



TUGAS AKHIR (AE43250)

**IMPLEMENTASI PENGENDALIAN *ELECTRICAL FACILITY*
DAN *ILLEGAL ACTIVITY DETECTION* DI MASJID
BAHRUL ULUM MENGGUNAKAN METODE YOLO**

Vemas Ardiyan Syah

NRP. 0922040025

DOSEN PEMBIMBING:

NOORMAN RINANTO, S.T., M.T., Ph.D.

M. LUKMAN ARIF, S.Pd.I., M.Pd.

**PROGRAM STUDI D4 TEKNIK OTOMASI
JURUSAN TEKNIK KELISTRIKAN KAPAL
POLITEKNIK PERKAPALAN NEGERI SURABAYA
SURABAYA**

2026

(halaman ini sengaja dikosongkan)



PPNS POLITEKNIK
PERKAPALAN
NEGERI SURABAYA

TUGAS AKHIR (AE43250)

**IMPLEMENTASI PENGENDALIAN *ELECTRICAL FACILITY*
DAN *ILLEGAL ACTIVITY DETECTION* DI MASJID
BAHRUL ULUM MENGGUNAKAN METODE YOLO**

Vemas Ardiyan Syah

NRP. 0922040025

DOSEN PEMBIMBING:

NOORMAN RINANTO, S.T., M.T., Ph.D.

M. LUKMAN ARIF, S.Pd.I., M.Pd.

**PROGRAM STUDI D4 TEKNIK OTOMASI
JURUSAN TEKNIK KELISTRIKAN KAPAL
POLITEKNIK PERKAPALAN NEGERI SURABAYA
SURABAYA**

2026

(halaman ini sengaja dikosongkan)

**LEMBAR PENGESAHAN
TUGAS AKHIR**

**IMPLEMENTASI PENGENDALIAN *ELECTRICAL FACILITY* DAN
ILLEGAL ACTIVITY DETECTION DI MASJID BAHRUL ULUM
MENGUNAKAN METODE YOLO**

Disusun Oleh:
Vemas Ardiyan Syah
0922040025

**Diajukan Untuk Memenuhi Salah Satu Syarat Kelulusan
Program Studi D4 Teknik Otomasi
Jurusan Teknik Kelistrikan Kapal
POLITEKNIK PERKAPALAN NEGERI SURABAYA**

**Disetujui oleh Tim Penguji Tugas Akhir Tanggal Ujian : 21 Juli 2026
Periode Wisuda : Oktober 2026**

Menyetujui,

Dosen Penguji

1. Noorman Rinanto, S.T., M.T., Ph.D.
2. Agus Khumaidi, S.ST., M.T.
1. Afif Zuhri Arfianto, S.T., M.T.
2. Muhammad Khoirul Hasin, S.Kom., M.Kom.

NIDN

(0014107607)
(0017089303)
(0003128701)
(0025128802)

Tanda Tangan

(.....)
(.....)
(.....)
(.....)

Dosen Pembimbing

1. Noorman Rinanto, S.T., M.T., Ph.D.
2. Muhammad Lukman Arif, S.Pd.I., M.Pd.

NIDN

(0014107607)
(0010099004)

Tanda Tangan

(.....)
(.....)

**Menyetujui
Ketua Jurusan,**



Isa Rachman, S.T., M.T.
NIP. 198008162008121001

**Mengetahui
Koordinator Program Studi,**

Agus Khumaidi, S.ST., M.T.
NIP. 199308172020121004

(halaman ini sengaja dikosongkan)

 PPNS PUSAT PENELITIAN DAN PENGABDIAN MASYARAKAT	<u>PERNYATAAN BEBAS PLAGIAT</u>	No. : F.WD I. Date : 3 Nopember 2015 Rev : 01 Page :1 dari 1
--	---	--

Yang bertanda tangan di bawah ini :

Nama : Vemas Ardiyan Syah

NRP : 0922040025

Jurusan/Prodi : Teknik Kelistrikan Kapal / D4 – Teknik Otomasi

Dengan ini menyatakan dengan sesungguhnya bahwa :

Tugas Akhir yang akan saya kerjakan dengan judul :

Implementasi Pengendalian *Electrical Facility* dan *Illegal Activity Detection* di Masjid Bahrul Ulum Menggunakan Metode YOLO

Adalah benar karya saya sendiri dan bukan plagiat dari karya orang lain.

Apabila dikemudian hari terbukti terdapat plagiat dalam karya ilmiah tersebut, maka saya bersedia menerima sanksi sesuai ketentuan peraturan yang berlaku.

Demikian surat pernyataan ini saya buat dengan penuh tanggung jawab.

Surabaya, 18 Agustus 2026

Yang membuat pernyataan



Vemas Ardiyan Syah
NRP. 0922040025

(halaman ini sengaja dikosongkan)

KATA PENGANTAR

Segala puji bagi Allah SWT atas segala rahmat dan karunia-Nya, saya diberi kesempatan untuk bisa menyusun Tugas Akhir ini dengan lancar sampai tuntas. Tidak lupa sholawat serta salam kepada junjungan nabi besar Muhammad SAW yang telah menuntun umat manusia dari jalan kegelapan menuju jalan yang terang benderang yaitu agama islam.

Tidak banyak yang bisa penulis ungkapkan disini selain beribu-ribu kata terima kasih kepada setiap pihak yang telah memberikan dukungan selama proses penyelesaian tugas akhir dengan judul “Implementasi Pengendalian *Electrical Facility* dan *Illegal Activity Detection* di Masjid Bahrul Ulum Menggunakan Metode YOLO”, sebagai salah satu syarat untuk menyelesaikan jenjang pendidikan di Politeknik Perkapalan Negeri Surabaya, Program Studi D4-Teknik Otomasi. Setiap segala jenis dukungan, semangat, arahan dan perhatian yang diberikan sangatlah berarti bagi penulis. Maka dengan ini, izinkan penulis menyampaikan rasa terima kasih tersebut kepada:

1. Kedua orang tua tercinta, Ayah Syaroni dan Ibu Mukaromi yang senantiasa memberikan doa, nasihat, apresiasi, dukungan, semangat dan kasih sayang yang tiada duanya kepada penulis. Penulis tidak akan bisa berproses dan berkembang baik sampai sejauh ini tanda sosok keduanya yang tidak pernah lelah membersamai, baik disaat bahagia maupun saat terpuruk. Doa-doa terbaik selalu penulis limpahkan kepada beliau-beliau sampai kapanpun.
2. Bapak Rachmad Tri Soelistijono, S.T., M.T., selaku Direktur Politeknik Perkapalan Negeri Surabaya.
3. Bapak Isa Rachman, S.T., M.T., selaku Ketua Jurusan Teknik Kelistrikan Kapal Politeknik Perkapalan Negeri Surabaya.
4. Bapak Agus Khumaidi, S.ST., M.T., selaku Koordinator Program Studi Teknik Otomasi Politeknik Perkapalan Negeri Surabaya.
5. Bapak Ryan Yudha Adhitya, S.ST., M.T., selaku Koordinator Tugas Akhir Program Studi Teknik Otomasi Politeknik Perkapalan Negeri Surabaya Tahun 2026.

6. Bapak Noorman Rinanto, S.T., M.T., Ph.D., selaku dosen pembimbing I yang senantiasa memberi saran, masukan, dukungan dan telah kebersamai penulis selama pengerjaan tugas akhir ini.
7. Bapak Muhammad Lukman Arif, S.Pd.I., M.Pd., selaku dosen pembimbing II yang senantiasa berkenan menjadi rekan diskusi yang suportif dan selalu memberikan akses penuh termasuk perizinan kepada penulis selama proses pengerjaan tugas akhir di Masjid Bahrul Ulum.
8. Seluruh dosen dan tenaga pendidik yang telah menyertai proses belajar penulis selama kurang lebih 4 tahun di Politeknik Perkapalan Negeri Surabaya.
9. Saudara Habibi, yang mendukung, menyemangati dan mendoakan yang terbaik kepada penulis. Tiada kata lelah dalam menghibur dan menyertai penulis.
10. Teman kepercayaan penulis, Alm. Agung Sukma Kurniawan. Sebagai sahabat yang siap sedia untuk berkenan kebersamai suka, duka, selama di organisasi SKI PPNS, sampai mencari uang bersama dari semester 1 sampai semester 6. Atas impian dan semangat bersama, penulis ingin mewujudkan gelar sarjana terapan dan usaha yang telah dibangun bersama, Al-fatihah.
11. Rekan-rekan seperjuangan Program Studi Teknik Otomasi angkatan 2022, yang saling menyemangati, membantu, menguatkan dan mendoakan satu sama lain.
12. Saudari Nayya, yang senantiasa kebersamai dan membantu secara ilmu dalam proses pengerjaan tugas akhir ini sedari awal, mendengarkan keluh kesah dan kerap memberi apresiasi kepada penulis agar selalu termotivasi dalam menyelesaikan studi sarjana terapan ini tepat waktu dan lulus bersama-sama.
13. Teman-teman dekat penulis selama kuliah, Majid, Bima, Rama, Ilyas, Najieb Mirza, Nadi, Fikri, Afifah, Kholidah, Mas Sindu dan Baihaqi. yang senantiasa *men-support*, menjadi tempat cerita dan diskusi. Tanpa kalian, penulis tidak akan bisa *survive* sampai sejauh ini.
14. Seluruh teman dan pihak yang tidak bisa penulis jabarkan satu persatu.

Semoga Allah SWT memberikan balasan yang berlipat ganda atas semua kebaikan yang telah diberikan kepada penulis. Tugas Akhir ini masih sangat jauh dari kata sempurna, kritik dan saran yang dapat menyempurnakan penyusunan Tugas Akhir ini sangat diperlukan. Terakhir, semoga Tugas Akhir ini bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi dimasa yang akan datang.

Surabaya, 18 Agustus 2026

Penulis

(halaman ini sengaja dikosongkan)

IMPLEMENTASI PENGENDALIAN *ELECTRICAL FACILITY* DAN *ILLEGAL ACTIVITY DETECTION* DI MASJID BAHRUL ULUM MENGGUNAKAN METODE YOLO

Vemas Ardiyan Syah

ABSTRAK

Pengelolaan fasilitas listrik di Masjid Bahrul Ulum PPNS, khususnya area jamaah wanita lantai dua, masih menghadapi kendala efisiensi energi dan pengawasan aktivitas ruang. Lampu dan kipas berpotensi tetap aktif saat tidak terdapat jamaah sehingga meningkatkan konsumsi daya listrik. Penelitian ini mengembangkan sistem kontrol cerdas berbasis *Internet of Things* (IoT) menggunakan metode *You Only Look Once* versi 8 (YOLOv8) untuk mendeteksi keberadaan jamaah, aktivitas ilegal membuang sampah, dan aktivitas ilegal menginap sebagai dasar pengendalian otomatis fasilitas listrik. Sistem dibangun menggunakan ESP32, kamera CCTV, sensor LDR, RTC DS3231, *Relay*, HiveMQ, Cloudflare Tunnel, Supabase, serta web *monitoring* berbasis *real-time*. Hasil pengujian menunjukkan perangkat mampu beroperasi dan berkomunikasi dengan baik melalui jaringan internet. Model YOLOv8 menghasilkan rata-rata nilai *confidence* sebesar 80% pada deteksi jamaah, 85% pada aktivitas tidur, dan 79% pada aktivitas membuang sampah. Implementasi sistem mampu mengurangi konsumsi daya listrik sekitar 449 Watt atau 32% dibandingkan kondisi sebelum penerapan sistem. Dengan demikian, sistem yang dikembangkan mampu meningkatkan efisiensi penggunaan energi, mendukung *monitoring* fasilitas secara *real-time*, serta membantu pengawasan aktivitas di lingkungan Masjid Bahrul Ulum PPNS.

Kata kunci: *Energy Efficiency*, *IoT Monitoring*, *Smart Mosque*, YOLO

(halaman ini sengaja dikosongkan)

IMPLEMENTATION OF ELECTRICAL FACILITY MONITORING AND ILLEGAL ACTIVITY DETECTION AT THE BAHRUL ULUM MOSQUE USING THE YOLO METHOD

Vemas Ardiyan Syah

ABSTRACT

The management of electrical facilities at Bahrul Ulum Mosque, PPNS, particularly in the women's prayer area on the second floor, faces challenges related to energy efficiency and activity monitoring. Lights and fans may remain on even when no worshippers are present, resulting in unnecessary electricity consumption. This study develops an IoT-based intelligent control and monitoring system that uses YOLOv8 to detect the presence of worshippers, littering, and unauthorized overnight stays, and to provide inputs for the automatic control of electrical facilities. The system integrates an ESP32 microcontroller, a CCTV camera, a LDR sensor, a DS3231 RTC, Relay modules, HiveMQ, Cloudflare Tunnel, Supabase, and a real-time web-based monitoring platform. Experimental results showed that all devices operated reliably and communicated effectively over the Internet. The YOLOv8 model achieved average confidence scores of 80% for worshipper detection, 85% for sleeping activity detection, and 79% for littering activity detection. In addition, the implemented system reduced the electrical load by approximately 449 W (32%) compared with the pre-implementation condition. These results indicate that the proposed system can improve energy efficiency, enable real-time facility monitoring, and support the detection of prohibited activities within Bahrul Ulum Mosque at PPNS.

Keywords: Energy Efficiency, IoT Monitoring, Smart Mosque, YOLO

(halaman ini sengaja dikosongkan)

DAFTAR ISI

LEMBAR PENGESAHAN TUGAS AKHIR	i
PERNYATAAN BEBAS PLAGIAT.....	iii
KATA PENGANTAR.....	v
ABSTRAK	ix
ABSTRACT	xi
DAFTAR ISI.....	xiii
DAFTAR TABEL	xvii
DAFTAR GAMBAR	xix
DAFTAR NOTASI	xxiii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian	4
1.4 Manfaat	4
1.5 Batasan Masalah	5
BAB 2 TINJAUAN PUSTAKA	7
2.1 Kajian Penelitian Terdahulu.....	7
2.2 Kajian Pustaka	11
2.2.1 <i>Machine vision</i>	11
2.2.2 <i>Internet of Things (IoT)</i>	13
2.2.3 <i>Edge Computing</i>	14
2.2.4 <i>You Only Look Once – YOLOv8</i>	15
2.2.5 <i>OpenCV</i>	22
2.3 <i>Hardware</i>	23
2.3.1 <i>ESP 32</i>	24
2.3.2 <i>TP-Link Tapo C212 (Camera)</i>	25
2.3.3 <i>Power Supply 10A/12V</i>	27
2.3.4 <i>MCB 4A</i>	28
2.3.5 <i>Laptop</i>	28
2.3.6 <i>DC Stepdown 5V 3A</i>	30
2.3.7 <i>LCD I2C</i>	31
2.3.8 <i>Relay 4 Channel Optocoupler</i>	32
2.3.9 <i>Sensor LDR (Light Dependent Resistor)</i>	33

2.3.10	Router <i>wireless</i>	33
2.3.11	Kabel AWG 0.75 dan 1.5.....	34
2.3.12	Fuse 2A dan Terminal.....	35
2.4	<i>Software</i>	36
2.4.1	Visual Studio Code.....	37
2.4.2	SketchUp	38
2.4.3	Arduino IDE	40
2.4.4	Phyton.....	41
2.4.5	GitHub	42
2.4.6	Hive MQ.....	43
2.4.7	CloudFlare Tunnels	44
BAB 3 METODOLOGI PENELITIAN		47
3.1	Konsep Penelitian.....	47
3.1.1	Diagram Blok Sistem	47
3.1.2	<i>Flowchart Kerja Sistem</i>	49
3.1.3	<i>Flowchart Backend Sistem</i>	51
3.1.4	<i>Flowchart Frontend Sistem</i>	53
3.1.5	<i>Flowchart Illegal Activity</i>	55
3.2	Tahapan Penelitian	57
3.3	Identifikasi Masalah	58
3.4	Studi Literatur	58
3.5	Analisis Kebutuhan Sistem	59
3.6	Pemilihan Komponen Sistem	59
3.7	Perancangan <i>Hardware</i>	60
3.8	Perancangan <i>Software</i>	60
3.9	Implementasi Sistem dan Pengujian	60
3.10	Evaluasi Keberhasilan Pengujian	61
3.11	Analisis Data	61
3.12	Ekstensi YOLOv8	61
3.13	Perancangan dan Implementasi	67
3.13.1	Penempatan <i>Hardware</i>	69
3.13.2	Perancangan <i>Software</i>	73
BAB 4 HASIL DAN PEMBAHASAN.....		77
4.1	Hasil Perancangan <i>Hardware</i>	77
4.1.1	Hasil Pembuatan <i>Printed Circuit Board</i> (PCB).....	77
4.1.2	Hasil Pengujian Sensor <i>Camera</i> (TAPO 212)	79

4.1.3	Pengujian Sensor RTC (<i>Real Time Clock</i>) DS3231M.....	81
4.1.4	Pengujian <i>Relay 4 Channel</i> Optocouler untuk 3 <i>Speed</i> Kipas	83
4.1.5	Pengujian <i>Relay 1 Channel</i> 5V (Include in Extension Microcontroller) 86	
4.1.6	Pengujian <i>LCD i2c</i> Terhadap Beberapa Input dan Output.....	87
4.1.7	Pengujian <i>LCD i2c</i> Terhadap Beberapa Input dan Output.....	89
4.1.8	Pengujian Jangkauan Wireless Web dan Sistem	91
4.2	Hasil Perakitan <i>Hardware</i> Sistem	93
4.3	Hasil Perancangan dan Konektivitas Software.....	96
4.3.1	Tampilan Data dan Status.....	97
4.3.2	Tampilan Kontrol Sistem	98
4.3.3	Tampilan System Log	99
4.4	Hasil Pelatihan Model YOLOv8m.....	100
4.4.1	Hasil Training Dataset.....	104
4.4.2	Pengujian Model YOLOv8 terhadap Seluruh Kelas Objek	109
4.5	Hasil Pengujian <i>Before-After</i> Menggunakan Metode YOLOv8	116
4.5.1	Hasil Integrasi Metode YOLOv8 pada Sistem.....	117
4.5.2	Perbandingan Dengan atau Tidak Menggunakan Metode YOLOv8.....	118
4.6	Hasil Perbandingan <i>Before-After</i> Terhadap Pengeluaran Daya Masjid.....	120
4.6.1	Hasil Pengukuran Sebelum Implementasi Sistem.....	121
4.6.2	Hasil Pengukuran Sesudah Implementasi Sistem	123
BAB 5 PENUTUP.....		127
5.1	Kesimpulan	127
5.2	Saran	128
DAFTAR PUSTAKA.....		131
LAMPIRAN 1 FORMULIR IZIN AMBIL DATA DAN IMPLEMENTASI ALAT DI MASJID BAHRUL ULUM PPNS		135
LAMPIRAN 2 SURAT PERNYATAAN Pengerjaan Tugas Akhir		139
BIODATA MAHASISWA.....		143

(halaman ini sengaja dikosongkan)

DAFTAR TABEL

Tabel 2.1. Penelitian terdahulu.....	7
Tabel 2.2. Spesifikasi ESP 32.....	24
Tabel 2.3. Tapo C-212 Camera.....	26
Tabel 2.4. Spesifikasi lengkap yang disarankan untuk dimiliki.....	29
Tabel 4.1. Hasil Pengujian Sensor Kamera (Deteksi Seluruh Tubuh).....	80
Tabel 4.2. Konektivitas Sensor RTC.....	82
Tabel 4.3. Data Hasil Pengujian Sensor RTC Terhadap Ketepatan Waktu	82
Tabel 4.4. Konektivitas <i>Relay 4 Channel</i>	84
Tabel 4.5. Pengujian Konektivitas <i>Relay</i> Terhadap <i>Speed</i> Kipas	85
Tabel 4.6. Pengujian Konektivitas <i>Relay</i> Terhadap Lampu.....	86
Tabel 4.7. Data Pengujian <i>Relay</i> Lampu Terhadap Jumlah Jamaah.....	87
Tabel 4.8. Konektivitas LCD I2C	88
Tabel 4.9. Hasil Pengujian dan Konektivitas LCD dengan <i>Output</i>	88
Tabel 4.10. Konektivitas Pin Sensor LDR.....	89
Tabel 4.11. Hasil Pengujian Sensor LDR di Masjid Bahrul Ulum PPNS.....	90
Tabel 4.12. Data Hasil Pengujian Jarak Stabil untuk Sistem <i>Wireless</i>	92
Tabel 4.13. Proporsi Pembagian Dataset.....	101
Tabel 4.14. Parameter Pelatihan YOLOv8m.....	103
Tabel 4.15. Pengujian Kelas Jamaah.....	110
Tabel 4.16. Pengujian Kelas <i>Illegal Activity</i> Tidur di Masjid	112
Tabel 4.17. Pengujian Kelas <i>Illegal Activity</i> Tidur di Masjid	114
Tabel 4.18. Perbandingan Penggunaan Metode YOLO	120
Tabel 4.19. Pengukuran Daya Sebelum Adanya Sistem	121
Tabel 4.20. Pengukuran Daya Sesudah Adanya Sistem	124

(halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

Gambar 2.1. Cara Kerja <i>Machine Vision</i>	12
Gambar 2.2. Cara Kerja IoT.....	13
Gambar 2.3. <i>Edge Computing</i>	14
Gambar 2.4. YOLOv8 <i>Model Size</i>	17
Gambar 2.5. FPS <i>Performance of YOLOv5 and YOLOv8</i>	18
Gambar 2.6. Alur dari OpenCV	23
Gambar 2.7. ESP32	24
Gambar 2.8. Tapo C-212	26
Gambar 2.9. <i>Power Supply</i> 12V 10A.....	27
Gambar 2.10. MCB 4A	28
Gambar 2.11. Laptop <i>IdeaPad Slim 3</i> 14ABR8.....	29
Gambar 2.12. DC <i>Stepdown</i> 5V 3	30
Gambar 2.13. LCD <i>Backlight With I2C</i>	31
Gambar 2.14. <i>Relay Module</i> 4 Channel Optocoupler	32
Gambar 2.15. Sensor LDR (<i>Light Dependent Resistor</i>)	33
Gambar 2.16. Router Penguat Sinyal Antar ESP32	34
Gambar 2.17. Kabel AWG	35
Gambar 2.18. Fuse 2A dan Terminal.....	36
Gambar 2.19. Tampilan Visual Studio Code.....	37
Gambar 2.20. <i>Flash Framework Python</i>	38
Gambar 2.21. Software SketchUp.....	39
Gambar 2.22. Tampilan Coding Arduino IDE	40
Gambar 2.23. Tampilan Coding Software Python	42
Gambar 2.24. Tampilan Data File GitHub	43
Gambar 2.25. Tampilan Software HiveMQ	44
Gambar 2.26. Software Cloudflare Tunnels.....	45
Gambar 3.1. Diagram Blok Sistem.....	48
Gambar 3.2. Diagram Blok Sistem <i>Wireless</i>	49
Gambar 3.3. <i>Flowchart</i> Kerja Sistem	50
Gambar 3.4. <i>Flowchart Backend System</i>	52

Gambar 3.5. <i>Flowchart Frontend Sistem</i>	54
Gambar 3.6. <i>Flowchart Illegal Activity</i>	56
Gambar 3.7. Diagram Alir Tahapan Penelitian	57
Gambar 3.8. Arsitektur YOLO	62
Gambar 3.9. Hasil <i>Processing</i> Ekstensi YOLOv8.....	65
Gambar 3.10. Diagram Blok Metode YOLO	65
Gambar 3.11. Perancangan <i>Hardware</i> dan <i>Software</i>	68
Gambar 3.12. Denah Implementasi <i>Hardware</i> Lantai 1 Masjid Bahrul Ulum.....	70
Gambar 3.13. Denah Implementasi <i>Hardware</i> Lantai 2 Masjid Bahrul Ulum.....	71
Gambar 3.14. Tampilan <i>Monitoring</i>	74
Gambar 3.15. Tampilan <i>Monitoring</i> dan Kontrol.....	75
Gambar 3.16. Tampilan Sistem Log.....	76
Gambar 4.1. <i>Routing</i> Board Bottom PCB (<i>Extention</i> ESP32).....	78
Gambar 4.2. <i>Routing</i> Board Top & Bottom PCB	78
Gambar 4.3. Hasil PCB yang Telah Dicitak dan Dirakit	79
Gambar 4.4. Konektivitas pada Terminal Untuk RSTP/IP pada CCTV	80
Gambar 4.5. Pengujian Pembacaan Sensor Kamera.....	80
Gambar 4.6. Pengujian Sensor RTC Terhadap Ketepatan Waktu.....	82
Gambar 4.7. Tampilan Terminal VS Code Terhadap Hasil Pembacaan <i>Speed</i> Kipas pada <i>Relay</i>	83
Gambar 4.8. Pengujian 3 <i>Channel</i> untuk Masing-Masing <i>Speed</i> kipas	84
Gambar 4.9. Pengukuran Tegangan dari <i>Relay</i> 5v untuk lampu	86
Gambar 4.10. Uji Coba Pembacaan LCD terhadap <i>Input</i> dan <i>Output</i>	88
Gambar 4.11. Hasil Pengujian Sensor LDR	90
Gambar 4.12. Pengukuran Jarak Efektif Untuk Sistem.....	91
Gambar 4.13. Desain Awal Perancangan Sistem.....	93
Gambar 4.14. Pemasangan CCTV Router, dan seluruh Sistem Lainnya	94
Gambar 4.15. Implementasi <i>Hardware</i> dan Pengujian <i>Relay</i> Lampu pada Malam Hari	95
Gambar 4.16. Konektivitas <i>Software</i> , HTTP, dan Supabase dengan Kamera.....	96
Gambar 4.17. Tampilan Bagian Data Sistem.....	97
Gambar 4.18. Tampilan Kontrol Sistem.....	98

Gambar 4.19. Tampilan <i>System Log</i>	100
Gambar 4.20. Pembagian Dataset dari Hasil <i>Labelling</i>	100
Gambar 4.21. <i>Result Hasil Training</i>	104
Gambar 4.22. <i>Confusion Matrix</i>	108
Gambar 4.23 Percobaan <i>Training</i> 8.948 Dataset dari 3 <i>Class</i> , dengan YOLOv8	117
Gambar 4.24. Konektivitas Hasil <i>Training</i> YOLOv8 dengan Web dan Kamera.	118
Gambar 4.25. <i>Object Detection</i> Tanpa Metode YOLO	119
Gambar 4.26. Tampilan Monitor CCTV	119

(halaman ini sengaja dikosongkan)

DAFTAR NOTASI

X	: Data masukan (<i>input</i>), umumnya berupa citra atau <i>feature map</i>
Y	: <i>Feature map</i> hasil operasi konvolusi
P	: <i>Feature map</i> hasil proses <i>pooling</i>
W	: Bobot atau kernel/filter konvolusi
b	: Bias pada neuron atau filter
$f($	: Fungsi aktivasi
Z	: Nilai aktivasi linear sebelum fungsi aktivasi
y'	: Output prediksi jaringan
$f(x)$	: Fungsi aktivasi non-linear (<i>ReLU</i> atau <i>Sigmoid</i>)
$\sigma(x)$	: Fungsi aktivasi <i>sigmoid</i>
Z	: Nilai aktivasi linear sebelum fungsi aktivasi
b_x, b_y	: Koordinat pusat <i>bounding box</i>
b_w, b_h	: Lebar dan tinggi <i>bounding box</i>
t_x, t_y, t_w, t_h	: Output jaringan
c_x, c_y	: Posisi <i>grid sel</i>
p_w, p_h	: Ukuran <i>anchor box</i>
IoU	: <i>Intersection over Union</i> , ukuran kesesuaian <i>bounding box</i>
$P (Object)$	: Probabilitas keberadaan objek dalam <i>bounding box</i>
<i>Confidence</i>	: Nilai keyakinan deteksi objek
θ	: Nilai ambang batas (<i>threshold</i>) keputusan <i>system</i>
$\mathcal{L}$	: Fungsi loss total pada model YOLO
$\mathcal{L}_{box}$	: <i>Loss</i> regresi <i>bounding box</i>
$\mathcal{L}_{obj}$	: <i>Loss</i> deteksi keberadaan objek

$\mathcal{L}_{cls}$: *Loss* klasifikasi objek

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi otomasi dan kecerdasan buatan telah membawa perubahan signifikan dalam pengelolaan fasilitas bangunan, khususnya yang berkaitan dengan efisiensi energi, keamanan, dan kenyamanan pengguna. Salah satu teknologi yang berkembang pesat adalah *Internet of Things* (IoT), yang memungkinkan proses pemantauan dan pengendalian sistem dilakukan secara terintegrasi melalui jaringan internet. Selain itu, kemajuan teknologi *machine vision* memungkinkan sistem memahami kondisi lingkungan berdasarkan citra visual, sehingga pengambilan keputusan tidak hanya bergantung pada kendali manual atau sensor konvensional, tetapi juga pada analisis aktivitas manusia di dalam suatu ruang secara *real-time*. Melalui pendekatan *object detection* berbasis *deep learning*, sistem mampu mengenali keberadaan manusia serta aktivitas tertentu yang terjadi di suatu area pengamatan. Salah satu algoritma yang banyak digunakan pada sistem deteksi objek modern adalah *You Only Look Once* (YOLO), yang mampu melakukan deteksi objek secara cepat dan akurat dalam satu proses komputasi. Kemampuan YOLO dalam mendeteksi objek manusia pada berbagai kondisi pencahayaan menjadikannya relevan untuk diterapkan pada sistem pengendalian fasilitas berbasis aktivitas pengguna.

Penerapan teknologi otomasi berbasis IoT dan *machine vision* kini tidak hanya terbatas pada sektor industri dan perkantoran, tetapi juga mulai diterapkan pada fasilitas publik, termasuk tempat ibadah. Dalam konteks masjid, sistem otomasi berperan penting dalam mendukung kenyamanan jamaah, efisiensi penggunaan energi, serta ketertiban pengelolaan fasilitas tanpa mengganggu aktivitas ibadah. Masjid Bahrul Ulum Politeknik Perkapalan Negeri Surabaya (PPNS) merupakan salah satu fasilitas kampus yang telah mendukung konsep keberlanjutan melalui program *UI Green Metric* dengan pemanfaatan Pembangkit Listrik Tenaga Surya (PLTS). Upaya ini menunjukkan komitmen institusi dalam efisiensi energi dan pelestarian lingkungan. Namun, berdasarkan hasil observasi lapangan dan diskusi dengan pengurus masjid, masih terdapat tantangan dalam

pengelolaan fasilitas, khususnya pada lantai dua yang diperuntukkan bagi jamaah wanita. Fasilitas penerangan dan kipas angin di area tersebut masih dioperasikan secara manual menggunakan saklar konvensional.

Dalam praktiknya, pengendalian fasilitas di lantai dua tidak selalu dapat disesuaikan dengan kondisi aktual ruang karena keterbatasan akses dan waktu pengawasan oleh petugas marbot maupun pengurus takmir. Area jamaah wanita memiliki batasan privasi sehingga tidak memungkinkan adanya pengawasan langsung secara terus-menerus. Akibatnya, pengoperasian fasilitas sering kali tidak responsif terhadap perubahan kondisi penggunaan ruang, terutama ketika area sudah tidak digunakan. Kondisi ini berpotensi menyebabkan penggunaan energi yang kurang efisien serta menurunkan efektivitas pengelolaan fasilitas masjid. Selain aspek pengendalian fasilitas, berdasarkan hasil pengamatan lapangan dan wawancara dengan beberapa pengurus masjid juga ditemukan permasalahan terkait ketertiban penggunaan ruang, khususnya adanya indikasi pembuangan sampah di area saf jamaah wanita. Keterbatasan pengawasan langsung menyebabkan perilaku tersebut sulit terpantau secara konsisten, sehingga diperlukan sistem pendukung yang mampu membantu proses pengawasan secara otomatis dan objektif tanpa melanggar privasi jamaah.

Selain permasalahan tersebut, pengurus masjid juga menyampaikan perlunya upaya pencegahan terhadap potensi pelanggaran pemanfaatan ruang, khususnya terkait aktivitas tidur atau menginap di area masjid di luar waktu operasional yang telah ditetapkan oleh peraturan kampus. Batasan jam malam yang mewajibkan area kampus steril setelah pukul 21.00 menjadikan pengawasan aktivitas ruang sebagai aspek penting dalam menjaga keamanan, ketertiban, serta fungsi masjid sebagai tempat ibadah. Dalam konteks ini, deteksi indikasi aktivitas tidur tidak dimaksudkan sebagai bentuk penilaian terhadap individu, melainkan sebagai langkah preventif berbasis sistem untuk membantu pengurus masjid dalam mengidentifikasi kondisi ruang yang tidak sesuai peruntukannya, tanpa melakukan pengawasan langsung dan tanpa mengganggu privasi jamaah. Berdasarkan kondisi tersebut, diperlukan pengembangan sistem kontrol cerdas berbasis IoT yang dilengkapi dengan kemampuan deteksi objek menggunakan metode YOLO. Sistem ini dirancang untuk memiliki tiga fokus utama, yaitu mendeteksi keberadaan

jamaah wanita sebagai dasar pengendalian otomatis fasilitas listrik berupa lampu dan kipas angin, mendeteksi indikasi aktivitas pembuangan sampah di area shaf lantai dua sebagai bagian dari sistem pengawasan kebersihan, serta mendeteksi indikasi aktivitas tidur atau menginap sebagai upaya pencegahan pelanggaran pemanfaatan ruang berdasarkan batasan waktu operasional. Kamera digunakan sebagai sensor utama untuk mendeteksi objek dan aktivitas, sementara sensor pendukung seperti *Light Dependent Resistor* (LDR) dan *Real Time Clock* (RTC) digunakan untuk meningkatkan akurasi pengambilan keputusan sistem. Pengendalian sistem dilakukan secara terdistribusi menggunakan mikrokontroler ESP32 yang berkomunikasi secara nirkabel, sehingga tidak memerlukan penambahan jalur kabel antar lantai.

Pengembangan sistem ini diharapkan menjadi penerapan nyata keilmuan Teknik Otomasi dalam pengelolaan fasilitas publik, sekaligus mendukung visi PPNS sebagai institusi pendidikan vokasi yang mendorong penerapan teknologi cerdas yang aplikatif, kontekstual, dan beretika.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang di atas, maka didapatkan rumusan masalah sebagai berikut.

1. Bagaimana proses pengelolaan dataset yang optimal untuk sistem deteksi berbasis kamera menggunakan metode YOLO dalam mendukung pengawasan aktivitas jamaah serta indikasi *illegal action* (*Indication of a person littering* dan *illegal overnight stay*) dan pengendalian fasilitas listrik di Masjid Bahrul Ulum?
2. Bagaimana tingkat akurasi sistem dalam mendeteksi objek dan aktivitas, meliputi keberadaan dan jumlah jamaah wanita serta indikasi *illegal action* (*Indication of a person littering* dan *illegal overnight stay*), menggunakan metode YOLO sebagai dasar pengambilan keputusan pengendalian fasilitas listrik secara otomatis?
3. Seberapa efektif dan efisien sistem yang dikembangkan dalam mengoptimalkan pengelolaan fasilitas listrik, dari respons sistem *real-time*, serta keandalan pengawasan berbasis deteksi objek, dan perbandingan pengeluaran daya *before-after* dari implementasi ini?

1.3 Tujuan Penelitian

Ditinjau dari uraian rumusan masalah di atas, maka tujuan penelitian ini adalah sebagai berikut.

1. Mengelola dan menyusun dataset citra sesuai dengan tingkat dan batas indikasi perilaku dalam pengawasan aktivitas jamaah wanita, indikasi *illegal action* (*Indication of a person littering* dan *illegal overnight stay*), serta pengendalian fasilitas listrik di Masjid Bahrul Ulum.
2. Menganalisis tingkat akurasi sistem deteksi objek dan aktivitas menggunakan metode YOLO dari setiap hasil *epoch* dan *training* yang dilakukan, dalam mendeteksi keberadaan dan jumlah jamaah wanita serta indikasi *illegal action*.
3. Menguji kinerja sistem kontrol dan efektivitas pengawasan dalam hal keakuratan *real-time detection*, konektivitas antar mikrokontroller, serta membandingkan data terkait pengeluaran daya *before-after* dari implementasi sistem

1.4 Manfaat

Berdasarkan uraian tujuan penelitian di atas, maka manfaat yang diperoleh adalah sebagai berikut.

1. Bagi Institusi
Sebagai bentuk nyata penerapan konsep IoT dan sistem kendali mikrokontroller ke dalam kehidupan nyata, khususnya dalam pengelolaan fasilitas kampus. Serta mendukung inisiatif *Green Campus* dan *Smart Mosque*, yaitu konsep masjid cerdas yang memanfaatkan teknologi untuk efisiensi dan keberlanjutan.
2. Bagi Masyarakat
Sebagai bahan referensi bagi penelitian selanjutnya di teknologi *automation engineering* terutama di bidang IoT yang bisa diterapkan bahkan di tempat keagamaan.
3. Bagi Pengurus Masjid Bahrul Ulum PPNS
Sistem ini dapat membantu pengelolaan fasilitas lantai dua dengan lebih mudah dan efisien, tanpa harus naik ke area jamaah akhwat. Penggunaan lampu dan kipas menjadi lebih terkontrol, hemat energi, serta sesuai dengan aktivitas dan batasan jamaah.

1.5 Batasan Masalah

Adapun ruang lingkup dari penelitian ini agar tidak terlalu luas tinjauannya dan tidak menyimpang dari rumasan masalah penelitian adalah sebagai berikut.

1. Lokasi dan ruang lingkup penelitian dibatasi pada lantai dua Masjid Bahrul Ulum PPNS, yang merupakan area khusus jamaah akhwat.
2. Sensor kamera yang digunakan hanya sebatas pengolahan citra berdasarkan bentuk tubuh manusia, ketika saat menggunakan mukenah, namun dengan batasan area yang hanya dapat dilihat oleh kamera
3. Penelitian difokuskan pada pembuatan prototype *Hardware* dan *software* dengan fungsi utama pengendalian otomatis serta monitoring jarak jauh, bukan pada desain mekanis atau struktural masjid.
4. Koneksi yang digunakan harus stabil dan menggunakan komunikasi lokal untuk server dan ESP 32 dengan penguatan *signal router* di masjid
5. Meskipun memiliki kelebihan dapat mengidentifikasi *illegal action*, namun pada dasarnya sistem ini menggunakan metode *You Only Look Once* (YOLO) yang berfokus untuk mendeteksi keberadaan *Indication of a person littering*, *illegal overnight stay*, dan *human presence detection* saja.
6. Deteksi pose tubuh manusia pada sistem dibatasi pada pengenalan postur dan rangkaian gerak tertentu sebagai indikasi aktivitas, yang meliputi:
 - a. Pose berdiri, berjalan, dan ruku' aktif sebagai dasar *human presence detection* untuk pengendalian fasilitas listrik
 - b. Pose berbaring tanpa perubahan posisi signifikan dalam durasi beberapa detik sebagai indikasi *illegal overnight stay* yang digunakan semata-mata sebagai upaya preventif keamanan berdasarkan batasan waktu operasional
 - c. Serta pose rangkaian gerak berupa menjulurkan tangan ke arah lantai disertai perubahan arah pandangan ke kanan dan kiri, diikuti dengan keberadaan objek yang ditinggalkan dan tidak diambil kembali dalam kurun waktu beberapa detik sebagai indikasi aktivitas membuang sampah (*indication of a person littering*).

(halaman ini sengaja dikosongkan)

BAB 2

TINJAUAN PUSTAKA

2.1 Kajian Penelitian Terdahulu

Pada tabel 2.1 berikut, merupakan rujukan dari beberapa penelitian sebelumnya, sehingga diharapkan dapat memberikan pandangan secara luas dan mendalam terhadap topik penelitian tugas akhir ini terkait implementasi yang akan diterapkan di Masjid Bahrul Ulum:

Tabel 2.1. Penelitian terdahulu

Nama dan Tahun Penelitian	Judul Penelitian	Hasil Penelitian
Utomo, I. T. <i>et al.</i> (2024)	<i>An Intelligent System for Light and Air Conditioner Control Using YOLOv8</i>	Beliau mengembangkan dan mengevaluasi sistem yang berbasis YOLOv8 untuk mendeteksi keberadaan manusia dan mengoptimalkan penggunaan listrik seperti lampu dan AC secara otomatis. Dan berdasarkan deteksi tersebut, menunjukkan akurasi tinggi dan potensi penghematan energi pada lingkungan nyata dalam lingkup ruang kelas.
K. Wijayanti <i>et al.</i> (2025)	<i>Non-Intrusive Real-time Tourist Crowd Monitoring Using YOLOv8</i>	Dikembangkan dengan pendekatan visi komputer berbasis YOLOv8 yang mampu mendeteksi dan menghitung jumlah individu secara <i>real-time</i> tanpa melakukan identifikasi personal, sehingga menjaga

Nama dan Tahun Penelitian	Judul Penelitian	Hasil Penelitian
		privasi pengguna. Dengan menekankan keunggulan YOLOv8 dalam mengekstraksi fitur spasial manusia pada berbagai kondisi pencahayaan dan kepadatan, serta menunjukkan bahwa data hasil deteksi dapat dimanfaatkan sebagai dasar pengambilan keputusan otomatis pada sistem manajemen ruang.
Ardiansyah <i>et al.</i> (2022)	<i>Development of Crowd Detection Warning System based on Deep Convolutional Neural Network using CCTV</i>	Pada perancangan <i>system</i> tersebut Adalah untuk mendeteksi kerumunan menggunakan CCTV dan algoritma YOLOv3-Tiny pada perangkat <i>edge</i> (Jetson Nano). Hasilnya sistem mampu mendeteksi kerumunan manusia secara <i>real-time</i> dan memberi peringatan. Relevan untuk deteksi keberadaan manusia di zona dan area yang Anda rancang.
Gawande, U. <i>et al.</i> (2024)	<i>Novel Person Detection and Suspicious Activity Recognition Using</i>	Penelitian ini mengembangkan sistem deteksi manusia dan

Nama dan Tahun Penelitian	Judul Penelitian	Hasil Penelitian
	<i>Enhanced YOLOv5 and Motion Feature Map</i>	pengenalan aktivitas mencurigakan menggunakan metode YOLOv5 yang ditingkatkan serta motion feature map. Sistem dirancang untuk melakukan pengawasan berbasis video. Hasil penelitian menunjukkan bahwa kombinasi deteksi objek dan informasi gerakan mampu meningkatkan kemampuan sistem dalam melakukan pengawasan.
Chaudhari, Xiao, Cheng, & Li (2024)	<i>Fundamentals, Algorithms, and Technologies of Occupancy Detection for Smart buildings Using IoT Sensors</i>	Meskipun berupa paper ilmiah namun berisi mengenai tinjauan yang menelaah berbagai metode deteksi okupansi menggunakan sensor IoT dan metode ML/DL serta efektif digunakan sebagai dasar pengambilan keputusan otomatis pada sistem pencahayaan dan pengendalian energi.

Berdasarkan penelitian jurnal pertama menunjukkan bahwa sistem pengendalian lampu dan pendingin ruangan berbasis YOLOv8 mampu mendeteksi keberadaan manusia secara real-time sehingga penggunaan energi listrik menjadi lebih efisien. Sistem tersebut memiliki tingkat akurasi yang tinggi dalam

mendeteksi okupansi ruangan, namun implementasinya masih berfokus pada pengendalian lampu dan AC di lingkungan ruang kelas serta belum mencakup pengawasan aktivitas manusia maupun integrasi dengan sistem monitoring berbasis IoT pada fasilitas publik. Pada jurnal kedua, penelitian mengembangkan sistem pemantauan kerumunan wisatawan secara real-time menggunakan YOLOv8 tanpa melakukan identifikasi identitas individu sehingga privasi pengguna tetap terjaga. Penelitian ini berhasil menunjukkan kemampuan YOLOv8 dalam mendeteksi dan menghitung jumlah manusia pada berbagai kondisi pencahayaan dan kepadatan. Akan tetapi, penelitian tersebut masih terbatas pada proses monitoring kepadatan manusia dan belum dimanfaatkan sebagai sistem kendali otomatis terhadap fasilitas listrik maupun deteksi aktivitas tertentu. Selanjutnya, jurnal ketiga mengembangkan sistem peringatan kerumunan menggunakan algoritma YOLOv3-Tiny yang diimplementasikan pada perangkat *edge computing* berbasis Jetson Nano. Hasil penelitian menunjukkan bahwa sistem mampu mendeteksi kerumunan secara *real-time* serta memberikan peringatan ketika jumlah orang melebihi batas tertentu. Meskipun demikian, penelitian ini hanya berfokus pada deteksi kerumunan dan belum melakukan klasifikasi aktivitas manusia ataupun pengendalian perangkat listrik secara otomatis. Pada jurnal keempat, penelitian lebih menitikberatkan pada deteksi aktivitas manusia secara *real-time* menggunakan kamera pengawas berbasis YOLO. Sistem mampu mengenali berbagai pola aktivitas manusia dan melakukan pencatatan kejadian berdasarkan waktu sehingga mendukung proses pengawasan pada suatu area. Namun, penelitian tersebut belum mengintegrasikan hasil deteksi dengan sistem *Internet of Things* (IoT) sebagai pengambil keputusan untuk mengendalikan perangkat elektronik maupun melakukan monitoring fasilitas secara terpadu. Sementara itu, jurnal kelima memberikan tinjauan komprehensif mengenai berbagai metode deteksi okupansi pada smart building menggunakan sensor IoT serta metode *Machine Learning* dan *Deep Learning*. Penelitian tersebut menjelaskan bahwa deteksi keberadaan manusia dapat dimanfaatkan sebagai dasar pengambilan keputusan otomatis untuk sistem pencahayaan dan pengelolaan energi bangunan. Akan tetapi, penelitian tersebut masih bersifat studi literatur sehingga belum menghasilkan implementasi sistem

yang menggabungkan *computer vision*, *edge computing*, dan IoT dalam satu platform yang terintegrasi.

