

**SISTEM PEMBERSIH *SOLAR PANEL* BERBASIS
MICROCONTROLLER DAN *IMAGE PROCESSING*
PROYEK AKHIR**

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung



Disusun Oleh :

Muhammad Ihsan Solih

NIM: 1052250

Syair Aulia

NIM: 1052258

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2025**

LEMBAR PENGESAHAN

**SISTEM PEMBERSIH *SOLAR PANEL* BERBASIS *MICROCONTROLLER*
DAN *IMAGE PROCESSING***

Oleh:

Muhammad Ihsan Solih/1052250

Syair Aulia/1052258

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan
Program Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



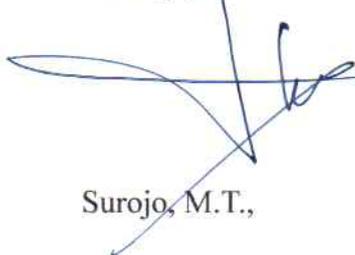
Zanu Saputra, M.Tr.T.

Pembimbing 2



Lesta, S.P., M.Si.

Penguji 1



Surojo, M.T.,

Penguji 2



Ni Luh Eta Yuspita,
S.Kel.,M.Si

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa 1 : Muhammad Ihsan Solih NIM: 1052250

Nama Mahasiswa 2 : Syair Aulia NIM: 1052258

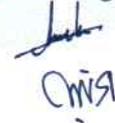
Dengan Judul : Sistem Pembersih *Solar Panel* Berbasis
Mikrokontroller dan Image Processing

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 04 Juli 2025

Nama Mahasiswa
1. Muhammad Ihsan Solih
2. Syair Aulia

Tanda Tangan



ABSTRAK

Pada era modern ini, energi terbarukan merupakan solusi utama dalam mengatasi perubahan iklim dan bertambahnya permintaan energi global. Salah satu sumber energi terbarukan yang paling menjanjikan adalah energi matahari (energi surya), salah satunya adalah pemanfaatan solar panel. Pada penelitian ini bertujuan untuk membuat suatu sistem pembersih solar panel berbasis microcontroller dan image processing serta membuat sistem control berdasarkan tingkat kekotoran solar panel. Pada sistem yang dirancang menggunakan mikrokontroler berupa ESP32 yang berfungsi menjadi pusat kontrol pembersihan, RTC sebagai sensor waktu nyata untuk penjadwalan pembersihan serta Raspberry Pi sebagai kontrol pada image processing untuk pendeteksian tingkat kekotoran melalui sensor kamera. Hasil penelitian ini berupa prototype pembersihan solar panel secara terjadwal dan berdasarkan tingkat kekotoran solar panel. Kesimpulan dari penelitian ini menunjukkan bahwa pembersihan solar panel sangat diperlukan demi menjaga efektivitas kinerja solar panel dalam menangkap energi dari sinar matahari. Teknologi ini masih memiliki potensi untuk dikembangkan secara luas agar dapat meminimalisir tenaga dan waktu dalam pembersihan solar panel terutama jika adanya dukungan penelitian lebih lanjut yang dapat lebih kompleks.

Kata kunci: ESP32; Raspberry Pi; Tingkat Kekotoran; Webcam; Yolo V5.

ABSTRACT

In the modern era, renewable energy is the primary solution for mitigating climate change and addressing the increasing global energy demand. One of the most promising renewable energy sources is solar energy, which includes the use of solar panels. This study aims to develop a solar panel cleaning system utilizing a microcontroller and image processing, as well as to create a control system based on the level of dirtiness of the solar panel. The system design uses a microcontroller in the form of ESP32, which functions as a cleaning control center, an RTC as a real-time sensor for scheduling cleaning, and a Raspberry Pi as a controller for image processing to detect the level of dirtiness through the camera sensor. The results of this study are in the form of a prototype of scheduled solar panel cleaning, based on the level of dirtiness of the solar panel. The conclusion of this study shows that cleaning solar panels is very necessary to maintain the effectiveness of solar panel performance in capturing energy from sunlight. This technology still has the potential to be developed widely in order to minimize energy and time in cleaning solar panels, especially if there is support for further research that can be more complex.

Keywords: ESP32; Raspberry Pi; Dirt Level; Webcam; Yolo V5.

KATA PENGANTAR

Assalamu'alaikum warahmatullahi wabarakatuh.

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas rahmatnya dan karunia-Nya sehingga penulis dapat menyelesaikan penyusunan Laporan Proyek Akhir ini dengan baik dan tepat waktu. Adapun tujuan disusunnya Laporan Proyek Akhir ini adalah sebagai salah satu syarat dan kewajiban akademik mahasiswa untuk menyelesaikan kurikulum program Pendidikan Sarjana Terapan di Politeknik Manufaktur Negeri Bangka Belitung.

Dalam proses penyusunan laporan ini, penulis mencoba untuk menerapkan ilmu pengetahuan serta keterampilan yang telah dipelajari selama mengampu pendidikan di Politeknik Manufaktur Negeri Bangka Belitung pada pembuatan alat dan Laporan Proyek Akhir ini. Melalui laporan ini, penulis berharap hasil dari pembelajaran penulis selama ini dapat diimplementasikan langsung sesuai dengan bidang keahlian.

Tersusunnya Laporan Proyek Akhir ini tentu bukan karena buah kerja penyusun semata, melainkan juga atas bantuan, bimbingan, dan dukungan dari berbagai pihak. Untuk itu, dengan kerendahan hati, penulis ucapkan terima kasih sebesar-besarnya kepada semua pihak yang membantu terselesaikannya Proyek Akhir ini, di antaranya:

1. Allah SWT yang telah menciptakan dan memberikan kehidupan didunia.
2. Orang tua serta keluarga besar yang senantiasa memberikan kasih sayang, doa, semangat dan dukungan baik moril maupun materil.
3. Bapak I Made Andik Setiawan, M.Eng., Ph.D. selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
4. Bapak Zanu Saputrra, M.Tr.T. selaku Kepala Jurusan Rekayasa Elektro dan Industri Pertanian Politeknik Manufaktur Negeri Bangka Belitung seekaligus pembimbing I yang telah banyak meluangkan waktu, tenaga, dan pikiran dalam

memberikan pengarahan selama pembuatan alat serta laporan Proyek Akhir ini hingga selesai.

5. Ibu Lesta, SP, M.Si. selaku pembimbing II yang telah memberikan saran dan solusi dari masalah-masalah yang dihadapi selama proses penyusunan laporan.
6. Bapak Aan Febriansyah, S.ST., MT. selaku Koor. Program Studi D-IV Teknik Elektronika Politeknik Manufaktur Negeri Bangka Belitung.
7. Seluruh dosen dan staf pengajar di Politeknik Manufaktur Negeri Bangka Belitung atas ilmu yang sangat bermanfaat bagi penulis.
8. Rekan-rekan mahasiswa Politeknik Manufaktur Negeri Bangka Belitung terkhusus angkatan 29 yang telah banyak membantu selama proses pembuatan Proyek Akhir ini.
9. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung dalam pembuatan Proyek Akhir ini yang tidak dapat disebutkan satu per satu.

Semoga Allah SWT senantiasa membalas kebaikan dan mencurahkan hidayah serta taufik-Nya kepada kita semua aamiin. Penulis juga menyadari bahwa Laporan Proyek Akhir ini masih jauh dari sempurna dikarenakan penulis adalah manusia biasa yang tidak luput dari kesalahan. Oleh karena itu, sangat diharapkan segala petunjuk, kritik dan saran yang membangun dari pembaca agar dapat menunjang pengembangan dan perbaikan penulisan selanjutnya. Penyusun berharap semoga laporan ini dapat bermanfaat untuk kita semua. Terima kasih.

Wassalamu 'alaikum warahmatullahi wabarakatuh.

Sungailiat, 04 Juli 2025

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	ii
PERNYATAAN BUKAN PLAGIAT	iii
ABSTRAK	iv
ABSTRACT.....	v
KATA PENGANTAR.....	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiv
BAB I	1
PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Tujuan Proyek Akhir	3
1.4. Batasan Masalah.....	3
BAB II.....	4
LANDASAN TEORI	4
2.1. <i>Solar Panel</i>	4
2.2. <i>Raspberry Pi 4 Model B</i>	5
2.3. Webcam.....	6
2.4. <i>Image Processing</i>	7
2.5. <i>YOLO (You Only Look Once)</i>	9
2.6. <i>Python</i>	10
2.7. <i>Open CV</i>	11
2.8. <i>MQTT (Message Queue Telemetry Transport)</i>	13
2.9. Klasifikasi Tingkat Kekotoran <i>Solar Panel</i>	13
2.10. Sistem Pembersih Otomatis	14

BAB III	16
METODE PELAKSANAAN.....	16
3.1. Diagram Alir.....	16
3.2. Studi Literatur	16
3.3. Perancangan Sistem	18
3.3.1 Blok Diagram	18
3.3.2 Desain Alat	19
3.4. Perancangan Elektrikal.....	19
3.5. Pengujian Komponen	20
3.5.1. Pengujian RTC 3231	21
3.5.2. Pengujian <i>Waterpump</i>	22
3.5.3. Pengujian Motor Stepper NEMA 17HS3401	23
3.5.4. Pengujian Motor DC 555	24
3.6. Pembuatan Sistem	25
3.7. Pembuatan Dataset	26
3.7.1. Pelabelan Gambar	26
3.7.2. Anotasi Label Gambar	27
3.7.3. Resize Label Gambar	28
3.7.4. Augmentasi Label Gambar	28
3.7.5. Training Label Gambar	29
3.8. Pengujian Alat Keseluruhan	30
3.9. Pembuatan Laporan dan Publikasi Ilmiah.....	31
BAB IV	32
HASIL DAN PEMBAHASAN	32
4.1. Sistem Kerja Keseluruhan Alat	32
4.2. Uji Coba Sistem Pendeteksian	35
4.3. Uji Coba Integrasi Sistem Penjadwalan.....	37
4.4. Uji Coba Integrasi Sistem Otomatis.....	40
BAB V.....	44
KESIMPULAN DAN SARAN.....	44
5.1. Kesimpulan	44
5.2. Saran.....	44
DAFTAR PUSTAKA	46



DAFTAR TABEL

Tabel	Halaman
3.1 Hasil Pengujian RTC 3231	21
3.2 Hasil Pengujian <i>Waterpump</i>	23
3.3 Hasil Pengujian Motor Stepper NEMA 17HS3401	24
3.4 Hasil Pengujian motor DC 555	25
4.1 Hasil Uji Coba Sistem Pendeteksian.....	36
4.2 Hasil Uji Coba Integrasi Sistem Penjadwalan	37
4.3 Hasil Uji Coba Integrasi Sistem Otomatis	41



DAFTAR GAMBAR

Gambar	Halaman
2.1 Solar Panel	5
2.2 Raspberry Pi 4 Model B.....	6
2.3 Webcam C270	7
2.4 Colour Image.....	8
2.5 Black and White (greyscale)	8
2.6 Binary Image.....	8
2.7 Kecepatan GPU mengukur waktu inferensi rata-rata per gambar	10
2.8 Python	10
2.9 OpenCV	12
3.1 Diagram Alir Sistem Pembersih Solar Panel berbasis Microcontroller dan Image Processing.....	17
3.2 Blok Diagram Sistem Pembersih Solar Panel berbasis Microcontroller dan Image Processing.....	18
3.3 Desain Alat Sistem Pembersih Solar Panel berbasis Microcontroller dan Image Processing.....	19
3.4 Rancangan Elektrikal	20
3.5 Rangkaian Pengujian RTC 3231	21
3.6 Rangkaian Pengujian Waterpump	22
3.7 Rangkaian Pengujian Motor Stepper NEMA 17HS3401.....	23
3.8 Rangkaian Pengujian Motor DC 555	24
3.9 Tampilan RoboFlow.....	27
3.10 Anotasi label gambar.....	27
3.11 Resize label gambar	28
3.12 Augmentasi label gambar.....	29
3.13 Training label gambar	30
4.1 Sistem kerja alat	34

