceNet* dan metrik *Cosine Similarity* dilakukan dalam proses verifikasi wajah agar sistem mampu

memberikan hasil yang akurat. Evaluasi performa sistem, baik dari sisi akurasi maupun efisiensi waktu, dibandingkan dengan sistem presensi konvensional yang masih banyak digunakan juga menjadi permasalahan yang perlu mendapat perhatian dalam penelitian ini.

Penelitian ini bertujuan untuk merancang, mengimplementasikan, dan mengevaluasi sistem presensi berbasis *face recognition* menggunakan model pra-latih *FaceNet* dan metrik *Cosine Similarity*. Selain itu, penelitian ini juga bertujuan untuk mengukur performa sistem dari aspek akurasi, presisi, dan *recall*, serta menganalisis efisiensi waktu proses presensi guna memastikan sistem mampu mendukung kegiatan praktikum secara lebih efektif, efisien, dan andal.

Pengembangan sistem dalam penelitian ini dilakukan menggunakan pendekatan *Waterfall* yang bersifat *sequential* dan terdokumentasi dengan baik, meliputi tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan evaluasi. Sistem dirancang dengan arsitektur *client-server* yang terintegrasi langsung dengan *website* praktikum sehingga proses presensi dapat dilakukan secara otomatis melalui layanan *backend*. Pada tahap evaluasi, sistem akan diuji menggunakan metrik performa serta dilakukan perbandingan efisiensi waktu dengan metode presensi konvensional. Pendekatan ini

diharapkan menghasilkan sistem yang terstruktur, mudah dikembangkan, dan mampu memberikan solusi presensi biometrik yang akurat di lingkungan pendidikan.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian oleh Benaboud membandingkan model pra-latih *FaceNet*, *ResNet*, dan *Inception-ResNet V1* dengan metrik *Cosine Similarity*, *Euclidean Distance*, dan *Minkowski* pada dataset CFP yang digabungkan dengan LFW, menggunakan proporsi 80% data latih dan 20% data uji. Hasil penelitian menunjukkan bahwa *FaceNet* dengan metrik *Cosine Similarity* memperoleh akurasi tertinggi sebesar 99,7%. Selain itu, Sugeng dan Barus mengembangkan sistem pengecekan foto paspor berbasis DNN untuk deteksi wajah dan *FaceNet* untuk ekstraksi fitur dengan *Cosine Similarity* sebagai pengukur kemiripan terhadap 6.519 citra wajah, serta menghasilkan akurasi sebesar 97,48%.

Berbagai penelitian lain juga menerapkan face recognition pada sistem presensi menggunakan metode seperti Triplet Loss, EigenFace, LBPH, SVM, dan DNN dengan tingkat akurasi yang bervariasi antara 70% hingga 99%. Secara umum, hasil penelitian menunjukkan bahwa metode face recognition efektif dalam meningkatkan efisiensi dan akurasi pencatatan kehadiran serta mampu mengurangi ketergantungan pada proses manual yang rentan kesalahan. Berdasarkan kajian tersebut, penelitian ini mengimplementasikan model pra-latih *FaceNet* dan metrik *Cosine Similarity* pada sistem presensi berbasis website menggunakan pendekatan Waterfall, dengan penambahan evaluasi presisi dan recall serta penentuan threshold *Cosine Similarity* sebesar 0,6 untuk memastikan sistem mampu membedakan wajah terdaftar dan tidak terdaftar secara optimal.

2.2. Sistem Presensi

Sistem presensi merupakan suatu mekanisme terstruktur yang dirancang untuk mencatat, memantau, dan mengelola kehadiran individu dalam berbagai aktivitas secara sistematis dan akurat. Berdasarkan Kamus Besar Bahasa Indonesia, sistem diartikan sebagai perangkat yang saling berkaitan membentuk suatu kesatuan, sedangkan presensi merujuk pada kehadiran individu dalam suatu kegiatan, sehingga sistem presensi dapat dipahami sebagai sistem pendataan kehadiran yang menjadi bagian penting dari pelaporan aktivitas suatu institusi. Data presensi yang terdokumentasi dengan baik berperan dalam mendukung evaluasi kinerja, kedisiplinan, serta perencanaan kegiatan lanjutan [2]. Seiring perkembangan teknologi, sistem presensi telah berevolusi dari metode manual menjadi sistem digital berbasis komputer maupun perangkat seluler yang mampu meningkatkan efisiensi, akurasi, dan transparansi dalam pengelolaan data kehadiran.

2.3. Face Recognition

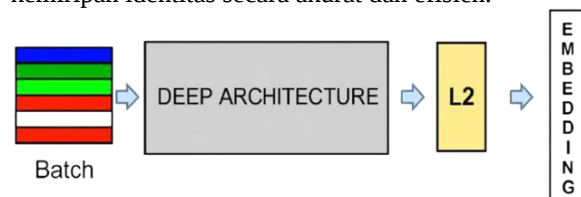
Face Recognition merupakan metode biometrik untuk mengenali identitas seseorang berdasarkan karakteristik unik wajah dengan membandingkan citra yang diambil dengan data pada basis data. Kinerjanya dipengaruhi oleh faktor seperti iluminasi, ekspresi, usia, gaya rambut, aksesoris, intensitas cahaya, dan pergerakan. Sebagai sistem identifikasi biometrik, *face recognition* banyak digunakan dalam autentikasi dan keamanan modern karena mampu membedakan individu berdasarkan fitur biologis unik, seperti sidik jari, retina, iris, dan suara [3].