Berdasarkan kelima penelitian tersebut, dapat disimpulkan bahwa masing-masing penelitian telah berhasil menunjukkan keunggulan YOLO maupun teknologi IoT dalam mendukung sistem otomasi dan efisiensi energi. Namun demikian, belum terdapat penelitian yang mengintegrasikan deteksi keberadaan manusia, deteksi aktivitas seperti *illegal overnight stay* dan *indication of a person littering*, pengendalian fasilitas listrik secara otomatis, serta monitoring berbasis web dalam satu sistem yang diterapkan pada lingkungan masjid. Melalui penelitian ini, penulis mengusulkan implementasi sistem kontrol cerdas berbasis *Internet of Things* (IoT), *Edge Computing*, dan metode YOLOv8 yang mampu mendeteksi keberadaan manusia sekaligus mengidentifikasi aktivitas tertentu sebagai dasar pengambilan keputusan otomatis untuk pengendalian lampu dan kipas angin. Selain itu, sistem dilengkapi dengan dashboard web untuk monitoring dan kontrol secara real-time sehingga diharapkan mampu meningkatkan efisiensi penggunaan energi listrik, mendukung pengawasan aktivitas di area masjid, serta memberikan solusi yang lebih komprehensif dibandingkan penelitian-penelitian sebelumnya.

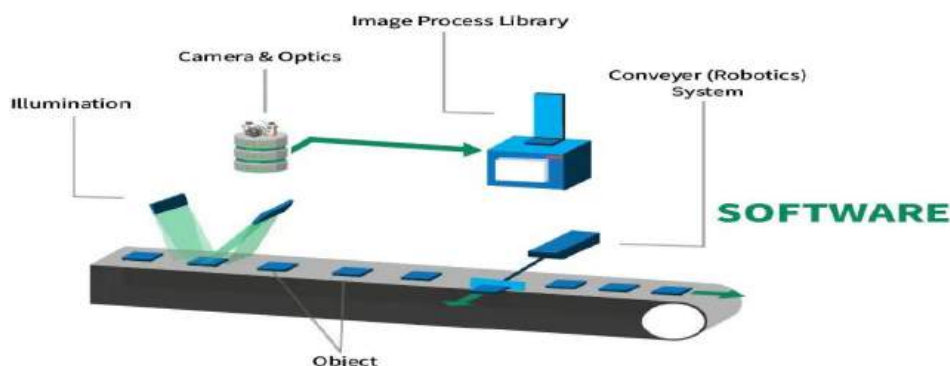
2.2 Kajian Pustaka

Dalam melakukan penelitian pada Tugas Akhir ini, penulis memerlukan beberapa dasar teori dan referensi ilmiah yang dapat mendukung serta memperkuat landasan dari topik yang diangkat. Kajian pustaka ini disusun untuk memberikan pemahaman yang lebih mendalam mengenai konsep-konsep utama yang berkaitan dengan penelitian. Beberapa dasar teori yang digunakan dalam penelitian tugas akhir ini meliputi spesifikasi dari komponen, *software* yang digunakan serta metode alat dan metode yang digunakan dalam penelitian ini meliputi hal-hal berikut:

2.2.1 *Machine vision*

Machine vision adalah teknologi canggih yang dirancang untuk meniru penglihatan dan persepsi manusia dalam konteks otomatisasi. Sistem ini mengintegrasikan kombinasi komponen perangkat keras dan perangkat lunak untuk menangkap dan menganalisis informasi visual dari lingkungan sekitar. Tujuan

utamanya adalah untuk memungkinkan mesin membuat keputusan berdasarkan data visual yang mereka peroleh.



Gambar 2.1. Cara Kerja *Machine Vision*

Sumber: <https://www.geeksforgeeks.org/electrical-engineering/what-is-machine-vision/>

Sesuai dengan gambar 2.1 diatas, cara kerja *machine vision* adalah Sistem penglihatan mesin yang beroperasi berdasarkan prinsip-prinsip dasar berikut:

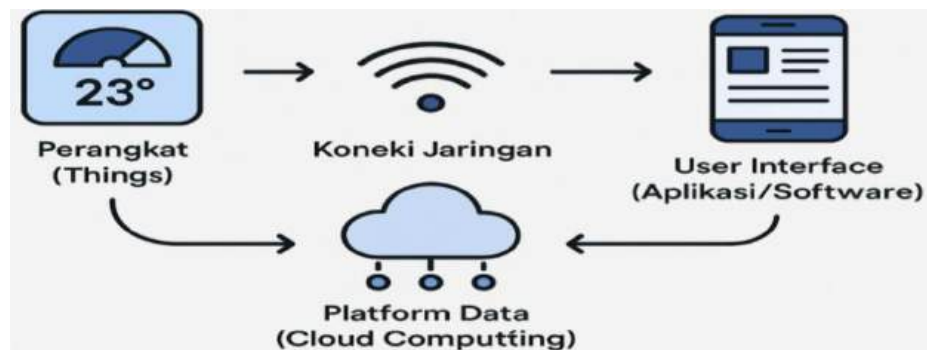
1. Pengambilan Gambar: Proses dimulai dengan pengambilan gambar objek atau pemandangan menggunakan kamera. Pilihan jenis kamera dan parameter seperti waktu pencahayaan dan kecepatan bingkai sangat penting.
2. Pencahayaan: Pencahayaan yang tepat sangat penting untuk meningkatkan kualitas gambar. Sumber pencahayaan disesuaikan untuk memberikan kondisi pencahayaan yang seragam dan optimal untuk inspeksi.
3. Praproses Citra: Citra mentah seringkali mengandung *noise* dan mungkin memerlukan praproses.
4. Pengenalan Pola: Sistem ini menggunakan berbagai algoritma untuk pengenalan pola yang membantu mengidentifikasi objek atau karakteristik spesifik dalam gambar.
5. Pengambilan Keputusan: Berdasarkan hasil analisis, sistem visi mesin membuat keputusan terkait objek yang diperiksa, seperti menerima atau menolak produk dalam proses manufaktur.
6. Komunikasi: Sistem ini mengkomunikasikan temuannya, seringkali secara real-time, kepada komponen lain dari sistem otomatisasi.

Dalam implementasinya, *machine vision* tidak bekerja secara terpisah, tetapi dikombinasikan dengan sensor pendukung seperti *Light Dependent Resistor* (LDR) dan *Real Time Clock* (RTC). Sensor LDR digunakan untuk memberikan informasi mengenai kondisi pencahayaan alami ruangan, sedangkan RTC berperan

sebagai pengatur waktu tunda pemadaman fasilitas. Namun demikian, keputusan utama tetap berada pada hasil deteksi *machine vision*, karena informasi visual dianggap paling representatif dalam menggambarkan keberadaan dan aktivitas manusia di dalam ruangan.

2.2.2 *Internet of Things (IoT)*

Web monitoring dan kontrol sistem merupakan salah satu pendekatan yang banyak digunakan dalam pengembangan sistem otomasi modern, khususnya pada sistem yang terintegrasi dengan konsep *Internet of Things (IoT)*. Melalui teknologi berbasis web, suatu sistem tidak hanya mampu bekerja secara otomatis, tetapi juga dapat dipantau dan dikendalikan dari jarak jauh secara *real-time*. Hal ini memberikan fleksibilitas tinggi bagi pengguna dalam mengawasi kondisi sistem serta melakukan intervensi manual apabila diperlukan. salah satu karakteristik utama sistem IoT adalah kemampuannya untuk menyediakan akses data dan kontrol melalui antarmuka berbasis internet.



Gambar 2.2. Cara Kerja IoT

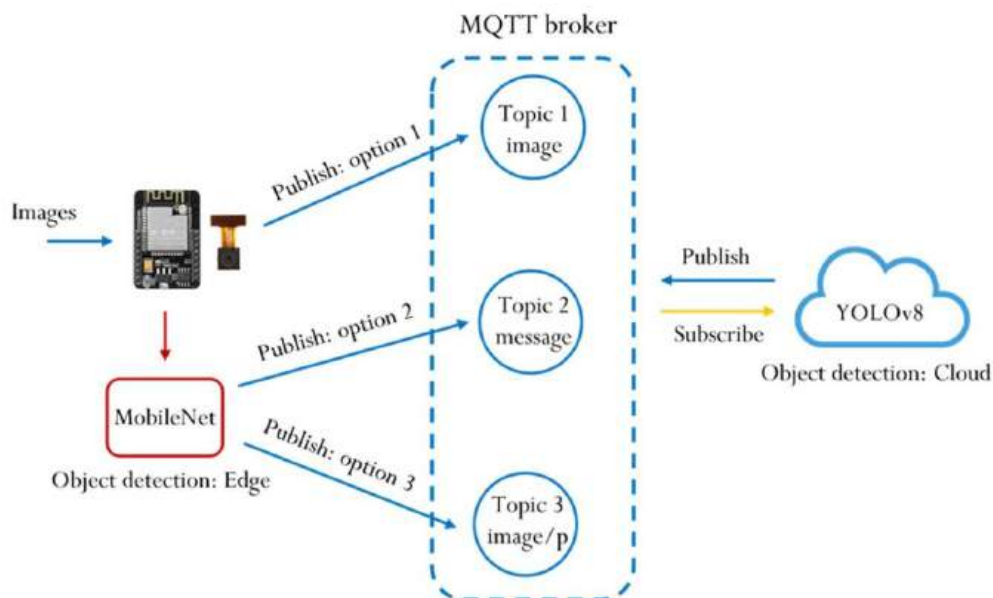
Sumber: <https://upttik.undiksha.ac.id/internet-of-things/>

Pada gambar 2.2, web berperan sebagai lapisan aplikasi (*application layer*) yang menjembatani interaksi antara pengguna dan perangkat fisik di lapangan. Dengan adanya web monitoring, data yang diperoleh dari sensor dan hasil pemrosesan sistem dapat disajikan dalam bentuk informasi yang mudah dipahami oleh pengguna tanpa harus berinteraksi langsung dengan perangkat keras. Menurut Gubbi *et al.* (2013). Selain aspek fungsional, web monitoring juga berkontribusi pada peningkatan efisiensi operasional. Dengan adanya informasi kondisi fasilitas secara *real-time*, pengurus masjid tidak perlu melakukan pengecekan langsung ke lokasi, terutama pada area lantai dua yang memiliki keterbatasan akses. Hal ini mendukung prinsip efisiensi energi dan kemudahan pengelolaan fasilitas publik,

sebagaimana dijelaskan oleh Buckman *et al.* (2014) bahwa sistem monitoring terpusat merupakan elemen penting dalam manajemen bangunan cerdas (*smart building*).

2.2.3 Edge Computing

Edge Computing merupakan pendekatan komputasi di mana proses pengolahan data dilakukan sedekat mungkin dengan sumber data, tanpa harus selalu bergantung penuh pada server pusat atau layanan *cloud*. Dalam konteks sistem, *Edge Computing* diterapkan dengan menempatkan proses deteksi manusia dan pengolahan citra langsung pada perangkat lokal (seperti laptop) yang terhubung ke kamera di area masjid. Pendekatan ini dipilih karena sistem membutuhkan respons yang cepat dan stabil, terutama dalam pengambilan keputusan untuk mengaktifkan atau mematikan fasilitas seperti lampu dan kipas secara otomatis.



Gambar 2.3. Edge Computing

Sumber: <https://www.mdpi.com/1424-8220/25/6/1656>

Sesuai dengan arsitektur pada Gambar 2.3, aliran data dan proses deteksi objek dikelola dan dijembatani menggunakan MQTT broker. Secara umum, sistem ini bekerja melalui pembagian tugas pemrosesan sebagai berikut:

- Object detection: *Edge* (MobileNet): Citra (*Images*) yang ditangkap oleh perangkat kamera diproses secara lokal terlebih dahulu di *Edge* menggunakan model MobileNet. Perangkat lokal ini kemudian memublikasikan (*Publish*)

data ke *MQTT broker* melalui beberapa opsi pengiriman: mengirimkan citra mentah ke *Topic 1*, mengirimkan pesan/hasil deteksi ke *Topic 2*, atau mengirimkan citra yang sudah diproses ke *Topic 3*.

- Object detection, Cloud (YOLOv8): Layanan *cloud* dengan model YOLOv8 dapat digunakan untuk pemrosesan tingkat lanjut. Sistem *cloud* ini saling bertukar informasi dengan cara berlangganan (*Subscribe*) untuk mengambil data dari *MQTT broker*, dan memublikasikan (*Publish*) kembali hasil deteksinya.

Dengan pemrosesan awal yang dilakukan menggunakan *Edge Computing* (MobileNet), keputusan otomatisasi di masjid dapat dilakukan dengan sangat cepat tanpa terhambat penundaan jaringan (*network delay*). Menurut Gubbi et al. (2013), *Edge Computing* sangat efektif diterapkan pada sistem IoT yang membutuhkan latensi rendah dan keandalan tinggi, seperti sistem otomasi bangunan dan ruang publik.

2.2.4 You Only Look Once – YOLOv8

Metode *You Only Look Once* (YOLO) merupakan algoritma *object detection* berbasis *deep learning* yang dirancang untuk melakukan deteksi objek secara *real-time* dengan akurasi tinggi, efisiensi komputasi, dan kemampuan untuk mendeteksi banyak objek sekaligus dalam satu langkah pemrosesan. Hal ini sesuai dengan temuan Wang et al. (2023) yang melaporkan bahwa YOLOv8 mampu mencapai nilai *mean Average Precision* (mAP) di atas 0.85 untuk deteksi manusia secara *real-time* di *environment indoor*, sekaligus mempertahankan waktu pemrosesan di bawah 30 ms per frame pada GPU standar. Temuan ini menunjukkan bahwa YOLO cocok untuk aplikasi berbasis kamera yang mengharuskan deteksi instan, seperti otomatisasi lampu dan kipas angin serta kemampuan untuk mendeteksi aktivitas ilegal sebagai upaya preventif pada area ibadah. Sistem YOLO terdiri atas tahapan berikut yang secara konseptual paralel dengan kebutuhan TA ini:

1. Input Layer

Input berupa citra/frame video dari kamera yang dipasang di area lantai dua Masjid Bahrul Ulum, dalam format resolusi tertentu (misalnya 640×640).

Kamera *day-night* memastikan citra yang diambil tetap informatif meskipun kondisi pencahayaan rendah.

2. *Feature Extraction Backbone*

YOLOv8 memanfaatkan jaringan CNN sebagai *backbone* untuk mengekstraksi fitur visual penting. Proses ini mengidentifikasi pola yang menjadi ciri objek seperti manusia, gerakan mencurigakan, serta objek lain yang menjadi indikasi aktivitas ilegal. Penelitian Nolan *et al.* (2022) menyatakan bahwa penggunaan *backbone* CNN dalam YOLOv8 memungkinkan *extraction* fitur spasial yang kompleks tanpa memerlukan *manual feature engineering*.

3. *Neck Layer (Feature Pyramid)*

Fitur yang diekstraksi kemudian diolah melalui lapisan *neck* untuk menggabungkan informasi dari berbagai skala, sehingga YOLOv8 mampu mendeteksi objek besar maupun kecil secara simultan. Tahapan ini penting untuk memastikan bahwa jamaah yang berada di area berbeda dalam frame, serta aktivitas dengan ukuran objek berbeda, dapat dikenali dengan akurat.

4. *Detection Head*

Pada lapisan *detection head*, YOLOv8 melakukan prediksi posisi objek (*bounding box*), kelas objek, serta *confidence score* dalam satu langkah. Informasi ini selanjutnya digunakan sistem untuk menentukan apakah objek yang dideteksi adalah jamaah, obyek aktivitas ilegal (misalnya seseorang tertidur, membuang sampah, atau pergerakan yang mencurigakan).

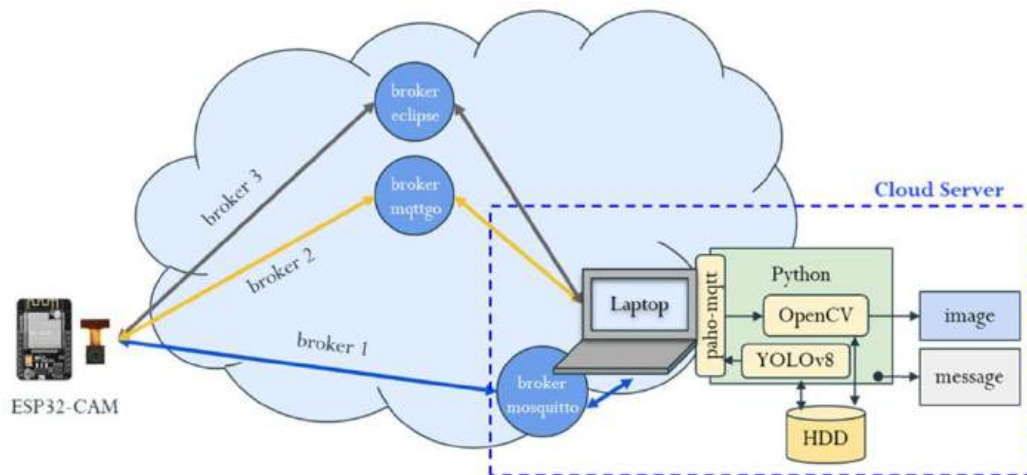
5. *Output dan Integrasi dengan Sistem*

Sesuai dengan implementasi saya, output deteksi YOLO ini berupa:

- Posisi objek (koordinat *bounding box*)
- Kelas objek (*person*, *Indication of a person littering*, dan *illegal overnight stay*)
- *Confidence score*

Pada Gambar 2.4 di bawah, diilustrasikan arsitektur sistem deteksi objek di mana perangkat keras ESP32-CAM terhubung dengan *Cloud Server* melalui protokol MQTT. Meskipun model YOLOv8 dikenal memiliki performa tinggi—dengan nilai *mean Average Precision* (mAP_{val50-95}) yang bervariasi antara 37.3% hingga 53.9% pada *benchmark* COCO—model ini membutuhkan daya komputasi

yang memadai, sehingga pada arsitektur ini proses inferensinya diletakkan di sisi server.



Gambar 2.4. YOLOv8 Model Size

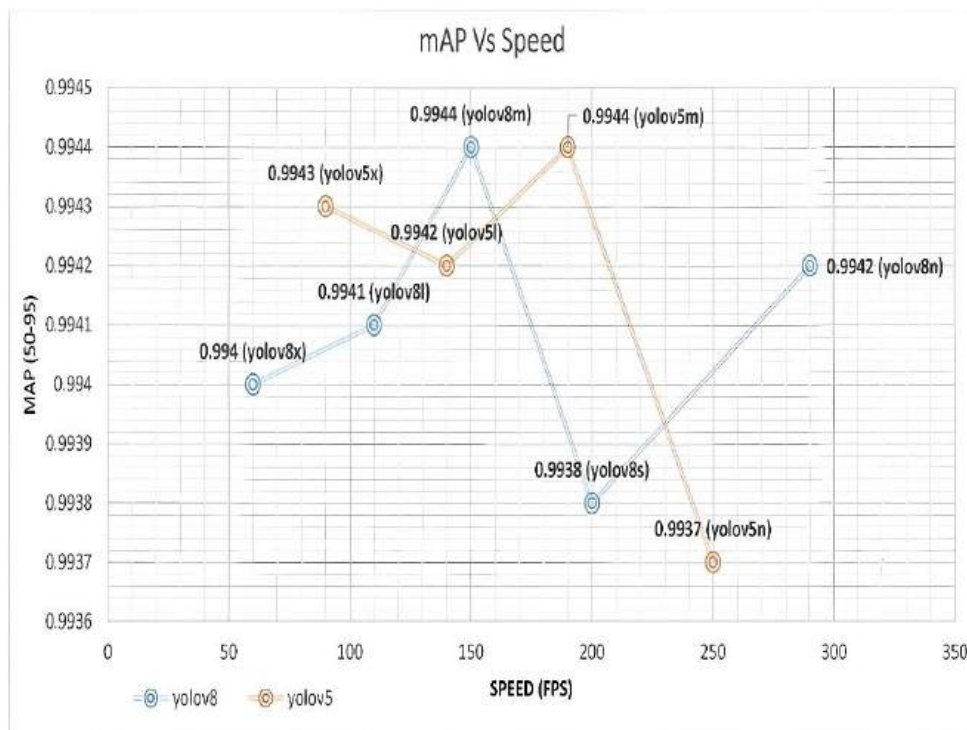
Sumber: <https://www.mdpi.com/1424-8220/25/6/1656>

Secara singkat, alur kerja sistem pada gambar tersebut dibagi menjadi beberapa bagian:

- Akuisisi Data (ESP32-CAM): Perangkat kamera menangkap citra dari lingkungan sekitar.
- Jalur Komunikasi (MQTT): Pengiriman data dari ESP32-CAM menuju server dijumpai oleh *MQTT broker*
- Pemrosesan di Cloud Server: Data yang dikirim melalui *broker* akan diterima oleh server (direpresentasikan oleh perangkat Laptop). Di dalam lingkungan pemrograman Python, data citra tersebut diolah lebih lanjut menggunakan OpenCV dan dianalisis oleh YOLOv8 untuk mendeteksi objek.
- Output dan Penyimpanan: Hasil deteksi dari YOLOv8 akan menghasilkan keluaran berupa citra (*image*) dan pesan peringatan (*message*), serta data tersebut dapat disimpan ke dalam media penyimpanan fisik (HDD).

Sedangkan pada gambar 2.5 dibawah ini, YOLO merupakan metode deteksi objek berbasis *deep learning* yang dirancang untuk melakukan proses deteksi dalam satu tahap komputasi, sehingga mampu bekerja secara cepat dan efisien. Pada penelitian ini digunakan YOLOv8 karena memiliki performa yang lebih ringan dan akurat dibandingkan generasi sebelumnya, terutama untuk kebutuhan deteksi manusia secara *real-time*. YOLOv8 bekerja dengan memproses satu citra secara

langsung untuk menghasilkan informasi berupa *bounding box*, kelas objek, dan tingkat kepercayaan (*confidence score*).



Gambar 2.5. FPS Performance of YOLOv5 and YOLOv8

Sumber: https://www.researchgate.net/figure/Comparison-FPS-performance-of-YOLOv5-and-YOLOv8-models-on-Nvidia-Quadro-RTX-4000_fig13_378882476

Dalam sistem TA ini, Pemilihan YOLO untuk Kasus Ini (berbasis data & praktik) pemilihan metode YOLO dalam penelitian ini didasarkan pada tiga alasan utama yang kuat dari sisi praktik nyata:

1) Real-time Detection Performance

YOLO dikenal sebagai *single-shot detector* yang mampu memproses deteksi objek dalam satu operasi jaringan, berbeda dengan metode berbasis region proposal yang memerlukan operasi berulang. Hal ini terbukti dalam berbagai studi terbaru seperti Khan *et al.* (2023) yang menunjukkan YOLOv8 mampu mencapai kecepatan di atas 30 FPS pada GPU kelas menengah dengan akurasi deteksi manusia di atas 90%.

2) Kemampuan Multi-Object and Multi-Class Detection

Dalam konteks masjid, sistem tidak hanya perlu mendeteksi keberadaan jamaah, tetapi juga indikasi aktivitas lain seperti tidur atau perilaku mencurigakan. YOLOv8 mampu mengenali lebih dari satu objek dalam satu frame dengan akurasi tinggi, termasuk pada kelas objek baru.

3) Studi Lapangan penelitian Implementasi YOLOv8 untuk *Smart building*

Dalam studi yang dilakukan oleh Utomo *et al.* (2024) pada sistem smart classroom, YOLOv8 digunakan untuk mendeteksi keberadaan manusia untuk pengendalian lampu dan AC secara otomatis. Hasil dilaporkan bahwa sistem tersebut:

- a. Mencapai akurasi deteksi 92%–95%
- b. Mampu menekan waktu deteksi rata-rata < 25 ms/frame
- c. Menunjukkan penghematan energi hingga 25–30% dibanding kontrol manual

Selain itu terdapat beberapa rumus yang saya gunakan untuk analisa dan pendukung untuk implementasi berbasis YOLO ini,

a. Rumus operasi konvolusi:

$$Y_{i,j,d} = \sum_{m=0}^{k-1} \sum_{n=0}^{k-1} \sum_{c=1}^C W_{m,n,c,d} \cdot X_{i+m,j+n,c} + b_d \quad (2.1)$$

Konvolusi, yaitu perhitungan gabungan antara input citra (x) dengan filter/kernel (W). Setiap posisi pada peta fitur keluaran Y dihitung dengan menjumlahkan hasil kali setiap elemen kernel dengan nilai yang sesuai pada citra input di area yang berkorespondensi, kemudian ditambah bias b_d . Zona operasi diatur oleh ukuran kernel $K \times K$ dan jumlah saluran warna/ fitur C.

b. Fungsi ReLU (*Rectified Linear Unit*)

$$f(x) = \max(0, x) \quad (2.2)$$

ReLU memetakan semua nilai negatif menjadi nol dan mempertahankan nilai positif apa adanya. Secara matematis, fungsi ini memilih nilai maksimum antara nol dan masukan asli. Pendekatan ini membantu jaringan memperkenalkan non-linearitas, mempercepat proses pelatihan, dan mengurangi masalah gradien memudar dibandingkan dengan fungsi tradisional seperti Sigmoid atau Tanh.

c. *Downsampling / Pooling*

$$P_{i,j,d} = \max_{(p,q) \in R} Y_{i+p,j+q,d} \quad (2.3)$$

Pooling mengambil bagian untuk pengecilan ukuran peta fitur agar jumlah parameter berkurang dan representasi lebih padat. Misalnya *max-pooling* memilih nilai terbesar di dalam setiap blok area R (contoh: 2×2) dari peta fitur konvolusi.

d. Rumus Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad (2.4)$$

Fungsi sigmoid memetakan nilai keluaran jaringan ke rentang probabilitas antara 0 dan 1. Pada YOLO, sigmoid digunakan untuk:

- Menentukan probabilitas keberadaan objek (*objectness score*)
- probabilitas kelas objek (misalnya jamaah wanita atau indikasi *illegal action*)

e. Rumus *Bounding box*

$$b_x = \sigma(t_x) + c_x \quad (2.5)$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w \cdot e^{t_w}$$

$$b_h = p_h \cdot e^{t_h}$$

Keterangan:

- b_x, b_y : koordinat pusat *bounding box*
- b_w, b_h : lebar dan tinggi *bounding box*
- t_x, t_y, t_w, t_h : output jaringan
- c_x, c_y : posisi grid sel
- p_w, p_h : ukuran anchor box

f. *Objectness Confidence score*

YOLO menghitung tingkat keyakinan bahwa suatu *bounding box* benar-benar mengandung objek

$$Confidence = P(Object) \times IoU \quad (2.6)$$

Nilai *confidence* digunakan untuk Menyaring deteksi palsu, menentukan apakah suatu kejadian layak disimpan sebagai rekaman ilegal action, dan menghindari pengaktifan fasilitas listrik akibat noise visual

g. *Intersection over Union (IoU)*

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (2.7)$$

Rumus IoU ini Digunakan dalam evaluasi akurasi deteksi dalam menentukan validitas deteksi manusia dan aktivitas dengan mempengaruhi keputusan sistem dalam menyimpan data dan mengirim notifikasi

h. Pengukuran Dimensi Output

Untuk lapisan konvolusi, perhitungan ukuran spasial keluaran menggunakan rumus dibawah setelah menerapkan operasi konvolusi dengan padding dan stride tertentudimensi keluaran di masing-masing arah (lebar/tinggi) sering dihitung sebagai:

$$O = \frac{I - K + 2P}{S} + 1 \quad (2.8)$$

Di mana:

- I = ukuran masukan (input)
- K = ukuran kernel
- P = padding (jumlah piksel tambahan di tepi)
- S = *stride* (langkah pergeseran kernel)

i. Rumus untuk Daya listrik AC satu fasa

Untuk perhitungan daya AC diperoleh dari pengukuran tegangan, arus, dan faktor daya, sehingga daya aktif dihitung menggunakan rumus dibawah ini:

$$P = V \times I \times \cos \phi \quad (2.9)$$

Dimana:

- Tegangan = 220 Volt
- $\cos \phi = 0.85$
- P = Daya Aktif (W/kW)
- V = Tegangan listrik (V)
- I = Kuat Arus Listrik (A)

Daya aktif merupakan daya listrik yang secara nyata digunakan oleh beban untuk melakukan kerja. Pada sistem listrik AC satu fasa, besarnya daya aktif dipengaruhi oleh tegangan, arus, dan faktor daya beban. Daya aktif dapat dihitung menggunakan Persamaan (2.9)

j. Rumus Rata-Rata Daya

$$P' = \frac{P_1 + P_2 + P_3 + \dots + P_n}{n} \quad (2.10)$$

dengan:

- P' = rata-rata daya (W)
- P_1 = daya pada pengukuran ke-1,2,3,dan seterusnya (W)
- n = jumlah data pengukuran

Nilai rata-rata digunakan untuk memperoleh representasi penggunaan daya selama periode pengujian. Perhitungan dilakukan dengan menjumlahkan seluruh nilai daya hasil pengukuran kemudian membaginya dengan jumlah data pengamatan.

k. Persentase Penggunaan Daya

$$P\% = \frac{P_{aktual}}{P_{acuan}} \times 100\% \quad (2.11)$$

Persentase penggunaan daya digunakan untuk mengetahui proporsi daya aktual terhadap daya acuan. Daya acuan dapat berupa daya maksimum peralatan atau daya pada kondisi operasi normal, tergantung skenario pengujian.

l. Persentase penghematan daya

Persentase penghematan daya menunjukkan besarnya penurunan penggunaan daya setelah penerapan sistem pengendalian dibandingkan kondisi konvensional. Seperti pada rumus dibawah ini:

$$\eta P = \frac{P_{sistem}}{P_{konvensional}} \times 100\% \quad (2.12)$$

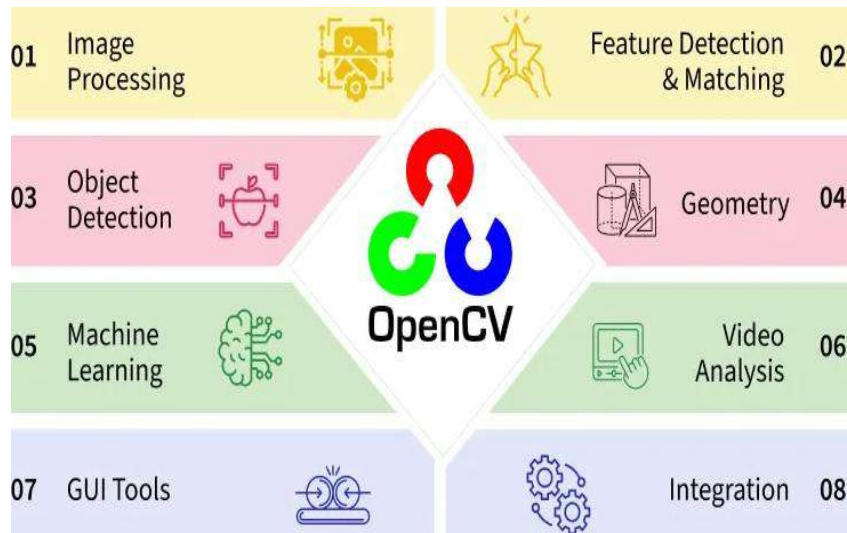
Dimana:

- ηP = Persentase penghematan daya (%)
- $P_{konvensional}$ = daya rata-rata sebelum/dengan pengoperasian konvensional (W)
- P_{sistem} = Daya rata-rata setelah menggunakan sistem pengendalian (W)

Jadi semakin besar nilai persentase penghematan, maka semakin besar penurunan kebutuhan daya yang diperoleh melalui sistem.

2.2.5 OpenCV

OpenCV (*Open Source Computer Vision Library*) merupakan pustaka *open-source* yang dirancang khusus untuk pengolahan citra digital dan visi laptop secara *real-time*. OpenCV menyediakan berbagai fungsi dasar hingga lanjutan, seperti akuisisi citra dari kamera, konversi warna, *filtering*, segmentasi objek, serta integrasi dengan algoritma pembelajaran mesin dan *deep learning*. Dalam sistem tugas akhir ini, OpenCV berperan sebagai komponen awal dalam *pipeline* IoT dan *machine vision*, yaitu menangani aliran data citra dari kamera USB sebelum diproses lebih lanjut oleh model YOLOv8.



Gambar 2.6. Alur dari OpenCV

Sumber: <https://www.geeksforgeeks.org/computer-vision/opencv-overview/>

Sesuai gambar 2.6 diatas, OpenCV memungkinkan sistem untuk melakukan *preprocessing* citra, seperti *resize*, normalisasi, dan konversi format, sehingga citra yang masuk ke model YOLO berada pada kondisi yang optimal. Menurut Bradski dan Kaehler (2008), OpenCV dirancang untuk mendukung aplikasi visi laptop yang membutuhkan kecepatan tinggi dan keandalan, sehingga sangat sesuai untuk sistem otomasi berbasis kamera yang bekerja secara kontinu. Dalam konteks masjid, OpenCV memastikan citra jamaah dapat diproses secara stabil meskipun terjadi perubahan intensitas cahaya atau kondisi lingkungan.

2.3 Hardware

Perangkat keras pada penelitian ini dirancang sebagai fondasi utama dalam merealisasikan sistem pengendalian fasilitas listrik dan pengawasan aktivitas berbasis IoT di Masjid Bahrul Ulum. Seluruh komponen *Hardware* dipilih dan dikonfigurasi untuk mendukung proses akuisisi data visual dan sensorik, pemrosesan di sisi *edge computing*, serta eksekusi pengendalian beban listrik secara aman dan andal. Integrasi antar perangkat keras memungkinkan sistem bekerja secara *real-time* dalam mendeteksi keberadaan jamaah, indikasi aktivitas ilegal, serta menyesuaikan kondisi penerangan dan sirkulasi udara sesuai kebutuhan aktual dalam beribadah.

Secara umum, arsitektur *Hardware* terdiri dari kamera pengawas sebagai sensor utama, *mini PC* sebagai unit pemrosesan YOLO, mikrokontroler ESP32

sebagai pengendali IoT, serta perangkat pendukung berupa sensor lingkungan, aktuator listrik, dan sistem proteksi. Perancangan ini mempertimbangkan keterbatasan akses pengelola masjid, kebutuhan efisiensi energi, serta keandalan sistem dalam operasi jangka panjang.

2.3.1 ESP 32

ESP32 adalah mikrokontroler dari sistem Espressif yang dilengkapi dengan WiFi terintegrasi langsung di dalam chip, menjadikannya ideal untuk mendukung pengembangan infrastruktur IoT modern yang tertera pada gambar 2.7 Dalam penelitian ini, ESP32 berfungsi untuk mengumpulkan data dari sensor dan mengirimkannya ke database, sehingga berperan sebagai penghubung utama yang memungkinkan akuisisi data secara efisien dan transmisi ke sistem penyimpanan.



Gambar 2.7. ESP32
Sumber: Pribadi

Mikrokontroler ini juga dilengkapi dengan 18 saluran konverter analog ke-digital (ADC), 2 saluran konverter digital-ke-analog (DAC), 16 modul untuk modulasi lebar pulsa (PWM), 10 sensor sentuh kapasitif, 2 saluran komunikasi UART, serta 10 pin antarmuka yang mendukung protokol seperti I2C, I2S, dan SPI. Untuk spesifikasi lengkap ESP 32 dapat dilihat pada tabel 2.2

Tabel 2.2. Spesifikasi ESP 32

Atribut	Detail
CPU	Tensilica Xtensa LX6 32bit Dual Core di 160/240MHz
RAM	520 KB
FLASH	2Mb (max, 64MB)
Tegangan	2,2V – 2,6 V
Arus Kerja	Rata – rata 80mA

Atribut	Detail
Dapat Diprogram	Ya (C,C++, Python, Lua, dll)
Open Sorce	Ya
Konektivitas	
Wi-Fi	802.11 b/g/n
Bluetooth	4.2BR/EDR+BLU
UART	3
I/O	
GPIO	32
SPI	4
I2C	2
PWM	8
ADC	18 (12-bit)
DAC	2 (8-bit)
Konektivitas pin esp 32 ke <i>hardware</i>	
<i>Relay Speed 1</i>	Pin 25
<i>Relay Speed 2</i>	Pin 26
<i>Relay Speed 3</i>	Pin 27
<i>Relay lampu</i>	Pin 33
SCL dan SDA (RTC)	Pin 21 dan 22
LCD I2C	Jumper Pin 21 dan 22
Sensor LDR	Pin 34

2.3.2 TP-Link Tapo C212 (*Camera*)

Pada gambar 2.8 Kamera yang digunakan pada sistem ini adalah TP-Link Tapo C212, yaitu kamera pengawas berbasis IP Camera yang berfungsi sebagai sensor utama pada sistem IoT dan *computer vision*. Kamera ini digunakan untuk menangkap citra visual area lantai dua Masjid Bahrul Ulum PPNS yang selanjutnya diproses oleh laptop menggunakan metode *You Only Look Once* (YOLO) untuk

mendeteksi keberadaan jamaah, *illegal overnight stay*, dan *Indication of a person littering*.



Gambar 2.8. Tapo C-212
Sumber: Shopee

Pemilihan kamera ini didasarkan pada beberapa pertimbangan, antara lain kualitas citra yang tinggi, kemampuan pemantauan jarak jauh secara *real-time*, serta kemudahan integrasi dengan sistem berbasis jaringan dan *deep learning*. Dengan resolusi hingga 2K QHD (2560×1440), kamera mampu menghasilkan citra yang detail sehingga mendukung proses deteksi manusia secara lebih akurat pada berbagai kondisi pencahayaan di area masjid. Hal ini sangat penting untuk meningkatkan performa sistem dalam mengenali keberadaan jamaah pada berbagai posisi shaf dan sudut ruangan secara *real-time*. Selain itu, fitur *night vision* dan *motion tracking* memungkinkan kamera tetap melakukan pemantauan secara optimal pada kondisi cahaya rendah maupun saat terdapat pergerakan objek di area pengawasan.

Desain kamera yang dapat berputar hingga 360 derajat juga memudahkan penyesuaian sudut pandang kamera sesuai dengan tata letak ruangan masjid

Tabel 2.3. Tapo C-212 Camera

Spesifikasi	Keterangan
Jenis kamera	IP Camera, Motion Tracking dan Night Vision
Prosesor	2560×1440 (2K QHD)
Resolusi maksimum	2560×1920 (Quad HD / 2K)
Resolusi operasional	1920×1080 (Full HD)
Kompatibilitas <i>Software</i>	OpenCV, Python, YOLOv8
Metode instalasi	WiFi IEEE 802.11 b/g/n

Spesifikasi	Keterangan
Sudut pandang	<i>Pan 360° dan Tilt 114°</i>
Jarak efektif deteksi	$\pm 1 - 8$ meter
Penyimpanan	<i>MicroSD dan Cloud Storage</i>
Sistem operasi	Windows dan Linux

2.3.3 *Power Supply 10A/12V*

Power Supply merupakan komponen yang memberikan pasokan daya listrik kepada satu atau lebih perangkat elektronik. Saat ini, *Power Supply* telah dirancang dengan kemampuan untuk mengubah berbagai Sumber energi dasar seperti energi matahari, angin, dan energi kimia menjadi energi listrik yang diperlukan. *Power supply* digunakan sebagai sumber tegangan utama untuk seluruh rangkaian sistem, baik pada sisi mikrokontroler, sensor, maupun modul pendukung. Pemilihan power supply dilakukan dengan mempertimbangkan kestabilan tegangan dan kemampuan arus agar sistem dapat beroperasi secara kontinu tanpa gangguan, terutama pada kondisi beban listrik yang berubah-ubah.



Gambar 2.9. *Power Supply 12V 10A*
Sumber: Pribadi

Pada gambar 2.9 Komponen ini memiliki peran yang krusial, terutama dalam sistem laptop dan perangkat elektronik lainnya, dan keberadaannya tidak dapat diabaikan. Gangguan atau masalah pada *Power Supply* dapat mengakibatkan disfungsi pada perangkat yang bersangkutan. Saat mengaktifkan perangkat seperti laptop, *Power Supply* melakukan pemeriksaan dan uji coba sebelum menjalankan sistem operasional perangkat tersebut. Pada dasarnya, *Power Supply* tidak hanya berperan sebagai penyedia energi listrik, tetapi juga mampu mengubah energi dari berbagai Sumber lainnya, termasuk energi mekanik dan jenis energi lainnya. (Akbar *et al.*, 2023)

2.3.4 MCB 4A

Miniature Circuit Breaker (MCB) merupakan komponen proteksi utama dalam sistem distribusi daya listrik yang berfungsi memutus arus listrik secara otomatis ketika terjadi kelebihan arus (*overcurrent*) atau hubungan singkat (*short circuit*).



Gambar 2.10. MCB 4A
Sumber: Tokopedia

Bentuk fisik MCB ada pada 2.10, dimana berperan penting dalam mencegah kerusakan pada perangkat listrik dan kebakaran akibat arus berlebih dengan mekanisme pemutusan otomatis yang cepat dan dapat diaktifkan kembali secara manual setelah gangguan diatasi (Prasad & Singh., 2020). Pada dasarnya keberadaan MCB sangat penting untuk menjaga keamanan instalasi listrik masjid serta melindungi perangkat elektronik dari kerusakan akibat lonjakan arus. Dalam sistem kontrol berbasis IoT dan mikrokontroler ESP32 di Masjid Bahrul Ulum, MCB tidak hanya berfungsi sebagai pengaman tetapi sebagai pembagi beban listrik ke setiap kelompok fasilitas (lampu, kipas, dan sistem kontrol). Selain itu, juga sebagai pelindung kabel dan komponendari kerusakan akibat arus berlebih.

2.3.5 Laptop

Laptop merupakan komputer portabel yang memiliki fungsi setara dengan komputer desktop, namun dirancang dalam bentuk yang lebih ringkas sehingga mudah dipindahkan dan digunakan di berbagai lokasi. Dengan perkembangan teknologi perangkat keras, laptop saat ini telah mampu menjalankan berbagai aplikasi komputasi berat, termasuk pengolahan citra digital, komputasi berbasis kecerdasan buatan (*Artificial Intelligence*), dan *Deep Learning*.

Pada penelitian ini, laptop digunakan sebagai pusat pemrosesan utama (*edge computing device*) yang menjalankan sistem berbasis Python dan algoritma

YOLOv8. Laptop yang saya gunakan sesuai pada gambar 2.11, dimana Perangkat ini bertanggung jawab melakukan akuisisi citra dari kamera CCTV, memproses setiap frame menggunakan model Convolutional Neural Network (CNN) YOLOv8 untuk mendeteksi keberadaan manusia, aktivitas membuang sampah, serta indikasi tidur atau menginap di area masjid.



Gambar 2.11. Laptop IdeaPad Slim 3 14ABR8
Sumber: Pribadi

Selanjutnya, hasil deteksi dikirimkan ke mikrokontroler ESP32 untuk mengendalikan lampu, kipas, dan perangkat pendukung lainnya melalui jaringan Internet of Things (IoT). Selain itu, laptop juga berfungsi sebagai server lokal yang menjalankan aplikasi backend berbasis Flask untuk menghubungkan sistem deteksi, database, dan dashboard monitoring secara *real-time*.

Penggunaan laptop sebagai perangkat *edge computing* memungkinkan seluruh proses inferensi dilakukan secara lokal tanpa bergantung pada layanan *cloud*, sehingga waktu respons sistem menjadi lebih cepat, penggunaan bandwidth internet lebih rendah, serta keamanan dan privasi data lebih terjaga. Konsep ini sejalan dengan Shi et al. (2016) yang menyatakan bahwa *Edge Computing* mampu meningkatkan respons sistem, mengurangi latensi komunikasi, serta menjaga privasi data, khususnya pada aplikasi berbasis pengolahan citra dan Internet of Things.

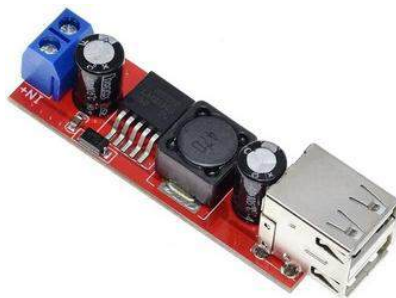
Tabel 2.4. Spesifikasi lengkap yang disarankan untuk dimiliki

Spesifikasi	Keterangan
Prosesor	Intel Core i7 Generasi ke-12 (atau AMD Ryzen 7 6800H ke atas)
Kecepatan prosesor	Hingga $\pm 4,7$ GHz (Turbo Boost)
Jumlah Core / Thread	Minimal 10 Core / 16 Thread

Spesifikasi	Keterangan
RAM	16 GB DDR5 (minimal 16 GB)
Penyimpanan	SSD NVMe 512 GB
GPU	NVIDIA GeForce RTX 4050 Laptop 6 GB (atau minimal RTX 3060 Laptop 6 GB)
Monitor	15,6 inci Full HD (1920 × 1080)
Konektivitas	Wi-Fi 6, Bluetooth 5.2, Gigabit Ethernet (melalui LAN atau adaptor USB-LAN)
Sistem operasi	Windows 11 64-bit

2.3.6 DC Stepdown 5V 3A

Pada gambar 2.12 dibawah ini, *DC Stepdown 5V 3A* berfungsi sebagai modul penurun tegangan (*buck converter*) yang digunakan untuk menstabilkan suplai daya pada perangkat elektronik bertegangan rendah. Komponen ini memiliki peran penting dalam sistem berbasis *Internet of Things (IoT)* karena mampu mengubah tegangan masukan DC yang lebih tinggi menjadi keluaran 5V yang stabil sesuai kebutuhan mikrokontroler dan modul pendukung lainnya. Dalam implementasi sistem di Masjid Bahrul Ulum PPNS, *DC Stepdown 5V 3A* digunakan untuk menyuplai daya pada *ESP32*, *LCD I2C*, sensor, dan perangkat pendukung kontrol sehingga sistem dapat bekerja secara kontinu tanpa gangguan tegangan berlebih.



