

Sistem *Smart Door Lock* Terintegrasi IoT dan *Face Recognition* Berbasis *Edge Computing*

Ardi Syawaldipa¹, Ideva Gaputra², Andre Febrian Kasmar³, Dian Eka Putra⁴

^{1,2,3,4}Teknologi Informasi, Politeknik Negeri Padang, Indonesia

ardisyawaldipa@pnp.ac.id, ideva@pnp.ac.id, andrefebrian@pnp.ac.id, dianekaputra@pnp.ac.id

Abstract

The current IoT-based facial recognition door security system often experiences high latency and is prone to failure when the internet connection is unstable due to centralized data processing in the cloud. Thus, it required local processing (*edge computing*) to authenticated and remote the control responses become faster and *real-time*. The using of physical door keys has various weaknesses such as being lost, damaged or duplicated easily. Therefore, this study discusses the design and implementation of *edge computing* door security systems with *Internet of Things*-based facial recognition with remote control. This study aims to design a smart door lock system based on *face recognition* and *edge computing* to improve home security through access restrictions. This system is also integrated with *Internet of Things (IoT)* technology in door locking so it can be accessed and controlled in *real-time* via an Android device. This system uses ESP32-CAM as a facial sampling tool and NodeMCU ESP32 as an actuator controller and a solenoid door lock as the door opening and closing mechanism. The facial verification process is carried out locally on the edge server by using Python and OpenCV without the cloud thereby increasing response speed and data security. This system is also equipped with an Android application as a remote control. Tests have shown that the system is capable of good facial recognition, automatic door control and remote door access. The results of this research have an impact on improving home security, privacy protection through local data processing and the convenience of remote access control. It also contributes to smart home innovation that successfully integrates facial recognition, *edge computing* and IoT. All of these impacts and contributions are realized in the form of a smart door lock hardware prototype. This system is expected to be a more effective, efficient and modern home door security solution.

Keywords: *Edge Computing, Face Recognition, ESP32-CAM, Automation, IOT.*

Abstrak

Sistem keamanan pintu pengenalan wajah berbasis IoT saat ini sering mengalami latensi tinggi dan rentan gagal saat koneksi internet tidak stabil akibat pemrosesan data yang terpusat di *cloud* sehingga memerlukan pemrosesan lokal (*edge computing*) agar respons autentikasi dan kendali jarak jauh menjadi lebih cepat dan *real-time*. Penggunaan kunci pintu fisik memiliki berbagai kelemahan seperti mudah hilang, rusak atau diduplikasi. Oleh karena itu, pada penelitian ini membahas rancangan dan implementasi *edge computing* sistem keamanan pintu dengan *face recognition* berbasis *Internet of Things* dengan kendali jarak jauh. Penelitian ini bertujuan untuk merancang sistem smart door lock berbasis *face recognition* dan *edge computing* untuk meningkatkan keamanan rumah melalui pembatasan akses. Pada sistem ini juga diintegrasikan dengan teknologi *Internet of Things (IoT)* sehingga penguncian dan akses pintu dapat dikendalikan secara *real-time* melalui perangkat Android. Sistem ini menggunakan ESP32-CAM sebagai alat pengambil sampel wajah, NodeMCU ESP32 sebagai alat pengendali aktuator serta *solenoid door lock* sebagai mekanisme buka dan tutup pintu. Proses verifikasi wajah dilakukan secara lokal pada *edge server* menggunakan Python dan OpenCV tanpa menggunakan *cloud*, sehingga meningkatkan kecepatan respon dan keamanan data. Sistem ini juga dilengkapi dengan aplikasi Android sebagai kendali jarak jauh. Pada pengujian yang telah dilakukan menunjukkan sistem mampu mengenali wajah yang

dengan baik, mengontrol pintu secara otomatis serta mengakses pintu dari jarak jauh. Hasil dari penelitian ini memberikan dampak terhadap peningkatan keamanan rumah, perlindungan privasi melalui pemrosesan data lokal, dan kenyamanan kontrol akses jarak jauh, sekaligus berkontribusi sebagai inovasi *smart home* yang berhasil mengintegrasikan *face recognition*, *edge computing*, dan IoT. Keseluruhan dampak dan kontribusi tersebut direalisasikan dalam bentuk prototipe perangkat keras *smart door lock* pada pintu. Sistem ini diharapkan dapat menjadi solusi keamanan pintu rumah yang lebih efektif, efisien dan modern.

Kata kunci: Edge Computing, Pengenalan wajah, ESP32-CAM, Automasi, IOT.

© 2026 Author

Creative Commons Attribution 4.0 International License



1. Pendahuluan

Sistem *smart door lock* adalah sebuah sistem dengan melakukan pengamanan terhadap pintu secara otomatis. Pengoperasiannya dapat menggunakan berbagai macam cara diantaranya, menggunakan QR Code, *Face Recognition*, *Password*. Fungsi dari *smart door lock* ini memberi Batasan akses hanya kepada orang-orang yang memiliki izin akses [1]. *Smart door lock* ini memanfaatkan teknologi *face recognition* sebagai metode pengoperasiannya dalam membuka pintu, dan juga bisa menggunakan aplikasi sebagai kontrol dari jarak jauh apabila kondisi rumah atau perusahaan sedang tidak ada orang.

