

Implementasi YOLOv8 dan FaceNet untuk Sistem Keamanan Real-Time Berbasis IoT

Ery Kurniawan¹, Rifqi Rahmandhani², Dimas Rizkyansyah³, Ika Kurniawati^{4*},
Waeisul Bismi⁵, Riza Fahlapi⁶

^{1,2,3,5,6}Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika
Jl. Kramat Raya No. 98 Jakarta, Indonesia

⁴Fakultas Teknologi Informasi, Universitas Nusa Mandiri
Jl. Jatiwaringin No. 2, Cipinang Melayu, Jakarta, Indonesia

erykurniawan919@gmail.com¹, rifqirahmandhani09@gmail.com²,
dimasrizkiyah721@gmail.com³, ika.iki@bsi.ac.id⁴, waeisul.wbn@bsi.ac.id⁵,
riza.rzf@bsi.ac.id⁶

*Penulis Korespondensi

ABSTRAK

Sistem keamanan CCTV konvensional umumnya hanya berfungsi sebagai perekam pasif tanpa kemampuan analisis otomatis, yang menyebabkan keterlambatan deteksi karena proses identifikasi dilakukan secara manual. Keterbatasan ini menimbulkan latensi tinggi dan akurasi deteksi yang rendah, sehingga menjadi masalah krusial dalam kebutuhan keamanan modern. Sistem keamanan yang baik dapat mencegah tindak kejahatan yang bisa merugikan penghuni rumah baik fisik maupun materiil. Penelitian ini mengusulkan pengembangan sistem keamanan cerdas berbasis Internet Of Things (IoT) dengan integrasi deteksi wajah menggunakan YOLOv8 dan pengenalan wajah FaceNet menggunakan modul ESP32-CAM. Sistem ini dirancang untuk mendeteksi wajah secara real-time, identifikasi individu secara otomatis, serta pengiriman notifikasi instan melalui Telegram ketika terdeteksi wajah yang tidak dikenal. Metode penelitian ini meliputi perancangan arsitektur IoT, pengambilan dataset wajah, preprocessing menggunakan MTCNN, FaceNet untuk menghasilkan facial embeddings, serta implementasi YOLOv8 sebagai detektor wajah real-time. Evaluasi kinerja pengenalan wajah dilakukan dengan menerapkan metode 5-fold cross-validation pada dataset embedding FaceNet menggunakan pengklasifikasi k-NN. Hasil eksperimen menunjukkan bahwa sistem mampu mendeteksi wajah dengan tingkat respon tinggi dan mengenali individu dengan akurasi yang konsisten pada pencahayaan dan jarak bervariasi. Hasil pengujian training rata-rata accuracy Top-1 mencapai 0.96 dan rata-rata accuracy Top-5 sebesar 0.99, YOLOv8 menunjukkan kemampuan deteksi wajah yang akurat dan cepat dengan waktu respon 1,86 detik pada server berbasis CPU Intel Core i5 dan GPU Intel UHD Graphics 620. Performa pengujian akurasi FaceNet dengan pengklasifikasi k-NN menghasilkan akurasi 99.35%, presisi 99.35%, recall 98,94%, F1-score 99,11%, dan FPR (False Positive Rate) 0,08%, hal ini menunjukkan bahwa sistem memiliki akurasi pengenalan wajah yang sangat tinggi dan konsisten. Sistem yang dikembangkan mampu memberikan peringatan instan kepada pengguna melalui Telegram saat terdeteksi wajah yang tidak dikenal, sehingga meningkatkan waktu respons terhadap potensi ancaman. Dengan performa yang stabil dan tangguh serta biaya implementasi yang rendah, sistem ini menawarkan solusi keamanan modern yang lebih adaptif, proaktif, efektif, dan efisien dibandingkan CCTV konvensional.

Article Info

Kata Kunci:

CCTV
ESP32-CAM
Deteksi Wajah
FaceNet
Internet Of Things (IoT)
YOLOv8

Riwayat artikel:

Submit 1 Des 2025
Revisi 10 Des 2025
Diterima 31 Des 2025

1. PENDAHULUAN

Keamanan merupakan aspek penting dalam berbagai lingkungan seperti rumah, kantor, maupun fasilitas publik. Sistem pengawasan berbasis Closed Circuit Television (CCTV) telah lama digunakan untuk memantau aktivitas manusia, namun sebagian besar sistem tersebut masih bersifat pasif dan membutuhkan pengawasan manual, tanpa kemampuan analisis otomatis. Kondisi ini menyebabkan efektivitas pengawasan menjadi terbatas, terutama ketika dibutuhkan respons cepat terhadap objek atau individu tertentu. Selain itu, sistem CCTV konvensional memiliki kelemahan utama berupa latensi tinggi dan akurasi rendah karena tidak adanya pemrosesan pengenalan otomatis atau kecerdasan artifisial secara langsung. Hal ini membuat deteksi ancaman sering terlambat dan tidak adaptif terhadap kondisi lingkungan.

Perkembangan Internet Of Things (IoT) dan Artificial Intelligence (AI), khususnya Deep Learning, telah membuka peluang untuk mengembangkan sistem keamanan yang lebih cerdas dan responsif. Teknologi deteksi objek dan pengenalan wajah kini dapat bekerja secara real-time dengan tingkat akurasi yang tinggi. Beberapa penelitian sebelumnya telah memanfaatkan ESP32-CAM sebagai perangkat pemantau berbasis IoT, namun sebagian besar masih terbatas pada fungsi streaming atau deteksi sederhana tanpa integrasi penuh algoritma deteksi modern seperti YOLO (You Only Look Once) dan metode pengenalan wajah berbasis deep learning [1][2]. Penelitian lain juga menunjukkan bahwa kemampuan pengenalan wajah akan meningkat signifikan ketika menggunakan model berbasis embedding seperti FaceNet, dibandingkan metode konvensional [3].

FaceNet adalah deep embedding model yang menghasilkan representasi wajah di ruang vektor yang sangat efisien untuk face recognition, dan telah mencapai state-of-the-art akurasi pada benchmark dikenal seperti LFW (Labeled Faces in the Wild) [4]. FaceNet memiliki representasi yang jauh lebih kuat daripada metode deteksi/klasifikasi klasik seperti Haar Cascade atau LBPH (Local Binary Pattern Histogram), yang biasanya hanya berbasis hand-crafted features dan cenderung kurang tahan terhadap kondisi variatif (pose, pencahayaan, occlusion). Studi yang dilakukan oleh [5] Haar dan YOLOv8 menjadi pilihan paling sesuai karena menawarkan keseimbangan antara akurasi, efisiensi, dan ukuran model. Haar ideal untuk skenario yang mengutamakan akurasi di lingkungan terkontrol, sedangkan YOLOv8 unggul dalam aplikasi real-time. Studi tersebut juga menegaskan pentingnya ukuran model untuk penerapan pada perangkat dengan sumber daya komputasi terbatas, dengan YOLOv8 muncul sebagai pilihan yang menarik berkat ukurannya yang ringkas dan akurasinya yang tinggi. Pada penelitian lain metode FaceNet menunjukkan kinerja terbaik dengan akurasi tertinggi dan FPR terendah melampaui Haar Cascade dan OpenFace pada dataset LFW dan FRGC (Face Recognition Grand Challenge), sehingga menjadikannya pilihan paling tepat untuk aplikasi yang membutuhkan pengenalan wajah yang akurat dan andal [6]. Studi lain menyebutkan bahwa model YOLOv8 seringkali mampu mempertahankan akurasi tinggi, menjalankan deteksi real-time dengan FPS (Frames Per Second) tinggi, memiliki footprint komputasi yang lebih terkendali dibanding model lain yang lebih kompleks. Performa YOLOv8n mencapai 94% precision, 92% recall, dan 92.9% F1-score, unggul dibandingkan MobileNetSSD yang tercatat di 85.2% pada F1-score [7]. Selanjutnya melalui studi yang dilakukan oleh [8] CNN/FaceNet secara konsisten mengungguli metode tradisional pada tolok ukur standar. Model deep learning juga sering menggabungkan face detection + embedding untuk real-world performance.