4.2 Perintah pada Windows PowerShell	35
4.3 Panel Bersih	38
4.4 Daun	38
4.5 Kotoran Burung.....	39
4.6 Panel Bersih	41
4.7 Daun	42
4.8 Kotoran Burung.....	42



DAFTAR LAMPIRAN

	Halaman
Lampiran 1: Daftar Riwayat Hidup.....	50
Lampiran 2: Kode Program.....	52



BAB I

PENDAHULUAN

1.1. Latar Belakang

Pada era modern ini, energi terbarukan menjadi solusi utama dalam mengatasi masalah perubahan iklim dan bertambahnya permintaan energi global. Salah satu sumber energi terbarukan yang paling menjanjikan adalah energi matahari (energi surya). Teknologi yang umum digunakan saat ini adalah *solar panel*. *Solar panel* adalah teknologi paling populer untuk menangkap energi matahari dan mengubahnya menjadi listrik. *Solar panel* tidak hanya ramah lingkungan, tetapi juga memiliki banyak keuntungan dikarenakan ketersediannya dalam jumlah besar di banyak wilayah.

Menurut data Kapasitas Terpasang PLN yang terpublikasi oleh BPS (Badan Pusat Statistik) pada tahun 2023, terhitung sebanyak 242 *megawatt (MW)* Pembangkit Listrik Tenaga Surya (PLTS) terpasang di Indonesia [1]. Data ini menunjukkan bahwa terdapat peningkatan PLTS yang relevan dibandingkan dengan tahun-tahun sebelumnya, disertai dengan penerapan Energi Baru Terbarukan (EBT) yang sedang dipercepat oleh pemerintah untuk mengurangi ketergantungan pada bahan bakar fosil yang semakin berkurang.

Penumpukan debu, kotoran dan kontaminasi menjadi permasalahan utama yang membuat menurunnya efektivitas kinerja *solar panel*, terutama di Indonesia yang dinobatkan sebagai negara dengan iklim tropis, dengan faktor lingkungan seperti polusi udara, curah hujan yang tidak merata, serta partikel debu yang terbawa angin, sehingga membuat intensitas cahaya matahari yang akan diterima oleh sel surya menurun dan mengurangi efektifitas kinerja sistemnya [2]. Oleh karena itu, perawatan dan pembersihan berkala secara rutin menjadi aspek yang sangat penting dalam pengelolaan PLTS, terkhusus di daerah beriklim tropis seperti di Indonesia. Namun, kebanyakan *solar panel* dibersihkan dengan cara manual sehingga memakan banyak waktu dan tenaga. Guna mempertahankan kinerja *solar panel*

yang optimal, diperlukan solusi inovatif yang dapat mengatasi permasalahan tersebut.

Pada penelitian serupa yang berjudul "Rancang Bangun Sistem Pembersih Otomatis Pada *Solar Panel* Menggunakan *Wiper* Berbasis Mikrokontroler" dimana dalam penelitian ini terdapat umpan balik (*feedback*) yang didapatkan dari sensor debu dengan membandingkan kadar debu saat ini dengan kadar debu referensi, sehingga sistem kendali ini dapat disebut juga dengan sistem kendali loop tertutup (*closed loop*) [3]. Akan tetapi, pada penelitian terbaru, penulis menggunakan terobosan baru yaitu *image processing* dengan mekanisme mikrokontroler serta adanya *feedback* setelah dibersihkan bahwa panel dalam kondisi sudah bersih. Adapun jurnal penelitian lain yang berjudul "rancang bangun sistem monitoring kebersihan lantai berbasis pengolahan citra menggunakan *Raspberry Pi* yang terintegrasi dengan robot pembersih lantai melalui komunikasi MQTT", pada penelitian tersebut menggunakan *image processing* untuk mendeteksi debu yang ada pada lantai, namun memiliki kekurangan dikarenakan *webcam* yang digunakan menangkap kualitas gambar yang kurang baik [4]. Dengan demikian, penulis mencoba mengimplementasikan *image processing* melalui pemanfaatan komunikasi MQTT dari *Raspberry PI 4* ke *ESP32* untuk mendeteksi debu di atas *panel cell* dengan meningkatkan *webcam* yang dapat menangkap kualitas gambar lebih baik.

Berdasarkan hal tersebut, maka dirancanglah sebuah sistem pembersih *solar panel* otomatis yang digunakan untuk membersihkan *solar panel* dengan sistem penjadwalan dan pengkondisian. Untuk itu, diajukannya judul proyek akhir "Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*". Mikrokontroler berfungsi untuk mengatur sistem penjadwalan melalui *RTC* serta *image processing* yang berfungsi untuk mengatur pengkondisian sesuai dengan warna *solar panel* yang ditangkap oleh sensor kamera.

1.2. Perumusan Masalah

Rumusan masalah yang dibuat dalam proyek akhir ini adalah:

1. Bagaimana merancang dan mengimplementasikan sistem pembersih *solar panel* secara otomatis yang dilengkapi dengan sistem pengkondisian?

2. Bagaimana membangun sistem kontrol pembersih *solar panel* dengan memanfaatkan teknologi mikrokontroler dan komunikasi nirkabel tanpa menurunkan performa *solar panel* tersebut?
3. Bagaimana penerapan metode *Image Processing* untuk pendeteksian tingkat kekotoran pada *solar cell* sebagai sinyal awal dalam aktivasi sistem pembersih otomatis?

1.3. Tujuan Proyek Akhir

Adapun tujuan yang ingin dicapai untuk pembuatan proyek akhir ini adalah:

1. Merancang dan mengimplementasikan sistem pembersih *solar panel* otomatis yang dilengkapi dengan sistem pengkondisian.
2. Membangun sistem kontrol untuk pembersihan *solar panel* dengan memanfaatkan teknologi mikrokontroler dan komunikasi nirkabel tanpa mengurangi performa *solar panel*.
3. Menerapkan metode *Image Processing* untuk pendeteksian tingkat kekotoran pada *solar cell* sebagai sinyal awal dalam aktivasi sistem pembersih otomatis.

1.4. Batasan Masalah

Agar pembahasan dalam proyek akhir ini tetap terfokus dan terarah pada tujuan utama, maka ditetapkan beberapa batasan-batasan masalah sebagai berikut:

1. Jalur gerak sistem ini terbatas, dalam artian hanya dapat bergerak mengikuti lintasan yang telah ditentukan, tanpa adanya kemampuan navigasi bebas pada bentuk panel.
2. Sistem dirancang untuk membersihkan ruang lingkup permukaan *solar panel*. Tidak termasuk pembersihan area sekitar panel seperti dudukan ataupun permukaan penyangga *solar panel*.
3. Perancangan dan pengujian sistem menggunakan satu jenis *solar panel* berukuran $1250 \times 808 \times 35\text{mm}$, sehingga hasil pengujian masih memerlukan pengembangan lanjutan sebelum diimplementasikan pada panel berukuran lain.

BAB II LANDASAN TEORI

2.1. Solar Panel

Panel surya, atau yang sering disebut *solar panel*, merupakan perangkat yang berfungsi untuk mengubah energi sinar matahari menjadi energi listrik melalui proses yang dikenal sebagai efek fotovoltaiik (*photovoltaic effect*). *Photovoltaic* merupakan sebuah proses dari energi matahari yang diubah menjadi energi listrik yang dipicu melalui hubungan kontak dari dua elektroda yang sudah terhubung ke dalam sistem ketika menerima pancaran sinar matahari [5].

Dikutip dari *website pranataprinting.com (2022)* *solar panel* juga sering disebut sebagai sistem energi terbarukan yang dinilai memiliki tingkat efisiensi yang tinggi serta lebih terjangkau karena ekonomis dan ramah lingkungan. Setiap bagian tertentu dari *solar panel* akan menghasilkan arus listrik ketika komponen utama terpancar sinar matahari, ini merupakan proses dari *solar panel* bekerja dengan efek fotolistrik. *Solar Panel* memiliki material kristal silicon dan setiap setengah dopin berfungsi sebagai dopan yang menghasilkan semikonduktor. Ketika sinar matahari menyinari solar cell, panel surya menyediakan energi yang sangat diperlukan untuk proses semikonduktor dalam menghasilkan arus DC. Dengan demikian, interaksi antara cahaya matahari dan solar cell menjadi kunci dalam menghasilkan energi listrik yang efisien.

Panel surya menerima intensitas cahaya matahari yang bervariasi, dipengaruhi oleh beberapa faktor, termasuk letak astronomi lokasi pemasangan, gerak semu harian dan tahunan matahari, serta kondisi cuaca. Dalam kondisi standar, sistem solar cell dapat mencapai efisiensi antara 10% hingga 30%. Pada tingkat intensitas matahari sebesar 1.000 W/m^2 dan suhu sekitar 30°C , panel ini mampu menghasilkan daya antara 100 hingga 300 Watt. Dengan demikian, efisiensi dan daya yang dihasilkan sangat bergantung pada faktor-faktor tersebut [6].



Gambar 2.1 *Solar Panel*

Pada proyek akhir, *solar panel* ditunjukkan pada gambar 2.1 yang akan digunakan sebagai objek utama dalam pengambilan sampel agar dapat mendukung proses dalam percobaan alat. Diharapkan hasil data yang didapatkan akurat dan relevan agar dapat menganalisis kinerja alat yang diuji.

2.2. *Raspberry Pi 4 Model B*

Raspberry Pi, yang sering disingkat sebagai Raspi, adalah sebuah komputer papan tunggal yang diciptakan oleh *Raspberry Pi Foundation*, sebuah yayasan nirlaba. Pengembangan perangkat ini melibatkan kolaborasi antara berbagai pengembang dan ahli komputer dari Universitas Cambridge di Inggris. Dengan demikian, *Raspberry Pi* merupakan hasil kerja sama yang menggabungkan inovasi teknologi dan pendidikan [7].

Adapun salah satu keunggulan utama dari *Raspberry Pi* adalah ketersediaan pin GPIO (*General Purpose Input Output*) yang sudah diprogram dengan baik, sehingga memungkinkan pengguna untuk mengendalikan berbagai modul sensor yang membutuhkan pin sebagai port serialnya [8].

Raspberry Pi memiliki kemampuan untuk mengendalikan berbagai komponen elektronik, yang disebabkan oleh adanya fitur pin *General Purpose Input Output (GPIO)*. Fitur ini memungkinkan komputer mini tersebut untuk menerima input dan mengirim output sesuai dengan instruksi yang telah diprogram. Dengan demikian, pengguna dapat memberikan perintah kepada *Raspberry Pi* secara langsung untuk mengatur komponen yang terhubung [9].