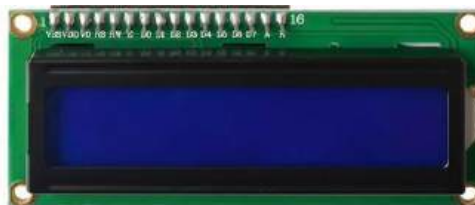
Gambar 2.12. DC Stepdown 5V 3
Sumber: Shopee

Modul ini bekerja dengan prinsip *switching regulator*, yaitu mengatur proses konversi daya secara efisien dengan tingkat kehilangan energi yang lebih rendah dibanding regulator linear konvensional. Menurut Rashid (2018), penggunaan *DC stepdown converter* sangat direkomendasikan pada sistem

elektronika modern karena memiliki efisiensi tinggi, ukuran yang ringkas, serta mampu menjaga kestabilan tegangan keluaran pada kondisi beban yang berubah-ubah. Hal ini sesuai dengan kebutuhan sistem otomasi masjid yang memiliki beberapa perangkat elektronik aktif secara bersamaan dalam satu rangkaian kontrol. Dalam tugas akhir ini, *DC Stepdown 5V 3A* menjadi bagian penting dalam menjaga keandalan sistem kontrol dan monitoring agar proses komunikasi data, pengolahan sensor, dan pengendalian fasilitas listrik dapat berlangsung dengan stabil dan aman selama sistem beroperasi secara *real-time*.

2.3.7 LCD I2C

Pada gambar 2.13 dibawah merupakan komponen LCD I2C yang untuk sistem ini digunakan sebagai media tampilan informasi lokal pada panel kontrol. Modul ini menampilkan status sistem seperti kondisi lampu, kipas, hasil pembacaan sensor, serta status koneksi ESP32. Modul ini menyederhanakan koneksi antara LCD dan mikrokontroler dengan mengurangi jumlah pin yang dibutuhkan untuk komunikasi dari 16 menjadi hanya 4. Selain itu LCD berbasis antarmuka I2C ini secara umum digunakan seperti kondisi operasi, hasil deteksi singkat, atau status koneksi.



Gambar 2.13. LCD *Backlight with I2C*
Sumber: Shopee

Penggunaan I2C dipilih untuk meminimalkan penggunaan pin mikrokontroler serta meningkatkan kerapian instalasi perangkat. Penggunaan antarmuka I2C dipilih karena efisien dalam penggunaan pin mikrokontroler dan mempermudah integrasi dengan ESP32. Informasi yang ditampilkan pada LCD membantu teknisi atau pengurus masjid untuk melakukan pengecekan cepat tanpa harus membuka antarmuka web. LCD ini berfungsi sebagai lapisan monitoring lokal.

2.3.8 *Relay 4 Channel Optocoupler*

Pada gambar 2.14 dibawah adalah modul *Relay 4 Channel Optocoupler* yang digunakan sebagai aktuator utama untuk mengendalikan beberapa beban listrik secara otomatis pada sistem. Modul *Relay* ini berfungsi sebagai saklar elektronik yang memungkinkan mikrokontroler ESP32 mengontrol perangkat AC seperti lampu dan kipas tanpa terhubung langsung dengan tegangan tinggi. Penggunaan optocoupler pada modul *Relay* bertujuan untuk memberikan isolasi antara rangkaian kontrol tegangan rendah dan rangkaian daya, sehingga meningkatkan keamanan sistem dan mengurangi risiko gangguan listrik terhadap mikrokontroler. Selain itu, modul *Relay 4 channel* memungkinkan pengendalian beberapa perangkat secara bersamaan dalam satu sistem terintegrasi. Pada penelitian ini, *Relay* dikendalikan berdasarkan hasil deteksi jumlah jamaah dan kondisi ruang yang diproses menggunakan metode YOLO, sehingga sistem mampu mengaktifkan atau mematikan fasilitas listrik secara otomatis sesuai kebutuhan penggunaan ruang ibadah. Menurut Bolton (2015), *Relay* merupakan komponen penting dalam sistem otomasi karena mampu menghubungkan sistem kendali digital dengan perangkat listrik berdaya lebih besar secara aman dan efisien. Penggunaan modul *Relay* dengan optocoupler juga membantu meningkatkan kestabilan sistem dengan meminimalkan gangguan elektromagnetik dan lonjakan arus pada rangkaian kontrol.



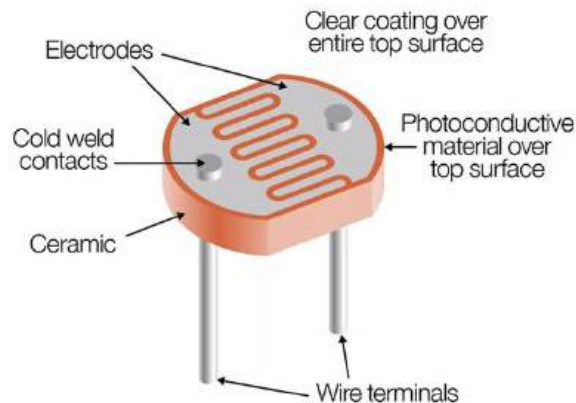
Gambar 2.14. *Relay Module 4 Channel Optocoupler*
Sumber: Shopee

Dalam sistem tugas akhir ini, *Relay 4 channel* digunakan untuk mengendalikan lampu dan beberapa tingkat kecepatan kipas angin berdasarkan hasil pembacaan sistem deteksi berbasis YOLO dan sensor pendukung. *Relay* menerima sinyal digital dari ESP32 untuk menentukan kondisi ON atau OFF pada masing-masing beban listrik secara otomatis. Dengan penggunaan *Relay* ini, proses

pengendalian fasilitas listrik dapat dilakukan secara real-time, lebih efisien, serta mendukung penerapan otomasi fasilitas pada lingkungan Masjid Bahrul Ulum PPNS.

2.3.9 Sensor LDR (*Light Dependent Resistor*)

Dari gambar 2.15 dibawah ini, sensor LDR digunakan sebagai sensor pendukung untuk membaca intensitas cahaya lingkungan di dalam ruangan masjid.



Gambar 2.15. Sensor LDR (*Light Dependent Resistor*)

Sumber: <https://dennydarlis.staff.telkomuniversity.ac.id/sensor-cahaya/>

Sensor ini bekerja berdasarkan prinsip perubahan resistansi terhadap intensitas cahaya yang diterimanya. Semakin terang kondisi ruangan, nilai resistansi LDR akan semakin kecil, dan sebaliknya. Dalam sistem ini, LDR tidak berperan sebagai pengambil keputusan utama, melainkan sebagai pemberi pertimbangan tambahan dalam pengendalian lampu.

Data dari LDR dikombinasikan dengan hasil deteksi manusia dari kamera berbasis YOLO. Jika kamera mendeteksi keberadaan manusia namun intensitas cahaya masih cukup terang (siang hari), maka sistem hanya mengaktifkan kipas tanpa menyalakan lampu. Sebaliknya, pada kondisi gelap atau malam hari, data LDR membantu sistem untuk mengaktifkan lampu secara otomatis. Hal ini mencerminkan konsep sensor fusion, di mana beberapa sensor digunakan secara bersamaan untuk meningkatkan akurasi keputusan sistem, sehingga mengurangi kemungkinan kesalahan kontrol akibat satu sensor saja.

2.3.10 Router *wireless*

Pada gambar 2.16 dibawah ini, Router berfungsi sebagai pusat penghubung komunikasi data antar perangkat dalam sistem *electrical facility control* yang

dikembangkan. Pada penelitian ini, router tidak digunakan sebagai penyedia akses internet publik, melainkan sebagai perangkat jaringan lokal (Local Area Network) yang mengatur lalu lintas data antara laptop utama (PC), ESP32 master, dan antarmuka web.



Gambar 2.16. Router Penguat Sinyal Antar ESP32
Sumber: Shopee

Router memungkinkan komunikasi berbasis protokol TCP/IP dan HTTP sehingga data hasil pemrosesan citra dari laptop dapat dikirimkan ke modul ESP32 secara stabil dan terstruktur. Dalam konteks *Internet of Things* (IoT), router berperan sebagai node penghubung yang menjamin reliabilitas pertukaran data antar perangkat tanpa ketergantungan pada koneksi internet eksternal. Hal ini sejalan dengan konsep IoT yang dikemukakan oleh Gubbi *et al.* (2013), yang menyatakan bahwa sistem IoT tidak selalu membutuhkan koneksi internet global, tetapi cukup dengan jaringan lokal yang andal untuk mendukung pengambilan keputusan otomatis. Penggunaan router juga mempermudah integrasi sistem dengan antarmuka web, baik untuk monitoring maupun kontrol manual, sehingga pengurus masjid dapat mengakses sistem melalui laptop atau perangkat seluler tanpa perlu interaksi langsung dengan panel kontrol.

2.3.11 Kabel AWG 0.75 dan 1.5

Sesuai gambar 2.17, Kabel AWG (*American Wire Gauge*) merupakan standar pengukuran diameter kabel yang umum digunakan pada instalasi sistem listrik dan elektronika. Dalam implementasi sistem pada masjid, kabel AWG digunakan untuk menghubungkan berbagai komponen seperti modul kontrol, sensor, *Relay*, dan aktuator. Dalam implementasinya, kabel AWG tidak hanya

berfungsi sebagai media penghantar daya, tetapi juga sebagai jalur komunikasi sinyal kontrol antara ESP32 dan perangkat keluaran. Konektivitas yang baik melalui kabel dengan spesifikasi yang tepat memastikan respon sistem tetap cepat dan akurat.



Gambar 2.17. Kabel AWG
Sumber: Shopee

Pemilihan ukuran AWG juga akan disesuaikan dengan arus dan tegangan yang dilewatkan, terutama pada jalur yang terhubung ke mcb, fuse, *Relay* 4 channel, dan beban listrik seperti lampu dan kipas. Sistem pengukuran AWG penting untuk menentukan kapasitas arus dan jarak transmisi yang sesuai untuk aplikasi tertentu. Standar ini mengacu pada diameter kawat, dengan nomor yang lebih kecil menunjukkan kawat yang lebih tebal. Penggunaan kabel AWG dengan ukuran 24 memiliki kabel serabut fleksibel yang cocok untuk proyek elektronik, IoT, dan robotika, memberikan konektivitas yang stabil.

2.3.12 Fuse 2A dan Terminal

Pada gambar 2.18, saya menggunakan tipe *Fuse* 2A digunakan sebagai komponen proteksi arus lebih (*overcurrent protection*) pada sistem kontrol listrik. Dalam implementasi tugas akhir ini, fuse berfungsi untuk melindungi rangkaian kontrol, modul ESP32, *Relay*, dan perangkat pendukung lainnya dari risiko kerusakan akibat lonjakan arus maupun hubungan singkat (*short circuit*). Penggunaan fuse menjadi sangat penting karena sistem bekerja secara terus-menerus dan terhubung langsung dengan beberapa perangkat elektronik berbasis catu daya DC maupun AC. Ketika terjadi arus yang melebihi batas kemampuan rangkaian, fuse akan memutus jalur listrik secara otomatis sehingga dapat mencegah kerusakan komponen serta mengurangi risiko gangguan pada sistem pengendalian fasilitas listrik berbasis *IoT*.



Gambar 2.18. Fuse 2A dan Terminal
Sumber: Shopee

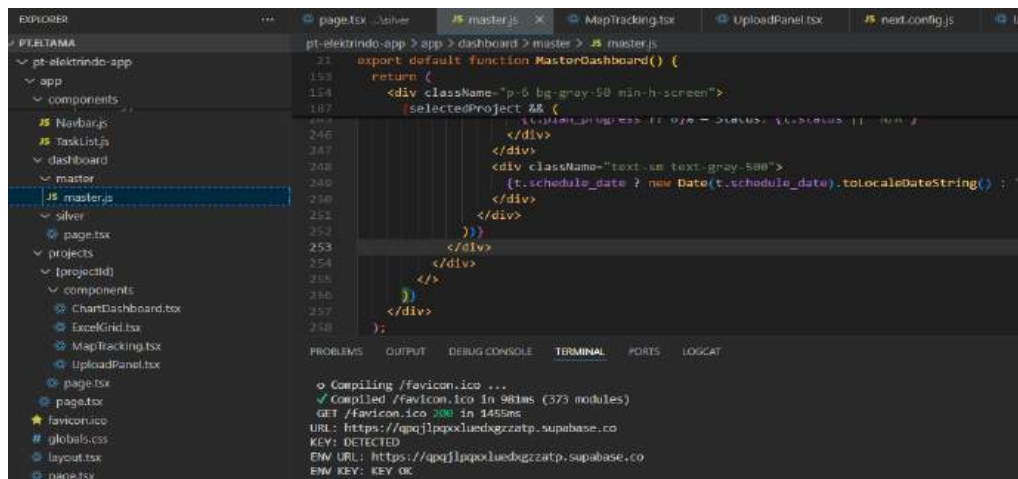
Menurut Herman dan McDonald (2016), fuse merupakan perangkat proteksi dasar yang banyak digunakan pada sistem kontrol modern karena memiliki respon pemutusan arus yang cepat, konstruksi sederhana, serta tingkat keandalan yang tinggi dalam melindungi perangkat elektronik sensitif. Selain itu, penggunaan fuse juga membantu menjaga kestabilan operasi sistem dan meningkatkan aspek keselamatan instalasi listrik, terutama pada lingkungan publik seperti masjid. Dalam sistem tugas akhir ini, fuse 2A ditempatkan pada jalur distribusi daya utama sebagai pengaman awal sebelum arus dialirkan menuju modul kontrol dan perangkat aktuator. Penggunaan fuse memungkinkan sistem bekerja dengan lebih aman dan stabil selama proses monitoring maupun pengendalian fasilitas berlangsung secara *real-time*.

2.4 Software

Perangkat lunak pada sistem ini berperan sebagai pengendali logika, pengolah data, serta penghubung antara proses deteksi visual, pengambilan keputusan, dan antarmuka pengguna. Perangkat lunak dirancang untuk mendukung pemrosesan citra berbasis metode YOLO secara *real-time*, komunikasi data antar perangkat melalui jaringan lokal, serta penyajian informasi sistem secara terstruktur melalui dashboard web. Sistem perangkat lunak mencakup lingkungan pengembangan pemrosesan citra dan *deep learning*, firmware mikrokontroler, serta aplikasi web untuk monitoring dan kontrol jarak jauh. Integrasi Software ini memungkinkan sistem tidak hanya melakukan deteksi objek dan aktivitas, tetapi juga menyimpan data, mengirim notifikasi, serta menyediakan kontrol manual sebagai bentuk redundansi dan keamanan operasional.

2.4.1 Visual Studio Code

Visual Studio Code adalah code editor yang populer, yang menawarkan interfacing yang sederhana dan dapat disesuaikan, serta fitur pengembangan yang lengkap. Pada gambar 2.19, Visual Studio Code juga digunakan untuk menghubungkan sistem lokal dengan server berbasis web, Hive MQ, dan CloudFlare, di mana data deteksi keberadaan manusia dari kamera dikirim ke server melalui protokol HTTP.

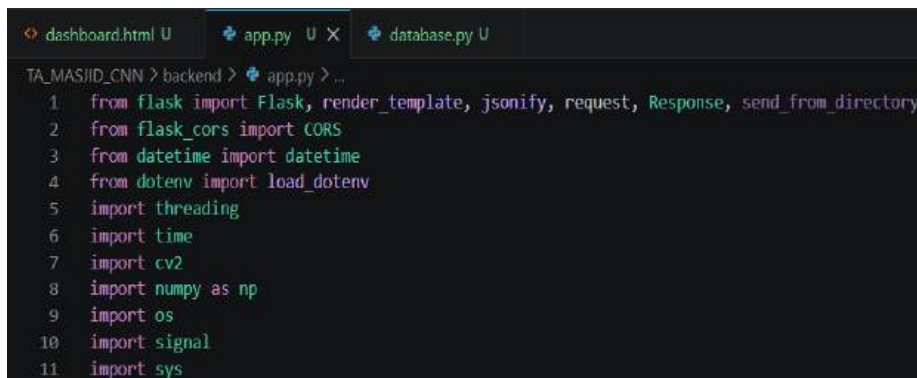


Gambar 2.19. Tampilan Visual Studio Code
Sumber: Pribadi

Dengan dukungan ekstensi yang luas, VS Code dapat disesuaikan dengan kebutuhan pengembangan pengguna, baik itu untuk bahasa pemrograman tertentu, integrasi dengan *platform* lain, atau tools khusus. Selain itu, terminal bawaan dan sistem manajemen file yang terintegrasi, pengembang dapat melakukan uji coba komunikasi antara ESP32, server, dan antarmuka web secara efisien. Penggunaan Git integration di VS Code juga mempermudah proses dokumentasi versi kode dan kolaborasi dengan dosen pembimbing. Sehingga VS Code tidak hanya berperan sebagai alat pengedit teks, tetapi juga sebagai *platform* pengembangan utama yang mengintegrasikan seluruh aspek sistem IoT, YOLO, dan kontrol fasilitas masjid dalam satu lingkungan yang ringkas, fleksibel, dan efisien.

Selain itu saya juga menggunakan *flash framework*, dimana Flask merupakan framework web berbasis bahasa pemrograman Python yang bersifat ringan (*micro web framework*) dan digunakan untuk membangun aplikasi web

maupun layanan *Application Programming Interface* (API). Flask menyediakan berbagai fungsi untuk menangani permintaan (*request*) dari pengguna, mengolah data, serta mengirimkan respons (*response*) kepada klien melalui protokol HTTP. Dengan arsitektur yang sederhana dan fleksibel, Flask banyak digunakan pada sistem *Internet of Things* (IoT), *Computer Vision*, dan *Artificial Intelligence* karena mudah diintegrasikan dengan berbagai pustaka Python seperti OpenCV, Ultralytics YOLO, MQTT, maupun basis data.

The image shows a code editor with three tabs: 'dashboard.html U', 'app.py U X', and 'database.py U'. The 'app.py' tab is active, displaying Python code for a Flask application. The code includes imports for Flask, flask_cors, datetime, dotenv, threading, time, cv2, numpy, os, signal, and sys. The code is as follows:

```
1 from flask import Flask, render_template, jsonify, request, Response, send_from_directory
2 from flask_cors import CORS
3 from datetime import datetime
4 from dotenv import load_dotenv
5 import threading
6 import time
7 import cv2
8 import numpy as np
9 import os
10 import signal
11 import sys
```

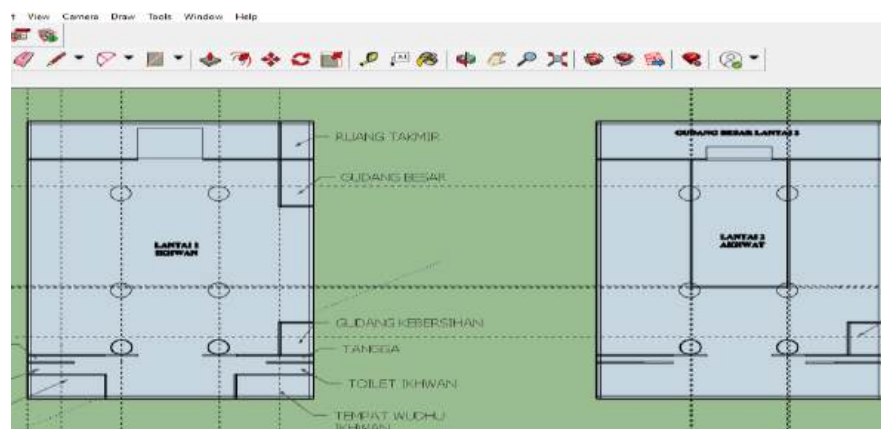
Gambar 2.20. *Flash Framework* Python
Sumber: Pribadi

Implementasinya Flask digunakan sebagai penghubung (*middleware*) antara sistem deteksi berbasis YOLOv8 yang berjalan pada laptop dengan perangkat ESP32, basis data, serta dashboard monitoring berbasis web. Pada gambar 2.20, Flask bertugas menerima hasil deteksi dari proses *Computer Vision*, mengolah informasi tersebut, kemudian mendistribusikannya kepada perangkat maupun antarmuka pengguna secara *real-time*.

2.4.2 SketchUp

SketchUp merupakan perangkat lunak pemodelan 3D berbasis grafis yang digunakan untuk membuat desain arsitektur, bangunan, tata ruang, serta simulasi penempatan perangkat dalam bentuk visual. *Software* ini dikembangkan dengan fokus pada kemudahan penggunaan, presisi geometris, dan kemampuan rendering cepat untuk mendukung tahap perancangan sistem teknis. Menurut Kashani *et al.* (2021), penggunaan SketchUp dalam perencanaan fasilitas dan sistem IoT dapat meningkatkan akurasi tata letak perangkat hingga 25% karena visualisasi ruang yang realistis dan terukur. Dalam konteks penelitian dan implementasi sistem kontrol cerdas berbasis *Internet of Things* (IoT) pada Masjid Bahrul Ulum. Sesuai

gambar 2.21, SketchUp digunakan sebagai perangkat lunak pemodelan tiga dimensi (3D) untuk membuat denah serta visualisasi lokasi implementasi sistem. Pemodelan ini mencakup tata letak ruang ibadah, posisi pemasangan kamera CCTV sebagai media akuisisi citra untuk deteksi menggunakan metode YOLOv8, penempatan lampu otomatis, sensor intensitas cahaya (LDR), modul *Real Time Clock* (RTC), panel kontrol, serta jalur instalasi kelistrikan. Melalui visualisasi tiga dimensi, peneliti dapat menganalisis sudut pandang (field of view) kamera, cakupan area deteksi, serta menentukan posisi perangkat yang paling optimal sehingga potensi *blind spot* dapat diminimalkan dan efektivitas sistem deteksi objek dapat ditingkatkan.



Gambar 2.21. Software SketchUp
Sumber: Pribadi

Selain berfungsi sebagai media perancangan, SketchUp juga mendukung proses validasi desain sebelum tahap implementasi dilakukan di lapangan. Dengan melakukan simulasi tata letak perangkat pada model tiga dimensi, peneliti dapat mengevaluasi kesesuaian posisi sensor dan aktuator terhadap kondisi nyata di masjid, mengidentifikasi kemungkinan kendala instalasi, serta merencanakan kebutuhan material seperti panjang kabel, posisi panel box, dan jalur distribusi daya. Pendekatan ini membantu mengurangi kesalahan pemasangan, meningkatkan efisiensi waktu dan biaya implementasi, serta memastikan bahwa sistem IoT yang dikembangkan dapat bekerja secara optimal sesuai dengan rancangan yang telah dibuat.

2.4.3 Arduino IDE

Arduino IDE merupakan lingkungan pengembangan terintegrasi yang digunakan untuk menulis, mengompilasi, dan mengunggah program ke mikrokontroler, termasuk ESP32. Pada tugas akhir ini, Arduino IDE digunakan untuk mengembangkan firmware ESP32 yang bertugas sebagai pengendali aktuator, seperti *Relay* 4 channel, serta sebagai penerima perintah dari sistem laptop melalui komunikasi jaringan.

```
1  #include <WiFi.h>
2  #include <PubSubClient.h>
3  #include <ArduinoJson.h>
4  #include <Wire.h>
5  #include <LiquidCrystal_I2C.h>
6  #include <RTCLib.h>
7  #include <WiFiClientSecure.h>
8  #include <EEPROM.h>
9
10 // =====
```

Gambar 2.22. Tampilan Coding Arduino IDE
Sumber: Pribadi

Pada Gambar 2.22 ditampilkan antarmuka Arduino Integrated Development Environment (Arduino IDE) yang digunakan sebagai media pemrograman mikrokontroler ESP32 pada penelitian ini. Arduino IDE merupakan perangkat lunak *open-source* yang menyediakan fasilitas penulisan, kompilasi, dan pengunggahan (*upload*) program ke berbagai jenis papan mikrokontroler, termasuk ESP32. Lingkungan pengembangan ini mendukung bahasa pemrograman C/C++ serta memiliki ribuan pustaka (*library*) yang memudahkan integrasi berbagai sensor, modul komunikasi, dan aktuator dalam pengembangan sistem Internet of Things (IoT).

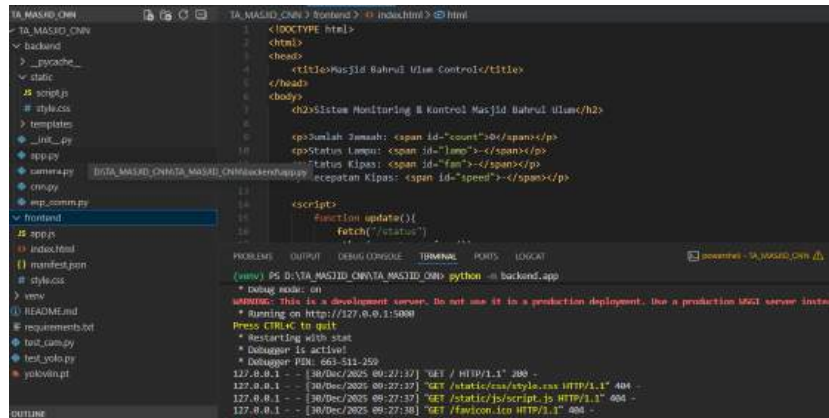
Keunggulan Arduino IDE terletak pada antarmuka yang sederhana, proses konfigurasi yang mudah, serta kompatibilitas yang luas dengan berbagai perangkat keras. Pada penelitian ini, Arduino IDE dimanfaatkan untuk mengembangkan program pada ESP32 yang bertugas menerima data hasil deteksi objek dari mini PC melalui protokol MQTT menggunakan HiveMQ, kemudian mengolah data tersebut menjadi logika kendali untuk mengaktifkan atau menonaktifkan *Relay* lampu dan kipas sesuai kondisi yang terdeteksi. Selain itu, ESP32 juga mengolah data dari sensor LDR dan RTC DS3231 sebagai dasar pengambilan keputusan otomatis terkait kondisi pencahayaan dan waktu operasional sistem, kemudian menampilkan

informasi status perangkat pada LCD I2C serta mengirimkan data ke web monitoring secara *real-time*.

Menurut Banzi dan Shiloh (2022), Arduino IDE dirancang untuk mempercepat proses pengembangan sistem *embedded* melalui lingkungan pemrograman yang sederhana namun fleksibel, sehingga dapat digunakan baik oleh pemula maupun pengembang profesional. Dukungan komunitas yang besar serta ketersediaan berbagai pustaka juga menjadikan Arduino IDE sebagai salah satu platform yang banyak digunakan dalam penelitian berbasis IoT dan otomasi. Oleh karena itu, pada penelitian ini Arduino IDE berperan penting sebagai media pemrograman yang menghubungkan sistem deteksi berbasis *computer vision* dengan perangkat keras pengendali fasilitas listrik Masjid Bahrul Ulum PPNS, sehingga seluruh proses monitoring dan pengendalian dapat berjalan secara otomatis, terintegrasi, dan andal.

2.4.4 Phyton

Python merupakan bahasa pemrograman tingkat tinggi yang banyak digunakan dalam bidang kecerdasan buatan, machine learning, dan visi laptop. Pada tugas akhir ini, Python digunakan sebagai bahasa utama untuk mengimplementasikan YOLO, OpenCV, serta server *backend* berbasis Flask. Selain itu, python juga digunakan pada sisi *edge computing* (laptop) untuk menjalankan proses deteksi objek dan aktivitas berbasis metode YOLO, sekaligus sebagai penghubung antara hasil analisis citra dengan sistem IoT dan antarmuka web. Sehingga dalam implementasinya, python menangani proses akuisisi video secara *real-time* dari kamera pengawas, pemrosesan citra menggunakan model YOLO yang telah dilatih, serta ekstraksi informasi berupa keberadaan, jumlah jamaah, *illegal overnight stay*, dan *Indication of a person littering*. Hasil deteksi ini kemudian dikonversi menjadi data terstruktur yang siap dikirimkan ke sistem pengendali dan layanan *backend*. Proses ini memastikan bahwa keputusan sistem tidak hanya berbasis citra mentah, tetapi berdasarkan hasil inferensi yang telah tervalidasi oleh model.

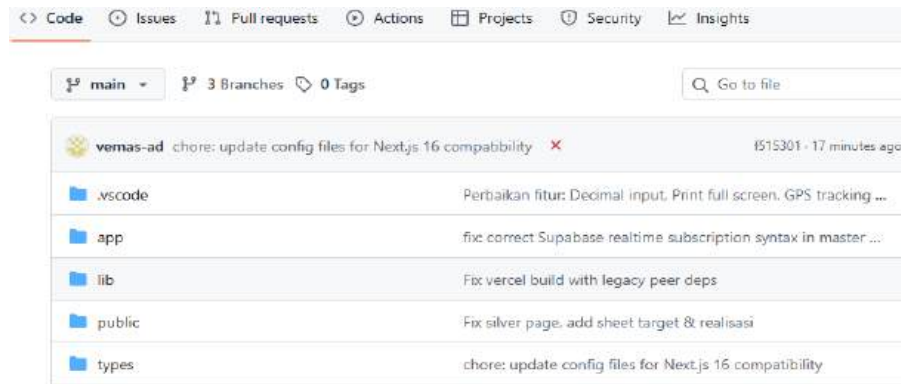


Gambar 2.23. Tampilan Coding *Software* Python
Sumber: Pribadi

Pada 2.23, saya memilih python karena memiliki sintaks yang sederhana namun didukung oleh pustaka yang sangat kuat, seperti OpenCV, PyTorch, dan Ultralytics YOLO. Menurut Goodfellow *et al.* (2016), ekosistem Python telah menjadi standar de facto dalam pengembangan *deep learning* karena fleksibilitas dan kemudahan integrasinya. Dalam sistem ini, Python memungkinkan proses deteksi manusia, pengolahan logika keputusan, serta pengiriman perintah kontrol ke ESP32 dilakukan dalam satu lingkungan pemrograman yang terintegrasi.

2.4.5 GitHub

GitHub merupakan *platform* version control berbasis Git yang digunakan untuk mengelola kode Sumber secara terstruktur dan kolaboratif. Dalam tugas akhir ini, GitHub berfungsi sebagai media penyimpanan dan dokumentasi *source* code sistem, mulai dari program YOLO, *backend* web, hingga firmware ESP32. GitHub mendukung penerapan *version control* sehingga setiap pembaruan kode, perbaikan bug, maupun penyesuaian parameter sistem dapat direkam secara kronologis. Hal ini sangat penting dalam penelitian berbasis eksperimen, terutama pada proses pelatihan model YOLO dan integrasi dengan sistem IoT, karena memungkinkan evaluasi performa sistem dilakukan secara konsisten pada setiap tahapan pengembangan. Selain sebagai repositori kode, GitHub juga dimanfaatkan untuk menyimpan dokumentasi teknis sistem, seperti struktur folder dataset, konfigurasi pelatihan model, serta panduan implementasi perangkat lunak.



Gambar 2.24. Tampilan Data File GitHub
Sumber: Pribadi

Pada gambar 2.24 saya menggunakan github, agar memungkinkan setiap perubahan code terdokumentasi dengan baik, sehingga memudahkan proses debugging dan pengembangan bertahap. Menurut Chacon dan Straub (2014), *version control* merupakan praktik penting dalam rekayasa perangkat lunak modern karena mampu meningkatkan kualitas dan keterlacakan pengembangan sistem. Dalam konteks akademis, GitHub juga menunjukkan bahwa sistem dikembangkan secara sistematis dan profesional.

2.4.6 Hive MQ

HiveMQ merupakan platform MQTT Broker berbasis cloud yang berfungsi sebagai perantara komunikasi (*message broker*) antar perangkat *Internet of Things* (IoT) menggunakan protokol *Message Queuing Telemetry Transport* (MQTT). Pada gambar 2.25, HiveMQ digunakan sebagai media pertukaran data antara *backend* berbasis *Flask* yang berjalan pada laptop dengan mikrokontroler ESP32. Melalui mekanisme publish dan subscribe, hasil deteksi objek dari model YOLO, status perangkat, serta perintah pengendalian lampu dan kipas dapat dikirimkan secara real-time tanpa memerlukan komunikasi langsung antar perangkat. Penggunaan HiveMQ memungkinkan sistem tetap dapat beroperasi secara stabil meskipun perangkat berada pada jaringan yang berbeda, sehingga mendukung implementasi sistem kontrol cerdas berbasis IoT di Masjid Bahrul Ulum.

Selain sebagai media komunikasi data, HiveMQ juga mendukung proses monitoring dan sinkronisasi informasi secara real-time antara mini PC, ESP32, dan antarmuka web monitoring. Setiap perubahan kondisi sistem, seperti hasil deteksi jumlah jamaah, intensitas cahaya, status lampu, kecepatan kipas, maupun kondisi

perangkat lainnya dapat dipublikasikan ke broker dan diterima oleh perangkat yang berlangganan (subscriber) sesuai topik (topic) yang telah ditentukan.



Gambar 2.25. Tampilan *Software HiveMQ*
Sumber: Pribadi

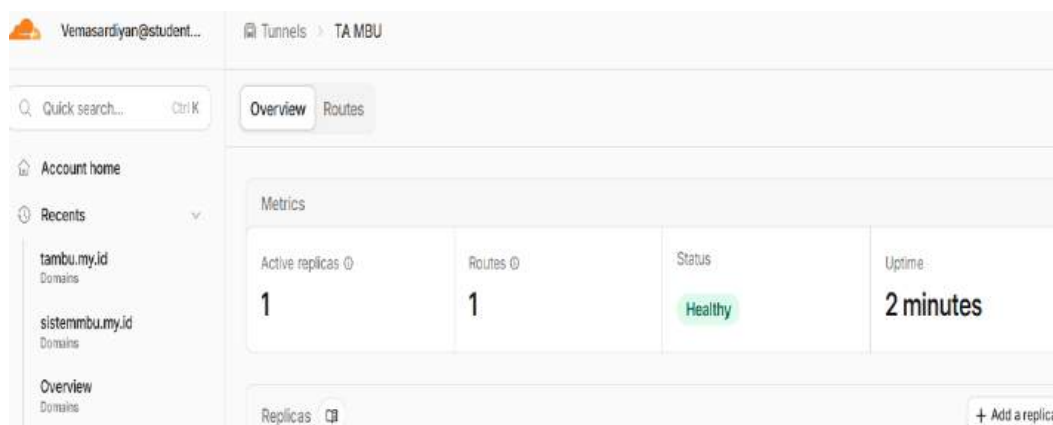
Mekanisme ini menghasilkan komunikasi yang ringan, cepat, serta efisien dalam penggunaan bandwidth sehingga sangat sesuai untuk implementasi sistem IoT yang memerlukan pertukaran data secara terus-menerus. Dengan memanfaatkan layanan HiveMQ sebagai MQTT Broker berbasis cloud, penelitian ini memperoleh media komunikasi yang andal dalam menghubungkan *Mini PC* sebagai pusat pemrosesan citra berbasis YOLO, ESP32 sebagai pengendali perangkat, serta sistem web monitoring yang diakses melalui jaringan internet.

2.4.7 CloudFlare Tunnels

Cloudflare Tunnel merupakan layanan secure tunneling yang memungkinkan server lokal diakses melalui internet tanpa memerlukan konfigurasi port forwarding, alamat IP publik, maupun layanan tunneling sementara seperti Ngrok. Pada penelitian ini, Cloudflare Tunnel berperan sebagai penghubung utama antara server lokal yang menjalankan aplikasi monitoring berbasis Python Flask dengan jaringan internet melalui domain. Dengan mekanisme tunneling yang terenkripsi, seluruh komunikasi data antara pengguna dan server dapat berlangsung secara aman, stabil, dan memiliki tingkat ketersediaan yang tinggi.

Dalam implementasi sistem tugas akhir ini, Cloudflare Tunnel digunakan untuk mempublikasikan aplikasi web monitoring dan kontrol sehingga dapat diakses secara real-time melalui browser dari berbagai lokasi. Dashboard tersebut menampilkan hasil deteksi objek berbasis YOLO, kondisi sensor, status lampu dan

kipas, konsumsi energi listrik, serta informasi aktivitas yang terjadi di area Masjid Bahrul Ulum. Selain itu, Cloudflare Tunnel juga berfungsi sebagai jalur komunikasi antara antarmuka web dengan server Flask yang mengelola proses inferensi YOLO, pengolahan data sensor, penyimpanan basis data, serta pengiriman perintah kontrol menuju mikrokontroler ESP32 melalui protokol MQTT menggunakan broker HiveMQ.



Gambar 2.26. *Software Cloudflare Tunnels*
Sumber: Pribadi

Pada gambar 2.26, penggunaan Cloudflare Tunnel memberikan beberapa keuntungan dibandingkan metode tunneling konvensional, seperti tidak memerlukan pembukaan port pada router, mengurangi risiko serangan terhadap jaringan lokal, serta menyediakan koneksi yang lebih andal melalui infrastruktur global Cloudflare. Dengan demikian, sistem monitoring dan kontrol dapat diakses secara aman menggunakan domain publik tanpa harus mengekspos alamat IP lokal server. Arsitektur ini juga meningkatkan fleksibilitas implementasi karena server dapat dipindahkan ke perangkat lain tanpa perlu melakukan perubahan konfigurasi jaringan secara signifikan, selama konektor Cloudflare Tunnel tetap terhubung dengan akun dan domain yang digunakan.

(halaman ini sengaja dikosongkan)

BAB 3

METODOLOGI PENELITIAN

Pada Bab 3 membahas mengenai metodologi penelitian yang digunakan dalam perancangan dan implementasi *electrical facility control* menggunakan YOLO untuk *illegal activity detection* di Masjid Bahrul Ulum Politeknik Perkapalan Negeri Surabaya (PPNS). Pembahasan diawali dengan perumusan konsep penelitian yang menjelaskan alur kerja sistem secara keseluruhan, mulai dari proses akuisisi data citra menggunakan sensor kamera hingga pengambilan keputusan untuk mengendalikan fasilitas lampu dan kipas angin secara otomatis. Selanjutnya, disusun *Flowchart* sistem yang bertujuan untuk menggambarkan tahapan proses kerja sistem pengendalian secara terstruktur dan sistematis sesuai dengan mekanisme yang dirancang pada penelitian tugas akhir ini. Tahap berikutnya mencakup perancangan dan desain sistem, yang meliputi rancangan perangkat lunak (*software*) dan perangkat keras (*hardware*) yang digunakan, termasuk sistem, blok diagram, serta integrasi antar komponen pendukung.

Pada bagian akhir bab ini, disajikan metode YOLO dengan ekstensi yolov8, perencanaan pelaksanaan penelitian yang mencakup jadwal pengerjaan sebagai acuan tahapan penyelesaian tugas akhir, serta perencanaan anggaran penelitian yang bertujuan untuk mengidentifikasi estimasi biaya yang dibutuhkan dalam proses perancangan, pengujian, dan implementasi sistem pengendalian yang dikembangkan.

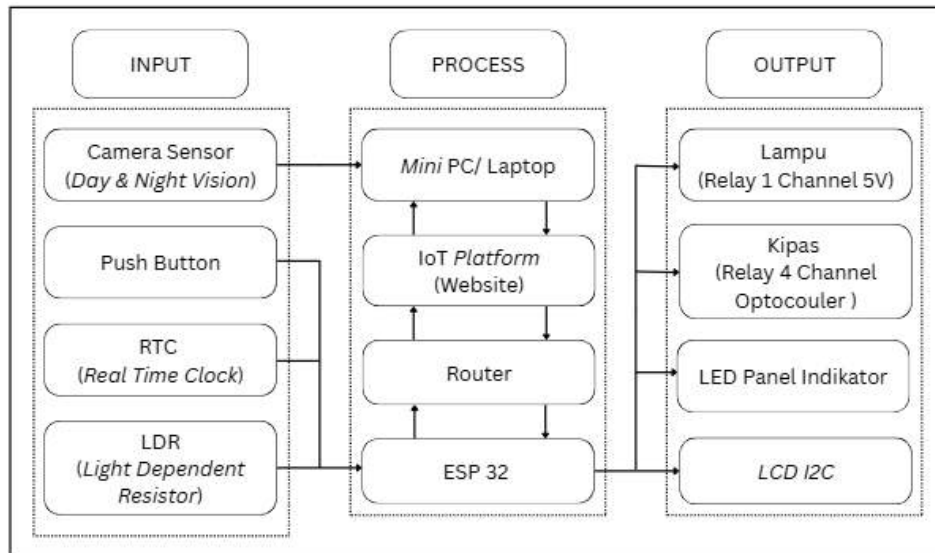
3.1 Konsep Penelitian

Pada sub bab ini akan membahas secara mendalam tentang konsep penelitian yang menjadi landasan utama untuk memahami konteks dan tujuan penelitian.

3.1.1 Diagram Blok Sistem

Pada Gambar 3.1 diagram blok sistem implementasi di Masjid Bahrul Ulum Politeknik Perkapalan Negeri Surabaya (PPNS) yang dirancang pada penelitian tugas akhir ini. Secara umum, sistem terbagi menjadi tiga bagian utama, yaitu bagian input, proses, dan output. Pembagian ini bertujuan untuk mempermudah

pemahaman alur kerja sistem serta menunjukkan hubungan antar komponen dalam mendukung pengendalian fasilitas masjid berbasis IoT menggunakan metode *You Only Look Once* (YOLO).

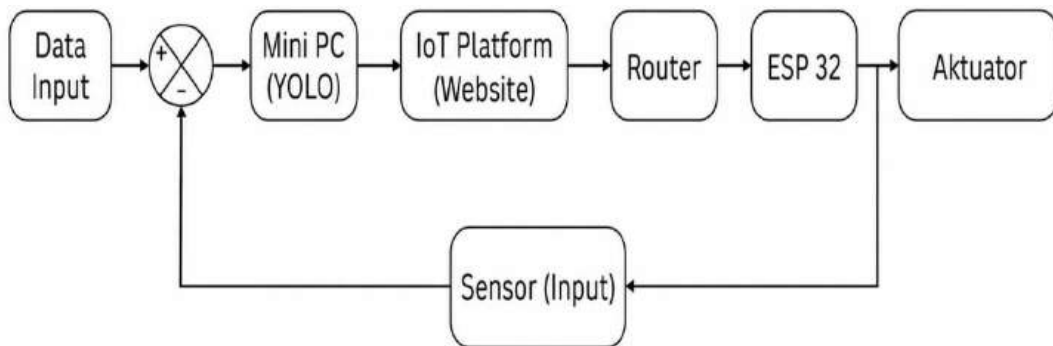


Gambar 3.1. Diagram Blok Sistem
Sumber: Pribadi

Bagian proses merupakan pusat pengolahan data dan pengambilan keputusan sistem. Pada bagian ini, laptop berperan sebagai unit pemrosesan utama untuk menjalankan algoritma YOLO dalam IoT dan *machine vision*. Data citra dari kamera diproses menggunakan metode YOLO untuk melakukan pemantauan aktivitas *human detection*, *illegal overnight stay*, dan *Indication of a person littering*. Hasil pemrosesan ini kemudian dikirimkan ke mikrokontroler ESP32 sebagai perintah kendali. Mikrokontroler ESP32 bertugas menerima data hasil pengolahan dari laptop, mengolahnya menjadi sinyal kontrol, serta mengatur komunikasi nirkabel antar modul ESP32 yang digunakan dalam sistem. Selain itu, ESP32 juga terhubung dengan *platform* IoT berbasis website yang berfungsi sebagai antarmuka monitoring dan kontrol jarak jauh. Melalui *platform* ini, pengguna dapat memantau kondisi sistem, jumlah jamaah yang terdeteksi, *Indication of a person littering*, *illegal overnight stay*, serta status lampu dan kipas secara *real-time*.

Pada gambar 3.2 juga diperjelas kembali mengenai koneksi dari sistem *wireless* yang digunakan, dimana Bagian output (aktuator) merupakan perangkat

yang dikendalikan langsung oleh sistem berdasarkan hasil pengolahan data pada bagian proses.



