

**Implementasi Internet of Things Untuk Mengontrol Peralatan Listrik
Pada Rumah Tinggal Menggunakan ESP32-CAM**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk meraih gelar serjana terapan (D-4) pada

Program Studi Teknologi Rekayasa Komputer Jaringan

Fakultas Teknologi Industri

Universitas Bung Hatta



Oleh :

Muhammad Azim Zuhaimi

2110017514013

PROGRAM STUDI TEKNOLOGI REKAYASA KOMPUTER JARINGAN

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS BUNG HATTA

PADANG

2026

LEMBAR PENGESAHAN
IMPLEMENTASI INTERNET OF THINGS UNTUK MENGONTROL PERALATAN
LISTRIK PADA RUMAH TINGGAL MENGGUNAKAN ESP32-CAM

TUGAS AKHIR

*Diajukan sebagai salah satu syarat untuk meraih gelar sarjana terapan (D-IV) pada
Program Studi Teknologi Rekayasa Komputer Jaringan*

*Fakultas Teknologi Industri
Universitas Bung Hatta*

Oleh:

MUHAMMAD AZIM ZUHAIMI

NPM: 2110017514013

Disetujui Oleh:
Pembimbing



Dr. Hidayat, ST, MT, IPM

NIDN : 1031057001

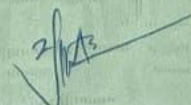
Diketahui Oleh:

Fakultas Teknologi Industri
Dekan,



Prof. Dr. Eng. Ir. Reni Desmiarti, S.T., M.T.
NIK : 990 500 496

Program Studi
Teknologi Rekayasa Komputer Jaringan
Ketua,



Zulfadli, S. Kom, M. Sc
NIDN: 1002058801



**FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNOLOGI REKAYASA KOMPUTER JARINGAN**

BERITA ACARA SEMINAR TUGAS AKHIR

Nomor : 10.0/Ata/FTI-TRKJ/II-2026

Pada hari ini *Sabtu* tanggal *Tujuh* bulan *Maret* tahun *Dua Ribu Dua Puluh Enam* kami panitia *Seminar Tugas Akhir Prodi Teknologi Rekayasa Komputer Jaringan (TRKJ)* Fakultas Teknologi Industri Universitas Bung Hatta, yang diangkat berdasarkan Surat Keputusan Dekan Fakultas Teknologi Industri Universitas Bung Hatta Nomor : 338.0/SK-Atk.16/FTI/Ut-2025. tanggal 07 Maret 2025, menyelenggarakan Seminar Proposal yang dimaksud terhadap mahasiswa :

Nama : Muhammad Azim Zuhaimi
 NPM : 2110017514013
 Dosen Pembimbing : Dr. Hidayat, ST, MT, IPM
 Judul Tugas Khusus : IMPLEMENTASI INTERNET OF THINGS UNTUK MENGONTROL PERALATAN ALAT LISTRIK PADA RUMAH TINGGAL MENGGUNAKAN ESP32-CAM

Dengan mengingat serta menimbang kami memutuskan bahwa mahasiswa yang tersebut diatas dinyatakan :

LULUS/GAGAL	
Nilai Angka	78
Nilai Huruf	B+

Prodi Teknologi Rekayasa Komputer Jaringan
Ketua Prodi

Padang, 07 Maret 2026

(Zulfadli, S.Kom, M.Sc)

Ketua Sidang,

(Dr.Hidayat, ST, MT, IPM)

Anggota :

1. Annita
2. Budi Sonargo
3. Hermyat

- 1.
- 2.
- 3.

Kampus Proklamator I : Jl. Sumatera Ulak Karang Padang, 25133, Telp. (0751) 7051670-7052005, Fax. (0751) 7065475
 Kampus Proklamator II : Jl. Bagindo Aziz Chan By Pass Ale Pacah Padang, Telp. (0751) 469250
 Kampus Proklamator III : Jl. Gajah Mada No.19, Olo Nenggalo, Padang 25143, Telp. (0751) 7051257, Fax. (0751) 7051341
 E-mail : sekretariat.rektor@bunghatta.ac.id, rektorat@bunghatta.ac.id, ihumas@bunghatta.ac.id

www.bunghatta.ac.id

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa ini sebagian maupun keseluruhan Tugas Akhir saya dengan judul “ **Implementasi Internet of Things Untuk Mengontrol Peralatan Listrik Pada Rumah Tinggal Menggunakan ESP32-CAM**” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua refrensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Padang, 07 Maret 2026



Muhammad Azim Zuhaimi
NPM: 211001751401

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT, karena atas berkat rahmat dan karunia-Nya penulis dapat menyelesaikan Tugas Akhir Tugas akhir dengan judul Implementasi Internet of Things Sistem Kontrol Listrik Berbasis Deteksi Manusia Menggunakan ESP32 – CAM dan Algoritma YOLO

Dalam penyusunan laporan Tugas Akhir ini, penulis banyak menerima bantuan dan dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini, penulis ingin mengucapkan terima kasih kepada:

1. Allah SWT yang telah mengizinkan hamba untuk menyelesaikan penulisan Tugas Akhir ini serta memberikan kesehatan.
2. Kedua orang tua yang selalu mendo'akan saya sebagai penulis, memberikan semangat serta memberikan nasehat kepada saya dan dapat membaggakan kedua orang tua saya.
3. Ibu Prof. Dr. Eng. Reni Desmiarti, ST. MT selaku Dekan Fakultas Teknologi Industri Universitas Bung Hatta Padang.
4. Bapak Zufadli, S.Kom, M.Sc selaku Kaprodi Teknologi Rekayasa Komputer Jaringan yang selalu memberi arahan terbaik.
5. Bapak Dr. Hidayat, ST, MT, IPM. selaku Dosen Pembimbing yang telah membimbing penulis selama ini sehingga dapat menyelesaikan penulisan tugas akhir ini.
6. Bapak/Ibu Dosen Program Studi Teknologi Rekayasa Komputer Jaringan Universitas Bung Hatta.
7. Terimakasih kepada teman-teman seperjuangan angkatan 21 dan seluruh angkatan TRKJ yang telah memberikan support dan saran dalam penulisan tugas akhir ini.

Penulis menyadari bahwa dalam menyelesaikan laporan ini, masih jauh dari kata sempurna, sehingga penulis memohon maaf.

Padang, 7 Maret 2026

Muhammad Azim Zuhaimi

DAFTAR ISI

KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR TABEL	x
BAB I.....	1
PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat Penelitian	4
BAB II	5
TINJAUAN PUSTAKA	5
2.1 Tinjauan Pustaka	5
2.1.1 Designing Home Security With ESP32-Cam and IoT-Based Alarm Notification Using Telegram	5
2.1.2 Embedded System for Human Detection Applied to Domotics.....	6
2.1.3 YOLO Algorithm for Detecting People in Social Distancing System	6
2.2 Landasan Teori.....	7
2.2.1 Internet of Things (IoT).....	7
2.2.2 ESP32-CAM.....	7
2.2.3 Sensor HC-SR501 (PIR Sensor)	8
2.2.4 Modul Relay.....	9
2.2.5 Switching Power Supply 5v	9
2.2.6 Arduino Uno.....	10
2.2.7 Algoritma YOLO (<i>You Only Look Once</i>).....	11
2.2.8 Arduino IDE	11
2.2.9 Telegram.....	12
2.2.10 Waterfall	13

2.2.11 UML (Unified Modelling Language).....	13
BAB III.....	18
METODOLOGI PENELITIAN	18
3.1 Metode	18
3.2 Metode Pengumpulan Data.....	18
3.3 Perancangan Sistem	20
3.3.1 Perancangan Hardware.....	20
3.3.2 Perancangan Software	23
3.4 Tahapan Pengembangan Sistem.....	26
3.4.1 Analisis Kebutuhan	26
3.4.2 Perancangan Sistem.....	26
3.4.4 Implementasi Sistem	34
3.4.5 Pengujian Sistem	34
3.4.6 Pemeliharaan dan Evaluasi.....	35
3.5 Teknik Pengujian Sistem.....	35
3.5.1 Uji Fungsionalitas	35
3.5.2 Uji Akurasi Deteksi	35
3.5.3 Pengujian Waktu Respon.....	35
3.5.4 Dokumentasi Hasil	36
3.6 Alat dan Bahan Penelitian.....	36
3.6.1 Alat dan Bahan Perangkat Keras	36
3.6.2 Alat dan Bahan Perangkat Lunak	36
BAB IV.....	38
HASIL DAN PEMBAHASAN	38
4.1 Implementasi Sistem.....	38
4.1.1 Implementasi Perangkat Keras	38
4.1.2 Implementasi Perangkat Lunak	39
4.1.3 Implementasi YOLO pada Server	41
4.2 Hasil Pengujian Sistem	42

4.2.1 Pengujian Fungsionalitas.....	42
4.2.2 Pengujian Akurasi Deteksi	44
4.2.3 Pengujian Waktu Respon.....	45
4.3 Analisis Hasil Pengujian	46
4.4 Efektivitas Sistem Terhadap Penghematan Energi	46
4.5 Kelebihan dan Keterbatasan Sistem.....	47
4.5.1 Kelebihan Sistem.....	47
4.5.2 Keterbatasan Sistem	48
BAB V	49
KESIMPULAN DAN SARAN	49
5.1 Kesimpulan	49
5.2 Saran	49
DAFTAR PUSTAKA	51

DAFTAR GAMBAR

Gambar 2. 1 ESP32-CAM.....	8
Gambar 2. 2 Sensor Pir.....	8
Gambar 2. 3 Modul Relay	9
Gambar 2. 4 Arduino Uno	10
Gambar 3 1 Metode Waterfal.....	18
Gambar 3 2 Wiring Rancangan Alat.....	21
Gambar 3 3 Diagram Perancangan Sistem	22
Gambar 3 4 Activity Diagram Sistem	24
Gambar 4. 1 Alat yang sudah jadi.....	38
Gambar 4. 2 Hasil Serial Monitor dari ESP32-CAM dan Sensor Pir.....	40
Gambar 4. 3 Hasil Notifikasi Bot Telegram	40
Gambar 4. 4 Hasil Deteksi YOLO.....	41
Gambar 4. 5 Pengujian Hasil Tangkapan ESP32-CAM	42
Gambar 4. 6 Hasil Pengujian Dari Relay dan Lampu Berfungsi.....	43

DAFTAR TABEL

Tabel 2.1 Use Case Diagram	14
Tabel 2.2 Activity Diagram	15
Tabel 3.1 Alat dan Bahan Perangkat Keras	36
Tabel 3.2 Alat dan Bahan Perangkat Lunak	37
Tabel 4. 1Pengujian Berdasarkan Jarak Objek	44
Tabel 4. 2 Pengujian Berdasarkan Kondisi Pencahayaan.....	44
Tabel 4. 3Kecepatan Jaringan Terhadap Waktu Respon.....	45
Tabel 4. 4 Pengaruh Proses Deteksi Terhadap Waktu Respon.....	45
Tabel 4. 5 Pengaruh Komunikasi Sensor terhadap Respon Sistem	46

ABSTRAK

Perkembangan teknologi Internet of Things (IoT) memberikan peluang untuk meningkatkan efisiensi penggunaan energi listrik melalui sistem otomatisasi. Salah satu permasalahan yang sering terjadi pada rumah tinggal adalah perangkat listrik seperti lampu yang tetap menyala meskipun tidak ada aktivitas manusia di dalam ruangan, sehingga menyebabkan pemborosan energi. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem kontrol peralatan listrik berbasis deteksi manusia menggunakan ESP32-CAM dan algoritma YOLO. Sistem bekerja dengan cara menangkap gambar menggunakan ESP32-CAM, kemudian gambar dikirim ke server untuk diproses menggunakan algoritma YOLO dalam mendeteksi keberadaan manusia. Jika manusia terdeteksi, sistem akan mengirimkan sinyal untuk mengaktifkan modul relay sehingga lampu menyala secara otomatis.

Metode penelitian yang digunakan adalah metode pengembangan sistem Waterfall yang meliputi tahap analisis, perancangan, implementasi, pengujian, dan evaluasi sistem. Pengujian dilakukan pada beberapa kondisi jarak dan pencahayaan untuk mengetahui performa sistem dalam mendeteksi manusia serta waktu respon dalam mengaktifkan lampu.

Berdasarkan hasil pengujian, sistem mampu mendeteksi keberadaan manusia pada jarak 1–4 meter dengan hasil yang baik, sedangkan pada jarak 5 meter akurasi mulai menurun. Waktu respon sistem berkisar antara 1–2 detik pada kondisi jaringan stabil dan dapat meningkat hingga lebih dari 3 detik pada jaringan yang lambat. Hasil penelitian menunjukkan bahwa sistem yang dirancang dapat bekerja dengan baik sebagai sistem kontrol lampu otomatis berbasis deteksi manusia untuk meningkatkan efisiensi penggunaan energi listrik pada rumah tinggal.

Kata kunci: Internet of Things, ESP32-CAM, YOLO, deteksi manusia, kontrol lampu otomatis.

ABSTRACT

The development of Internet of Things (IoT) technology provides opportunities to improve energy efficiency through automation systems. One common problem in residential environments is that electrical devices such as lights remain turned on even when there is no human activity in the room, resulting in energy waste. This study aims to design and implement an electrical device control system based on human detection using ESP32-CAM and the YOLO algorithm. The system works by capturing images using the ESP32-CAM module and sending them to a server for processing using the YOLO algorithm to detect the presence of humans. When a human is detected, the system sends a signal to activate a relay module, which automatically turns on the household light.

The research method used in this study is the Waterfall development method, which includes the stages of analysis, design, implementation, testing, and evaluation. System testing was conducted under various conditions of distance and lighting to evaluate the system's ability to detect humans and measure the response time for activating the light.

The test results show that the system is able to detect humans at a distance of 1–4 meters with good accuracy, while detection performance begins to decrease at a distance of 5 meters. The system response time ranges from 1–2 seconds under stable network conditions and can exceed 3 seconds under slower network conditions. These results indicate that the proposed system can function effectively as an automatic lighting control system based on human detection to improve energy efficiency in residential environments.

Keywords: Internet of Things, ESP32-CAM, YOLO, human detection, automatic lighting control.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan pesatnya perkembangan teknologi, berbagai aspek kehidupan manusia mengalami perubahan yang signifikan, termasuk dalam hal pengelolaan energi listrik. Salah satu tantangan besar dalam pengelolaan energi adalah memastikan bahwa penggunaan listrik berlangsung secara efisien. Pada kenyataannya, perangkat elektronik seperti lampu, pendingin ruangan, dan peralatan listrik lainnya kerap dibiarkan menyala meskipun tidak ada aktivitas manusia di sekitarnya. Hal ini menyebabkan pemborosan energi yang cukup besar serta biaya listrik yang meningkat [1].

Perkembangan teknologi *Internet of Things* (IoT) telah memberikan dampak besar terhadap cara manusia mengelola berbagai aspek kehidupan, termasuk dalam upaya meningkatkan efisiensi penggunaan energi listrik. Dalam kehidupan sehari-hari, sering kali perangkat listrik seperti lampu atau pendingin ruangan tetap menyala meskipun tidak sedang digunakan, yang menyebabkan pemborosan energi. Untuk mengatasi hal tersebut, diperlukan sistem otomatis yang mampu mengenali keberadaan manusia secara cerdas agar perangkat listrik hanya beroperasi ketika dibutuhkan. Salah satu pendekatan yang efektif adalah dengan memanfaatkan teknologi deteksi berbasis penglihatan. Dalam hal ini, ESP32-CAM menjadi pilihan yang tepat karena memiliki modul kamera terintegrasi serta kemampuan komputasi dan konektivitas nirkabel yang mendukung penerapan sistem deteksi manusia secara real-time [2], [3].

Dengan demikian, penggunaan ESP32-CAM dapat membantu mewujudkan sistem pengendalian perangkat listrik yang efisien, cerdas, dan hemat energi. Selain itu, algoritma deteksi objek canggih seperti YOLO (*You Only Look Once*) memiliki keunggulan dalam mengenali dan melacak objek secara real-time dengan akurasi tinggi. Dengan kecepatan deteksi yang luar biasa [4], [5]. YOLO cocok untuk diintegrasikan dengan perangkat seperti ESP32-CAM dalam aplikasi sistem otomatisasi listrik berbasis IoT. Kombinasi teknologi ini memungkinkan sistem untuk mendeteksi manusia secara efisien dan menyesuaikan pengoperasian perangkat listrik sesuai dengan kehadiran manusia. [6].

