

Sistem Presensi Berbasis Kamera Smartphone Dengan Model Arsitektur MobileFaceNet Pada Siswa SMPN 1 Poncol Magetan

1st Machfud Ichsan Khoirudin

SI Teknologi Informasi
Telkom University

Surabaya, Indonesia
machfud

@student.telkomuniversity.ac.id

2nd Mustafa Kamal, S.Kom., M.Kom.

SI Teknologi Informasi
Telkom University

Surabaya, Indonesia
mustafakamal

@telkomuniversity.ac.id

3rd Yohanes Setiawan, S.Si., M.Kom.

SI Teknologi Informasi
Telkom University

Surabaya, Indonesia
yohanessetiawan

@telkomuniversity.ac.id

Abstrak — Sistem presensi siswa merupakan komponen penting dalam pengelolaan administrasi sekolah, namun metode manual seperti pemanggilan nama dan tanda tangan masih banyak digunakan dan rentan terhadap keterlambatan serta kecurangan. Penelitian ini bertujuan merancang dan mengimplementasikan sistem presensi berbasis pengenalan wajah menggunakan kamera smartphone dengan arsitektur MobileFaceNet yang dioptimalkan untuk perangkat mobile. Proses pengembangan mencakup pembangunan aplikasi Android menggunakan Jetpack Compose, CameraX, dan Google ML Kit untuk deteksi wajah secara real-time. Model MobileFaceNet dilatih menggunakan dataset wajah siswa kelas 8A dan dikonversi ke TensorFlow Lite untuk menghasilkan embedding wajah berukuran 192 dimensi. Proses verifikasi dilakukan menggunakan metode *cosine similarity* terhadap facebank yang tersimpan pada perangkat. Pengujian terhadap 32 siswa menunjukkan bahwa sistem bekerja dengan lancar pada perangkat berspesifikasi menengah, dengan nilai *threshold optimal* sebesar 0.6. Evaluasi System Usability Scale (SUS) menunjukkan bahwa aplikasi mudah digunakan dan antarmukanya intuitif bagi guru maupun siswa. Hasil penelitian ini membuktikan bahwa integrasi Jetpack Compose, CameraX, ML Kit, dan MobileFaceNet mampu menghasilkan sistem presensi yang efisien, akurat, dan layak diterapkan dalam lingkungan pendidikan.

Kata kunci— Presensi siswa, pengenalan wajah, mobilefacenet, cosine distance, tensorflow lite.

I. PENDAHULUAN

Presensi siswa merupakan komponen penting dalam sistem administrasi pendidikan karena berfungsi sebagai dasar pencatatan kehadiran, evaluasi kedisiplinan, serta bahan pertimbangan dalam penilaian akademik siswa. Data presensi juga memiliki peran strategis bagi pihak sekolah dalam penyusunan laporan dan pengambilan keputusan. Namun, pada praktiknya, sistem presensi yang masih banyak diterapkan di sekolah-sekolah di Indonesia umumnya bersifat manual, seperti pemanggilan nama atau tanda tangan pada lembar presensi. Metode ini relatif mudah diterapkan, tetapi memiliki sejumlah kelemahan, antara lain membutuhkan

waktu yang cukup lama, rentan terhadap kesalahan pencatatan, serta berpotensi menimbulkan manipulasi data kehadiran seperti praktik titip absen.[1]

Berbagai metode presensi alternatif telah dikembangkan untuk mengatasi keterbatasan presensi manual, seperti presensi berbasis kartu identitas, kode QR, dan biometrik. Presensi berbasis kartu maupun kode QR dinilai lebih efisien, namun masih memiliki celah dari sisi keamanan karena memungkinkan penggunaan oleh pihak lain. Presensi biometrik berbasis sidik jari menawarkan tingkat keamanan yang lebih baik, tetapi memerlukan kontak fisik dan perangkat khusus, sehingga kurang fleksibel serta berpotensi menimbulkan permasalahan higienitas. Berdasarkan perbandingan tersebut, presensi berbasis pengenalan wajah dinilai mampu mengatasi sebagian besar kelemahan metode presensi sebelumnya karena bersifat nirkontak, lebih aman, dan praktis untuk digunakan.

Dalam penelitian ini, dilakukan pelatihan model pengenalan wajah menggunakan arsitektur MobileFaceNet dengan memanfaatkan data wajah siswa sebagai dataset kustom. Model yang dihasilkan kemudian dikonversi ke dalam format TensorFlow Lite dan digunakan sebagai ekstraktor fitur wajah pada sistem presensi berbasis perangkat mobile. Pendekatan ini bertujuan untuk menghasilkan sistem pengenalan wajah yang ringan, efisien, serta mampu berjalan secara optimal pada perangkat smartphone dengan keterbatasan sumber daya, tanpa mengorbankan tingkat akurasi pengenalan wajah.