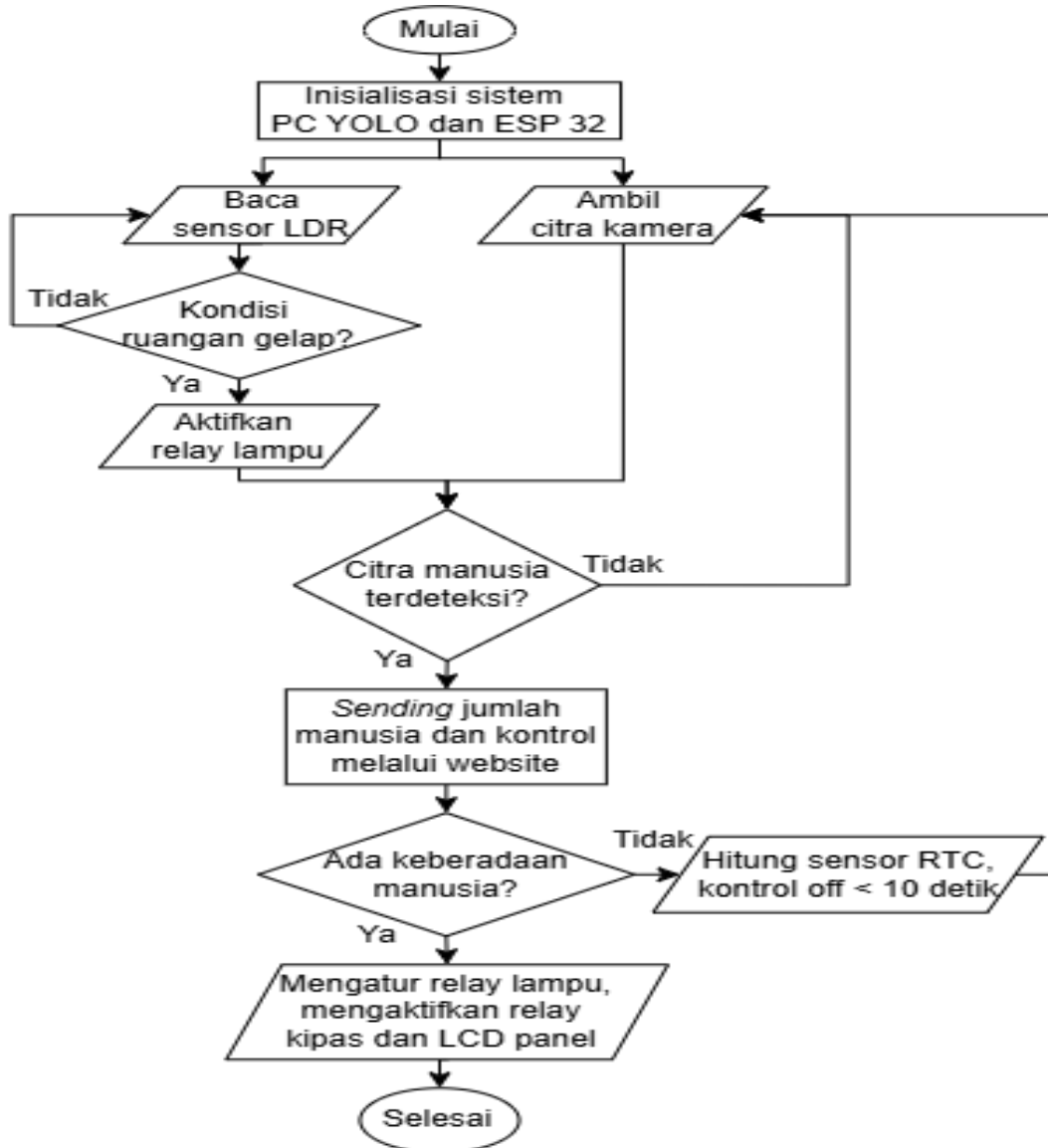
Gambar 3.2. Diagram Blok Sistem *Wireless*
Sumber: Pribadi

Lampu dikendalikan menggunakan *Relay* yang berfungsi sebagai penghubung antara sistem kontrol dan beban listrik, sehingga pengoperasian lampu dapat dilakukan secara aman dan andal. Kipas angin juga dikendalikan menggunakan *Relay* dengan tipe yang sama namun dengan tambahan 3 channel untuk kontrol *speed* 1, 2, dan 3 serta memastikan kestabilan dan keamanan sistem, baik pada kondisi ON maupun OFF. Selain itu, sistem dilengkapi dengan LED panel indikator yang berfungsi sebagai penanda visual status sistem, seperti kondisi aktif, tidak aktif, atau proses deteksi sedang berlangsung. *Relay* 4 channel digunakan sebagai perantara sinyal kontrol dari ESP32 menuju beban listrik tertentu untuk memastikan proses *switching* berjalan cepat, Minim percikan, dan aman untuk penggunaan jangka panjang namun akan terdengar suara "klik" saat *switching*.

3.1.2 Flowchart Kerja Sistem

Gambar 3.3 adalah kerja sistem pada penelitian tugas akhir ini menggambarkan alur operasional sistem *electrical facility control*. Alur sistem dimulai dari proses inialisasi, pengambilan data sensor, pengolahan citra, hingga pengambilan keputusan untuk mengendalikan fasilitas masjid secara otomatis dan terintegrasi. Proses diawali dengan sistem dalam kondisi mulai (*start*), kemudian dilakukan inialisasi seluruh perangkat yang digunakan, yaitu laptop sebagai prosesor YOLO dan mikrokontroler ESP32 sebagai pengendali sistem. Pada tahap ini, sistem memastikan bahwa koneksi antara laptop, ESP32, sensor, dan perangkat output berada dalam kondisi siap digunakan. Setelah proses inialisasi selesai,

sistem mulai membaca data dari sensor LDR untuk mengetahui kondisi intensitas cahaya di dalam ruangan, serta secara bersamaan mengambil citra dari sensor kamera yang terpasang di area pengamatan lantai dua masjid.



Gambar 3.3. Flowchart Kerja Sistem
Sumber: Pribadi

Selanjutnya, sistem melakukan pengecekan kondisi ruangan berdasarkan data dari sensor LDR untuk menentukan apakah ruangan berada dalam kondisi gelap atau terang. Apabila ruangan terdeteksi dalam kondisi gelap, maka sistem akan mengaktifkan *Relay* lampu untuk menyalakan penerangan ruangan. Setelah itu, citra hasil tangkapan kamera diproses menggunakan metode YOLO untuk mendeteksi keberadaan manusia. Jika citra manusia tidak terdeteksi, sistem akan

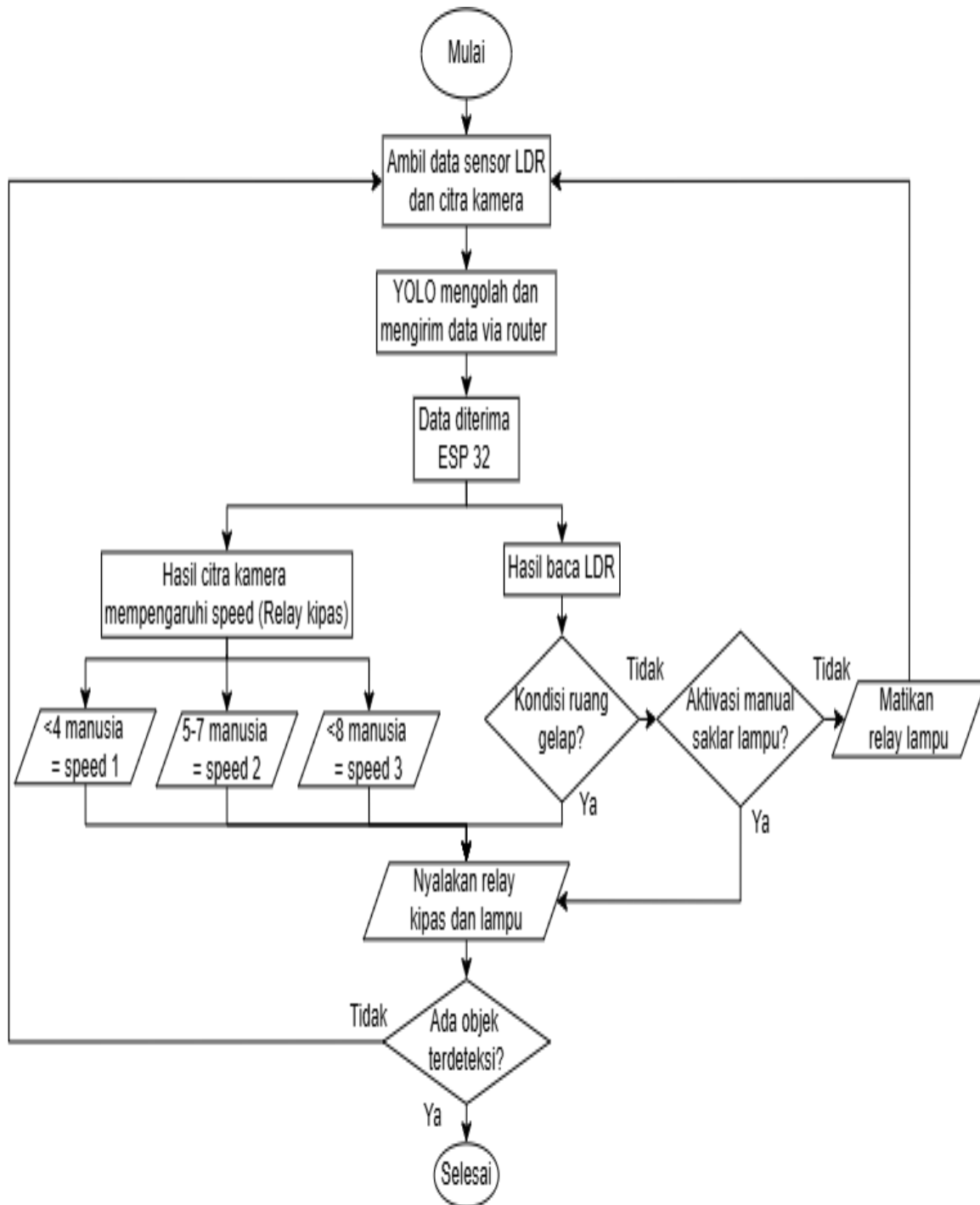
kembali ke proses pengambilan citra kamera dan pembacaan sensor, sehingga sistem terus melakukan pemantauan secara berulang.

Namun, apabila hasil pemrosesan citra menunjukkan bahwa manusia terdeteksi, maka sistem akan melanjutkan ke tahap pengendalian fasilitas. Pada tahap ini, sistem mengatur *Relay* 4 channel untuk mengaktifkan masing-masing *speed* kipas dan LED panel indikator sebagai penanda bahwa sistem dalam kondisi aktif. Setelah fasilitas diaktifkan, sistem akan mengirimkan informasi jumlah manusia yang terdeteksi serta status kontrol fasilitas ke *platform* berbasis website sebagai media monitoring. Tahap berikutnya adalah proses pengecekan lanjutan terhadap keberadaan manusia di dalam ruangan. Sistem akan terus memantau apakah masih terdapat aktivitas manusia berdasarkan hasil deteksi kamera. Jika masih terdapat manusia di area pengamatan, maka sistem akan tetap berada pada kondisi *looping* untuk melakukan pemantauan secara berkelanjutan. Sebaliknya, apabila tidak terdeteksi lagi keberadaan manusia, maka sistem akan memanfaatkan data dari sensor RTC untuk melakukan kontrol waktu tunda. Dalam kondisi ini, sistem akan mematikan lampu, kipas, dan LED panel secara otomatis setelah tidak terdeteksi aktivitas manusia selama waktu yang telah ditentukan, yaitu satu menit serta sistem akan melakukan *looping*, sistem *shutdown* apabila di kontrol oleh pengguna.

3.1.3 Flowchart Backend Sistem

Gambar 3.4 menggambarkan alur proses pengolahan data secara internal, mulai dari akuisisi data sensor hingga pengambilan keputusan untuk mengendalikan perangkat output secara otomatis. Proses *backend* sistem diawali dengan kondisi sistem mulai (*start*), kemudian sistem melakukan pengambilan data dari sensor LDR dan sensor kamera secara bersamaan. Data citra yang diperoleh dari kamera selanjutnya diproses menggunakan metode YOLO pada laptop untuk mendeteksi keberadaan manusia di dalam area pengamatan. Hasil pengolahan YOLO ini kemudian dikirimkan ke mikrokontroler ESP32 sebagai data perintah kontrol. Setelah data diterima, ESP32 melakukan inisialisasi perintah dan meneruskan informasi tersebut ke modul ESP32 lain yang berfungsi sebagai pengendali perangkat output. Gambar 3.4 dibawah ini, menggambarkan alur proses pengolahan data secara internal, mulai dari akuisisi data sensor hingga pengambilan

keputusan untuk mengendalikan perangkat output secara otomatis. Proses *backend* sistem diawali dengan kondisi sistem mulai (*start*), kemudian sistem melakukan pengambilan data dari sensor LDR dan sensor kamera secara bersamaan. Data citra yang diperoleh dari kamera selanjutnya diproses menggunakan metode YOLO pada laptop untuk mendeteksi keberadaan manusia di dalam area pengamatan.



Gambar 3.4. Flowchart Backend System
Sumber: Pribadi

Hasil pengolahan YOLO ini kemudian dikirimkan ke mikrokontroler ESP32 sebagai data perintah kontrol. Setelah data diterima, ESP32 melakukan inisialisasi perintah dan meneruskan informasi tersebut ke modul ESP32 lain yang berfungsi sebagai pengendali perangkat output.

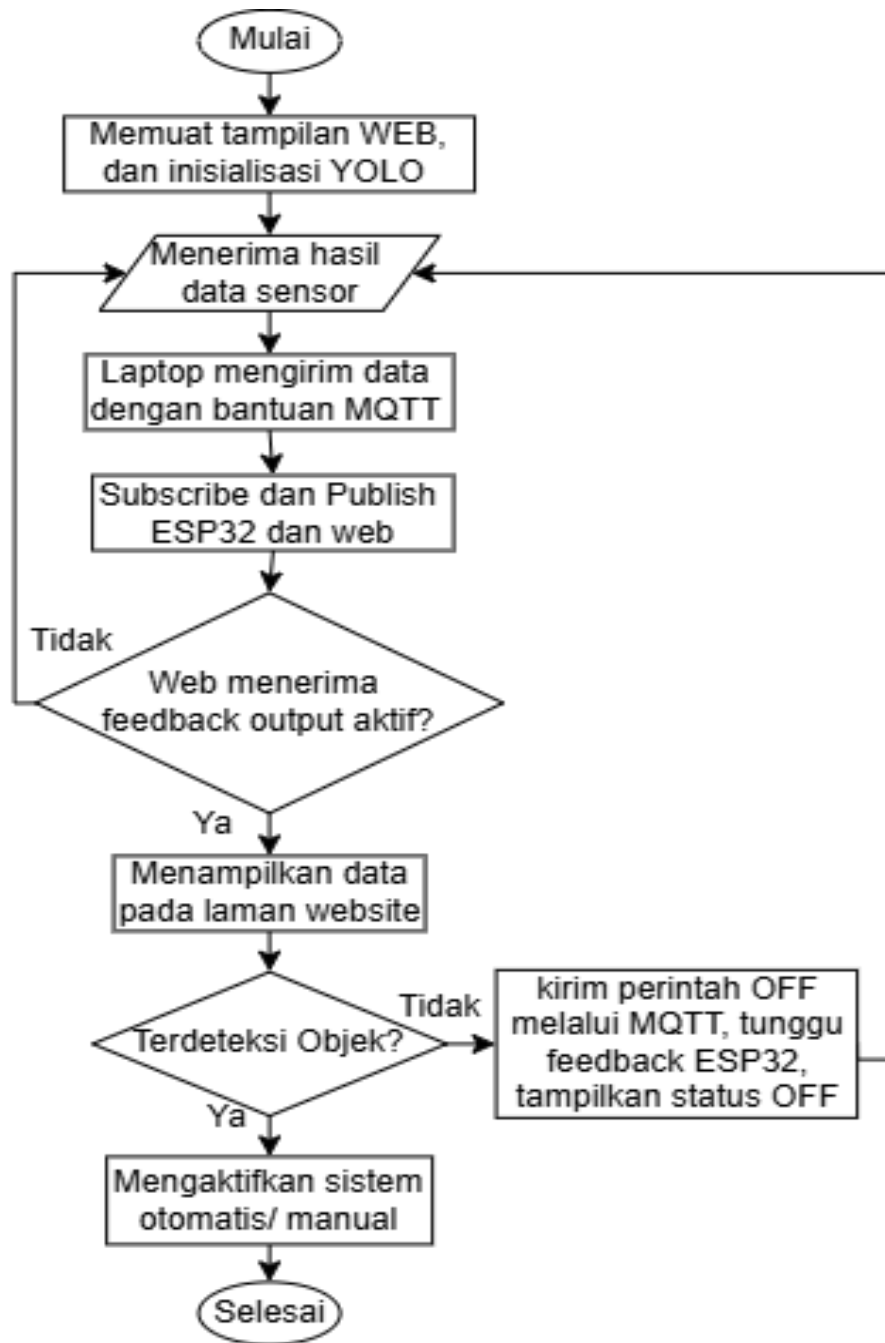
Berdasarkan hasil citra kamera, sistem menentukan jumlah manusia yang terdeteksi untuk mengatur kecepatan kipas angin 25 watt. Apabila jumlah manusia yang terdeteksi kurang dari empat orang, maka sistem akan mengatur kipas pada kecepatan rendah (*speed 1*, 1000-1150 rpm). Jika jumlah manusia lebih dari empat orang, kipas akan diatur pada kecepatan sedang (*speed 2*, 1150-1350 rpm), dan apabila jumlah manusia melebihi delapan orang, sistem akan meningkatkan kecepatan kipas ke tingkat tinggi (*speed 3*, 1350-1600 rpm). Pengaturan kecepatan ini dilakukan melalui pengendalian *Relay 4 channel* yang terhubung ke sistem penggerak kipas.

Selanjutnya, sistem membaca data dari sensor LDR untuk menentukan kondisi pencahayaan ruangan. Apabila ruangan terdeteksi dalam kondisi gelap, sistem akan melanjutkan ke proses pengecekan aktivitas manusia. Jika terdapat aktivitas manusia, maka sistem akan mengaktifkan *Relay* lampu dan kipas secara bersamaan. Sebaliknya, jika tidak terdapat aktivitas manusia atau tidak ada perintah dari steker manual, maka sistem akan mematikan *Relay* lampu untuk menghindari pemborosan energi. Setelah fasilitas diaktifkan, sistem akan melakukan pembacaan data dari sensor *Real Time Clock* (RTC) untuk mengatur waktu tunda pemadaman. Apabila dalam rentang waktu yang telah ditentukan tidak terdeteksi kembali keberadaan manusia, maka sistem akan melakukan proses output *shutdown*, yaitu mematikan lampu, kipas, dan indikator panel secara otomatis. Namun, jika masih terdeteksi aktivitas manusia, sistem akan kembali ke kondisi *looping* untuk terus melakukan pemantauan dan pengendalian fasilitas secara berkelanjutan.

3.1.4 Flowchart Frontend Sistem

Gambar 3.5 menunjukkan diagram alir sistem *frontend* yang berfungsi sebagai antarmuka pemantauan dan pengendalian *electrical facility control* berbasis IoT. Diagram ini menggambarkan alur kerja *frontend* mulai dari proses inisialisasi sistem hingga interaksi pengguna dalam melakukan monitoring maupun kontrol manual terhadap sistem. Proses diawali dengan sistem memulai (*start*), kemudian

frontend memuat tampilan antarmuka website sekaligus melakukan inisialisasi modul *You Only Look Once* (YOLO) pada laptop. Setelah antarmuka berhasil ditampilkan, sistem menerima data awal yang berasal dari sensor, baik berupa hasil pembacaan sensor lingkungan maupun data citra yang telah diproses oleh YOLO. Data tersebut kemudian dikirimkan oleh laptop ke modul ESP32 sebagai informasi kondisi sistem.



Gambar 3.5. Flowchart Frontend Sistem
Sumber: Pribadi

Selanjutnya, ESP32 menerima alamat IP dari router yang digunakan sebagai media komunikasi nirkabel, kemudian meneruskan data tersebut ke modul ESP32 lain yang berfungsi sebagai pengendali perangkat output. Setelah proses pengiriman data selesai, *frontend* menunggu umpan balik (*feedback*) berupa status aktivasi output dari sistem. Apabila website menerima informasi bahwa output dalam kondisi aktif, maka data kondisi sistem, seperti status lampu, kipas, dan indikator lainnya, akan ditampilkan secara *real-time* pada halaman website.

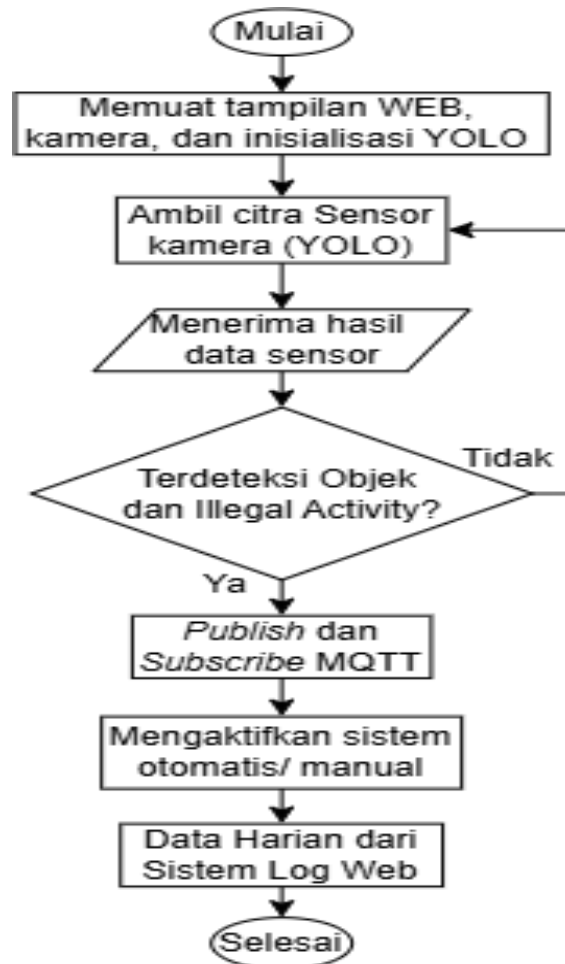
Pada tahap berikutnya, sistem *frontend* memeriksa apakah terdapat umpan balik dari sensor *Real Time Clock* (RTC) yang menandakan waktu pemadaman telah tercapai. Jika kondisi tersebut terpenuhi, *frontend* akan mengirimkan perintah untuk mematikan output secara otomatis. Selain itu, *frontend* juga menyediakan fitur kontrol manual berupa perintah sistem *Off* yang dapat diakses oleh pengguna melalui antarmuka website. Apabila pengguna mengaktifkan perintah tersebut, *frontend* akan mengirimkan sinyal ke sistem untuk melakukan pemadaman seluruh perangkat dan menghentikan proses kerja sistem. Apabila tidak terdapat perintah pemadaman dari RTC maupun dari pengguna, sistem akan kembali ke kondisi *looping*, di mana *frontend* terus melakukan pemantauan data sensor dan status sistem secara berkelanjutan. Proses ini berlangsung selama sistem dalam kondisi aktif. Diagram alir *frontend* ini menunjukkan bahwa antarmuka website tidak hanya berperan sebagai media visualisasi data, tetapi juga sebagai pusat kontrol yang memungkinkan sistem bekerja secara interaktif, responsif, dan terintegrasi dengan proses IoT yang berjalan di sisi *backend*.

3.1.5 Flowchart Illegal Activity

Berdasarkan hasil observasi lapangan serta wawancara dengan pengurus dan takmir Masjid Bahrul Ulum, diperoleh fakta bahwa indikasi perilaku membuang sampah (*littering*) merupakan permasalahan yang paling sering terjadi dan bersifat berulang, khususnya di area jamaah akhwat lantai dua. Temuan lapangan menunjukkan bahwa hampir setiap bulan terdapat sampah yang tertinggal di area shaf, seperti bungkus makanan, tisu, kertas, dan sisa konsumsi berukuran kecil. Kondisi ini dinilai lebih mengkhawatirkan dibandingkan aktivitas lain yang sebelumnya dipertimbangkan, karena:

1. Frekuensinya tinggi dan konsisten terjadi
2. Berdampak langsung pada kebersihan dan kenyamanan ruang ibadah
3. Sulit diawasi secara manual karena keterbatasan akses dan privasi area akhwat

Selain permasalahan kebersihan, perlunya upaya pencegahan terhadap potensi pemanfaatan ruang yang tidak sesuai peruntukan, khususnya indikasi *illegal overnight stay* di luar waktu operasional masjid.



Gambar 3.6. *Flowchart Illegal Activity*
Sumber: Pribadi

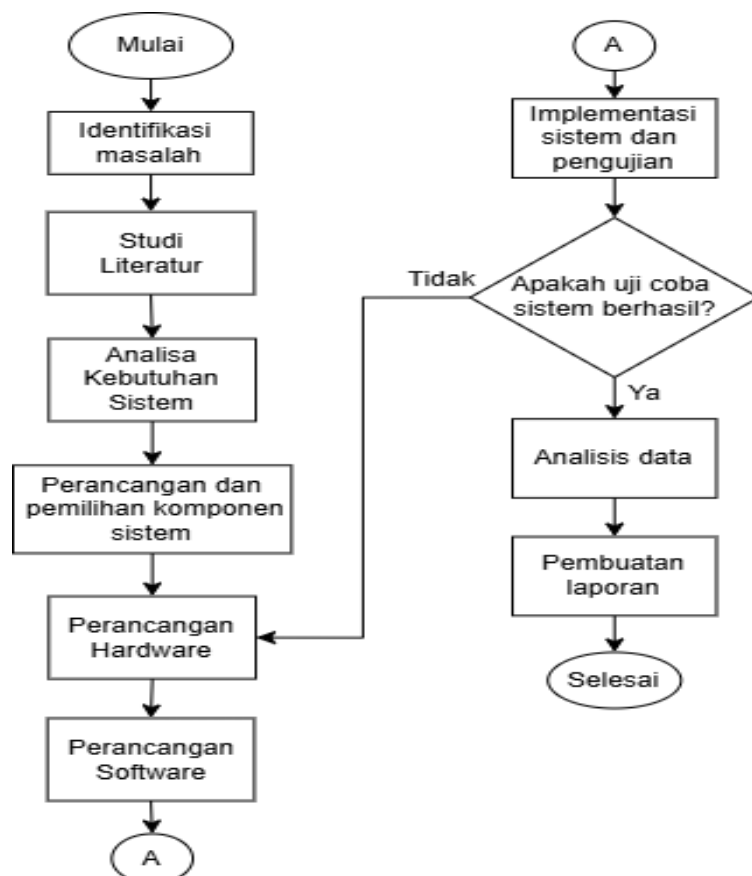
Digambarkan pada Gambar 3.6 sistem deteksi indikasi *person littering* dan *illegal overnight stay* dirancang dengan alur kerja bersifat *event-based*, di mana sistem hanya akan melakukan pencatatan ketika terdeteksi suatu kejadian tertentu. Proses diawali dengan sistem dalam kondisi aktif dan melakukan pengambilan citra secara kontinu menggunakan sensor kamera yang terhubung dengan modul deteksi berbasis YOLO. Setiap *frame* citra yang diperoleh dianalisis secara *real-time* oleh model YOLO untuk mengidentifikasi keberadaan manusia sebagai dasar *human*

presence detection, sekaligus mendeteksi pola postur dan rangkaian gerak yang mengindikasikan aktivitas membuang sampah atau keberadaan manusia dalam kondisi berbaring statis dalam durasi tertentu.

Pada tahap pengambilan keputusan, sistem melakukan klasifikasi terhadap hasil inferensi YOLO dengan mempertimbangkan informasi postur, durasi waktu, serta data *Real Time Clock* (RTC) untuk menentukan apakah aktivitas yang terdeteksi termasuk dalam kategori *Indication of a person littering* atau *illegal overnight stay*.

3.2 Tahapan Penelitian

Penelitian dalam tugas akhir ini bertujuan untuk merancang perangkat yang mampu mengintegrasikan data ke 2 ke dalam website berbasis IoT. Oleh karena itu untuk hasil yang baik maka diperlukan diagram alir untuk tugas akhir ini sesuai pada gambar 3.7 dibawah ini.



Gambar 3.7. Diagram Alir Tahapan Penelitian
Sumber: Pribadi

Dilakukan analisis kebutuhan sistem yang meliputi perancangan perangkat keras (*hardware*) dan perangkat lunak (*software*), diikuti dengan proses pembuatan alat. Setelah itu, komponen-komponen diintegrasikan sebelum memasuki tahap pengujian kinerja. Jika hasil pengujian sesuai dengan standar yang telah ditentukan, penelitian dilanjutkan dengan analisis dan penarikan kesimpulan hingga tugas akhir selesai. Seluruh proses ini mencakup pemahaman yang mendalam terhadap masalah, pengumpulan data yang terstruktur, analisis kebutuhan yang komprehensif, perancangan yang sistematis, pembuatan alat yang presisi, integrasi komponen yang teliti, pengujian kinerja yang cermat, serta analisis hasil yang mendalam untuk memastikan penyelesaian tugas akhir sesuai dengan tujuan yang telah ditetapkan.

3.3 Identifikasi Masalah

Tahap awal penelitian dimulai dengan mengidentifikasi permasalahan yang terjadi di lingkungan Masjid Bahrul Ulum, khususnya pada pengelolaan pencahayaan dan kipas angin di ruang dalam masjid dan masalah *littering*. Berdasarkan pengamatan langsung di lapangan, sistem pengendalian yang masih bersifat manual menyebabkan penggunaan energi kurang efisien, terutama ketika ruangan kosong namun lampu dan kipas masih menyala. Selain itu, tidak adanya mekanisme otomatis yang menyesuaikan kondisi jumlah jamaah membuat kenyamanan ruang belum optimal. Dari permasalahan tersebut, muncul kebutuhan akan sebuah sistem yang mampu bekerja secara cerdas, aman, dan berkelanjutan dengan memanfaatkan teknologi pengolahan citra dan sensor lingkungan.

3.4 Studi Literatur

Setelah permasalahan teridentifikasi, tahap selanjutnya adalah melakukan studi literatur untuk memperoleh dasar teori dan referensi teknis yang relevan dengan topik penelitian. Studi ini mencakup pemahaman mengenai konsep *You Only Look Once* (YOLO) untuk deteksi manusia, sistem kendali berbasis mikrokontroler ESP32, sensor LDR untuk pembacaan intensitas cahaya, serta penggunaan RTC sebagai pengatur waktu sistem. Literatur diperoleh dari jurnal ilmiah, buku referensi, dokumentasi teknis, dan tugas akhir terdahulu yang

memiliki keterkaitan. Tahap ini tidak hanya bertujuan mengumpulkan teori, tetapi juga untuk memahami kelebihan dan keterbatasan metode yang akan diterapkan.

3.5 Analisis Kebutuhan Sistem

Tahap analisis kebutuhan sistem dilakukan untuk menerjemahkan permasalahan dan hasil studi literatur ke dalam kebutuhan teknis yang lebih terstruktur. Pada tahap ini ditentukan kebutuhan seperti pada tabel 3.1 dibawah ini.

Tabel 3.1. Kebutuhan Sistem

No	Hardware	Software
1	ESP 32	Visual Studio Code
2	Laptop	Python
3	CCTV/ Sensor Kamera	Arduino IDE
4	Sensor RTC	Google Colab
5	Sensor LDR	Cloudflare Tunnels
6	<i>Power Supply</i> 10A 12V	Sketchup
7	LCD I2C	Hive MQ
8	<i>Panel Lamp</i>	Layanan Hosting
9	Terminal Blok	
10	MCB 4A	
11	DC-DC Stepdown 5V 3A	
12	Fuse 2A	
13	Router <i>wireless</i>	
14	<i>Relay</i> Optocoupler	

Selain itu, dianalisis pula kebutuhan sistem komunikasi berbasis jaringan agar sistem dapat dipantau dan dikendalikan melalui website.

3.6 Pemilihan Komponen Sistem

Berdasarkan hasil analisis kebutuhan, dilakukan perancangan awal sistem dan pemilihan komponen yang sesuai. Pada tahap ini ditentukan spesifikasi laptop dengan spesifikasi maksimal memakai gpu nvidia untuk menjalankan YOLO, jenis kamera yang mendukung kondisi siang dan malam, serta modul ESP32 yang digunakan sebagai penghubung komunikasi nirkabel. Pemilihan komponen juga mempertimbangkan aspek keamanan, keandalan jangka panjang, serta ketersediaan

dan biaya agar sistem tetap ekonomis namun fungsional. Tahap ini memastikan bahwa seluruh komponen yang digunakan saling mendukung dan dapat diintegrasikan tanpa menimbulkan konflik teknis.

3.7 Perancangan *Hardware*

Perancangan *Hardware* dilakukan dengan menyusun hubungan fisik antar komponen, mulai dari penempatan sensor, aktuator, hingga sistem catu daya. Pada tahap ini ditentukan posisi kamera, sensor LDR, dan modul ESP32 di area implementasi, serta penempatan perangkat pemrosesan di ruang kontrol. Selain itu, dirancang pula sistem pengendalian lampu dan kipas menggunakan komponen yang aman dan tahan lama untuk penggunaan terus-menerus. Perancangan *Hardware* ini bertujuan memastikan sistem dapat bekerja stabil, aman, dan mudah dirawat dalam jangka panjang.

3.8 Perancangan *Software*

Tahap perancangan *Software* mencakup pengembangan program untuk pengolahan citra berbasis YOLO, logika kendali pada ESP32, serta sistem komunikasi data ke website. Pada bagian ini dirancang alur logika pengambilan keputusan, mulai dari deteksi manusia, perhitungan jumlah jamaah, hingga penentuan kondisi lampu dan kecepatan kipas. Perancangan *Software* dilakukan secara bertahap agar setiap fungsi dapat diuji secara terpisah sebelum diintegrasikan, sehingga meminimalisir kesalahan pada saat sistem dijalankan secara keseluruhan.

3.9 Implementasi Sistem dan Pengujian

Setelah perancangan *Hardware* dan *Software* selesai, tahap berikutnya adalah implementasi sistem secara nyata sesuai dengan rancangan yang telah dibuat. Pada tahap ini seluruh komponen dirakit, program diunggah, dan sistem mulai dijalankan di lingkungan pengujian. Pengujian dilakukan untuk memastikan setiap bagian sistem bekerja sesuai fungsi, mulai dari sensor, pengolahan data, hingga aktuator. Pengujian ini dilakukan berulang kali untuk melihat respon sistem terhadap berbagai kondisi, seperti perubahan jumlah jamaah, *illegal activity*, kondisi terang, dan gelap, serta interaksi manual dari pengguna.

3.10 Evaluasi Keberhasilan Pengujian

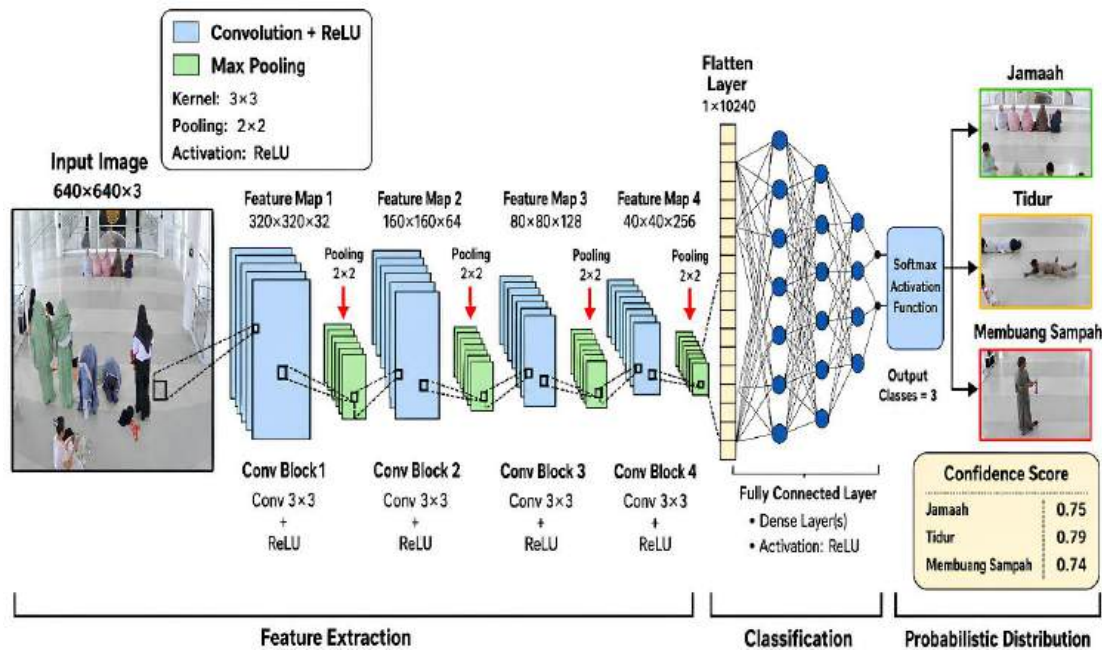
Hasil pengujian kemudian dievaluasi untuk menentukan apakah sistem telah bekerja sesuai dengan tujuan penelitian. Jika pada tahap ini ditemukan ke tidak sesuaian atau kesalahan dalam respon sistem, maka dilakukan perbaikan pada tahap perancangan *Hardware* atau *software*. Proses ini bersifat iteratif hingga sistem mampu bekerja secara stabil dan konsisten. Tahap evaluasi ini penting untuk memastikan bahwa sistem yang dihasilkan memang benar benar bisa mendeteksi setiap aktivitas yang dirasa ilegal dan bukan hanya berfungsi secara teknis, tetapi juga layak diterapkan di lingkungan nyata.

3.11 Analisis Data

Setelah sistem dinyatakan berjalan dengan baik, dilakukan analisis data dari hasil pengujian. Data yang dianalisis meliputi akurasi deteksi manusia, aktivitas yang terdeteksi termasuk dalam kategori *Indication of a person littering* atau *illegal overnight stay*, respon sistem terhadap perubahan kondisi, serta efisiensi pengendalian lampu dan kipas. Analisis ini bertujuan untuk menilai sejauh mana sistem mampu menyelesaikan permasalahan yang telah diidentifikasi pada tahap awal penelitian, sekaligus mengukur kelebihan dan keterbatasan sistem yang dibangun.

3.12 Ekstensi YOLOv8

Sesuai dengan gambar 3.8 metode utama yang digunakan adalah IoT berbasis YOLOv8 (*You Only Look Once version 8*) sebagai ekstensi deteksi objek. Redmon *et al.* (2018) menyatakan bahwa YOLO sebagai metode *object detection* berbasis *deep learning* memiliki keunggulan dalam melakukan ekstraksi fitur spasial dan klasifikasi objek secara end-to-end dalam satu tahap pemrosesan, sehingga memungkinkan deteksi objek manusia yang cepat dan efisien tanpa memerlukan perancangan fitur manual, serta sangat sesuai untuk aplikasi pengawasan berbasis kamera. Arsitektur model pada penelitian ini dirancang untuk melakukan pengenalan aktivitas pada area masjid berdasarkan citra dari kamera CCTV. Secara umum, arsitektur dibagi menjadi tiga bagian utama, yaitu feature extraction, classification, dan probabilistic distribution. Input sistem berupa citra berwarna hasil tangkapan CCTV yang kemudian diproses secara bertahap untuk mengenali tiga kelas aktivitas, yaitu jamaah, tidur, dan membuang sampah.



Gambar 3.8. Arsitektur YOLO
Sumber: Pribadi

Arsitektur model pada penelitian ini terdiri atas tiga tahapan utama, yaitu *feature extraction*, *classification*, dan *probabilistic distribution* yang di jelaskan seperti dibawah ini:

1. Input Citra

Tahap pertama adalah input image, yaitu citra masjid yang digunakan sebagai masukan ke dalam model. Pada arsitektur yang ditunjukkan, citra masukan memiliki ukuran $640 \times 640 \times 3$. Ukuran 640×640 menunjukkan dimensi panjang dan lebar citra, sedangkan angka 3 menunjukkan bahwa citra memiliki tiga kanal warna, yaitu *Red*, *Green*, *Blue* (RGB). Penggunaan ukuran input yang seragam bertujuan agar proses komputasi pada jaringan menjadi konsisten serta memudahkan model dalam mengekstraksi pola visual dari seluruh data.

2. Tahap Feature Extraction

Merupakan inti awal dari jaringan karena berfungsi untuk mengambil ciri-ciri penting dari citra input. Pada gambar arsitektur, proses ekstraksi fitur dilakukan melalui beberapa convolution block yang masing-masing terdiri dari convolution 3×3 , aktivasi ReLU, dan max pooling 2×2 .

a. Convolution

Pada setiap convolution block, citra diproses menggunakan kernel/filter berukuran 3×3 . Proses konvolusi bertujuan untuk menangkap pola-pola

visual lokal pada citra, seperti tepi, sudut, tekstur, kontur objek, dan bentuk tubuh. Pada layer awal, filter cenderung mempelajari ciri dasar, misalnya perbedaan warna lantai, batas tubuh manusia, dan pola sederhana lainnya.

b. Aktivasi ReLU

Setelah proses konvolusi, hasilnya dilewatkan ke fungsi aktivasi ReLU (Rectified Linear Unit). Fungsi aktivasi ini digunakan untuk memberikan non-linearitas pada model, sehingga jaringan tidak hanya mempelajari hubungan linear sederhana. ReLU bekerja dengan mempertahankan nilai positif dan mengubah nilai negatif menjadi nol.

c. Max Pooling

Setelah konvolusi dan aktivasi, dilakukan max pooling 2×2 . Pooling berfungsi untuk mengurangi ukuran spasial feature map tanpa menghilangkan informasi utama yang paling penting. Dengan reduksi ukuran ini, beban komputasi menjadi lebih ringan, model menjadi lebih efisien, dan sensitivitas terhadap pergeseran kecil pada objek dapat dikurangi.

3. Perkembangan Feature Map

Hasil dari proses ekstraksi fitur divisualisasikan dalam bentuk beberapa feature map. Sesuai gambar, jumlah dan ukuran feature map berubah secara bertahap sebagai berikut:

- Feature Map 1 : $320 \times 320 \times 32$
- Feature Map 2 : $160 \times 160 \times 64$
- Feature Map 3 : $80 \times 80 \times 128$
- Feature Map 4 : $40 \times 40 \times 256$

Perubahan ini menunjukkan model secara bertahap mengurangi detail lokasi piksel yang tidak terlalu penting, tetapi memperkaya representasi ciri yang lebih abstrak dan lebih diskriminatif. Pada Feature Map 1, model mulai menangkap fitur dasar dari citra input. Pada Feature Map 2 dan Feature Map 3, model mulai mengenali susunan pola yang lebih kompleks, seperti bentuk tubuh manusia, arah posisi tubuh, dan kemiripan bentuk objek tertentu. Kemudian pada Feature Map 4, model telah memperoleh representasi fitur yang lebih tinggi, sehingga siap digunakan untuk proses klasifikasi akhir.

4. Flatten Layer

Pada gambar arsitektur, hasil flatten ditunjukkan sebagai 1×10240 . Proses flatten bertujuan untuk mengubah data berbentuk matriks feature map menjadi bentuk vektor agar dapat diproses oleh lapisan klasifikasi. Secara teknis, flatten berfungsi sebagai jembatan antara bagian feature extraction dan bagian classification.

5. Fully Connected Layer

Fully connected layer berfungsi untuk mempelajari hubungan antar fitur yang telah diekstraksi sebelumnya, kemudian mengubahnya menjadi keputusan klasifikasi. Lapisan ini dapat dipahami sebagai tahap penggabungan informasi. Jika pada feature extraction model belajar “ciri-ciri apa yang muncul”, maka pada fully connected layer model belajar “kombinasi ciri seperti apa yang menunjukkan kelas jamaah, tidur, atau membuang sampah”. Aktivasi yang digunakan pada dense layer ini adalah ReLU, sehingga kemampuan model dalam membentuk hubungan non-linear antar fitur tetap dipertahankan.

6. Softmax Activation Function

Fungsi softmax bertugas mengubah nilai keluaran jaringan menjadi nilai probabilitas untuk masing-masing kelas. Dengan demikian, model dapat menentukan kelas mana yang memiliki tingkat keyakinan tertinggi.

7. Output Klasifikasi

Output ini menggambarkan bahwa model telah dirancang secara khusus untuk membedakan aktivitas normal di masjid dan aktivitas yang termasuk kategori pelanggaran atau aktivitas ilegal.

- a. Jamaah merepresentasikan individu yang berada dalam kondisi normal atau melakukan aktivitas ibadah.
- b. Tidur merepresentasikan individu yang berada dalam posisi tidur atau berbaring di area masjid.
- c. Membuang Sampah merepresentasikan aktivitas membuang sampah sembarangan di lingkungan masjid.

8. Confidence Score

Nilai tersebut menunjukkan tingkat keyakinan atau performa pengenalan model terhadap masing-masing kelas berdasarkan hasil evaluasi. Dari ketiga kelas

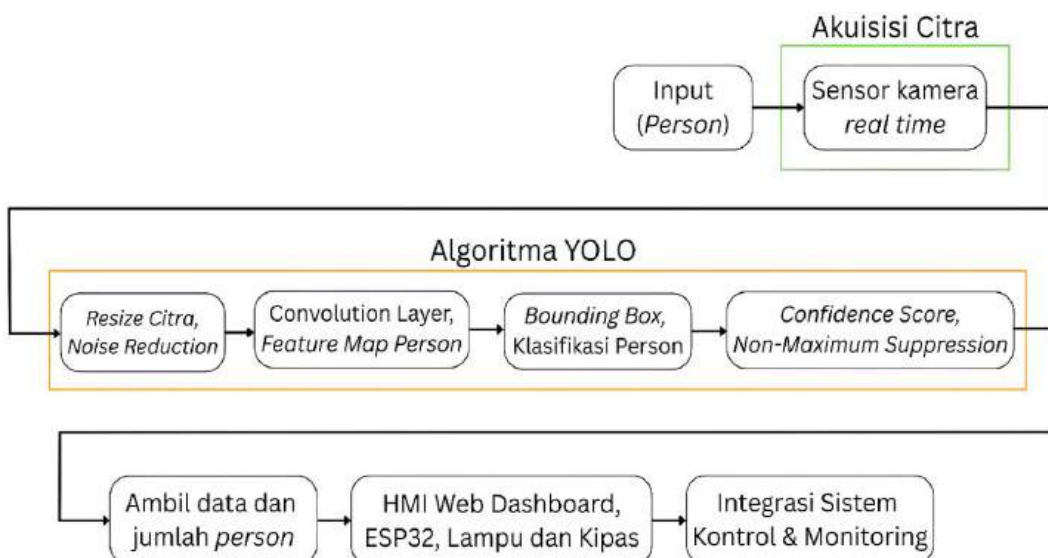
tersebut, kelas tidur memiliki nilai tertinggi, yaitu 0,79, yang menunjukkan bahwa model paling baik dalam mengenali pola visual aktivitas tidur. Sementara itu, kelas jamaah memiliki confidence score 0,75, dan kelas membuang sampah memiliki confidence score 0,74.