Gambar 2.2 Raspberry Pi 4 Model B

Dalam proyek akhir ini, *Raspberry Pi 4 Model B* yang ditunjukkan pada gambar 2.2 diatas digunakan sebagai pengontrolan kamera *webcam* untuk menangkap gambar permukaan *solar panel*. Hasil tangkapan gambar tersebut kemudian dianalisis untuk penentuan tingkat kekotoran pada permukaan panel. Berdasarkan hasil analisis, data akan dikirimkan ke *ESP32* untuk pengaktifan sistem pembersih secara otomatis sesuai dengan tingkat kekotoran yang telah dideteksi.

2.3. Webcam

Umumnya, webcam merupakan alat yang digunakan dalam pengambilan gambar maupun video melalui komputer, webcam sendiri merupakan sebutan untuk kamera yang mampu menampilkan gambar secara langsung dan pengaksesannya dapat melalui jaringan seperti *World Wide Web*, *program* pesan instan, atau aplikasi panggilan video [10] [11].

Webcam adalah kamera digital kecil yang dihubungkan ke komputer melalui port USB atau port COM pada komputer yang selama ini hanya digunakan dalam merekam atau menampilkan suatu objek secara langsung, namun, seiring dengan kemajuan teknologi saat ini, webcam telah mengalami perkembangan yang signifikan, webcam tidak hanya digunakan dalam kebutuhan komunikasi dan dokumentasi visual saja, akan tetapi banyak sekali digunakan dalam pemrosesan data berupa gambar [10] [11][12].

Seiring waktu, webcam mengalami perkembangan fitur yang semakin beragam, baik dari desain fisik hingga mekanisme dalam pergerakan lensa yang bisa dalam posisi tetap maupun dapat disesuaikan sudut pandangnya sesuai dengan [12].

Pada proyek akhir ini, penulis menggunakan webcam sebagai sensor kamera yang menangkap gambar kondisi permukaan *solar panel* seperti yang dapat dilihat pada gambar 2.3.



Gambar 2.3 Webcam C270

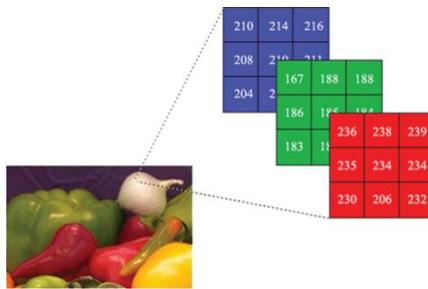
2.4. *Image Processing*

Pengolahan Citra Digital (*Digital Image Processing*) merupakan bidang ilmu yang mempelajari berbagai teknik untuk mengolah citra. Citra yang dimaksud mencakup foto serta gambar bergerak, seperti yang dihasilkan oleh webcam. Istilah "digital" merujuk pada proses pengolahan citra atau gambar yang dilakukan secara digital melalui komputer [13].

Image processing sendiri merupakan sebuah proses yang digunakan untuk meningkatkan resolusi citra digital dengan memperbaiki kualitas gambar seperti *blurring*, kontras rendah, maupun gangguan visual yang lain agar mudah dipahami oleh sistem komputer dan menampilkan informasi yang jelas kepada pengguna [14]. Terdapat tiga jenis citra digital, sebagai berikut:

1. *Colour Image atau RGB (Red, Green, Blue)*

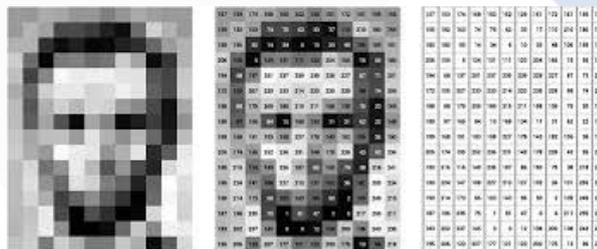
Sistem warna RGB (Red, Green, Blue) seperti yang ditampilkan pada gambar 2.4 mengatur setiap piksel dengan warna tertentu, yaitu merah, hijau, dan biru. Masing-masing warna memiliki rentang nilai antara 0 hingga 255, yang menghasilkan total kombinasi sebanyak 16.581.375 (atau 16 K). Oleh karena itu, citra berwarna terdiri dari tiga matriks yang masing-masing mewakili nilai-nilai untuk setiap warna. [13].



Gambar 2.4 Colour Image

2. Black and White Image

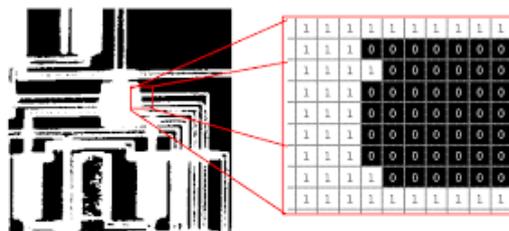
Gambar 2.5 menunjukkan gradasi warna yang dimiliki setiap pikselnya dimulai dari putih sampai hitam, sedangkan penggunaan 8bit atau 1bit dalam representasi warna memungkinkan rentang warna pada citra hitam-putih menjadi sangat sesuai untuk pengolahan file gambar. Biasanya digunakan dalam bidang kedokteran seperti X-ray [13].



Gambar 2.5 Black and White (greyscale)

3. Binary Image

Representasi citra hitam-putih yang menggunakan 1 bit per piksel (0 dan 1) atau 8 bit (0 hingga 255) sebagaimana dapat dilihat pada gambar 2.6 menawarkan efisiensi penyimpanan yang tinggi. Setiap piksel hanya terdiri dari dua warna, yaitu hitam dan putih, dan gambar direpresentasikan dalam bentuk biner. Oleh karena itu, metode ini sangat sesuai untuk aplikasi seperti teks (tulisan tangan), sidik jari (fingerprint), dan gambar arsitektur [13].



Gambar 2.6 Binary Image

Pada proyek akhir ini, penulis menggunakan *image processing* untuk pemrosesan gambar pada permukaan *solar panel*. Sistem yang dibuat diharapkan mampu mengenali tingkat kekotoran sehingga memungkinkan pengambilan keputusan pembersihan yang tepat. Secara otomatis sistem dapat mengenali pola kotoran dan pengoptimalan jadwal pembersihan sesuai kebutuhan.

2.5. YOLO (*You Only Look Once*)

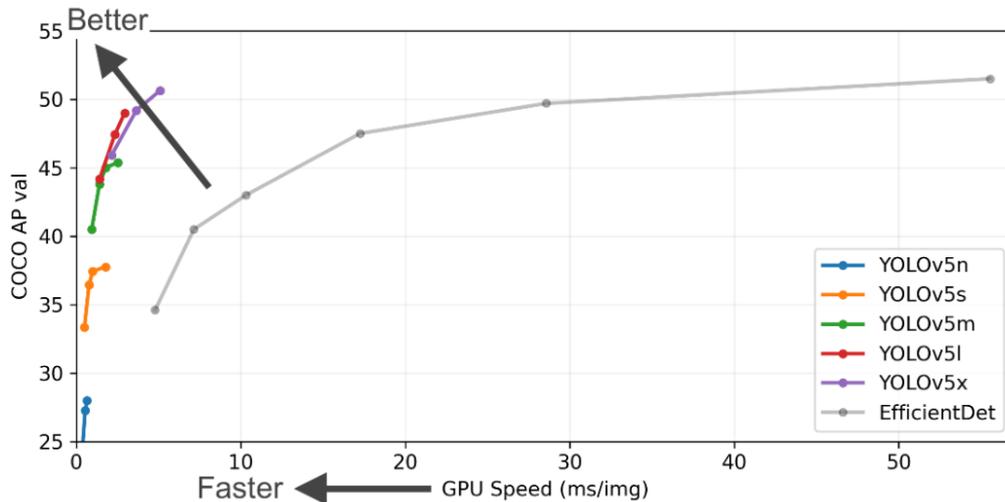
YOLO adalah sebuah jaringan saraf yang berfungsi untuk menganalisis citra dengan membagi citra tersebut menjadi beberapa daerah. Setiap daerah kemudian diprediksi dengan bounding box dan probabilitas yang terkait. Selanjutnya, probabilitas yang diperoleh untuk setiap bounding box digunakan untuk menentukan apakah objek tersebut termasuk dalam kategori objek atau bukan [15].

Algoritma *YOLO* merupakan algoritma *deep learning* yang digunakan untuk mendeteksi objek melalui pendekatan jaringan saraf tunggal yang diterapkan ke seluruh gambar dan akan memprediksi kelas probabilitas jika terdapat objek di setiap grid lalu kemudian dikalikan dengan nilai *bounding box* [16].

YOLO meringkaskan objek yang terdeteksi sebagai masalah regresi tunggal, dimana piksel citra langsung terhubung ke *bounding box* spasial yang terpisah dan probabilitas kelas yang terkait. *YOLO* menjalankan pendeteksian dan pengenalan objek menggunakan sebuah jaringan syaraf tunggal, yang memprediksi *bounding box* dan probabilitas kelas (*class*) secara langsung dalam satu evaluasi [17].

YOLO digunakan sebagai metode pengambilan objek pada permukaan *solar panel* untuk mengidentifikasi kotoran dan partikel lain pada *solar cell*. Kemampuan deteksi secara *real-time* diharapkan mampu dilakukan oleh sistem untuk mendeteksi serta menandai adanya indikasi debu dan kotoran secara cepat dan akurat. Sehingga dapat menentukan pengambilan keputusan dalam sistem pembersihan. Pada proyek akhir ini, penulis menggunakan *YOLO v5* yang memiliki akurasi lebih baik dari versi sebelumnya, didukung juga dengan sembilan *pre-trained model* yang dapat disesuaikan dengan *hardware* untuk mengoptimalkan proses input [18]. *YOLO v5* memiliki kemampuan untuk beroperasi pada CPU serta

GPU sehingga sangat cocok pada pengimplementasian deteksi objek secara *real-time* [19].



Gambar 2.7 Kecepatan GPU mengukur waktu inferensi rata-rata per gambar

Kecepatan rata-rata inferensi pada gambar 2.7 menunjukkan keunggulan *YOLO v5* sebagai *One Stage Detector* yang mampu mengenali objek secara langsung tanpa tahap awal sehingga menghasilkan sebuah prediksi yang lebih cepat serta menjadikannya solusi ideal untuk pengaplikasian deteksi *real-time* dengan presisi dan kecepatan yang seimbang [19].

2.6. Python



Gambar 2.8 Python

Python adalah bahasa pemrograman tingkat tinggi yang dirancang untuk mempermudah proses pembacaan dan penulisan kode program. Dengan sintaksis yang sederhana dan menyerupai bahasa manusia, *Python* memungkinkan pengembang menulis kode yang efisien dan mudah dipahami. Selain itu, sifatnya yang *open source* membuat *Python* dapat digunakan, dimodifikasi, dan

didistribusikan secara bebas, sehingga menjadi salah satu pilihan utama dalam pengembangan perangkat lunak di berbagai bidang [20]. Logo *Python* yang ditampilkan pada gambar 2.8 telah menjadi identitas visual yang telah dikenal luas di kalangan *developer*.

Python memiliki berbagai keunggulan yang menjadikannya populer di kalangan pengembang perangkat lunak. Salah satu kelebihanannya adalah desain sintaksis yang baik, sederhana, dan mudah dipahami, sehingga memudahkan dalam proses pengembangan program. Selain itu, *Python* dilengkapi dengan banyak pustaka (*library*) yang mendukung berbagai kebutuhan, mulai dari pengolahan data hingga pengembangan aplikasi berbasis web dan kecerdasan buatan. *Python* juga bersifat fleksibel karena dapat diintegrasikan dengan berbagai bahasa pemrograman lain seperti C, C++, Java, dan lainnya, sehingga memungkinkan pengembangan sistem yang kompleks dan lintas platform [21].