Melihat tantangan tersebut, pengembangan sistem otomatisasi listrik berbasis deteksi manusia menjadi solusi yang sangat potensial dalam mengurangi pemborosan energi. Sistem ini memungkinkan perangkat listrik di suatu ruangan untuk bekerja hanya saat terdeteksi adanya aktivitas manusia, sehingga efisiensi penggunaan energi dapat ditingkatkan.

Dengan memanfaatkan teknologi deteksi berbasis kamera yang digabungkan dengan kecerdasan buatan, diharapkan solusi ini dapat memberikan kontribusi yang nyata terhadap efisiensi energi yang lebih baik dan penggunaan energi yang berkelanjutan. Lebih jauh lagi, penerapan sistem otomatisasi berbasis deteksi manusia ini dapat memperpanjang umur perangkat listrik dengan mengurangi waktu operasional yang tidak perlu. Dengan mengurangi frekuensi penyalaan dan pemadaman perangkat secara manual, keausan komponen dapat diminimalkan. Hal ini tidak hanya meningkatkan efisiensi energi, tetapi juga mengurangi biaya perawatan dan penggantian perangkat dalam jangka panjang. Dalam skala yang lebih luas, adopsi teknologi ini dapat memberikan dampak positif bagi lingkungan dengan menurunkan emisi karbon yang dihasilkan dari pemborosan energi listrik [1], [3], [7].

Saat ini, data menunjukkan bahwa sektor perumahan dan komersial menyumbang sekitar 40% dari total konsumsi energi global, dengan sebagian besar energi digunakan untuk perangkat pencahayaan, pemanas, ventilasi, dan pendinginan. Menurut laporan dari International Energy Agency (IEA), sekitar 10-15% energi listrik yang dikonsumsi terbuang sia-sia karena perangkat elektronik yang tetap menyala meskipun tidak digunakan. Dengan angka ini, pengelolaan energi yang lebih cerdas menjadi kebutuhan yang mendesak [1], [7].

Penerapan teknologi IoT berbasis deteksi visual seperti ESP32-CAM dapat memberikan solusi konkret untuk mengurangi pemborosan tersebut. Sebagai contoh, studi menunjukkan bahwa sistem deteksi keberadaan manusia berbasis kamera dapat mengurangi konsumsi energi hingga 30% dalam aplikasi skala bangunan perkantoran, terutama untuk perangkat pencahayaan otomatis. Teknologi ini mampu mengidentifikasi aktivitas manusia secara real-time, dan ketika dikombinasikan dengan algoritma deteksi objek seperti YOLO, performa sistem otomatisasi listrik dapat ditingkatkan secara signifikan [5][3].

Penggunaan Internet of Things (IoT) pada proyek ini sangat penting karena memungkinkan sistem untuk terhubung, otomatis, dan cerdas. Dengan IoT, data dari perangkat ESP32-CAM dapat dikirimkan secara real-time ke perangkat lain, seperti ponsel atau server, untuk pemrosesan lebih lanjut atau pemberitahuan langsung kepada pemilik rumah. IoT juga memungkinkan pengawasan jarak jauh, penghematan sumber daya, dan integrasi dengan teknologi lain, seperti cloud computing atau perangkat rumah pintar. Dengan IoT, sistem deteksi manusia dapat bekerja secara efisien, responsif, dan mudah dikelola, menjadikannya solusi unggul untuk meningkatkan keamanan dan kenyamanan rumah tinggal [8].

ESP32-CAM adalah pilihan ideal untuk proyek ini karena perangkat ini menggabungkan kamera, modul Wi-Fi, dan prosesor dalam satu paket yang ringkas, terjangkau, dan efisien. Dengan kemampuan mengirimkan gambar atau video secara real-time melalui jaringan, ESP32-CAM sangat sesuai untuk aplikasi IoT seperti sistem deteksi manusia. Perangkat ini juga hemat daya, mendukung integrasi dengan algoritma seperti YOLO, dan memiliki dukungan komunitas yang luas, sehingga mempermudah pengembangan dan pemeliharaan sistem. Keunggulan ini menjadikan ESP32-CAM solusi terbaik untuk menciptakan sistem keamanan rumah tinggal yang modern, ekonomis, dan mudah diimplementasikan [7].

Algoritma YOLO (You Only Look Once) adalah pilihan terbaik untuk sistem deteksi manusia pada proyek ini karena menawarkan kecepatan, akurasi, dan efisiensi yang sangat sesuai dengan kebutuhan sistem berbasis IoT dan ESP32-CAM. YOLO memungkinkan deteksi manusia secara real-time dengan kemampuan membedakan manusia dari objek lain secara akurat. Selain itu, varian ringan seperti Tiny-YOLO dapat diadaptasi untuk perangkat dengan keterbatasan komputasi, menjadikannya ideal untuk perangkat seperti ESP32-CAM yang kecil dan hemat daya. Dengan menggunakan YOLO, sistem dapat memberikan notifikasi instan kepada pemilik rumah saat mendeteksi manusia, sehingga meningkatkan keamanan dan kenyamanan dalam pengelolaan rumah tinggal secara cerdas dan otomatis [9], [7].

1.2 Rumusan Masalah

Adapun pembuatan tugas akhir ini terdapat 3 tujuan, yaitu:

1. Bagaimana merancang sistem Peralatan listrik berbasis ESP32-CAM yang dapat mendeteksi keberadaan manusia secara akurat?
2. Bagaimana menerapkan algoritma YOLO untuk mendeteksi manusia secara real-time dengan keandalan yang tinggi?
3. Bagaimana mengimplementasikan IoT agar terhubung dengan telegram?

1.3 Tujuan Penelitian

Adapun pembuatan tugas akhir ini terdapat 3 tujuan, yaitu:

1. Terciptanya Prototype sistem otomatisasi listrik yang dapat mendeteksi keberadaan manusia dengan menggunakan ESP32-CAM.
2. Mengaplikasikan algoritma YOLO dalam deteksi manusia secara real-time.
3. Mengevaluasi efektivitas sistem dalam mengurangi penggunaan listrik yang boros.

1.4 Batasan Masalah

Adapun pembuatan tugas akhir ini terdapat 3 batasan masalah, yaitu:

1. Rancang sistem menggunakan ESP32-CAM sebagai perangkat utama untuk mendeteksi keberadaan manusia.
2. Algoritma yang digunakan untuk deteksi adalah YOLO, dengan modifikasi tertentu agar dapat berfungsi optimal di perangkat dengan kapasitas terbatas seperti ESP32-CAM.
3. Pengujian sistem akan dilakukan dalam lingkungan dalam ruangan (*indoor*) dengan kondisi pencahayaan yang memadai.
4. Implementasi menggunakan IOT dan telegram
5. Mengimplementasi menggunakan dashboard

1.5 Manfaat Penelitian

Adapun pembuatan tugas akhir ini terdapat 3 manfaat, yaitu:

1. Menyediakan cara yang efisien untuk menghemat energi listrik, yang dapat mengurangi pengeluaran biaya listrik di rumah.
2. Memberikan wawasan baru dan pengalaman dalam pengembangan sistem IOT dan penerapan algoritma pendeteksian objek.
3. Menawarkan solusi efisien untuk bangunan pintar yang dapat memaksimalkan penghematan energi.

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian terkait merupakan landasan penting dalam pengembangan sistem berbasis *Internet of Things* (IoT) dan algoritma komputer vision. Dalam konteks ini, berbagai studi telah dilakukan untuk mengimplementasikan teknologi seperti ESP32-CAM dan algoritma YOLO guna mendeteksi keberadaan manusia, meningkatkan efisiensi energi, dan menciptakan sistem otomatisasi yang lebih cerdas. Kajian ini bertujuan untuk memberikan gambaran komprehensif tentang pendekatan, hasil, dan kontribusi dari penelitian-penelitian terdahulu yang relevan. Dengan memahami berbagai metode yang telah diterapkan, keunggulan yang diperoleh, serta tantangan yang dihadapi, penelitian ini berupaya mengidentifikasi peluang pengembangan lebih lanjut. Pembahasan berikut menyajikan tinjauan dari beberapa penelitian yang mendukung implementasi IoT untuk kontrol listrik berbasis deteksi manusia menggunakan ESP32-CAM dan algoritma YOLO, sekaligus memberikan komparasi terhadap fokus penelitian yang sedang dikembangkan [10], [3].

2.1.1 Designing Home Security With ESP32-Cam and IoT-Based Alarm Notification Using Telegram

Penelitian ini menggunakan modul ESP32-CAM yang dilengkapi sensor PIR untuk mendeteksi gerakan manusia. Sistem ini menghasilkan notifikasi otomatis melalui aplikasi Telegram dan mengaktifkan alarm sebagai bentuk respons keamanan. Implementasi sistem melibatkan pengaturan perangkat keras yang stabil dengan error tegangan rata-rata hanya 3,9%, memastikan perangkat dapat bekerja dengan konsisten meskipun terjadi fluktuasi kecil. Sistem berhasil mendeteksi kehadiran manusia dalam jangkauan sensor PIR dengan akurasi tinggi, menunjukkan potensi besar untuk penerapan di skenario keamanan rumah. Selain itu, notifikasi real-time yang dikirimkan melalui Telegram memungkinkan pengguna untuk mendapatkan informasi cepat dan akurat terkait ancaman keamanan. Sistem ini memiliki kesamaan menggunakan ESP32-CAM sebagai modul utama untuk mendeteksi manusia, dengan fokus pada pengolahan data visual. Penelitian ini menghadirkan solusi inovatif dalam pengelolaan perangkat listrik berbasis IoT, yang tidak hanya meningkatkan efisiensi energi melalui otomatisasi, tetapi juga memberikan kemudahan akses jarak jauh untuk pengaturan yang lebih praktis dan ekonomis. Selain itu, sistem ini dirancang untuk mendukung keberlanjutan dengan mengurangi pemborosan energi melalui pengelolaan yang lebih cerdas dan terintegrasi.

2.1.2 Embedded System for Human Detection Applied to Domotics

Sistem ini menggunakan *Raspberry Pi 3 B* yang didukung oleh kamera V2 dan sensor PIR. Untuk deteksi manusia, digunakan algoritma multilayer perceptron yang dioptimalkan untuk komputasi tinggi pada platform *Raspberry Pi*. Implementasi algoritma melibatkan pengolahan citra secara real-time dengan memanfaatkan teknik *High Performance Computing (HPC)*. Penelitian ini mencapai akurasi deteksi manusia sebesar 96%, yang cukup tinggi untuk aplikasi keamanan rumah atau otomasi bangunan. Platform *Raspberry Pi* memungkinkan pengolahan data yang kompleks, meskipun memerlukan daya komputasi lebih besar dibandingkan ESP32-CAM [11].

Penelitian memiliki kesamaan dengan penelitian yang akan kita lakukan yang menggunakan teknologi *embedded system* untuk deteksi manusia secara real-time. Penelitian ini menggunakan *Raspberry Pi* yang lebih canggih, sedangkan penelitian kita memanfaatkan ESP32-CAM untuk efisiensi biaya dan energi, membuatnya lebih cocok untuk aplikasi berskala kecil dan hemat daya [11].

2.1.3 YOLO Algorithm for Detecting People in Social Distancing System

Sistem ini menggunakan algoritma YOLOv3 untuk mendeteksi manusia secara akurat dalam konteks penerapan jarak sosial. Pendekatan ini juga mengintegrasikan algoritma *Euclidean Distance* untuk menghitung jarak antar individu berdasarkan koordinat yang dihasilkan dari deteksi objek. Uji coba dilakukan pada dataset video luar ruangan untuk mengevaluasi performa sistem dalam lingkungan nyata. Sistem menunjukkan kinerja yang baik dengan *F1-score* di atas 0.8, bahkan ketika mendeteksi individu dalam ukuran kecil pada kerangka kamera [12].

Hal ini mengindikasikan bahwa algoritma YOLOv3 cukup robust untuk aplikasi berbasis deteksi manusia di berbagai kondisi pencahayaan dan sudut pandang. Sama dengan penelitian yang di lakukan penulis yang menggunakan algoritma YOLO untuk deteksi manusia secara visual. Penelitian ini difokuskan pada pengukuran jarak sosial untuk kepatuhan terhadap protokol kesehatan, sementara penelitian kita bertujuan pada kontrol perangkat listrik berbasis keberadaan manusia untuk efisiensi energi. Selain itu, penelitian kita juga memanfaatkan algoritma ini untuk mengoptimalkan respons sistem secara real-time, sehingga meningkatkan akurasi dan efektivitas dalam mendukung pengelolaan energi yang lebih cerdas. Penggunaan algoritma ini juga memungkinkan integrasi dengan berbagai sensor IoT, yang memberikan kemampuan adaptasi sistem terhadap perubahan lingkungan secara dinamis dan otomatis. [12]

2.2 Landasan Teori

2.2.1 Internet of Things (IoT)

Perkembangan teknologi *Internet of Things* (IoT) dalam bidang kontrol listrik telah memberikan dampak signifikan dalam meningkatkan efisiensi energi, kenyamanan, dan keamanan. Teknologi ini memungkinkan perangkat listrik seperti lampu, pendingin ruangan, dan peralatan elektronik lainnya untuk diatur secara otomatis melalui internet. Salah satu contoh implementasi IoT adalah *smart home*, di mana perangkat-perangkat listrik dapat dipantau dan dikendalikan dari jarak jauh melalui aplikasi, memungkinkan pengaturan yang lebih efisien dan penghematan energi. Contohnya, penggunaan *smart plug* untuk memantau dan memutus aliran listrik perangkat yang tidak digunakan[13], [9].

Selain itu, IoT juga mendukung otomasi sistem, seperti pengaturan pencahayaan dan pendingin ruangan secara otomatis berdasarkan kondisi lingkungan. Hal ini tidak hanya mengurangi pemborosan energi tetapi juga meningkatkan kenyamanan penghuni rumah. Teknologi seperti smart meter juga memainkan peran penting dengan memberikan data penggunaan energi secara real-time, memungkinkan pengguna untuk mengidentifikasi dan mengoptimalkan pola konsumsi listrik mereka. Dengan adanya sistem IoT, pengguna dapat memantau dan mengontrol perangkat listrik dari jarak jauh, bahkan mengatur pengoperasian perangkat berdasarkan kebiasaan atau jadwal tertentu. Selain itu, IoT memungkinkan pengintegrasian berbagai sensor untuk menyesuaikan penggunaan energi dengan kebutuhan nyata, sehingga meningkatkan efisiensi dan mengurangi pemborosan. Teknologi ini juga mendukung konsep rumah pintar yang lebih ramah lingkungan, karena dapat menyesuaikan sistem pengatur suhu, pencahayaan, dan konsumsi energi lainnya secara otomatis untuk mendukung keberlanjutan.

2.2.2 ESP32–CAM

ESP32-CAM adalah mikrokontroler canggih yang telah dilengkapi dengan kamera OV2640, menjadikannya pilihan ideal untuk aplikasi yang membutuhkan pengambilan dan transmisi gambar atau video. Mikrokontroler ini dapat diprogram menggunakan Arduino IDE, sebuah editor yang populer dan mudah digunakan, sehingga mendukung berbagai proyek berbasis IoT. Dalam penerapannya, ESP32-CAM sering dipadukan dengan sensor HC-SR501, sebuah sensor inframerah pasif (PIR) yang mendeteksi pergerakan. Ketika sensor PIR aktif, ESP32-CAM akan mengambil gambar atau video dan mengirimkannya melalui jaringan Wi-Fi ke perangkat smartphone. Kemampuan ini membuat ESP32-CAM sangat berguna dalam sistem keamanan[4], pemantauan jarak jauh, atau aplikasi otomatisasi lainnya, memberikan solusi

yang hemat biaya dengan kinerja yang andal. Dengan demikian, ESP32-CAM memberikan fleksibilitas dalam desain sistem, memungkinkan integrasi mudah dengan berbagai sensor dan platform, serta menyediakan pengiriman data secara real-time, yang sangat penting untuk aplikasi yang memerlukan respons cepat, seperti pengawasan keamanan rumah atau area publik.[10], [14].