Pada gambar 3.9 dibawah ini, YOLO berfungsi sebagai inti pemrosesan visual yang menganalisis citra masjid untuk mendeteksi keberadaan manusia sebagai dasar pengambilan keputusan kontrol fasilitas.



Gambar 3.9. Hasil *Processing* Ekstensi YOLOv8
Sumber: Pribadi

YOLOv8 bekerja dengan membagi citra masukan menjadi beberapa grid dan secara simultan melakukan prediksi *bounding box*, kelas objek, serta nilai *confidence*. Proses ini memungkinkan sistem mendeteksi lebih dari satu manusia dalam satu frame citra tanpa perlu pemrosesan berulang.



Gambar 3.10. Diagram Blok Metode YOLO
Sumber: Pribadi

Implementasi metode YOLO sebagai dasar pengambilan data ekstensi objek ditunjukkan pada Gambar diagram 3.10. proses sistem diawali dari tahap.

1. Akuisisi Citra

Yaitu kamera menangkap objek manusia secara real-time dan mengirimkan setiap frame menuju algoritma YOLO. YOLO termasuk metode *single-stage object detection* karena proses penentuan lokasi objek dan klasifikasi dilakukan dalam satu jaringan secara langsung. Berbeda dengan metode yang lebih dahulu membuat kandidat wilayah objek, YOLO memproses keseluruhan citra dan menghasilkan koordinat *bounding box* beserta probabilitas kelas dalam satu kali proses inferensi. Karakteristik tersebut membuat YOLO sesuai diterapkan pada sistem monitoring yang membutuhkan respons cepat dari kamera menuju sistem kontrol.

2. Preprocessing

Sebelum citra diproses oleh jaringan, dilakukan tahap berupa penyesuaian ukuran atau *resize* citra agar sesuai dengan ukuran input model. Pada penelitian ini, ukuran citra disesuaikan melalui parameter *imgsz* sehingga setiap frame mempunyai dimensi yang seragam sebelum masuk ke model. Proses pengurangan gangguan atau *noise reduction* dapat ditempatkan sebagai preprocessing tambahan apabila citra kamera memiliki gangguan visual, pencahayaan tidak stabil, atau kualitas gambar yang rendah. Namun, *noise reduction* bukan lapisan wajib di dalam arsitektur YOLOv8, sehingga pada diagram lebih tepat dipahami sebagai proses pendukung sebelum citra memasuki jaringan utama. Pada proses inferensi Ultralytics, ukuran citra, nilai *confidence*, dan ambang IoU dapat diatur sesuai kebutuhan sistem.

3. Ekstraksi Fitur

Citra yang telah disesuaikan kemudian masuk ke bagian yang dituliskan sebagai *Convolution Layer-Feature Map Person*, jaringan konvolusi mempelajari karakteristik visual objek, seperti bentuk tubuh, pola posisi, kontur, tekstur, dan hubungan spasial antarbagian tubuh. Secara lebih teknis, YOLOv8 menggunakan susunan lapisan Conv, modul C2f, dan SPPF pada bagian *backbone*. Informasi fitur tersebut kemudian digabungkan pada beberapa skala melalui proses *upsampling*, *concatenation*, dan modul C2f

sebelum diteruskan menuju bagian deteksi. Penggunaan fitur pada beberapa skala membantu model mengenali objek manusia yang terlihat dekat, sedang, maupun relatif jauh dari kamera. Informasi fitur tersebut kemudian digabungkan pada beberapa skala melalui proses *upsampling*, *concatenation*, dan modul C2f sebelum diteruskan menuju bagian deteksi.

4. Pembentukan Bounding Box dan Klasifikasi Objek

Model memperkirakan koordinat kotak pembatas yang menunjukkan posisi manusia di dalam citra. Bersamaan dengan itu, model menentukan kelas aktivitas berdasarkan pola yang telah dipelajari selama proses training. Dalam penelitian ini, bagian “Klasifikasi *Person*” tidak hanya menunjukkan keberadaan manusia secara umum, tetapi mengelompokkan hasil deteksi ke dalam tiga kelas dataset, yaitu Jamaah, Tidur/Menginap, dan Buang Sampah.

5. Confidence Score

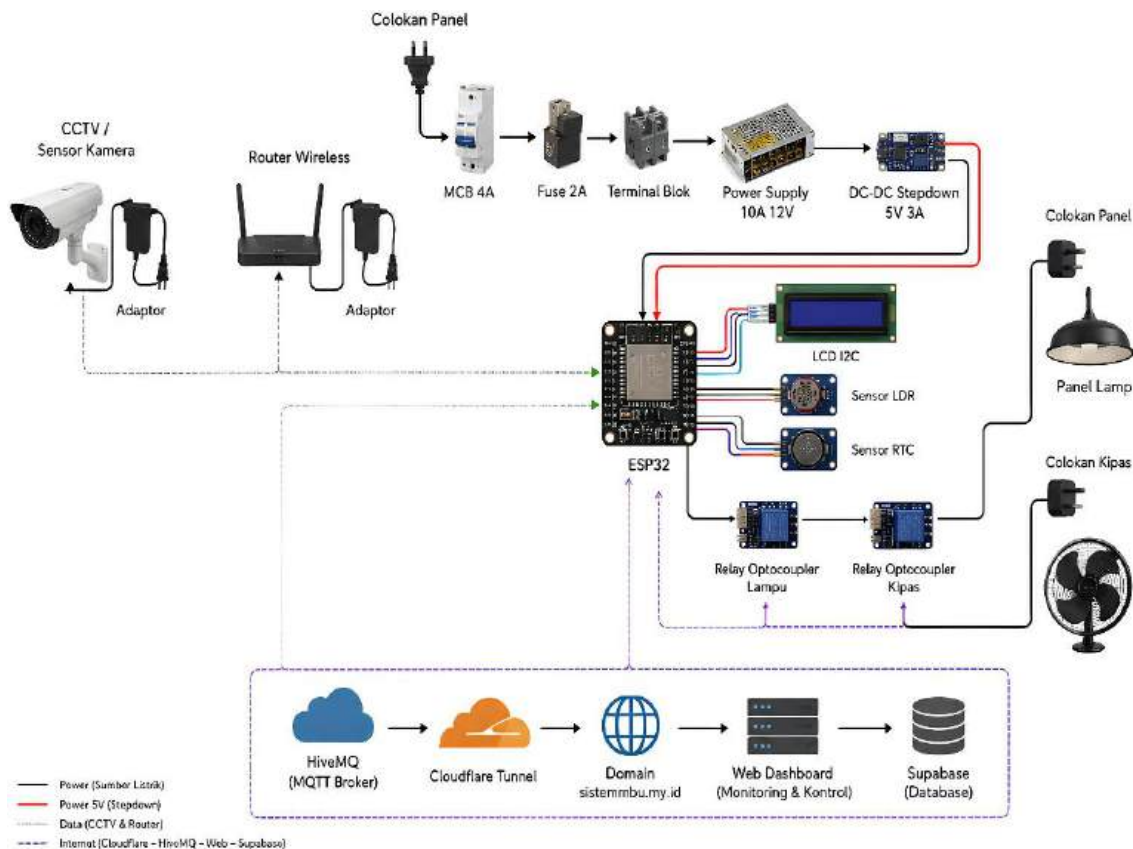
Yaitu nilai yang menunjukkan tingkat keyakinan model terhadap kelas dan lokasi objek yang terdeteksi. Prediksi dengan confidence di bawah ambang yang telah ditentukan dapat diabaikan untuk mengurangi hasil deteksi yang kurang meyakinkan. Selanjutnya, proses *Non-Maximum Suppression* (NMS) digunakan untuk mengurangi beberapa *bounding box* yang saling bertumpuk pada objek yang sama. NMS membandingkan tingkat tumpang tindih kotak menggunakan nilai *Intersection over Union* dan mempertahankan hasil deteksi yang paling relevan. Dalam implementasi Ultralytics, nilai *conf* digunakan sebagai batas minimum keyakinan, sedangkan nilai *iou* digunakan untuk mengatur proses NMS dalam menghilangkan kotak deteksi ganda. Setelah proses NMS selesai, sistem memperoleh keluaran akhir berupa jumlah manusia, lokasi *bounding box*, kelas aktivitas, dan confidence score.

3.13 Perancangan dan Implementasi

Setelah melatih model YOLOv8 di Google Colab dan memperoleh file “best.pt”, langkah berikutnya adalah mengimplementasikan model yang telah dilatih untuk mendeteksi objek produk secara *real-time*. Implementasi ini dapat dilakukan dengan menggunakan kamera (cctv). Perancangan hardware pada penelitian ini menggunakan arsitektur berbasis ESP32 sebagai pusat pengendali seluruh perangkat input, output, dan komunikasi IoT. Sistem dirancang agar

memiliki jalur distribusi daya yang aman, komunikasi data yang stabil, serta mudah dalam proses instalasi dan pemeliharaan. Secara keseluruhan, sistem terdiri atas tiga bagian utama, yaitu sistem catu daya (*power system*), sistem kontrol (*control system*), dan sistem komunikasi berbasis internet (*IoT system*).

Sesuai gambar 3.11, pada sisi catu daya panel, sumber listrik AC 220 V berasal dari colokan panel yang kemudian masuk menuju MCB 4A sebagai pengaman utama terhadap arus lebih (*overcurrent*) maupun hubungan singkat (*short circuit*). Setelah melewati MCB, arus listrik diteruskan menuju Fuse 2A sebagai pengaman tambahan untuk melindungi komponen elektronika apabila terjadi gangguan pada rangkaian kontrol. Dan pada perancangan dan desain ini penulis akan menjelaskan perancangan dan Implementasi yang akan dibuat sebagai berikut.



Gambar 3.11. Perancangan *Hardware* dan *Software*
Sumber: Pribadi

Selanjutnya, sumber listrik masuk ke Terminal Block yang berfungsi sebagai titik distribusi dan percabangan kabel sehingga proses pemasangan maupun perawatan panel menjadi lebih mudah. Dari terminal block, tegangan AC disalurkan

menuju Power Supply Switching 12 V 10 A yang mengubah tegangan AC menjadi tegangan DC 12 Volt sebagai sumber utama sistem kontrol. Output dari power supply kemudian masuk menuju DC-DC Step Down 5 V 3 A. Modul step-down ini berfungsi menurunkan tegangan dari 12 Volt menjadi 5 Volt yang sesuai dengan kebutuhan ESP32. Pada rancangan ini, step-down hanya terhubung ke ESP32, sehingga seluruh kebutuhan tegangan mikrokontroler diperoleh dari modul tersebut tanpa membebani perangkat lain.

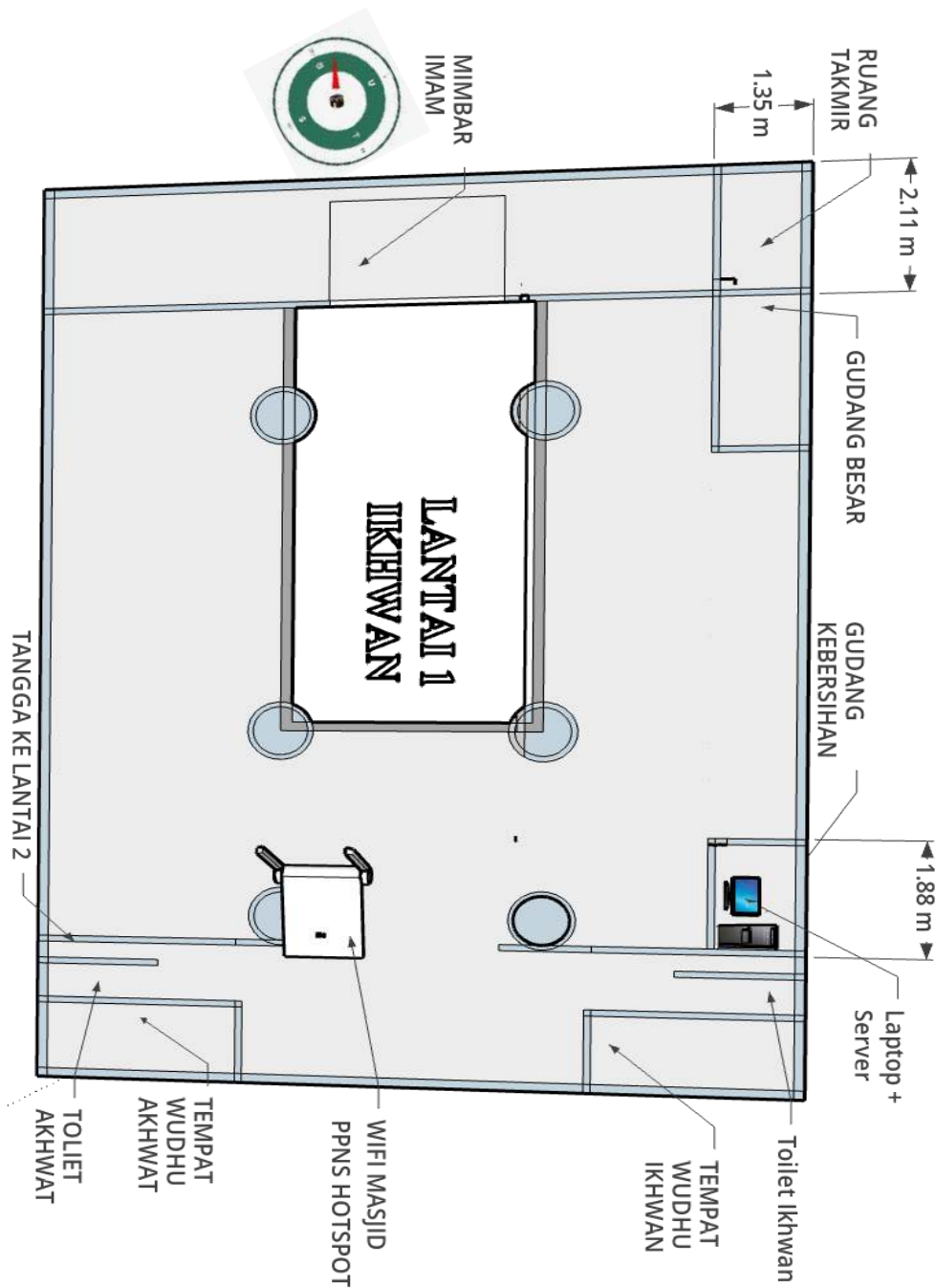
3.13.1 Penempatan *Hardware*

Gambar 3.12 dan gambar 3.13 Implementasi tugas akhir ini dirancang sebagai sistem jangka panjang yang berfokus pada efisiensi energi, keamanan operasional, dan kemudahan pengawasan, khususnya pada pengendalian pencahayaan *indoor* dan kipas angin di lingkungan Masjid Bahrul Ulum. Sistem dirancang sebagai sistem *monitoring* dan kontrol otomatis yang bekerja secara *real-time* dan berkelanjutan (*continuous operation*) untuk meningkatkan efisiensi penggunaan energi listrik, mempermudah pengawasan fasilitas masjid, serta mendukung pengendalian lampu dan kipas angin secara otomatis berdasarkan hasil deteksi objek menggunakan metode YOLOv8. Seluruh proses monitoring dan pengendalian dilakukan melalui website sehingga pengurus masjid dapat mengakses kondisi sistem maupun melakukan kontrol perangkat dari berbagai lokasi selama masih terhubung ke jaringan yang telah ditentukan.

Meskipun sistem telah dirancang untuk bekerja secara otomatis, implementasi penelitian ini masih difokuskan pada sebagian area ruang salat akhwat di lantai dua berdasarkan hasil observasi lapangan dan wawancara dengan pengurus serta jamaah akhwat. Tantangan utama dalam implementasi sistem ini terletak pada perencanaan jalur instalasi perangkat, penempatan panel kontrol, serta kestabilan komunikasi jaringan, terutama disaat jam istirahat dan pergantian waktu perkuliahan yang banyak meluangkan waktunya untuk beristirahat di masjid.

1. Area Monitoring dan Server Sistem (Lantai 1)

Area lantai satu berfungsi sebagai pusat akses monitoring sistem melalui jaringan lokal Masjid Bahrul Ulum. Pada penelitian ini tidak terdapat panel kontrol maupun perangkat sensor yang dipasang secara permanen di lantai satu.



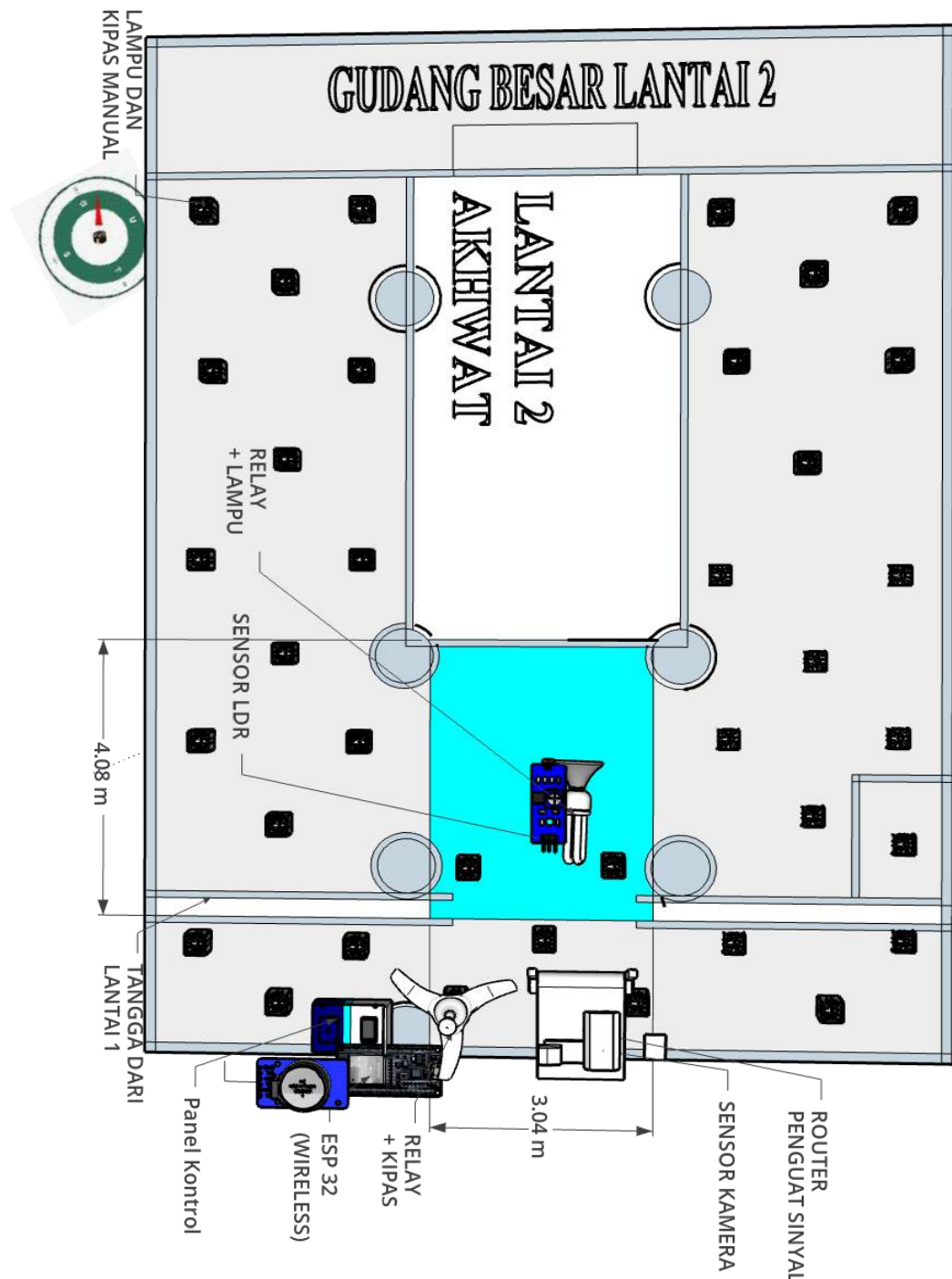
Gambar 3.12. Denah Implementasi *Hardware* Lantai 1 Masjid Bahrul Ulum
Sumber: Pribadi

Sebagai gantinya, sebuah laptop server digunakan untuk menjalankan aplikasi berbasis Python yang bertugas melakukan pengolahan citra menggunakan model YOLOv8, komunikasi MQTT melalui HiveMQ, penyimpanan data ke Supabase, serta menjalankan web monitoring. Akses tersebut dilakukan melalui domain “sistemmbu.my.id” yang dihubungkan menggunakan layanan Cloudflare Tunnel, sehingga web monitoring tetap

dapat diakses secara aman tanpa memerlukan konfigurasi *port forwarding* pada jaringan lokal masjid.

2. Area Implementasi Hardware dan Akuisisi Data (Lantai 2)

Lantai dua merupakan area utama implementasi perangkat keras karena seluruh proses akuisisi data dan pengendalian aktuator berlangsung pada area ini.



Gambar 3.13. Denah Implementasi *Hardware* Lantai 2 Masjid Bahrul Ulum
Sumber: Pribadi

Pada bagian belakang ruang salat akhwat ditempatkan sebuah panel kontrol yang berfungsi sebagai pusat distribusi daya sekaligus tempat pemasangan seluruh perangkat elektronika. Panel kontrol terdiri atas ESP32, *Relay* Lampu, *Relay* Kipas, LCD I2C, Sensor RTC, Lampu Indikator (Pilot Lamp), sistem distribusi daya, serta perangkat pendukung lainnya yang ditempatkan di dalam panel agar terlindungi dari gangguan lingkungan dan memudahkan proses pemeliharaan. Di area yang sama juga ditempatkan CCTV dan router TP-Link. CCTV digunakan sebagai sensor utama untuk memperoleh citra kondisi ruang salat secara *real-time*, sedangkan router berfungsi sebagai penghubung komunikasi antara CCTV, laptop server, ESP32, dan web *monitoring*. Kedua perangkat tersebut memperoleh sumber daya dari adaptor masing-masing yang langsung terhubung ke jaringan listrik sehingga tidak mengambil catu daya dari panel kontrol. Kamera dipasang pada posisi belakang ruangan dengan sudut pandang mengarah ke area tengah ruang salat akhwat sehingga mampu menjangkau sebagian besar aktivitas jamaah. Posisi tersebut dipilih untuk meminimalkan blind spot sekaligus memperoleh citra yang optimal bagi proses deteksi manusia, aktivitas tidur, maupun keberadaan sampah menggunakan algoritma YOLOv8

3. Penempatan Sensor dan Aktuator

Sensor LDR dipasang pada lokasi yang mampu menerima intensitas cahaya ruangan secara representatif sehingga pembacaan kondisi terang maupun gelap tidak dipengaruhi oleh pencahayaan lokal. Sensor ini berfungsi sebagai pendukung keputusan sistem dalam menentukan kebutuhan pencahayaan ruangan dan bekerja bersamaan dengan hasil deteksi kamera. Sensor RTC (*Real Time Clock*) digunakan sebagai pencatat waktu sistem sekaligus penghitung durasi ketika tidak terdapat manusia yang terdeteksi oleh kamera. Apabila dalam waktu satu menit tidak ditemukan aktivitas manusia, ESP32 secara otomatis memberikan perintah untuk mematikan lampu dan kipas. Selain itu, RTC juga digunakan sebagai pencatat waktu setiap aktivitas sistem yang selanjutnya dikirimkan ke database sebagai data histori monitoring. Sebagai aktuator utama, lampu dan kipas dipasang pada bagian tengah ruang salat akhwat agar distribusi pencahayaan maupun sirkulasi udara dapat

menjangkau area ibadah secara merata. Kedua perangkat memperoleh sumber listrik langsung dari instalasi listrik bangunan melalui jalur yang terpisah dari panel kontrol. Meskipun demikian, jalur fase menuju masing-masing beban tetap melewati *Relay* sehingga kondisi ON maupun OFF dapat dikendalikan sepenuhnya oleh ESP32 sesuai hasil pengolahan data sistem.

4. Sistem Komunikasi dan Monitoring

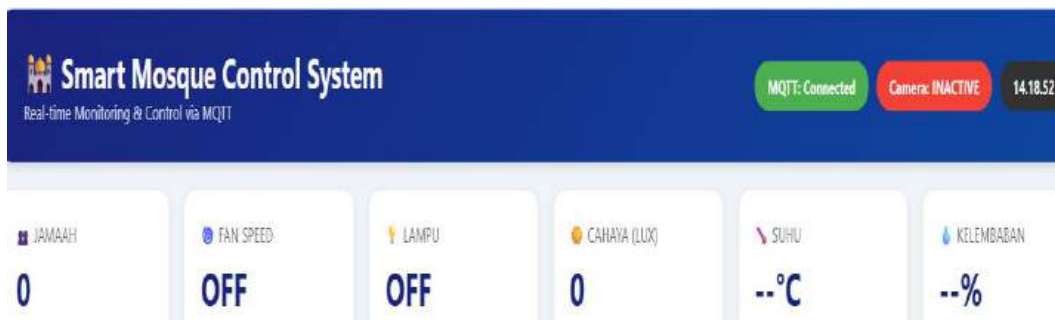
Komunikasi antarperangkat dilakukan menggunakan jaringan *Wireless Local Area Network* (WLAN) yang dibangun menggunakan router TP-Link. CCTV mengirimkan video menuju laptop server melalui jaringan lokal, kemudian laptop melakukan proses inferensi menggunakan model YOLOv8. Hasil deteksi selanjutnya dikirimkan menuju broker HiveMQ menggunakan protokol MQTT agar dapat diterima oleh ESP32 secara *real-time*. Selain mengendalikan aktuator, laptop juga mengirimkan data hasil monitoring menuju Supabase Database yang kemudian ditampilkan pada website *monitoring* melalui domain sistemmbu.my.id menggunakan Cloudflare Tunnel. Dengan arsitektur tersebut, pengurus masjid dapat memantau kondisi perangkat, jumlah jamaah, status lampu, status kipas, intensitas cahaya, serta melakukan kontrol manual terhadap aktuator melalui web dari lokasi mana pun selama server utama tetap aktif dan terhubung pada jaringan lokal router TP-Link.

3.13.2 Perancangan *Software*

Perangkat lunak yang ditunjukkan pada gambar 3.14 samapai dengan 3.16 merupakan aplikasi monitoring dan kontrol terintegrasi berbasis web yang berfungsi sebagai *Human Machine Interface* (HMI) pada sistem otomasi masjid yang dikembangkan dalam tugas akhir ini. Antarmuka ini dirancang untuk menjembatani interaksi antara sistem IoT berbasis YOLO dengan perangkat keras kendali seperti ESP32, *Relay* 1 channel, dan *Relay* 4 channel, sehingga seluruh proses pemantauan dan pengendalian fasilitas dapat dilakukan secara terpusat, *real-time*, dan aman.

1. Gambar 3.14 menunjukkan halaman utama *Web Human Machine Interface* (HMI) yang berfungsi sebagai pusat monitoring kondisi sistem secara real-time. Pada bagian header ditampilkan identitas sistem dengan judul Smart Mosque

Control System beserta indikator status konektivitas yang meliputi status MQTT, status kamera, dan alamat IP perangkat sebagai informasi kondisi komunikasi sistem. Terdapat indikator yang menampilkan data hasil pembacaan sensor dan proses deteksi, yaitu jumlah jamaah yang terdeteksi, status kecepatan kipas, status lampu, nilai intensitas cahaya (Lux), suhu ruangan, dan kelembaban.

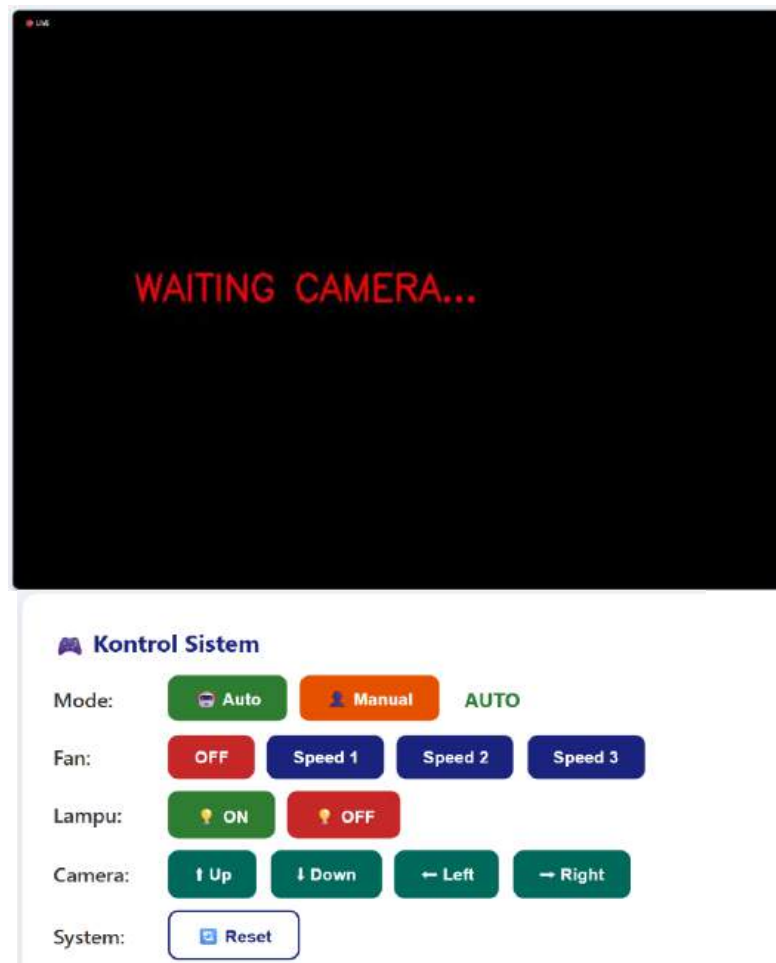


Gambar 3.14. Tampilan *Monitoring*
Sumber: Pribadi

Data tersebut diperoleh dari integrasi proses deteksi manusia menggunakan model YOLOv8 pada komputer pengolah citra serta pembacaan sensor lingkungan yang dikirimkan oleh modul ESP32 melalui protokol MQTT. Seluruh informasi diperbarui secara berkala sehingga kondisi sistem dapat dipantau secara langsung melalui satu halaman antarmuka. Integrasi antara hasil deteksi objek dan data sensor lingkungan memungkinkan sistem mengambil keputusan kontrol secara lebih akurat dibandingkan apabila hanya mengandalkan satu jenis masukan.

2. Gambar 3.15 merupakan halaman utama pengawasan yang menampilkan live streaming kamera sekaligus panel kendali sistem. Bagian tengah halaman digunakan sebagai area monitoring visual yang menampilkan hasil tangkapan kamera secara *real-time*. Apabila kamera belum aktif atau belum menerima aliran video, sistem akan menampilkan informasi "*Waiting Camera...*" sebagai indikator bahwa proses streaming masih menunggu koneksi dari perangkat kamera. Pada bagian bawah terdapat panel kontrol sistem yang digunakan untuk mengoperasikan perangkat secara manual maupun otomatis. Selain itu tersedia tombol reset untuk mengembalikan sistem ke kondisi awal apabila diperlukan. Fungsi ini memungkinkan Takmir melakukan intervensi secara langsung ketika diperlukan, sementara pada mode otomatis seluruh keputusan pengendalian

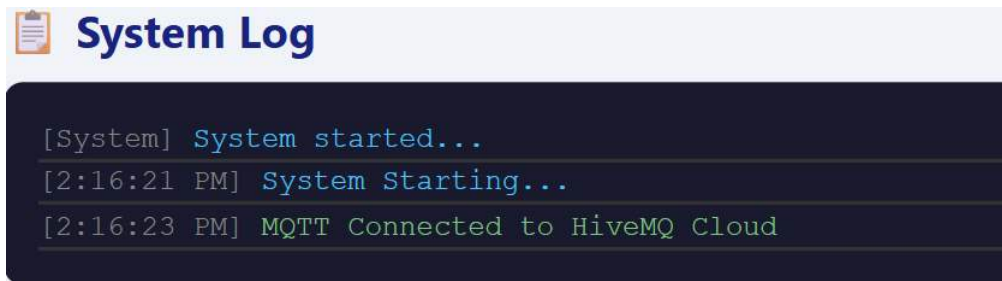
dijalankan berdasarkan hasil deteksi jumlah jamaah dan pembacaan sensor lingkungan.



Gambar 3.15. Tampilan *Monitoring* dan Kontrol
Sumber: Pribadi

Dengan ini, antarmuka ini tidak hanya berfungsi sebagai media pemantauan, tetapi juga sebagai sarana interaksi antara pengguna dengan sistem kendali berbasis *Internet of Things* (IoT).

3. Gambar 3.16 menunjukkan halaman System Log yang berfungsi sebagai media pencatatan aktivitas dan status komunikasi seluruh komponen sistem secara *real-time*. Informasi yang ditampilkan meliputi proses inisialisasi sistem, status koneksi MQTT, serta berbagai kejadian penting selama sistem beroperasi. Setiap aktivitas dicatat beserta *timestamp* sehingga memudahkan Takmir dalam mengetahui urutan proses yang terjadi sejak sistem dijalankan. Status log berperan sebagai indikator keberhasilan komunikasi antara komputer pengolah data, broker MQTT, dan modul ESP32.



Gambar 3.16. Tampilan Sistem Log
Sumber: Pribadi

Melalui tampilan ini Takmir dapat dengan mudah melakukan identifikasi apabila terjadi gangguan koneksi, kegagalan pengiriman data, maupun proses inisialisasi yang belum berhasil. Dengan adanya pencatatan aktivitas secara berkelanjutan, proses pemeliharaan, pengujian, dan evaluasi performa sistem menjadi lebih mudah dilakukan karena seluruh informasi operasional tersimpan dalam bentuk log yang dapat digunakan sebagai acuan analisis ketika terjadi kesalahan maupun saat proses pengembangan sistem di tahap selanjutnya.

BAB 4

HASIL DAN PEMBAHASAN

Pada bab ini berisi tentang hasil perancangan serta pengujian pada Tugas Akhir beserta analisisnya. Setelah merancang alat yang telah direncanakan, tahap selanjutnya adalah merealisasikan alat tersebut melalui pembuatan *hardware*, mekanik, serta *software*. Untuk memastikan bahwa setiap sistem yang dirancang mampu berfungsi sesuai dengan rencana, dilakukan serangkaian pengujian, khususnya terhadap fungsi pengendalian fasilitas listrik dan pengawasan aktivitas di Masjid Bahrul Ulum PPNS.

4.1 Hasil Perancangan *Hardware*

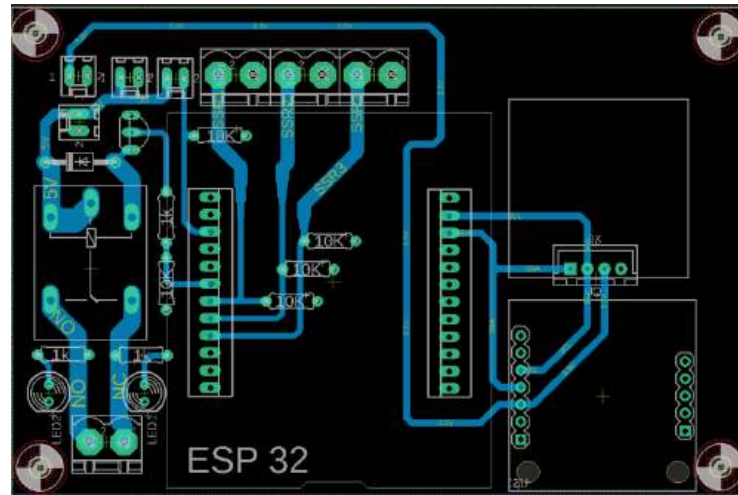
Perancangan *hardware* elektronika dilakukan dengan merancang konfigurasi beberapa sensor dan perangkat komunikasi yang berfungsi sebagai masukan (*input*) sistem, yaitu kamera CCTV sebagai sensor utama untuk memperoleh citra kondisi ruang salat akhwat yang selanjutnya diproses menggunakan metode YOLOv8, sensor LDR (*Light Dependent Resistor*) untuk mendeteksi intensitas cahaya ruangan, serta sensor RTC (*Real Time Clock*) untuk menyediakan informasi waktu dan pengaturan logika sistem berdasarkan durasi. Seluruh data hasil pembacaan sensor diproses oleh ESP32 sebagai pusat pengendali sistem sebelum diteruskan melalui komunikasi MQTT menggunakan HiveMQ untuk mendukung proses monitoring dan kontrol secara real-time.

Sebagai keluaran (*output*), sistem mengintegrasikan beberapa aktuator yang terdiri atas *Relay* lampu dan *Relay* kipas sebagai pengendali otomatis perangkat listrik berdasarkan hasil deteksi manusia dan kondisi lingkungan. Selain itu, LCD I2C digunakan sebagai media informasi untuk menampilkan status sistem pada panel. Seluruh komponen tersebut diintegrasikan dengan sistem monitoring berbasis web yang memanfaatkan Cloudflare Tunnel, Supabase Database, dan domain sistemmbu.my.id.

4.1.1 Hasil Pembuatan *Printed Circuit Board* (PCB)

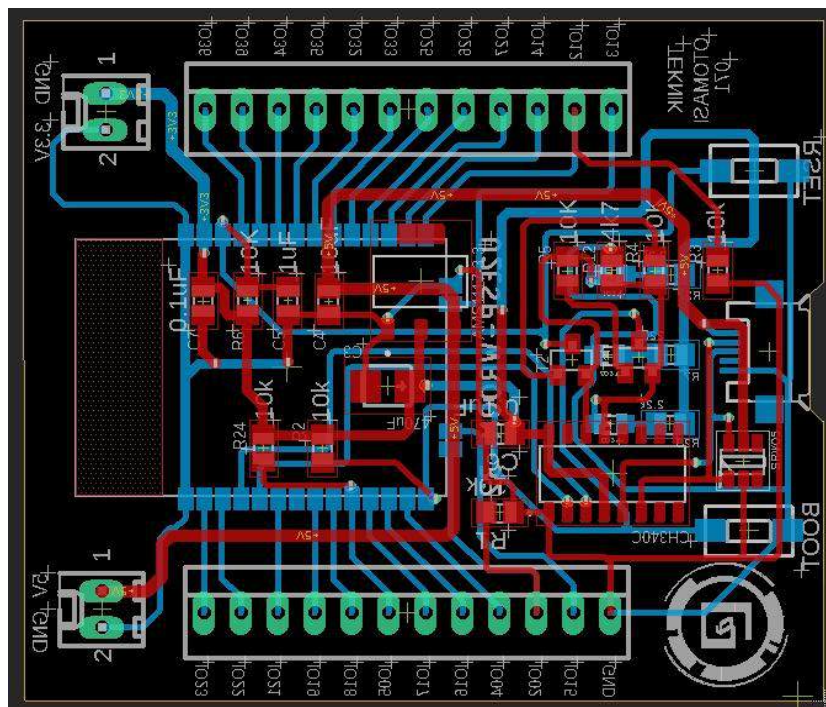
Pada tahap ini dilakukan proses perancangan rangkaian elektronik yang menghubungkan antara sensor sebagai input dan aktuator sebagai output ke chip

mikrokontroler ESP32, yang berfungsi sebagai pusat pengolahan data. Seluruh rangkaian dirancang menggunakan PCB custom sehingga menghasilkan instalasi yang lebih rapi, aman, dan mudah dalam proses pemasangan pada panel kontrol.



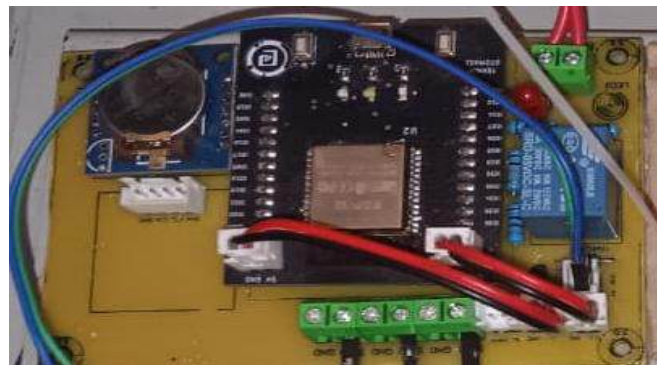
Gambar 4.1. *Routing Board Bottom PCB (Extension ESP32)*
Sumber: Pribadi

Hasil perancangan board PCB ditunjukkan pada Gambar 4.1. Langkah pertama diawali dengan pembuatan skematik rangkaian dan board tata letak jalur PCB yang disesuaikan dengan kebutuhan sistem. Pada tahap ini dilakukan penentuan posisi setiap komponen, seperti ESP32, *Relay* lampu, *Relay* kipas, sensor RTC, LCD I2C, serta konektor catu daya.



Gambar 4.2. *Routing Board Top & Bottom PCB*
Sumber: Pribadi

Ketelitian dalam memperhatikan hubungan antarjalur (*routing*), perancangan PCB juga mempertimbangkan aspek kemudahan instalasi, distribusi daya, keamanan rangkaian, serta efisiensi ruang di dalam panel kontrol. Selain desain *extention* dari esp 32, berikut juga merupakan desain dari desain bottom dan top untuk chip esp 32 dan wiring untuk modul beserta pinnya yang dapat dilihat pada Gambar 4.2. Kemudian dilakukan percetakan dari hasil routing ke dalam papan pcb, dan dilakukan perakitan komponen pada papan pcb yang telah dicetak yang dapat dilihat pada Gambar 4.3.



Gambar 4.3. Hasil PCB yang Telah Dicetak dan Dirakit
Sumber: Pribadi

Setelah proses cetak dan perakitan PCB selesai, seluruh komponen dipasang sehingga membentuk modul kontrol ESP32 seperti yang ditunjukkan pada Gambar 4.3 Modul ini berfungsi sebagai pusat kendali sistem IoT yang menerima data dari sensor, menjalankan perintah yang dikirimkan melalui jaringan MQTT, serta mengendalikan *Relay* untuk aktuator sesuai hasil deteksi YOLO yang diproses pada laptop server. Penggunaan PCB custom menghasilkan instalasi yang lebih ringkas, memudahkan proses pemeliharaan, serta meningkatkan keamanan dan kestabilan koneksi dibandingkan penggunaan rangkaian berbasis breadboard atau kabel jumper.

4.1.2 Hasil Pengujian Sensor *Camera* (TAPO 212)

Pengujian sensor kamera dilakukan untuk memastikan bahwa fungsi utama yaitu pengelihan visual input dapat berfungsi dengan baik dan mampu mengirimkan hasil data visual secara akurat ke sistem. Selain ditampilkan pada gambar 4.4 ang ditampilkan pada terminal VS code, Pengujian ini melibatkan dashboard *monitoring* berbasis visi laptop yang mengintegrasikan model YOLOv8 dengan perangkat Tapo C212 melalui akun cctv dan RSTP yang ada pada perangkat

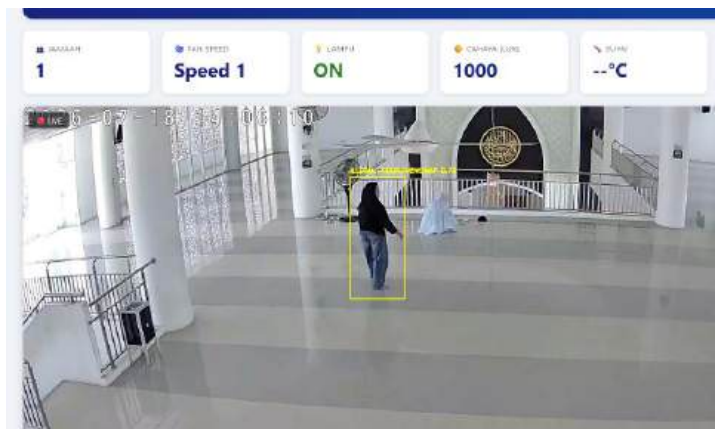
untuk melakukan deteksi objek secara *real-time* di dalam area masjid, khususnya dalam mengidentifikasi keberadaan jamaah wanita.

```
Dashboard: http://localhost:5000/
Video Feed: http://localhost:5000/video_feed
MQTT Broker: bd9483829e9e46668c77b8c217576c72.s1.eu.hivemq.cloud
MQTT Status: ✔ Connected

[INFO] Mencoba URL #1: rtsp://vemasardiyan:vemas1009@192.168.1.101:554/stream1
```

Gambar 4.4. Konektivitas pada Terminal Untuk RSTP/IP pada CCTV
Sumber: Pribadi

Sesuai gambar 4.5, tampilan antarmuka menunjukkan bahwa sistem mampu melakukan klasifikasi dan *counting* secara langsung dari *video stream*, dimana objek manusia yang terdeteksi akan diproses oleh model *deep learning* dan divisualisasikan dalam bentuk *bounding box* serta informasi jumlah pada panel analitik di sisi kiri.