Dalam penulisan program *Python*, terdapat dua mode penulisan yang dapat digunakan, yaitu mode interaktif yang memungkinkan pengguna memberikan perintah kepada *Python* melalui *Python Shell* namun kelemahannya adalah tidak mendukung penyimpanan program untuk dijalankan berulang kali, adapun mode lain yaitu mode skrip yang menyediakan lembar kerja dengan nama *Python IDLE* untuk menulis, mengedit, menyimpan, serta menjalankan program [22].

Dalam proyek akhir, *Python* digunakan sebagai bahasa pemrograman utama. Pemilihan *Python* didasarkan pada kemampuan dalam kecerdasan buatan serta memiliki *library* (pustaka) yang luas sehingga mendukung kebutuhan dalam pengolahan citra digital nantinya. Selain itu, *Python* sangat mudah digunakan untuk mempercepat proses pemrograman.

2.7. *Open CV*

OpenCV (Open Computer Vision Library) merupakan pustaka sumber terbuka yang dikembangkan dengan menggunakan bahasa pemrograman C dan C++. Pustaka ini bersifat lintas platform, sehingga dapat dijalankan pada berbagai sistem operasi seperti Linux, Windows, dan macOS. *OpenCV* dirancang untuk mendukung proses komputasi yang efisien dengan performa tinggi, khususnya pada aplikasi

yang berjalan secara real-time. Dengan kemampuan tersebut, *OpenCV* sangat sesuai digunakan dalam pengembangan berbagai aplikasi di bidang visi komputer (*computer vision*) dan pembelajaran mesin (*machine learning*) [20].

Pada *library* ini terdapat 2.500+ algoritma yang telah dioptimalkan untuk mempercepat proses persepsi mesin dalam memahami dan memproses informasi visual. Beragam algoritma tersebut memungkinkan pengembangan berbagai fitur dan fungsi, antara lain deteksi serta pengenalan wajah, identifikasi objek, dan klasifikasi gerakan manusia dalam video. Selain itu, *OpenCV* juga mendukung pelacakan gerakan kamera dan objek yang bergerak, ekstraksi model objek tiga dimensi, serta pembuatan awan titik (*point cloud*) 3D menggunakan kamera stereo [23].

OpenCV (Open Computer Vision) adalah sebuah Antarmuka Pemrograman Aplikasi (API) yang dikenal luas dalam bidang pengolahan citra digital dan visi komputer. Visi komputer sendiri adalah cabang dari pengolahan citra yang bertujuan untuk memungkinkan komputer menginterpretasikan informasi visual dengan cara yang mirip dengan bagaimana manusia melihat. Melalui penerapan teknologi ini, komputer dapat melakukan pengenalan objek, membuat keputusan secara mandiri, dan mengambil tindakan berdasarkan analisis terhadap data visual yang diterimanya [24]. Logo resmi dari *OpenCV* dapat dilihat pada gambar 2.9 yang sering digunakan sebagai identitas visual dalam dokumentasi interface pengguna berbagai aplikasi berbasis *OpenCV*.



Gambar 2.9 *OpenCV*

OpenCV merupakan suatu pustaka *opensource vision* yang memiliki fungsi-fungsi untuk *image processing* berupa analisis gambar dan pola yang terus beriringan dikembangkan dengan kemajuan *PC (Personal Computer)* yang semakin meningkat [25].

Dalam proyek ini, penulis menggunakan *OpenCV* sebagai *library* utama untuk pemrograman pengolahan citra digital. Pemilihan *library* ini berdasarkan pada kemampuannya dalam menyediakan berbagai fungsi dalam pengolahan citra. *OpenCV* juga terintegrasi dengan bahasa pemrograman *Python* dan kompatibel dengan berbagai media sehingga memudahkan proses sistem pembersihan *solar panel*.

2.8. MQTT (Message Queue Telemetry Transport)

MQTT merupakan salah satu protokol komunikasi *publish/subscribe topic based* yang ringan dan juga sederhana, dirancang untuk perangkat yang memiliki keterbatasan kemampuan, jaringan bandwidth rendah dan latensi tinggi, serta memiliki tujuan untuk menghemat sumber daya sekaligus menjamin kestabilan dalam pengiriman data [26].

MQTT bekerja pada platform TCP/IP dan dikenal ringan karena ukuran paket datanya yang kecil serta konsumsi daya yang minim, dengan *Mosquitto Broker* sebagai *broker open-source* yang ideal untuk pengembangan sistem, di mana server *MQTT* mengirim data ke *broker*, kemudian diteruskan ke *MQTT client* yang melakukan *subscribe* dan ditampilkan dalam bentuk monitoring dashboard [27].

Pada proyek ini, penulis menggunakan komunikasi *MQTT* untuk mengintegrasikan *Raspberry Pi 4* dengan *ESP32*. Penggunaan *MQTT* harus dalam jaringan yang sama antara laptop, *Raspberry Pi4* dan *ESP32* yang diprogram melalui software Arduino IDE dan *Thony* (bawaan *Raspberry*)

2.9. Klasifikasi Tingkat Kekotoran Solar Panel

Dalam proyek akhir yang berjudul “Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*” menggunakan mikrokontroler serta *image processing* untuk mengetahui seberapa kotor kondisi permukaan *solar panel* agar

cara pembersihan sesuai. Kotoran pada permukaan *solar panel* dibagi menjadi tiga tingkatan, yaitu:

1. Kotor Ringan

Kotor ringan umumnya hanya berupa debu tipis atau partikel lainnya yang menempel pada permukaan *solar panel*. Kotoran ini tidak terlalu mengurangi efektivitas kinerja *solar panel*. Pembersihannya cukup menggunakan semprotan sedikit air dan putaran sikat serta pembersihan dilakukan pada sistem penjadwalan.

2. Kotor Sedang

Ditandai dengan adanya debu yang menumpuk dan noda air hujan yang mengering pada permukaan *solar panel*. Tingkat kekotoran ini mulai mempengaruhi efektivitas kinerja *solar panel*, sehingga diperlukannya pembersihan dengan semprotan air dan putaran sikat. Proses pembersihan dilakukan sebanyak 2 siklus agar permukaan panel bersih secara menyeluruh.

3. Kotor Berat

Kerak, lumut, atau kotoran yang membandel seperti kotoran burung yang menempel kuat pada permukaan panel merupakan tingkatan kotor berat. Hal ini sangat memengaruhi efektivitas kinerja *solar panel* dan diperlukannya pembersihan yang lebih ekstra. Proses pembersihan dapat dilakukan dengan pengulangan siklus sebanyak 3 kali hingga *solar panel* benar-benar bersih dan tidak adanya kotoran yang tersisa.

2.10. Sistem Pembersih Otomatis

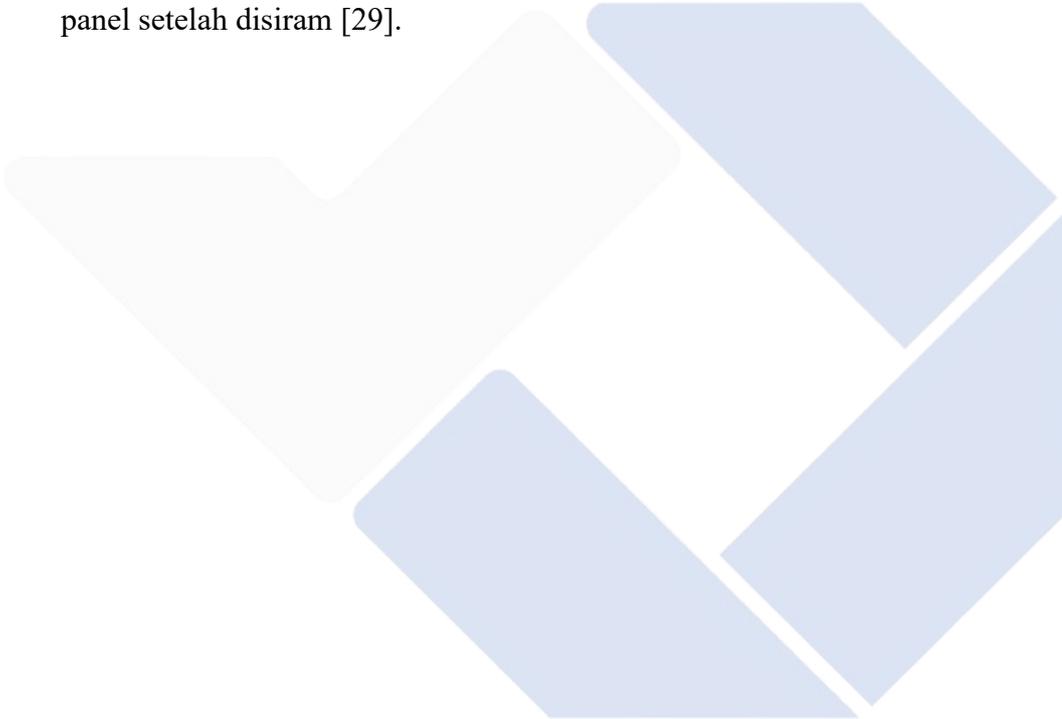
Untuk menjaga kinerja *solar panel* agar tetap efektivitas dibutuhkan solusi yang dapat menghilangkan debu, kotoran, lumut, dan partikel lain yang menempel pada permukaannya. Penelitian menunjukkan bahwa penurunan efektivitas kinerja *solar panel* hingga mencapai 30% [6]. Oleh karena itu, diperlukannya solusi pembersihan otomatis, khususnya untuk panel-panel di area yang sulit dijangkau. Adapun beberapa contoh jenis pembersihan otomatis untuk *solar panel* telah dikembangkan, termasuk:

1. Sistem Pembersih berbasis Wiper Otomatis

Wiper otomatis yang digunakan bergerak melintasi permukaan *solar panel* untuk membersihkan debu dan partikel kotoran lain. Dilengkapi dengan sensor RTC dan Arduino Uno untuk penginputan program mikrokontroler yaitu pengaturan waktu pembersihan secara otomatis [28].

2. Sistem Pembersih berbasis Wiper dan Semprotan Air

Pada sistem ini menggabungkan wiper dan semprotan air untuk membersihkan kotoran pada permukaan panel. Semprotan air membantu melarutkan debu serta wiper yang berfungsi untuk menyapu air dari permukaan panel setelah disiram [29].