Secara umum, metode *face recognition* diklasifikasikan menjadi dua jenis, yaitu *face verification (one-to-one)* dan *face identification (one-to-many)*, di mana proses verifikasi mencocokkan satu citra wajah dengan identitas tertentu, sedangkan identifikasi membandingkan citra wajah dengan seluruh data dalam basis data untuk menemukan tingkat kemiripan tertinggi [4]. Proses pengenalan wajah umumnya dilakukan melalui empat tahapan utama, yaitu akuisisi citra untuk mendeteksi dan mengambil wajah, preprocessing untuk standarisasi dan penyesuaian citra, ekstraksi fitur untuk memperoleh ciri khas wajah, serta klasifikasi untuk menentukan identitas berdasarkan fitur yang telah diekstraksi.

2.4. Model Pra-Latih *FaceNet*

Model pra-latih (*pretrained model*) adalah model yang telah dilatih pada dataset berskala besar dan dapat digunakan kembali untuk menyelesaikan permasalahan baru tanpa pelatihan dari awal, melainkan hanya dengan melakukan penyesuaian pada bagian tertentu dari model [5]. Pendekatan ini membuat pengembangan sistem lebih efisien dari segi waktu dan sumber daya komputasi karena model sudah mampu mengekstraksi fitur penting dari data. Beberapa arsitektur *Convolutional Neural Network (CNN)* yang umum digunakan sebagai model pra-latih antara lain *LeNet-5* dan *GoogleNet (Inception)*.

FaceNet merupakan salah satu model pra-latih berbasis *Deep Convolutional Neural Network (DCNN)* yang dikembangkan oleh Google dan merupakan pengembangan dari arsitektur *Inception* dan *ResNet*. Model ini dirancang untuk mengintegrasikan proses deteksi dan pengenalan wajah dalam satu sistem yang menyeluruh dengan memetakan citra wajah ke dalam ruang vektor berdimensi 128 melalui proses embedding. Representasi vektor ini memungkinkan perhitungan jarak geometris antar wajah untuk menentukan tingkat kemiripan identitas secara akurat dan efisien.



Gambar 1. Alur ekstraksi fitur algoritma *FaceNet*

Gambar 1 menunjukkan arsitektur dari model *FaceNet* yang terdiri dari beberapa komponen utama, yaitu input *batch layer*, jaringan DCNN untuk ekstraksi fitur, proses normalisasi L2, dan *embedding layer*. Jaringan DCNN bertugas mengekstraksi fitur spasial dan semantik wajah melalui proses pelatihan *end-to-end*, kemudian hasilnya dinormalisasi menggunakan L2 *normalization* untuk memastikan vektor *embedding* berada pada skala yang seragam. Vektor *embedding* yang dihasilkan bersifat kompak dan informatif, di mana jarak antar vektor mencerminkan tingkat kemiripan antar wajah, sehingga sangat efektif digunakan dalam sistem verifikasi dan identifikasi wajah [6].

2.5. Cosine Similarity

Cosine Similarity merupakan metrik yang digunakan untuk mengukur tingkat kesamaan antara dua objek atau vektor dengan membandingkan sudut di antara keduanya dalam ruang vektor. Berbeda dengan *Euclidean Distance* yang mengukur jarak minimum antar objek, *Cosine Similarity* menilai kesamaan berdasarkan arah vektor, sehingga lebih stabil terhadap perbedaan skala data dan sangat cocok digunakan pada data berdimensi tinggi seperti citra atau *embedding* wajah [7]. Oleh karena itu, metrik banyak diterapkan dalam sistem pengenalan pola.

Nilai *Cosine Similarity* berada pada rentang 0 hingga 1, di mana nilai yang mendekati 1 menunjukkan tingkat kemiripan yang tinggi, sedangkan nilai mendekati 0 menunjukkan rendahnya kemiripan antar vektor yang dibandingkan [8].

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad (1)$$

2.6. Waterfall

Metodologi *Waterfall* merupakan salah satu model pengembangan perangkat lunak klasik yang masih relevan digunakan, terutama pada proyek *website* dengan kebutuhan yang jelas dan stabil. Model ini bersifat sekuensial dan linier, di mana setiap tahapan pengembangan dilakukan secara berurutan, meliputi analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Pada tahap analisis, kebutuhan sistem dikumpulkan melalui wawancara, observasi, dan studi pustaka sebagai dasar perancangan arsitektur, *database*, serta antarmuka pengguna. Tahap implementasi dilakukan dengan merealisasikan desain ke dalam kode program, kemudian dilanjutkan dengan pengujian untuk memastikan fungsionalitas sistem berjalan sesuai kebutuhan. Keunggulan metode *Waterfall* terletak pada alur kerja yang terstruktur, terdokumentasi dengan baik, dan mudah dikontrol, sehingga cocok digunakan pada proyek dengan ruang lingkup yang terdefinisi dengan jelas [9].

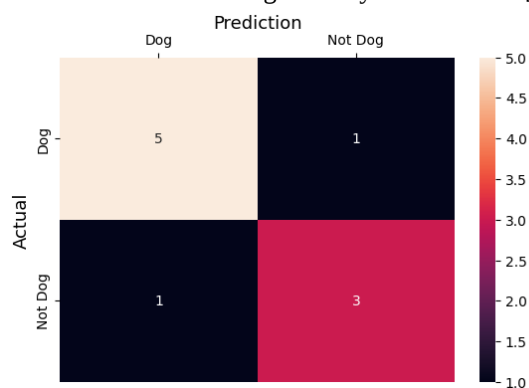
2.7. Arsitektur Client-Server dan REST API

Arsitektur *client-server* merupakan fondasi utama dalam pengembangan *website* modern yang memisahkan peran antara *client* sebagai antarmuka pengguna dan *server* sebagai pengelola logika bisnis serta penyimpanan data. Komunikasi antara kedua komponen dilakukan melalui protokol HTTP/HTTPS sehingga memungkinkan pertukaran data yang efisien dan aman. Dalam arsitektur ini, *client* mengirimkan permintaan ke *server* untuk mengakses sumber daya, sedangkan *server* memproses permintaan tersebut melalui logika bisnis, pengelolaan *database*, serta mekanisme autentikasi sebelum mengirimkan respons kembali ke *client*. Pembagian tanggung jawab ini memungkinkan optimasi performa sistem, karena pemrosesan data difokuskan pada sisi *server* sementara *client* menangani tampilan antarmuka pengguna [10].