Meskipun metode deteksi dan pengenalan wajah klasik seperti Haar Cascade + LBPH atau MobileNet SSD bersifat ringan dan memiliki biaya komputasi yang rendah, metode tersebut sering menunjukkan tingkat ketangguhan dan akurasi yang lebih rendah pada lingkungan tidak terkontrol dengan variasi pose, pencahayaan, atau oklusi. Sebaliknya, model deep learning seperti YOLOv8 dan FaceNet telah terbukti mencapai kinerja yang secara konsisten lebih tinggi pada tolok ukur standar, di mana YOLOv8 menawarkan keseimbangan yang baik antara kecepatan inferensi dan akurasi deteksi, serta FaceNet mencapai kinerja pengenalan yang mendekati state-of-the-art (akurasi mencapai 99% pada dataset LFW). Sehingga kombinasi model yang dipilih memberikan kinerja real-time yang unggul sekaligus tetap layak untuk diterapkan pada sistem IoT dengan sumber daya terbatas.

Namun demikian, sebagian besar penelitian terdahulu umumnya hanya menggabungkan satu jenis model, misalnya hanya pengenalan wajah tanpa deteksi objek real-time, atau hanya deteksi objek tanpa identifikasi wajah secara individual. Hampir tidak ada penelitian yang mengintegrasikan algoritma YOLO sebagai face detection bersama model FaceNet sebagai face recognition dalam satu sistem keamanan CCTV berbasis IoT menggunakan ESP32-CAM. Padahal kombinasi kedua metode tersebut berpotensi menghasilkan sistem keamanan yang jauh lebih akurat, cepat, dan efisien pada kondisi lingkup nyata. Selain itu, sebagian penelitian sebelumnya mencoba melakukan pemrosesan langsung di perangkat (edge processing), namun ESP32-CAM memiliki kapasitas memori dan kemampuan komputasi yang relatif rendah sehingga tidak cocok untuk menjalankan model deep learning besar seperti YOLOv8 dan FaceNet tanpa optimasi perangkat eksternal [9]. Oleh karena itu, arsitektur client-server yang memindahkan proses inferensi ke server Python merupakan solusi yang lebih memungkinkan, sekaligus memberikan fleksibilitas dalam pengelolaan data, peningkatan model, dan tampilan monitoring.

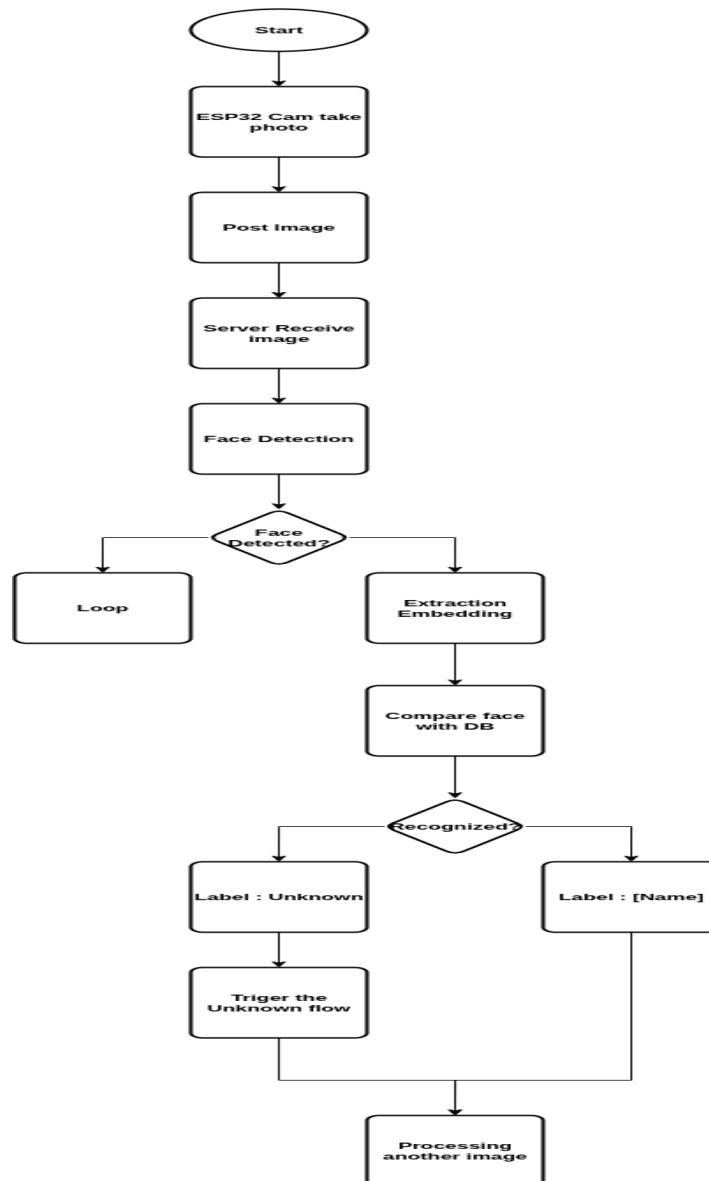
Penelitian ini mengusulkan pengembangan sistem keamanan CCTV cerdas berbasis ESP32-CAM yang mengintegrasikan algoritma YOLOv8 untuk deteksi wajah secara real-time dan FaceNet untuk pengenalan identitas

wajah. Sistem dikombinasikan dengan server Python untuk memproses citra, menampilkan hasil deteksi melalui web dashboard, serta mengirimkan notifikasi otomatis Telegram apabila terdeteksi wajah yang tidak dikenal, untuk meningkatkan respons terhadap potensi ancaman. Integrasi tiga komponen utama IoT, deteksi objek, dan pengenalan wajah menjadi keunggulan utama yang membedakan penelitian ini dari penelitian sebelumnya. Selain itu, penelitian ini juga mengembangkan koneksi antara perangkat dan server Python guna mendukung pemrosesan data serta menyediakan dashboard monitoring yang informatif. Tujuan utama dari penelitian ini adalah mengotomatisasi proses pemantauan keamanan dengan menerapkan teknologi computer vision dan IoT secara terintegrasi, sehingga sistem mampu melakukan deteksi dan identifikasi wajah secara real-time. Penelitian ini mengevaluasi kinerja sistem melalui aspek akurasi deteksi, kecepatan pemrosesan, dan evaluasi perangkat ESP32-CAM dan server melalui network latency. Hasil penelitian ini diharapkan dapat menghadirkan solusi sistem keamanan berbasis IoT yang lebih proaktif dan efisien dibandingkan sistem keamanan konvensional.