Gambar 2. 1 ESP32-CAM

2.2.3 Sensor HC-SR501 (PIR Sensor)

Sensor PIR (Passive Infrared) adalah sensor yang digunakan untuk mendeteksi pergerakan manusia atau hewan berdasarkan perubahan radiasi inframerah di sekitarnya. Sensor ini bekerja dengan mendeteksi perubahan suhu yang dihasilkan oleh tubuh manusia yang berada di dalam jangkauan deteksi. Prinsip kerja PIR bergantung pada fakta bahwa tubuh manusia mengeluarkan radiasi inframerah yang lebih tinggi dibandingkan dengan lingkungan sekitarnya[15].

PIR sensor terdiri dari dua elemen sensor yang saling berpasangan dan bekerja bersama untuk mendeteksi perbedaan radiasi yang datang dari dua sisi. Ketika ada objek yang bergerak di dalam area pengamatan, perbedaan radiasi antara dua elemen sensor ini memicu keluaran sinyal. Hal ini memungkinkan perangkat yang menggunakan sensor PIR untuk mendeteksi gerakan secara efektif.



Gambar 2. 2Sensor Pir

2.2.4 Modul Relay

Modul relay adalah perangkat elektronik yang berfungsi sebagai saklar elektronik untuk mengendalikan sirkuit listrik yang terhubung dengan perangkat lainnya, seperti lampu, kipas, atau perangkat rumah tangga. Modul ini bekerja dengan memanfaatkan prinsip elektromekanis, di mana sinyal listrik kecil dari mikrokontroler atau sistem pengendali seperti Arduino, Raspberry Pi, atau ESP32 dapat digunakan untuk mengontrol perangkat dengan tegangan dan arus yang lebih tinggi [16]. Relay biasanya terdiri dari beberapa komponen utama: kumparan elektromagnetik, kontak saklar, dan terminal input/output. Ketika kumparan mendapatkan sinyal, medan magnet yang dihasilkan akan menggerakkan kontak saklar, membuka atau menutup sirkuit listrik yang terhubung, dapat dilihat pada gambar 2.3 berikut ini:



Gambar 2. 3 Modul Relay

2.2.5 Switching Power Supply 5v

Power supply (pasokan daya) adalah komponen yang sangat penting dalam rangkaian elektronik, karena menyediakan energi listrik yang dibutuhkan untuk mengoperasikan perangkat dan sensor. Dalam sistem berbasis IoT seperti yang melibatkan ESP32-CAM, sensor PIR, dan relay, pasokan daya memastikan bahwa semua komponen berfungsi dengan baik dan stabil. Pasokan daya harus memenuhi persyaratan tegangan dan arus dari setiap perangkat dalam sistem tersebut.

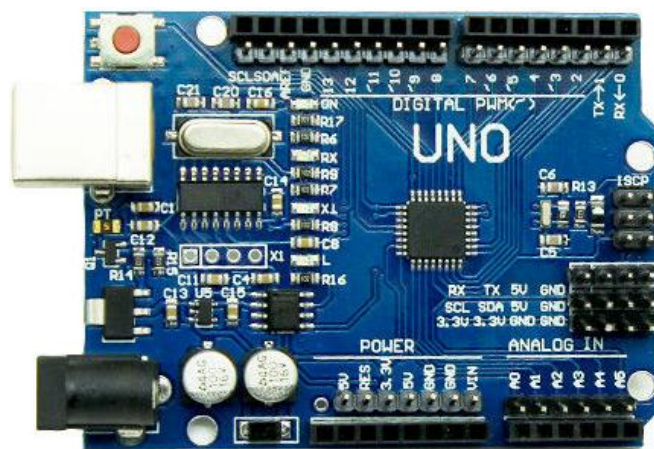
Sebagai contoh, ESP32-CAM biasanya membutuhkan pasokan 5V, yang bisa diperoleh dari port USB atau adaptor daya khusus. Sensor PIR, di sisi lain, biasanya bekerja pada tegangan yang lebih rendah, seperti 3.3V atau 5V, tergantung pada model yang digunakan. Modul relay, yang digunakan untuk mengontrol perangkat listrik, umumnya memerlukan tegangan pasokan terpisah, seperti 5V atau 12V, tergantung pada spesifikasinya. Dapat dilihat pada gambar 2.4 berikut ini:



Gambar 2 1 Switching Power Supply 5v

2.2.6 Arduino Uno

Arduino Uno adalah sebuah papan mikrokontroler yang digunakan untuk membuat dan mengendalikan berbagai proyek elektronik dan sistem otomatis. Arduino Uno menggunakan mikrokontroler ATmega328P sebagai “otak” utama yang berfungsi untuk menjalankan program sesuai instruksi yang diberikan. Arduino Uno mudah digunakan karena dapat diprogram melalui Arduino IDE menggunakan kabel USB dan bahasa pemrograman yang sederhana. Papan ini memiliki pin input/output digital dan analog yang memungkinkan Arduino Uno membaca data dari sensor (seperti sensor suhu atau cahaya) dan mengendalikan perangkat lain seperti LED, motor, relay, atau modul komunikasi. Karena bersifat open-source, murah, dan didukung komunitas yang luas, Arduino Uno sangat cocok untuk pemula, pembelajaran, serta pengembangan prototipe di bidang elektronika, IoT, dan sistem kendali [17].



Gambar 2. 4 Arduino Uno

2.2.7 Algoritma YOLO (*You Only Look Once*)

Algoritma YOLO (*You Only Look Once*) adalah salah satu teknik paling populer dalam bidang *computer vision* untuk deteksi objek secara real-time [18]. Dikembangkan oleh Joseph Redmon dan kawan-kawan, YOLO memanfaatkan pendekatan yang berbeda dari metode deteksi objek tradisional dengan mengubah proses deteksi menjadi masalah regresi tunggal. Alih-alih memproses gambar dalam beberapa langkah terpisah seperti ekstraksi fitur dan klasifikasi, YOLO membagi gambar menjadi grid, lalu memproses setiap bagian untuk mendeteksi objek dan menentukan lokasinya sekaligus dalam satu tahap.[8], Berikut ini merupakan beberapa keunggulan dan kekurangan Algoritma Yolo (*You Only Look Once*) :

1. Keunggulan Algoritma Yolo (*You Only Look Once*)
 - 1) Kecepatan Tinggi: YOLO dirancang untuk deteksi objek secara real-time, bahkan pada perangkat keras dengan sumber daya terbatas.
 - 2) Akurasi Spasial: YOLO memberikan prediksi yang lebih baik dalam hal posisi objek, karena memproses seluruh gambar secara langsung.
 - 3) Kemudahan Implementasi: YOLO banyak digunakan karena tersedia dalam pustaka yang populer, seperti *Darknet*, *TensorFlow*, dan *PyTorch*.
2. Kekurangan Algoritma Yolo (*You Only Look Once*)
 - 1) Kesulitan dalam Deteksi Objek Kecil: Karena pembagian grid, YOLO terkadang sulit mendeteksi objek kecil dalam gambar yang kompleks.
 - 2) *Trade-off* antara Kecepatan dan Akurasi: Meskipun cepat, versi awal YOLO cenderung kurang akurat dibandingkan metode seperti Faster R-CNN pada skenario kompleks.

2.2.8 Arduino IDE

Arduino Integrated Development Environment (IDE) adalah perangkat lunak sumber terbuka yang dirancang untuk memudahkan pengembangan dan pemrograman pada perangkat mikrokontroler Arduino. Platform ini menyediakan antarmuka pengguna yang intuitif, memungkinkan pengguna untuk menulis, mengedit, dan mengunggah kode ke papan Arduino dengan mudah. Arduino IDE mendukung berbagai bahasa pemrograman seperti C dan C++, serta dilengkapi dengan pustaka yang ekstensif untuk berbagai modul dan sensor. Salah satu fitur utama adalah *serial monitor*, yang memungkinkan pengguna memantau data yang dikirimkan oleh perangkat secara *real-time* untuk debugging dan analisis[17]. Pengembangan

terbaru mencakup versi berbasis cloud, yang memungkinkan akses lebih fleksibel ke lingkungan pengembangan dari berbagai perangkat tanpa memerlukan instalasi lokal.

Keberadaan IDE ini mendukung inovasi di berbagai bidang, mulai dari pendidikan STEM hingga aplikasi industri. Penelitian menunjukkan bahwa Arduino IDE telah digunakan dalam prototipe perangkat berbasis *Internet of Things* (IoT), kontrol otomatisasi, dan integrasi sensor untuk pengukuran data lingkungan. Versi terbaru IDE juga menambahkan kompatibilitas dengan MicroPython, yang memperluas fleksibilitas pengembang untuk proyek yang memerlukan pendekatan lebih ringan terhadap pemrograman. Dengan pembaruan yang terus dilakukan, Arduino IDE tetap relevan dalam komunitas open-source dan memainkan peran penting dalam mendorong kemajuan teknologi melalui ekosistem Arduino yang kolaboratif [17].

2.2.9 Telegram

Telegram adalah aplikasi perpesanan instan berbasis cloud yang memungkinkan pengguna mengirim dan menerima pesan teks, gambar, video, serta file lainnya secara real-time melalui jaringan internet. Telegram banyak digunakan dalam pengembangan sistem Internet of Things (IoT) karena menyediakan fitur Telegram Bot API yang memungkinkan sistem atau perangkat otomatis berkomunikasi langsung dengan pengguna tanpa memerlukan interaksi manual.

Dalam konteks sistem IoT, Telegram berfungsi sebagai media notifikasi dan monitoring jarak jauh. Dengan memanfaatkan bot Telegram, sistem dapat mengirimkan informasi secara otomatis kepada pengguna ketika suatu kondisi tertentu terpenuhi. Pada penelitian ini, Telegram digunakan untuk mengirimkan notifikasi deteksi manusia yang dilakukan oleh sistem berbasis ESP32-CAM dan algoritma YOLO [19]. Ketika sistem mendeteksi keberadaan manusia, server YOLO akan memicu pengiriman pesan ke akun Telegram pengguna sebagai pemberitahuan bahwa aktivitas telah terdeteksi dan perangkat listrik (lampu) telah diaktifkan.

Keunggulan penggunaan Telegram dibandingkan media notifikasi lainnya adalah kemudahan implementasi, kecepatan pengiriman pesan, serta dukungan multiplatform yang memungkinkan akses melalui smartphone maupun komputer. Selain itu, Telegram Bot API mendukung pengiriman pesan secara aman menggunakan protokol HTTPS, sehingga komunikasi antara sistem dan pengguna tetap terjaga. Telegram juga tidak memerlukan biaya tambahan seperti layanan SMS, sehingga lebih efisien dan ekonomis untuk diterapkan pada sistem otomasi rumah berbasis IoT [20].

Dengan integrasi Telegram pada sistem ini, pengguna dapat memantau kondisi rumah secara real-time, menerima informasi secara cepat, serta meningkatkan aspek keamanan dan kenyamanan [21]. Oleh karena itu, Telegram menjadi salah satu komponen pendukung penting dalam sistem kontrol peralatan listrik berbasis deteksi manusia yang dikembangkan pada penelitian ini [22], [15].

2.2.10 Waterfall

Model *Waterfall* adalah pendekatan pengembangan perangkat lunak yang sistematis dan berurutan, sering disebut sebagai metode siklus hidup klasik. Setiap tahap dalam model ini harus diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya. Tahapan utama meliputi analisis kebutuhan, desain sistem, implementasi, pengujian, dan pemeliharaan. Pada tahap analisis kebutuhan, tim pengembang bekerja sama dengan pemangku kepentingan untuk mengidentifikasi kebutuhan perangkat lunak secara mendalam. Selanjutnya, hasil analisis diterjemahkan menjadi desain sistem yang mencakup struktur data, arsitektur perangkat lunak, dan antarmuka pengguna. Implementasi dilakukan dengan mengubah desain menjadi kode program, diikuti oleh pengujian yang memastikan perangkat lunak berfungsi sesuai spesifikasi sebelum diimplementasikan dan diserahkan kepada pengguna [23].

Pendekatan ini memiliki beberapa kelebihan, seperti proses kerja yang terstruktur, target yang jelas pada setiap tahap, dan pengendalian proyek yang lebih mudah. Namun, kelemahannya terletak pada kurangnya fleksibilitas, karena perubahan pada tahap awal sulit dilakukan jika proyek sudah berada pada tahap lanjut. *Waterfall* cocok untuk proyek yang persyaratan awalnya sudah jelas dan tidak banyak berubah. Meskipun metode ini dianggap konvensional, *Waterfall* tetap relevan dalam pengembangan perangkat lunak berskala besar atau dengan kebutuhan yang stabil dan terdefinisi dengan baik.

2.2.11 UML (Unified Modelling Language)

Unified Modeling Language (UML) adalah bahasa modeling yang umum dan digunakan untuk mendeskripsikan, membuat, dan dokumen sistem perangkat lunak. Tujuan utama *UML* adalah untuk mendefinisikan cara standar untuk memvisualisasikan bagaimana sistem telah dirancang. *UML* tidak merupakan bahasa pemrograman, melainkan bahasa visual. *UML* digunakan untuk membuat diagram yang mencerminkan perilaku dan struktur dari sistem. Diagram ini membantu insinyur perangkat lunak, pengusaha, dan arsitek sistem dalam proses modeling, desain, dan analisis.

UML dibagi menjadi dua kategori utama: diagram struktural dan diagram perilaku. Diagram struktural menunjukkan entitas dalam sistem dan hubungan antara entitas tersebut. Diagram perilaku menunjukkan interaksi antara objek dalam sistem. *UML* pertama kali dikembangkan pada tahun 1994-1995 oleh *Rational Software*, dan telah dikembangkan lebih lanjut oleh mereka sejak tahun 1996. Pada tahun 1997, *UML* diadopsi sebagai standar oleh *Object Management Group (OMG)*, dan telah dikelola oleh organisasi ini sejak itu. Pada tahun 2005, *UML* juga dipublikasikan oleh *International Organization for Standardization (ISO)* dan *International Electrotechnical Commission (IEC)* sebagai standar ISO/IEC 19501.

UML (Unified Modelling Language) adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasikan objek.

UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram teks-teks pendukung.


UML (Unified Modeling Language) adalah Keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman *berorientansi objek (OO)*. Definisi ini merupakan definisi yang sederhana.



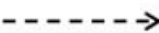
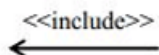
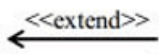
Jadi *UML (Unified Modeling Language)* dapat diartikan sebagai bahasa visual untuk menggambarkan definisi-definisi tentang *requirement*, membuat analisis dan desain serta menggambar arsitektur dalam pemrograman berorientasikan objek dengan menggunakan teks-teks pendukung.

a. Use Case Diagram

Use case diagram adalah satu dari berbagai jenis diagram *UML (Unified Modelling Language)* yang menggambarkan hubungan interaksi antara sistem dan aktor. *Use Case* dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya.

Tabel 2.2 Use Case Diagram

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case.




	Use Case : Abstraksi dan interaksi antara sistem dan aktor.
	Association : Abstraksi dari penghubung antara aktor dengan use case
	Generalisasi : Menunjukkan Spesialisasi aktor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.
	Menunjukkan bahwa suatu use case merupakan tambahan fungsionalitas dari use case lainnya jika suatu kondisi terpenuhi.




b. Activity Diagram

Activity diagram, dalam bahasa Indonesia diagram aktivitas, yaitu diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem. Runtutan proses dari suatu sistem digambarkan secara vertikal. *Activity diagram* merupakan pengembangan dari *Use Case* yang memiliki alur aktivitas.

Alur atau aktivitas berupa bisa berupa runtutan menu-menu atau proses bisnis yang terdapat di dalam sistem tersebut. Dalam buku *Rekayasa Perangkat Lunak* karangan Rosa A.S mengatakan, “Diagram aktivitas tidak menjelaskan kelakuan aktor. Dapat diartikan bahwa dalam pembuatan activity diagram hanya dapat dipakai untuk menggambarkan alur kerja atau aktivitas sistem saja”.

Tabel 2.3 Activity Diagram

Simbol	Nama	Keterangan
	Status Awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan system, aktivitas yang diawali dengan kata kerja.
	Percabangan / Decision	Percabangan Dimana ada pilihan aktifitas yang lebih dari satu.