Gambar 4.5. Pengujian Pembacaan Sensor Kamera
Sumber: Pribadi

Data pengujian ini nantinya diolah secara teknis, dengan pendekatan memanfaatkan arsitektur *YOLOv8* yang telah dioptimasi untuk *object detection*, sehingga mampu bekerja cukup stabil pada kondisi pencahayaan *indoor* dan variasi posisi jamaah seperti duduk, berjalan, maupun berkumpul.

Tabel 4.1. Hasil Pengujian Sensor Kamera (Deteksi Seluruh Tubuh)

No.	Tanggal Pengujian	Waktu pengamatan	Jumah Yang Terdeteksi	Jumlah aktual	Akurasi(%)
1	Sabtu, 25 April 2026	08.09 – 09.30	15	15	100
2	Minggu, 26 April 2026	09.16 – 12.22	10	11	97.82

No.	Tanggal Pengujian	Waktu pengamatan	Jumah Yang Terdeteksi	Jumlah aktual	Akurasi(%)
3	Senin, 27 April 2026	07.10 – 09.53	30	34	93.75
4	Selasa, 28 April 2026	12.30 – 14.10	37	38	97.82
5	Rabu, 29 April 2026	09.21 – 11.30	40	45	95.65
6	Kamis, 30 April 2026	07.50 – 10.25	26	28	98.43
7	Sabtu, 2 Mei 2026	11.09 - 18.53	59	62	98.15
8	Minggu, 3 Mei 2026	12.16 - 20.36	17	17	100
Akurasi Rata – Rata					97.70

Berdasarkan data pengujian tabel 4.1 selama satu minggu, sistem menunjukkan performa yang relatif konsisten dengan rata-rata akurasi sebesar 97,70%. Nilai ini diperoleh dari perbandingan antara jumlah deteksi sistem dengan jumlah aktual hasil observasi manual. Pada beberapa kondisi, seperti tanggal 27 April 2026, terjadi penurunan akurasi menjadi 93,75% yang kemungkinan disebabkan oleh *occlusion* (tumpang tindih antar objek) serta sudut pengambilan gambar yang cukup lebar sehingga mengurangi presisi deteksi. Namun demikian, pada kondisi lain seperti 25 April dan 3 Mei 2026, sistem mampu mencapai akurasi 100%, yang menunjukkan bahwa model YOLOv8 cukup andal dalam kondisi lingkungan yang lebih terkontrol. Didapatkan hasil data yang menunjukkan bahwa semakin tinggi kepadatan jamaah, potensi *error* juga meningkat, meskipun masih dalam batas toleransi sistem deteksi berbasis visi laptop. Secara ilmiah, hasil ini sejalan dengan beberapa penelitian terdahulu yang menyatakan bahwa YOLOv8 memiliki performa tinggi dalam tugas *real-time object detection* dengan keseimbangan antara kecepatan dan akurasi.

4.1.3 Pengujian Sensor RTC (*Real Time Clock*) DS3231M

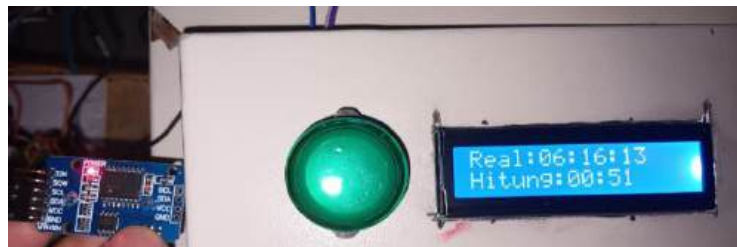
Pengujian sensor RTC DS3231 dilakukan untuk mengevaluasi tingkat ketepatan waktu sistem dalam mendukung proses monitoring dan pengendalian otomatis berbasis waktu nyata (*real-time*). Pengujian ini membandingkan waktu yang dihasilkan oleh modul RTC dengan waktu referensi yang diperoleh dari

perangkat eksternal yang telah tersinkronisasi dengan *Network Time Protocol* (NTP). Dan pada tabel 4.2 dibawah ini adalah konfigurasi pin antara esp32 dan ke komponen sensor RTC.

Tabel 4.2. Koneksivitas Sensor RTC

RTC	ESP 32
VCC	5V
SCL	PIN 21
SDA	PIN 22
GND	GND

Berdasarkan hasil pengujian yang dilakukan pada tabel 4.3 selama interval waktu tertentu, diperoleh selisih waktu antara RTC dan waktu pengujian 1 menit. Untuk mekanisme pengujiannya Adalah dengan sesor RTC dan dengan batuan LCD i2c dengan didapatkan hasil pengujian ketepatan waktu antara waktu real dengan modul RTC, diperoleh selisih waktu yang sangat kecil pada setiap percobaan.



Gambar 4.6. Pengujian Sensor RTC Terhadap Ketepatan Waktu
Sumber: Pribadi

Dari 7 kali pengujian, selisih waktu terbesar hanya mencapai 2 detik, sedangkan beberapa pengujian menunjukkan selisih 0 detik yang menandakan sinkronisasi waktu berjalan sangat baik. Nilai rata-rata error yang diperoleh sebesar 0,85 detik menunjukkan bahwa modul RTC memiliki tingkat akurasi yang tinggi dan masih berada dalam batas toleransi yang sangat baik untuk sistem otomatisasi *monitoring* dan kontrol perangkat elektronik.

Tabel 4.3. Data Hasil Pengujian Sensor RTC Terhadap Ketepatan Waktu

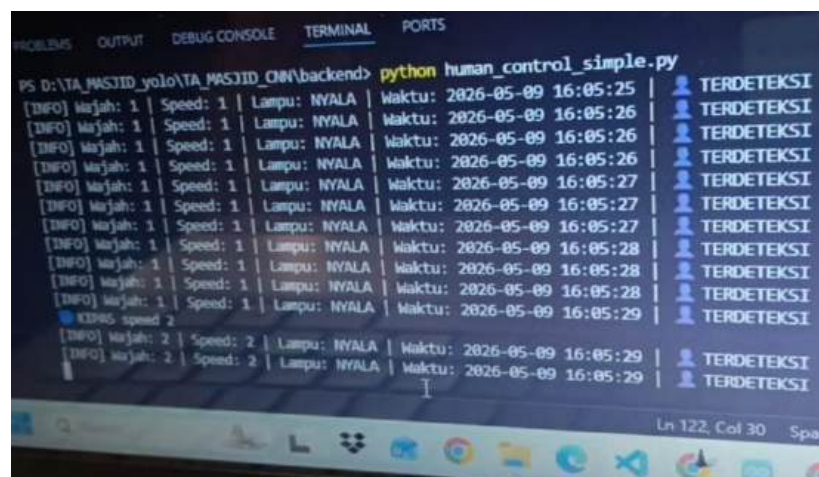
Pengujian	Waktu Real	Waktu RTC	Selisih waktu
1	06:12:35	06:12:36	1 detik
2	06:14:37	06:14:37	0 detik
3	06:16:13	06:16:14	1 detik
4	06:18:14	06:18:15	1 detik

Pengujian	Waktu Real	Waktu RTC	Selisih waktu
5	06:20:16	06:20:18	2 detik
6	06:22:18	06:22:18	0 detik
7	06:24:18	06:24:19	1 detik
Rata-rata eror			0.85 detik

Hasil pengujian juga menunjukkan bahwa modul RTC mampu mempertahankan kestabilan waktu secara konsisten selama sistem berjalan. Perbedaan waktu yang muncul kemungkinan dipengaruhi oleh delay proses komunikasi serial, waktu eksekusi program mikrokontroller, serta keterlambatan pembacaan data pada sistem monitoring. Namun demikian, selisih tersebut tidak memberikan pengaruh signifikan terhadap performa sistem secara keseluruhan karena nilai *error* masih berada di bawah 1 detik. Meskipun terdapat selisih waktu kecil yang cenderung meningkat seiring durasi pengamatan, hal ini tidak memberikan dampak signifikan terhadap kinerja sistem secara keseluruhan. Faktor-faktor seperti delay komunikasi serial, proses pembacaan sensor, serta latensi pemrosesan data turut berkontribusi terhadap deviasi yang terjadi.

4.1.4 Pengujian *Relay 4 Channel Optocouler* untuk 3 *Speed Kipas*

Maksud dari pengujian *Relay* ini adalah untuk mengetahui keandalan utama dari komponen ini. Sesuai dengan gambar 4.7 dan 4.8 dibawah ini, sistem yang dibuat sudah berjalan secara real time menggunakan *YOLOv8* dengan *Speed (Relay)* dan jumlah orang disertai menu *serial monitor* yang mendukung data keaktifan *Relay*.



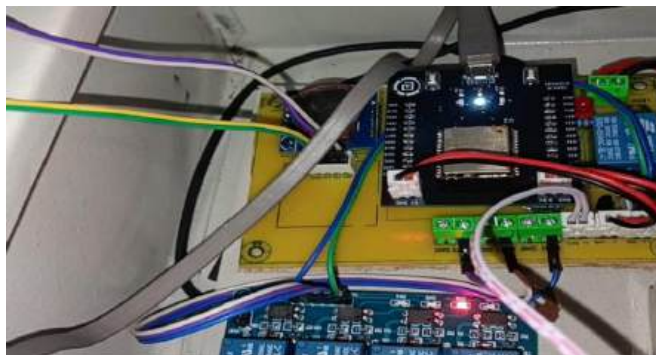
Gambar 4.7. Tampilan Terminal VS Code Terhadap Hasil Pembacaan *Speed Kipas* pada *Relay*
Sumber: Pribadi

Hasil tersebut menunjukkan hasil implementasi sistem kontrol kipas otomatis berbasis pendeteksian jumlah jamaah menggunakan kamera dan mikrokontroler ESP32.

Tabel 4.4. Koneksivitas *Relay 4 Channel*

<i>Relay 4 Channel</i>	ESP 32
DC+	5V
IN1	PIN 25
IN2	PIN 26
IN3	PIN 27
DC-	GND
K1	
COM1	Kabel Netral (Hitam)
NO1	<i>Speed 1</i> (Biru)
K2	
COM2	Kabel Netral (Hitam)
NO2	<i>Speed 2</i> (Merah)
K3	
COM3	Kabel Netral (Hitam)
NO3	<i>Speed 3</i> (Kuning)

Pengujian dilakukan untuk mengetahui apakah sistem dapat mengaktifkan *Relay* sesuai dengan jumlah jamaah yang terdeteksi secara otomatis. Setiap channel *Relay* digunakan untuk mengatur tingkat kecepatan kipas yang berbeda, sehingga semakin banyak jumlah jamaah yang terdeteksi maka semakin tinggi kecepatan kipas yang diaktifkan.



Gambar 4.8. Pengujian 3 *Channel* Untuk Masing-Masing *Speed* kipas
Sumber: Pribadi

Sesuai dari data pengujian awal yang saya lakukan, terlihat bahwa sistem memiliki performa yang stabil. Pada kondisi tanpa jamaah, seluruh *Relay* dalam keadaan mati dan sistem tidak menghasilkan output kecepatan, yang menandakan tidak adanya jamaah. Dan sebaliknya ketika mendeteksi keberadaan jamaah, maka salah 1 channel *Relay* akan diaktifkan sesuai jumlah jamaah. Untuk mendapatkan gambaran performa yang lebih komprehensif, dilakukan simulasi data pemantauan selama 1 minggu dengan skenario penggunaan riil di lingkungan masjid. Hasil pengamatan ditujukan seperti pada tabel 4.5 dibawah ini.

Tabel 4.5. Pengujian konektivitas *Relay* Terhadap *Speed* Kipas

Pengujian	Jumlah Jamaah	<i>Relay</i> aktif	<i>Speed</i>	Keterangan
1	0	Semua <i>Off</i>	0/ <i>Off</i>	Berhasil
2	3	Channel 1	1	Berhasil
3	5	Channel 2	2	Berhasil
4	7	Channel 2	3	Berhasil
5	9	Channel 3	3	Berhasil
Rata-rata percobaan Channel <i>Relay</i>				Berhasil

Pada pengujian pertama, sistem mendeteksi tidak ada jamaah (0 orang), sehingga seluruh Channel *Relay* berada pada kondisi *off* dan kipas tidak menyala. Hal ini menunjukkan bahwa sistem mampu bekerja secara efisien karena tidak memberikan daya listrik ketika ruangan kosong. Pada kondisi ini kecepatan kipas berada pada level 0 atau mati dan sistem dinyatakan berhasil. Pada pengujian kedua, ketika terdeteksi 3 orang jamaah, sistem secara otomatis mengaktifkan Channel 1 sehingga kipas berjalan pada *Speed* 1. Kondisi ini menunjukkan bahwa sistem mampu memberikan respon awal terhadap keberadaan jamaah dengan mengaktifkan ventilasi pada tingkat rendah untuk menghemat konsumsi daya listrik.

Selanjutnya pada pengujian ketiga, jumlah jamaah meningkat menjadi 5 orang dan sistem berhasil mengaktifkan Channel 2 yang mengatur kipas pada *Speed* 2. Hal ini membuktikan bahwa logika kontrol pada mikrokontroler berjalan sesuai program yang telah dirancang, yaitu peningkatan jumlah jamaah akan meningkatkan kecepatan kipas secara bertahap. Pada pengujian keempat dan kelima, jumlah jamaah yang terdeteksi masing-masing sebanyak 7 orang dan 9

orang. Pada kedua kondisi tersebut sistem mengaktifkan Chnanel 3 sehingga kipas bekerja pada *Speed* tertinggi.

4.1.5 Pengujian *Relay* 1 Channel 5V (Include in Extension Microcontroller)

Pengujian *Relay* 1 channel 5V dilakukan untuk memastikan kemampuan sistem dalam mengendalikan lampu secara otomatis berdasarkan logika deteksi yang diperoleh dari sistem *deep learning* dan mikrokontroler ESP32.

Tabel 4.6. Pengujian koneksivitas *Relay* terhadap lampu

<i>Relay</i> 4 Channel	ESP 32
DC+	5V
IN	PIN 33
DC-	GND
COM	Kabel Netral
NO	Kabel Fasa

Pada tabel 4.6 diatas koneksivitas *Relay* terhadap esp32, adalah berfungsi sebagai saklar elektronik yang menghubungkan dan memutus tegangan AC menuju lampu sesuai kondisi sistem. Ketika sistem memberikan logika HIGH, *Relay* akan aktif dan mengalirkan tegangan listrik menuju lampu, sedangkan logika LOW menyebabkan *Relay* tidak aktif sehingga aliran listrik terputus.



Gambar 4.9. Pengukuran Tegangan dari *Relay* 5v untuk lampu
Sumber: Pribadi

Pada pengujian dilakukan pada gambar 4.9 dan hasilnya di tabel 4.7, didapat interval perubahan kondisi setiap 1 menit untuk mengetahui kestabilan respon *Relay* terhadap perubahan logika sistem dan tegangan keluaran yang dihasilkan.

Pada pengujian pertama, sistem berada pada kondisi *system off* sehingga *Relay* dalam keadaan tidak aktif. Tegangan output yang terukur sebesar 0,9 VAC menunjukkan tidak terdapat aliran tegangan utama menuju lampu. Nilai tegangan kecil tersebut masih dapat muncul akibat noise listrik atau induksi pada alat ukur, namun tidak cukup untuk menyalakan beban.

Tabel 4.7. Data Pengujian *Relay* Lampu Terhadap Jumlah Jamaah

Pengujian	Timer per-1 menit	Logika sistem	Kondisi <i>Relay</i>	Tegangan Output (V)	Kondisi Lampu
1	<i>System Off</i>	<i>Off</i>	Tidak Aktif	0.9 VAC	Mati
2	06.49	<i>On</i>	Aktif	221.4 VAC	Nyala
3	06.50	<i>Off</i>	Tidak Aktif	0.15 VAC	Mati
4	06.51	<i>On</i>	Aktif	220.9 VAC	Nyala
5	06.52	<i>Off</i>	Tidak Aktif	0.11 VAC	Mati

Pada pengujian kedua, timer menunjukkan waktu 06.49 dengan logika sistem dalam kondisi *on*. *Relay* berhasil aktif dan menghasilkan tegangan output sebesar 221,4 VAC. Nilai tersebut menunjukkan *Relay* berhasil menghubungkan sumber listrik PLN menuju lampu sehingga lampu menyala secara normal. Selanjutnya pada pengujian ketiga saat timer berubah menjadi 06.50, logika sistem kembali *off* dan *Relay* tidak aktif. Tegangan output naik menjadi 0,11 VAC yang menandakan aliran listrik menuju lampu telah terputus.

Pengujian keempat dilakukan pada waktu 06.51 ketika sistem kembali aktif. *Relay* kembali berhasil bekerja dan menghasilkan tegangan output sebesar 220,9 VAC sehingga lampu dapat menyala kembali sesuai logika sistem. Pada pengujian kelima, timer menunjukkan waktu 06.52 dengan kondisi sistem *off*. *Relay* kembali tidak aktif dan tegangan output turun menjadi 0,11 VAC. Berdasarkan hasil pengujian tersebut, *Relay* mampu bekerja secara stabil dalam mengendalikan lampu otomatis sesuai perubahan logika sistem yang telah dirancang pada implementasi *Electrical Facility Control* di Masjid Bahrul Ulum.

4.1.6 Pengujian *LCD i2c* Terhadap Beberapa Input dan Output

Pengujian LCD I2C dilakukan untuk memastikan bahwa sistem mampu menampilkan informasi kondisi secara *real-time* berdasarkan hasil pemrosesan data jumlah jamaah yang diperoleh dari sistem deteksi berbasis visi laptop. Parameter yang diuji meliputi kesesuaian antara jumlah jamaah yang terdeteksi, status kipas

(*on/off*), tingkat kecepatan kipas (*Speed* 1–3), serta status lampu terhadap informasi yang ditampilkan pada LCD, dengan konfigurasi pin ke esp32 seperti pada tabel 4.8 dibawah ini.



Gambar 4.10. Uji Coba Pembacaan LCD terhadap Input dan Output
Sumber: Pribadi

Tabel 4.8. Koneksivitas LCD I2C

LCD I2C	ESP 32
VCC	5V
SCL	PIN 21
SDA	PIN 22
GND	GND

Berdasarkan hasil pengujian tabel 4.9 dan gambar 4.10, seluruh fungsi sistem bekerja sesuai dengan logika yang telah dirancang. Pada kondisi pertama saat jumlah jamaah bernilai 0, sistem mendeteksi bahwa ruangan dalam keadaan kosong sehingga kipas dan lampu berada pada kondisi nonaktif (*Off*). LCD menampilkan informasi “Jamaah:0 Kipas:Off Lampu:Off” yang menandakan tidak terdapat aktivitas di dalam ruangan.

Tabel 4.9. Hasil Pengujian dan Koneksivitas LCD dengan Output

No.	Jumlah Jamaah	Status kipas	<i>Speed</i> Kipas	Status Lampu	Tampilan LCD	Kesesuaian
1	0	Off	0	Off	Jamaah:0 Kipas:Off Lampu:Off	Sesuai
2	1	On	1	On	Jamaah:1 Kipas: On Lampu: On	Sesuai
3	2	On	1	On	Jamaah:2 Kipas: On Lampu: On	Sesuai
4	3	On	1	On	Jamaah:3 Kipas: On Lampu: On	Sesuai

No.	Jumlah Jamaah	Status kipas	<i>Speed</i> Kipas	Status Lampu	Tampilan LCD	Kesesuaian
5	5	On	2	On	Jamaah:5 Kipas: On Lampu: On	Sesuai
6	6	On	2	On	Jamaah:6 Kipas: On Lampu: On	Sesuai
7	8	On	3	On	Jamaah:8 Kipas:On Lampu: On	Sesuai

Pada pengujian kedua sampai ke empat, saat jumlah jamaah sebanyak 1 hingga 3 orang, sistem mulai mengaktifkan kipas pada *Speed* 1 kipas dan menyalakan lampu. Selanjutnya pada pengujian kelima dan keenam, ketika jumlah jamaah meningkat menjadi 5 hingga 6 orang, sistem secara otomatis meningkatkan *Speed* 2 kipas. Perubahan kecepatan ini dilakukan untuk menyesuaikan kebutuhan sirkulasi udara berdasarkan tingkat kepadatan ruangan.

Pada pengujian ketujuh, jumlah jamaah meningkat menjadi 8 orang. Sistem kemudian meningkatkan *Speed* 3 kipas sebagai kecepatan tertinggi untuk menjaga kenyamanan suhu dan sirkulasi udara di dalam ruangan. Lampu tetap menyala dan LCD terus memperbarui informasi jumlah jamaah serta status perangkat secara otomatis.

4.1.7 Pengujian *LCD i2c* Terhadap Beberapa Input dan Output

Pengujian sensor LDR dilakukan untuk mengetahui kemampuan sensor dalam mendeteksi perubahan intensitas cahaya pada area Masjid Bahrul Ulum dengan konektivitas pin ke esp 32 ada pada tabel 4.10 dibawah ini. Pengukuran dilakukan dengan memberikan variasi pencahayaan mulai dari kondisi sangat gelap hingga kondisi sangat terang dengan rentang pengukuran sebesar 45–1000 lux.

Tabel 4.10. Konektivitas Pin Sensor LDR

LCD I2C	ESP 32
VCC	5V
SCL	PIN 21
SDA	PIN 22

Nilai intensitas cahaya yang terbaca kemudian dikirimkan ke ESP32 sebagai parameter pendukung dalam pengambilan keputusan sistem kontrol otomatis. Berdasarkan hasil pengujian, sensor mampu mendeteksi perubahan intensitas cahaya secara bertahap dengan baik. Pada kondisi pencahayaan rendah (45–285 lux), sistem mengidentifikasi lingkungan sebagai kondisi gelap sehingga lampu diaktifkan apabila kamera mendeteksi keberadaan manusia.



Gambar 4.11. Hasil Pengujian Sensor LDR
Sumber: Pribadi

Sebaliknya, ketika intensitas cahaya mencapai 470–1000 lux, sistem mengidentifikasi lingkungan sebagai kondisi terang sehingga lampu tidak perlu dinyalakan karena pencahayaan alami telah mencukupi.

Tabel 4.11. Hasil pengujian Sensor LDR di Masjid Bahrul Ulum PPNS

No.	Intensitas Cahaya (Lux)	Kondisi Ruang	Logika Sistem	Kondisi Lampu	Status LDR Panel	Status Deteksi Jamaah
1	45	Sangat Gelap	On	Menyala	L45	Terdeteksi
2	120	gelap	On	Menyala	L120	Terdeteksi
3	285	Agak Redup	On	Menyala	L285	Terdeteksi
4	470	Cahaya Sedang	Off	Mati	L470	Terdeteksi
5	650	Terang	Off	Mati	L650	Terdeteksi
6	820	Sangat Terang	Off	Mati	L820	Terdeteksi
7	1000	Super Terang	Off	Mati	L1000	Terdeteksi

Pada hasil implementasi pada tabel 4.11, sistem secara keseluruhan, sensor LDR tidak bekerja secara mandiri, melainkan dikombinasikan dengan metode YOLOv8 *Human Detection* dan RTC. Keputusan menyalakan atau mematikan

lampu ditentukan berdasarkan hasil penggabungan ketiga parameter tersebut, yaitu tingkat pencahayaan, keberadaan manusia, dan waktu tunda sistem. Terkhususnya oleh deteksi CCTV terhadap kehadiran jamaah sebagai keputusan utama, dengan mekanisme tersebut, sistem mampu mengoptimalkan penggunaan energi listrik tanpa mengurangi kenyamanan pengguna masjid.

4.1.8 Pengujian Jangkauan Wireless Web dan Sistem

Pengujian web monitoring dilakukan untuk mengetahui performa komunikasi data antara mini PC, ESP32, web *monitoring*, dan database melalui jaringan internet menggunakan infrastruktur Cloudflare Tunnel dan protokol MQTT HiveMQ. Parameter yang diamati meliputi waktu respons (delay) sistem pada berbagai jarak pengujian terhadap access point (WiFi) Masjid Bahrul Ulum.



Gambar 4.12. Pengukuran Jarak Efektif untuk Sistem
Sumber: Pribadi

Berdasarkan hasil pengujian pada Tabel 4.12, diperoleh rata-rata waktu respons sistem berkisar antara 1,6 detik hingga 7 detik. Pada jarak 1–2 meter, sistem menunjukkan performa yang baik dengan rata-rata delay masing-masing sebesar 2 detik dan 1,6 detik, sehingga komunikasi data berlangsung secara stabil. Kondisi ini menunjukkan bahwa ketika kualitas sinyal WiFi masih kuat, proses pengiriman data dari ESP32 ke web monitoring maupun sebaliknya dapat berlangsung dengan cepat dan responsif. Namun demikian, pada beberapa titik pengujian diperoleh nilai delay yang relatif tinggi meskipun jaraknya tidak terlalu jauh dari access point. Sebagai contoh, pada jarak 3 meter diperoleh rata-rata delay mencapai 7 detik, sedangkan pada jarak 6 meter sebesar 5,6 detik, jarak 10 meter sebesar 4,6 detik,

dan jarak 11 meter sebesar 4 detik. Kondisi tersebut menyebabkan sistem hanya berada pada kategori cukup stabil hingga kurang stabil.

Tabel 4.12. Data Hasil Pengujian Jarak Stabil untuk Sistem *Wireless*

Jarak Real	Pengujian 1	Pengujian 2	Pengujian 3	Rata-Rata Delay Respon	Kondisi Sistem
1M	2 s	3 s	1 s	2 s	Stabil
2M	1 s	1 s	3 s	1,6 s	Stabil
3M	5 s	10 s	6 s	7 s	Kurang Stabil
4M	3 s	3 s	2 s	2,6 s	Stabil
5M	2 s	3 s	4 s	3 s	Stabil
6M	7 s	4 s	6 s	5,6 s	Cukup Stabil
7M	1 s	3 s	4 s	2,6 s	Stabil
8M	2 s	4 s	1 s	2.3 s	Stabil
9M	3 s	3 s	3 s	3 s	Stabil
10M	4 s	6 s	4 s	4,6 s	Cukup Stabil
11M	8 s	3 s	1 s	4 s	Cukup stabil
12M	2	2 s	5 s	3'	Stabil

Kondisi jaringan saat pengujian:

- Menggunakan wifi utama ppns hotspot,
- Terdapat beberapa perangkat lain yang aktif terutama hari kerja,
- Penempatan router yang tidak lebih 6 meter ,
- Pengujian dilakukan sebanyak 3 kali pada tiap jarak

Hasil tersebut menunjukkan bahwa jarak bukan merupakan faktor utama yang memengaruhi performa sistem, melainkan kualitas jaringan internet WiFi yang tersedia di Masjid Bahrul Ulum. Berdasarkan pengamatan selama proses pengujian, jaringan WiFi masjid memiliki kualitas sinyal yang kurang stabil sehingga menyebabkan fluktuasi waktu respons pada sistem. Bahkan pada kondisi masjid yang relatif sepi tanpa banyak pengguna jaringan, koneksi internet masih mengalami penurunan performa yang berdampak pada meningkatnya delay komunikasi data.

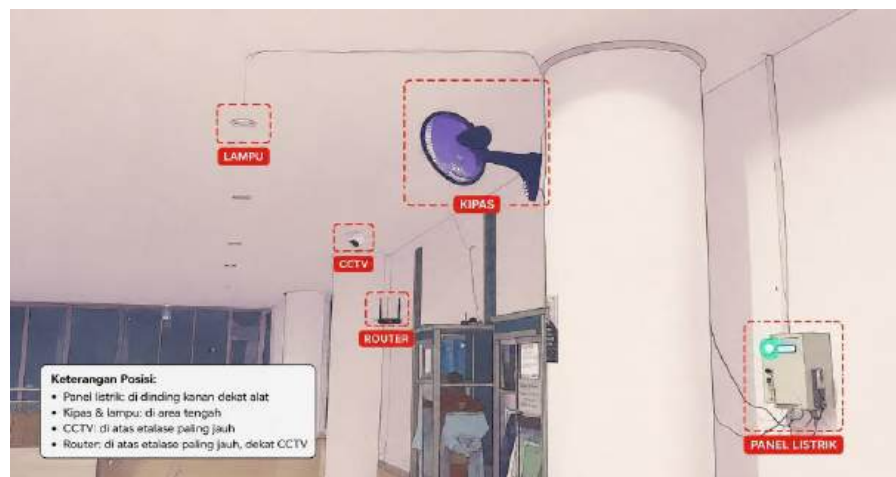
Sesuai pada tabel 4.12, ketidakstabilan jaringan tersebut memengaruhi seluruh proses komunikasi pada sistem, mulai dari pengiriman hasil deteksi YOLO dari mini PC menuju database Supabase, komunikasi MQTT melalui HiveMQ, hingga proses pembaruan (update) tampilan web monitoring secara real-time. Akibatnya, status aktuator seperti lampu dan kipas pada dashboard terkadang

mengalami keterlambatan pembaruan beberapa detik dibandingkan kondisi sebenarnya di lapangan. Meskipun demikian, seluruh data yang dikirimkan tetap berhasil diterima oleh sistem tanpa mengalami kehilangan paket (packet loss) maupun kegagalan komunikasi. Hal ini menunjukkan bahwa implementasi Cloudflare Tunnel, HiveMQ MQTT, dan Supabase telah bekerja dengan baik dalam menjaga koneksi sistem, sementara keterlambatan yang terjadi lebih disebabkan oleh kualitas jaringan internet lokal dibandingkan oleh arsitektur perangkat lunak yang digunakan.

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem monitoring berbasis web telah mampu beroperasi dengan baik untuk kebutuhan monitoring dan pengendalian perangkat IoT secara real-time. Akan tetapi, untuk memperoleh performa yang lebih optimal pada implementasi di lapangan, diperlukan jaringan internet dengan kualitas yang lebih stabil, bandwidth yang memadai, serta kekuatan sinyal WiFi yang merata di seluruh area Masjid Bahrul Ulum sehingga waktu respons sistem dapat dipertahankan pada kisaran 1–3 detik secara konsisten.

4.2 Hasil Perakitan *Hardware* Sistem

Perancangan mekanik alat merupakan suatu proses kompleks yang melibatkan berbagai langkah dan pertimbangan untuk menciptakan alat atau sistem mekanik yang efektif, efisien, dan dapat diandalkan. Proses ini memerlukan pemahaman teknis dan perhitungan secara matang.



Gambar 4.13. Desain Awal Perancangan Sistem
Sumber: Pribadi

Gambar 4.13 tersebut merupakan desain awal (pre-implementation design) yang dibuat sebelum proses implementasi perangkat keras dilakukan di Masjid

Bahrul Ulum PPNS. Desain ini berfungsi sebagai acuan dalam menentukan tata letak perangkat, jalur instalasi kabel, serta hubungan antar komponen yang akan digunakan pada sistem kontrol cerdas berbasis IoT. Melalui rancangan ini, dapat memastikan bahwa setiap perangkat, mulai dari panel kontrol, ESP32, sensor, *Relay*, lampu, kipas, router, hingga kamera CCTV, telah memiliki posisi pemasangan dan jalur koneksi yang sesuai dengan kondisi lapangan sebelum proses instalasi dilakukan.



Gambar 4.14. Pemasangan CCTV Router, dan seluruh sistem lainnya
Sumber: Pribadi

Lalu pada gambar 4.14 diatas adalah tahap perakitan perangkat keras merupakan proses implementasi hasil perancangan sistem yang telah disusun pada tahap sebelumnya. Seluruh komponen dipasang secara langsung pada area implementasi penelitian, yaitu lantai dua Masjid Bahrul Ulum PPNS, tepatnya pada area shaf paling belakang jamaah putri. Lokasi tersebut dipilih karena menjadi titik yang paling strategis untuk proses instalasi panel kontrol, distribusi kabel, serta memudahkan proses pemeliharaan tanpa mengganggu aktivitas ibadah. Selain itu, posisi ini juga berada cukup dekat dengan sumber listrik utama yang digunakan sebagai catu daya sistem sehingga proses instalasi dapat dilakukan dengan lebih aman dan rapi.

Panel listrik berfungsi sebagai pusat kendali seluruh perangkat elektronik yang digunakan dalam penelitian ini. Di dalam panel terdapat beberapa komponen utama, yaitu MCB 4A sebagai pengaman terhadap arus lebih, fuse sebagai proteksi tambahan, terminal block sebagai pusat distribusi kabel, power supply 12 VDC 10 A untuk mengubah tegangan AC menjadi DC, modul DC-DC step-down 5 V sebagai catu daya ESP32, *Relay* optocoupler sebagai sakelar elektronik lampu dan kipas, LCD I2C sebagai media informasi kondisi sistem, pilot lamp sebagai

indikator panel, serta mikrokontroler ESP32 yang bertugas sebagai pengendali utama seluruh proses otomasi. Sesuai pada gambar 4.15, seluruh komponen dipasang secara sistematis di dalam panel agar memudahkan proses pengawatan, inspeksi, maupun perawatan apabila terjadi gangguan selama sistem beroperasi.



Gambar 4.15. Implementasi *Hardware* dan pengujian *Relay* lampu pada malam hari
Sumber: Pribadi

Di sekitar lokasi panel juga ditempatkan router nirkabel yang terhubung dengan jaringan internet Masjid Bahrul Ulum. Router ini berfungsi sebagai media komunikasi antara ESP32, laptop, website monitoring, broker MQTT HiveMQ Cloud, dan database Supabase sehingga seluruh perangkat dapat saling bertukar data melalui jaringan internet secara real-time. Penempatan router dibuat sedekat mungkin dengan panel kontrol untuk meminimalkan panjang kabel jaringan yang digunakan sekaligus menjaga kestabilan koneksi selama proses monitoring dan pengendalian berlangsung. Penggunaan jalur instalasi di sepanjang plafon juga bertujuan untuk melindungi kabel dari gangguan mekanis maupun sentuhan langsung pengguna masjid selama sistem beroperasi.

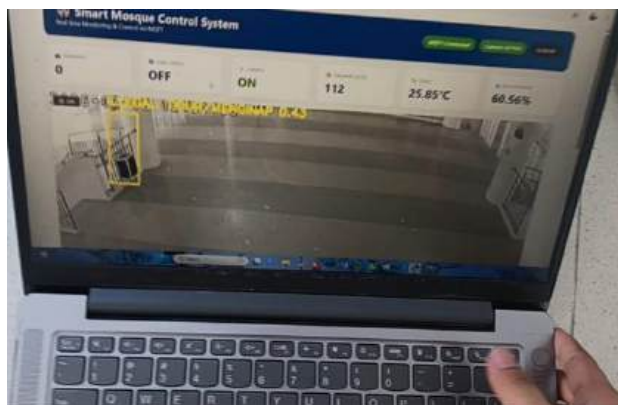
Perangkat aktuator berupa kipas dinding dan lampu penerangan dipasang pada area shaf tengah dengan jarak sekitar empat meter dari panel listrik. Posisi tersebut dipilih berdasarkan hasil observasi lapangan agar distribusi aliran udara dan pencahayaan dapat menjangkau area utama tempat jamaah melaksanakan ibadah. Kipas dinding dikendalikan melalui *Relay* yang dihubungkan dengan ESP32 sehingga dapat menyala maupun mati secara otomatis sesuai hasil deteksi jumlah jamaah oleh sistem YOLOv8. Sementara itu, lampu penerangan juga

dikendalikan menggunakan *Relay* terpisah dengan mempertimbangkan kondisi intensitas cahaya yang dibaca sensor LDR serta waktu operasional yang diperoleh dari modul RTC.

4.3 Hasil Perancangan dan Konektivitas Software

Perancangan perangkat lunak ini bertujuan untuk mengintegrasikan seluruh perangkat keras dengan sistem *monitoring* berbasis web melalui jaringan internet menggunakan protokol MQTT dan layanan Cloudflare Tunnel. Sistem ini dirancang agar proses deteksi objek menggunakan metode YOLOv8, pengendalian aktuator, monitoring sensor, serta penyimpanan data dapat berlangsung secara otomatis, terintegrasi, dan tersinkronisasi secara *real-time*. Perangkat lunak utama dikembangkan menggunakan bahasa pemrograman Python pada Visual Studio Code sebagai backend yang berfungsi untuk mengolah hasil deteksi kamera CCTV, membaca data sensor, mengendalikan ESP32, serta mengirimkan data menuju database Supabase dan web monitoring.

Komunikasi antara mini PC, ESP32, dan web monitoring menggunakan protokol MQTT melalui broker HiveMQ sehingga proses pertukaran data dapat berlangsung dengan latensi yang rendah dan mendukung komunikasi dua arah (*publish-subscribe*). Selanjutnya, Cloudflare Tunnel dimanfaatkan untuk menghubungkan server lokal yang berjalan pada mini PC ke jaringan internet secara aman tanpa memerlukan konfigurasi port forwarding pada jaringan lokal masjid. Dengan mekanisme tersebut, halaman web monitoring dapat diakses menggunakan domain sistemmbu.my.id dari berbagai perangkat yang terhubung ke internet, baik untuk memantau kondisi sistem maupun melakukan pengendalian aktuator secara jarak jauh.



Gambar 4.16. Konektivitas Software, HTTP, dan Supabase dengan Kamera
Sumber: Pribadi

Pada gambar 4.16 tampilan antarmuka sistem dikembangkan dalam bentuk website menggunakan Visual Studio Code dengan memanfaatkan *framework* berbasis Python dan terhubung secara langsung dengan database Supabase sebagai media penyimpanan data. Informasi yang ditampilkan pada web meliputi hasil deteksi manusia menggunakan YOLOv8, status lampu dan kipas, nilai intensitas cahaya dari sensor LDR, suhu lingkungan, waktu dari sensor RTC, serta status koneksi perangkat IoT.

4.3.1 Tampilan Data dan Status

Pada gambar 4.17 di bagian paling atas ditampilkan identitas sistem berupa *Smart Mosque Control System* beserta informasi konektivitas perangkat. Indikator MQTT *Connected* menunjukkan bahwa komunikasi antara ESP32, mini PC, dan broker HiveMQ berhasil dilakukan sehingga seluruh proses publish maupun subscribe data dapat berjalan secara normal. Selain itu terdapat indikator Camera Status yang memberikan informasi apakah kamera CCTV sedang aktif melakukan streaming ataupun tidak. Informasi alamat IP lokal mini PC juga ditampilkan sebagai identitas perangkat yang sedang menjalankan server.



Gambar 4.17. Tampilan Bagian Data Sstem
Sumber: Pribadi

Di bawah bagian header terdapat beberapa widget monitoring yang menampilkan parameter penting sistem, yaitu jumlah jamaah hasil deteksi YOLOv8, kecepatan kipas, status lampu, intensitas cahaya berdasarkan sensor LDR, suhu, dan kelembaban ruangan. Proses pembaruan data berlangsung melalui mekanisme berikut.

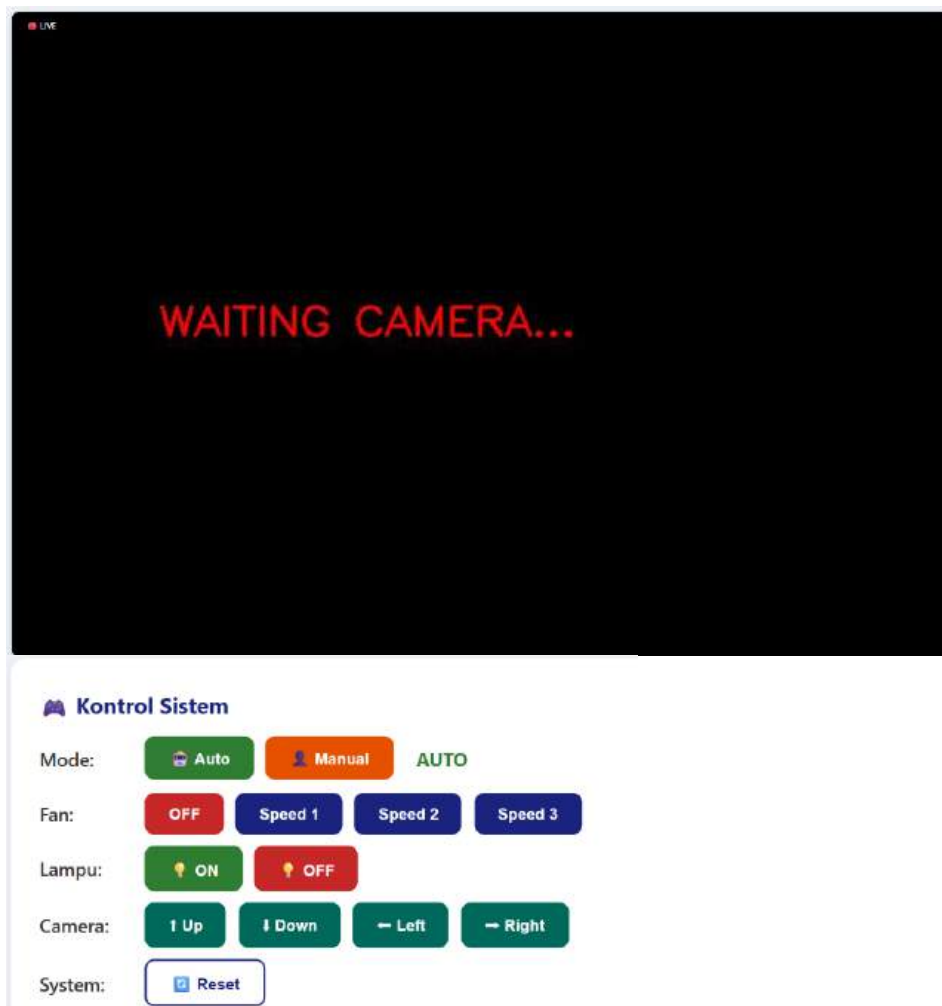
1. CCTV mengirimkan video secara kontinu menuju mini PC.
2. Mini PC melakukan proses inferensi menggunakan model YOLOv8.
3. Hasil deteksi berupa jumlah jamaah, keberadaan sampah, maupun aktivitas tidur diterjemahkan menjadi logika kontrol.

4. Nilai sensor LDR dan RTC dikirim oleh ESP32 melalui MQTT menuju HiveMQ Cloud.
5. Website melakukan subscribe terhadap topic MQTT sehingga setiap perubahan data langsung ditampilkan pada dashboard.

Data tersebut diperbarui secara otomatis tanpa perlu melakukan refresh halaman sehingga pengguna selalu memperoleh kondisi sistem terkini.

4.3.2 Tampilan Kontrol Sistem

Panel Kontrol Sistem merupakan halaman utama pada website *monitoring* yang berfungsi sebagai antarmuka (*interface*) bagi pengguna untuk memantau kondisi perangkat serta mengendalikan aktuator yang terpasang



Gambar 4.18. Tampilan Kontrol Sistem
Sumber: Pribadi

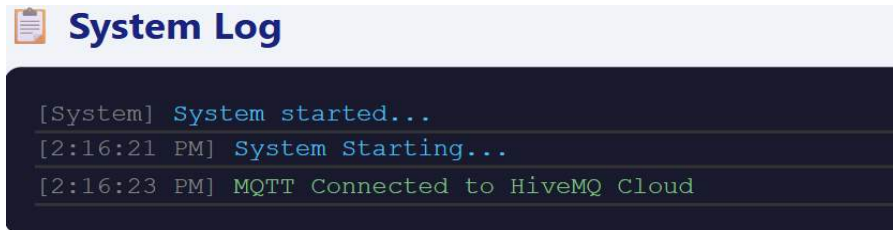
Pada gambar 4.18, tampilan tersebut digunakan sebagai area live monitoring kamera CCTV, yaitu menampilkan hasil streaming video dari kamera yang digunakan sebagai media pengambilan citra. Tampilan ini memungkinkan pengguna mengamati kondisi ruangan secara langsung untuk memastikan bahwa sistem bekerja sesuai keadaan sebenarnya. Apabila kamera belum terhubung atau proses streaming belum berjalan, maka sistem akan menampilkan informasi "Waiting Camera" sebagai indikator bahwa koneksi kamera masih belum tersedia. Lalu pada mode manual, operator dapat mengendalikan seluruh aktuator melalui website tanpa bergantung pada keputusan sistem otomatis. Beberapa fungsi kontrol yang tersedia meliputi:

- Menghidupkan atau mematikan lampu.
- Mengatur kecepatan kipas pada *Speed 1*, *Speed 2*, dan *Speed 3*.
- Mengendalikan arah kamera CCTV PTZ.
- Melakukan reset sistem apabila terjadi gangguan komunikasi.

Ketika salah satu tombol kontrol ditekan, website akan mengirimkan perintah ke broker HiveMQ menggunakan mekanisme publish MQTT. Selanjutnya ESP32 yang telah melakukan subscribe pada topic yang sama akan menerima data tersebut, memproses isi pesan, kemudian mengaktifkan *Relay* sesuai perintah yang diterima. Setelah *Relay* berubah kondisi, ESP32 kembali mengirimkan status perangkat menuju MQTT sehingga website memperoleh umpan balik (feedback) bahwa aktuator telah berhasil dijalankan.