2. METODE PENELITIAN

2.1 Dataset dan Prapemrosesan

Metodologi penelitian ini dirancang untuk mengembangkan sistem keamanan berbasis ESP32-CAM yang melakukan deteksi wajah menggunakan YOLOv8 dan pengenalan identitas wajah menggunakan FaceNet secara real-time sebagaimana ditunjukkan pada Gambar 1. Metode ini mencakup serangkaian tahapan yang saling berkaitan untuk memastikan sistem dapat bekerja secara akurat dan optimal, mulai dari pengumpulan data hingga evaluasi akhir.

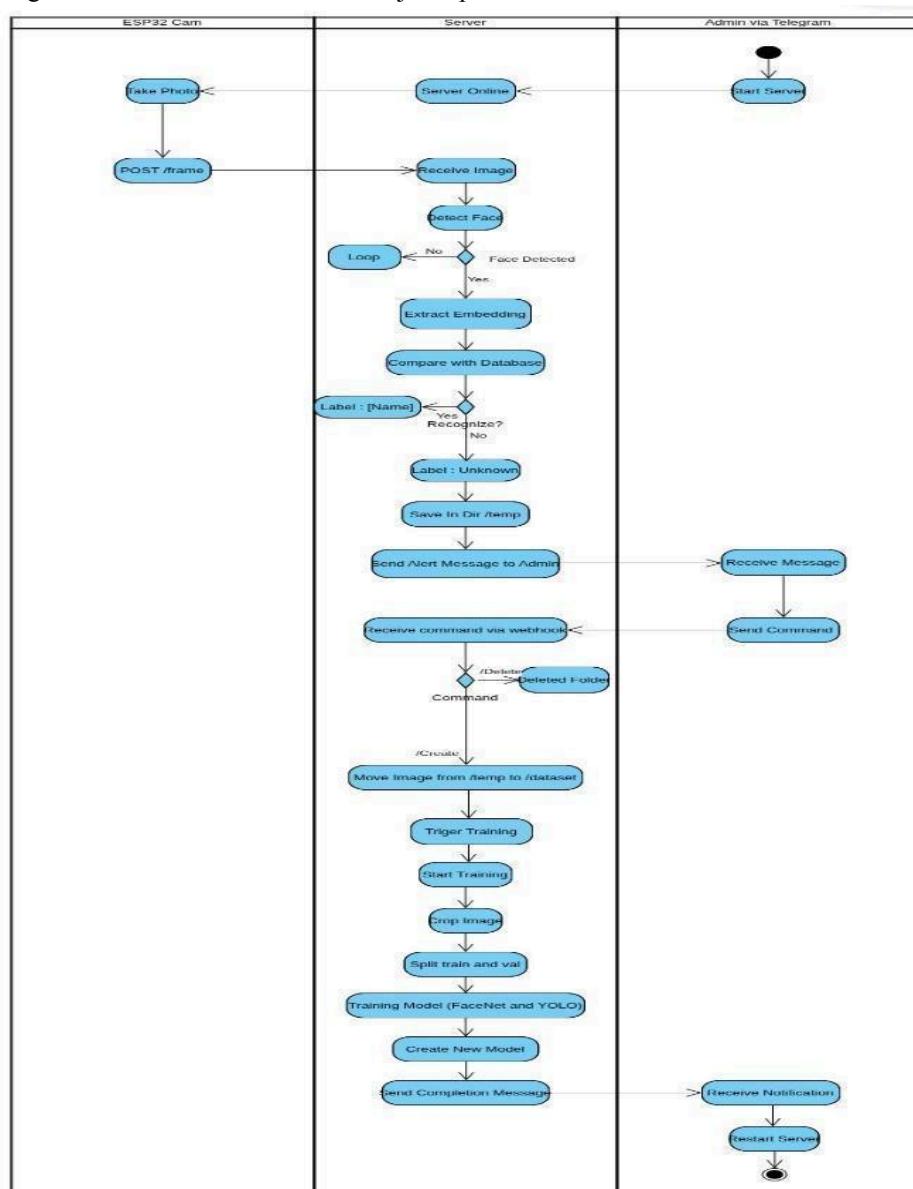


Gambar 1. Alur kerja sistem

Tahapan metodologi penelitian ini dijelaskan secara komprehensif melalui diagram alir sistem. Tahap awal dimulai dengan proses server menerima citra mentah yang dikirimkan oleh perangkat ESP32-CAM. Citra tersebut kemudian diproses menggunakan YOLOv8 yang berperan sebagai pendekripsi objek canggih yang mampu mengidentifikasi wajah secara efisien dan akurat dalam scenario real-time. Wajah yang terdeteksi kemudian dipotong (cropped faces) dan diproses sebelum diteruskan ke model FaceNet untuk proses ekstraksi embedding. Model FaceNet digunakan untuk pengenalan wajah melalui representasi vector fitur berdimensi tinggi [10], yang selanjutnya dibandingkan dengan basis data wajah terdaftar untuk menentukan identitas individu. Jika wajah yang terdeteksi tidak ditemukan dalam basis data (unknown), sistem secara otomatis akan mengirimkan notifikasi peringatan ke Telegram. Hasil deteksi dan pengenalan wajah akan ditampilkan pada dashboard monitoring untuk proses pengawasan secara real-time.

2.2 Perancangan Sistem

Perancangan sistem difokuskan pada integrasi antara perangkat IoT, modul pemrosesan berbasis server, dan antarmuka pengguna untuk mendukung proses deteksi dan pengenalan wajah secara real-time. Sistem dirancang menggunakan ESP32-CAM sebagai perangkat pengambilan citra dan server Python sebagai pusat pemrosesan utama. Interaksi antara perangkat, server, dan pengguna disajikan melalui Activity Diagram yang menggambarkan alur kerja sistem, termasuk proses pengiriman data, pemrosesan citra, serta pengiriman notifikasi melalui Telegram Bot. Alur perancangan sistem secara keseluruhan disajikan pada Gambar 2.



Gambar 2 Activity Diagram

Tahap awal dimulai dengan admin mengaktifkan server melalui perintah yang dikirimkan melalui Telegram Bot. Setelah server aktif, perangkat ESP32-CAM mulai melakukan pengambilan citra dan mengirimkannya ke

server secara berkala. Setiap citra yang diterima kemudian diproses oleh server untuk mendeteksi data wajah. Apabila wajah terdeteksi, sistem melakukan proses pencocokan dengan basis data wajah yang tersimpan pada database untuk menentukan apakah wajah tersebut termasuk dalam kategori dikenal atau tidak dikenal. Jika hasil pencocokan sesuai, data hasil deteksi disimpan sebagai output sistem. Sebaliknya, apabila wajah tidak dikenali, citra tersebut disimpan sementara pada folder sementara (temporary folder) dan sistem secara otomatis mengirimkan notifikasi peringatan kepada admin melalui Telegram. Selanjutnya, admin memiliki opsi untuk melakukan pengelolaan data melalui Telegram Bot. Admin dapat memilih perintah Create untuk menambahkan data wajah baru ke dalam database dan memulai proses pelatihan ulang (re-training) model. Selain itu, admin juga dapat menggunakan perintah Delete untuk menghapus data wajah yang tidak diperlukan. Setelah proses pelatihan model selesai, sistem kembali beroperasi dengan menggunakan model yang telah diperbarui, sehingga mampu mengenali identitas wajah yang baru ditambahkan.