Pada penelitian sebelumnya, berbagai studi telah menunjukkan efektivitas penggunaan model ringan *MobileFaceNet* pada lingkungan dengan keterbatasan sumber daya. [2] mengoptimalkan arsitektur MobileFaceNet untuk lingkungan komputasi terbatas sehingga ukuran model dapat dikompresi menjadi sekitar 3,4 MB dengan tetap mempertahankan akurasi tinggi, yaitu mencapai 98,52% pada dataset LFW dan 97,54% pada VGGFace2. Penelitian lain oleh [3] mengimplementasikan MobileFaceNet dalam aplikasi presensi berbasis perangkat mobile menggunakan TensorFlow Lite dan memperoleh kinerja memori sebesar 84,5% serta tingkat generalisasi sekitar 70%. Selain itu, [4] menerapkan MobileFaceNet pada sistem akses kendaraan

berbasis pengenalan wajah dan melaporkan akurasi deteksi hingga 98,9%, dengan peningkatan akurasi pengenalan wajah pada beberapa dataset benchmark. Sementara itu, [5] mengembangkan sistem autentikasi wajah pada aplikasi Android berbasis ML Kit dengan tingkat akurasi sekitar 90%.

Teknologi pengenalan wajah (FRT) memiliki keunggulan dari sisi keamanan, kecepatan, higienitas, dan biaya dibandingkan metode biometrik lain.[6] FRT nirkontak, efisien, serta mampu memberikan tingkat keamanan tinggi pada basis data besar. Untuk sistem presensi berbasis smartphone, diperlukan model yang ringan dan akurat, salah satunya MobileFaceNet, yang dirancang khusus untuk perangkat mobile dengan sumber daya terbatas. Agar berjalan optimal, model dikonversi ke format TensorFlow Lite untuk memperkecil ukuran dan meningkatkan kecepatan inferensi, serta dikombinasikan dengan Google ML Kit untuk deteksi wajah secara real-time.[7] Dalam implementasinya, wajah yang terdeteksi diproses oleh MobileFaceNet untuk menghasilkan embedding wajah, kemudian dikenali dengan membandingkannya terhadap data tersimpan menggunakan cosine similarity, yang lebih sesuai untuk embedding ternormalisasi dibandingkan Euclidean Distance.[8]

Penelitian ini berfokus pada perancangan dan implementasi sistem presensi siswa berbasis kamera smartphone menggunakan algoritma pengenalan wajah MobileFaceNet, dengan tujuan mengevaluasi akurasi pengenalan wajah pada perangkat Android berspesifikasi RAM 8 GB dan penyimpanan 128 GB, efektivitas penyimpanan serta pengelolaan data presensi secara efisien dan real-time, serta tingkat kegunaan sistem berdasarkan metode System Usability Scale (SUS). Penelitian ini diharapkan memberikan manfaat bagi sekolah dan guru melalui peningkatan efisiensi, akurasi, dan otomatisasi pencatatan kehadiran, serta menjadi referensi akademik dalam pengembangan sistem face recognition berbasis perangkat mobile. Adapun batasan penelitian meliputi pelaksanaan di kelas 8A SMP Negeri 1 Poncol, penggunaan model MobileFaceNet berformat TFLite dengan data wajah siswa, pengujian pada smartphone Android dalam skenario nyata dengan wajah menghadap kamera, ruang lingkup sistem yang terbatas pada pendaftaran wajah, presensi, rekap data, serta evaluasi hasil, dan pencatatan presensi yang hanya mencakup status hadir dan tidak hadir.

II. KAJIAN TEORI

A. Pengenalan Wajah (*face Recognition*)

Pengenalan wajah mencakup berbagai teknologi untuk membangun sistem pengenalan wajah, seperti deteksi wajah, penentuan posisi wajah, identifikasi, dan pemrosesan gambar. Deteksi wajah adalah proses menemukan wajah dalam suatu gambar dengan memindai area tertentu. Hasilnya bisa berupa kotak persegi atau persegi panjang yang menunjukkan lokasi wajah. Posisi wajah adalah titik koordinat fitur wajah dalam gambar, seperti mata, hidung, dan mulut. Teknologi *deep learning* telah mengembangkan metode akurat untuk menentukan posisi ini. Kecepatan pemrosesan penentuan posisi wajah biasanya lebih cepat dibandingkan deteksi wajah, karena sistem hanya fokus pada fitur wajah yang sudah ditemukan.[9]