Keamanan fisik suatu bangunan, baik rumah tinggal maupun bangunan lainnya, merupakan kebutuhan dasar yang terus berkembang. Secara tradisional, sistem keamanan pintu bergantung pada kunci mekanis fisik. Namun, penggunaan kunci fisik memiliki kelemahan yang mendasar seperti risiko kehilangan kunci yang rusak atau mudah diduplikasi. Seiring dengan pesatnya perkembangan *Internet of Things* (IoT) dan konsep *smart home*, paradigma keamanan bergeser menuju sistem *smart door lock* [2,3]. Sistem ini menawarkan otomatisasi penguncian dan pembatasan akses yang lebih ketat, memastikan bahwa pintu hanya dapat dibuka oleh individu yang telah memiliki otorisasi.

Berbagai penelitian yang telah dilakukan sebelumnya mengeksplorasi berbagai metode penggunaan *smart door lock*, mulai dari penggunaan PIN/Password, kartu RFID, hingga pemindaian *QR Code*. Akan tetapi metode tersebut masih memiliki celah keamanan, seperti kata sandi yang dapat ditebak, atau kartu/kode yang bisa disebarkan dan dicuri. Maka, otentikasi biometrik, khususnya pengenalan wajah, merupakan metode yang saat ini dianggap jauh lebih aman karena menggunakan karakteristik fisik manusia yang unik dan tidak dapat dipindahtanggankan untuk mengidentifikasi seseorang [4].

Perkembangan sistem keamanan rumah pintar saat ini menuntut integrasi teknologi yang tidak hanya akurat, tetapi juga responsif dan menjamin privasi data pengguna. Meskipun beberapa penelitian terdahulu telah banyak mengeksplorasi *smart door lock* dengan teknologi *face recognition* berbasis IoT maupun sistem autentikasi biometrik yang terpusat pada *cloud computing*, pendekatan tersebut seringkali masih dihadapkan pada kendala waktu tunda (*latency*) jaringan serta risiko kerentanan keamanan data wajah. Berpijak pada batasan (*state of the art*) tersebut, kebaruan (*novelty*) dari penelitian ini ditunjukkan melalui penerapan arsitektur *edge computing* yang memungkinkan pemrosesan data pengenalan wajah dieksekusi secara lokal pada perangkat untuk memangkas latensi dan memastikan data privasi tidak keluar ke jaringan publik.

Teknologi pengenalan wajah menawarkan tingkat keamanan biometrik yang tinggi, namun jika implementasikan pada sistem IoT konvensional saat ini seringkali menghadapi kendala dalam hal arsitektur. Mayoritas sistem yang ada sangat bergantung pada *cloud computing* untuk memproses dan mencocokkan data gambar wajah.[5,6] Ketergantungan pada arsitektur terpusat ini memunculkan masalah signifikan: sistem rentan mengalami *latency* yang tinggi dan berpotensi gagal total jika koneksi internet tidak stabil. Selain itu, pengiriman data biometrik secara konstan ke server eksternal juga memunculkan kekhawatiran terkait privasi dan keamanan data pengguna. Pendekatan ini secara signifikan mempercepat waktu respons otentikasi menjadi *real-time*, menjaga privasi data biometrik pengguna agar tetap tersimpan di jaringan lokal, namun tetap menawarkan fleksibilitas kontrol jarak jauh berbasis IoT [7]. Hasilnya adalah sebuah sistem keamanan cerdas yang lebih andal, responsif, modern, dan tangguh terhadap gangguan konektivitas.

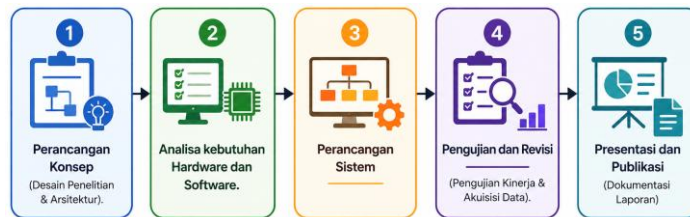
Untuk mengatasi permasalahan tersebut, pada penelitian ini akan dilakukan rancangan dan implementasi *smart*

Submitted : 01-06-2026 | Reviewed : 05-06-2026 | Accepted : 30-06-2026

door lock berbasis Face Recognition yang tidak lagi bergantung pada cloud computing, melainkan diintegrasikan dengan aplikasi pengontrol jarak jauh. Sistem ini bekerja ganda: pertama, sebagai sistem otonom yang memverifikasi wajah pengunjung di lokasi secara langsung untuk membuka pintu otomatis; kedua, sebagai sistem yang dapat dikendalikan dari jarak jauh menggunakan aplikasi seluler. Penggunaan aplikasi ini berfungsi sebagai fitur krusial untuk memberikan akses darurat dari mana saja, misalnya ketika pemilik sedang tidak berada di rumah atau perusahaan, namun ada tamu atau teknisi yang memerlukan akses masuk.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan *Research and Development (R&D)* dengan metode *Prototyping* untuk merancang dan mengimplementasikan sistem keamanan pintu berbasis edge computing[8, 9]. Penggunaan *edge computing* pada arsitektur IoT bertujuan untuk mereduksi latensi jaringan dan meningkatkan keamanan privasi data dengan memproses komputasi biometrik secara lokal. Alur penelitian dilaksanakan melalui lima tahapan kronologis sebagai berikut:

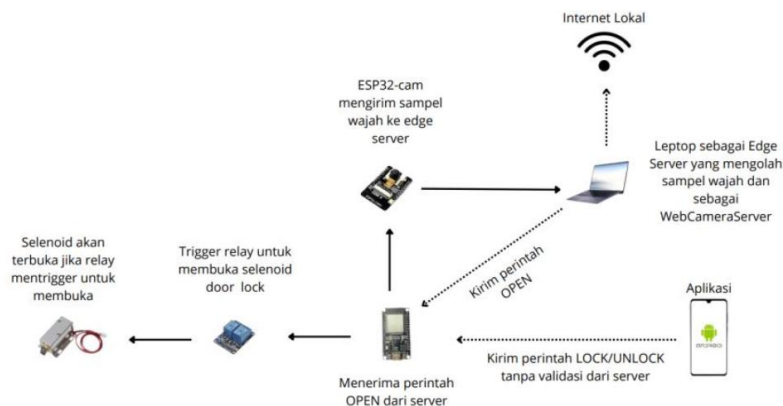


Gambar 1. Metode Penelitian

Gambar diagram alir tersebut mengilustrasikan tahapan metodologi penelitian pengembangan sistem *smart door lock* secara berurutan yang diawali dengan tahap Perancangan Konsep untuk menentukan desain penelitian dan arsitektur dasar. kemudian, alur dilanjutkan pada tahap Analisa Kebutuhan Hardware dan Software guna mengidentifikasi spesifikasi teknis komponen yang diperlukan, yang kemudian direalisasikan secara teknis dalam tahap Perancangan Sistem. Sistem yang telah dirangkai selanjutnya memasuki tahap Pengujian dan Revisi untuk mengevaluasi kinerja serta mengakuisisi data; pada tahap ini terdapat alur kondisional di mana apabila sistem belum optimal maka akan dikembalikan ke tahap perancangan untuk direvisi, namun jika pengujian dinyatakan berhasil, alur akan diteruskan ke tahap penyelesaian akhir yaitu Presentasi dan Publikasi yang berfokus pada penyusunan dokumentasi laporan.

2.1. Perancangan Konsep (Desain Penelitian & Arsitektur).

Tahap ini difokuskan pada pemodelan arsitektur sistem. Desain penelitian menetapkan bahwa pemrosesan citra wajah (*face recognition*) tidak dikirim ke *cloud server*, melainkan diproses secara lokal (*edge processing*).



Gambar 2. Perancangan Sistem *Smart Door Lock*

Pada gambar rancangan infrastruktur sistem diatas, terdapat beberapa komponen utama, yaitu perangkat IoT, Laptop (*edge server*), Android. Pada perangkat keras terdapat komponen IoT yang terdiri dari ESP32-CAM, NodeMCU ESP32, Relay 5v 2 Channel, *Solenoid Door Lock*, Adaptor 12v. ESP32-CAM dan laptop akan

terhubung melalui kabel USB. ESP32-CAM ini akan menampilkan *Web Camera Server* untuk menampilkan *streaming* sebagai pengambilan sampel wajah dengan menekan *enroll face*, kemudian sampel wajah akan di kirim ke *edge server* untuk mengolah dan menyimpan sampel tersebut. Selanjutnya NodeMCU ESP32 akan menerima perintah OPEN melalui metode HTTP dari *edge server* yang telah mengolah dan mencocokkan wajah yang telah ada, kemudian NodeMCU ESP32 akan *trigger relay* untuk membuka *solenoid door lock*. Aplikasi disini terhubung dengan NodeMCU ESP32 melalui metode HTTP yang berfungsi untuk mengakses *solenoid door lock* membuka atau menutup tanpa harus ada wajah yang dideteksi maupun yang telah disimpan dalam *server*[10,11].

2.2. Analisa kebutuhan *Hardware* dan *Software*.

Agar sistem dapat terealisasi, diperlukan sejumlah perangkat keras dan perangkat lunak yang saling terintegrasi. Perangkat keras (*Hardware*) bertugas dalam proses pengambilan data serta membuka *solenoid door lock*, sedangkan perangkat lunak (*Software*) bertugas sebagai pemrosesan data dan pengiriman data ke *Edge Server* dan NodeMCU ESP32. Adapun rincian kebutuhan perangkat keras dan perangkat lunak sistem ini sebagai berikut:

Perancangan perangkat keras meliputi elektronik seperti rangkaian mikrokontroler. Berikut adalah bahan yang digunakan dalam perancangan perangkat keras.