REST API (*Representational State Transfer Application Programming Interface*) merupakan teknologi yang digunakan untuk mendukung komunikasi antara *client* dan *server* dalam arsitektur *client-server*. REST API diimplementasikan pada sisi *server* untuk menyediakan layanan yang dapat diakses oleh aplikasi *web* maupun *mobile*, sehingga memudahkan integrasi antar sistem tanpa perlu membangun ulang fungsi yang sudah tersedia. REST API dirancang dengan prinsip kesederhanaan, kinerja, dan portabilitas, serta memanfaatkan protokol HTTP dan struktur URL sebagai *endpoint* untuk pertukaran data [11].

2.8. Confusion Matrix

Confusion matrix merupakan tabel dua dimensi untuk mengevaluasi performa model klasifikasi dengan membandingkan hasil prediksi dan data aktual. Baris menunjukkan kelas aktual, sedangkan kolom menunjukkan kelas prediksi, di mana sel diagonal merepresentasikan prediksi benar dan sel lainnya menunjukkan kesalahan klasifikasi. *Confusion matrix* memberikan gambaran pola kesalahan model yang lebih informatif dibandingkan hanya nilai akurasi [12].



Gambar 2. Visualisasi *confusion matrix*

Gambar 2 menunjukkan komponen utama dalam *confusion matrix* yang terdiri dari *True Positive* (TP), *False Positive* (FP), *False Negative* (FN), dan *True Negative* (TN), keseluruhan komponen ini

menggambarkan seluruh kemungkinan hasil prediksi model terhadap kondisi aktual data. Komponen tersebut digunakan untuk beberapa metrik evaluasi performa, antara lain akurasi, presisi, dan *recall*. Akurasi mengukur proporsi prediksi yang benar terhadap seluruh data, presisi menunjukkan ketepatan prediksi positif, sedangkan *recall* mengukur kemampuan model dalam mengenali seluruh data positif aktual.

2.9. Blackbox Testing

Blackbox Testing merupakan metode pengujian perangkat lunak yang berfokus pada pengujian fungsional aplikasi tanpa memperhatikan struktur internal atau kode program. Pengujian dilakukan dengan membandingkan input yang diberikan dengan output yang dihasilkan, sehingga merepresentasikan sudut pandang pengguna akhir dalam menggunakan sistem. Metode ini sangat relevan dalam pengembangan *website* untuk memastikan seluruh fitur dan alur kerja berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan [13]. Selain itu, *Blackbox Testing* efektif dalam mengidentifikasi cacat fungsional, inkonsistensi, serta masalah kegunaan, termasuk validasi form, manajemen sesi, dan interaksi antar komponen sistem pada aplikasi *web* modern.

3. METODE PENELITIAN

3.1. Identifikasi Masalah

Identifikasi masalah pada penelitian ini berfokus pada proses presensi praktikum Program Studi Sistem Informasi yang masih menggunakan metode gabungan manual dan digital, namun belum efisien dan akurat. Proses pencatatan kehadiran memerlukan waktu 3–5 menit per sesi sehingga mengurangi durasi efektif praktikum, serta berisiko menimbulkan kesalahan akibat keterbatasan perhatian asisten dan kondisi ruangan yang kurang kondusif. Presensi digital yang digunakan juga masih rawan kesalahan teknis, sehingga validitas data kehadiran belum terjamin. Oleh karena itu, diperlukan solusi berupa sistem presensi berbasis biometrik *face recognition* yang terintegrasi dengan *website* praktikum, mampu mencatat kehadiran secara *real-time*, akurat, dan lebih andal.

3.2. Studi Literatur

Studi literatur dilakukan untuk memahami penerapan teknologi *face recognition* dalam sistem presensi berbasis biometrik. Teknologi ini memanfaatkan fitur unik wajah manusia untuk melakukan verifikasi kehadiran secara otomatis dan akurat. *FaceNet* digunakan sebagai model ekstraksi fitur wajah yang menghasilkan vektor *embedding* berdimensi tetap, kemudian dibandingkan menggunakan metrik *Cosine Similarity* yang terbukti efektif dalam meningkatkan akurasi pengenalan wajah. Berbagai penelitian sebelumnya menunjukkan bahwa kombinasi *FaceNet* dan *Cosine Similarity* menghasilkan performa yang tinggi, meskipun

sebagian besar masih dikembangkan dalam bentuk aplikasi Python GUI. Oleh karena itu, penelitian ini mengembangkan sistem presensi berbasis arsitektur *client-server* menggunakan REST API dengan pendekatan *Waterfall*, sehingga sistem lebih skalabel, terintegrasi, dan siap digunakan secara *real-time* pada lingkungan *web* praktikum.

3.3. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk merumuskan spesifikasi dan batasan sistem presensi berbasis *face recognition* yang akan dikembangkan agar sesuai dengan kebutuhan praktikan. Sistem dirancang menggunakan arsitektur *client-server*, di mana antarmuka pengguna berjalan pada *browser*, sementara pengolahan citra wajah dan logika bisnis dilakukan di sisi *server* melalui REST API berbasis *FastAPI*. Setiap praktikan melakukan registrasi dengan merekam 50 citra wajah yang dibagi menjadi data latih (80%) dan data uji (20%). Proses deteksi wajah dilakukan secara *real-time* di sisi *client* menggunakan *MediaPipe*, sedangkan ekstraksi fitur dilakukan di *server* menggunakan model *FaceNet* yang menghasilkan *embedding* wajah berdimensi 128. Proses verifikasi dilakukan dengan pendekatan *face verification (one-to-one)* menggunakan metrik *Cosine Similarity* untuk menentukan status kehadiran praktikan.