B. Cosine Similarity

Dalam bidang pengenalan wajah, cosine similarity merupakan metrik penting untuk mengukur kemiripan antara dua vektor fitur yang dihasilkan dari ekstraksi fitur menggunakan model jaringan saraf konvolusional (*Convolutional Neural Network*), seperti MobileFaceNet yang digunakan dalam sistem presensi ini. Metrik ini menghitung kesamaan berdasarkan sudut antara dua vektor dalam ruang berdimensi tinggi, sehingga sangat sesuai untuk fitur wajah yang telah dinormalisasi. Rumus cosine similarity didefinisikan sebagai berikut:

$$\text{cosine_similarity}(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \quad (1)$$

Dengan A dan B sebagai vektor fitur, $A \cdot B$ merupakan hasil perkalian titik, sedangkan $\|A\|$ dan $\|B\|$ menyatakan norma Euclidean dari masing-masing vektor. Nilai cosine similarity berada pada rentang -1 hingga 1 , di mana nilai 1 menunjukkan bahwa kedua vektor bersifat identik atau sangat mirip, nilai 0 menunjukkan tidak adanya korelasi, dan nilai -1 menunjukkan bahwa kedua vektor saling berlawanan atau sangat berbeda.

C. MobileFaceNet

MobileFaceNet merupakan arsitektur jaringan saraf konvolusional yang dirancang khusus untuk aplikasi pengenalan wajah pada perangkat dengan sumber daya terbatas. Arsitektur ini mengadopsi prinsip *inverted residual bottleneck* dari MobileNetV2 dengan memanfaatkan kombinasi *depthwise convolution* dan *pointwise convolution* untuk menekan kompleksitas komputasi.[2] Pendekatan tersebut memungkinkan proses ekstraksi fitur wajah tetap efektif dengan konsumsi memori dan jumlah operasi komputasi (FLOPs) yang lebih rendah dibandingkan arsitektur CNN konvensional. Meskipun memiliki ukuran model yang relatif kecil, sekitar 4 MB, dan jumlah parameter di bawah satu juta, MobileFaceNet tetap mampu memberikan performa yang kompetitif dalam menghasilkan representasi fitur wajah yang berbeda.

D. TensorFlow Lite

TensorFlow Lite merupakan versi ringan dari TensorFlow yang dikembangkan untuk menjalankan model *machine learning* secara lokal pada perangkat dengan keterbatasan sumber daya, seperti *smartphone*, sistem *embedded*, dan perangkat Internet of Things (IoT). Melalui mekanisme *on-device inference*, model dapat dijalankan tanpa memerlukan koneksi internet, sehingga latensi pemrosesan menjadi lebih rendah, privasi data pengguna lebih terjaga, serta respons aplikasi menjadi lebih cepat. Selain itu, TensorFlow Lite mendukung berbagai teknik optimasi, seperti *quantization* dan akselerasi perangkat keras, yang memungkinkan model berjalan secara *real-time* pada aplikasi Android dengan penggunaan sumber daya yang lebih efisien tanpa penurunan akurasi yang signifikan. [10]

E. System Usability Scale (SUS)

System Usability Scale (SUS) merupakan instrumen evaluasi berbentuk kuesioner yang digunakan untuk mengukur tingkat kegunaan (*usability*) suatu sistem komputer atau perangkat lunak berdasarkan persepsi dan penilaian subjektif pengguna. Metode SUS banyak digunakan karena bersifat sederhana, cepat, serta mampu memberikan gambaran umum mengenai kemudahan penggunaan, kenyamanan, dan penerimaan pengguna

terhadap sistem yang diuji. Kuesioner SUS terdiri dari 10 butir pertanyaan yang dirancang untuk mengevaluasi berbagai aspek usability, seperti kemudahan dipelajari, efisiensi penggunaan, dan kepercayaan pengguna terhadap sistem. Setiap pertanyaan dijawab menggunakan skala Likert dengan lima pilihan, mulai dari “Sangat Tidak Setuju” hingga “Sangat Setuju”, sehingga memungkinkan peneliti memperoleh data kuantitatif yang dapat dianalisis untuk menentukan tingkat usability sistem secara keseluruhan.[11]

III. METODE

Penelitian ini menggunakan pendekatan kuantitatif yang berfokus pada pengukuran data dan analisis numerik untuk mengevaluasi kinerja sistem yang dikembangkan. Pendekatan kuantitatif dipilih karena penelitian ini melibatkan proses pelatihan data, mulai dari pembuatan model hingga implementasi model MobileFaceNet pada sistem presensi, yang selanjutnya dievaluasi menggunakan metrik pengujian. Penelitian dilaksanakan di salah satu ruang kelas SMP Negeri 1 Poncol, Kabupaten Magetan. Tahapan penelitian yang dilakukan ditunjukkan pada Gambar 1.