2.3 Model Training

Proses pelatihan model serta pembentukan embedding wajah menggunakan FaceNet. Pada tahap ini, FaceNet mengevaluasi citra uji yang disediakan dan menghasilkan performa yang optimal. Hasil pembentukan embedding wajah selanjutnya disajikan dalam Tabel 1.

Hasil evaluasi embedding FaceNet menunjukkan bahwa seluruh identitas pada data uji berhasil dikenali dengan benar, dengan tingkat kecocokan sebesar 100% untuk setiap kelas. Hal ini mengindikasikan bahwa representasi embedding yang dihasilkan mampu memisahkan fitur wajah antar individu secara efektif pada dataset yang digunakan.

Tabel 1. Embedding Model FaceNet

Class Name	Accuracy
Class Person 01	100%
Class Person 02	100%
Class Person 03	100%
Class Person 04	100%
Class Person 05	100%
Class Person 06	100%
Class Person 07	100%
Class Person 08	100%
Class Person 09	100%
Class Person 10	100%
Mean	100%

Spesifikasi yang digunakan dalam proses pelatihan disajikan secara ringkas, mencakup spesifikasi perangkat, library yang digunakan, serta parameter pelatihan dan FaceNet, untuk memberikan Gambaran mengenai eksperimen dan mendukung keterulangan penelitian.

2.3.1 Spesifikasi Perangkat

- CPU : Intel Core i5-8365U (8 Cores @ 4.10 GHz)
- GPU : Intel UHD Graphics 620 (Integrated)
- Memori : 15.25 GB
- Sistem Operasi : Kali Linux

2.3.2 Library yang Digunakan

- Ultralytics 8.3.255
- Facenet_pytorch 2.60
- FastAPI 0.121.2
- Flask 3.1.2
- Scikit-learn 1.7.2
- Multi-task Cascaded Convolutional Neural Network (MTCNN) 1.0.0

2.3.3 Parameter Training

- Model : YOLOv8x-cls
- Batch Size : 16
- Learning Rate : 0.0001
- Epoch : 20
- Early stopping : 10
- Augmentasi :

- Rotasi 15°
- Flip up-down = 0.3
- Mixup = 0.15

2.3.4 Parameter FaceNet

- Model : Inception-ResNet-v1
- Pretrained : VGGFace2
- Model Klasifikasi : k-NN
- Nilai k : 5
- Threshold Euclidean : 0.6
- Embedding : 512 dimensi

Untuk menentukan tingkat kemiripan antar wajah, jarak antara embedding wajah dihitung menggunakan Euclidean Distance. Persamaan (1) digunakan untuk mengukur jarak antara dua vektor embedding yang dihasilkan oleh model FaceNet.

$$d(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2}$$

Di mana x dan y merupakan embedding wajah, serta n menyatakan dimensi vektor. Nilai jarak yang dihasilkan kemudian dibandingkan dengan threshold sebesar 0.6 untuk menentukan apakah wajah diklasifikasikan sebagai dikenal (*known*) atau tidak dikenal (*unknown*).

Hasil pemantauan kinerja pelatihan model YOLOv8 pada fase training ditampilkan pada Tabel 2.

Tabel 2 Hasil Training YOLO

Epoch	Time	Train/Loss	Metrics/ Accuracy top1	Metrics/ Accuracy top5	Val/Loss
1	124.21	2.11	0.57	0.88	1.66
2	249.7	1.55	0.81	0.98	0.84
3	374.46	0.92	0.94	1	0.32
4	498.49	0.49	0.98	1	0.11
5	621.03	0.36	0.99	1	0.07
6	741.64	0.26	0.99	1	0.05
7	861.93	0.35	0.99	1	0.04
8	982.7	0.24	0.99	1	0.05
9	1085.21	0.22	1	1	0.03
10	1184.49	0.21	1	1	0.03
11	1283.89	0.15	1	1	0.03
12	1382.9	0.14	0.99	1	0.04
13	1481.9	0.12	0.98	1	0.06
14	1581.07	0.13	0.98	1	0.04
15	1680.21	0.09	0.99	1	0.02
16	1779.35	0.1	1	1	0.02
17	1878.43	0.08	1	1	0.02
18	1978.3	0.08	1	1	0.01
19	2076.47	0.09	1	1	0.01
Average	1149.81	0.4	0.96	0.99	0.18

Tabel 2 menunjukkan perkembangan performa pelatihan model YOLOv8 selama 19 epoch berdasarkan metrik *Train Loss*, *Accuracy Top-1*, *Accuracy Top-5*, dan *Validation Loss*. Hasil pelatihan memperlihatkan peningkatan performa yang konsisten, ditandai dengan penurunan *train loss* yang signifikan dari 2.11 pada epoch ke-1 menjadi < 0.3 setelah epoch ke-6, serta peningkatan *Accuracy Top-1* mencapai nilai maksimum pada epoch ke-9. Rata-rata *train loss* sebesar 0.4 menunjukkan bahwa proses optimisasi model berjalan secara efektif pada dataset yang digunakan. Nilai *validation loss* mengalami tren penurunan yang sejalan dengan *train loss* dan berada pada tingkat yang rendah serta stabil pada epoch-epoch akhir, yang menandakan kemampuan generalisasi model yang baik. Stabilitas *Accuracy Top-1* dan *Accuracy Top-5* pada nilai mendekati maksimum menunjukkan bahwa

model telah mencapai konvergensi atau tidak lagi menunjukkan peningkatan yang signifikan pada epoch ke-19, sebagaimana ditunjukkan oleh stagnasi nilai *validation loss*.

Setelah proses pelatihan model selesai, hasil penerapan *embedding* untuk identitas dikenal dan tidak dikenal dianalisis lebih lanjut, ditampilkan pada Gambar 3 dan Gambar 4.



Gambar 3 Wajah Known



Gambar 4 Wajah Unknown

2.4 Pengumpulan Dataset

Pada penelitian ini dataset dari gabungan data yang dikumpulkan sendiri oleh peneliti serta data open-source dari Kaggle <https://www.kaggle.com/datasets/vasukipatel/face-recognition-dataset>. Penggabungan ini memberikan variasi yang lebih beragam, sehingga mendukung proses pelatihan model lebih optimal. Total terdapat 413 gambar yang kami gunakan, terbagi ke dalam 10 kelas orang yang berbeda.