	Pergabungan / join	Penggabungan Dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Berikut penjelasan lengkapnya mengenai komponen-komponen pada activity diagram di atas :

1. *Start Point* atau *Initial State* (Titik Mulai/Status Awal) *Start Point* adalah lingkaran hitam kecil. Biasanya digunakan untuk menandakan status awal, tindakan awal, atau titik awal aktivitas untuk setiap *activity diagram*.
2. *Activity* (Aktivitas)
Activity merupakan aktivitas yang dilakukan atau sedang terjadi dalam sistem. Biasanya diawali dengan “kata kerja” dari aktivitas yang dilakukan.
3. *Decision* atau Percabangan
Percabangan atau *decision* merupakan suatu titik atau point yang mengindikasikan suatu kondisi di mana adanya kemungkinan dalam perbedaan transisi. Hal tersebut diperlukan ketika sistem yang dimiliki memiliki beberapa kemungkinan atau jalan alternatif.
4. *Synchronization*
Synchronization dibagi menjadi 2 bagian, yaitu fork dan join.
 - a. *Fork* (percabangan) digunakan untuk memecah *behaviour* (tingkah laku) menjadi *activity* atau *action* (aksi) secara paralel.
 - b. *Join* (penggabungan) digunakan untuk menghubungkan kembali *activity* dengan *action* secara paralel.
5. *Merge*
Menggabungkan *flow* yang sudah dipecah menjadi beberapa bagian oleh suatu *flow*.

6. *Swimlanes*

Memecah *activity diagram* menjadi kolom dan baris untuk membagi tanggung jawab objek-objek yang melakukan suatu aktivitas.

7. *Transition*

Digunakan untuk menunjukkan aktivitas selanjutnya dan sebelumnya.

8. Notasi akhir (*end state*)

Notasi akhir digunakan untuk menandakan proses tersebut berakhir. Pada *UML*, notasi akhir dapat digambarkan dengan simbol sebuah *bull's eye* (mata sapi).

c. Class Diagram

Class diagram atau diagram kelas adalah salah satu jenis diagram struktur pada *UML* yang menggambarkan dengan jelas struktur serta deskripsi class, atribut, metode, dan hubungan dari setiap objek. Ia bersifat statis, dalam artian diagram kelas bukan menjelaskan apa yang terjadi jika kelas-kelasnya berhubungan, melainkan menjelaskan hubungan apa yang terjadi.

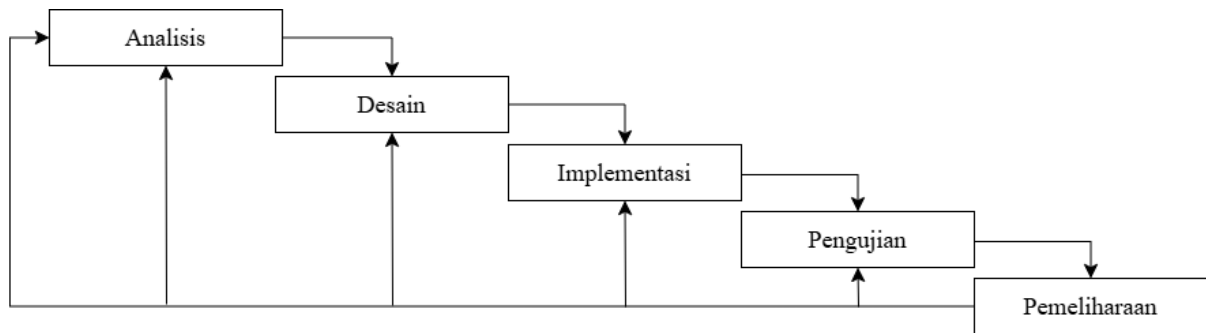
Desain model dari diagram kelas ini sendiri dibagi menjadi dua bagian. Bagian pertama merupakan penjabaran dari database. Bagian kedua merupakan bagian dari modul MVC, yang memiliki *class interface*, *class control*, dan *class entity*.

BAB III

METODOLOGI PENELITIAN

3.1 Metode

Metode penelitian yang digunakan dalam penelitian ini adalah metode penelitian dan pengembangan *Waterfall*. Model pengembangan sistem informasi akademik berbasis web ini mengadopsi model Waterfall yang dikemukakan oleh Pressman (2020). Model ini merupakan pendekatan sistematis yang terdiri dari enam tahapan utama untuk memastikan perangkat lunak yang dikembangkan sesuai dengan kebutuhan pengguna dan memiliki kualitas tinggi. Tahapan dalam model Waterfall berdasarkan Pressman (2020) yaitu Analisis, Desain, Implementasi, Pengujian, dan Pemeliharaan [23]. Kelima tahapan tersebut dapat divisualisasikan sebagai berikut:



Gambar 3 1 Metode Waterfall

3.2 Metode Pengumpulan Data

Dalam penelitian ini, pengumpulan data dilakukan untuk mendukung proses perancangan, implementasi, dan evaluasi sistem deteksi manusia berbasis *Internet of Things* (IoT) menggunakan ESP32-CAM dan algoritma YOLO. Metode pengumpulan data dibagi menjadi dua jenis, yaitu data primer dan data sekunder.

a. Data Primer

Data primer diperoleh langsung melalui proses eksperimen dan pengujian sistem yang telah dirancang. ESP32-CAM digunakan sebagai perangkat utama untuk menangkap gambar, yang kemudian dikirimkan ke server atau perangkat pemroses yang menjalankan algoritma YOLO. Sistem ini diuji secara langsung di lingkungan rumah tinggal untuk mengamati performa dalam mendeteksi kehadiran manusia dan mengendalikan perangkat elektronik (seperti lampu) secara otomatis melalui modul relay.

Data primer yang dikumpulkan meliputi:

- 1) Jumlah keberhasilan dan kegagalan deteksi manusia (True Positive, False Negative, dan False Positive).
- 2) Waktu respon sistem dari proses pendeteksian hingga aktivasi output (lampu).
- 3) Dokumentasi berupa foto dan video hasil uji coba.
- 4) Catatan log dari ESP32 dan/atau server YOLO untuk memantau kestabilan dan akurasi sistem.

Pengujian dilakukan di beberapa kondisi pencahayaan, sudut pandang kamera, dan jarak deteksi untuk memastikan sistem berjalan efektif dalam berbagai skenario.

b. Data Sekunder

Data sekunder diperoleh melalui studi literatur yang berkaitan dengan *Internet of Things*, ESP32–CAM, algoritma YOLO, dan penerapan sistem otomatisasi rumah. Literatur ini digunakan sebagai dasar teori dan pembandingan untuk memperkuat validitas sistem yang dirancang. Sumber data sekunder diambil dari jurnal ilmiah yang terindeks dan terbit dalam lima tahun terakhir, antara lain:

1. Firdaus et al. (2025) membandingkan performa YOLOv5 Nano dan FOMO pada ESP32–CAM, dengan fokus pada akurasi deteksi, waktu inferensi, dan efisiensi memori. Hasilnya menunjukkan YOLOv5 Nano unggul dalam akurasi namun memerlukan waktu inferensi lebih lama.
2. Tamsir et al. (2021) menerapkan YOLOv4 pada ESP32–CAM dengan metode pre-processing gambar untuk sistem keamanan rumah, dan menunjukkan precision mencapai 0.98.
3. Setiarso et al. (2023) mengembangkan sistem pendeteksian manusia berbasis IoT dengan YOLO untuk mengurangi false alarm pada sistem monitoring.
4. Saputra & Sulistyono (2024) mengembangkan sistem keamanan rumah menggunakan ESP32–CAM dan deteksi wajah dengan notifikasi real-time via Telegram.
5. Nugraha Pratama (2025) menerapkan YOLOv5 pada sistem presensi wajah real-time berbasis ESP32–CAM dan web server. Sistem ini mencapai akurasi 90,65% dengan precision 68% dan recall 87,1%.

Data sekunder ini membantu membentuk dasar teoretis, mendukung pemilihan komponen sistem, serta menjadi acuan dalam membandingkan performa sistem yang dikembangkan dengan penelitian serupa.

3.3 Perancangan Sistem

Perancangan sistem dilakukan untuk menggambarkan bagaimana sistem bekerja secara keseluruhan, baik dari aspek perangkat keras (hardware) maupun perangkat lunak (software). Sistem ini dirancang untuk mendeteksi keberadaan manusia menggunakan kamera ESP32-CAM yang terhubung dengan server YOLO sebagai pengolah citra, kemudian memberikan perintah untuk menyalakan atau mematikan perangkat rumah (lampu) melalui modul relay.

3.3.1 Perancangan Hardware

Perancangan hardware pada penelitian ini menggunakan ESP32-CAM sebagai perangkat utama untuk mendeteksi keberadaan manusia berbasis kamera yang terhubung dengan sensor PIR sebagai pendeteksi awal gerakan. Ketika sensor PIR mendeteksi adanya pergerakan, ESP32-CAM akan aktif untuk mengambil gambar dan melakukan proses deteksi manusia menggunakan algoritma YOLO melalui server. Apabila manusia terdeteksi, ESP32-CAM akan mengirimkan sinyal trigger ke Arduino Uno yang berfungsi sebagai pengendali aktuator. Arduino Uno kemudian mengaktifkan modul relay yang berperan sebagai saklar elektronik untuk mengontrol lampu sebagai beban listrik. Selain mengendalikan lampu, sistem juga terintegrasi dengan Telegram untuk mengirimkan notifikasi kepada pengguna sebagai informasi bahwa sistem telah mendeteksi keberadaan manusia dan perangkat listrik telah diaktifkan secara otomatis [19].

a) ESP32-CAM

ESP32-CAM berfungsi sebagai perangkat utama dalam sistem yang digunakan untuk mengambil gambar dari area yang dipantau. Modul ini juga bertugas mengirimkan gambar ke server untuk diproses menggunakan algoritma YOLO serta mengirimkan sinyal trigger ke Arduino ketika manusia terdeteksi.

b) Sensor PIR (Passive Infrared Sensor)

Sensor PIR digunakan untuk mendeteksi adanya gerakan atau perubahan radiasi inframerah yang dihasilkan oleh tubuh manusia. Ketika sensor mendeteksi gerakan, sensor akan mengirimkan sinyal ke ESP32-CAM sebagai tanda bahwa terdapat aktivitas yang perlu diproses oleh sistem.

c) Arduino Uno

Arduino Uno berfungsi sebagai mikrokontroler yang menerima sinyal dari ESP32-CAM untuk mengontrol modul relay. Arduino memproses sinyal tersebut dan

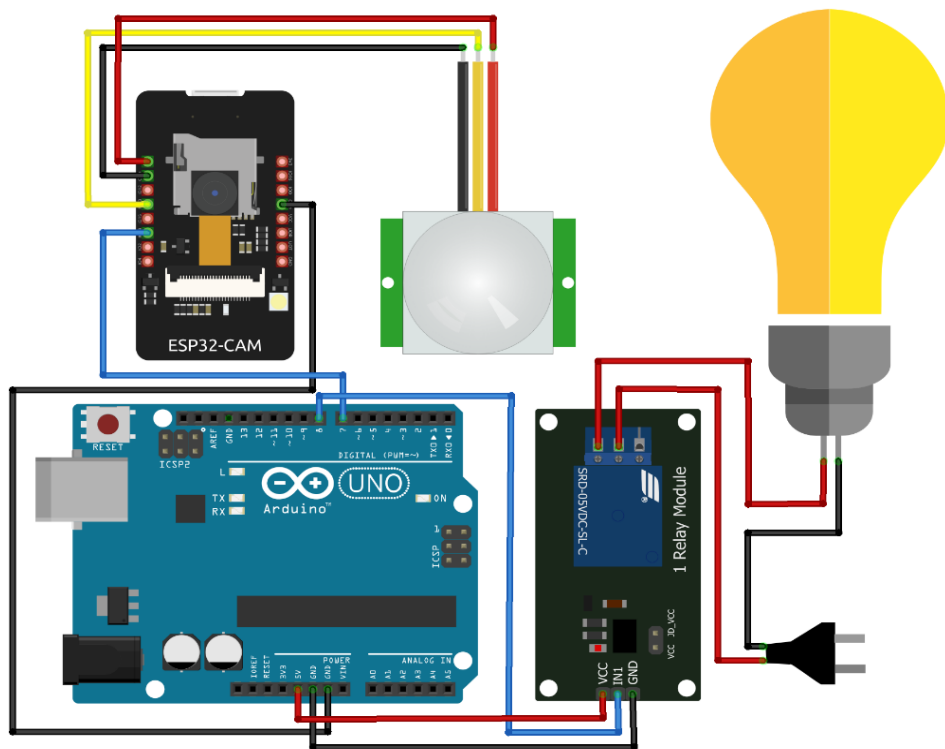
kemudian mengaktifkan atau menonaktifkan relay sesuai dengan kondisi yang diterima dari ESP32-CAM.

d) Modul Relay

Relay berfungsi sebagai saklar elektronik yang digunakan untuk menghubungkan atau memutuskan aliran listrik pada lampu. Relay akan aktif ketika menerima sinyal dari Arduino sehingga lampu dapat menyala secara otomatis.

e) Lampu

Lampu merupakan perangkat output pada sistem yang digunakan sebagai indikator sekaligus penerangan ruangan. Lampu akan menyala secara otomatis ketika sistem mendeteksi keberadaan manusia melalui proses deteksi yang dilakukan oleh ESP32-CAM dan relay.



Gambar 3 2 Wiring Rancangan Alat

Perancangan Hardware Sistem

a) ESP32-CAM

ESP32-CAM digunakan sebagai perangkat utama untuk mendeteksi keberadaan manusia berbasis kamera dan berfungsi sebagai pengendali awal sistem yang menerima sinyal dari sensor PIR serta mengirimkan perintah ke Arduino Uno dan notifikasi ke Telegram

b) Sensor PIR (HC-SR501)

Sensor PIR berfungsi sebagai pendeteksi awal gerakan manusia yang terhubung langsung ke ESP32-CAM dan akan memicu sistem untuk mulai melakukan pengambilan gambar ketika terdeteksi adanya pergerakan

c) Arduino Uno

Arduino Uno digunakan sebagai pengendali aktuator yang menerima sinyal trigger dari ESP32-CAM dan bertugas mengontrol modul relay agar dapat mengaktifkan atau menonaktifkan perangkat listrik

d) Modul Relay

Modul relay berfungsi sebagai saklar elektronik yang dikendalikan oleh Arduino Uno untuk menghubungkan atau memutus aliran listrik ke lampu sesuai perintah dari sistem

e) Lampu (Beban Listrik)

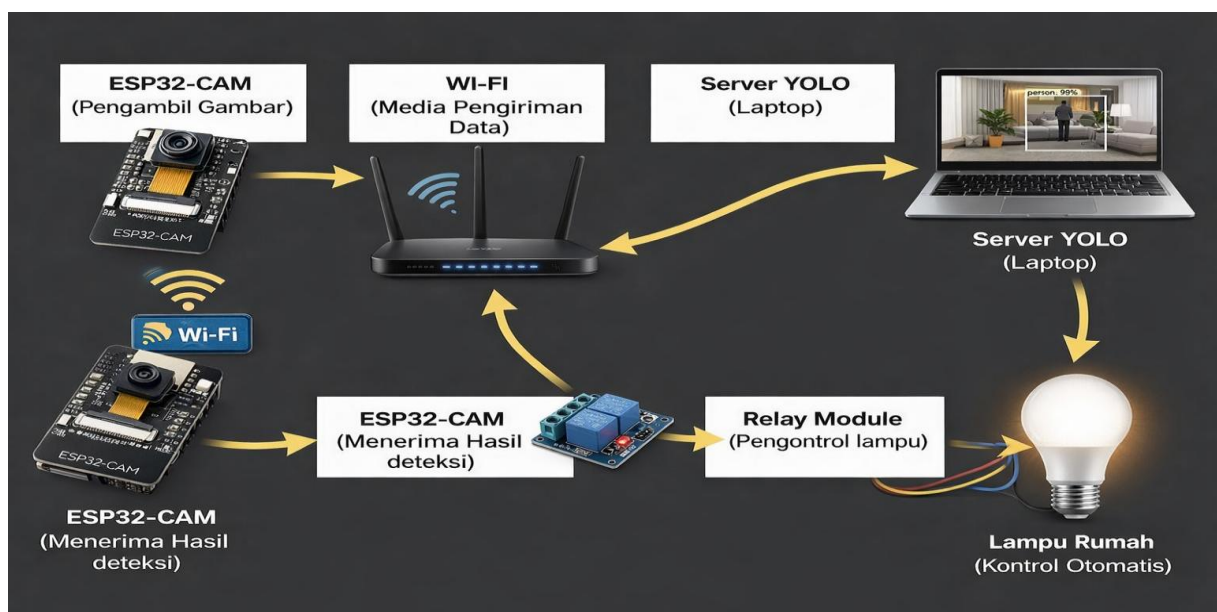
Lampu digunakan sebagai beban listrik yang akan menyala secara otomatis ketika relay diaktifkan sebagai hasil dari deteksi keberadaan manusia

f) Catu Daya (Power Supply)

Catu daya digunakan untuk menyuplai tegangan ke seluruh komponen hardware agar sistem dapat bekerja secara stabil dan berkelanjutan.