GAMBAR 1
(METODE PENELITIAN)

A. Studi Literatur

Tahap awal penelitian dilakukan dengan studi literatur terkait berbagai metode sistem presensi yang umum digunakan serta penelitian-penelitian terdahulu yang relevan. Studi literatur ini bertujuan sebagai dasar teoritis dan referensi dalam menentukan pendekatan serta metode yang digunakan dalam penelitian.

B. Identifikasi masalah

Pada tahap ini dilakukan identifikasi terhadap proses presensi yang berlangsung di SMP Negeri 1 Poncol. Berdasarkan hasil observasi, sistem presensi siswa masih dilakukan secara manual dengan cara pemanggilan nama satu per satu sesuai daftar pada lembar presensi. Metode tersebut memiliki kelemahan, terutama dari segi efisiensi waktu dan keamanan data kehadiran.

C. Analisis kebutuhan

Tahap analisis kebutuhan dilakukan untuk menentukan kebutuhan sistem, meliputi kebutuhan dataset wajah siswa, algoritma pengenalan wajah menggunakan MobileFaceNet, serta kebutuhan perangkat lunak dalam pengembangan aplikasi presensi agar sistem dapat berjalan secara optimal.

D. Pembuatan model

Pada tahap ini, dataset wajah yang telah diperoleh diolah melalui proses *preprocessing*, yang meliputi konversi citra ke format RGB, pemotongan citra (*cropping*), dan perubahan ukuran citra (*resize*). Selanjutnya, model dilatih menggunakan variasi nilai *batch size* dan *learning rate* untuk memperoleh model dengan performa terbaik yang akan digunakan pada tahap selanjutnya.

E. Desain sistem

Tahap desain sistem meliputi perancangan arsitektur sistem dan basis data aplikasi presensi. Basis data yang digunakan adalah SQLite dengan memanfaatkan plugin Room. Struktur basis data yang digunakan terdiri dari beberapa entitas sebagai berikut.

TABEL 1
(TIPE DATA KELAS)

No	Atribut	Tipe data
1	class_id	integer
2	name	string

Tabel 1 menunjukkan struktur entitas kelas yang terdiri dari dua atribut, yaitu class_id dan name.

TABEL 2
(TIPE DATA SISWA)

No	atribut	Tiipe data
1	student_id	integer
2	name	string

Tabel 2 menunjukkan struktur entitas siswa yang terdiri dari dua atribut, yaitu student_id dan name.

TABEL 3
(TIPE DATA LOG KEHADIRAN)

No	Atribut	Tipe data
1	log_id	Integer
2	tipePresensi	Integer
3	timestamp	Long
4	faceImagePath	String
5	method	String
6	scoreDistance	Float
7	fps	float
8	latency	float

Tabel 3 menunjukkan struktur entitas log kehadiran yang terdiri dari delapan atribut utama yang digunakan dalam proses pencatatan presensi siswa.

F. Pengembangan aplikasi

Pada tahap ini dilakukan pengembangan aplikasi presensi yang difokuskan pada platform android, aplikasi dibuat menggunakan bahasa pemrograman Kotlin dengan pendekatan antarmuka Jetpack Compose serta menggunakan Android Studio sebagai lingkungan pengembangan.

G. Integrasi Model

Model terbaik yang diperoleh dari hasil pelatihan selanjutnya dikonversi ke dalam format TensorFlow Lite. Konversi ini bertujuan untuk memperkecil ukuran model dan mempermudah proses integrasi dengan aplikasi. Proses integrasi model ke dalam aplikasi dilakukan menggunakan *plugin* TensorFlow Lite.

H. Pengujian sistem

Pada tahap pengujian sistem, aplikasi presensi diuji secara langsung pada siswa di dalam kelas. Pengujian ini bertujuan untuk memastikan bahwa aplikasi dan model pengenalan wajah dapat berjalan dengan baik pada kondisi penggunaan nyata.

I. Evaluasi

Tahap evaluasi dilakukan melalui dua pendekatan, yaitu analisis hasil pengujian sistem dan penyebaran kuesioner kepada beberapa pengajar. Evaluasi ini bertujuan untuk menilai kinerja sistem serta tingkat kelayakan dan kemudahan penggunaan sistem presensi berbasis pengenalan wajah.

IV. HASIL DAN PEMBAHASAN

A. Pelatihan Model

Pada tahap pelatihan, model MobileFaceNet dilatih menggunakan beberapa variasi nilai *batch size* dan *learning*

rate untuk memperoleh konfigurasi model dengan performa terbaik. Variasi *batch size* yang digunakan meliputi 8, 16, 32, 64, dan 128, sedangkan nilai *learning rate* yang diterapkan adalah 0,005; 0,001; dan 0,0001. Evaluasi performa model dilakukan berdasarkan nilai *validation accuracy* dan *F1-score* pada setiap konfigurasi pelatihan.