Tabel 1. Analisis Kebutuhan Hardware

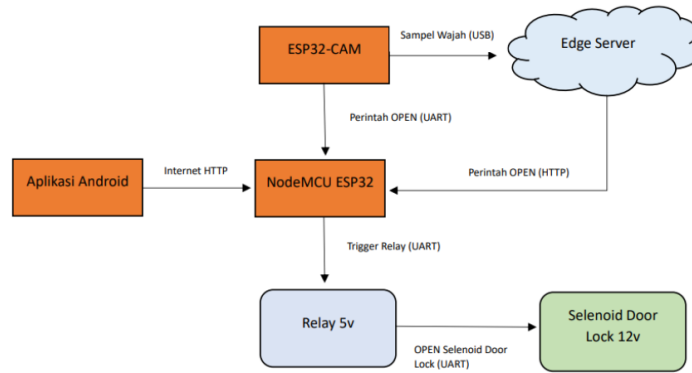
Komponen	Jumlah	Fungsi
Esp32-Camera	1	ESP32-CAM berfungsi untuk mengambil gambar dan menampilkan live 23 stream dari kamera bawaan yang akan dikirim ke <i>Edge Server</i>
NodeMCU ESP32	1	Berfungsi sebagai mengendalikan relay untuk membuka solenoid jika ada perintah OPEN dari <i>edge Server</i> , serta menjadi control IoT melalui android
Breadboard dan Kabel Jumper	1	Digunakan untuk Menyusun dan menghubungkan rangkaian elektronik
Solenoid Door Lock	1	Berfungsi sebagai alat pengunci pintu otomatis
Relay 2 Channel 5v	1	Relay berfungsi saklar <i>lock/unlock</i> tapi dikendalikan oleh sinyal dari NodeMCU ESP32

Tabel 2. Analisis Kebutuhan Software

Komponen	Jumlah	Fungsi
Arduino IDE		Digunakan untuk menulis, mengunggah dan mengatur program NodeMCU ESP32
Android Studio		Untuk membuat aplikasi pengaturan proses lock/unlock
Visual Studio Code		Untuk membuat <i>Edge Server</i> yang mengolah semua sampel wajah dari ESP32-CAM
Python OpenCV		Untuk menjalankan <i>Edge Server</i> dan sebagai mengolah serta menyimpan data
Flask		Sebagai routing dari ESP32-CAM ke <i>Edge Server</i>

2.3. Perancangan Sistem

Perancangan perangkat keras dalam sistem *smart door lock* berbasis *face recognition*, *Internet of Things* dan *edge computing* dilakukan melalui integrasi berbagai komponen perangkat keras dan lunak. Komponen utama yang digunakan yaitu ESP32-CAM, yang berfungsi untuk mengambil sampel wajah sekaligus melakukan proses pengenalan wajah secara lokal tanpa bergantung kepada *server* eksternal [12]. Proses ini memungkinkan sistem bekerja secara cepat dan efisien melalui penerapan konsep *edge computing*. Hasil perekaman data wajah saat proses enroll disimpan secara lokal pada media penyimpanan microSD. Untuk kendali aktuator, digunakan modul relay 5v yang terhubung ke *solenoid door lock* sebagai mekanisme buka-tutup pintu[13,14]. Selain itu, ESP32-CAM dihubungkan ke jaringan WIFI, sehingga memungkinkan kontrol pintu jarak jauh melalui aplikasi Android. Dengan rancangan ini sistem mampu menyediakan solusi keamanan pintu yang efisien, mandiri, serta mudah diakses [15].



Gambar 3. Rancangan Topologi Diagram Jaringan Sistem

Topologi jaringan digunakan untuk menggambarkan hubungan antara perangkat yang terlibat dalam sistem, media komunikasi, serta *edge computing* yang digunakan. Pada diagram dibawah, menunjukkan rancangan dimana ESP32-CAM akan mengirim sampel wajah yang terhubung ke laptop dengan USB [16]. NodeMCU ESP32 yang

terhubung dengan ESP32-CAM melalui pin GPIO untuk menerima daya dan juga menerima perintah OPEN dengan protocol UART. Laptop sebagai *edge server* yang mengolah sampel wajah seperti verifikasi wajah, *save enroll* dan mengirim perintah OPEN melalui protokol HTTP yang dimana protokol HTTP ini langsung mengirim perintah dengan cepat tanpa harus ada loading beberapa detik, serta laptop juga menyimpan semua hasil sampel wajah yang sudah di enroll didalam sebuah folder *enrolled faces* [17]. Aplikasi Android juga digunakan untuk melakukan *Lock/Unlock* tanpa harus menunggu *verify* dari server.

2.4. Pengujian dan Revisi (Pengujian Kinerja & Akuisisi Data).

Pengujian dilakukan untuk memvalidasi hipotesis bahwa *edge computing* lebih andal. Pada tahapan pengujian ini dilakukan untuk mengetahui apakah seluruh komponen perangkat keras dan perangkat lunak telah berfungsi dengan baik dan terintegrasi secara optimal [18]. Fokus utama dalam pengujian ini mencakup beberapa aspek penting, seperti memastikan bahwa semua kabel dan sambungan pada perangkat keras telah terpasang dengan benar dan sesuai tempatnya, memverifikasi bahwa sistem telah terintegrasi dengan aplikasi mobile secara tepat, serta mengecek apakah semua perangkat keras seperti ESP32-CAM [19], modul kendali pintu, dan *edge server* dapat bekerja secara fungsional.