BAB III

METODE PELAKSANAAN

3.1. Diagram Alir

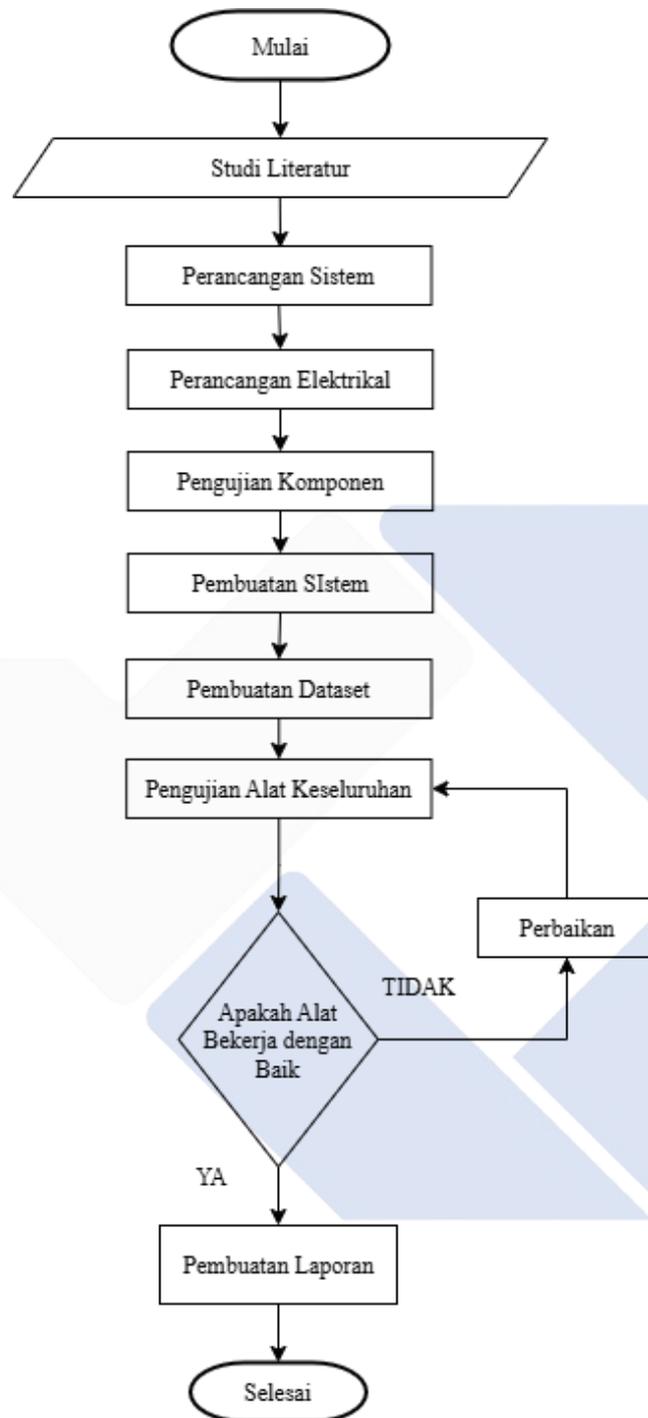
Diagram alir merupakan gambaran berupa tahapan kegiatan yang akan dilakukan dalam pelaksanaan proyek akhir. Pada proyek akhir dengan judul “Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*”, diagram alir digunakan untuk memberikan gambaran umum tahapan proses secara terstruktur yang dimulai dari studi literatur hingga pembuatan laporan. Metode pelaksanaan dapat dilihat pada gambar 3.1.

3.2. Studi Literatur

Tahapan terpenting dalam pembuatan proyek akhir dan merupakan landasan dasar dalam menafsirkan cakupan ruang dan latar belakang teknologi yang dipakai disebut dengan studi literatur. Tujuan dilakukannya studi literatur adalah untuk mencari dan mengumpulkan data-data berupa referensi yang relevan agar nantinya dapat digunakan sebagai acuan dalam proses perancangan dan pengimplementasian proyek akhir ini.

Tujuan paling utama dalam melakukan studi literatur yaitu untuk memahami dan memperdalam pemahaman yang berkaitan dengan faktor pendukung tema proyek akhir penulis. Contoh pengumpulan data yang dilakukan antara lain tentang prinsip kerja panel surya, dasar mikrokontroler dan Teknik pengolahan *Image Processing*.

Metode yang dilakukan untuk pengumpulan informasi berasal dari sumber-sumber terpercaya seperti jurnal ilmiah, buku yang relevan, skripsi atau tugas akhir terdahulu, dan website dari situs kredibel. Sumber-sumber yang diambil memiliki keterkaitan langsung dengan proyek akhir berjudul “Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*”. Dengan dilakukannya studi literatur secara mendalam, diharapkan rancangan sistem penulis memiliki dasar ilmiah yang dapat menjawab setiap permasalahan.



Gambar 3.1 Diagram Alir Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*

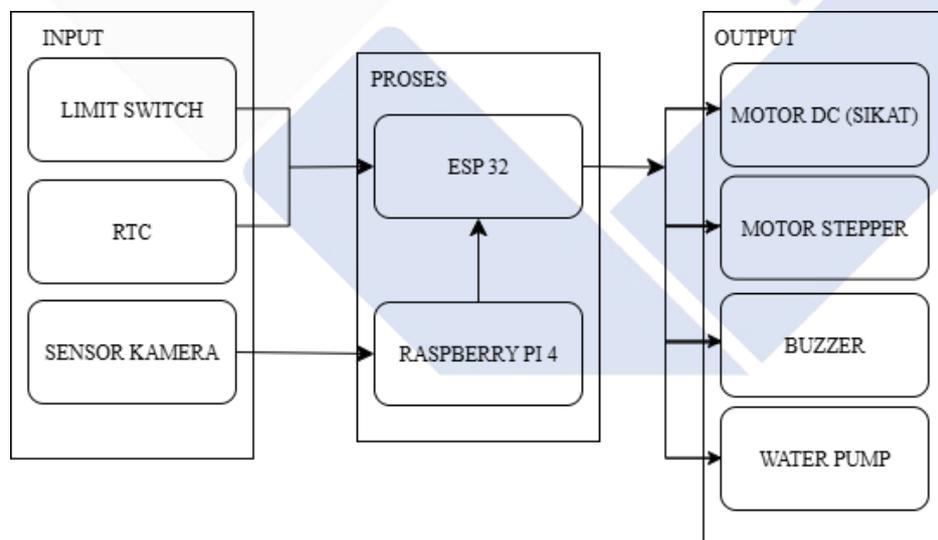
3.3. Perancangan Sistem

Tahap perencanaan proyek ini merupakan proses awal untuk merencanakan dan mengembangkan proyek dari awal hingga akhir pembuatan proyek secara terstruktur. Tujuan, komponen yang dibutuhkan, serta penyusunan rencana implementasi sistem berjalan secara sistematis tercakup dalam tahap ini.

Melalui tahap perencanaan, dapat dilakukan penentuan alur kerja tiap bagian sistem. Kemudian, tahapan ini juga mencakup pengintegrasian sensor kamera yang digunakan sebagai pemantau kondisi *solar panel*. Kamera berperan sebagai sumber data yang paling utama dalam pemrosesan *image processing* sehingga hasil tangkapan gambar dapat dianalisis untuk penentuan pembersihan yang akan dilakukan oleh sistem.

3.3.1 Blok Diagram

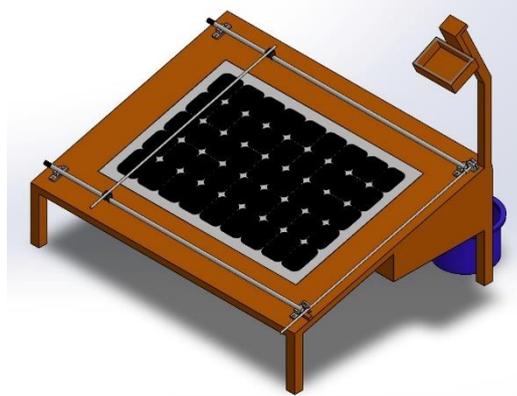
Blok diagram berfungsi untuk menggambarkan alur kerja dan hubungan antar komponen utama dalam sebuah rancangan alat. Gambar 3.2 menunjukkan blok diagram dari sistem yang sedang dikembangkan.



Gambar 3.2 Blok Diagram Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*

3.3.2 Desain Alat

Representasi visual dari alat yang sedang dikembangkan dapat dilihat pada gambar 3.3 yang memperlihatkan desain alat yang dirancang pada proyek ini.

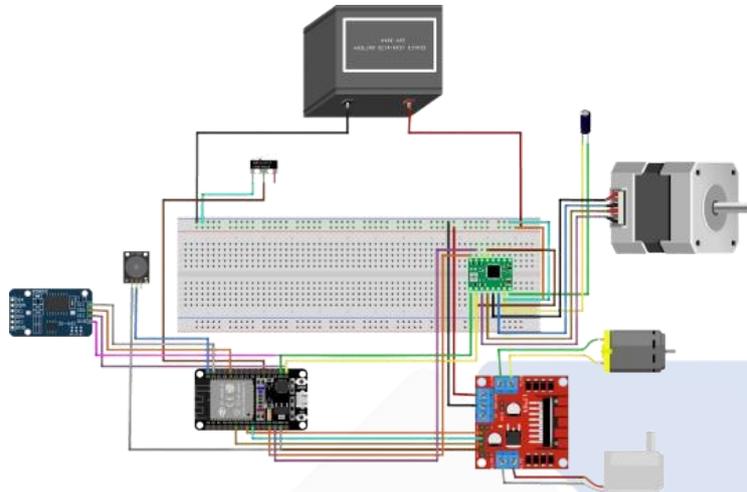


Gambar 3.3 Desain Alat Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*

3.4. Perancangan Elektrikal

Tahapan yang dilakukan selanjutnya adalah merancang sistem elektrikal. Bagian ini memiliki peran sebagai sistem kontrol yang berfungsi untuk mengatur dan menghubungkan seluruh komponen, agar dapat berjalan sesuai dengan perencanaan. Pada tahapan ini, penulis merancang rangkaian antar komponen sehingga dapat terintegrasi. Komponen yang digunakan seperti *ESP32*, motor *stepper NEMA 17HS3401*, motor *DC 555*, *RTC DS3231*, *driver motor stepper A4988*, *driver motor DC L298N*, *transistor 100 μ F*, dan *buzzer* diintegrasikan dalam sebuah rangkaian yang saling terhubung satu sama lain.

Penulis menggunakan aplikasi *fritzing* untuk memudahkan dalam proses perancangan elektrikal. Aplikasi ini dipilih karena lebih mudah dalam penggunaannya dan *library* yang digunakan juga mudah dicari. Adanya gambar visual dari perencanaan ini memudahkan penulis dalam memahami alur kerja sistem dan acuan dalam proses perakitan alat. Hasil rancangan dapat dilihat pada gambar 3.4 yang menampilkan rancangan hubungan antar komponen secara detail antara satu sama lain yang terintegrasi.

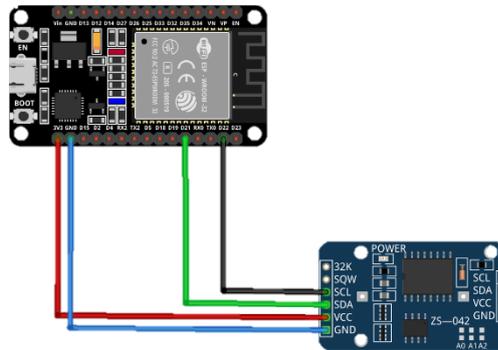


Gambar 3.4 Rancangan Elektrikal

3.5. Pengujian Komponen

Pengujian komponen merupakan langkah yang perlu dilakukan sebelum sistem diintegrasikan. Tujuan dilakukan pengujian komponen adalah untuk memastikan bahwa setiap komponen yang dipilih dapat berfungsi dengan baik, sehingga angka potensi kesalahan saat pengintegrasian sistem masih dalam skala kecil. Komponen yang diuji antara lain *RTC*, *water pump*, motor stepper, motor DC, *Raspberry Pi 4*, *ArduCAM*. Berikut adalah penjelasan pengujian tiap-tiap komponen

3.5.1. Pengujian RTC 3231



Gambar 3.5 Rangkaian Pengujian RTC 3231

Tujuan dilakukannya pengujian pada RTC (*Real Time Clock*) adalah untuk memastikan bahwa sistem dapat memberikan waktu dan tanggal sesuai dengan waktu aktual. Modul RTC yang digunakan dengan tipe DS3231. Dilakukan dengan cara membaca waktu dan tanggal dari RTC melalui *software* yang telah diprogram. Kemudian akan disesuaikan dengan waktu yang sebenarnya, dapat menggunakan jam digital maupun gawai sebagai acuan. Komponen dinyatakan berfungsi dengan baik jika data yang ditampilkan sesuai dan tidak memiliki selisih waktu yang signifikan dengan waktu nyata. Rangkaian pengujian yang digunakan untuk menguji modul *RTC DS3231* ditampilkan pada gambar 3.5.