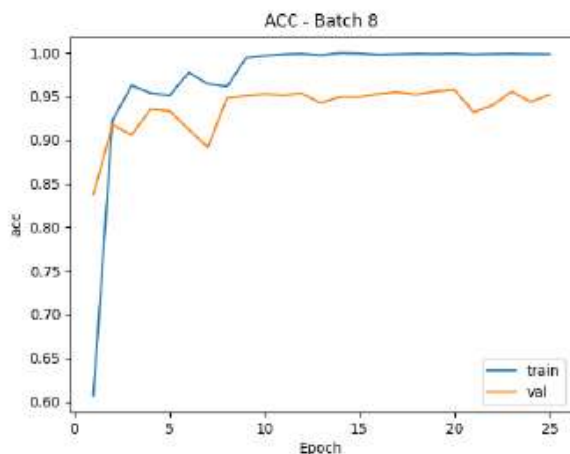
Hasil pelatihan model MobileFaceNet dengan berbagai kombinasi *batch size* dan *learning rate* dirangkum pada Tabel 4.

TABEL 4
(RANGKUMAN HASIL PELATIHAN MODEL MOBILEFACENET)

No	Batch size	Learning Rate	Best Epoch	Best Val Acc	F1-Score
1	8	0.005	5	0.9167	0.9114
2	8	0.001	20	0.9479	0.9552
3	8	0.0001	16	0.9444	0.9453
4	16	0.005	12	0.9604	0.9503
5	16	0.001	13	0.9306	0.9275
6	16	0.0001	20	0.9374	0.9326
7	32	0.005	11	0.9396	0.9441
8	32	0.001	16	0.9340	0.9242
9	32	0.0001	29	0.8979	0.8881
10	64	0.005	16	0.9319	0.9155
11	64	0.001	8	0.9153	0.9214
12	64	0.0001	28	0.7597	0.7254
13	128	0.005	5	0.9236	0.9198
14	128	0.001	12	0.8951	0.9014
15	128	0.0001	30	0.5507	0.5251

Berdasarkan hasil yang ditunjukkan pada Tabel 4, dapat diamati bahwa pemilihan nilai *batch size* dan *learning rate* sangat berpengaruh terhadap performa model. Secara umum, penggunaan *learning rate* yang terlalu kecil, khususnya 0,0001, cenderung menghasilkan penurunan akurasi dan nilai F1-score, terutama pada *batch size* yang besar. Hal ini disebabkan oleh proses pembaruan bobot yang berjalan terlalu lambat sehingga model sulit mencapai konvergensi optimal.

Berdasarkan konfigurasi terbaik yang diperoleh, analisis lebih lanjut dilakukan melalui pengamatan kurva akurasi pelatihan dan validasi, sebagaimana ditunjukkan pada Gambar 2.

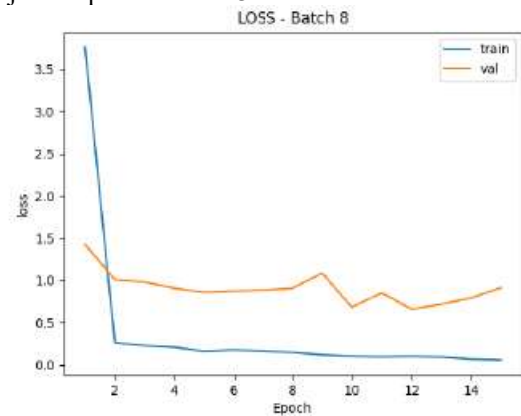


GAMBAR 2
(KURVA AKURASI PADA MODEL TERBAIK)

Gambar 2 menunjukkan kurva akurasi pelatihan dan validasi pada konfigurasi terbaik dengan *batch size* 8 dan *learning rate* 0,001. Akurasi pelatihan meningkat secara cepat pada beberapa epoch awal dan stabil mendekati nilai maksimum pada epoch selanjutnya. Akurasi validasi juga menunjukkan tren peningkatan yang stabil pada kisaran 0,94 dengan fluktuasi yang relatif kecil. Selisih antara akurasi

pelatihan dan validasi masih berada dalam batas wajar, sehingga menunjukkan bahwa model memiliki kemampuan generalisasi yang baik tanpa indikasi overfitting yang signifikan.

Selain kurva akurasi, evaluasi kinerja model juga dilakukan melalui pengamatan kurva loss, sebagaimana ditunjukkan pada Gambar 3.