2.5. Presentasi dan Publikasi (Dokumentasi Laporan)

Tahap finalisasi di mana hasil log pengujian, metrik kinerja sistem, dan visualisasi data dirangkum. Dokumentasi ini tidak hanya berguna sebagai *Standard Operating Procedure* (SOP) pengoperasian alat di masa mendatang tetapi juga disusun menjadi manuskrip ilmiah untuk didesiminasikan melalui publikasi pada jurnal nasional terakreditasi maupun internasional.

3. Hasil dan Pembahasan

Pada bagian ini menjelaskan hasil pengujian terhadap prototipe sistem *smart door lock* yang telah dirancang bangun. Serangkaian pengujian yang dilakukan untuk mengukur performa dari integrasi teknologi *face recognition*, *edge computing*, serta Internet of Things (IoT). pemaparan hasil uji kinerja ini, akan dievaluasi sejauh mana sistem beroperasi secara optimal, serta membuktikan keberhasilan implementasi dalam menjawab rumusan masalah serta tujuan penelitian yang telah ditetapkan.

Dalam implementasi penelitian sistem *smart door lock*, yang mencakup proses pengambilan sampel wajah, pengiriman sampel wajah ke *edge server*, *verify* wajah, serta menu *lock/unlock* pada android. Pembahasan adalah penjelasan dasar, hubungan dan generalisasi yang ditunjukkan oleh hasil. Uraianya menjawab pertanyaan penelitian. Jika ada hasil yang meragukan maka tampilkan secara objektif.

3.1. Pengujian Perangkat

Pengujian dilakukan untuk menguji apakah perangkat dapat terhubung ke WiFi, menampilkan *IP address* dari *Web Camera Server* untuk melakukan pengambilan sampel wajah.

1. ESP32-CAM berhasil terhubung ke WiFi

```

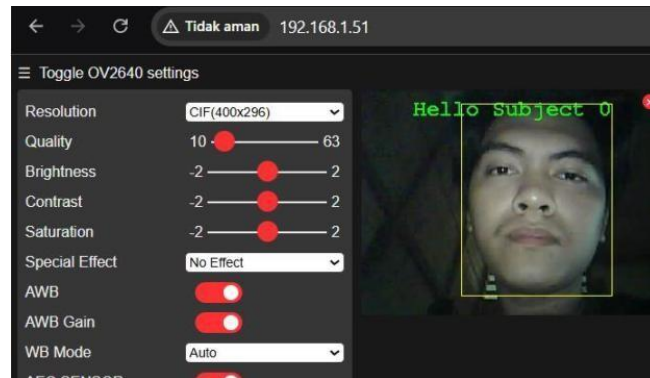
xx
WiFi Connected
Camera ready at IP: 192.168.1.51
Starting web server on port: '80'
Starting stream server on port: '81'

```

Gambar 4. Serial Monitor saat ESP32-CAM terhubung WiFi

2. Melakukan *Enroll face* pada Web Server Camera.

Jika IP Address sudah ada pada serial monitor pastikan IP Address dapat diakses pada web browser, jika sudah aktifkan menu start stream, *face detection*, *face recognition*, H-mirror (optional), sebagai langkah awal untuk mendaftarkan wajah. untuk melakukan pendaftaran wajah sesuaikan wajah dengan camera, dan pencahayaannya harus lumayan terang jangan redup. Berikut contoh wajah yang sudah terdaftar akan ada tulisan “Hello Subject 0”



Gambar 5. Hasil Dari Enroll Face Pertama

3. Melakukan pengecekan pengambilan sampel wajah

Pada serial monitor bisa dilihat saat *enroll face* ditekan akan ada pengambilan 5 sampel wajah, sampel wajah tersebut akan dikirim ke edge server untuk disimpan ke dalam folder yang sudah disediakan di awal, dan juga untuk verifikasi wajah supaya solenoid door lock terbuka.

```

Wajah baru dikenali (User 1), kirim OPEN
OPEN → NodeMCU via HTTP
Sending enroll to URL: http://192.168.1.49:5000/save_enroll?id=1&sample=3
Enroll response: {"file": "1_3.jpg", "status": "saved"}

Sending enroll to URL: http://192.168.1.49:5000/save_enroll?id=1&sample=4
Enroll response: {"file": "1_4.jpg", "status": "saved"}

Sending enroll to URL: http://192.168.1.49:5000/save_enroll?id=1&sample=5
Enroll response: {"file": "1_5.jpg", "status": "saved"}