3.3.2 Diagram Perancangan

Berikut ini merupakan rancang diagram sistemnya sebagai berikut :



Gambar 3 3 Diagram Perancangan Sistem

Diagram blok sistem menggambarkan alur kerja utama sistem deteksi manusia berbasis IoT. Berikut adalah blok sistem yang dirancang:

- a) ESP32–CAM sebagai perangkat pengambil gambar.
- b) Wi-Fi sebagai media pengiriman gambar ke server YOLO.
- c) Server YOLO (misalnya laptop) sebagai pemroses citra untuk mendeteksi manusia.
- d) ESP32–CAM menerima hasil deteksi dan mengaktifkan relay module.
- e) Relay akan mengontrol lampu rumah secara otomatis berdasarkan hasil deteksi.

3.3.2 Perancangan Software

Sistem deteksi manusia berbasis IoT dirancang untuk mengotomatiskan pengendalian perangkat listrik di dalam ruangan, seperti lampu, berdasarkan keberadaan manusia. Teknologi ini menggabungkan kamera ESP32-CAM, algoritma YOLO (*You Only Look Once*) untuk deteksi objek, dan *web interface* untuk memantau aktivitas secara *real-time* [24].

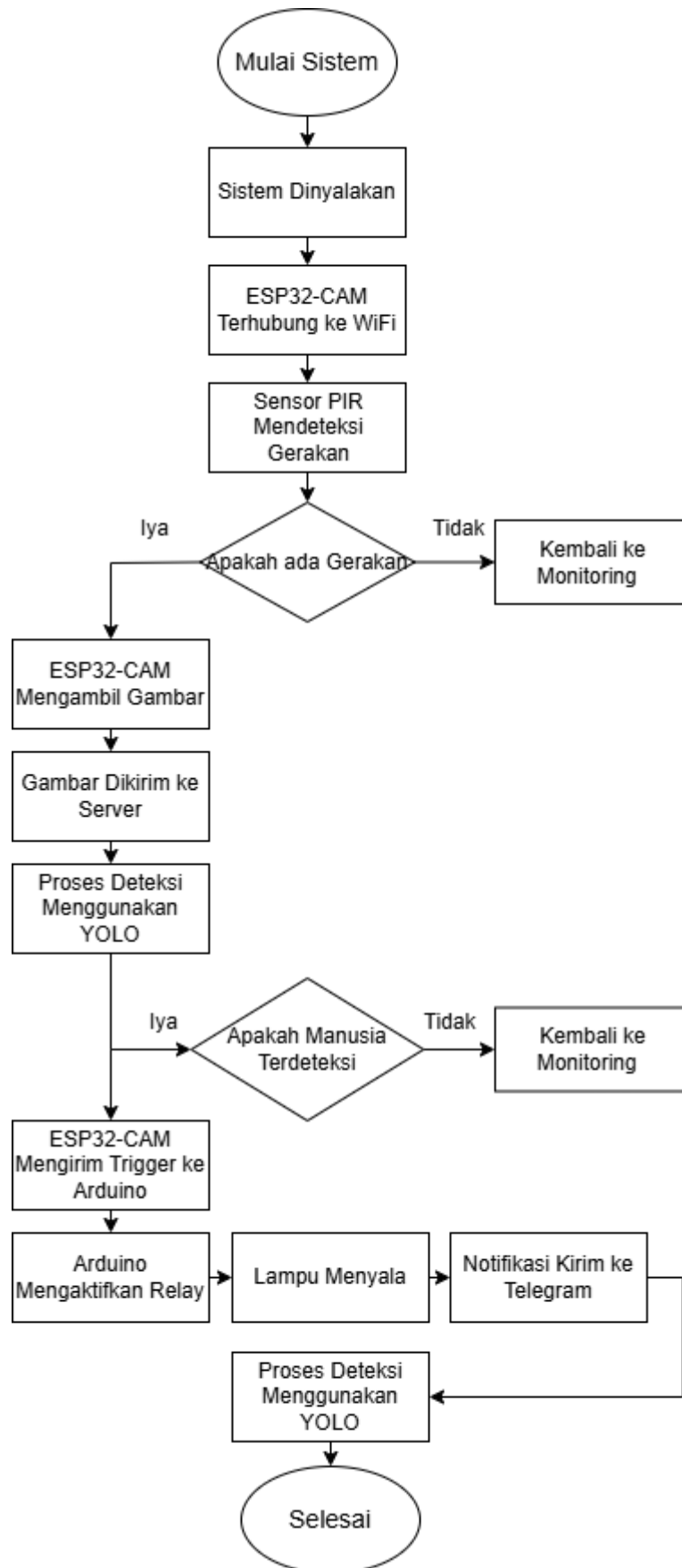
Dengan pendekatan ini, sistem mampu:

- a) Mengambil gambar secara langsung dari lingkungan menggunakan ESP32-CAM.
- b) Mengirim gambar ke server yang menjalankan YOLO untuk mengenali keberadaan manusia.
- c) Mengendalikan perangkat elektronik (seperti lampu) secara otomatis melalui modul relay.
- d) Menyimpan data deteksi ke dalam database dan menampilkannya melalui antarmuka web.

Tujuan utama dari sistem ini adalah untuk:

- a) Menghemat energi listrik dengan mengaktifkan perangkat hanya saat dibutuhkan.
- b) Meningkatkan efisiensi pemantauan ruangan dengan pelaporan berbasis data secara otomatis.

Dengan kombinasi komponen IoT dan kecerdasan buatan (AI), sistem ini menjadi solusi cerdas dalam otomasi rumah tinggal maupun ruang publik.



Gambar 3.4. Activity Diagram Sistem

Activity diagram pada sistem ini menggambarkan alur kerja sistem dalam mendeteksi keberadaan manusia dan mengontrol lampu secara otomatis. Proses dimulai ketika sistem diaktifkan dan seluruh perangkat mendapatkan sumber daya listrik. Setelah itu, ESP32-CAM akan melakukan koneksi ke jaringan WiFi agar dapat berkomunikasi dengan server dan sistem notifikasi. Selanjutnya, sensor PIR mulai melakukan pemantauan terhadap lingkungan sekitar untuk mendeteksi adanya pergerakan di dalam ruangan.

Apabila sensor PIR tidak mendeteksi adanya gerakan, maka sistem akan tetap berada pada kondisi pemantauan (monitoring) dan terus melakukan pengecekan secara berulang. Namun, ketika sensor PIR mendeteksi adanya gerakan, ESP32-CAM akan mengambil gambar dari area yang dipantau menggunakan kamera yang terpasang pada modul tersebut. Gambar yang telah diambil kemudian dikirimkan ke server untuk diproses menggunakan algoritma YOLO yang berfungsi untuk melakukan deteksi objek, khususnya untuk mengidentifikasi keberadaan manusia.

Setelah proses deteksi dilakukan, sistem akan mengevaluasi hasil dari algoritma YOLO. Jika pada gambar tidak terdeteksi manusia, maka sistem akan kembali ke kondisi pemantauan seperti semula. Namun apabila manusia terdeteksi, maka ESP32-CAM akan mengirimkan sinyal atau trigger ke Arduino Uno yang terhubung dengan modul relay. Relay kemudian akan aktif sehingga lampu yang terhubung dengan relay dapat menyala secara otomatis.

Selain mengaktifkan lampu, sistem juga akan mengirimkan notifikasi ke aplikasi Telegram sebagai informasi kepada pengguna bahwa sistem telah mendeteksi keberadaan manusia pada area yang dipantau. Setelah seluruh proses tersebut selesai, sistem akan kembali ke kondisi pemantauan untuk mendeteksi aktivitas berikutnya sehingga sistem dapat bekerja secara berkelanjutan dan otomatis.

Catatan Tambahan:

1. Sistem ini berjalan berulang (*looping*) untuk memantau lingkungan secara *real-time*.
2. Penggunaan YOLO memungkinkan deteksi yang cepat dan akurat meskipun hanya dengan satu gambar.
3. ESP32-CAM tidak memproses gambar, hanya mengirimkannya—komputasi berat dilakukan di server.

3.4 Tahapan Pengembangan Sistem

Tahapan pengembangan sistem dalam penelitian ini mengikuti metode *Waterfall*, yang terdiri dari beberapa tahapan berurutan dan sistematis. Setiap tahapan diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya. Berikut adalah tahapan-tahapan dalam pengembangan sistem:

3.4.1 Analisis Kebutuhan

Tahap ini dilakukan untuk mengidentifikasi kebutuhan sistem yang akan dibangun. Kebutuhan dibagi menjadi dua bagian:

- a) Kebutuhan Perangkat Keras: ESP32–CAM, FTDI USB to TTL, *relay module*, lampu, *power supply*, kabel jumper, dan breadboard.
- b) Kebutuhan Perangkat Lunak: Arduino IDE, Python, YOLOv5/YOLOv4-*tiny*, *Jupyter Notebook/Google Colab*, dan koneksi Wi-Fi.

Tujuan utama sistem adalah mendeteksi keberadaan manusia secara real-time dan mengendalikan perangkat rumah (lampu) secara otomatis.

3.4.2 Perancangan Sistem

Pada tahap ini dilakukan pembuatan desain sistem yang meliputi:

- a) Diagram blok sistem.
- b) Perancangan alur kerja sistem (*flowchart*).
- c) Perancangan rangkaian elektronik ESP32–CAM dan *relay*.
- d) Perancangan komunikasi antara ESP32–CAM dan server pendeteksi YOLO.

3.4.3 Perancangan Program Sistem

Perancangan program sistem dilakukan untuk mengatur alur kerja perangkat agar dapat bekerja sesuai dengan fungsi yang diinginkan. Program dibuat menggunakan Arduino IDE yang berfungsi untuk mengontrol ESP32-CAM, sensor PIR, Arduino Uno, serta modul relay. Dalam program tersebut diatur proses pendeteksian gerakan oleh sensor PIR, pengambilan gambar oleh ESP32-CAM, pengiriman data ke server untuk proses deteksi menggunakan algoritma YOLO, serta pengiriman sinyal ke Arduino untuk mengaktifkan relay. Selain itu, sistem juga diprogram untuk mengirimkan notifikasi ke Telegram ketika manusia terdeteksi sehingga pengguna dapat mengetahui kondisi ruangan secara langsung.

a) Soft Code ESP32-CAM dan Sensor Pir

Berikut ini merupakan *soft code* dari ESP32-CAM dan sensor sebagai berikut:

```
#include <WiFi.h>
#include <WiFiClientSecure.h>

/* ===== TELEGRAM ===== */
#include <UniversalTelegramBot.h>
#include <ArduinoJson.h>

/* ===== ESP32-CAM ===== */
#include "esp_camera.h"
#include "esp_http_server.h"

/* ===== WIFI CONFIG ===== */
const char* ssid = "Tiga ABDUL";
const char* password = "aYAhIBU7";

/* ===== TELEGRAM CONFIG ===== */
#define BOT_TOKEN
"8292899895:AAFx4h8Ju8vy37YtQsCvGPt1btTzDx0sMNQ"
#define CHAT_ID "5997301391"

WiFiClientSecure client;
UniversalTelegramBot bot(BOT_TOKEN, client);

/* ===== PIN CONFIG ===== */
#define PIR_PIN 13
#define TRIGGER_PIN 14

bool motionState = false;

/* ===== CAMERA PIN (AI THINKER) ===== */
#define PWDN_GPIO_NUM 32
#define RESET_GPIO_NUM -1
#define XCLK_GPIO_NUM 0
#define SIOD_GPIO_NUM 26
#define SIOC_GPIO_NUM 27

#define Y9_GPIO_NUM 35
#define Y8_GPIO_NUM 34
#define Y7_GPIO_NUM 39
#define Y6_GPIO_NUM 36
#define Y5_GPIO_NUM 21
#define Y4_GPIO_NUM 19
#define Y3_GPIO_NUM 18
#define Y2_GPIO_NUM 5
```

```

#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM    22

/* ===== STREAM SERVER ===== */
httpd_handle_t stream_httpd = NULL;

/* ===== STREAM HANDLER ===== */
esp_err_t stream_handler(httpd_req_t *req) {
    camera_fb_t *fb;
    char part_buf[64];

    httpd_resp_set_type(req, "multipart/x-mixed-
replace;boundary=frame");

    while (true) {
        fb = esp_camera_fb_get();
        if (!fb) continue;

        httpd_resp_send_chunk(req, "\r\n--frame\r\n",
strlen("\r\n--frame\r\n"));
        sprintf(part_buf, sizeof(part_buf),
                "Content-Type: image/jpeg\r\nContent-Length:
%u\r\n\r\n",
                fb->len);
        httpd_resp_send_chunk(req, part_buf,
strlen(part_buf));
        httpd_resp_send_chunk(req, (char *)fb->buf, fb->len);

        esp_camera_fb_return(fb);
        vTaskDelay(30 / portTICK_PERIOD_MS);
    }
}

/* ===== TELEGRAM MESSAGE ===== */
void sendTelegramNotification() {
    String message;
    message = "👤 Gerakan terdeteksi\n";
    message += "📺 Silakan buka live stream:\n";
    message += "http://";
    message += WiFi.localIP().toString();
    message += "/stream";

    bot.sendMessage(CHAT_ID, message, "");
}

/* ===== SETUP ===== */
void setup() {

```

```

Serial.begin(115200);
Serial.println();

pinMode(PIR_PIN, INPUT);
pinMode(TRIGGER_PIN, OUTPUT);
digitalWrite(TRIGGER_PIN, LOW);

/* ===== CAMERA CONFIG ===== */
camera_config_t config;
config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sccb_sda = SIOD_GPIO_NUM;
config.pin_sccb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
config.frame_size = FRAMESIZE_VGA; // bisa QVGA jika
mau lebih ringan
config.jpeg_quality = 12;
config.fb_count = 1;

if (esp_camera_init(&config) != ESP_OK) {
    Serial.println("✘ Camera Init Failed");
    return;
}

/* ===== WIFI CONNECT ===== */
WiFi.begin(ssid, password);
Serial.print("Connecting WiFi");
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}

Serial.println("\n☑ WiFi Connected");

```

```

Serial.print("IP Address: ");
Serial.println(WiFi.localIP());

client.setInsecure();

/* ===== STREAM SERVER ===== */
httpd_config_t http_config = HTTPD_DEFAULT_CONFIG();
httpd_start(&stream_httpd, &http_config);

httpd_uri_t stream_uri = {
    .uri      = "/stream",
    .method   = HTTP_GET,
    .handler  = stream_handler,
    .user_ctx = NULL
};
httpd_register_uri_handler(stream_httpd, &stream_uri);

Serial.println("📡 Streaming Ready!");
}

/* ===== LOOP ===== */
void loop() {
    int motion = digitalRead(PIR_PIN);

    if (motion == HIGH && motionState == false) {
        Serial.println("👤 Gerakan Terdeteksi");

        sendTelegramNotification();

        digitalWrite(TRIGGER_PIN, HIGH); // trigger ke
        Arduino
        delay(5000);
        digitalWrite(TRIGGER_PIN, LOW);

        motionState = true;
    }

    if (motion == LOW) {
        motionState = false;
    }

    delay(200);
}

```

(Disclaimer!! : Sumber Soft Code ini di ambil dari Chat GPT.)

b) Soft Code Arduino Uno dan Relay

Berikut ini merupakan codingan arduino uno dan relay sebagai berikut:

```
#define PIR_PIN 2
#define RELAY_PIN 8
#define ESP_TRIGGER 7

unsigned long lastMotionTime = 0;
const unsigned long delayOff = 60000; // 1 menit (60000 ms)

bool lampState = false;

void setup() {
  pinMode(PIR_PIN, INPUT);
  pinMode(RELAY_PIN, OUTPUT);
  pinMode(ESP_TRIGGER, OUTPUT);

  // Relay OFF (karena active LOW)
  digitalWrite(RELAY_PIN, HIGH);
  digitalWrite(ESP_TRIGGER, LOW);

  Serial.begin(9600);
}

void loop() {

  int motion = digitalRead(PIR_PIN);

  // Jika ada gerakan
  if (motion == HIGH) {
    lastMotionTime = millis();

    if (!lampState) {
      lampState = true;
      digitalWrite(RELAY_PIN, LOW); // ON
      digitalWrite(ESP_TRIGGER, HIGH); // Kirim trigger ke ESP32
      Serial.println("Gerakan terdeteksi - Lampu ON");
    }
  }

  // Jika sudah lewat waktu tanpa gerakan
  if (lampState && (millis() - lastMotionTime > delayOff)) {
    lampState = false;
  }
}
```

```

digitalWrite(RELAY_PIN, HIGH); // OFF
digitalWrite(ESP_TRIGGER, LOW);
Serial.println("Tidak ada gerakan - Lampu OFF");
}
}