Tabel 3.1 Hasil Pengujian RTC 3231

No.	Tanggal	Waktu	Keterangan
1.	22/5/2025	09:30:37	Pagi
2.	22/5/2025	17:10:56	Sore
3.	22/5/2025	20:05:11	Malam

Tabel 3.1 menunjukkan hasil pengujian pada RTC yang dilakukan pada tanggal 22 Mei 2025. Tujuan dari pengujian ini adalah untuk memastikan bahwa RTC dapat berfungsi sesuai dengan waktu yang tepat dan konsisten sepanjang hari.

Pengujian dilakukan sebanyak tiga kali dalam sehari, yaitu pagi, sore, dan malam hari:

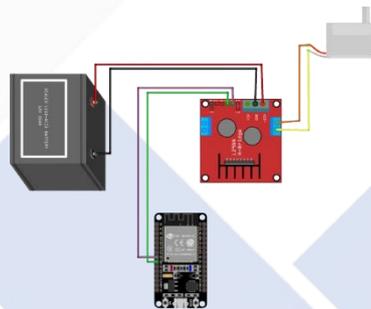
- Pukul 09:30:37 (pagi). Pengujian pertama dilakukan pada pagi hari untuk memeriksa apakah RTC dapat mencatat waktu dengan benar setelah sistem

dinyalakan atau di reset. Hasil pengujian menunjukkan waktu yang ditampilkan sesuai dengan waktu nyata.

- Pukul 17:10:56 (sore). Pengujian kedua dilakukan pada sore hari. Pada tahap ini, RTC masih menunjukkan waktu yang akurat, yang berarti RTC terus berjalan dengan stabil sesuai dengan pengujian pertama.
- Pukul 20:05:11 (malam). Pengujian terakhir dilakukan di malam hari. Waktu yang tercatat oleh RTC sesuai dengan waktu nyata, menunjukkan bahwa RTC bekerja dengan konsisten.

Dari hasil pengujian diatas dapat disimpulkan bahwa RTC berfungsi sesuai dengan waktu nyata pada saat pengujian dilakukan. Adapun untuk selisih perbedaan antara waktu nyata dan waktu yang ditampilkan di *serialmonitor* saat pengujian hanya dalam kisaran bebrapa *milisecond*. Selisih ini dapat ditoleransi dan tidak akan berdampak terlalu signifikan terhadap keakuratan dalam pencatatan waktu.

3.5.2. Pengujian *Waterpump*



Gambar 3.6 Rangkaian Pengujian *Waterpump*

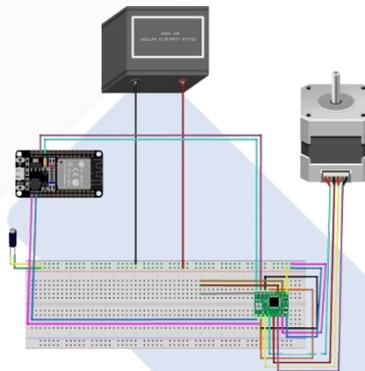
Pengujian komponen ini dilakukan untuk memastikan bahwa *waterpump* dapat berfungsi dalam sistem, terkhusus sesuai fungsinya yaitu memompa air dalam waktu tertentu dan sesuai kebutuhan. Proses pengujian dilakukan dengan cara mengaktifkan pompa selama 5 detik, 10 detik, dan 15 detik untuk mengamati apakah air mengalir lancar hingga ke wadah penampung. Rangkaian pengujian *waterpump* dapat dilihat pada gambar 3.6.

Tabel 3.2 Hasil Pengujian *Waterpump*

No.	Waktu Operasi (detik)	Volume Air yang Dipompa (ml)	Debit Air (ml/detik)	Keterangan
1	5	247	43	Pompa menyala dengan baik
2	10	490	46	Debit stabil
3	15	729	50	Pompa menunjukkan aliran terbaik

Data hasil pengujian menunjukkan bahwa pompa bekerja sesuai dengan fungsinya, puncak kinerja pompa pada pengujian yang ketiga dengan durasi selama 15 detik seperti terlihat pada tabel 3.2. Dengan demikian, dapat disimpulkan bahwa semakin lama pompa dinyalakan, maka pompa akan menunjukkan kestabilan dalam kinerjanya.

3.5.3. Pengujian Motor Stepper NEMA 17HS3401



Gambar 3.7 Rangkaian Pengujian Motor Stepper *NEMA 17HS3401*

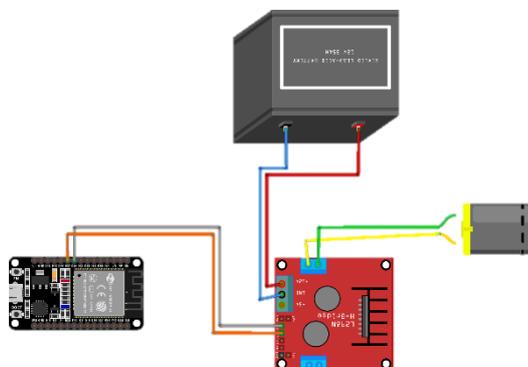
Pengujian dilakukan untuk menggerakkan motor searah jarum jam dan berlawanan arah jarum jam dengan siklus tertentu. Menghubungkan motor stepper ke driver L298N dan mengirimkan sinyal *step* dan *direction* dari mikrokontroler. Komponen berfungsi dengan baik apabila motor bergerak stabil dan akurat. Menggunakan kapasitor 100 mikrofarat untuk menstabilkan tegangan yang masuk ke driver motor, meminimalisir adanya lonjakan tegangan. Rangkaian pengujian motor stepper *NEMA 17HS3401* dapat dilihat pada gambar 3.7.

Tabel 3.3 Hasil Pengujian Motor Stepper NEMA 17HS3401

No.	Arah Putaran	Jumlah Step	Hasil (Putaran)	Kecepatan (RPM)	Keterangan
1.	Searah Jarum Jam	200	360°	1000 μ s	Selesai 1 Putaran
2.	Berlawanan Arah Jarum Jam	200	360°	1000 μ s	Selesai 1 Putaran
3.	Searah Jarum Jam	200	360°	1000 μ s	Selesai 1 Putaran
4.	Berlawanan Arah Jarum Jam	200	360°	1000 μ s	Selesai 1 Putaran
5.	Searah Jarum Jam	200	360°	1000 μ s	Selesai 1 Putaran
6.	Berlawanan Arah Jarum Jam	200	360°	1000 μ s	Selesai 1 Putaran

Berdasarkan pengujian yang telah dilakukan pada motor stepper dengan step sebanyak 200, dapat disimpulkan bahwa motor dapat berputar sebanyak 1 putaran penuh (360°) baik searah jarum jam maupun berlawanan arah jarum jam. Dari seluruh pengujian yang dilakukan menunjukkan bahwa motor berfungsi dengan baik dan stabil, seperti terlihat pada tabel 3.3.

3.5.4. Pengujian Motor DC 555



Gambar 3.8 Rangkaian Pengujian Motor DC 555

Tegangan dialirkan melalui driver L298N dan sinyal PWM untuk mengatur kecepatan putar motor DC 555. Pengujian dilakukan untuk mendeteksi motor berputar dengan lancar dan dapat berubah arah. Rangkaian pengujian motor DC 555 ditampilkan pada gambar 3.8.

Tabel 3.4 Hasil Pengujian motor DC 555

No.	Arah Putaran	Durasi (s)	Status Motor	Keterangan
1.	Searah jarum jam	5	Berputar agak cepat	Terjadi getaran kejut
2.	Berlawanan arah	10	Berputar agak lambat	Terjadi sedikit getaran
3.	Searah jarum jam	15	Berputar stabil	Kecepatan stabil
4.	Berlawanan arah	20	Berputar stabil	Kecepatan stabil

Berdasarkan pengujian diatas, motor menunjukkan kinerja yang bervariasi. Saat motor mulai bergerak, terjadi getar kejut dan berputar dengan sangat cepat. Begitu pula saat motor berputar berlawanan arah jarum jam. Namun, saat pengujian dengan durasi yang lebih Panjang, kecepatan akan stabil dan getaran terjadi dengan sangat minim, seperti terlihat pada tabel 3.4.

3.6. Pembuatan Sistem

Tahap pengimplementasian dari perencanaan yang telah dilakukan sebelumnya adalah pembuatan sistem. Di tahap ini, semua komponen yang dipilih dan dirancang mulai diintegrasikan dengan cara pembentukan sistem kerja yang menyeluruh dan berfungsi sesuai dengan tujuan proyek akhir ini. Perangkaian sistem tidak hanya berbentuk fisik, tetapi juga mulai mencicil pengujian awal fungsinya apakah sesuai dengan tujuan yang telah ditetapkan. Proses ini dilakukan dengan ekstra hati-hati untuk meminimalisir adanya *troubleshooting*. Dengan demikian, tahap ini merupakan landasan utama dalam memulai penyusunan alur sistem terintegrasi agar nantinya dapat dilakukan pengujian secara menyeluruh.

3.7. Pembuatan Dataset

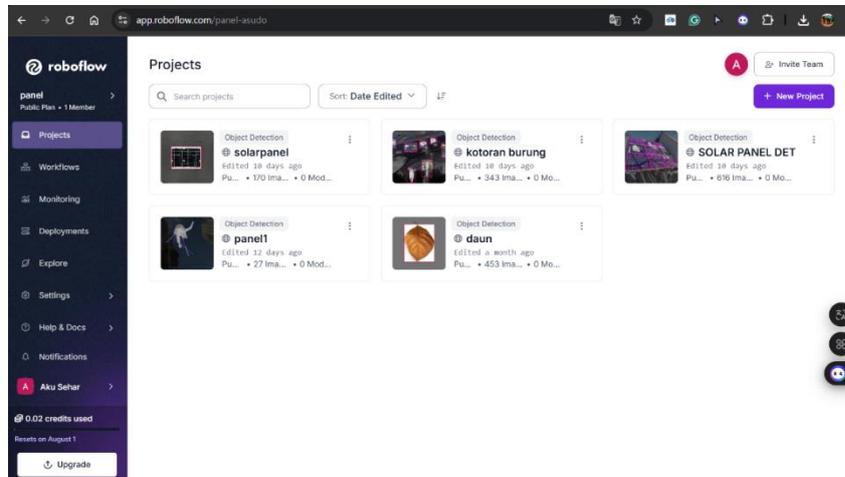
Pada proyek akhir ini penulis mengadakan pembuatan dataset objek permukaan *solar panel* berupa foto dengan sudut pandang tampak atas agar lebih memudahkan dalam merepresentasikan visual yang jelas terhadap kondisi permukaannya. Setelah penangkapan foto dilakukan, terdapat kumpulan data dengan pembagian 3 tingkatan kekotoran yang terlihat pada permukaan *panel cell*, yaitu kotor ringan, kotor sedang, dan kotor berat. Pada tahapan ini dilakukan penangkapan gambar secara manual menggunakan kamera gawai dan webcam untuk ketepatan visual berdasarkan kondisi yang aktual dari tiap-tiap gambar yang ditangkap.