Sistem ini terintegrasi dengan *website* utama praktikum melalui *session* atau *cookie* hasil login, sehingga tidak memerlukan fitur login terpisah. Sistem menerima identitas praktikan dari *website* utama, kemudian mengirimkan kembali status verifikasi kehadiran untuk diproses lebih lanjut oleh sistem utama. Pendekatan ini memastikan konsistensi data dan kelancaran alur informasi tanpa mengganggu sistem yang sudah berjalan.

Dari sisi kebutuhan non-fungsional, sistem dirancang agar stabil selama jam praktikum, mampu melakukan verifikasi wajah dalam waktu kurang dari 5 detik, serta aman dengan hanya menyimpan data dalam bentuk *embedding*, bukan citra wajah. Sistem juga bersifat portabel karena dapat diakses melalui *browser* di berbagai perangkat yang memiliki kamera, mudah diintegrasikan dengan sistem lain, dan mampu menangani banyak pengguna secara bersamaan tanpa penurunan performa yang signifikan.

3.4. Perancangan Sistem

Perancangan sistem presensi dilakukan menggunakan arsitektur *client-server* dengan layanan REST API yang menghubungkan antarmuka pengguna dan *server*. Sistem ini dikembangkan untuk melakukan verifikasi wajah dengan pendekatan *face verification*.

3.5. Arsitektur Model

FaceNet digunakan dalam sistem presensi untuk mengekstraksi fitur wajah dengan mengubah citra wajah menjadi vektor *embedding* berdimensi tetap.

Model ini memproses citra wajah sebagai input untuk menghasilkan representasi numerik yang digunakan pada tahap verifikasi wajah.

```
array([ 1.8630524, -1.2883751, -0.26330343, 0.4060825, -1.4930112,
        0.04194283, 0.5366742, -1.5315478, -0.8199895, 1.141684,
        -0.9820187, -0.39845252, -0.26111835, 0.67693186, -1.1482742,
        -0.00436858, 0.10645946, -1.4834762, 0.03582224, 0.92423356,
        0.6945138, -0.23059475, -0.8705194, 0.13284585, 0.9412558,
        -0.3150682, 0.25655016, -0.51082534, -0.4814728, -0.02302771,
        0.10926945, 0.5947657, -1.1671576, -0.03692105, 0.24883932,
        0.05052355, -1.0284573, -0.60186493, 1.096345, -0.5197489,
        0.56739664, 0.8654664, 0.16737503, -1.2106476, 0.31544933,
        -1.08405, -0.40912414, 2.3330898, -1.4190598, 0.27882645,
        -0.55143285, 0.97984797, 0.19340041, -1.2868431, 0.6797951,
        0.10718144, 0.4374029, 0.21094814, 1.0192963, -0.41130677,
        0.42843115, 0.17711377, 0.13484177, 1.3389393, 0.06584957,
        1.6536247, 0.7763515, 0.9670694, -0.6451823, 1.2065457,
        0.5128179, -0.59590405, 1.0535465, -0.39617723, -0.4667322,
        -0.56861615, 0.11612426, -0.22631495, -0.1945281, -1.2008042,
        0.0890996, -0.90865064, -0.36028314, 1.2406405, 1.8607384,
        -2.063767, -0.23035836, -1.86226, -0.2555508, -0.9054414,
        0.5506839, 0.09993919, 0.5578571, 1.2058746, 0.37803653,
        -0.09240614, 1.5897188, 1.4587421, -1.5544238, 0.07799052,
        0.42279005, 0.26740518, -0.23773858, -0.65404725, -0.8052854,
        -0.22684993, 0.89914584, 0.01673314, -1.3139844, 0.21601504,
        0.45872065, 1.5960048, 2.3457112, -0.78951114, -0.45617726,
        0.2467384, -0.6505567, 1.8112195, -0.82550323, 0.6479339,
        1.5175676, 0.03704251, 1.0670711, -0.12793806, 1.3369213,
        -1.2630515, 0.3943154, -0.15486093], dtype=float32)
```

Gambar 3. Output ekstraksi wajah

Gambar 3 menampilkan hasil proses ekstraksi wajah, di mana citra wajah berukuran $160 \times 160 \times 3$ diproses oleh model *FaceNet* dan diubah menjadi vektor *embedding* berukuran 128 dimensi. Vektor *embedding* ini merepresentasikan ciri unik wajah setiap praktikan dalam bentuk numerik, sehingga memudahkan proses perbandingan antar wajah menggunakan metrik *Cosine Similarity*. Representasi tersebut memungkinkan sistem melakukan verifikasi wajah secara konsisten meskipun terdapat variasi pencahayaan dan posisi wajah. Pada sisi *client*, sistem menangkap citra wajah praktikan dengan *webcam*, kemudian mengirimkan data tersebut ke *server* melalui *endpoint* REST API. *Client* juga menampilkan informasi hasil registrasi maupun verifikasi kepada pengguna secara *real-time*. Antarmuka ini dikembangkan menggunakan PHP, JavaScript, serta library tambahan untuk mengakses kamera perangkat.

Sisi *server* bertanggung jawab terhadap seluruh proses pemrosesan data, meliputi ekstraksi fitur wajah menggunakan *FaceNet*, penyimpanan *embedding* hasil registrasi, serta verifikasi wajah menggunakan metrik *Cosine Similarity*. Seluruh proses komunikasi antara *client* dan *server* dikelola oleh REST API yang dibangun menggunakan *framework* FastAPI, sehingga sistem dapat bekerja secara terpusat, konsisten, dan mudah diintegrasikan dengan *website* praktikum yang sudah ada.