```

(Disklaimer!! : Sumber Soft Code ini di ambil dari Chat GPT.)

c) Soft Code Yolo

Berikut ini merupakan codingan menjalankan yolo menggunakan IDLE (*Python*) seperti berikut :

```

import urllib.request
import cv2
import numpy as np
import time
# =====
# URL ESP32-CAM
# =====
url = "http://192.168.69.25/capture"

# LOAD YOLO (LEBIH RINGAN)

net = cv2.dnn.readNet("yolov3-tiny.weights", "yolov3-tiny.cfg")
net.setPreferableBackend(cv2.dnn.DNN_BACKEND_OPENCV)
net.setPreferableTarget(cv2.dnn.DNN_TARGET_CPU)

with open("coco.names", "r") as f:
    classes = [line.strip() for line in f.readlines()]

layer_names = net.getLayerNames()
output_layers = [layer_names[i - 1] for i in
net.getUnconnectedOutLayers()]

# PARAMETER OPTIMASI
CONFIDENCE_THRESHOLD = 0.35
NMS_THRESHOLD = 0.4
YOLO_SIZE = 320
REQUEST_DELAY = 0.1
print("[INFO] YOLO ESP32-CAM STARTED")
while True:
    try:

```

```

# AMBIL GAMBAR

img_resp = urllib.request.urlopen(url, timeout=5)
img_np = np.array(bytearray(img_resp.read()), dtype=np.uint8)
frame = cv2.imdecode(img_np, cv2.IMREAD_COLOR)

if frame is None:
    print("Frame kosong")
    time.sleep(0.1)
    continue

height, width, _ = frame.shape
# YOLO PROCESS

blob = cv2.dnn.blobFromImage(
frame, 1/255.0, (416, 416), swapRB=True, crop=False
)
net.setInput(blob)
outputs = net.forward(output_layers)
boxes, confidences, class_ids = [], [], []
for output in outputs:
    for detection in output:
        scores = detection[5:]
        class_id = np.argmax(scores)
        confidence = scores[class_id]

        if confidence > CONFIDENCE_THRESHOLD:
            center_x = int(detection[0] * width)
            center_y = int(detection[1] * height)
            w = int(detection[2] * width)
            h = int(detection[3] * height)
            x = int(center_x - w / 2)
            y = int(center_y - h / 2)

            boxes.append([x, y, w, h])
            confidences.append(float(confidence))
            class_ids.append(class_id)

indexes = cv2.dnn.NMSBoxes(
    boxes, confidences,
    CONFIDENCE_THRESHOLD, NMS_THRESHOLD
)

```

```

if len(indexes) > 0:
    for i in indexes.flatten():
        if classes[class_ids[i]] == "person":
            x, y, w, h = boxes[i]
            conf = confidences[i]
            cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)
            cv2.putText(
                frame,
                f"person {conf:.2f}",
                (x, y - 10),
                cv2.FONT_HERSHEY_SIMPLEX,
                0.6,
                (0, 255, 0),
                2
            )

cv2.imshow("YOLO ESP32-CAM (Optimized)", frame)

if cv2.waitKey(1) == 27:
    break
time.sleep(REQUEST_DELAY)

except Exception as e:
    print("Error:", e)
    time.sleep(5)

cv2.destroyAllWindows()
(Disklaimer!! : Sumber Soft Code ini di ambil dari Chat GPT.)

```

3.4.4 Implementasi Sistem

Pada tahap implementasi, seluruh perangkat keras dan perangkat lunak yang telah dirancang mulai dirakit dan diprogram. ESP32-CAM diprogram menggunakan Arduino IDE untuk menangkap gambar dan mengirimkannya ke server. Server kemudian menjalankan model YOLO untuk mendeteksi manusia. Jika manusia terdeteksi, sistem akan mengaktifkan relay untuk menyalakan lampu.

3.4.5 Pengujian Sistem

Sistem diuji dengan beberapa skenario berbeda, seperti:

- a) Kondisi pencahayaan terang dan redup.

- b) Jarak dan sudut pandang kamera terhadap objek.
- c) Deteksi satu atau beberapa orang.

Pengujian dilakukan untuk memastikan sistem berjalan sesuai dengan kebutuhan dan dapat merespon secara otomatis serta akurat.

3.4.6 Pemeliharaan dan Evaluasi

Tahap ini dilakukan untuk melakukan perbaikan atau penyesuaian jika terdapat kesalahan dalam sistem atau untuk meningkatkan kinerja sistem. Evaluasi dilakukan dengan membandingkan hasil pengujian terhadap target yang diinginkan.

3.5 Teknik Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui seberapa baik sistem bekerja dalam mendeteksi keberadaan manusia dan mengontrol perangkat rumah tinggal secara otomatis. Teknik pengujian dilakukan secara fungsional dan evaluatif dengan pendekatan berikut:

3.5.1 Uji Fungsionalitas

Dilakukan untuk memastikan bahwa setiap komponen sistem berfungsi sebagaimana mestinya:

- a) ESP32–CAM mampu menangkap dan mengirim gambar.
- b) Server YOLO mampu mendeteksi manusia secara akurat.
- c) ESP32–CAM mampu menerima sinyal dari server dan mengontrol relay.
- d) Relay berfungsi menyalakan dan mematikan lampu sesuai perintah.

3.5.2 Uji Akurasi Deteksi

Dilakukan dengan mencatat jumlah deteksi benar (*true positive*), deteksi salah (*false positive*), dan kegagalan deteksi (*false negative*). Pengujian dilakukan dalam beberapa kondisi:

- a) Siang dan malam hari.
- b) Jarak 1–5 meter dari kamera.
- c) Objek bergerak dan diam.

3.5.3 Pengujian Waktu Respon

Pengujian ini dilakukan untuk mengetahui selang waktu dari saat manusia terdeteksi hingga lampu menyala. Waktu respon yang baik akan menunjukkan efisiensi sistem secara real-time.

3.5.4 Dokumentasi Hasil

Semua hasil pengujian akan didokumentasikan dalam bentuk tabel, grafik, dan foto untuk dianalisis lebih lanjut dan menjadi bahan evaluasi sistem.

3.6 Alat dan Bahan Penelitian

Dalam pelaksanaan penelitian ini, dibutuhkan beberapa alat dan bahan pendukung yang terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*). Pemilihan alat dan bahan disesuaikan dengan kebutuhan sistem serta kompatibilitas antar komponen agar sistem dapat berjalan secara optimal.

3.6.1 Alat dan Bahan Perangkat Keras

Tabel 3.1 Alat dan Bahan Perangkat Keras

No	Komponen	Fungsi
1.	ESP32-CAM	Modul mikrokontroler dengan kamera bawaan, digunakan untuk menangkap gambar dan mengirimkannya ke server.
2.	FTDI USB to TTL	Untuk menghubungkan ESP32-CAM ke komputer/laptop saat proses pemrograman.
3.	Relay Module (1-Channel)	Sebagai saklar elektronik untuk mengontrol lampu berdasarkan hasil deteksi.
4.	Lampu 220V	Sebagai indikator atau beban sistem rumah tinggal yang dikendalikan secara otomatis.
5.	Power Supply 5V	Sumber daya untuk ESP32-CAM dan modul lainnya. Bisa berupa adaptor atau power bank.
6.	Kabel Jumper	Menghubungkan pin-pin antar komponen di breadboard atau PCB.
7.	Breadboard / PCB	Tempat merakit sementara atau permanen rangkaian elektronik.
8.	Laptop / Komputer	Digunakan untuk pemrograman ESP32-CAM serta sebagai server YOLO (pendeteksi objek).
9.	Arduino Uno	Sebagai otak dalam menjalankan sebuah proyek elektronika

3.6.2 Alat dan Bahan Perangkat Lunak

Tabel 3.2 Alat dan Bahan Perangkat Lunak

No	Software	Fungsi
1.	Arduino IDE	Untuk memprogram ESP32–CAM agar dapat menangkap gambar dan berkomunikasi dengan server.
2.	Python + YOLOv5 / YOLOv4-tiny	Algoritma pendeteksian manusia dari citra yang dikirim ESP32–CAM. YOLO dijalankan di server.
3.	<i>Idle Python</i>	Untuk menjalankan model YOLO dan proses inferensi gambar.
4.	Driver FTDI	Dibutuhkan untuk menghubungkan modul FTDI dengan laptop.
5.	Serial Monitor / Web Server	Untuk melihat hasil output atau status deteksi dari ESP32.
6.	Sistem Operasi (<i>Windows/Linux</i>)	Menjalankan semua tools pendukung. Linux lebih direkomendasikan untuk YOLO.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Implementasi sistem merupakan tahap penerapan dari perancangan hardware dan software yang telah dibuat. Pada tahap ini dilakukan perakitan ESP32-CAM, sensor PIR, Arduino Uno, dan relay menjadi satu sistem yang terintegrasi, serta pemrograman menggunakan Arduino IDE. Sistem bekerja dengan mendeteksi gerakan melalui sensor PIR, memproses gambar menggunakan YOLO di server, kemudian mengaktifkan relay dan mengirim notifikasi Telegram. Tahap ini memastikan sistem dapat berjalan sesuai dengan rancangan sebelum dilakukan pengujian.

4.1.1 Implementasi Perangkat Keras

Sistem dibangun dengan mengintegrasikan beberapa komponen utama yang saling terhubung untuk membentuk satu kesatuan sistem deteksi dan kontrol otomatis. Pada tahap ini dilakukan proses perakitan, pengkabelan, serta pengujian koneksi antar komponen untuk memastikan sistem dapat bekerja sesuai dengan perancangan.



Gambar 4. 1 Alat Yang Sudah Jadi

Komponen utama yang digunakan dalam implementasi sistem ini meliputi:

1. ESP32-CAM sebagai modul pengambil citra yang berfungsi menangkap gambar lingkungan secara real-time dan mengirimkannya ke server untuk proses deteksi manusia

2. Sensor PIR sebagai pendeteksi awal gerakan yang memberikan sinyal trigger kepada ESP32-CAM ketika terdapat aktivitas di area pemantauan
3. Arduino Uno sebagai pengendali aktuator yang menerima sinyal dari ESP32-CAM dan mengontrol modul relay
4. Modul relay sebagai pengontrol beban listrik yang berfungsi sebagai saklar otomatis untuk menyalakan atau mematikan lampu
5. Lampu sebagai perangkat output yang dikendalikan berdasarkan hasil deteksi manusia
6. Power supply sebagai sumber daya untuk menyuplai tegangan ke seluruh komponen system

Pada implementasinya, sensor PIR dihubungkan ke ESP32-CAM melalui pin input digital sebagai pemicu awal sistem. Ketika sensor mendeteksi gerakan, ESP32-CAM akan aktif untuk mengambil gambar. Jika hasil deteksi dari server menunjukkan adanya manusia, ESP32-CAM akan mengirimkan sinyal digital ke Arduino Uno. Arduino Uno kemudian mengaktifkan modul relay melalui pin output, sehingga arus listrik mengalir dan lampu menyala.

Setelah seluruh komponen dirangkai sesuai dengan diagram rangkaian, dilakukan pengujian konektivitas untuk memastikan setiap perangkat dapat berkomunikasi dengan baik. Pengujian ini meliputi pengecekan tegangan catu daya, kestabilan sinyal digital, serta respons relay terhadap perintah yang diberikan. Hasil pengujian menunjukkan bahwa sistem perangkat keras dapat bekerja secara stabil sesuai dengan rancangan.

4.1.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak dilakukan pada dua bagian utama, yaitu program pada ESP32-CAM dan sistem deteksi pada server yang menjalankan algoritma YOLO. Kedua bagian ini dirancang agar dapat saling berkomunikasi melalui jaringan internet menggunakan protokol HTTP.

a. Program Pada ESP32-CAM

ESP32-CAM diprogram menggunakan Arduino IDE sebagai platform pengembangan yang digunakan untuk menulis, mengedit, dan mengunggah kode program ke perangkat mikrokontroler. Arduino IDE dipilih karena memiliki antarmuka yang sederhana, mendukung library ESP32, serta memudahkan proses konfigurasi koneksi WiFi, pengaturan kamera, dan komunikasi data dengan server maupun Telegram.

b) Hasil Serial Monitor

Berikut ini merupakan hasil dari serial monitor sebagai berikut:

```
Connecting WiFi.....  
✅ WiFi Connected  
IP Address: 192.168.1.13  
📺 Streaming Ready!  
🚨 Gerakan Terdeteksi
```

Gambar 4. 2 Hasil Serial Monitor dari ESP32-CAM dan Sensor Pir

c) Notifikasi Telegram

Ketika sistem dinyalakan, ESP32-CAM akan melakukan koneksi ke jaringan WiFi. Setelah terhubung, perangkat akan berada dalam kondisi siaga menunggu sinyal dari sensor PIR. Jika terdeteksi gerakan, kamera akan mengambil gambar dan mengirimkannya ke server. Berdasarkan hasil deteksi yang diterima, ESP32-CAM akan menentukan apakah relay perlu diaktifkan atau tidak.



Gambar 4. 3 Hasil Notifikasi Bot Telegram

Sistem juga mengintegrasikan Telegram Bot sebagai media notifikasi. Jika manusia terdeteksi dan lampu dinyalakan, ESP32-CAM akan mengirimkan pesan otomatis ke akun Telegram pengguna sebagai bentuk pemberitahuan real-time.

b. Sistem Deteksi pada Server

Saat sistem diaktifkan, ESP32-CAM akan melakukan koneksi ke jaringan WiFi yang telah dikonfigurasi. Setelah koneksi berhasil, perangkat akan mulai melakukan

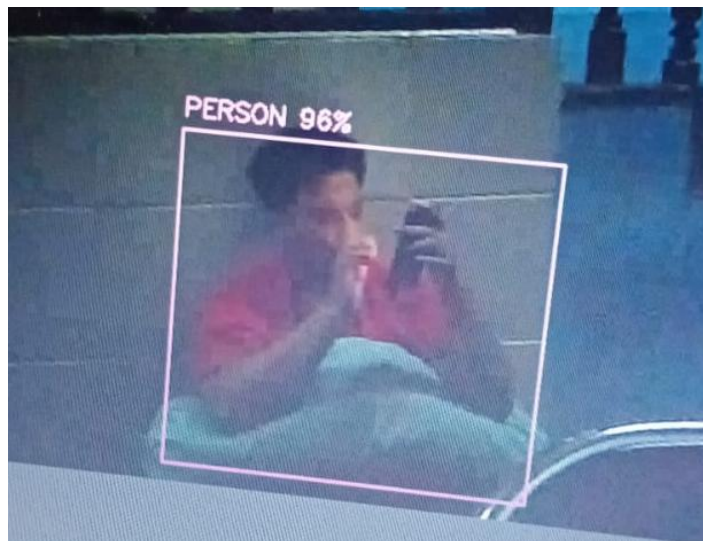
pengambilan gambar secara berkala sesuai interval yang ditentukan dalam program. Gambar yang diambil kemudian dikirimkan ke server menggunakan protokol HTTP untuk dianalisis lebih lanjut. Proses ini berlangsung secara terus-menerus selama sistem aktif. Apabila hasil deteksi dari server menunjukkan adanya manusia, ESP32-CAM akan mengaktifkan pin output digital yang terhubung ke modul relay sehingga lampu menyala. Selain itu, sistem juga mengirimkan notifikasi kepada pengguna sebagai informasi bahwa manusia telah terdeteksi dan perangkat listrik telah diaktifkan.