Pembuatan dataset meliputi beberapa tahapan sebelum dataset bisa dijadikan sebagai model dalam deteksi objek. Adapun tahapan tahapan yang dimaksud akan dijelaskan di bawah ini.

3.7.1. Pelabelan Gambar

Dalam proses pengidentifikasi tingkat kekotoran solar panel terdapat 1609 label gambar yang terkumpul. Kelas utamanya seperti panel bersih, daun, kotoran burung, debu ringan, debu sedang, debu berat, dan *solar panel*. Adapun web *software* yang digunakan untuk memudahkan dalam pelabelan gambar yaitu *RoboFlow*. *RoboFlow* sendiri dirancang agar dapat membantu para pengembang *computer vision* dalam pengelolaan, penganotasian serta persiapan untuk dataset agar dapat digunakan sebagai model deteksi objek [30].

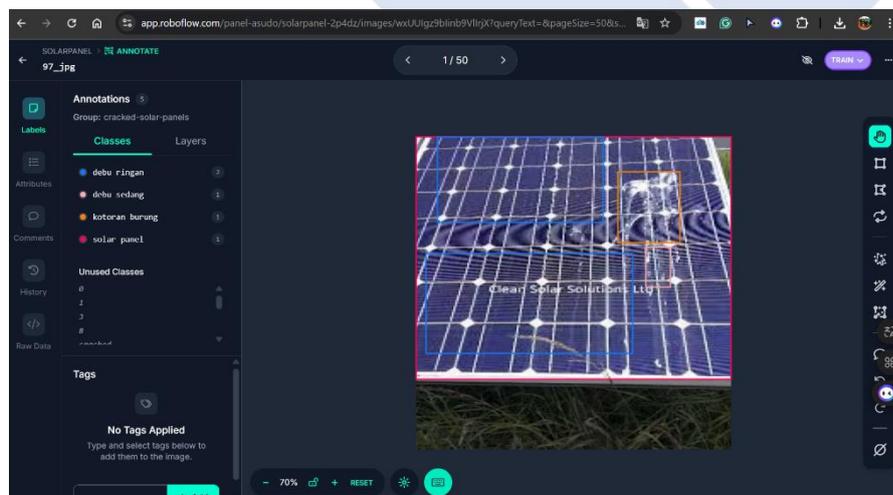
RoboFlow akan membuat sebuah label gambar yang memberikan sebuah informasi mengenai objek yang dideteksi. Label gambar ini nantinya akan berfungsi sebagai inputan dalam proses pengolahan citra untuk memberikan analisis sistem pembersihan yang akan dilakukan. Label yang dihasilkan diklasifikasikan menjadi 3 kelas yaitu *test*, *train*, dan *valid*. Tampilan *interface RoboFlow* yang digunakan dalam proses pelabelan dapat dilihat pada gambar 3.9 di bawah ini.



Gambar 3.9 Tampilan *RoboFlow*

3.7.2. Anotasi Label Gambar

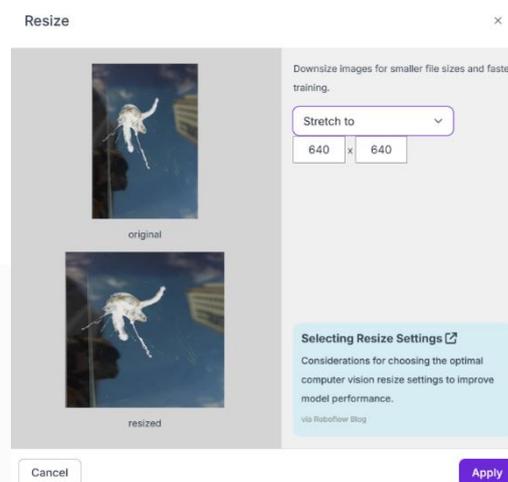
Proses pengannotasian gambar merupakan tahapan awal dalam pembuatan label gambar. Anotasi dilakukan dengan memberi tanda pada objek tertentu yang telah ditentukan kelasnya berupa *bounding box* (kotak pembatas) di sekeliling objek yang disertai dengan label klasifikasi menunjukkan jenis objek tersebut. Proses anotasi dilakukan agar akurasi dalam deteksi objek akan semakin baik dan akurat. Semakin akurat anotasi, semakin baik kinerja model dalam mengidentifikasi objek di dunia nyata. Gambar 3.10 menunjukkan tampilan web proses anotasi label gambar.



Gambar 3.10 Anotasi label gambar

3.7.3. Resize Label Gambar

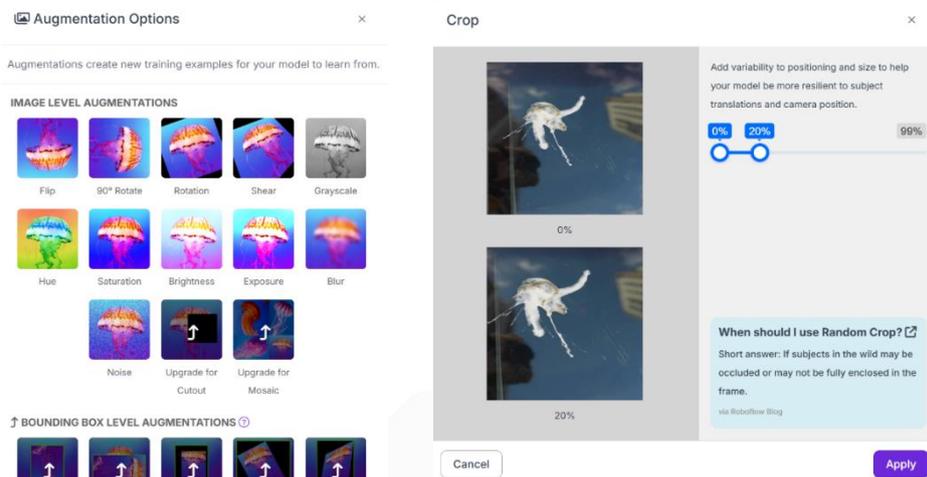
Proses resize dilakukan untuk mengubah ukuran tiap-tiap label gambar agar ukuran semua label sama dan sesuai dengan model YOLO yang dipilih. Tujuan dilakukan resize agar proses train menjadi lebih stabil, efisien, dan kompatibel dengan model YOLO yang digunakan. Selain itu, ukuran gambar yang sama akan membuat sistem memproses data lebih cepat dan akurat. Gambar 3.11 menampilkan suntingan gambar resize label gambar ke ukuran 640x640 piksel.



Gambar 3.11 Resize label gambar

3.7.4. Augmentasi Label Gambar

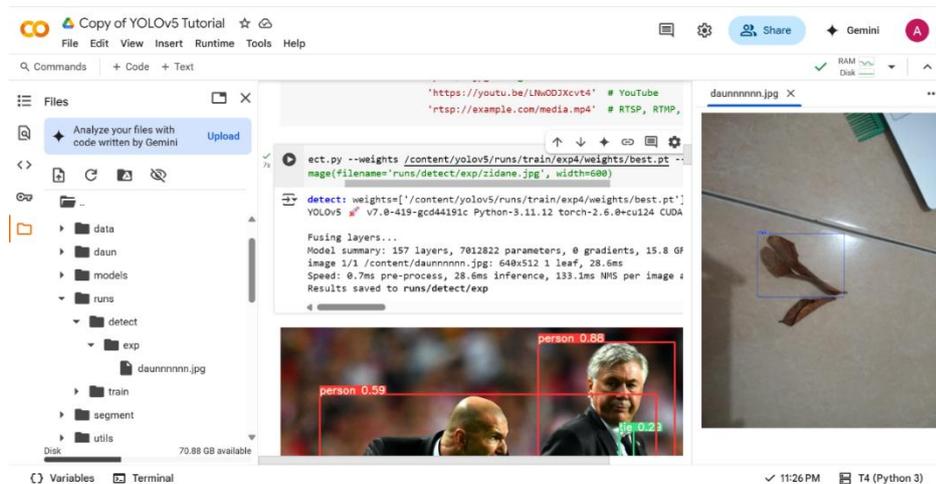
Augmentasi label gambar merupakan suatu teknik untuk menambah variasi data penelitian dengan cara memodifikasi citra yang sudah ada secara visual. Beberapa teknik augmentasi yang umum digunakan adalah memutar gambar, gambar flipping, noise reduction, pencahayaan, dan lain-lain. Tujuan augmentasi adalah untuk membuat model lebih robust atau tahan terhadap berbagai kondisi nyata, seperti pencahayaan yang berbeda, sudut pandang yang berbeda, atau kualitas gambar yang kurang ideal. Hal ini dapat meningkatkan akurasi model saat diterapkan di lapangan. Tampilan augmentasi gambar dapat dilihat pada gambar 3.12.



Gambar 3.12 Augmentasi label gambar

3.7.5. Training Label Gambar

Training label gambar merupakan label gambar yang telah diproses (melalui pengubahan ukuran, anotasi, dan penambahan) dan digunakan untuk melatih model AI sehingga dapat menganalisis data dan objek sesuai dengan label yang telah diberikan. Pada tahap ini, model akan diajarkan melalui ilustrasi berdasarkan fitur visual yang ada. Proses pelatihan dilakukan dengan menggunakan algoritma *machine learning* atau *deep learning* seperti *YOLO*. Semakin beragam dan melimpah data yang digunakan untuk pelatihan, semakin besar kemungkinan model akan melakukan deteksi dengan akurasi yang lebih tinggi. Pada proses training penulis menggunakan platform berbasis website yaitu *Google Colab*. *Google Colab* merupakan sebuah media yang memungkinkan pengguna dalam menjalankan serta mengirimkan *notebook Jupyter* secara online [31]. Pemilihan *Google Colab* dikarenakan platform ini gratis dan mudah digunakan terutama pada proyek yang menggunakan *python*. Tampilan proses *training* gambar dapat dilihat pada gambar 3.13.



Gambar 3.13 Training label gambar

3.8. Pengujian Alat Keseluruhan

Pengujian alat merupakan tahapan terpenting untuk memastikan bahwa rancangan sistem yang telah diintegrasikan dapat berfungsi dengan menyeluruh dan sesuai dengan tujuan awal. Sistem yang diuji dilakukan secara *real-time* untuk memudahkan dalam proses pengevaluasian kendala dalam pengidentifikasian tingkat kekotoran permukaan solar panel serta pengambilan keputusan untuk proses pembersihan.

Cakupan pengujian ini berupa kondisi tingkat kekotoran *solar panel*, mulai dari kondisi bersih hingga kotor berat. Dari sini dapat dinilai apakah *image processing* mampu mendeteksi kotoran pada permukaan *solar panel* serta apakah mampu membedakan permukaan panel yang perlu dibersihkan.

Pengujian alat dilakukan untuk mengidentifikasi apakah kemampuannya dalam mendeteksi tingkat kekotoran pada permukaan *solar panel* sudah memenuhi kriteria atau perlu dilakukannya perbaikan, baik pada pengolahan *image processing*, kecepatan respon, gerakan pembersih, maupun kestabilan sistem secara keseluruhan dalam jangka waktu tertentu. Sistem yang diuji dilakukan secara *real-time* untuk memudahkan dalam proses pengevaluasian sistem.