```

Gambar 6. Sampel Wajah yang Telah Di Enroll

4. Memastikan sampel wajah berhasil dikirim ke Edge Server.

Setelah berhasil dikirim ke edge server dan disimpan dalam folder, kemudian arahkan wajah ke camera untuk melakukan verifikasi untuk membuka solenoid door lock, jika muncul lagi tulisan “Hallo Subject 0” untuk wajah pertama, maka di serial monitor akan ada output match face id menunjukkan wajah cocok, dan akan ada aksi pengiriman sinyal OPEN ke NodeMCU ESP32 untuk membuka solenoid door lock seperti gambar dibawah:

```

✔ Match Face ID: 0
✖ Wajah baru dikenali, kirim OPEN ke NodeMCU
✖ Verify failed, error: connection refused
🔌 CLOSE → NodeMCU via HTTP
✔ Perintah OPEN terkirim ke NodeMCU
Verify response: {"confidence":30.463936457061592,"message":"Face verified","status":"success","user":"1"}

✔ Wajah terverifikasi, kirim OPEN ke NodeMCU
MJPG: 13207B 375ms (0.3fps), AVG: 1135ms (0.9fps), 126+205+3246+144=3723 DETECTED 0
✔ Match Face ID: 0
MJPG: 13305B 116ms (0.9fps), AVG: 1138ms (0.9fps), 125+194+676+143=1140 DETECTED 0
✔ Match Face ID: 0
🔌 Verify response: {"confidence":31.015590397246307,"message":"Face verified","status":"success","user":"1"}

```

Gambar 7. Hasil Verifikasi Wajah

3.2. Pengujian Edge Server

Pengujian ini bertujuan untuk memastikan apakah semua data wajah dari perangkat keras ESP32-CAM dapat terkirim ke *Edge Server* menggunakan *python OpenCV* sesuai dengan struktur yang sudah dirancang

1. Memastikan Edge Server berhasil menerima sampel wajah.

Bisa dilihat pada terminal Visual Studio Code, Edge Server berhasil menerima semua sampel wajah yang di enroll melalui Web Server Kamera pada ESP32-CAM, semua sampel akan disimpan pada folder yang telah dibuat. Untuk setiap sampel wajah akan ada threshold yang ditentukan saat wajah berhasil disimpan threshold ini bertujuan untuk mencocokkan struktur wajah

```

[2025-12-09 14:06:47] ✔ Saved enroll: 1_1.jpg
192.168.1.51 - - [09/Dec/2025 14:06:47] "POST /save_enroll?id=1&sample=1 HTTP/1.1" 200 -
[2025-12-09 14:06:48] ✔ Saved enroll: 1_2.jpg
192.168.1.51 - - [09/Dec/2025 14:06:48] "POST /save_enroll?id=1&sample=2 HTTP/1.1" 200 -
• Predicted: 1 (confidence=31.27)
192.168.1.51 - - [09/Dec/2025 14:06:48] "POST /verify HTTP/1.1" 200 -
[2025-12-09 14:06:48] ✔ Saved enroll: 1_3.jpg
192.168.1.51 - - [09/Dec/2025 14:06:48] "POST /save_enroll?id=1&sample=3 HTTP/1.1" 200 -
[2025-12-09 14:06:49] ✔ Saved enroll: 1_4.jpg
192.168.1.51 - - [09/Dec/2025 14:06:49] "POST /save_enroll?id=1&sample=4 HTTP/1.1" 200 -
[2025-12-09 14:06:49] ✔ Saved enroll: 1_5.jpg
192.168.1.51 - - [09/Dec/2025 14:06:49] "POST /save_enroll?id=1&sample=5 HTTP/1.1" 200 -
• Predicted: 1 (confidence=31.83)

```

Gambar 8. Semua Hasil Sampel Wajah Terkirim ke Edge Server

2. Server mengirim verify request post ke ESP32-CAM.

Server akan mengirim request post verifikasi ke ESP32-CAM yang dimana ini bertujuan untuk apakah ada wajah yang ingin diverifikasi.

```

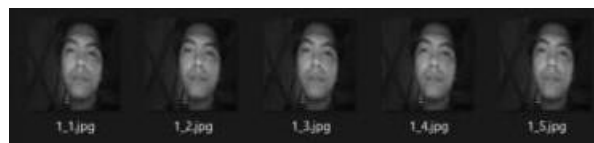
• Predicted: 1 (confidence=103.93)
192.168.1.51 - - [09/Dec/2025 14:08:20] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=103.08)
192.168.1.51 - - [09/Dec/2025 14:08:23] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=104.12)
192.168.1.51 - - [09/Dec/2025 14:08:25] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=107.19)
192.168.1.51 - - [09/Dec/2025 14:08:27] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=103.37)
192.168.1.51 - - [09/Dec/2025 14:08:29] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=104.39)
192.168.1.51 - - [09/Dec/2025 14:08:31] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=102.81)
192.168.1.51 - - [09/Dec/2025 14:08:33] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=98.89)
192.168.1.51 - - [09/Dec/2025 14:08:35] "POST /verify HTTP/1.1" 200 -
• Predicted: 1 (confidence=101.85)

```

Gambar 9. Server Mengirim Request Verify ke ESP32-CAM

3. Menyimpan semua hasil sampel wajah.

Sampel wajah yang berhasil di enroll dan berhasil disimpan dalam folder enrolled faces.