3.6. Arsitektur Sistem

Sistem presensi pada penelitian ini dibangun menggunakan arsitektur *client-server*, di mana proses utama seperti registrasi wajah, verifikasi kehadiran, serta pengelolaan *embedding* dijalankan pada sisi *server* melalui layanan REST API. Pendekatan ini dipilih karena fleksibel, terstruktur, serta memudahkan pengembangan dan pemeliharaan sistem. Pemisahan yang jelas antara sisi *client* dan *server* juga

memungkinkan sistem dikembangkan secara modular tanpa mengganggu komponen yang sudah berjalan.

Sisi *client* berfungsi sebagai antarmuka pengguna berbasis *browser* untuk pengambilan citra wajah dan menampilkan hasil verifikasi secara *real-time*, sedangkan *server* menangani pemrosesan citra, ekstraksi fitur menggunakan *FaceNet*, perhitungan kemiripan *embedding*, serta pengiriman status presensi ke *client* agar sistem berjalan efisien dan konsisten. Sisi *client* juga memastikan pengalaman pengguna yang responsif dan interaktif selama proses presensi.

4. HASIL DAN PEMBAHASAN

4.1. Alat dan Spesifikasi Sistem

Implementasi sistem presensi berbasis *face recognition* membutuhkan infrastruktur *server* yang mampu mendukung proses komputasi dan penyimpanan data secara andal. Oleh karena itu, sistem dijalankan pada layanan *Virtual Private Server* (VPS) yang berfungsi sebagai *backend* aplikasi sekaligus media penyimpanan data hasil ekstraksi fitur wajah praktikan. Spesifikasi VPS yang digunakan disesuaikan dengan kebutuhan minimum sistem agar layanan dapat berjalan stabil selama proses presensi berlangsung.

Tabel 1. Spesifikasi VPS

Komponen	Spesifikasi
Penyedia Layanan VPS	IDCloudHost
CPU	2 Core (2 vCPU)
RAM	2 GB
Penyimpanan	20 GB
Sistem Operasi	Ubuntu 24.04 LTS

Dari Tabel 1 dapat dilihat bahwa spesifikasi VPS yang digunakan telah mencukupi untuk mendukung proses ekstraksi fitur dan verifikasi wajah secara efisien. Kombinasi CPU, RAM, dan kapasitas penyimpanan memungkinkan sistem berjalan stabil saat diakses oleh banyak pengguna, sementara penggunaan sistem operasi Ubuntu 24.04 LTS memberikan dukungan keamanan dan kestabilan layanan *backend* selama kegiatan praktikum.

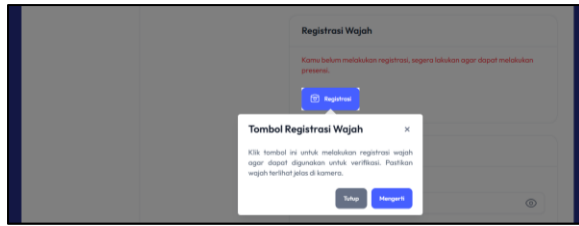
4.2. Hasil Perancangan Tampilan



Gambar 4. Pemberitahuan Status Registrasi

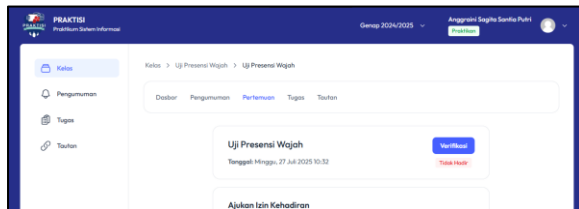
Gambar 4 pada halaman sebelumnya menunjukkan tampilan awal sistem berupa pemberitahuan status registrasi wajah praktikan, yang menampilkan pesan informasi dan tombol untuk memulai proses registrasi. Pemberitahuan ini akan

muncul selama registrasi belum dilakukan dan akan otomatis hilang setelah proses registrasi berhasil.



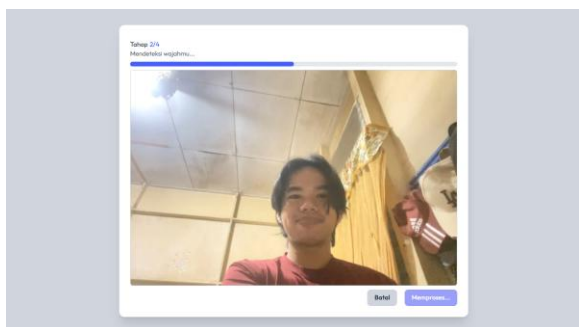
Gambar 5. Halaman Registrasi Wajah

Gambar 5 menunjukkan tampilan halaman registrasi wajah yang dirancang untuk memudahkan praktikan dalam melakukan proses registrasi. Pada halaman ini ditampilkan pesan informasi penting, dan halaman registrasi dapat diakses langsung melalui menu sistem. Desain antarmuka dibuat sederhana dan informatif agar pengguna dapat mengikuti setiap tahapan registrasi dengan jelas dan terarah.



Gambar 6. Halaman Verifikasi Wajah

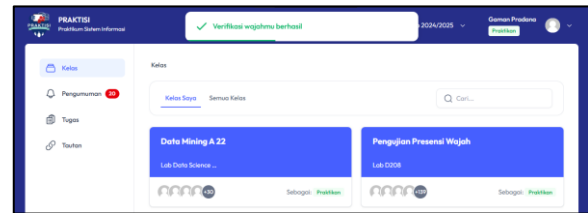
Gambar 6 menunjukkan halaman verifikasi presensi berbasis *face recognition* yang dilengkapi tombol untuk memulai proses verifikasi wajah. Antarmuka halaman ini dirancang responsif sehingga dapat diakses dengan baik melalui berbagai perangkat. Tombol ini tersedia pada setiap pertemuan praktikum sehingga praktikan dapat melakukan presensi secara mandiri, yang berdampak pada peningkatan efisiensi waktu pelaksanaan praktikum.