2.4.1 Detail Dataset

- | | |
|--------------------------------|------------------------------|
| • Rasio pembagian | : 80% training, 20% validasi |
| • Teknik Split | : Random Shuffle |
| • Jumlah Kelas | : 10 Orang |
| • Rata – rata gambar per kelas | : 38 |
| • Distribusi tidak seimbang | : |
| • Kelas maksimal | : 82 gambar |
| • Kelas minimal | : 15 gambar |

Berikut adalah salah satu contoh kolase dataset yang kami gunakan:



Gambar 5 Dataset Wajah

2.5 Preprocessing Data

Pada tahap ini, data diatur, diubah, dan dipersiapkan agar sesuai dengan kebutuhan penelitian. Proses ini penting untuk meningkatkan kualitas serta konsistensi data sehingga hasil penelitian menjadi lebih akurat. Dalam penelitian ini, langkah pertama yang dilakukan adalah *cropping* area wajah pada setiap gambar. Tujuannya agar model lebih mudah fokus pada bagian penting saat pelatihan, fitur dan fungsi yang kami gunakan untuk melakukan preprocesing adalah MTCNN. MTCNN merupakan metode deteksi wajah berbasis cascaded CNN yang mampu melakukan deteksi wajah dan ekstraksi landmark secara simultan dengan akurasi tinggi dan efisiensi komputasi yang baik [11]. Pada penelitian ini, MTCNN hanya digunakan untuk preprocessing offline (*cropping* & *alignment*). Deteksi wajah real-time dilakukan sepenuhnya oleh YOLOv8. Selain itu, dataset juga diperluas melalui augmentasi data, seperti rotasi, resize, efek negatif, hingga penambahan kotak hitam untuk menutupi sebagian fitur wajah. Teknik- teknik ini membantu menambah variasi dan ketahanan model terhadap kondisi gambar yang berbeda. Setelah seluruh proses augmentasi selesai, jumlah dataset meningkat menjadi 1890 gambar.

2.5.1 Parameter preprocessing

- | | |
|----------------------------|---|
| • Face detection (offline) | : MTCNN |
| • Crop dan resize | : Bounding box terbesar → 160x160 pixel |
| • Konversi warna | : BGR → RGB |
| • Normalisasi | : Standard deviation [0.5, 0.5, 0.5] |
| • Augmentasi | : rotasi, flip, efek negatif, dan mixup |

Penerapan teknik augmentasi meningkatkan jumlah dataset menjadi 1.890 citra, dengan contoh hasil augmentasi yang disajikan pada Gambar 6.



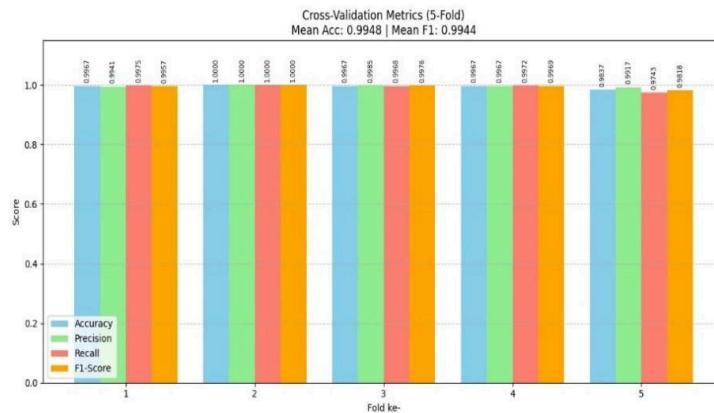
Gambar 6. Hasil Augmentasi

2.6 Pengujian Akurasi Pengenalan Wajah

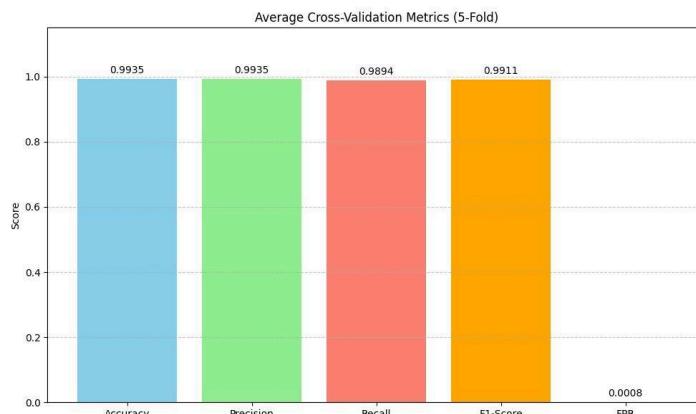
Pengujian akurasi pengenalan wajah dilakukan menggunakan metode FaceNet dengan klasifikasi k-NN, yang dievaluasi menggunakan k-fold cross-validation. Dalam penelitian ini, nilai k = 5, sehingga dataset dibagi menjadi lima bagian. Proses pelatihan dan pengujian dilakukan dalam lima iterasi. Pada setiap iterasi, satu subset digunakan sebagai data pengujian, sedangkan empat subset lainnya digunakan sebagai data pelatihan. Untuk mengevaluasi kinerja model, akurasi klasifikasi rata-rata dihitung dengan merata-ratakan hasil yang diperoleh pada seluruh iterasi.

cross-validation digunakan untuk meningkatkan kemampuan generalisasi model klasifikasi, mencegah terjadinya overfitting, serta meningkatkan keandalan performa model [12]. Hasil evaluasi metrik performa per fold dan rata-rata cross validation ditunjukkan pada Gambar 7 dan 8, yang mencakup nilai *Accuracy*, *Precision*, *Recall*, *F1-score*, dan *False Positive Rate* (FPR).

Hasil pengujian menggunakan 5-fold cross-validation menunjukkan bahwa performa model FaceNet + k-NN relatif konsisten pada seluruh fold. Setiap fold menghasilkan nilai akurasi, presisi, recall, dan F1-score yang tinggi dengan variasi yang sangat kecil antar iterasi. Hal ini mengindikasikan bahwa pembagian data yang berbeda tidak memberikan pengaruh signifikan terhadap kinerja model. Konsistensi hasil pada seluruh fold menegaskan bahwa model memiliki kemampuan generalisasi yang baik serta stabil dalam mengenali wajah pada data yang belum pernah dilihat sebelumnya.



Gambar 7. Metrik per fold dari 5 fold cross-validation



Gambar 8. Metrik rata-rata cross-validation

Berdasarkan hasil pengujian, model FaceNet + k-NN menunjukkan performa yang sangat tinggi dengan nilai akurasi rata-rata sebesar 0.9935, yang mengindikasikan bahwa sebagian besar wajah pada data uji berhasil dikenali dengan benar. Nilai *precision* dan *recall* yang masing-masing mencapai 0.9935 dan 0.9894 menunjukkan bahwa sistem tidak hanya mampu mengenali identitas wajah secara akurat, tetapi juga memiliki tingkat kesalahan yang sangat rendah dalam mengklasifikasikan wajah yang dikenal dan tidak dikenal. Hal ini diperkuat oleh nilai *F1-score* sebesar 0.9911, dan nilai FPR yang sangat kecil, yaitu 0.0008, menunjukkan bahwa sistem jarang mengidentifikasi wajah yang tidak dikenal sebagai wajah yang terdaftar. Hasil ini menjelaskan bahwa kombinasi FaceNet dan k-NN menghasilkan *embedding* wajah yang diskriminatif dan konsisten pada dataset yang digunakan.