Gambar 9. Hasil Dari Sampel Wajah yang Disimpan dalam Folder

Inovasi fundamental dalam penelitian ini bertumpu pada pergeseran paradigma arsitektur dari *Cloud Computing*

Submitted : 01-06-2026 | Reviewed : 05-06-2026 | Accepted : 30-06-2026

menuju *Edge Computing*. Melalui pemindahan beban komputasi pengenalan wajah secara *Edge Computing*, penelitian ini secara efektif mengeliminasi ketergantungan mutlak terhadap konektivitas internet, sehingga memastikan sistem penguncian pintu tetap beroperasi secara otonom dan andal meskipun dalam kondisi offline. Sehingga, dapat memangkas transmisi data ke server eksternal demi memecahkan masalah waktu tunda (*latency*) untuk responsivitas *real-time*, transisi ke arsitektur *edge* ini juga mengusung prinsip *privacy-by-design* sebagai justifikasi keamanan utama; data biometrik wajah yang sangat sensitif hanya diekstraksi dan disimpan di dalam memori lokal perangkat, sehingga menutup celah peretasan di jaringan publik.

4. Kesimpulan

Berdasarkan pembahasan dan pengujian yang telah dilakukan, dapat disimpulkan bahwa penelitian ini telah berhasil menjawab permasalahan utama pada latar belakang melalui implementasi sistem *smart door lock* yang mengintegrasikan *face recognition*, *edge computing*, dan *Internet of Things* (IoT). Secara spesifik, sistem terbukti secara efektif mampu meningkatkan keamanan rumah dengan membatasi akses masuk secara eksklusif hanya untuk pengguna yang wajahnya telah di enroll atau yang sudah didaftarkan, dan verifikasi wajah yang bertujuan untuk mengakses *solenoid door lock* supaya terbuka otomatis. Sistem ini berhasil melakukan *face recognition* dan *face detection* menggunakan ESP32-CAM Pada *Edge Computing* sebagai fitur verifikasi wajah serta digunakan juga untuk metode penyimpanan data wajah. Implementasi integrasi antara perangkat IoT, *Edge Computing*, dan

Aplikasi Android dapat berjalan dengan baik. Hasil pengujian menunjukkan seluruh fitur yang diimplementasikan, termasuk, *face recognition*, *face detection*, *verify*, penyimpanan data, serta tombol *lock* dan *unlock* berfungsi dengan baik. Penggunaan arsitektur *edge computing* pada sistem ini berhasil menunjukkan tingkat akurasi pengenalan wajah yang tinggi dengan pemangkasan waktu tunda (*latency*) pemrosesan yang signifikan, sekaligus mengeliminasi risiko kebocoran data biometrik ke *cloud* publik. Keberhasilan pengamanan fisik dan privasi tersebut disempurnakan oleh kinerja integrasi IoT yang andal. Untuk pengembangan lebih lanjut, penulis memberikan beberapa saran yang diharapkan dapat memberikan manfaat yang lebih optimal Sistem dapat ditingkatkan dengan penggunaan kamera yang lebih bagus seperti *webcam* yang memiliki akurasi yang lebih tinggi dan kualitas kamera jauh lebih baik serta Menambahkan fitur akses kamera pada Android, untuk bisa melakukan pengecekan pada rumah, jadi tidak perlu membuka *Web Server Camera*, cukup pada aplikasinya. Menambahkan fitur sidik jari pada aplikasi Android sebagai keamanan supaya tidak dibajak dari user lain. Memungkinkan user saat verifikasi wajah menggunakan berbagai ekspresi wajah agar tetap terdeteksi.