Gambar 7. Proses Registrasi dan Verifikasi Wajah

Gambar 7 menunjukkan tampilan proses registrasi wajah yang dilengkapi indikator kemajuan berupa empat tahapan, yaitu persiapan, pendeteksian wajah, penyimpanan data, dan hasil, serta bilah progres yang menampilkan perkembangan proses registrasi. Ketika proses verifikasi dilakukan, sistem menampilkan indikator kemajuan yang terdiri atas tiga

tahapan, yaitu persiapan, pemrosesan wajah, dan hasil, serta bilah progres yang menunjukkan perkembangan proses verifikasi hingga identitas pengguna berhasil dikonfirmasi.



Gambar 8. Informasi Proses

Gambar 8 menunjukkan notifikasi keberhasilan registrasi dan verifikasi wajah yang ditampilkan secara singkat dengan pesan dan ikon hijau sebagai indikator proses berhasil. Sistem juga menampilkan notifikasi berwarna merah sebagai tanda kegagalan sehingga pengguna perlu mengulangi proses, serta notifikasi kuning sebagai peringatan apabila terdapat kondisi yang perlu diperhatikan.

4.3. Hasil Pengujian Sistem

Pengujian sistem dilakukan setelah seluruh komponen *client*, *server*, REST API, dan komponen pendukung lainnya selesai diimplementasikan, dengan tujuan memastikan sistem berjalan sesuai kebutuhan dan spesifikasi. Pengujian difokuskan pada validasi fungsionalitas menggunakan *Blackbox Testing* untuk mengevaluasi respons sistem terhadap berbagai skenario registrasi dan verifikasi wajah dari sudut pandang pengguna. Pelaksanaan pengujian disimulasikan menyerupai satu sesi praktikum nyata yang melibatkan praktikan dengan perangkat dan kondisi lingkungan yang beragam, sehingga sistem dapat diuji pada kondisi mendekati penggunaan riil. Seluruh interaksi pengguna, mulai dari akses halaman hingga respons sistem, didokumentasikan untuk mendukung hasil pengujian yang diharapkan mampu merepresentasikan performa sistem secara objektif.

Tabel 2. Pengujian *Blacbox* Fitur Registrasi

Skenario Pengujian	Hasil yang Diharapkan	Status
Akses halaman registrasi	Halaman terbuka dan <i>webcam</i> aktif	Sesuai
Akuisisi citra wajah	Sistem menyimpan 50 citra wajah	Sesuai
Pengiriman data ke <i>server</i>	Data diterima dan diproses	Sesuai
Ekstraksi dan penyimpanan	<i>Embedding</i> tersimpan, respons berhasil	Sesuai
Registrasi berhasil	Notifikasi berhasil ditampilkan	Sesuai
Registrasi gagal	Notifikasi gagal ditampilkan	Sesuai

Berdasarkan Tabel 5, seluruh skenario pengujian pada fitur registrasi wajah menunjukkan hasil sesuai dengan yang diharapkan. Sistem mampu menampilkan halaman registrasi, mengakuisisi citra wajah,

mengirim data ke *server*, serta menyimpan *embedding* wajah dengan baik. Selain itu, sistem juga berhasil menampilkan notifikasi yang sesuai pada kondisi registrasi berhasil maupun gagal, sehingga dapat disimpulkan bahwa fitur registrasi telah berfungsi dengan benar secara fungsional.

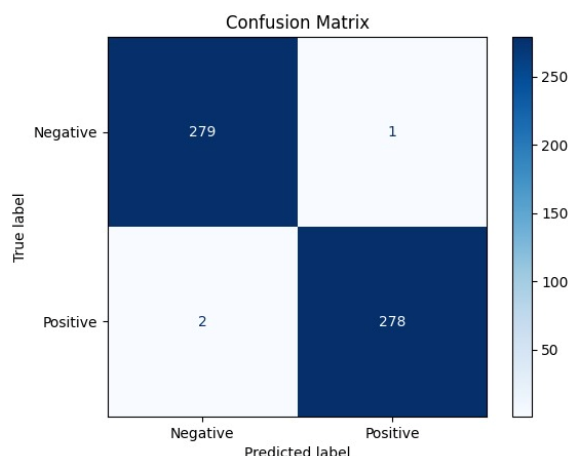
Tabel 3. Pengujian *Blackbox* Fitur Verifikasi

Skenario Pengujian	Hasil yang Diharapkan	Status
Akses halaman verifikasi	Halaman terbuka dan <i>webcam</i> aktif	Sesuai
Akuisisi citra wajah	Sistem menangkap 1 citra wajah	Sesuai
Pengiriman data ke <i>server</i>	Data terkirim tanpa <i>error</i>	Sesuai
Ekstraksi dan pemilihan data	<i>Embedding</i> diekstraksi dan dimuat	Sesuai
Proses verifikasi	Perbandingan dengan <i>Cosine Similarity</i>	Sesuai
Verifikasi berhasil	Notifikasi berhasil ditampilkan	Sesuai

Berdasarkan Tabel 6, seluruh skenario pengujian pada fitur verifikasi wajah berjalan sesuai dengan hasil yang diharapkan. Sistem mampu melakukan akuisisi citra, pemrosesan *embedding*, serta proses perbandingan wajah dengan baik, dan menampilkan notifikasi yang sesuai pada kondisi verifikasi berhasil maupun gagal.