GAMBAR 3
(KURVA LOSS PADA BEST MODEL)

Gambar 3 memperlihatkan kurva loss pelatihan dan validasi pada konfigurasi *batch size* 8 dan *learning rate* 0,001. Nilai loss pelatihan menurun secara signifikan pada awal pelatihan dan terus stabil hingga akhir epoch. Sementara itu, loss validasi menunjukkan penurunan yang lebih bertahap dengan fluktuasi kecil yang masih dapat diterima. Hasil ini mengindikasikan bahwa proses pelatihan berjalan secara stabil dan konfigurasi yang digunakan menghasilkan model yang optimal untuk diimplementasikan pada sistem presensi siswa.

Model terbaik yang diperoleh dari hasil pelatihan selanjutnya dikonversi ke dalam format TensorFlow Lite untuk mempermudah proses integrasi dengan aplikasi presensi berbasis Android.

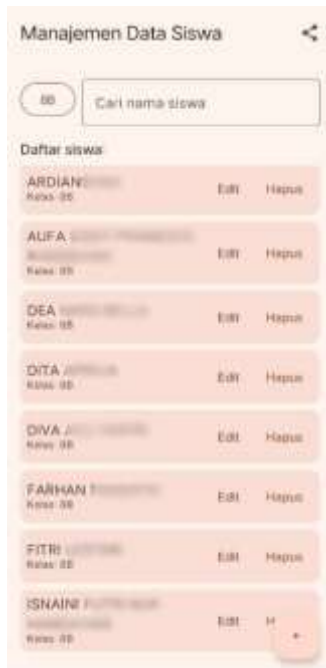
B. Implementasi Presensi



GAMBAR 4
(TAMPILAN HALAMAN UTAMA)

Gambar 4 menunjukkan halaman utama aplikasi presensi siswa berbasis pengenalan wajah. Pada halaman ini tersedia

empat menu utama, yaitu Data Siswa, Presensi, Rekap Data, dan Ekspor File. Menu Data Siswa digunakan untuk pengelolaan data siswa beserta data wajah (*facebank*) yang berfungsi sebagai data pembandingan dalam proses presensi. Menu Presensi menyediakan fitur presensi otomatis berbasis pengenalan wajah menggunakan model terbaik hasil pelatihan, serta presensi manual sebagai alternatif. Menu Rekap Data menampilkan seluruh data kehadiran siswa yang dikelompokkan berdasarkan kelas dan waktu presensi. Selanjutnya, menu Ekspor File memungkinkan hasil data presensi diekspor untuk dibagikan atau digunakan pada aplikasi lain.



GAMBAR 5
(TAMPILAN MANAJEMEN DATA SISWA)

Gambar 5 menunjukkan fitur manajemen data siswa yang memungkinkan guru atau administrator untuk menambahkan, mengedit, dan menghapus data siswa. Data yang dikelola pada fitur ini mencakup atribut utama berupa nama siswa dan kelas, yang digunakan sebagai identitas dasar dalam sistem presensi.



GAMBAR 6
(TAMPILAN MENAMBAHKAN DATA FACEBANK)

Gambar 6 menunjukkan proses pengambilan data wajah siswa pada aplikasi presensi. Pada tahap ini, citra wajah siswa direkam dan diekstraksi menjadi vektor fitur menggunakan model pengenalan wajah yang telah diintegrasikan dalam sistem. Hasil ekstraksi tersebut disimpan dalam format JSON dengan representasi vektor berdimensi 192 untuk setiap siswa sebagai data pembandingan dalam proses presensi.



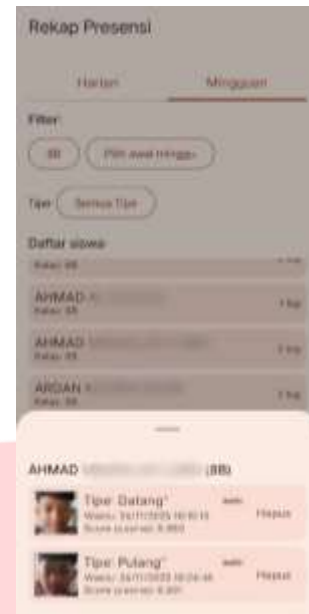
GAMBAR 7
(GAMBAR PRESENSI OTOMATIS)

Gambar 7 menunjukkan proses presensi dengan metode *face recognition* menggunakan model *mobilefacenet*, data wajah yang telah diekstraksi kemudian dibandingkan dengan metode *cosine similarity* untuk mendapatkan perbandingan terdekat, jika wajah terdeteksi maka akan disimpan pada log presensi siswa.