3. HASIL DAN ANALISIS

3.1 Implementasi Sistem

Sistem deteksi dan pengenalan wajah berhasil diimplementasikan dengan memanfaatkan ESP32-CAM, sementara proses komputasi utama dilakukan pada server Python. Model YOLOv8 digunakan untuk mendeteksi wajah secara real-time, sedangkan FaceNet berperan dalam mengekstraksi *embedding* wajah berdimensi 512

sebelum proses klasifikasi dilakukan menggunakan algoritma k-NN ($k = 5$). Wajah yang terdeteksi sebagai tidak dikenal (*unknown*) secara otomatis sistem melakukan pengiriman notifikasi peringatan melalui Telegram Bot.

3.1.1 Implementasi Hasil Pengenalan Wajah FaceNet Known/Unknown

Pengujian pengenalan wajah dilakukan pada 10 kelas individu menggunakan metode 5-fold cross-validation untuk mengevaluasi performa dan kemampuan generalisasi sistem. Model FaceNet digunakan untuk mengekstraksi embedding wajah berdimensi 512, yang selanjutnya dibandingkan menggunakan Euclidean Distance. Penentuan identitas dilakukan berdasarkan nilai ambang (*threshold*), di mana wajah diklasifikasikan sebagai known jika jarak embedding lebih kecil dari 0.6, dan sebagai unknown jika jarak sama dengan atau lebih besar dari 0.6.

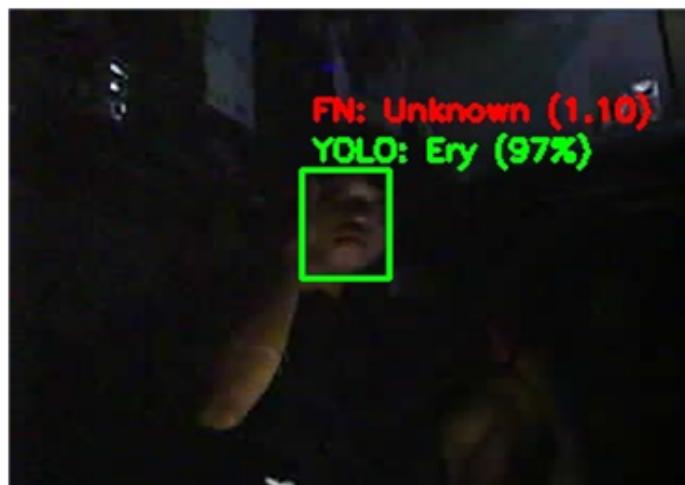
Hasil pengujian menunjukkan bahwa seluruh data uji berhasil diklasifikasikan dengan sangat baik, dengan akurasi rata-rata sebesar 0.9935 dan FPR sebesar 0.0008. Nilai ini mengindikasikan bahwa sistem mampu membedakan wajah yang dikenal dan tidak dikenal secara akurat serta konsisten pada seluruh fold pengujian. Tingginya performa tersebut dipengaruhi oleh beberapa faktor, antara lain jumlah kelas yang relatif kecil dan kondisi dataset yang terkontrol, sehingga variasi antar wajah masih terbatas dan *embedding* antar kelas cenderung memiliki jarak yang jelas.

Proses akuisisi citra dilakukan dalam lingkungan yang relatif stabil, dengan resolusi kamera ESP32-CAM yang seragam dan kondisi pencahayaan yang konsisten. Jumlah citra per kelas yang terbatas juga berkontribusi terhadap rendahnya kemungkinan terjadinya tumpang tindih (*overlapping*) antar *embedding*.

Apabila jumlah subjek diperluas hingga ratusan atau ribuan individu, variasi fitur wajah akan meningkat dan jarak antar *embedding* berpotensi menjadi lebih berdekatan. Kondisi tersebut dapat menyebabkan penurunan akurasi pengenalan, sehingga diperlukan penyesuaian nilai *threshold*, peningkatan jumlah data per kelas, serta strategi optimasi tambahan untuk menjaga performa sistem. Visualisasi hasil pengenalan wajah untuk kategori *known* dan *unknown* disajikan pada Gambar 9 dan 10.



Gambar 9 Hasil pengenalan wajah known (FaceNet)



Gambar 10 Hasil pengenalan wajah unknown (FaceNet)

3.2 Implementasi Hasil Pengujian Sistem

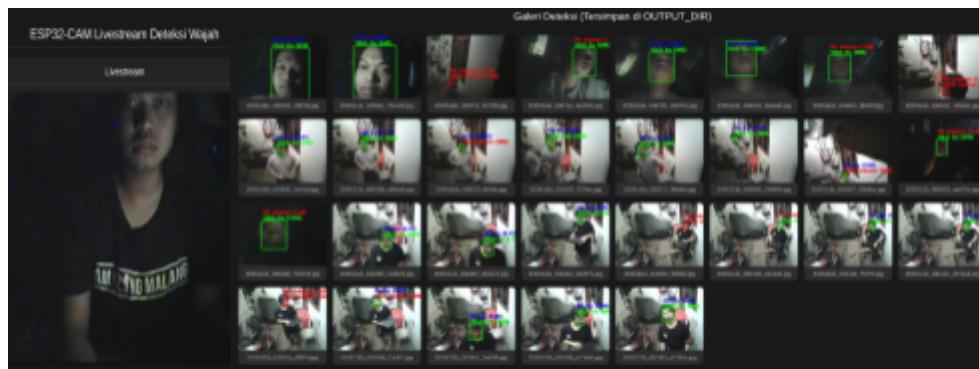
Implementasi hasil pengujian sistem pengenalan wajah yang telah dikembangkan. Pengujian dilakukan untuk mengevaluasi kinerja sistem dalam mengenali wajah berdasarkan variasi kondisi pencahayaan dan jarak. Pada pengujian ini digunakan 10 kelas individu, dan hasil pengujian tersebut ditunjukkan pada Gambar 11.

Waktu Pengujian	Jarak	Foto	Keterangan
Siang hari	1 meter		Pada jarak ini, pengenalan wajah berfungsi dengan baik dan berhasil menentukan wajah dengan benar
	2 meter		Pada jarak ini, pengenalan wajah berfungsi dengan baik dan berhasil menentukan wajah dengan benar
	3 meter		Pada jarak ini, pendekripsi wajah berfungsi dengan baik, tetapi model FaceNet gagal menentukan wajah dengan benar
Malam hari	1 meter		Pada jarak ini di malam hari, pendekripsi wajah tidak berhasil
	2 meter		Pada malam hari, pendekripsi wajah tidak berhasil
	3 meter		Pada malam hari di jarak ini, dan beberapa pengambilan yang terjadi salah satunya berhasil mendekripsi keberadaan wajah namun gagal menentukan wajah

Gambar 11 Implementasi Hasil Pengujian Sistem

3.3 Tampilan Web Monitoring

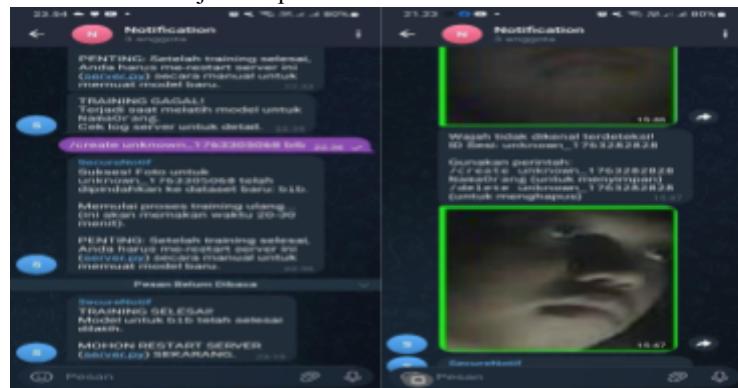
Web monitoring dirancang untuk menampilkan hasil deteksi wajah secara langsung dan hasil tangkapan citra yang telah diproses oleh sistem. Pengguna dapat melihat hasil streaming dari ESP32-CAM serta galeri hasil deteksi wajah yang tersimpan secara otomatis oleh sistem. Tampilan ini memudahkan pengguna dalam melakukan pemantauan dan evaluasi kinerja sistem deteksi wajah. Gambar 12 menyajikan tampilan web monitoring.