3.7. Evaluasi Performa Model

Evaluasi performa model dilakukan untuk menilai kemampuan sistem dalam melakukan verifikasi wajah menggunakan *embedding* yang dihasilkan oleh model *FaceNet* dan metrik *Cosine Similarity*. Setiap praktikan memiliki 50 citra wajah hasil registrasi, dengan 80% digunakan sebagai data referensi (*embedding*) dan 20% sebagai data uji. Data uji wajah praktikan kemudian digabungkan dengan data uji acak dari dataset *Labeled Faces in the Wild* (LFW) sebagai representasi wajah tidak terdaftar, sehingga sistem diuji dalam membedakan wajah valid dan tidak valid.

Gambar 9. *Confusion matrix* registrasi

Berdasarkan gambar 9, diperoleh nilai *True Positive* (TP) sebesar 278, *True Negative* (TN) sebesar 279, *False Positive* (FP) sebesar 2, dan *False Negative* (FN) sebesar 1. Hasil ini menunjukkan bahwa sistem mampu mengenali wajah praktikan yang terdaftar serta menolak wajah yang tidak terdaftar dengan tingkat kesalahan yang sangat rendah. Jumlah FP dan FN yang minimal mengindikasikan bahwa sistem memiliki kemampuan verifikasi yang stabil dan selektif dalam membedakan wajah yang valid dan tidak valid.

Tabel 4. Evaluasi metrik proses registrasi

Metrik	Nilai	Keterangan
Akurasi	99,4%	Ketepatan klasifikasi sangat tinggi
Presisi	99,6%	Risiko penerimaan wajah tidak sah sangat rendah
Recall	99,2%	Kemampuan mengenali wajah valid sangat andal

Tabel 7 menunjukkan model *FaceNet* yang dikombinasikan dengan *Cosine Similarity* memiliki performa yang sangat baik dalam sistem presensi berbasis *face recognition*. Nilai akurasi yang tinggi menandakan ketepatan klasifikasi secara keseluruhan, presisi yang tinggi menunjukkan bahwa sistem jarang menerima wajah yang tidak terdaftar, sedangkan nilai recall yang tinggi menunjukkan kemampuan sistem dalam mengenali hampir seluruh wajah praktikan yang valid.

3.8. Evaluasi Performa Sistem

Evaluasi performa sistem dilakukan untuk menilai konsistensi dan keandalan sistem presensi berbasis *face recognition* dalam mengenali wajah praktikan pada skenario penggunaan nyata. Sebanyak 28 praktikan melakukan proses verifikasi wajah dengan jumlah sepuluh kali dalam satu sesi praktikum, sehingga hasil evaluasi mencerminkan kemampuan sistem dalam mengenali wajah pengguna secara berulang pada kondisi pengambilan citra yang bervariasi.

Tabel 5. Ringkasan hasil evaluasi sistem presensi

Praktikan	Perangkat	Presensi Berhasil	Keberhasilan
...	...	...	...
P2	iPhone 15	0	0%
P1	Infinix Note 30	2	20%
P17	Xiaomi 14T	6	60%
P4	Infinix Note 30	8	80%
P3	Redmi Note 14 5G	10	100%
Rata-rata			78, 93%

Tabel 8 menunjukkan sebagian besar praktikan memiliki tingkat keberhasilan yang tinggi, bahkan mencapai 100%, yang mengindikasikan bahwa sistem mampu mengenali wajah pengguna secara konsisten

ketika kondisi pengambilan citra relatif stabil dan sesuai dengan data registrasi. Hal ini menunjukkan bahwa sistem presensi berbasis *face recognition* memiliki performa yang baik dan dapat diandalkan untuk digunakan pada kegiatan praktikum. Selain itu, perangkat yang digunakan praktikan umumnya memiliki kualitas kamera yang memadai, sehingga faktor perangkat tidak menjadi penyebab utama penurunan performa sistem.

Namun, beberapa praktikan memperoleh tingkat keberhasilan rendah hingga gagal sepenuhnya, yang menunjukkan bahwa sistem sangat bergantung pada kesesuaian citra wajah antara tahap registrasi dan verifikasi. Variasi sudut pengambilan, ekspresi wajah, jarak kamera, serta pencahayaan menyebabkan perbedaan *embedding* wajah yang signifikan. Selain itu, penggunaan *Cosine Similarity* sebagai metrik pembandingan bersifat non-adaptif, sehingga sistem tidak dapat menyesuaikan diri terhadap variasi wajah yang kompleks. Oleh karena itu, diperlukan peningkatan standar akuisisi citra serta pengembangan metode verifikasi yang lebih adaptif agar performa sistem dapat lebih stabil dan optimal.

3.9. Evaluasi Efektivitas Sistem

Evaluasi efektivitas sistem menunjukkan bahwa presensi berbasis *face recognition* mampu meningkatkan efisiensi waktu secara signifikan dibandingkan metode konvensional. Pada satu sesi praktikum dengan minimal 25 praktikan, metode manual membutuhkan waktu sekitar 5 menit untuk mencatat kehadiran seluruh peserta, sedangkan sistem *face recognition* mampu menyelesaikan proses presensi dalam waktu kurang dari 1 menit, dengan rata-rata durasi verifikasi per praktikan sekitar 1 detik.