4.3.3 Tampilan System Log

Setiap kejadian penting yang berlangsung pada sistem akan bisa dilihat secara real time secara otomatis. Informasi tersebut meliputi proses inisialisasi server, keberhasilan koneksi MQTT menuju HiveMQ Cloud, status kamera CCTV, perubahan mode operasi, perubahan kondisi *Relay*, hingga berbagai aktivitas lain yang berkaitan dengan komunikasi antar perangkat. Sebagai contoh, ketika sistem pertama kali dijalankan akan muncul informasi System Started, kemudian dilanjutkan dengan proses System Starting, dan setelah komunikasi berhasil dilakukan akan muncul status MQTT Connected to HiveMQ Cloud. Informasi tersebut menunjukkan bahwa backend Flask telah aktif, broker MQTT berhasil dihubungi, serta ESP32 siap melakukan pertukaran data dengan website.

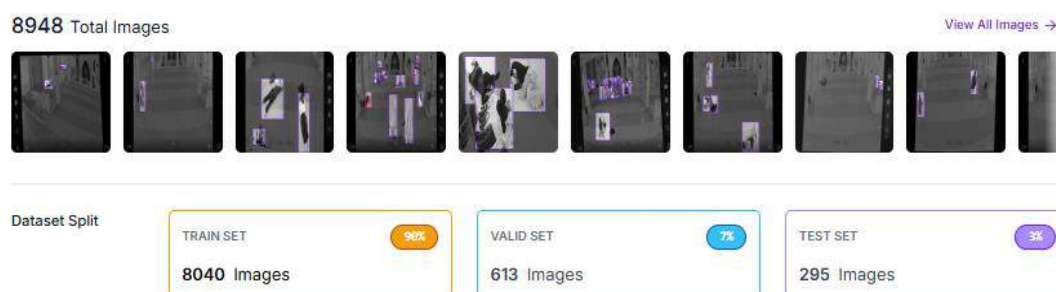


Gambar 4.19. Tampilan *System Log*
Sumber: Pribadi

Sesuai pada gambar 4.19 fungsi lain dari System Log adalah karena memiliki peranan penting selama proses implementasi maupun pengujian sistem. Melalui halaman ini, pengembang dapat mengetahui secara cepat apabila terjadi gangguan komunikasi jaringan, kegagalan koneksi MQTT, terputusnya koneksi kamera, maupun kesalahan konfigurasi perangkat. Dengan demikian proses identifikasi dan penelusuran sumber kesalahan (*troubleshooting*) dapat dilakukan lebih cepat dibandingkan hanya mengandalkan indikator pada perangkat keras.

4.4 Hasil Pelatihan Model YOLOv8m

Setelah proses pelabelan (*labeling*) dan pembagian dataset selesai dilakukan, tahapan berikutnya adalah konfigurasi serta pelatihan model deteksi objek menggunakan algoritma YOLOv8. Pada penelitian ini digunakan model YOLOv8 Medium (YOLOv8m) yang dikembangkan oleh Ultralytics. Pemilihan model YOLOv8m didasarkan pada kemampuannya dalam menghasilkan keseimbangan antara akurasi deteksi dan kecepatan inferensi. Dibandingkan dengan varian yang lebih ringan, YOLOv8m memiliki jumlah parameter yang lebih besar sehingga mampu mempelajari karakteristik objek secara lebih mendalam, namun tetap memiliki waktu inferensi yang cukup cepat untuk diterapkan pada sistem edge computing berbasis mini PC yang digunakan pada penelitian ini. Model ini digunakan untuk mendeteksi tiga kelas objek, yaitu jamaah, sampah, dan tidur, yang menjadi dasar pengambilan keputusan.



Gambar 4.20. Pembagian Dataset dari Hasil Labelling
Sumber: Pribadi

Sesuai pada gambar 4.20 diatas, dataset yang digunakan untuk penelitian saya terdiri dari 8.948 citra yang dibagi menjadi tiga kelompok, yaitu training set, validation set, dan test set. Lalu untuk pembagian proporsi dataset yang digunakan adalah seperti pada tabel 4.13 dibawah ini.

Tabel 4.13. Proporsi Pembagian Dataset

No.	Jenis Dataset	Fungsi Utama	Jumlah Citra	Persentase
1.	Training set	Melatih jaringan dan memperbarui bobot, hasilnya dapat diamati dari penurunan <i>train box loss</i> , <i>classification loss</i> , dan <i>distribution focal loss</i>	8.040	90%
2.	Validation Set	Mengevaluasi jaringan pada model selama proses training, dan menghasilkan nilai validation loss dan metrik evaluasi	613	7%
3.	Test set	Menguji bobot/model final pada data model, Hasil pengujian dapat dianalisis melalui <i>confusion matrix</i> dan <i>normalized confusion matrix</i>	295	3%
Total			8.948	100%

Pembagian dataset dengan komposisi 90%, 7%, 3% dipilih untuk memberikan porsi data pelatihan yang lebih besar sehingga model memperoleh informasi yang cukup dalam mempelajari karakteristik setiap objek. Sementara itu, sebagian data dialokasikan sebagai validation set untuk memantau proses pelatihan dan mencegah overfitting, serta sebagian lainnya digunakan sebagai test set agar evaluasi performa model dilakukan secara objektif menggunakan data yang belum pernah dikenali sebelumnya. Penjelasan masing-masing jenis dataset adalah sebagai berikut.

1. Training Set (90%)

Training set merupakan kumpulan data utama yang digunakan dalam proses pembelajaran model YOLOv8. Pada tahap ini, model mempelajari karakteristik visual dari setiap objek yang terdapat pada dataset, yaitu jamaah,

aktivitas ilegal membuang sampah, dan aktivitas ilegal menginap (tidur). Melalui proses pelatihan secara berulang (epoch), model akan menyesuaikan bobot (*weights*) jaringan saraf sehingga mampu mengenali pola, bentuk objek, posisi, ukuran, serta variasi kondisi pencahayaan yang terdapat pada lingkungan Masjid Bahrul Ulum PPNS.

Sebanyak 8.040 citra (90%) digunakan sebagai data pelatihan. Jumlah yang relatif besar dipilih agar model memperoleh variasi data yang lebih banyak sehingga kemampuan generalisasi terhadap objek yang dideteksi menjadi lebih baik. Pada training set, data digunakan secara langsung untuk memperbarui bobot jaringan melalui proses forward propagation, perhitungan loss, dan backpropagation. Hasil proses training dapat diamati dari penurunan train box loss, classification loss, dan distribution focal loss. Semakin menurun nilai loss, semakin kecil kesalahan jaringan terhadap data training.

2. Validation Set (7%)

Validation set digunakan untuk mengevaluasi performa model selama proses pelatihan berlangsung. Sebanyak 613 citra (7%) sebagai validation set, data pada kelompok ini tidak digunakan untuk memperbarui bobot model, melainkan hanya digunakan sebagai acuan dalam memantau perkembangan hasil pelatihan pada setiap epoch. Melalui validation set, dapat diketahui apakah model mengalami peningkatan performa atau justru mulai mengalami overfitting, yaitu kondisi ketika model sangat baik mengenali data pelatihan tetapi kurang mampu mengenali data baru. Validation set menghasilkan nilai validation loss dan metrik evaluasi berupa precision, recall, mAP50, serta mAP50-95 yang saya tampilkan pada results.png dan results.csv. Validation digunakan untuk memantau kemampuan generalisasi jaringan pada setiap epoch dan menentukan bobot terbaik selama proses pelatihan.

3. Test Set (3%)

Test set merupakan kumpulan data yang digunakan pada tahap akhir penelitian untuk menguji kemampuan model terhadap data yang benar-benar belum pernah dilihat sebelumnya, baik selama proses pelatihan maupun validasi. Dengan demikian, hasil pengujian menggunakan test set dapat menggambarkan kemampuan model dalam menghadapi kondisi nyata (*real-*

world performance). Sebanyak 295 citra (3%) sebagai data pengujian akhir, data tersebut digunakan untuk mengevaluasi kemampuan model YOLOv8 dalam mendeteksi tiga kelas objek, yaitu jamaah, aktivitas ilegal membuang sampah, dan aktivitas ilegal menginap (tidur). Testing set sebanyak 295 citra atau sekitar 3% digunakan setelah proses pembelajaran selesai untuk menguji bobot final best.pt pada data yang tidak digunakan saat training. Hasil pengujian dapat dianalisis melalui *confusion matrix* dan *normalized confusion matrix* sehingga dapat diketahui jumlah maupun proporsi prediksi benar dan kesalahan antar kelas.

Proses pelatihan model dilakukan menggunakan Kaggle Notebooks yang menyediakan fasilitas komputasi awan (*cloud computing*) dengan dukungan GPU NVIDIA Tesla T4. Pemanfaatan Kaggle dipilih karena menyediakan waktu penggunaan GPU gratis yang relatif lebih panjang dibandingkan platform lain sehingga mampu mempercepat proses pelatihan dataset yang berjumlah lebih dari 8.948 citra.

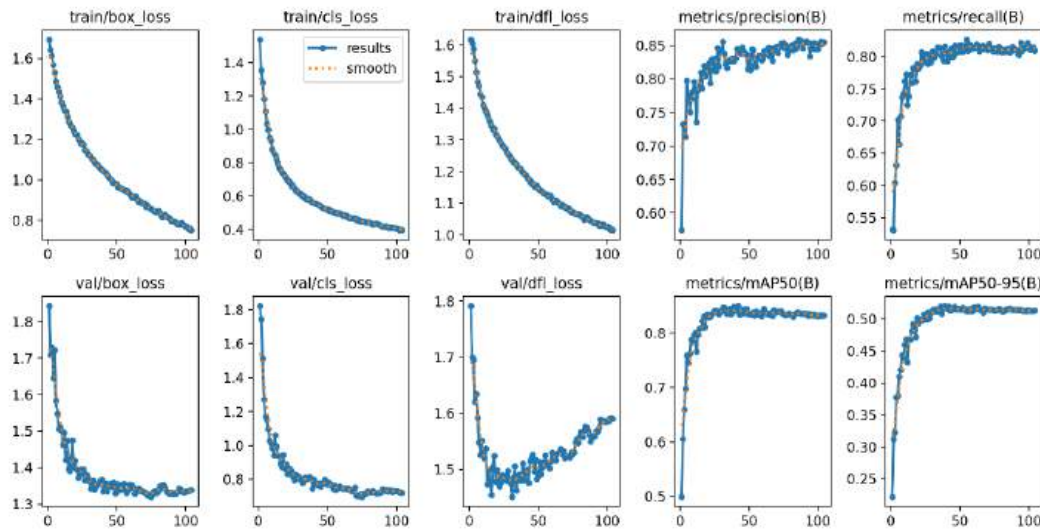
Tabel 4.14. Parameter Pelatihan YOLOv8m

Parameter	Value
Model	YOLOv8m
Image Size	640 × 640 piksel
Batch Size	16
Optimizer	Otomatis (Ultralytics)
Epoch Maksimum	220
Device	GPU NVIDIA Tesla T4
Framework	Ultralytics YOLOv8
Platform Training	Kaggle Notebooks

Parameter pelatihan yang digunakan pada penelitian ini ditunjukkan pada Tabel 4.14 diatas. Seluruh dataset yang telah melalui proses pelabelan kemudian diunggah ke Kaggle dalam format YOLOv8 sehingga dapat langsung digunakan sebagai data masukan pada proses pelatihan model.

4.4.1 Hasil Training Dataset

Proses pelatihan model YOLOv8m dilakukan sebanyak 220 epoch, dengan pemantauan terhadap berbagai metrik kinerja seperti training loss, validation loss, Precision, Recall, dan mean Average Precision (mAP).



Gambar 4.21. Result Hasil Training

Sumber: Pribadi

Pada Gambar 4.21 ditampilkan hasil proses pelatihan model YOLOv8m menggunakan dataset sebanyak 8.948 citra yang terdiri atas tiga kelas objek, yaitu *person*, *sampah*, dan *tidur*. Pelatihan dirancang dengan jumlah maksimum sebanyak 220 *epoch* menggunakan GPU NVIDIA Tesla T4 pada Kaggle Notebook serta menerapkan mekanisme *Early Stopping* dengan nilai *patience* sebesar 45. Berdasarkan hasil pelatihan, performa validasi terbaik diperoleh pada sekitar *epoch* ke-64. Proses pelatihan dihentikan secara otomatis pada sekitar *epoch* ke-109, dengan demikian, model tidak harus menyelesaikan seluruh 220 *epoch* karena proses pembelajaran telah mencapai kondisi konvergen. Bobot model dengan performa validasi terbaik pada *epoch* ke-64 kemudian disimpan sebagai file *best.pt* dan digunakan sebagai model akhir dalam proses implementasi sistem.

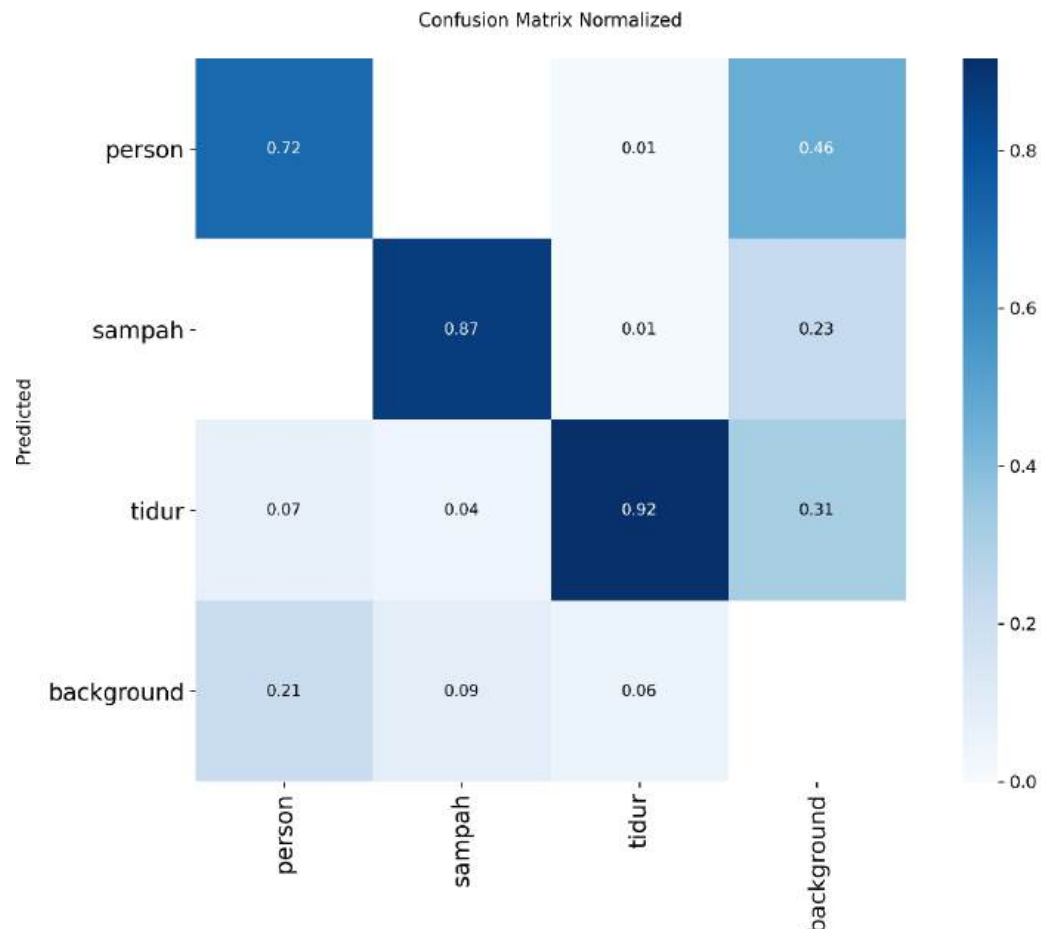
Nilai *train/box_loss* mengalami penurunan dari kisaran 1,68 menjadi sekitar 0,75 pada akhir proses pelatihan. Penurunan tersebut menunjukkan bahwa kemampuan model dalam menentukan posisi dan ukuran *bounding box* terhadap objek semakin baik. Nilai *train/cls_loss* juga mengalami penurunan yang cukup signifikan dari sekitar 1,52 menjadi sekitar 0,40, yang menunjukkan bahwa kemampuan jaringan dalam membedakan karakteristik kelas *person*, *sampah*, dan

tidur semakin meningkat. Sementara itu, nilai *train/dfl_loss* mengalami penurunan dari sekitar 1,62 menjadi sekitar 1,02. Penurunan nilai *Distribution Focal Loss* tersebut menunjukkan bahwa proses regresi batas *bounding box* semakin baik seiring bertambahnya proses pembelajaran yang dilakukan oleh model.

Pada sisi validasi, nilai *val/box_loss* pada awal pelatihan berada pada kisaran 1,84, kemudian mengalami penurunan secara cukup cepat dan selanjutnya relatif stabil pada kisaran 1,33–1,35. Nilai *val/cls_loss* juga menunjukkan penurunan yang cukup besar dari sekitar 1,82 menjadi sekitar 0,71–0,73 pada tahap akhir pelatihan. Hasil tersebut menunjukkan bahwa model mampu mempelajari karakteristik masing-masing kelas dan tetap memberikan kemampuan generalisasi yang relatif stabil pada data validasi yang tidak digunakan secara langsung untuk memperbarui bobot jaringan. Sementara itu, nilai *val/dfl_loss* mengalami penurunan dari sekitar 1,79 hingga mencapai nilai terendah pada kisaran 1,46–1,48 pada tahap awal hingga pertengahan pelatihan, kemudian secara bertahap meningkat kembali hingga sekitar 1,58–1,59 pada tahap akhir. Kondisi tersebut menunjukkan adanya kecenderungan *overfitting* ringan pada aspek lokalisasi *bounding box*, karena *training loss* terus mengalami penurunan sedangkan sebagian *validation loss* tidak lagi menunjukkan peningkatan yang berarti. Meskipun terdapat kecenderungan peningkatan pada *val/dfl_loss*, kondisi tersebut tidak menyebabkan penurunan performa model secara drastis. Hal ini dapat diamati dari nilai *Precision*, *Recall*, *mAP@0.5*, dan *mAP@0.5:0.95* yang relatif stabil setelah model mencapai kondisi konvergen. Mekanisme *Early Stopping* dengan nilai *patience* sebesar 45 digunakan untuk mencegah proses pembelajaran berlangsung terlalu lama ketika performa validasi tidak lagi mengalami peningkatan. Model terbaik tidak ditentukan berdasarkan *epoch* terakhir, melainkan berdasarkan nilai *fitness* validasi terbaik yang diperoleh selama proses pelatihan.

Berdasarkan grafik metrik evaluasi, nilai *Precision* mengalami peningkatan secara cepat pada tahap awal pelatihan dan kemudian relatif stabil pada kisaran 0,84–0,86, dengan nilai akhir sekitar 0,85. Nilai tersebut menunjukkan bahwa sebagian besar prediksi objek yang dihasilkan model memiliki kesesuaian dengan objek sebenarnya sehingga jumlah *false positive* dapat ditekan. Nilai *Recall* juga meningkat secara signifikan dan kemudian stabil pada kisaran 0,81–0,82. Kondisi

tersebut menunjukkan bahwa model mampu menemukan sebagian besar objek yang tersedia pada data validasi. Grafik $mAP@0.5$ menunjukkan peningkatan dari sekitar 0,50 pada tahap awal hingga mencapai kisaran 0,83–0,84, sedangkan nilai $mAP@0.5:0.95$ meningkat hingga kisaran 0,51–0,52. Setelah mencapai nilai tertinggi, kedua metrik tersebut cenderung stabil hingga proses pelatihan dihentikan oleh mekanisme *Early Stopping*.



Gambar 4. 22. Hasil Confusion Matrix Normalized
Sumber: Pribadi

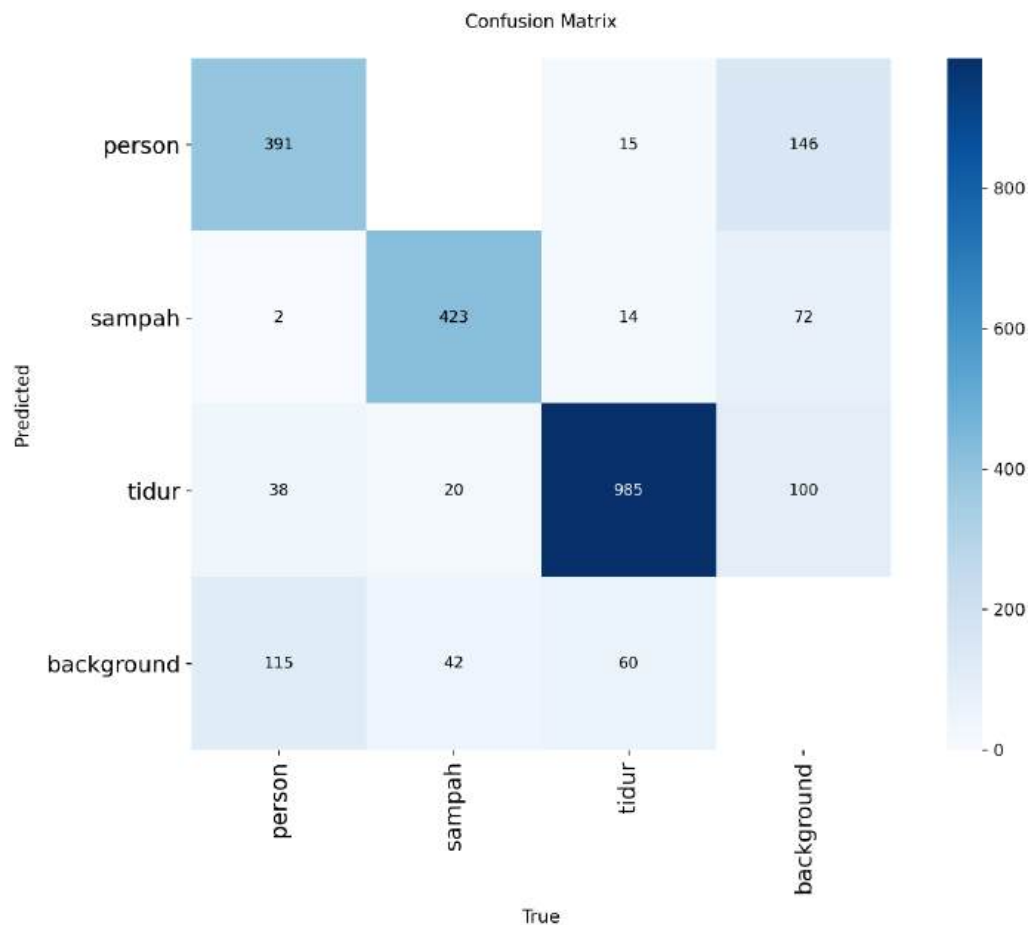
Berdasarkan *Normalized Confusion Matrix* pada Gambar 4.22, kelas *person* atau jamaah memiliki tingkat prediksi benar sebesar 0,72 atau 72%. Artinya, sekitar 72% *ground truth* kelas *person* berhasil memperoleh prediksi yang sesuai sebagai kelas *person*. Sekitar 7% kelas *person* salah diprediksi sebagai *tidur*, kurang dari 1% diprediksi sebagai *sampah*, sedangkan sekitar 21% tidak memperoleh pasangan prediksi yang sesuai sehingga masuk ke kategori *background*. Kesalahan tersebut menunjukkan bahwa kelas *person* masih menjadi kelas dengan tingkat deteksi benar paling rendah dibandingkan dua kelas lainnya. Kondisi ini dapat terjadi pada

jamaah yang berada jauh dari kamera, berukuran relatif kecil, mengalami *occlusion*, berada pada sisi atau sudut citra CCTV, maupun memiliki bentuk tubuh yang berubah akibat posisi duduk, membungkuk, sujud, atau penggunaan pakaian seperti mukena sehingga secara visual memiliki karakteristik yang berbeda dibandingkan posisi berdiri normal.

Pada kelas *sampah*, model memperoleh tingkat prediksi benar sebesar 0,87 atau 87%. Berdasarkan *Normalized Confusion Matrix*, sekitar 4% *ground truth* kelas *sampah* salah diprediksi sebagai *tidur*, sedangkan sekitar 9% tidak memperoleh pasangan deteksi sehingga masuk ke *background*. Kesalahan antara kelas *sampah* dan *person* relatif sangat kecil. Hasil tersebut menunjukkan adanya peningkatan kemampuan model dalam membedakan aktivitas membuang sampah dibandingkan hasil pelatihan sebelumnya. Meskipun demikian, masih terdapat sebagian aktivitas membuang sampah yang teridentifikasi sebagai aktivitas tidur. Hal ini dapat terjadi karena gerakan membungkuk, menurunkan badan, atau posisi tangan ketika meletakkan maupun membuang suatu objek dapat menghasilkan pola visual yang memiliki kemiripan dengan beberapa bagian karakteristik kelas tidur, terutama ketika ukuran objek relatif kecil atau posisi manusia berada jauh dari kamera. Sementara itu, kelas *tidur* memperoleh tingkat prediksi benar tertinggi sebesar 0,92 atau 92%. Sekitar 1% *ground truth* kelas tidur salah diprediksi sebagai *person*, sekitar 1% salah diprediksi sebagai *sampah*, dan sekitar 6% tidak berhasil memperoleh pasangan prediksi sehingga masuk ke *background*. Hasil tersebut menunjukkan bahwa model telah mampu mempelajari karakteristik visual kelas tidur dengan lebih baik dibandingkan dua kelas lainnya. Tingginya tingkat prediksi benar pada kelas tidur menunjukkan bahwa pola orientasi tubuh, posisi tubuh yang relatif horizontal, serta karakteristik objek pada aktivitas tidur telah dapat dikenali secara cukup konsisten oleh model.

Tahap selanjutnya dilakukan evaluasi terhadap bobot jaringan terbaik menggunakan *testing set* melalui *Confusion Matrix* sebagaimana ditunjukkan pada Gambar 4.23. Pada *Confusion Matrix* biasa, nilai ditampilkan dalam bentuk jumlah instance yang berhasil maupun gagal diklasifikasikan oleh model. Berdasarkan hasil tersebut, terdapat sebanyak 391 instance kelas *person* yang berhasil diprediksi dengan benar sebagai *person*. Sebanyak 2 instance *person* salah diprediksi sebagai

sampah, 38 instance salah diprediksi sebagai *tidur*, sedangkan 115 instance tidak memperoleh pasangan prediksi yang sesuai sehingga masuk ke *background*.



Gambar 4.23. Hasil Confusion Matrix
Sumber: Pribadi

Dengan demikian, jumlah keseluruhan *ground truth* kelas *person* yang dievaluasi adalah sebanyak 546 instance. Pada kelas *sampah*, sebanyak 423 instance berhasil diklasifikasikan dengan benar sebagai *sampah*, tidak terdapat instance yang diprediksi sebagai *person* pada pembulatan matriks tersebut, sebanyak 20 instance diprediksi sebagai *tidur*, dan 42 instance masuk ke *background*. Dengan demikian, terdapat sebanyak 485 *ground truth instance* kelas *sampah* yang dievaluasi. Pada kelas *tidur*, sebanyak 985 instance berhasil diprediksi dengan benar sebagai *tidur*, sedangkan 15 instance salah diprediksi sebagai *person*, 14 instance diprediksi sebagai *sampah*, dan 60 instance tidak memperoleh pasangan prediksi sehingga masuk ke *background*. Jumlah keseluruhan *ground truth instance* kelas *tidur* yang dievaluasi adalah sebanyak 1.074 instance. Apabila ketiga kelas dijumlahkan, *validation set* tersebut memuat

sebanyak 2.105 instance, yang tersebar pada 613 citra validasi. Hasil tersebut menunjukkan bahwa sebagian besar instance pada ketiga kelas berada pada diagonal utama *Confusion Matrix*, terutama pada kelas *sampah* dan *tidur*.

Pada kolom *background* juga ditemukan sebanyak 146 prediksi kelas *person*, 72 prediksi kelas *sampah*, dan 100 prediksi kelas *tidur* yang tidak memiliki pasangan dengan anotasi *ground truth*. Nilai tersebut merepresentasikan *false positive* atau prediksi objek yang tidak berhasil dipasangkan dengan objek sebenarnya pada proses evaluasi. Nilai 0,46, 0,23, dan 0,31 pada kolom *background* pada *Normalized Confusion Matrix* tidak berarti bahwa 46%, 23%, dan 31% seluruh citra latar belakang salah dideteksi, melainkan menunjukkan proporsi dari keseluruhan prediksi yang tidak memiliki pasangan *ground truth* pada kolom *background*. Hasil pelatihan menunjukkan bahwa model YOLOv8m telah mencapai kondisi konvergen sebelum jumlah maksimum 220 *epoch*. Hal tersebut dibuktikan dengan diperolehnya bobot terbaik pada sekitar *epoch* ke-64 dan dihentikannya proses pelatihan pada sekitar *epoch* ke-109 melalui mekanisme *Early Stopping*. Model memperoleh nilai *Precision* sekitar 0,85, *Recall* sekitar 0,81–0,82, *mAP@0.5* sekitar 0,83–0,84, serta *mAP@0.5:0.95* sekitar 0,51–0,52. Berdasarkan *Normalized Confusion Matrix*, tingkat prediksi benar masing-masing kelas mencapai 72% untuk kelas *person*, 87% untuk kelas *sampah*, dan 92% untuk kelas *tidur*.

4.4.2 Pengujian Model YOLOv8 terhadap Seluruh Kelas Objek

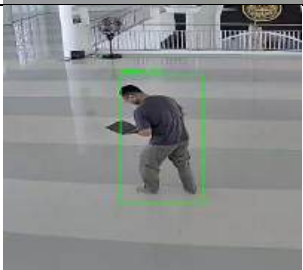
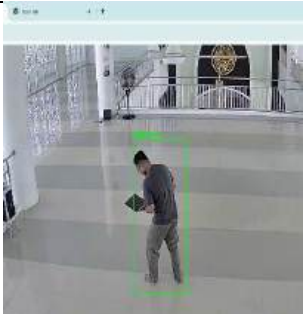
Dataset yang digunakan terdiri atas 8.948 citra yang terbagi menjadi tiga kelas utama, yaitu *jamaah*, *sampah*, dan *tidur*, dengan proses pelatihan menggunakan platform Kaggle Notebook yang memanfaatkan GPU NVIDIA Tesla T4. Meskipun jumlah *epoch* yang ditentukan sebanyak 220 *epoch*, proses pelatihan berhenti secara otomatis pada *epoch* ke-109 karena mekanisme *Early Stopping*, di mana model tidak lagi menunjukkan peningkatan performa selama 45 *epoch* berturut-turut. Berdasarkan hasil pelatihan tersebut, model terbaik (*best.pt*) diperoleh pada *epoch* ke-64, sehingga model tersebut digunakan sebagai dasar pada seluruh proses pengujian. Pengujian dilakukan menggunakan data validasi yang tidak pernah digunakan selama proses pelatihan sehingga hasil evaluasi mampu menggambarkan kemampuan generalisasi model terhadap data baru. Setiap objek

yang muncul pada citra diuji berdasarkan hasil prediksi berupa *bounding box*, nama kelas, dan nilai *confidence score* yang dihasilkan oleh model. Selain mengevaluasi keberhasilan klasifikasi objek, pengujian juga bertujuan untuk mengetahui kemampuan model dalam menentukan lokasi objek secara tepat pada berbagai kondisi pencahayaan, jumlah objek, sudut pengambilan gambar, maupun ukuran objek yang berbeda.

Secara keseluruhan, pengujian model bertujuan untuk memastikan bahwa model YOLOv8m mampu bekerja secara stabil sebelum diintegrasikan dengan laptop, ESP32, CCTV IP Camera, web *monitoring* berbasis Cloudflare Tunnel, komunikasi MQTT HiveMQ Cloud, serta database Supabase. Model yang memiliki tingkat akurasi dan konsistensi deteksi yang baik diharapkan mampu mendukung proses pengambilan keputusan secara otomatis, seperti pengendalian lampu, pengaturan kipas, pendeteksian aktivitas tidur di luar jam operasional masjid, serta pendeteksian sampah secara *real-time*.

Tabel 4.15. Pengujian kelas Jamaah

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
1.		Jamaah	Jamaah	0.80	Benar
2.		Jamaah	Jamaah	0.81	Benar
3		Jamaah	Jamaah	0.80	Benar

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
4		Jamaah	Jamaah	0.78	Benar
5		Jamaah	Jamaah	0.84	Benar
6		Jamaah	Jamaah	0.79	Benar
7		Jamaah	Jamaah	0.88	Benar
8		Jamaah	Jamaah	0.74	Benar
9		Jamaah	Jamaah	0.82	Benar

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
10		Jamaah	Jamaah	0.77	Benar

Berdasarkan hasil pengujian yang tercantum pada Tabel 4.15, model YOLOv8m berhasil mendeteksi seluruh citra dalam kelas Jamaah secara tepat sesuai dengan label aslinya. Rentang nilai confidence score berada antara 0.74 hingga 0.88, dengan rata-rata mencapai 0.80. Selain menggunakan objek yang terdaftar dalam label, juga menggunakan objek yang tidak terdaftar (kondisi jamaah real) dan juga berhasil terdeteksi. Hal ini mengindikasikan bahwa model memiliki kemampuan yang andal dan konsisten dalam mengidentifikasi objek dari kategori tersebut. Stabilitas nilai confidence menunjukkan bahwa proses deteksi berjalan dengan baik, serta model mampu mempertahankan tingkat keyakinan tinggi terhadap hasil klasifikasi.

Tabel 4.16. Pengujian Kelas *Illegal Activity* Tidur di Masjid

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
1.		Tidur	Tidur	0.85	Benar
2.		Tidur	Tidur	0.84	Benar

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
3		Tidur	Tidur	0.83	Benar
4		Tidur	Tidur	0.85	Benar
5		Tidur	Tidur	0.84	Benar
6		Tidur	Tidur	0.84	Benar
7		Tidur	Tidur	0.86	Benar
8		Tidur	Tidur	0.87	Benar

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
9		Tidur	Tidur	0.83	Benar
10		Tidur	Tidur	0.86	Benar

Berdasarkan hasil pengujian terhadap 10 citra pada kelas ilegal aktivitas tidur sebagaimana disajikan pada Tabel 4.16, model YOLOv8m menunjukkan kemampuan klasifikasi yang baik. Seluruh objek berhasil dikenali dan diklasifikasikan secara benar sesuai dengan label aktual yang telah ditentukan dalam dataset dengan nilai confidence tertinggi untuk class tidur sebesar 0,87. Nilai confidence score yang dihasilkan selama pengujian berada dalam rentang 0.83 hingga 0.87, dengan rata-rata sebesar 0.85 dari berbagai macam posisi untuk aktivitas tidur di masjid. Selain menggunakan objek yang terdaftar dalam labelling yolo, pengambilan data pengujian juga menggunakan objek yang tidak termasuk dalam proses labelling yolo, sehingga bisa lebih meyakinkan bahwa sistem deteksi dapat bekerja dengan baik dan sesuai yang diharapkan.

Tabel 4.17. Pengujian Kelas Illegal Activity Tidur di Masjid

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
1.		Membuang Sampah	Membuang Sampah	0.66	Benar

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
2.		Membuang Sampah	Membuang Sampah	0.76	Benar
3		Membuang Sampah	Membuang Sampah	0.90	Benar
4		Membuang Sampah	Membuang Sampah	0.76	Benar
5		Membuang Sampah	Membuang Sampah	0.88	Benar
6		Membuang Sampah	Membuang Sampah	0.82	Benar
7		Membuang Sampah	Membuang Sampah	0.76	Benar

No	Gambar Citra	Klasifikasi		Confidence Score	Keterangan
		Aktual	Deteksi		
8		Membuang Sampah	Membuang Sampah	0.83	Benar
9		Membuang Sampah	Membuang Sampah	0.74	Benar
10		Membuang Sampah	Membuang Sampah	0.88	Benar

Berdasarkan hasil pengujian terhadap 10 citra pada kelas membuang sampah sebagaimana ditampilkan pada Tabel 4.17, model YOLOv8m berhasil mengklasifikasikan seluruh citra dengan label yang sesuai dengan kondisi aktual. Nilai confidence score yang diperoleh berada pada rentang 0.66 hingga 0.90, dengan rata-rata sebesar 0.79. Seluruh klasifikasi dinyatakan benar, dengan berbagai macam aktivitas yang dicurigai sebagai aktivitas membuang sampah yaitu menjulurkan tangan dan terdapat sampah disekitar objek. Sehingga rata-rata confidence yang masih cukup tinggi menunjukkan bahwa model memiliki performa yang cukup baik.

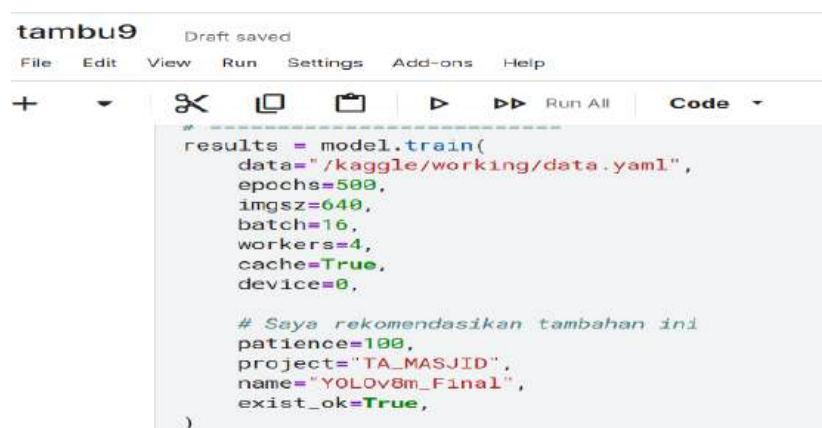
4.5 Hasil Pengujian *Before-After* Menggunakan Metode YOLOv8

Pemilihan saya untuk memilih YOLOv8 didasarkan pada kemampuannya dalam mengintegrasikan proses deteksi dan klasifikasi objek dalam satu tahap (*single-stage detector*) sehingga menghasilkan kinerja yang lebih cepat dan efisien dibandingkan metode dua tahap. Selain itu, YOLOv8 juga termasuk dalam jenis

kategori metode *deep learning* yang cukup ringan untuk dijalankan dan di *training* terutama dengan jumlah dataset yang kurang lebih 3.022 serta telah mengalami peningkatan pada sisi arsitektur *backbone*, *neck*, dan *head*, yang memungkinkan model untuk bekerja lebih optimal pada kondisi pencahayaan *indoor*, variasi sudut pandang kamera, serta kepadatan objek yang dinamis seperti pada lingkungan masjid. Berikut adalah hasil pengujian metode dan perbandingan antara tidak dan memakai metode.

4.5.1 Hasil Integrasi Metode YOLOv8 pada Sistem

Pada tahap pengembangan model, dataset yang digunakan merupakan hasil akumulasi dan akurasi data citra sebanyak lebih dari 8.948 gambar dengan 3 class, yaitu deteksi jamaah, deteksi aktivitas ilegal membuang sampah sembarangan di masjid dan deteksi aktivitas ilegal tidur/ menginap di masjid. Keseluruhan data diperoleh dari berbagai kondisi nyata di lapangan. Setelah proses *training* selesai, model terbaik (best.pt) kemudian diimplementasikan ke dalam sistem *backend* berbasis Python yang terintegrasi dengan *framework Flask*. Gambar training ditunjukkan pada gambar 4.24 dibawah ini.



```
results = model.train(
    data="/kaggle/working/data.yaml",
    epochs=500,
    imgsz=640,
    batch=16,
    workers=4,
    cache=True,
    device=0,

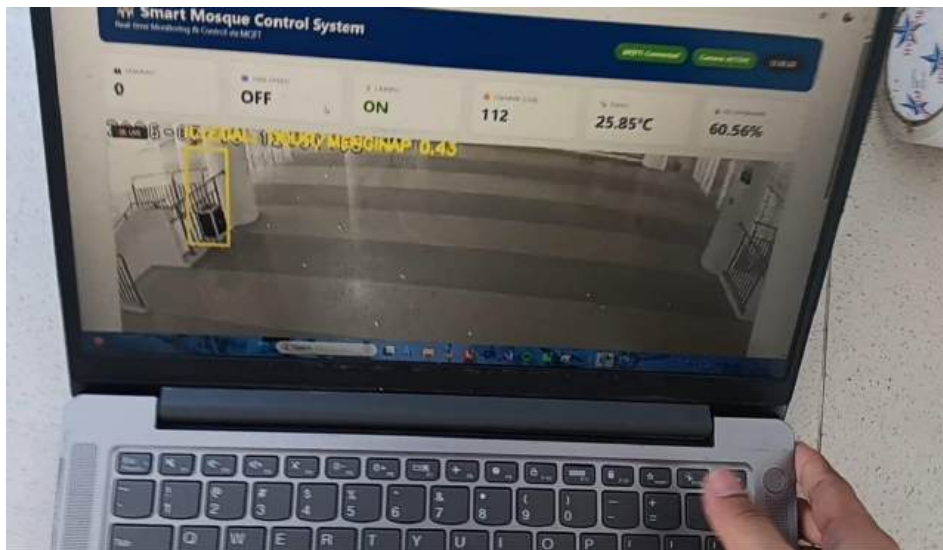
    # Saya rekomendasikan tambahan ini
    patience=100,
    project="TA_MASJID",
    name="YOLOv8m_Final",
    exist_ok=True,
)
```

Gambar 4.24. Percobaan Training 8.948 Dataset dari 3 class, dengan YOLOv8
Sumber: Pribadi

Dataset ini mencakup variasi posisi jamaah seperti berdiri, duduk, bergerak, serta kondisi berkelompok dengan tingkat kepadatan yang berbeda. Proses anotasi dilakukan menggunakan platform Roboflow dengan pemberian *bounding box* pada setiap objek manusia yang terdeteksi, sehingga menghasilkan dataset yang terstruktur dan siap digunakan untuk proses *training*. Selain itu, dilakukan pula tahap *preprocessing* berupa *resizing* citra, normalisasi, serta augmentasi data seperti rotasi, *flipping*, dan perubahan pencahayaan guna meningkatkan kemampuan

generalisasi model terhadap kondisi lingkungan yang bervariasi. Model ini digunakan untuk melakukan inferensi secara langsung terhadap video stream yang diperoleh dari kamera CCTV berbasis RTSP, yaitu TP-Link Tapo C212. Berdasarkan hasil pengujian sistem *backend*, diperoleh bahwa proses akuisisi data visual melalui kamera CCTV berbasis RTSP dapat berjalan secara kontinu dengan mekanisme *auto-reconnect* ketika terjadi kegagalan pembacaan frame.

Selanjutnya pada gambar 4.25 beberapa kondisi, sistem mencatat peringatan berupa kegagalan pembacaan frame yang disebabkan oleh fluktuasi jaringan, namun sistem mampu melakukan inisialisasi ulang koneksi secara otomatis tanpa menghentikan proses utama.



Gambar 4.25. Konektivitas Hasil *Training* YOLOv8 dengan Web dan Kamera
Sumber: Pribadi

Selain itu, server berbasis Flask menunjukkan respons yang stabil dengan seluruh endpoint API berhasil diakses (HTTP status 200). Pengujian terhadap konektivitas mikrokontroler/ pcb ESP32 belum dilakukan secara langsung pada tahap ini, sehingga sistem masih berjalan dalam mode simulasi tanpa pengiriman data serial. Secara keseluruhan, sistem *backend* telah menunjukkan performa yang stabil dan siap untuk tahap integrasi perangkat keras.

4.5.2 Perbandingan Dengan atau Tidak Menggunakan Metode YOLOv8

Setelah melakukan pengujian *system object detection* dengan atau tidaknya menggunakan metode YOLOv8 Berdasarkan hasil pengujian pada dua kondisi sistem pemantauan masjid, terlihat adanya perbedaan yang cukup signifikan antara

sistem yang menerapkan metode deteksi objek YOLOv8 dan sistem CCTV konvensional tanpa metode pendeteksian.



Gambar 4.26. *Object Detection* Tanpa Metode YOLO
Sumber: Pribadi

Sementara itu, pada gambar 4.25, sistem hanya menampilkan tampilan kamera biasa tanpa adanya proses identifikasi objek ataupun analisis jumlah jamaah secara otomatis. Penerapan metode YOLOv8 memberikan peningkatan kemampuan sistem dalam melakukan pemantauan kondisi ruangan secara *real-time*. Sistem tidak hanya berfungsi sebagai media pengawasan visual, tetapi juga mampu melakukan analisis terhadap keberadaan manusia di area masjid. Deteksi dilakukan secara otomatis dengan memberikan penanda pada objek manusia yang berhasil dikenali oleh model. Hal ini memungkinkan sistem untuk menghitung jumlah jamaah secara langsung dan mengirimkan data tersebut ke mikrokontroller ESP32 sebagai dasar pengendalian kipas dan lampu otomatis. CCTV hanya menampilkan kondisi ruangan tanpa adanya kemampuan analisis objek. Sekelompok wanita yang sedang berkumpul di bagian kiri bawah gambar tidak dapat dikenali ataupun dihitung secara otomatis oleh sistem. Akibatnya, pengendalian perangkat seperti kipas dan lampu tidak dapat dilakukan berdasarkan kondisi aktual jumlah jamaah di dalam ruangan.



Gambar 4.27. Tampilan Monitor CCTV
Sumber: Pribadi

Sedangkan pada gambar 4.26, sistem yang sudah menggunakan metode YOLO terlihat bahwa objek manusia yang sedang berjalan dengan membawa barang yang terindikasi dapat menghasilkan sampah, berhasil dikenali dengan baik meskipun posisi objek cukup jauh dari kamera. Bounding box dan label deteksi menunjukkan bahwa model YOLOv8 mampu melakukan identifikasi secara stabil pada kondisi pencahayaan ruangan yang terang dan area masjid yang cukup luas.