4.1.3 Implementasi YOLO pada Server

Pada sisi server, digunakan algoritma YOLO (You Only Look Once) untuk mendeteksi keberadaan manusia pada gambar yang dikirimkan oleh ESP32-CAM. Implementasi dilakukan menggunakan bahasa pemrograman Python dengan library pendukung deteksi objek.

a. Hasil Deteksi YOLO

Berikut ini merupakan hasil display menjalankan YOLO untuk mendeteksi manusia seperti berikut:



Gambar 4. 4 Hasil Deteksi YOLO

Proses yang dilakukan pada server dimulai dengan menerima gambar dari ESP32-CAM melalui koneksi jaringan. Gambar tersebut kemudian diproses menggunakan model YOLO yang telah dilatih untuk mengenali objek manusia. Proses deteksi dilakukan dengan menganalisis bounding box dan nilai confidence score untuk memastikan apakah objek yang terdeteksi termasuk kategori manusia.

Setelah proses deteksi selesai, server akan mengirimkan hasil berupa status terdeteksi atau tidak terdeteksi kembali ke ESP32-CAM. Jika manusia terdeteksi, server mengirimkan sinyal

konfirmasi yang akan memicu ESP32-CAM untuk mengaktifkan relay sehingga lampu menyala secara otomatis. Sebaliknya, jika tidak terdeteksi manusia, sistem akan kembali melakukan proses pengambilan gambar sesuai interval yang telah ditentukan.

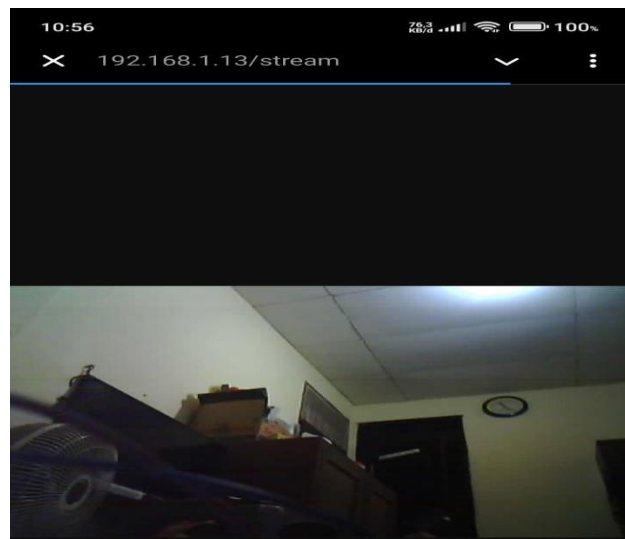
4.2 Hasil Pengujian Sistem

4.2.1 Pengujian Fungsionalitas

Pengujian fungsionalitas dilakukan untuk memastikan bahwa seluruh komponen sistem bekerja sesuai dengan tujuan dan rancangan yang telah ditetapkan pada tahap perancangan. Pengujian ini dilakukan dengan menjalankan sistem secara keseluruhan mulai dari proses pendeteksian gerakan hingga pengaktifan lampu dan pengiriman notifikasi. Berdasarkan hasil pengujian yang telah dilakukan, diperoleh hasil sebagai berikut:

a. Pengujian Hasil Tangkapan ESP32-Cam

Berdasarkan hasil pengujian yang telah dilakukan, ESP32-CAM mampu menangkap gambar dengan baik dan berhasil menghasilkan citra yang jelas serta stabil sesuai dengan konfigurasi sistem yang telah ditentukan. Selama proses pengujian berlangsung, tidak ditemukan kegagalan dalam proses pengambilan gambar (*capture*), sehingga citra yang dihasilkan dapat digunakan secara optimal sebagai input dalam proses deteksi manusia pada server menggunakan algoritma YOLO. Kualitas gambar yang diperoleh dinilai cukup untuk mendukung proses identifikasi objek secara akurat.



Gambar 4. 5 Pengujian Hasil Tangkapan ESP32-CAM

b. Pengujian Hasil Deteksi YOLO

Berdasarkan hasil pengujian yang telah dilakukan, algoritma YOLO pada server mampu mendeteksi keberadaan manusia dengan baik pada gambar yang dikirimkan oleh ESP32-CAM. Sistem berhasil mengidentifikasi objek manusia dengan jumlah bervariasi,

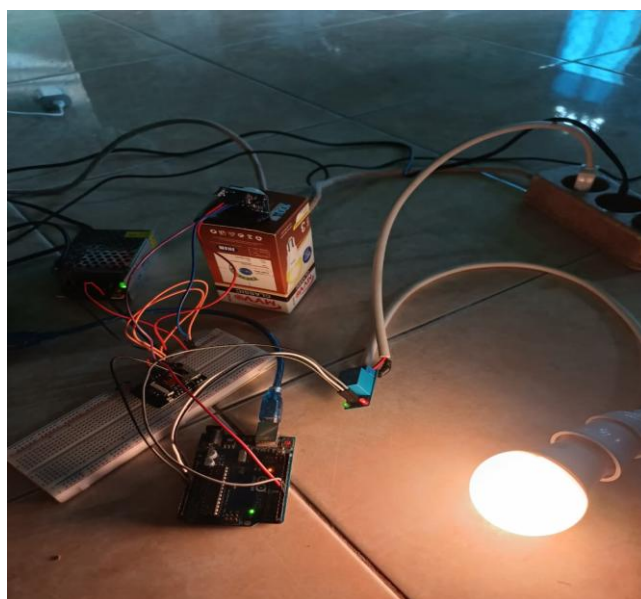
mulai dari 1 hingga 5 orang dalam satu frame gambar [8], Proses deteksi dilakukan dengan menampilkan bounding box pada setiap objek yang teridentifikasi sebagai manusia serta menghasilkan nilai confidence yang menunjukkan tingkat kepercayaan model terhadap hasil deteksi. Selama pengujian pada kondisi pencahayaan dan jarak yang memadai, YOLO menunjukkan performa yang stabil dan mampu membedakan objek manusia secara akurat tanpa kesalahan deteksi yang signifikan.

c. Pengujian Relay

Berdasarkan hasil pengujian sistem, modul relay berhasil aktif secara otomatis ketika sistem mendeteksi keberadaan manusia melalui proses deteksi menggunakan algoritma YOLO. Sinyal trigger yang dikirimkan oleh ESP32-CAM diterima dengan baik oleh rangkaian relay sehingga terjadi perubahan kondisi dari OFF ke ON tanpa keterlambatan yang signifikan. Hal ini menunjukkan bahwa komunikasi antara perangkat deteksi dan aktuator berjalan dengan stabil dan sesuai dengan rancangan sistem.

d. Pengujian Lampu

Lampu sebagai beban listrik berhasil menyala secara otomatis ketika relay dalam kondisi aktif. Selama proses pengujian, lampu merespon dengan baik setiap kali sistem mendeteksi manusia dan mengirimkan sinyal pengaktifan. Tidak ditemukan kegagalan dalam proses penyalan lampu, sehingga dapat disimpulkan bahwa sistem kontrol perangkat listrik bekerja sesuai dengan fungsi yang dirancang dan mampu beroperasi secara otomatis serta responsive.



Gambar 4. 6 Hasil Pengujian Dari Relay dan Lampu Berfungsi.

4.2.2 Pengujian Akurasi Deteksi

Pengujian akurasi dilakukan untuk mengetahui kemampuan sistem dalam mendeteksi manusia berdasarkan variasi jarak dan kondisi pencahayaan. Pengujian dilakukan beberapa kali dengan skenario berbeda untuk melihat konsistensi deteksi sistem.

Tabel 4. 1 Pengujian Berdasarkan Jarak Objek

No.	Jarak (Meter)	Jumlah Orang	Hasil Deteksi	Keterangan
1.	1 Meter	1 Orang	Terdeteksi	Bounding box jelas dan stabil
2.	2 Meter	2 Orang	Terdeteksi	Deteksi akurat tanpa delay
3.	3 Meter	3 Orang	Terdeteksi	Masih terdeteksi dengan baik
4.	4 Meter	4 Orang	Terdeteksi	Ukuran objek mulai kecil namun masih terbaca
5.	5 Meter	5 Orang	Kurang Stabil	Beberapa objek tidak terdeteksi sempurna

Berdasarkan pengujian jarak, sistem bekerja optimal pada jarak 1–3 meter. Pada jarak 4 meter sistem masih mampu mendeteksi manusia, namun pada jarak 5 meter akurasi mulai menurun karena ukuran objek pada citra menjadi lebih kecil sehingga mempengaruhi nilai confidence dari YOLO.

Tabel 4. 2 Pengujian Berdasarkan Kondisi Pencahayaan.

No.	Kondisi Cahaya	Intensitas Cahaya	Hasil Deteksi	Keterangan
1.	Terang	± 300 lux	Terdeteksi	Deteksi masih stabil
2.	Cukup	± 150 lux	Terdeteksi	Masih Akurat
3.	Redup	± 50 lux	Kurang Stabil	Beberapa delay deteksi
4.	Sangat redup	< 30 lux	Tidak stabil	Deteksi sering gagal

Hasil pengujian menunjukkan bahwa sistem bekerja optimal pada kondisi pencahayaan terang hingga cukup. Pada kondisi cahaya redup, kualitas citra menurun sehingga mempengaruhi performa model YOLO dalam mengenali objek manusia.

a) Analisis Hasil Pengujian

Berdasarkan hasil pengujian, akurasi deteksi dipengaruhi oleh beberapa faktor utama, yaitu intensitas cahaya, jarak objek dari kamera, dan sudut pengambilan gambar. Semakin dekat objek dengan kamera dan semakin baik pencahayaan ruangan, maka tingkat akurasi deteksi menjadi lebih tinggi. Sebaliknya, pada jarak yang terlalu jauh dan kondisi

pencahayaannya yang rendah, sistem mengalami penurunan performa karena kualitas citra yang diterima tidak optimal untuk proses identifikasi objek.

4.2.3 Pengujian Waktu Respon

Pengujian waktu respon dilakukan untuk mengetahui kecepatan sistem dalam merespon deteksi manusia hingga lampu menyala. Pengukuran dilakukan mulai dari saat manusia terdeteksi oleh sistem hingga relay aktif dan lampu menyala.

Tabel 4. 3 Kecepatan Jaringan Terhadap Waktu Respon.

No.	Kondisi Jaringan	Waktu Respon (Detik)	Keterangan
1.	Stabil (Wifi Lancar)	1-2 detik	Pengiriman gambar dan penerimaan hasil deteksi berjalan cepat
2.	Kurang Stabil	2-3 detik	Terjadi sedikit delay saat pengiriman data ke server
3.	Lambat	> 3 detik	Waktu respon meningkat akibat keterlambatan komunikasi data

Kecepatan koneksi jaringan sangat mempengaruhi waktu respon sistem karena proses pengiriman gambar ke server dan penerimaan hasil deteksi bergantung pada kestabilan jaringan internet. Semakin stabil jaringan, semakin cepat sistem merespon.

Tabel 4. 4 Pengaruh Proses Deteksi Terhadap Waktu Respon

No.	Kondisi Objek	Waktu Respon (Detik)	Keterangan
1.	Jarak dekat & Cahaya cukup	1-2 Detik	Deteksi cepat dan confidence tinggi
2.	Jarak sedang	2 Detik	Masih terdeteksi dengan baik
3.	Jarak jauh/Cahaya redup	2-3 Detik	Terjadi sedikit delay karena objek kurang jelas

Proses deteksi pada server dipengaruhi oleh jarak objek dan kondisi pencahayaan. Objek yang lebih dekat dan pencahayaan yang baik menghasilkan proses inferensi yang lebih cepat dibandingkan kondisi jarak jauh atau cahaya redup.

Tabel 4. 5 Pengaruh Komunikasi Sensor terhadap Respon Sistem

No.	Kondisi Pengujian	Respon Sistem	Keterangan
1.	Jarak Dekat	Stabil	Sensor langsung mengirim sinyal ke ESP32
2.	Jarak Sedang	Stabil	Tidak ada kegagalan trigger
3.	Berbagai Kondisi Cahaya	Stabil	Komunikasi teta berjalan normal

Komunikasi antara sensor PIR dan ESP32-CAM menunjukkan performa yang stabil pada seluruh kondisi pengujian. Sensor tetap mampu mengirimkan sinyal trigger dengan baik sehingga tidak memberikan pengaruh signifikan terhadap keterlambatan sistem.

4.3 Analisis Hasil Pengujian

Sistem yang dibangun mampu mendeteksi keberadaan manusia secara otomatis melalui proses pengambilan gambar oleh ESP32-CAM yang kemudian dianalisis menggunakan algoritma YOLO pada server. Ketika manusia terdeteksi, sistem secara langsung mengirimkan sinyal untuk mengaktifkan modul relay sehingga lampu dapat menyala tanpa memerlukan intervensi manual dari pengguna. Proses pengendalian ini berjalan sesuai dengan logika program yang telah dirancang, sehingga fungsi otomatisasi dapat terlaksana dengan baik. Dari sisi waktu respon, sistem mampu memberikan respon dalam waktu yang relatif cepat, meskipun terdapat sedikit delay beberapa detik yang dipengaruhi oleh proses pengiriman gambar ke server, proses deteksi menggunakan YOLO, serta kestabilan jaringan internet. Namun demikian, keterlambatan tersebut masih berada dalam batas wajar dan tidak mengganggu fungsi utama sistem sebagai sistem kontrol otomatis berbasis deteksi manusia.

Namun, performa sistem masih dipengaruhi oleh kondisi lingkungan seperti pencahayaan dan stabilitas jaringan. Pada kondisi pencahayaan rendah, tingkat deteksi mengalami penurunan. Selain itu, karena proses deteksi dilakukan pada server, sistem sangat bergantung pada konektivitas jaringan.

4.4 Efektivitas Sistem Terhadap Penghematan Energi

Sistem yang dikembangkan dalam penelitian ini menunjukkan efektivitas dalam mengontrol penggunaan lampu berdasarkan keberadaan manusia di dalam ruangan. Mekanisme

kerja sistem memungkinkan lampu menyala secara otomatis ketika manusia terdeteksi oleh kamera dan dianalisis menggunakan algoritma YOLO, serta tetap dalam kondisi mati ketika tidak terdapat aktivitas di dalam ruangan. Dengan pola kerja tersebut, penggunaan energi listrik menjadi lebih terkontrol karena lampu tidak lagi menyala secara terus-menerus tanpa adanya kebutuhan.

Berdasarkan hasil pengujian yang telah dilakukan, sistem mampu mencegah kondisi pemborosan energi yang umumnya terjadi akibat kelalaian pengguna dalam mematikan lampu. Meskipun terdapat sedikit delay dalam proses deteksi dan pengaktifan lampu yang dipengaruhi oleh koneksi jaringan dan proses analisis pada server, hal tersebut tidak mengurangi fungsi utama sistem dalam mengoptimalkan penggunaan energi. Selama kondisi jaringan stabil dan pencahayaan memadai, sistem dapat bekerja secara konsisten dalam mengontrol beban listrik.

Dengan demikian, implementasi sistem ini berpotensi mendukung efisiensi penggunaan energi pada rumah tinggal, khususnya dalam mengurangi konsumsi listrik yang tidak diperlukan. Sistem berbasis deteksi manusia ini menjadi salah satu solusi otomatisasi yang dapat membantu meningkatkan kesadaran serta efektivitas pengelolaan energi di lingkungan rumah tangga.

4.5 Kelebihan dan Keterbatasan Sistem

Setiap sistem yang dikembangkan tentu memiliki kelebihan dan keterbatasan berdasarkan hasil implementasi dan pengujian yang telah dilakukan. Analisis terhadap kelebihan dan keterbatasan ini bertujuan untuk mengetahui sejauh mana sistem mampu memenuhi tujuan penelitian serta mengidentifikasi aspek yang masih dapat dikembangkan pada penelitian selanjutnya. Berikut merupakan kelebihan dan keterbatasan dari sistem deteksi manusia berbasis ESP32-CAM dan algoritma YOLO yang telah dirancang dan diuji. Berikut ini merupakan kelebihan dan keterbatasan system sebagai berikut:

4.5.1 Kelebihan Sistem

a) Sistem bekerja secara otomatis

Sistem mampu mendeteksi keberadaan manusia dan mengontrol lampu tanpa memerlukan intervensi manual dari pengguna sehingga meningkatkan efisiensi dan kenyamanan.

b) Respon real-time

Sistem memberikan respon dalam waktu relatif cepat dengan jeda beberapa detik, sehingga masih dapat dikategorikan sebagai sistem real-time untuk kebutuhan rumah tinggal.