Pengurangan durasi presensi lebih dari 80% ini membuktikan bahwa sistem yang dikembangkan sangat efektif untuk diterapkan pada kegiatan praktikum dengan jumlah peserta besar. Proses presensi menjadi lebih cepat, otomatis, dan minim interaksi manual, sehingga dapat mengurangi antrian, meningkatkan efisiensi waktu pelaksanaan praktikum, serta memberikan kenyamanan yang lebih baik bagi pengguna.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil perancangan, implementasi, pengujian, dan evaluasi, sistem presensi berbasis *face recognition* pada *website* praktikum Sistem Informasi terbukti mampu memenuhi tujuan penelitian dari sisi fungsionalitas, performa pengenalan wajah, dan efektivitas waktu presensi. Sistem diimplementasikan menggunakan arsitektur *client-server* berbasis REST API dengan *FastAPI* sebagai *server-side* dan *Laravel* sebagai *client-side*, serta memanfaatkan *FaceNet* untuk ekstraksi *embedding* wajah dan *Cosine Similarity* sebagai metode verifikasi, dengan performa model di atas 99% pada pengujian laboratorium. Evaluasi di lingkungan nyata menunjukkan tingkat keberhasilan rata-rata 78,93%, dengan variasi antar

pengguna akibat perbedaan kondisi citra dan keterbatasan metode verifikasi. Meskipun demikian, sistem mampu mengurangi durasi presensi secara signifikan dari sekitar 5 menit menjadi kurang dari 1 menit, sehingga lebih efektif dibandingkan metode konvensional. Pengembangan selanjutnya disarankan meliputi metode verifikasi lebih adaptif seperti kombinasi SVM atau *threshold* dinamis, peningkatan kualitas data registrasi, penambahan mekanisme keamanan seperti *liveness detection*, serta optimasi sistem untuk menangani lebih banyak pengguna secara simultan.

DAFTAR PUSTAKA

- [1] A. Putra Meldyantono dan B. Satrio Waluyo Poetro, "Implementasi Sistem Absensi Berbasis Pengenalan Wajah Menggunakan Metode CNN dan Model FaceNet," *Jurnal Rekayasa Sistem Informasi dan Teknologi*, vol. 3, no. 2, hlm. 996–1006, Feb 2025, doi: <https://doi.org/10.70248/jrsit.v2i3.1857>.
- [2] A. M. Wijaya, J. Eric Samodra, dan Suyoto, "Sistem Presensi Pegawai Dengan Face Recognition Menggunakan Deep Learning CNN," *Jurnal Informatika Atma Jogja*, vol. 4, no. 2, hlm. 163–168, Nov 2023, doi: [10.24002/jiaj.v4i2.7660](https://doi.org/10.24002/jiaj.v4i2.7660).
- [3] D. A. Alwi dan N. E. Budiyanto, "Aplikasi Presensi Karyawan Berbasis Android Menggunakan Pustaka OpenCV (Studi Kasus Radio Rasika Ungaran Salatiga Ambarawa)," dalam *Prosiding Sains Nasional dan Teknologi*, Semarang, Nov 2021, hlm. 171–177. doi: doi.org/10.36499/psnst.v1i1.5248.
- [4] M. Khatama Insani dan D. Budi Santoso, "Perbandingan Kinerja Model Pre-Trained CNN (VGG16, RESNET, dan INCEPTIONV3) untuk Aplikasi Pengenalan Wajah pada Sistem Absensi Karyawan," *Jurnal Indonesia: Manajemen Informatika dan Komunikasi (JIMIK)*, vol. 5, no. 3, hlm. 2612–2622, Sep 2024, doi: <https://doi.org/10.35870/jimik.v5i3.925>.
- [5] I. N. Alam, "Metode Transfer Learning Pada Deep Convolutional Neural Network (DCNN) Untuk Pengenalan Ekspresi Wajah," *Binus University, Jakarta*, 2022. [Daring]. Tersedia pada: <https://www.researchgate.net/publication/364330227>
- [6] A. Y. Nadzif, "Pengenalan Pola Wajah Gender Face Recognition Menggunakan Metode Cosine Similarity," Undergraduate Thesis, Universitas Islam Sultan Agung, Semarang, 2024. Diakses: 2 Juni 2025. [Daring]. Tersedia pada: repository.unissula.ac.id/34056/
- [7] F. Adnan, I. Amelia, dan S. U. Shiddiq, "Implementasi Voice Recognition Berbasis Machine Learning," *Edu ElektriKa Journal*, vol. 11, no. 1, hlm. 24–29, Jan 2022, doi: [10.15294/eej.v11i1.57283](https://doi.org/10.15294/eej.v11i1.57283).

- [8] R. A. Wibowo, D. Indrayana, dan Arsiyanik, "Implementasi Metode Waterfall dalam Pengembangan Aplikasi Toko Online Bebas Web di Cv Aishastore.Id," *Syntax Admiration*, vol. 6, no. 1, hlm. 465–82, Jan 2025, doi: 10.46799/jsa.v6i1.2058.
- [9] R. Patria, "Apa Itu Client Server? Pengertian, Fungsi dan Cara Kerjanya!" Diakses: 6 Juni 2025. [Daring]. Tersedia pada: <https://www.domainesia.com/berita/client-server/>
- [10] M. K. N. F. Affrianto dan A. B. Cahyono, "Implementasi REST API Untuk Fitur Rencana Strategis Program Pada SIMPEDA," *AUTOMATA*, vol. 3, no. 2, hlm. 1–7, Agu 2022, Diakses: 2 Juni 2025. [Daring]. Tersedia pada: journal.uin.ac.id/AUTOMATA/article/view/241
- [11] A. P. Suryotomo, B. M. Akbar, dan R. Husaini, "Analisis Performa Framework FastAPI pada Aplikasi Manajemen Penanganan Lost Circulation di Pemboran Sumur Minyak," *Jurnal Informatika dan Teknologi Informasi*, vol. 21, no. 1, hlm. 110–121, Feb 2024, doi: 10.31515/telematika.v21i1.13259.
- [12] S. Yang dan G. Berdine, "Confusion Matrix," ., vol. 12, no. 53, hlm. 75–79, Okt 2024, doi: 10.12746/swrccc.v12i53.1391.
- [13] R. MV dan P. DR, "Blackbox Testing," *International Journal of Research Publication and Reviews*, vol. 5, no. 5, hlm. 5567–5570, Mei 2024, Diakses: 6 Juni 2025. [Daring]. Tersedia pada: ijrpr.com/uploads/V5ISSUE5/IJRPR27829.pdf