GAMBAR 8
(TAMPILAN INPUT PRESENSI MANUAL)

Pada gambar 8 menunjukkan fitur presensi manual yang disediakan sebagai alternatif apabila proses presensi otomatis berbasis pengenalan wajah tidak dapat dilakukan. Pada fitur ini, proses presensi dilakukan tanpa melibatkan metode pengenalan wajah, sehingga memungkinkan guru untuk tetap mencatat kehadiran siswa dalam kondisi tertentu.



GAMBAR 10
(DETAIL PRESENSI SISWA)

Gambar 10 menunjukkan tampilan detail rekap presensi siswa pada menu rekap data. Setiap entri presensi menampilkan informasi berupa tipe kehadiran, waktu presensi, serta nilai cosine similarity sebagai hasil pencocokan wajah. Pada menu ini juga tersedia fitur hapus yang digunakan untuk mengoreksi data presensi apabila terjadi duplikasi atau kesalahan pencatatan oleh sistem.



GAMBAR 9
(REKAP DATA PRESENSI OTOMATIS)

Pada gambar 9 menunjukkan tampilan rekapitulasi data presensi siswa yang menyajikan informasi kehadiran berdasarkan data yang diperoleh dari presensi otomatis maupun manual. Seluruh data presensi disimpan dalam database dan ditampilkan pada menu rekap, di mana pengguna dapat melakukan penyaringan data berdasarkan kelas, periode waktu, dan jenis presensi yang dipilih.



GAMBAR 11
(TAMPILAN MENU EKSPOR DATA PRESENSI)

Gambar 11 menunjukkan fitur ekspor data presensi yang memungkinkan data kehadiran siswa diekspor ke dalam dua format, yaitu CSV dan PDF. Pada format PDF, data presensi dapat disertai dengan foto hasil presensi, sedangkan format CSV disajikan tanpa foto. Selain itu, fitur ini dilengkapi dengan mekanisme penyaringan data berdasarkan kriteria tertentu sehingga guru atau administrator dapat menyesuaikan data yang akan diekspor sesuai kebutuhan.

C. Skor SUS

Evaluasi tingkat kegunaan sistem dilakukan menggunakan metode System Usability Scale (SUS) dengan melibatkan 11 staf pengajar sebagai responden. Setiap responden diminta untuk mengisi kuesioner SUS yang terdiri dari 10 pernyataan dengan skala penilaian 1 hingga 5. Hasil pengisian kuesioner oleh seluruh responden ditunjukkan pada Tabel 5.

TABEL 5
(NILAI SUS TERHADAP 11 RESPONDEN)

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	4	2	5	1	4	2	5	1	4	2
R2	5	1	4	2	5	1	4	2	5	1
R3	5	2	5	1	4	2	5	1	4	2
R4	5	1	4	2	5	1	4	2	5	1
R5	4	2	5	1	4	2	5	1	4	2
R6	5	1	4	2	5	1	4	2	5	1
R7	4	2	5	1	4	2	5	1	4	2
R8	5	1	4	2	5	1	4	2	5	1
R9	4	2	5	1	4	2	5	1	4	2
R10	5	1	4	2	5	1	4	2	5	1
R11	4	2	5	1	4	2	4	2	4	2

Selanjutnya, nilai kuesioner disesuaikan sesuai dengan metode perhitungan SUS, di mana untuk pernyataan bernomor ganjil (Q1, Q3, Q5, Q7, dan Q9) nilai dihitung dengan cara mengurangi jawaban responden dengan angka 1, sedangkan untuk pernyataan bernomor genap (Q2, Q4, Q6, Q8, dan Q10) nilai dihitung dengan cara mengurangkan jawaban dari angka 5. Hasil penyesuaian nilai SUS ditunjukkan pada Tabel 6.

TABEL 6
(NILAI SUS YANG SUDAH DISESUAIKAN)

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	3	3	4	4	3	3	4	4	3	3
R2	4	4	3	3	4	4	3	3	4	4
R3	4	3	4	4	3	3	4	4	3	3
R4	4	4	3	3	4	4	3	3	4	4
R5	3	3	4	4	3	3	4	4	3	3
R6	4	4	3	3	4	4	3	3	4	4
R7	3	3	4	4	3	3	4	4	3	3
R8	4	4	3	3	4	4	3	3	4	4
R9	3	3	4	4	3	3	4	4	3	3
R10	4	4	3	3	4	4	3	3	4	4
R11	3	3	4	4	3	3	3	3	3	3

Setelah dilakukan penyesuaian, seluruh nilai pada masing-masing responden dijumlahkan, kemudian dikalikan dengan faktor 2,5 untuk memperoleh skor akhir SUS. Hasil perhitungan skor SUS untuk setiap responden disajikan pada Tabel 7.