Gambar 12 Tampilan Web Monitoring

3.4 Tampilan Notifikasi Telegram

Sistem secara otomatis mengirimkan notifikasi *Unknown Face* melalui Telegram ketika terdeteksi wajah yang tidak terdaftar dalam basis data. Notifikasi yang dikirimkan mencakup foto wajah yang terdeteksi, informasi waktu (*timestamp*), serta status *Unknown*. Proses pengiriman notifikasi dilakukan dalam waktu kurang dari satu detik, sehingga sistem mampu memberikan respons yang cepat untuk mendukung keamanan rumah. Tampilan notifikasi Telegram untuk wajah tidak dikenal ditunjukkan pada Gambar 13.



Gambar 13 Notifikasi Telegram

3.5 Evaluasi Kinerja Sistem

Hasil pengujian sistem dirangkum dalam Tabel 3. Parameter yang diuji mencakup jumlah dataset, akurasi model, dan waktu proses inferensi.

Tabel 3 Evaluasi Kinerja Sistem

No	Parameter	Hasil
1.	Jumlah Dataset Awal	413 Gambar
2.	Dataset Setelah Augmentasi	1890 Gambar
3.	Jumlah Kelas	10
4.	Akurasi FaceNet	99,35 %
5.	Akurasi YOLOv8	99,6%
6.	Latency Pengiriman ESP32 Ke Server	± 1 Detik
7.	Waktu Inferensi YOLOv8	± 0.5 Detik
8.	Waktu Inferensi FaceNet	± 0.3 Detik
9.	Total Waktu Deteksi	1.86 Detik

Evaluasi kinerja sistem dilakukan dengan menggunakan dataset awal sebanyak 413 gambar, yang kemudian ditingkatkan melalui augmentasi menjadi 1890 gambar, terbagi ke dalam 10 kelas. Hasil pengujian menunjukkan bahwa FaceNet mencapai akurasi 99,35 %, sedangkan YOLOv8 memiliki akurasi 99,6%. Latensi pengiriman data dari ESP32 ke server tercatat sekitar \pm 1 detik, menjadi faktor penentu utama dalam total waktu deteksi sistem yang mencapai 1,86 detik. Waktu inferensi untuk YOLOv8 tercatat \pm 0,5 detik, sedangkan FaceNet memerlukan \pm 0,3 detik untuk menghitung embedding 512 dimensi dan melakukan klasifikasi menggunakan k-NN dengan $k = 5$, yang memberikan beban komputasi yang relatif ringan.

Analisis latensi mengidentifikasi bahwa jaringan merupakan bottleneck terbesar, dipengaruhi oleh kualitas sinyal Wi-Fi, jarak perangkat, dan keterbatasan bandwidth ESP32. Meski YOLOv8 berjalan menggunakan GPU terintegrasi (UHD 620), waktu inferensi tetap memadai untuk deteksi real-time sederhana. FaceNet mampu menghitung *embedding* dengan cepat, sehingga keterlambatan utama berasal dari pengiriman data, bukan dari model AI itu sendiri. Optimasi sistem sebaiknya difokuskan pada peningkatan kualitas jaringan, kompresi gambar sebelum pengiriman, dan pengaturan interval capture ESP32.

Dalam identifikasi wajah *unknown*, FaceNet menggunakan jarak Euclidean untuk menentukan apakah wajah termasuk *known* atau *unknown*. Threshold memiliki pengaruh signifikan terhadap akurasi sistem. Threshold sebesar 0,6 dipilih setelah percobaan, karena memberikan keseimbangan terbaik antara *false rejection*, yang terjadi jika threshold terlalu kecil ($< 0,5$), dan *false acceptance*, yang terjadi jika threshold terlalu besar ($> 0,9$).

Beberapa kendala sistem juga diidentifikasi, antara lain latensi jaringan yang cukup tinggi, penurunan performa deteksi dan *embedding* pada kondisi pencahayaan gelap, kesulitan deteksi wajah jika jarak melebihi 3 meter, serta ketidakseimbangan dataset yang memengaruhi performa training awal. Sistem yang diusulkan menunjukkan kinerja yang memadai dan tangguh pada lingkungan terkontrol dan cocok untuk aplikasi keamanan rumah berskala kecil. Akurasi FaceNet yang hampir sempurna ini merupakan indikasi keterbatasan dataset dan kondisi pengujian yang ideal. Latensi sistem lebih banyak dipengaruhi oleh keterbatasan jaringan dibanding model AI, sehingga upaya optimasi sebaiknya difokuskan pada peningkatan kualitas jaringan.

4. KESIMPULAN

Hasil penelitian menunjukkan bahwa sistem keamanan berbasis YOLOv8 dan FaceNet mampu mendeteksi serta membedakan wajah *known* dan *unknown* secara real-time pada lingkungan terkontrol dengan jumlah kelas terbatas (10 orang). Hasil tersebut mengindikasikan bahwa sistem dapat bekerja secara konsisten dalam kondisi pengujian yang terstruktur, meskipun performanya dapat berubah ketika jumlah identitas diperbesar.

Algoritma YOLOv8 menunjukkan kemampuan deteksi wajah yang akurat dan cepat dengan tingkat akurasi 99,6% dan waktu respon 1,86 detik, namun waktu respon keseluruhan sangat dipengaruhi oleh latensi jaringan ESP32-CAM. FaceNet memastikan proses ekstrasi fitur dan verifikasi identitas berjalan optimal, performa yang dihasilkan mencapai akurasi rata-rata sebesar 99,35% dan FPR sebesar 0,08%. ESP32-CAM juga berfungsi efektif sebagai kamera IoT berbiaya rendah untuk akuisisi video, meskipun proses inferensi tetap dilakukan di server untuk menjaga performa. Sistem ini mampu mengirimkan notifikasi melalui Telegram Bot saat mendeteksi wajah yang tidak dikenal, lengkap dengan *capture*, *bounding box*, waktu kejadian, dan label identifikasi. Fitur tersebut meningkatkan respons keamanan karena peringatan diterima dalam hitungan detik. Sistem yang dikembangkan terbukti stabil pada siang hari dan malam hari dan berhasil menghadirkan solusi keamanan berbasis IoT dan *deep learning* yang lebih responsif dan efisien dibandingkan pendekatan konvensional.