- c) Biaya implementasi relatif rendah
Penggunaan ESP32-CAM, Arduino Uno, dan modul relay menjadikan sistem ini lebih ekonomis dibandingkan sistem smart home berbasis perangkat komersial.
- d) Mudah diterapkan pada rumah tinggal
Rangkaian sistem sederhana dan dapat dipasang pada instalasi lampu yang sudah ada tanpa perubahan besar pada infrastruktur rumah.
- e) Terintegrasi dengan notifikasi Telegram
Sistem mampu memberikan notifikasi jarak jauh kepada pengguna ketika manusia terdeteksi sehingga meningkatkan aspek monitoring dan keamanan.
- f) Mampu mendeteksi lebih dari satu orang
Algoritma YOLO dapat mengenali hingga beberapa orang dalam satu frame sehingga sistem lebih fleksibel terhadap kondisi ruangan.

4.5.2 Keterbatasan Sistem

- a) Bergantung pada koneksi internet
Karena proses deteksi dilakukan pada server, sistem sangat bergantung pada kestabilan jaringan untuk mengirim dan menerima data.
- b) Performa dipengaruhi kondisi pencahayaan
Kualitas deteksi menurun pada kondisi cahaya redup karena citra yang dihasilkan kurang optimal untuk proses identifikasi objek.
- c) Proses deteksi tidak dilakukan langsung di ESP32
ESP32-CAM hanya berfungsi sebagai pengambil gambar, sedangkan proses inferensi dilakukan pada server sehingga membutuhkan waktu tambahan untuk komunikasi data.
- d) Terdapat sedikit delay dalam proses respon
Waktu respon dipengaruhi oleh kecepatan jaringan dan proses deteksi sehingga tidak sepenuhnya instan.
- e) Jarak deteksi terbatas
Akurasi sistem menurun ketika objek berada pada jarak yang terlalu jauh dari kamera karena ukuran objek dalam citra menjadi lebih kecil.
- f) Tidak menggunakan sistem cadangan (*offline mode*)
Apabila koneksi internet terputus, sistem tidak dapat melakukan deteksi karena tidak tersedia mekanisme pemrosesan lokal.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil implementasi dan pengujian sistem yang telah dilakukan pada BAB IV, dapat disimpulkan bahwa:

1. Sistem yang dirancang berhasil mengimplementasikan algoritma YOLO pada ESP32-CAM untuk mendeteksi manusia secara real-time. Berdasarkan hasil pengujian, sistem mampu mendeteksi 1–4 orang pada jarak 1–4 meter dengan hasil deteksi yang baik, sedangkan pada jarak 5 meter deteksi mulai kurang stabil karena ukuran objek lebih kecil. Sistem juga bekerja optimal pada kondisi pencahayaan ± 300 lux dan ± 150 lux, sementara pada ± 50 lux deteksi mulai kurang stabil dan di bawah 30 lux sering gagal. Waktu respon sistem berkisar 1–2 detik pada jaringan stabil, 2–3 detik pada jaringan kurang stabil, dan dapat lebih dari 3 detik pada jaringan lambat, namun sistem masih mampu memberikan respon cukup cepat sehingga tetap dapat dikategorikan sebagai sistem real-time untuk otomatisasi rumah tinggal.
2. Kinerja sistem dipengaruhi oleh kualitas jaringan internet dan proses pengolahan data pada server. Berdasarkan hasil pengujian, waktu respon sistem dalam mendeteksi manusia dan mengaktifkan lampu berada pada kisaran 1–2 detik pada kondisi jaringan stabil, sekitar 2–3 detik pada jaringan yang kurang stabil, dan dapat mencapai lebih dari 3 detik ketika jaringan lambat. Hal ini terjadi karena proses sistem melibatkan beberapa tahapan, yaitu pengambilan gambar oleh ESP32-CAM, pengiriman data ke server, proses deteksi menggunakan algoritma YOLO, serta pengiriman kembali perintah ke modul relay. Meskipun terdapat keterlambatan beberapa detik, sistem masih mampu memberikan respon yang cukup cepat sehingga tetap dapat digunakan untuk sistem kontrol otomatis pada rumah tinggal.
3. Integrasi antara perangkat keras yang digunakan, yaitu ESP32-CAM, sensor PIR, modul relay, dan lampu, dapat bekerja dengan baik sesuai dengan perancangan sistem. Berdasarkan hasil pengujian, sensor PIR mampu mendeteksi pergerakan manusia pada jarak sekitar 1–5 meter, dimana deteksi paling optimal terjadi pada jarak 1–3 meter, masih dapat terdeteksi pada 4 meter, dan mulai menurun pada 5 meter. Ketika sensor mendeteksi pergerakan dan sistem mendeteksi keberadaan manusia melalui kamera, modul relay akan mengaktifkan lampu secara otomatis. Proses ini berlangsung dengan

waktu respon beberapa detik sehingga menunjukkan bahwa sistem otomatisasi yang dirancang mampu berfungsi dengan baik dalam mengontrol peralatan listrik pada rumah tinggal.

5.2 Saran

Pada penelitian selanjutnya, ada beberapa poin saran yang dapat diberikan berdasarkan hasil penelitian:

1. Mengimplementasikan deteksi langsung pada perangkat (edge computing)
Untuk mengurangi ketergantungan pada server dan koneksi internet, proses deteksi dapat dikembangkan agar berjalan langsung di perangkat menggunakan model yang lebih ringan seperti Tiny-YOLO atau TensorFlow Lite.
2. Meningkatkan akurasi deteksi pada kondisi minim Cahaya
Penambahan lampu inframerah (IR) atau kamera dengan fitur night vision dapat meningkatkan performa sistem pada kondisi pencahayaan rendah.
3. Menambahkan fitur monitoring konsumsi daya Listrik
Sistem dapat dikembangkan dengan menambahkan sensor arus (misalnya PZEM atau ACS712) untuk mengukur secara langsung penghematan energi yang dihasilkan.
4. Mengintegrasikan dengan platform smart home lainnya
5. Sistem dapat dikembangkan agar terhubung dengan aplikasi seperti Google Home atau platform IoT dashboard sehingga kontrol menjadi lebih fleksibel.
6. Mengoptimalkan waktu respon system
Perbaikan pada sisi jaringan atau penggunaan protokol komunikasi yang lebih ringan seperti MQTT dapat mempercepat respon sistem.
7. Menambahkan fitur identifikasi jumlah orang secara lebih detail
Sistem dapat dikembangkan tidak hanya mendeteksi keberadaan manusia, tetapi juga menghitung jumlah orang untuk kebutuhan analisis ruangan.
8. Meningkatkan keamanan data dan system
Implementasi enkripsi data dan autentikasi yang lebih kuat diperlukan untuk meningkatkan keamanan komunikasi antara perangkat dan server.

DAFTAR PUSTAKA

- [1] “Wulandari, L., Umar, D. D., Septiani, D., Iskandar, H. H., Safina, M., & Haq, V. A. (2022). Analisis pengaruh globalisasi dan perkembangan teknologi nuklir terhadap lingkungan hidup yang berkelanjutan (sustainable environment). *Jurnal Bisnis Dan Manajemen West Science*, 1(01), 36-50.”
- [2] “Yusuf, M., & Sodik, M. (2023). Penggunaan teknologi Internet of Things (IoT) dalam pengelolaan fasilitas dan infrastruktur lembaga pendidikan Islam. *PROPHETIK: Jurnal Kajian Keislaman*, 1(2), 65-82.
- [3] Megawati, S. (2021). Pengembangan sistem teknologi internet of things yang perlu dikembangkan negara indonesia. *JIEET (Journal of Information Engineering and Educational Technology)*, 5(1), 19-26.
- [4] M. Syaifullah Tamsir, A. B. Arief, A. Achmad, S. Panggalo, P. Studi, and T. Elektro, “IMPLEMENTATION OF YOLOv4 WITH IMAGE PRE-PROCESSING FILTERS FOR OPTIMIZATION OF ESP32-CAM BASED HOME SECURITY MONITORING SYSTEM IMPLEMENTASI YOLOv4 DENGAN FILTER PRE PROCESSING GAMBAR UNTUK OPTIMALISASI SISTEM MONITORING KEAMANAN RUMAH BERBASIS ESP32-CAM,” *Journal of Scientech Research and Development*, vol. 6, no. 2, 2024, [Online]. Available: <https://idm.or.id/JSCR/in>
- [5] M. Wibowo, R. Tullah, and W. Ricesa, “Studi Perbandingan Algoritma YOLO dan FOMO untuk Object Detection pada Perangkat ESP32-CAM,” vol. 11, no. 1, 2025.
- [6] A. Rianto, “Penerapan Pengenalan Wajah dengan Algoritma Viola-Jones Artificial Intelligence Berbasis Internet of Things Menggunakan ESP32-CAM,” *Jurnal FORTECH*, vol. 5, no. 1, pp. 48–57, Jun. 2024, doi: 10.56795/fortech.v5i1.5107.
- [7] Aldiasyah, M. I., Sujono, S., & Widya, M. A. A. (2026). Sistem Pengendalian dan Monitoring Konsumsi Daya Lampu Rumah Berbasis IoT. *Exact Papers in Compilation (EPiC)*, 8(1), 1-12.
- [8] I. Putu, S. Yoga, G. Sukadarmika, R. S. Hartati, and Y. Divayana, “Pendeteksi Jumlah Orang pada Sistem Bangunan Pintar Menggunakan Algoritma You Only Look Once,” *Majalah Ilmiah Teknologi Elektro*, vol. 22, no. 1, doi: 10.24843/MITE.

- [9] J. Renhard, O. Tampubolon, W. Ramadan, J. Buulolo, P. P. Purba, and S. Dohot Siregar, “Rancang Bangun IoT Otomatis Berbasis Sensor PIR untuk Menghemat Energi Listrik pada saat Ruangan Kosong,” *Jurnal Sistem Komputer dan Informatika (JSON) Hal*, vol. 6, no. 4, pp. 257–263, 2025, doi: 10.30865/json.v6i4.
- [10] A. Herwandi, A. A. Ramadhan, N. T. Sunggono, and F. Ferawati, “Analisis Kinerja ESP32-CAM Dalam Mendeteksi Objek,” *bit-Tech*, vol. 7, no. 3, pp. 1014–1021, Apr. 2025, doi: 10.32877/bt.v7i3.2296.
- [11] González, O. A. G., Soberanes, A. M. P., Ortega, V. G., & Savedra, J. C. S. (2019). Embedded system for human detection applied to domotics. *Research in Computing Science*, 148(10), 103-115.
- [12] F. Dharma Adhinata, D. Putra Rakhmadani, A. Jala, and T. Segara, “YOLO Algorithm for Detecting People in Social Distancing System,” *TRANSFORMTIKA*, vol. 19, no. 1, pp. 1–7, 2021, [Online]. Available: <https://pjreddie.com/>.
- [13] R. Gustia, M. Hasanah, W. Febriani, A. Fradana, T. Informasi Teknik Komputer, and P. Negeri Padang, “Telegram dan Spreadsheet,” *Jurnal Pustaka AI*, vol. 5, no. 3, pp. 695–700, 2025, doi: 10.55382/jurnalpustakaai.v5i3.1482.
- [14] D. Febrian Munggaran and Z. Alamsyah, “Pemanfaatan Tongkat Berbasis IoT dan Yolo V3 Untuk Meningkatkan Mobilitas dan Keamanan Penyandang Tunanetra,” vol. 6, no. 2, pp. 94–103, 2025, doi: 10.37859/coscitech.v6i2.9793.
- [15] S. Suherman, H. Hermansyah, and J. Syahpita, “Sistem Alarm Deteksi Gerak Berbasis IoT Menggunakan Sensor PIR dan ESP32 dengan Notifikasi Telegram Real-Time,” *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, vol. 4, no. 2, pp. 1469–1476, Nov. 2025, doi: 10.62712/juktisi.v4i2.701.
- [16] Irawan, R. (2023). Analisis Sistem Kerja Interlock Pada Relay Dengan Menggunakan Tegangan Direct Current (DC) Sebagai Pengaman Rumah Tinggal. *ISMETEK*, 16(1).
- [17] U. Mahanin Tyas, A. Apri Buckhari, P. Studi Pendidikan Teknologi Informasi, and P. Studi Pendidikan Teknologi dan Kejuruan, “IMPLEMENTASI APLIKASI ARDUINO IDE PADA MATA KULIAH SISTEM DIGITAL,” 2023.

- [18] M. R. Pratama and D. Laksmiati, "Prototipe Sistem Deteksi Burung Menggunakan ESP32-Cam dan Algoritma YOLO," *Jurnal Ilmiah Giga*, vol. 27, no. 2, pp. 78–87, Jan. 2025, doi: 10.47313/jig.v27i2.3826.
- [19] H. A. Kusuma, S. B. Wijaya, and D. Nusyirwan, "SISTEM KEAMANAN RUMAH BERBASIS ESP32-CAM DAN TELEGRAM SEBAGAI NOTIFIKASI," *Infotronik : Jurnal Teknologi Informasi dan Elektronika*, vol. 8, no. 1, p. 30, Jun. 2023, doi: 10.32897/infotronik.2023.8.1.2291.
- [20] M. W. Aprilia and A. Purwanto, "IOT-BASED BOARDING ROOM SECURITY SYSTEM USING PIR SENSOR AND ESP32-CAM WITH TELEGRAM NOTIFICATION," *Jurnal Ilmu Fisika dan Terapannya*, vol. 12, no. 2, pp. 95–102, Oct. 2025, doi: 10.21831/jifta.v12i2.25388.
- [21] S. Rumere and V. Manullang, "RANCANG BANGUN SISTEM KEAMANAN RUMAH BERBASIS INTERNET OF THINGS (IoT) DENGAN MENGGUNAKAN SENSOR PIR DAN ESP32-CAM," vol. 12, no. 1, pp. 9–16, 2024.
- [22] J. T. Sirait and A. I. Santoso, "Perancangan Sistem Keamanan Rumah berbasis IoT dan Aplikasi Telegram," *Jurnal Minfo Polgan*, vol. 14, no. 1, pp. 676–681, May 2025, doi: 10.33395/jmp.v14i1.14834.
- [23] P. Pebriana Sitanggang *et al.*, "Penerapan Metode Waterfall Pada Sistem Informasi Pendaftaran dan Pembayaran Membership Komunitas United Indonesia," *Jurnal Komputer Antartika*, vol. 1, p. 2023, [Online]. Available: <https://ejournal.mediaantartika.id/index.php/jka>
- [24] A. Lutfiyani, F. Fachri, E. Wahyuningsih, and P. Studi Teknik informatika, "INSOLOGI: Jurnal Sains dan Teknologi Deteksi Hewan Secara Real-Time Menggunakan Algoritma You Only Look Once (YOLO)," *Media Cetak*, vol. 5, no. 1, pp. 263–269, 2026, doi: 10.55123/insologi.v5i1.7592.
- [25] Pakpahan, M. Y., & Hidayat, H. (2025). *RANCANGAN APLIKASI MOBILE UNTUK KONSELING DAN PENGADUAN PEREMPUAN DAN ANAK DI KABUPATEN AGAM PADA UNIT PELAKSANA TEKNIS DAERAH PERLINDUNGAN PEREMPUAN DAN ANAK (UPTD PPA)* (Doctoral dissertation, universitas bung hatta).

- [26] Sunaryo, B., Rusdi, J. F., Salam, S., Abu, N. A., Sabar, M., & Agustina, N. (2023, June). Tracking technology for tourist behavior. In *THE 6TH ENGINEERING SCIENCE AND TECHNOLOGY INTERNATIONAL CONFERENCE (ESTIC 2021): Applied Technology for Sustainable Development* (Vol. 2691, No. 1, p. 080002). AIP Publishing LLC.
- [27] Zulfadli, Arnita, Sunaryo, B., Amelia, R., & Sukma, D. N. E. (2023, June). Computerized framework for assessment of OBE in Bung Hatta University. In *THE 6TH ENGINEERING SCIENCE AND TECHNOLOGY INTERNATIONAL CONFERENCE (ESTIC 2021): Applied Technology for Sustainable Development* (Vol. 2691, No. 1, p. 080001). AIP Publishing LLC.
- [28] Rika, A., & Arnita, A. (2025). *RANCANG BANGUN SISTEM PEMANTAUAN DAN FILTERING AIR PADA BUDIDAYA IKAN AIR TAWAR BERBASIS IOT (INTERNET OF THINGS)* (Doctoral dissertation, Universitas Bung Hatta).

LAMPIRAN