TABEL 7
(HASIL PENJUMLAHAN TIAP RESPONDEN)

	Hasil Penjumlahan	Nilai x 2,5
R1	34	85
R2	36	90
R3	35	87,5
R4	36	90
R5	34	85
R6	36	90
R7	34	85
R8	36	90
R9	34	85
R10	36	90
R11	32	80

Berdasarkan hasil perhitungan pada Tabel 7, diperoleh skor SUS terendah sebesar 80 dan skor tertinggi sebesar 90. Nilai rata-rata skor SUS dari seluruh responden adalah sebesar 87,04. Berdasarkan interpretasi skor SUS, nilai tersebut menunjukkan bahwa sistem presensi siswa berbasis pengenalan wajah yang dikembangkan memiliki tingkat kegunaan yang baik dan dapat diterima oleh pengguna.

V. KESIMPULAN

Berdasarkan rangkaian proses perancangan, implementasi, pelatihan model, pengujian, serta evaluasi terhadap sistem presensi siswa berbasis pengenalan wajah menggunakan MobileFaceNet, dapat disimpulkan bahwa sistem berhasil diimplementasikan dengan baik menggunakan Jetpack Compose, Google ML Kit, dan model MobileFaceNet dalam format TensorFlow Lite. Implementasi ini memungkinkan sistem berjalan pada perangkat dengan spesifikasi rendah dengan tingkat latensi yang minimal. Dari sisi teknis, model MobileFaceNet menunjukkan performa yang optimal pada konfigurasi pelatihan dengan batch sebesar 8 dan learning rate sebesar 0,001, yang menghasilkan tingkat akurasi sebesar 0,94. Selain itu, pengujian terhadap nilai threshold menunjukkan bahwa nilai 0,6 merupakan parameter terbaik untuk melakukan pencocokan wajah pada sistem presensi siswa. Terakhir, berdasarkan evaluasi menggunakan metode System Usability Scale (SUS), diperoleh skor rata-rata sebesar 78,18 yang mengindikasikan bahwa sistem berada pada kategori layak dan mudah dioperasikan oleh guru maupun siswa tanpa kendala yang berarti.

REFERENSI

- [1] Y. Irfayanti, E. K. Wijayanti, P. Studi, S. Informasi, P. Studi, and T. Informatika, "PADA SMA DUA MEI CIPUTAT MENGGUNAKAN METODE HAAR-CASCADE CLASSIFIER BERBASIS," vol. 5, no. 1, pp. 551–565, 2024.
- [2] J. Xiao, G. Jiang, and H. Liu, "EAI Endorsed Transactions A Lightweight Face Recognition Model based on MobileFaceNet for Limited Computation Environment," vol. 7, no. 27, pp. 1–9, 2022.
- [3] R. Setiawan, N. Pradana, T. Prakisy, and R. A. Yuana, "JIPTEK : Jurnal Ilmiah Pendidikan Teknik dan Kejuruan," vol. 17, no. 1, pp. 36–48, 2024.
- [4] W. Fang and Z. Yu, "Automobile Entry System Based on Face Recognition," vol. 10, no. 1, pp. 573–585, 2025.
- [5] A. S. Ibnu Saputra, Teti Desyani, "Implementasi login aplikasi dengan fitur autentikasi pengguna menggunakan flutter dan ml kit face recognition," vol. 3, no. 1, pp. 106–115, 2025.
- [6] Jan Lunter, "Everyday Biometrics: Can Face Replace Fingerprint Recognition," *Biometric Technol. Today MA Bussines London*, vol. 4, pp. 7–10, 2021.
- [7] N. P. Fajar Abdillah Ahmad, "Implementasi Pengenalan Wajah , Deteksi Kehadiran , dan Geolokasi menggunakan TensorFlow Lite dan Google ML Kit pada Aplikasi Absensi Mobile Implementation of Face Recognition , Attendance Detection , and Geolocation," vol. 14, pp. 172–186, 2025.
- [8] A. Yadav, D. Thakran, and R. Gupta, "Real - Time

- Image Processing using Flutter and Tflite Packages,” no. 5, pp. 38–40, 2021.
- [9] L. Li, X. Mu, S. Li, and H. Peng, “A Review of Face Recognition Technology,” *IEEE Access*, vol. 8, pp. 139110–139120, 2020, doi: 10.1109/ACCESS.2020.3011028.
- [10] H. Nurhuda, “Masalah-Masalah Pendidikan Nasional,” *J. Pemikir. dan Pendidik. Dasar*, p. 130, 2022.
- [11] R. A. Putri, S. Kom, and M. Ti, *Monograf Evaluasi Usability Microsoft Teams Menggunakan System Usability Scale*.