Pada penelitian selanjutnya, dapat dicoba pada peningkatan skalabilitas dengan memperluas jumlah kelas wajah untuk menguji *robustness* sistem pada kondisi dunia nyata dan . Optimasi model melalui teknik *quantization* atau model *compression* mengindikasi proses inferensi dijalankan langsung pada perangkat *edge* yang lebih kuat, seperti Raspberry Pi, sehingga latensi jaringan dapat dikurangi. Disarankan untuk menambahkan metode *liveness detection* untuk mencegah pemalsuan identitas menggunakan foto atau layar digital.

5. UCAPAN TERIMAKASIH

Penulis mengucapkan terimakasih yang sebesar-besarnya kepada seluruh pihak yang telah berkontribusi dalam pelaksanaan penelitian ini. Penulis juga memberikan apresiasi yang tinggi kepada dosen pembimbing atas bimbingan, arahan, serta masukan yang sangat membantu dalam penyempurnaan penelitian ini. Penulis juga berterima kasih kepada seluruh rekan-rekan yang turut berkontribusi dalam proses pengumpulan data maupun pelaksanaan pengujian. Segala bentuk bantuan dan dukungan yang diberikan memiliki peran penting dalam terselesainya penelitian ini dengan baik.

REFERENSI

- [1] H. Syafutra, T. Muhammad Nur Aziz, I. Novianty, M. Chusnu, and D. Prayoga, "Implementasi Sistem Keamanan Pintu Otomatis Berbasis Face Recognition di Proactive Robotic: Integrasi ESP32-Cam dan Telegram," *Jurnal Riset Fisika Indonesia*, vol. 4, no. 2, 2024, [Online]. Available: <https://journal.ubb.ac.id/jrfi/article/view/5380Halaman65>
- [2] G. Firmasyah, A. Mulyo Widodo, B. Tjahjono, and U. Esa Unggul, "PREVENTING CHILD KIDNAPING AT HOME USING CCTV THAT UTILIZES FACE RECOGNITION WITH YOU ONLY LOOK ONCE (YOLO) ALGORITHM", [Online]. Available: <http://ijsr.internationaljournallabs.com/index.php/ijsr>
- [3] D. Anugrah Pratama and M. Bahrul Ulum, "RANCANG BANGUN SISTEM KEAMANAN RUMAH DENGAN FACE RECOGNITION BERBASIS ESP32-CAM 1,*," 2024.
- [4] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," Jun. 2015, doi: 10.1109/CVPR.2015.7298682.
- [5] W. A. Shobaki and M. Milanova, "A Comparative Study of YOLO, SSD, Faster R-CNN, and More for Optimized Eye-Gaze Writing," *Sci*, vol. 7, no. 2, Jun. 2025, doi: 10.3390/sci7020047.
- [6] A. Setiawan, K. T. Prasetyo, A. Rusdyansyah, and D. Ardian, "Implementation of facial recognition technology in the verification system for api banyuwangi cadets using the haar cascade algorithm," *Bulletin of Social Informatics Theory and Application*, vol. 8, no. 2, pp. 309–330, Dec. 2024, doi: 10.31763/businta.v8i2.778.
- [7] D. P. Sari, "Comparative Study: Performance Comparison of You Only Look Once and Convolutional Neural Networks Algorithms in Human Object Detection," *Indonesian Journal of Artificial Intelligence and Data Mining (IJAIDM)*, vol. 8, no. 2, pp. 123–135, 2025, [Online]. Available: <https://ejournal.uin-suska.ac.id/index.php/IJAIDM/article/view/37676>
- [8] A. Nemavholia, C. Chibaya, and S. Viriri, "A Systematic Review of CNN Architectures, Databases, Performance Metrics, and Applications in Face Recognition," Feb. 01, 2025, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/info16020107.
- [9] M. R. Dewanto, M. N. Farid, M. A. Rafdi Syah, A. A. Firdaus, and H. Arof, "YOLO vs. CNN Algorithms: A Comparative Study in Masked Face Recognition," *Scientific Journal of Informatics*, vol. 11, no. 1, pp. 139–146, Feb. 2024, doi: 10.15294/sji.v11i1.48723.
- [10] G. Fairuz Mumtaz, J. Zeniarja, A. Luthfiarta, and A. N. Imam Muttaqin, "Optimizing Face Recognition and Emotion Detection in Student Identification Using FaceNet and YOLOv8 Models," *Inform : Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 10, no. 1, pp. 34–44, Jan. 2025, doi: 10.25139/inform.v10i1.9304.
- [11] H. Lin, Y. Hu, S. Chen, J. Yao, and L. Zhang, "Fine-Grained Classification of Cervical Cells Using Morphological and Appearance Based Convolutional Neural Networks," Oct. 2018, [Online]. Available: <http://arxiv.org/abs/1810.06058>
- [12] M. S. Uzer, "Deep Learning-Based Classification Consisting of Pre-Trained Models and Proposed Model Using K-Fold Cross-Validation for Pistachio Species," *Applied Sciences (Switzerland)*, vol. 15, no. 8, Apr. 2025, doi: 10.3390/app15084516.
- [13] C. Baretina, S. Saripudin, R. Listiana, and E. Damayanti, "Rancang Bangun Sistem Smart Door Lock Menggunakan Deteksi Wajah."
- [14] C. I. Zamorano, K. Prawiroedjo, E. S. Julian, and E. Djuana, "Rancang Bangun Sistem Kamera Pengawas dengan Pengenalan Wajah untuk Keamanan Berbasis Blynk Legacy."
- [15] A. M. S. M. Koroy, G. Mandar, and A. H. Muhammad, "RANCANG BANGUN SISTEM KEAMANAN PINTU RUMAH MENGGUNAKAN ESP32-CAM," *Jurnal Teknik Informatika (J-Tifa)*, vol. 3, no. 2, pp. 32–36, Sep. 2020, doi: 10.52046/j-tifa.v3i2.1038.
- [16] A. Peryanto, A. Yudhana, and D. R. Umar, "Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network," 2019. [Online]. Available: <https://www.mathworks.com/discovery/convolutional-neural-network.html>
- [17] A. M. S. M. Koroy, G. Mandar, and A. H. Muhammad, "RANCANG BANGUN SISTEM KEAMANAN PINTU RUMAH MENGGUNAKAN ESP32-CAM," *Jurnal Teknik Informatika (J-Tifa)*, vol. 3, no. 2, pp. 32–36, Sep. 2020, doi: 10.52046/j-tifa.v3i2.1038.
- [18] M. Akbar Syahbana Pane *et al.*, "Journal of Information System and Technology Research Low-Cost CCTV for Home Security With Face Detection Base on IoT," 2024. [Online]. Available: <http://creativecommons.org/licenses/by-sa/4.0/>
- [19] C. Ferry, M. Saputra, and W. Sulistyo, "Alat Keamanan Depan Rumah Berbasis Internet of Things (IoT) Menggunakan ESP32-CAM yang Terintegrasi dengan Face Detection dan Telegram," *Jurnal Teknologi Informatika dan Komunikasi*, vol. 8, no. 1, p. 2024, 2024, doi: 10.35870/jti.
- [20] A. P. Y. Waroh, N. Sajangbati, S. Sawidin, M. A. S. Kondoj, and T. J. Wungkana, "Sistem Keamanan Rumah Melalui Pengenalan Wajah Dengan Webcam Berbasis Raspberry Pi4," *Jambura Journal of Electrical and Electronics Engineering*, vol. 7.