Tabel 4.18. Perbandingan Penggunaan Metode YOLO

Aspek Perbandingan	Menggunakan YOLOv8	Tanpa YOLOv8
Deteksi Manusia	Otomatis terdeteksi dan diberi label	Tidak ada deteksi objek
<i>Bounding Box</i> Objek	Ada	Tidak ada
Integrasi dengan ESP32	Terhubung otomatis	Tidak terhubung analisis
Akurasi Pemantauan	Lebih Akurat	Terbatas visual manual
Pengenalan Objek	Terdapat klasifikasi khusus	Semua objek sama

Selanjutnya pada tabel 4.18, dari sisi akurasi pemantauan, metode YOLOv8 juga menunjukkan keunggulan karena mampu melakukan deteksi secara cepat dan responsif. Sistem dapat mengenali objek manusia dalam hitungan milidetik sehingga perubahan kondisi ruangan dapat segera ditindaklanjuti oleh ESP32. Sedangkan pada sistem CCTV biasa, seluruh proses pengamatan bergantung pada operator manusia yang memiliki keterbatasan dalam memantau ruangan secara terus-menerus.

4.6 Hasil Perbandingan Before-After Terhadap Pengeluaran Daya Masjid

Dalam membandingkan besaran arus pemakaian listrik pada Masjid baik sebelum atau sesudah merupakan hal yang harus dilakukan dengan penuh ketelitian. Selain dipengaruhi oleh sistem pengendalian yang masih manual, besarnya arus yang terukur juga dipengaruhi oleh aktivitas dan kepadatan jamaah yang menggunakan fasilitas di lantai 2. Pengambilan data dilakukan pada hari kerja, di mana aktivitas mahasiswa, dosen, maupun sivitas akademika PPNS masih berlangsung secara normal. Pada waktu-waktu tertentu, terutama menjelang waktu salat berjamaah, jumlah pengguna ruangan meningkat sehingga penggunaan lampu

dan kipas juga lebih tinggi. Sebaliknya, ketika jumlah jamaah berkurang, kondisi peralatan listrik tidak selalu mengikuti perubahan aktivitas tersebut karena belum terdapat sistem yang mampu mengendalikan beban secara otomatis. Oleh sebab itu, arus listrik yang mengalir cenderung tetap tinggi walaupun kebutuhan energi sebenarnya telah menurun.

4.6.1 Hasil Pengukuran Sebelum Implementasi Sistem

Sistem Kelitrikan di lantai 2 Masjid Bahrul Ulum PPNS menggunakan rangkaian 1 fasa yang dibagi menjadi rangkaian Arus R, S dan T. Berdasarkan hasil pengukuran arus listrik sebelum implementasi sistem kontrol cerdas berbasis IoT, diperoleh nilai rata-rata arus pada fasa R sebesar 7,50 A, fasa S sebesar 0,88 A, dan fasa T sebesar 3,05 A. Nilai tersebut menunjukkan bahwa beban listrik pada lantai 2 Masjid Bahrul Ulum PPNS belum memiliki mekanisme pengendalian otomatis, sehingga lampu dan kipas angin masih beroperasi secara manual sesuai kebiasaan pengguna. Akibatnya, peralatan listrik sering kali tetap menyala meskipun jumlah jamaah telah berkurang atau bahkan ruangan dalam kondisi kosong. Kondisi tersebut menyebabkan konsumsi arus pada fasa R menjadi relatif tinggi dibandingkan dua fasa lainnya karena sebagian besar beban penerangan dan kipas pada area penelitian terhubung pada jalur tersebut. Selanjutnya pada tabel 4.19 dibawah ini adalah hasil dari pengukuran daya listrik masjid yang dibutuhkan untuk fasilitas listrik di Masjid Bahrul Ulum PPNS sebelum adanya implementasi sistem.

Tabel 4.19. Pengukuran Daya Sebelum Adanya Sistem

No	Waktu Pengukuran	Tegangan Antar Fasa	Hasil Pengukuran	Dokumentasi
1.	17.00	Kabel Fasa R	7.84A	
		Kabel Fasa S	0.89A	
		Kabel Fasa T	3.10A	
2.	17.20	Kabel Fasa R	7.89A	
		Kabel Fasa S	0.98A	
		Kabel Fasa T	3.07A	
3.	17.40	Kabel Fasa R	7.81A	

No	Waktu Pengukuran	Tegangan Antar Fasa	Hasil Pengukuran	Dokumentasi
		Kabel Fasa S	0.86A	
		Kabel Fasa T	3.09A	
4.	18.00	Kabel Fasa R	7.71A	
		Kabel Fasa S	0.81A	
		Kabel Fasa T	3.03A	
5.	18.20	Kabel Fasa R	7.77A	
		Kabel Fasa S	0.91A	
		Kabel Fasa T	3.04A	
6.	18.40	Kabel Fasa R	7.35A	
		Kabel Fasa S	0.87A	
		Kabel Fasa T	3.09A	
7.	20.00	Kabel Fasa R	7.44A	
		Kabel Fasa S	0.82A	
		Kabel Fasa T	2.99A	
8.	20.20	Kabel Fasa R	7.38A	
		Kabel Fasa S	0.86A	
		Kabel Fasa T	3.06A	
9.	20.40	Kabel Fasa R	7.46A	
		Kabel Fasa S	0.89A	
		Kabel Fasa T	3.02A	
10.	21.00	Kabel Fasa R	7.36A	
		Kabel Fasa S	0.87A	
		Kabel Fasa T	3.03A	

Dari hasil pengukuran tersebut dapat disimpulkan bahwa kondisi sebelum implementasi sistem masih menunjukkan penggunaan energi listrik yang kurang efisien. Hal ini disebabkan oleh pengoperasian peralatan listrik yang sepenuhnya bergantung pada pengguna, sehingga potensi pemborosan energi masih cukup besar. Untuk memperoleh nilai hasil pengujian sebelum adanya sistem, digunakan perhitungan rata-rata daya berdasarkan seluruh data hasil pengukuran. Nilai rata-rata daya dihitung menggunakan Persamaan (2.10), seperti dibawah ini:

Perhitungan Rata-Rata Arus Sebelum Implementasi Sistem

- **Fasa R, Data:**

$$7.84 + 7.89 + 7.81 + 7.71 + 7.77 + 7.35 + 7.44 + 7.38 + 7.46 + 7.36 = 75.01 \text{ A}$$

$$\text{Rata-rata } I_R = \frac{75.01}{10} = 7.501 \text{ A}$$

- **Fasa S, Data:**

$$0.89 + 0.98 + 0.86 + 0.81 + 0.91 + 0.87 + 0.82 + 0.86 + 0.89 + 0.87 = 8.76 \text{ A}$$

$$\text{Rata-rata } I_S = \frac{8.76}{10} = 0.876 \text{ A}$$

- **Fasa T, Data:**

$$3.10 + 3.07 + 3.09 + 3.03 + 3.04 + 3.09 + 2.99 + 3.06 + 3.02 + 3.03 = 30.52 \text{ A}$$

$$\text{Rata-rata } I_T = \frac{30.52}{10} = 3.052 \text{ A}$$

4.6.2 Hasil Pengukuran Sesudah Implementasi Sistem

Setelah sistem kontrol cerdas berbasis IoT diimplementasikan, dilakukan kembali pengukuran arus listrik pada titik pengukuran yang sama. Berdasarkan hasil pengujian diperoleh nilai rata-rata arus sebesar 5,10 A pada fasa R, 1,29 A pada fasa S, dan 3,04 A pada fasa T. Dibandingkan dengan kondisi sebelum implementasi, terlihat adanya penurunan arus yang cukup signifikan pada fasa R, sedangkan pada fasa S terjadi sedikit peningkatan dan pada fasa T nilainya relatif tetap.

Penurunan arus pada fasa R menunjukkan bahwa sistem yang dikembangkan mampu mengurangi konsumsi daya pada beban utama, yaitu lampu dan kipas yang dikendalikan secara otomatis menggunakan deteksi objek berbasis YOLOv8, sensor LDR, sensor RTC, serta ESP32 sebagai pengendali utama. Ketika kamera tidak mendeteksi keberadaan jamaah atau kondisi pencahayaan ruangan masih mencukupi, sistem secara otomatis mematikan lampu maupun kipas sesuai

logika yang telah dirancang. Mekanisme tersebut membuat beban listrik hanya aktif ketika benar-benar diperlukan, sehingga konsumsi arus pada fasa R mengalami penurunan dibandingkan kondisi sebelum implementasi.

Tabel 4.20. Pengukuran Daya Sesudah Adanya Sistem

No	Waktu Pengukuran	Tegangan Antar Fasa	Hasil Pengukuran	Dokumentasi
1.	12.10	Kabel Fasa R	4.88A	
		Kabel Fasa S	1.37A	
		Kabel Fasa T	3.03A	
2.	12.30	Kabel Fasa R	5.35A	
		Kabel Fasa S	1.36A	
		Kabel Fasa T	2.97A	
3.	12.50	Kabel Fasa R	5.26A	
		Kabel Fasa S	1.37A	
		Kabel Fasa T	3.02A	
4.	13.10	Kabel Fasa R	5.54A	
		Kabel Fasa S	1.27A	
		Kabel Fasa T	3.06A	
5.	13.30	Kabel Fasa R	5.67A	
		Kabel Fasa S	1.26A	
		Kabel Fasa T	3.06A	
6.	13.50	Kabel Fasa R	5.66A	
		Kabel Fasa S	1.24A	
		Kabel Fasa T	3.04A	
7.	14.10	Kabel Fasa R	4.70A	
		Kabel Fasa S	1.20A	
		Kabel Fasa T	3.06A	
8.	14.30	Kabel Fasa R	4.58A	
		Kabel Fasa S	1.27A	
		Kabel Fasa T	3.06A	

9.	14.50	Kabel Fasa R	4.69A	
		Kabel Fasa S	1.25A	
		Kabel Fasa T	3.08A	
10.	15.10	Kabel Fasa R	4.70A	
		Kabel Fasa S	1.28A	
		Kabel Fasa T	3.02A	

Penurunan arus pada fasa R menunjukkan bahwa sistem yang dikembangkan mampu mengurangi konsumsi daya pada beban utama, yaitu lampu dan kipas yang dikendalikan secara otomatis menggunakan deteksi objek berbasis YOLOv8, sensor LDR, sensor RTC, serta ESP32 sebagai pengendali utama. Ketika kamera tidak mendeteksi keberadaan jamaah atau kondisi pencahayaan ruangan masih mencukupi, sistem secara otomatis mematikan lampu maupun kipas sesuai logika yang telah dirancang. Mekanisme tersebut membuat beban listrik hanya aktif ketika benar-benar diperlukan, sehingga konsumsi arus pada fasa R mengalami penurunan dibandingkan kondisi sebelum implementasi.

Sementara itu, kenaikan arus pada fasa S tidak secara langsung menunjukkan bahwa sistem menjadi kurang efisien. Pengukuran dilakukan pada hari kerja sehingga jumlah jamaah yang hadir pada setiap waktu pengujian tidak selalu sama. Selanjutnya, digunakan perhitungan rata-rata daya setelah adanya sistem berdasarkan seluruh data hasil pengukuran. Nilai rata-rata daya dihitung menggunakan persamaan (2.10), seperti dibawah:

Perhitungan Rata-Rata Arus Sesudah Implementasi Sistem

- **Fasa R, Data:**

$$4.88 + 5.35 + 5.26 + 5.54 + 5.67 + 5.66 + 4.70 + 4.58 + 4.69 + 4.70 = 51.03 \text{ A}$$

$$\text{Rata-rata } I_R = \frac{51.03}{10} = 5.103 \text{ A}$$

- **Fasa S, Data:**

$$1.37 + 1.36 + 1.37 + 1.27 + 1.26 + 1.24 + 1.20 + 1.27 + 1.25 + 1.28 = 12.87 \text{ A}$$

$$\text{Rata-rata } I_S = \frac{12.87}{10} = 1.287 \text{ A}$$

- **Fasa T, Data:**

$$3.03 + 2.97 + 3.02 + 3.06 + 3.06 + 3.04 + 3.06 + 3.06 + 3.08 + 3.02 = 30.40 \text{ A}$$

$$\text{Rata-rata } I_T = \frac{30.40}{10} = 3.04A$$

Perhitungan Penghematan Daya Terhadap Implementasi Sistem di Masjid

Daya listrik merupakan besarnya energi listrik yang digunakan oleh suatu peralatan dalam satuan waktu. Pada sistem listrik AC satu fasa, daya aktif dipengaruhi oleh nilai tegangan, arus, dan faktor daya. Besarnya daya aktif dapat dihitung menggunakan persamaan (2.9). dengan perhitungan seperti dibawah ini:

1. Sebelum, $P_{awal} = 220 \times 7.50 \times 0.85 = 1402.5W \approx 1.40 \text{ kW}$
2. Sesudah, $P_{akhir} = 220 \times 5.10 \times 0.85 = 953.7W \approx 0.95 \text{ kW}$
3. Penghematan, $1402.5 - 953.7 = 448.8 \text{ Watt} \approx 449 \text{ Watt}$

Selanjutnya, efektivitas pengendalian penggunaan daya dapat dievaluasi berdasarkan selisih dan persentase penghematan antara kondisi acuan dengan kondisi setelah penerapan sistem. Persentase penghematan daya dihitung menggunakan persamaan (2.12), sehingga didapatkan hasil persentase penghematan seperti dibawah ini:

1. Persentase Penghematan, $\frac{448.8}{1402.5} \times 100\% = 32.0 \%$
2. Jika Sistem Beroperasi 12 Jam:
 - Per-Hari, $0.448.8 \times 12 = 5.39 \text{ kWh/hari}$
 - Per-Bulan, $5.39 \times 30 = 161.7 \text{ kWh/bulan}$

BAB 5

PENUTUP

Bab ini berisi tentang kesimpulan dan juga saran dari Tugas Akhir yang telah dikerjakan. Berisi mengenai permasalahan yang terjadi saat proses pengerjaan Tugas Akhir dan analisa-analisa yang didapatkan selama pengerjaan Tugas Akhir sampai selesai.

5.1 Kesimpulan

Berdasarkan percobaan dan pengujian yang telah dilakukan pada Tugas Akhir yang berjudul “Implementasi Pengendalian *Electrical Facility* dan *Illegal Activity Detection* di Masjid Bahrul Ulum Menggunakan Metode YOLO” ini dapat disimpulkan sebagai berikut:

1. Penelitian ini berhasil melakukan pengelolaan dataset melalui pengumpulan citra dari berbagai kondisi nyata di area Masjid Bahrul Ulum, PPNS. Kemudian dilakukan pelabelan menggunakan *bounding box* pada tiga kelas, yaitu jamaah, *indication of a person littering*, dan *illegal overnight stay*. Dataset yang terkumpul sebanyak 8.948 citra selanjutnya melalui tahap *preprocessing* berupa *resize*, normalisasi, serta augmentasi seperti rotasi, *flipping*, dan perubahan pencahayaan. Dataset kemudian dibagi menjadi 8.040 citra *training* atau 90%, 613 citra *validation* atau 7%, dan 295 citra *test* atau 3%. Pembagian tersebut memberikan porsi pelatihan yang cukup sekaligus menyediakan data validasi dan pengujian terpisah, sehingga model YOLOv8m dapat dilatih dan dievaluasi secara terstruktur sebelum diimplementasikan pada sistem.
2. Model YOLOv8m yang digunakan untuk mendeteksi keberadaan jamaah, *indication of a person littering*, dan *illegal overnight stay* menunjukkan performa yang cukup baik sebagai dasar pengambilan keputusan sistem. Proses *training* dilakukan menggunakan Kaggle Notebook dengan GPU NVIDIA Tesla T4 dan menghasilkan model terbaik pada epoch ke-64, sedangkan proses pelatihan berhenti secara otomatis pada epoch ke-109 dari 220 epoch melalui mekanisme *Early Stopping*. Berdasarkan hasil evaluasi, model memperoleh *Precision* sekitar 0,85, *Recall* 0,81–0,82, *mAP@0.5* sekitar 0,83–0,84, dan *mAP@0.5:0.95* sekitar 0,51–0,52. Pada *Normalized Confusion Matrix*, tingkat

prediksi benar mencapai 72% untuk jamaah, 87% untuk membuang sampah, dan 92% untuk tidur. Pengujian citra juga menghasilkan rata-rata *confidence* 0,80 pada jamaah, 0,85 pada tidur, dan 0,79 pada membuang sampah, sehingga model dapat digunakan pada proses deteksi dan pengendalian otomatis.

3. Implementasi sistem terbukti dapat membantu pengelolaan fasilitas listrik dan pengawasan aktivitas secara otomatis dengan mengintegrasikan CCTV, YOLOv8m, ESP32, LDR, RTC, *relay*, MQTT HiveMQ, Supabase, Cloudflare Tunnel, dan *web monitoring*. Hasil pengujian komunikasi menunjukkan rata-rata waktu respons sistem berada pada rentang 1,6–7 detik, sehingga kestabilan jaringan Wi-Fi masih sangat memengaruhi kecepatan pembaruan data dan respons kontrol. Dari sisi efisiensi energi, rata-rata arus fasa R menurun dari 7,50 A menjadi 5,10 A, sedangkan daya berkurang dari sekitar 1,40 kW menjadi 0,95 kW. Selisih tersebut menghasilkan penghematan sekitar 448,8 Watt atau 32%, sehingga pengendalian lampu dan kipas berdasarkan kondisi aktual ruangan mampu mengurangi penggunaan daya dibandingkan sistem konvensional.

5.2 Saran

Dari hasil penelitian yang telah dilakukan, terdapat beberapa hal yang perlu diperhatikan dan ditingkatkan pada penelitian selanjutnya untuk mencapai hasil yang maksimal, yaitu :

1. Peningkatan dataset dan model deteksi pada 3 class (Jamaah, aktivitas ilegal membuang sampah, dan aktivitas ilegal menginap di Masjid) pada konsistensi dan peningkatan nilai *confidence* , dataset perlu diperluas dengan variasi kondisi pencahayaan, posisi kamera, sudut pengambilan gambar, jumlah objek, serta aktivitas jamaah yang lebih beragam. Selain itu, disarankan untuk menggunakan metode labelling khusus (*Pose Estimation*) sebagai bantuan keakuratan model class agar tidak saling terkait dengan class yang lainnya.
2. Tampilan web monitoring masih dapat dikembangkan dengan menambahkan fitur notifikasi otomatis melalui Telegram, atau email ketika sistem mendeteksi aktivitas ilegal, kondisi perangkat mengalami gangguan, maupun ketika koneksi MQTT terputus. Selain itu, dashboard dapat dilengkapi dengan grafik

histori deteksi objek, serta *line chart* berapa kali penggunaan fasilitas secara periodik.

3. Diharapkan sekaligus menjadi pesan dari pengurus Takmir Masjid Bahrul Ulum PPNS agar kedepannya bisa dikembangkan menjadi lebih banyak kipas dan lampu yang bisa di kontrol (terutama seluruh fasilitas di lantai 2) agar dapat memaksimalkan penghematan dan efisiensi dalam penggunaan listrik Masjid.
4. Untuk meningkatkan kestabilan komunikasi data, disarankan menggunakan jaringan internet yang lebih baik atau menyediakan akses point khusus bagi sistem IoT sehingga proses komunikasi antara mini PC, ESP32, web monitoring, dan HiveMQ Cloud tidak terpengaruh oleh kepadatan pengguna jaringan Wi-Fi masjid.

(halaman ini sengaja dikosongkan)

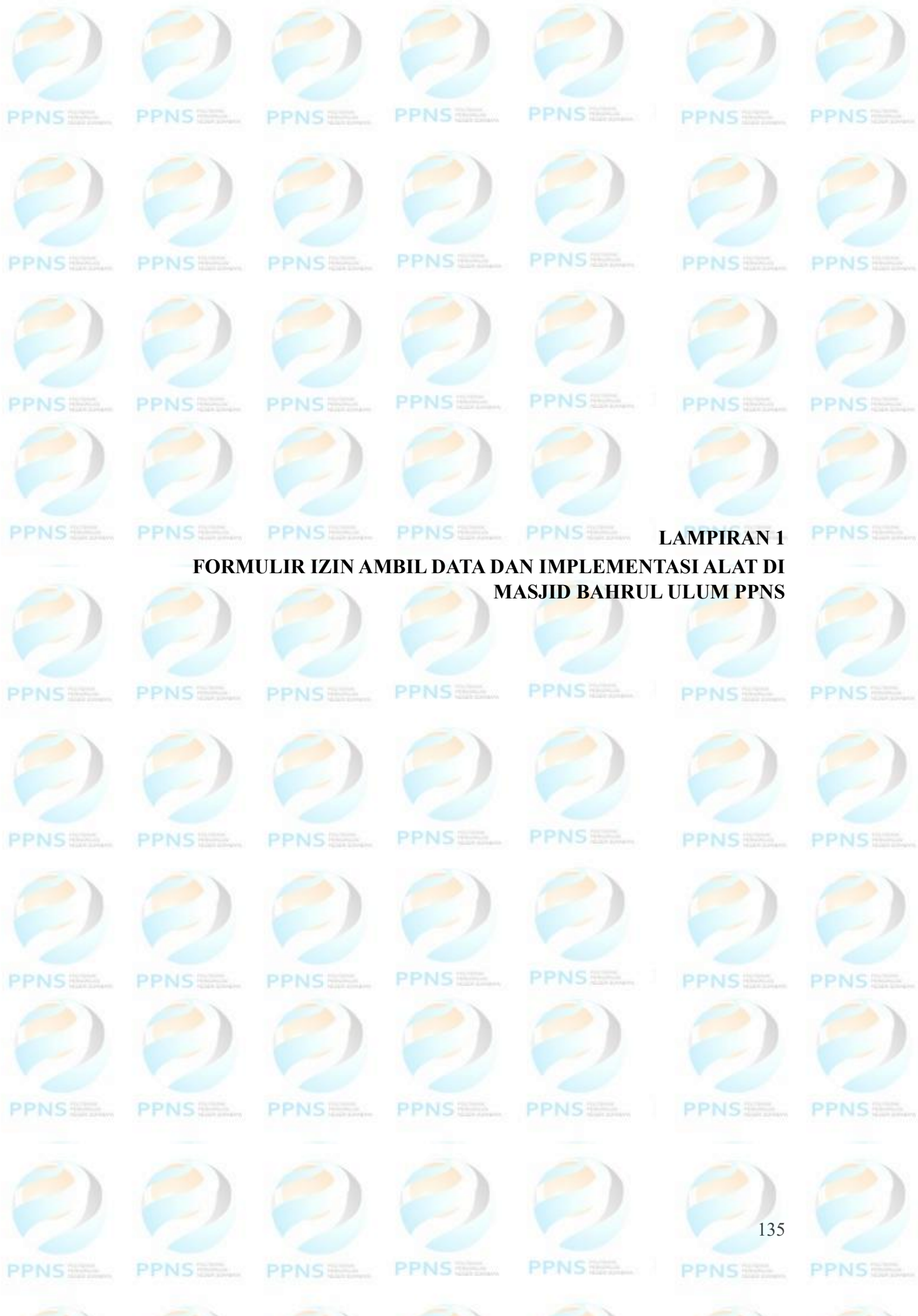
DAFTAR PUSTAKA

- Alzahrani, N., Bchir, O., & Ismail, M. M. B. (2025). YOLO-Act: Unified spatiotemporal detection of human actions across multi-frame sequences. *Sensors*, 25(10), 3013. <https://doi.org/10.3390/s25103013>
- Ardiansyah, M. N., Kurniasari, M., Amien, M. D., Wijaya, D., & Setialana, P. (2022). Development of crowd detection warning system based on deep convolutional neural network using CCTV. *Journal of Engineering and Applied Technology*, 3(1), 35–41. <https://doi.org/10.21831/jeatech.v3i1.43771>
- Challa, K., AlHmoud, I. W., Jaiswal, C., Gokaraju, B., & Tesireo, R. (2025). Gas sensors and real-time video for accurate classroom occupancy detection. *MethodsX*, 14, 103386. <https://doi.org/10.1016/j.mex.2025.103386>
- Chaudhari, P., Xiao, Y., Cheng, M. M.-C., & Li, T. (2024). Fundamentals, algorithms, and technologies of occupancy detection for smart buildings using IoT sensors. *Sensors*, 24(7), 2123. <https://doi.org/10.3390/s24072123>
- Chen, S., Liu, Y., Zhang, H., & Cai, Y. (2025). A human location and action recognition method based on improved YOLOv11 model. *Discover Artificial Intelligence*, 5, 232. <https://doi.org/10.1007/s44163-025-00492-6>
- Gawande, U., Hajari, K., & Golhar, Y. (2024). Novel person detection and suspicious activity recognition using enhanced YOLOv5 and motion feature map. *Artificial Intelligence Review*, 57, 16. <https://doi.org/10.1007/s10462-023-10630-0>
- Gaya-Morey, F. X., Manresa-Yee, C., & Buades-Rubio, J. M. (2024). Deep learning for computer vision based activity recognition and fall detection of the elderly: A systematic review. *Applied Intelligence*, 54, 8982–9007. <https://doi.org/10.1007/s10489-024-05645-1>
- Han, L., Ma, X., Dai, M., & Bai, L. (2025). A WAD-YOLOv8-based method for classroom student behavior detection. *Scientific Reports*, 15, 9655. <https://doi.org/10.1038/s41598-025-87661-w>
- Hermens, F. (2024). Automatic object detection for behavioural research using YOLOv8. *Behavior Research Methods*, 56(7), 7307–7330. <https://doi.org/10.3758/s13428-024-02420-5>
- Huang, K., & Abisado, M. B. (2025). Lightweight construction safety behavior detection model based on improved YOLOv8. *Discover Applied Sciences*, 7, 326. <https://doi.org/10.1007/s42452-025-06766-z>
- Khan, I., Zedadra, O., Guerrieri, A., & Spezzano, G. (2024). Occupancy prediction in IoT-enabled smart buildings: Technologies, methods, and future directions. *Sensors*, 24(11), 3276. <https://doi.org/10.3390/s24113276>

- Khakan, A. R., Aghdasi, H. S., & Salehpour, P. (2025). Fast and high-precision human fall detection using improved YOLOv8 model. *IEEE Access*, 13, 5271–5283. <https://doi.org/10.1109/ACCESS.2024.3470319>
- Nasir, R., Jalil, Z., Nasir, M., Alsubait, T., Ashraf, M., & Saleem, S. (2025). An enhanced framework for real-time dense crowd abnormal behavior detection using YOLOv8. *Artificial Intelligence Review*, 58, 202. <https://doi.org/10.1007/s10462-025-11206-w>
- Ouardirhi, Z., Mahmoudi, S. A., & Zbakh, M. (2024). Enhancing object detection in smart video surveillance: A survey of occlusion-handling approaches. *Electronics*, 13(3), 541. <https://doi.org/10.3390/electronics13030541>
- Parambil, M. M. A., Ali, L., Swavaf, M., Bouktif, S., Gochoo, M., Aljassmi, H., & Alnajjar, F. (2024). Navigating the YOLO landscape: A comparative study of object detection models for emotion recognition. *IEEE Access*, 12, 109427–109442. <https://doi.org/10.1109/ACCESS.2024.3439346>
- Park, S., Kim, J., Wang, S., & Kim, J. (2025). Effectiveness of image augmentation techniques on non-protective personal equipment detection using YOLOv8. *Applied Sciences*, 15(5), 2631. <https://doi.org/10.3390/app15052631>
- Priadana, A., Nguyen, D.-L., Vo, X.-T., Choi, J., Ashraf, R., & Jo, K. (2025). HFD-YOLO: Improved YOLO network using efficient attention modules for real-time one-stage human fall detection. *IEEE Access*, 13, 41248–41258. <https://doi.org/10.1109/ACCESS.2025.3547360>
- Sayed, A. N., Mahmud, S., Bensaali, F., Chowdhury, M. E. H., & Himeur, Y. (2025). Optimizing energy efficiency through precise occupancy detection: A tailored CNN architecture for smart buildings and beyond. *Neural Computing and Applications*, 37, 16487–16503. <https://doi.org/10.1007/s00521-025-11348-6>
- Sedaghati, N., Ardebili, S., & Ghaffari, A. (2025). Application of human activity/action recognition: A review. *Multimedia Tools and Applications*, 84, 33475–33504. <https://doi.org/10.1007/s11042-024-20576-2>
- Song, W., & Calautit, J. (2025). Impact of occupancy behavior on building energy efficiency: What's next in detection and monitoring technologies? *Next Energy*, 8, 100350. <https://doi.org/10.1016/j.nxener.2025.100350>
- Sun, K., Wang, X., Xing, T., Liu, S., & Zhao, Q. (2024). High-accuracy occupancy counting at crowded entrances for smart buildings. *Energy and Buildings*, 319, 114509. <https://doi.org/10.1016/j.enbuild.2024.114509>
- Utomo, I. T., Firdaus, M. N., Adinandra, S., & Murnani, S. (2024). An intelligent system for light and air conditioner control using YOLOv8. *Journal of Innovation Information Technology and Application (JINITA)*, 6(2), 141–152. <https://doi.org/10.35970/jinita.v6i2.2446>
- Wang, J., Jiang, L., Yu, H., Feng, Z., Castaño-Rosa, R., & Cao, S.-J. (2024). Computer vision to advance the sensing and control of built environment towards occupant-centric sustainable development: A critical review.

- Renewable and Sustainable Energy Reviews*, 192, 114165. <https://doi.org/10.1016/j.rser.2023.114165>
- Wijayanti, K., Mutiara, G. A., Suryawardani, B., Ervina, E., & Kusuma, G. P. (2025). Non-intrusive real-time tourist crowd monitoring for overtourism mitigation using YOLOv8-based head detection and tracking. *Journal of Robotics and Control (JRC)*, 6(4), 1985–2004. <https://doi.org/10.18196/jrc.v6i4.26396>
- Wu, Y., Chen, S., Jin, Y., Xu, H., Zhou, X., Wang, X., Chong, A., Li, J., & Yan, D. (2024). Novel occupancy detection method based on convolutional neural network model using PIR sensor and smart meter data. *Advanced Engineering Informatics*, 62, 102589. <https://doi.org/10.1016/j.aei.2024.102589>
- Xu, Y., & Fu, Y. (2024). Complex indoor human detection with You Only Look Once: An improved network designed for human detection in complex indoor scenes. *Applied Sciences*, 14(22), 10713. <https://doi.org/10.3390/app142210713>
- Zhang, H., Liu, L., Bi, J., Liu, H., & Wen, Z. (2026). Modified lightweight YOLOv8 model for fast and precise indoor occupancy detection. *Applied Sciences*, 16(1), 335. <https://doi.org/10.3390/app16010335>
- Zhang, W., Calautit, J., Tien, P. W., Wu, Y., & Wei, S. (2024). Deep learning models for vision-based occupancy detection in high occupancy buildings. *Journal of Building Engineering*, 98, 111355. <https://doi.org/10.1016/j.jobe.2024.111355>
- Zhang, W., Li, J., Tien, P. W., & Calautit, J. K. S. (2025). Vision-based occupancy detection in indoor environments: A comparison of standard RGB and thermal cameras. *Journal of Building Engineering*, 114, 114215. <https://doi.org/10.1016/j.jobe.2025.114215>
- Zhang, Y., Liang, W., Yuan, X., Zhang, S., Yang, G., & Zeng, Z. (2024). Deep learning-based abnormal behavior detection for elderly healthcare using consumer network cameras. *IEEE Transactions on Consumer Electronics*, 70(1), 2414–2422. <https://doi.org/10.1109/TCE.2023.3309852>
- Zhou, S., Zhang, W., Tien, P. W., & Calautit, J. K. S. (2025). Evaluating single-shot and two-stage vision-based detection of occupancy activity in energy-efficient buildings. *Energy and Buildings*, 345, 116017. <https://doi.org/10.1016/j.enbuild.2025.116017>
- Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3), 257–276. <https://doi.org/10.1109/JPROC.2023.3238524>

(halaman ini sengaja dikosongkan)



LAMPIRAN 1
FORMULIR IZIN AMBIL DATA DAN IMPLEMENTASI ALAT DI
MASJID BAHRUL ULUM PPNS

(halaman ini sengaja dikosongkan)

Kepada Yth.

Bapak Ketua Takmir

Masjid Bahrul Ulum PPNS

Dengan hormat,

Saya yang bertanda tangan di bawah ini merupakan mahasiswa semester 8 D4-Teknik Otomasi, yang sedang melaksanakan Tugas Akhir dengan judul "Implementasi Pengendalian *Electrical Facility* dan *Illegal Activity Detection* di Masjid Bahrul Ulum Menggunakan Metode YOLO". Sehubungan dengan hal tersebut, saya bermaksud mengajukan permohonan izin untuk melakukan pemasangan serta pengujian perangkat sistem berbasis *IoT* di area Masjid Bahrul Ulum PPNS.

Adapun kegiatan ini bertujuan untuk pengembangan sistem monitoring dan kontrol otomatis guna mendukung efisiensi penggunaan fasilitas masjid. Seluruh proses pemasangan dan pengujian akan dilakukan dengan tetap menjaga ketertiban, keamanan, serta tidak mengganggu aktivitas ibadah.

Demikian permohonan ini saya sampaikan. Besar harapan saya untuk mendapatkan izin dan dukungan dari Bapak. Atas perhatian dan kerja samanya, saya ucapkan terima kasih.

Ketua Takmir Masjid Bahrul Ulum PPNS,



Farizi Rachman, S.SI., M.SI.
NIP. 199003292015041002

Pengusul,



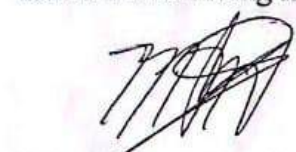
Vemas Ardiyan Syah
NRP. 0922040025

Dosen Pembimbing I,



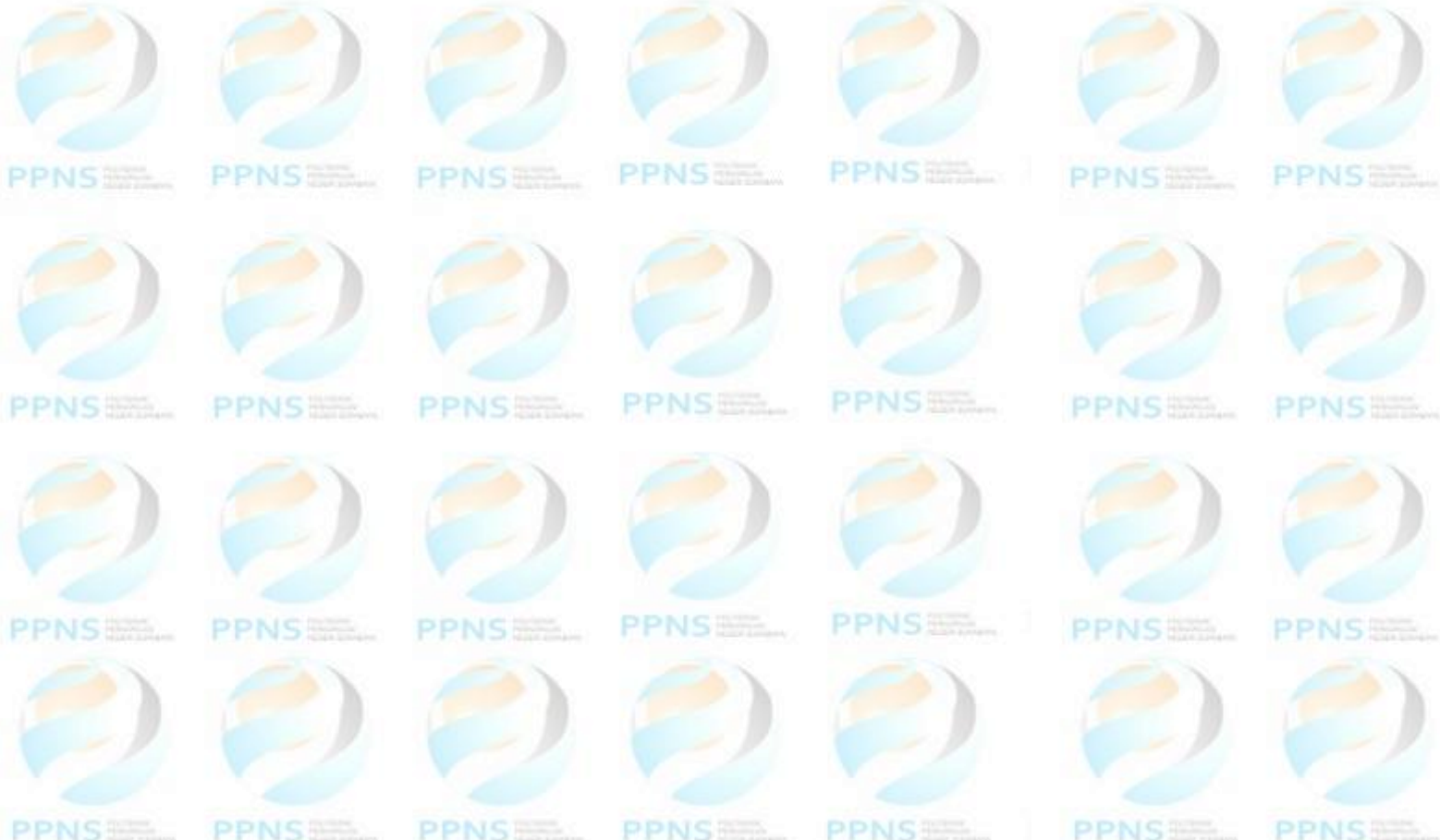
Noorman Rinanto, S.T., M.T., Ph.D.
NIP. 197610142012121002

Dosen Pembimbing II,

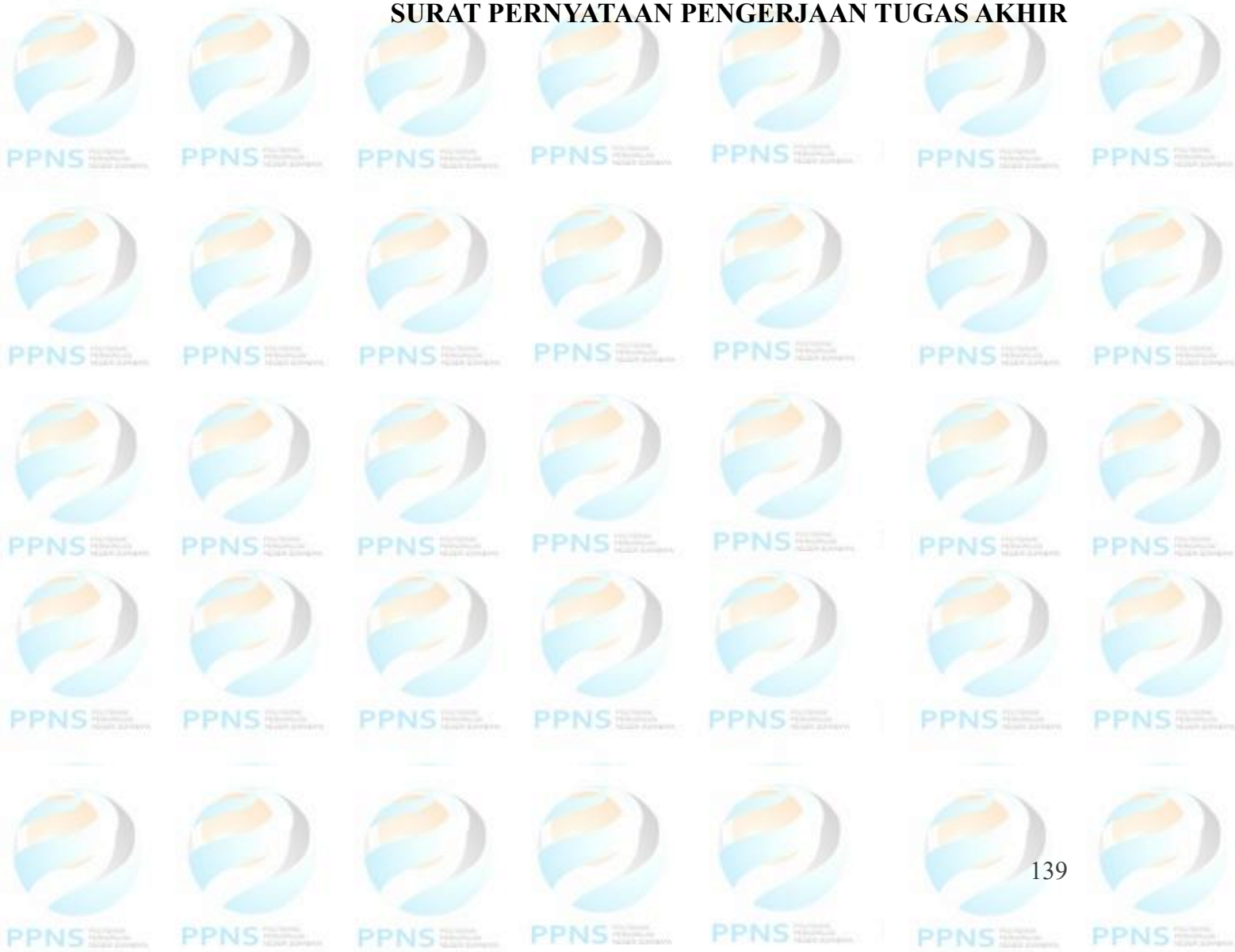


M. Lukman Arif, S.Pd.I., M.Pd.I
NIP. 199009102019031021

(halaman ini sengaja dikosongkan)



LAMPIRAN 2
SURAT PERNYATAAN Pengerjaan TUGAS AKHIR



(halaman ini sengaja dikosongkan)

	FORMULIR PERMOHONAN Pengerjaan TA	No. : F.WDII.09 Date : 1 Februari 2017 Rev. : 02 Page : 2 dari 2
---	--	---

SURAT PERNYATAAN Pengerjaan Tugas Akhir

Saya yang bertanda tangan di bawah ini :

Nama : Vemas Ardiyan Syah
 NRP. : 0922040025
 Jurusan : Teknik Kelistrikan Kapal/ D4-Teknik Otomasi

Dengan ini saya mengajukan permohonan izin untuk melaksanakan pengerjaan TA di Masjid Bahrul Ulum Politeknik Perkapalan Negeri Surabaya (PPNS) dalam rangka mengerjakan dan menyelesaikan Tugas Akhir. Permohonan ini diajukan untuk menunjang kegiatan penelitian, pemrograman, implementasi alat, serta penyusunan laporan Tugas Akhir yang memerlukan waktu dan konsentrasi yang optimal.

Adapun jadwal kegiatan dan masa izin menginap yang saya ajukan adalah sebagai berikut:

Hari	Tanggal	Waktu
Senin, Rabu, Jumat	06,08,10	Pukul 19:00 Sampai pukul 06:00 WIB
Senin, Rabu, Jumat,	13,15,17	Pukul 19:00 Sampai pukul 06:00 WIB
Senin, Rabu, Jumat,	20,22,24	Pukul 19:00 Sampai pukul 06:00 WIB
Senin, Rabu, Jumat,	27,29,31	Pukul 19:00 Sampai pukul 06:00 WIB

Saya menyatakan bersedia mematuhi seluruh peraturan dan tata tertib yang berlaku di lingkungan Masjid PPNS selama masa izin menginap. Saya juga berkomitmen untuk menjaga kebersihan, ketertiban, keamanan, serta kenyamanan lingkungan masjid, dan bertanggung jawab atas penggunaan fasilitas yang saya gunakan. Setelah masa izin berakhir, saya akan meninggalkan area yang digunakan dalam keadaan bersih, rapi, dan sebagaimana kondisi semula.

Mengetahui,
Dosen Pembimbing,

Surabaya, 06 Juli 2026

Takmir MBU PPNS,

Peminjam,


 (Noorman Rinanto, S.T., M.T., Ph.D.)
 NIP. 197610142012121002


 (M. Lukman Arif, S.Pd.I., M.Pd.I.)
 NIP. 199009102019031021


 (Vemas Ardiyan Syah)
 NRP. 0922040025

Acc

 Tofu
 Bag. Umur

(halaman ini sengaja dikosongkan)



BIODATA MAHASISWA



(halaman ini sengaja dikosongkan)

BIODATA MAHASISWA

1. Nama : Vemas Ardiyan Syah
2. NRP : 0922040025
3. Program Studi : Teknik Otomasi
4. Agama : Islam
5. Status : Belum Menikah
6. Nomor Telepon : 085746044128
7. Jenis Kelamin : Laki-Laki
8. Pekerjaan : Mahasiswa
9. Tempat, Tanggal Lahir : Gresik, 10 September 2003
10. Nama Orang Tua : Syaroni
11. Alamat Asal : Jl. Dr. Wahidin S.H. GG 26, No.44A, Kebomas, Gresik
12. Alamat Orang Tua : Jl. Dr. Wahidin S.H. GG 26, No.44A, Kebomas, Gresik
13. Telepon Orang Tua : 085731753119
14. Email : vemasardiyan@student.ppns.ac.id



PENDIDIKAN FORMAL			
Jenjang Pendidikan	Tahun	Tempat Pendidikan	Jurusan
Diploma IV	2022 - Sekarang	Politeknik Perkapalan Negeri Surabaya	Teknik Otomasi
SMK	2019- 2022	SMK PGRI 1 Gresik	Teknik Kimia Industri
SMP	2016 - 2019	UPT SMP Negeri 22 Gresik	-
SD	2010 - 2016	UPT SD Negeri 29 Gresik	-

(halaman ini sengaja dikosongkan)