Daftar Rujukan

- [1] M. Z. Abdillah, M. U. Harun Al Rasyid, and R. Sigit, "The Implementation of Face Recognition Using Deep Metric Learning for Automatic Door Openers Based on Edge Computing," *JOIV : International Journal on Informatics Visualization*, vol. 10, no. 1, p. 146, Jan. 2026, doi: 10.62527/joiv.10.1.3851.
- [2] P. Badoni, M. Wadhwa, and R. Walia, "System of IntelliGuard Access Using IoT," in *2024 International Conference on Intelligent Systems for Cybersecurity (ISCS)*, IEEE, May 2024, pp. 1–6. doi: 10.1109/ISCS61804.2024.10581055.
- [3] N. Harun and M. Shamian Zainal, "Development of Face Recognition Smart Door Lock System Using ESP32-CAM and Telegram Application As Media Control and Monitoring," *Progress in Engineering Application and Technology*, vol. 4, no. 2, pp. 35–048, 2023, doi: 10.30880/peat.2023.04.02.004.
- [4] M. Nasir and U. Erdiansyah, "Implementation of WebSocket in an IoT-Based Smart Home Door Security System Using ESP32-CAM with Face Recognition," *Journal of Artificial Intelligence and Software Engineering*, vol. 6, no. 1, pp. 55–67, 2026, doi: 10.30811/jaise.v6i1.9052.
- [5] T. Handayani, A. Basuki, S. Sudiana, and I. Dirgantara, "Rancang Bangun Sistem Keamanan Pintu menggunakan Metode Pengenalan Wajah berbasis Internet of Things," *AVITEC*, vol. 5, no. 1, p. 1, Dec. 2022, doi: 10.28989/avitec.v5i1.1393.
- [6] D. A. Pratama and M. B. Ulum, "RANCANG BANGUN SISTEM KEAMANAN RUMAH DENGAN FACE RECOGNITION BERBASIS ESP32-CAM," *Jurnal Komputasi*, vol. 12, no. 1, pp. 70–78, Apr. 2024, doi: 10.23960/komputasi.v12i1.241.
- [7] E. P. Lumbanraja, S. Saniman, and T. Tugiono, "Sistem Monitoring Keamanan Brankas Menggunakan Face Recognition Berbasis Mikrokontroler ESP32-CAM," *Jurnal Sistem Komputer Triguna Dharma (JURSIK TGD)*, vol. 2, no. 3, pp. 169–176, May 2023, doi: 10.53513/jursik.v2i3.6560.
- [8] Dr.Ravi Bolimera, Dr.S.Karthick, Sunkoju.Karthikeya, Vavilala.Ankith Kumar, and Resoju.Sai Siddhartha, "ESP32-CAM FACE DETECTION AUTOMATIC DOOR LOCK," *International Journal of Data Science and IoT Management System*, vol. 4, no. 4(2), pp. 107–111, Dec. 2025, doi: 10.64751/ijdim.2025.v4.n4(2).pp107-111.
- [9] F. Firdayanti *et al.*, "Integrated Face Recognition and IoT-Based Electronic Equipment Management System," *Journal of Applied Science, Technology & Humanities*, vol. 1, no. 1, pp. 35–48, Feb. 2024, doi: 10.62535/k4rv4267.
- [10] E. Octavia, R. Dijaya, A. Eviyanti, and N. L. Azizah, "Rancangan Bangun Sistem Keamanan Rumah Kost Berbasis IoT dengan ESP32-Cam," *Indonesian Journal of Applied Technology*, vol. 1, no. 3, p. 16, Jul. 2024, doi: 10.47134/ijat.v1i3.3073.

- [11] Rizal Fahmi Alifa, “Analisis Skalabilitas Sistem Pengenalan Wajah menggunakan Library Face Recognition di Lingkungan Cloud Computing,” *Explore*, vol. 14, no. 2, pp. 113–117, Jul. 2024, doi: 10.35200/ex.v14i2.120.
- [12] B. Das and K. K. Halder, “Face Recognition Using ESP32-Cam for Real-Time Tracking and Monitoring,” in *2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems (iCACCESS)*, IEEE, Mar. 2024, pp. 01–06. doi: 10.1109/iCACCESS61735.2024.10499606.
- [13] V. Viantyazar Septyanlie, V. Ikawati, E. Subiyanta, N. Lestari, and J. K. Saleh I Desa Cipinang Kecamatan Rajagaluh, “Face Recognition-Based Door Lock Security System Using TensorFlow Lite,” *Journal homepage: Journal of Electrical Engineering and Computer (JEECOM)*, vol. 6, no. 2, 2024, doi: 10.33650/jeeecom.v4i2.
- [14] A. Ipanhar, T. K. Wijaya, and P. Gunoto, “PERANCANGAN SISTEM MONITORING PINTU OTOMATIS BERBASIS IOT MENGGUNAKAN ESP32-CAM,” *SIGMA TEKNIKA*, vol. 5, no. 2, pp. 333–350, Nov. 2022, doi: 10.33373/sigmateknika.v5i2.4590.
- [15] Suradi, A. Martani, I. Arfiani, and Sarli, “Perancangan Sistem Pintu Otomatis Menggunakan ESP32CAM,” *Jurnal Teknologi dan Komputer (JTEK)*, vol. 2, no. 01, Jun. 2022, doi: 10.56923/jtek.v2i01.60.
- [16] B. Zuna *et al.*, “Implementasi Edge IoT dan ThingsBoard untuk Monitoring Kandang Sapi Real-Time,” *Jurnal Pustaka AI (Pusat Akses Kajian Teknologi Artificial Intelligence)*, vol. 5, no. 3, pp. 522–527, Dec. 2025, doi: 10.55382/jurnalpustakaai.v5i3.1563.
- [17] R. Gustia, W. Febriani, M. Hasanah, and A. Fradana, “Perancangan Smart Home Berbasis Iot Menggunakan Esp32, Telegram dan Spreadsheet,” *Jurnal Pustaka AI (Pusat Akses Kajian Teknologi Artificial Intelligence)*, vol. 5, no. 3, pp. 695–700, Dec. 2025, doi: 10.55382/jurnalpustakaai.v5i3.1482.
- [18] P. D. P. Adi and Y. Wahyu, “Performance evaluation of ESP32 Camera Face Recognition for various projects,” *Internet of Things and Artificial Intelligence Journal*, vol. 2, no. 1, pp. 10–21, Feb. 2022, doi: 10.31763/iota.v2i1.512.
- [19] A. Anisa, G. A. Mutiara, and M. R. Alfarisi, “Implementation of Real-Time Face Recognition for Secure Weapon Storage Access Control,” *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 8, no. 2, pp. 689–695, Apr. 2026, doi: 10.35882/jeeemi.v8i2.1608.