3.9. Pembuatan Laporan dan Publikasi Ilmiah

Setelah dilakukannya pengujian alat secara keseluruhan, tahap selanjutnya adalah menganalisis dan mengevaluasi kinerjanya. Jika alat belum berfungsi sesuai dengan rancangan awal, maka akan dilakukan perbaikan hingga sesuai dengan rancangan. Sebaliknya, jika alat sudah berfungsi dengan baik dan sesuai dengan rancangan, dilanjutkan dengan proses penyusunan laporan proyek akhir serta persiapan untuk publikasi ilmiah.



BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini penulis akan menjelaskan dan menguraikan terkait proses pembuatan Proyek Akhir dengan judul “Sistem Pembersih Solar Panel berbasis Microcontroller dan Image Processing”. Berikut tahapan proses yang dilakukan penulis:

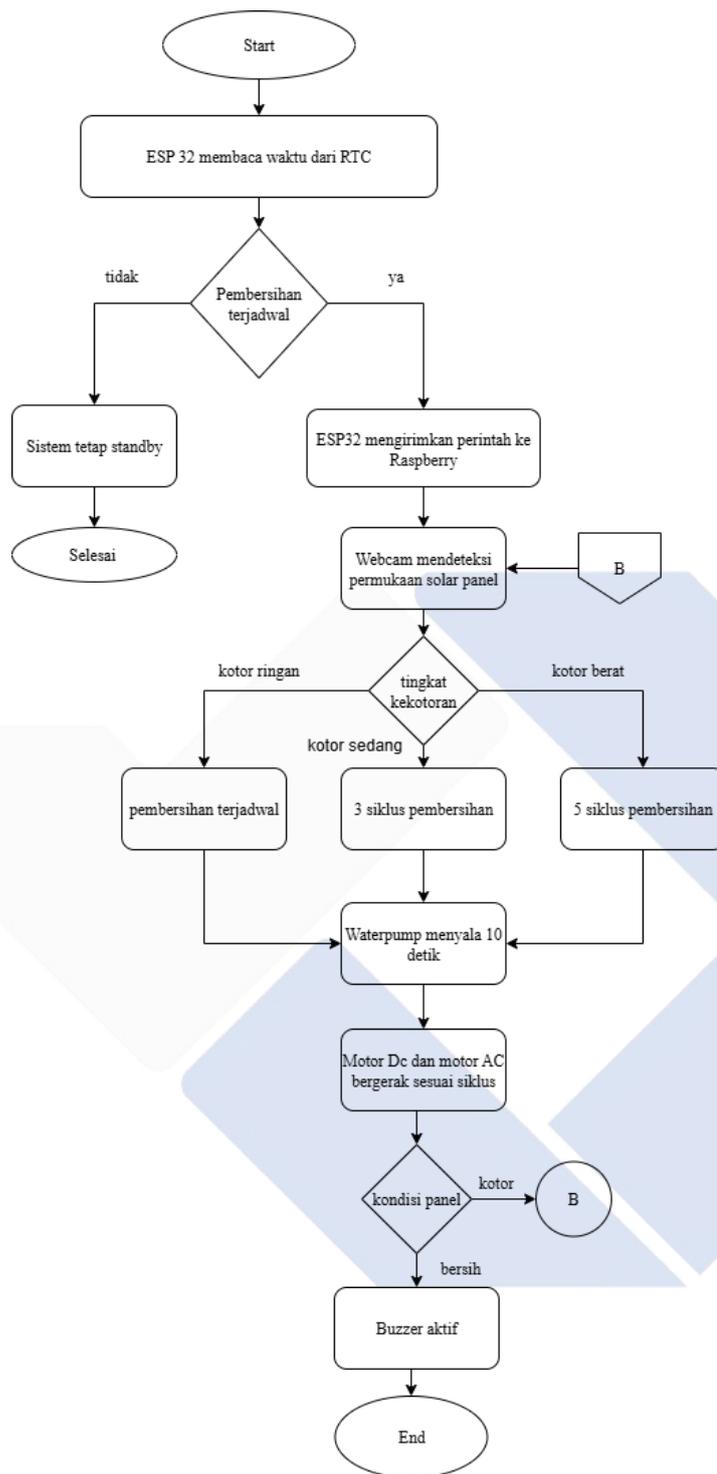
4.1. Sistem Kerja Keseluruhan Alat

Agar gambaran mengenai sistem kerja keseluruhan alat dapat diberikan secara jelas, maka diperlukannya sebuah flowchart yang akan menggambarkan alur sistem kerjanya. Flowchart berfungsi untuk memahami proses kerja sistem dengan lebih mudah terutama bagi para pembaca yang berkemungkinan belum terlalu familiar dengan sistem yang diuji. Adapun bentuk flowchart dari alat yang dibuat dapat dilihat pada gambar 4.1.

1. Pada sistem ini, *Raspbbery Pi 4* dan *ESP32* merupakan komponen utama yang saling terhubung untuk bekerja sama dalam mengontrol proses pembersihan permukaan *solar panel* secara keseluruhan, kedua perangkat ini dihubungkan melalui komunikasi *MQTT*. Kamera webcam yang terhubung langsung ke *raspberry* digunakan untuk proses penangkapan gambar serta analisis gambar yang ditangkap.
2. *ESP32* akan membaca waktu dari *RTC* secara berkala untuk memastikan waktu pembersihan terjadwal sesuai dengan jadwal yang ditentukan. Jika waktu sesuai, *ESP32* akan mengirimkan sebuah perintah ke *Raspbbery Pi* untuk menganalisis gambar agar nantinya dapat dilakukan pembersihan terjadwal. Jika waktu pembersihan sesuai jadwal yang ditentukan, *ESP32* mengirim perintah ke *Raspbbery Pi* untuk melakukan proses analisis gambar.
3. *Raspbbery Pi* mengaktifkan webcam untuk menangkap gambar permukaan *solar panel*. Gambar yang berhasil ditangkap dikirimkan melalui *OpenCv* dengan menggunakan bahasa pemrograman *Python* yang bertujuan untuk

mendeteksi kondisi permukaan *solar panel*. Jika hasil analisis yang didapatkan bahwa *solar panel* dalam keadaan bersih, sistem akan berada pada kondisi *standby* dan sistem tidak bekerja. Namun, jika terdeteksi adanya kotoran, *raspberry* akan mengirimkan perintah pembersihan ke *ESP32* untuk memuali prosesnya.

4. Pompa air akan menyala selama 10 detik untuk membasahi permukaan panel. Kemudian setelah 10 detik, motor *DC* akan menyala melalui *driver L298N* bersamaan dengan motor *Stepper NEMA 17HS3401* melalui *driver A4988* yang akan bergerak untuk membersihkan permukaan *solar panel*. Sistem ini dirancang agar permukaan panel dapat dibersihkan secara merata.
5. Setelah 1 siklus pembersihan selesai, *Raspberry Pi* akan kembali menangkap gambar untuk memastikan bahwa panel dalam kondisi bersih. Jika panel masih dalam kondisi kotor, siklus pembersihan akan diulang hingga panel terdeteksi bersih.
6. Setelah panel terdeteksi bersih dan sistem pembersihan dikatakan selesai, *Raspberry Pi* akan mengirimkan perintah untuk menghentikan aktivitas sistem pembersihan ke *ESP32*. *Buzzer* akan menyala sebagai triger bahwa panel sudah dalam kondisi bersih.



Gambar 4.1 Sistem kerja alat

4.2. Uji Coba Sistem Pendeteksian

Pada tahap ini, dilakukan pengujian terhadap sistem pendeteksian kotoran pada permukaan *solar panel*. Uji coba dilakukan dari penginputan gambar melalui *webcam* yang bertujuan untuk mengetahui bahwa model *image processing* yang dilatih mampu mengenali objek dengan sangat baik. Pengujian dilakukan menggunakan laptop dan *webcam* tanpa terintegrasi dengan mikrokontroler maupun aktuator.

Program dijalankan melalui *Windows PowerShell* dimana program yang digunakan berbasis *python* telah terintegrasi *YOLOv5* dan *environment virtual python* terlebih dahulu telah diaktifkan. Urutan perintah pendeteksian yang dijalankan ditunjukkan pada gambar 4.2.

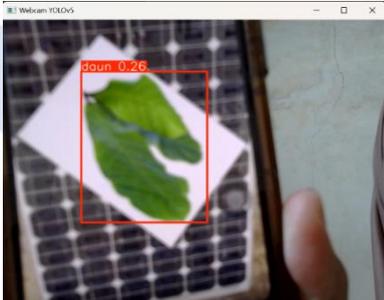
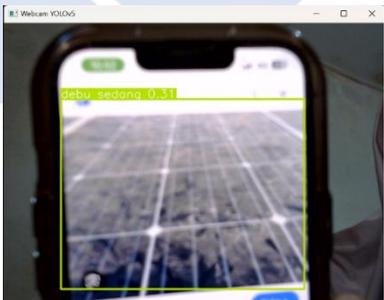
```
PS C:\Users\Lenovo> cd D:\yolov5-project
PS D:\yolov5-project> .\yolov5-env\Scripts\Activate.ps1
(yolov5-env) PS D:\yolov5-project> cd yolov5
(yolov5-env) PS D:\yolov5-project\yolov5> python detect_webcam.py
```

Gambar 4.2 perintah pada *Windows PowerShell*

Model pendeteksian *Image Processing* yang sebelumnya telah dilatih menggunakan dataset *solar panel* dengan berbagai kondisi dijalankan melalui perintah `python detect_webcam.py`. Perintah ini dipanggil dalam *environment python* menggunakan *Windows PowerShell*. Fungsinya untuk mengaktifkan proses inferensi secara *real-time* dari input gambar *webcam*. Hasil deteksi yang ditampilkan berupa *Bounding Box* dengan label dan *confidence score* yang menunjukkan jenis objek yang terdeteksi, misalnya panel bersih, daun, kotoran, dan lainnya sesuai dengan anotasi dataset yang telah dilakukan sebelumnya.

Untuk mengetahui tingkat akurasi sistem, dilakukan persentase error dari setiap hasil deteksi. Nilai ini dihitung berdasarkan selisih nilai *confidence* terhadap keyakinan sempurna (100%). Semakin tinggi persentase error, maka semakin rendah tingkat kepercayaan sistem mengenali objek. Hasil uji coba dapat dilihat pada tabel 4.1.

Tabel 4.1 Hasil Uji Coba Sistem Pendeteksian

No.	Label Deteksi	Jarak Deteksi	Confidence Score	Hasil Deteksi
1.	Panel bersih	10 cm	0.35	
2.	Daun	10 cm	0.26	
3.	Kotoran burung	10 cm	0.38	
4.	Debu sedang	10 cm	0.31	

Pengujian dilakukan pada empat kelas objek, hasil pengujian disajikan dalam bentuk confidence score yang menunjukkan tingkat keyakinan yang dikenali. Tabel menunjukkan hasil *confidence score* berada di bawah angka 0.5 dengan nilai terendah 0.26 (error 74%). Hal ini dapat terjadi karena adanya beberapa factor

seperti kualitas pencahayaan saat uji coba, sudut pengambilan gambar yang kurang ideal, dataset yang masih kurang untuk menunjukkan akurasi, serta resolusi kamera yang dipakai.

4.3. Uji Coba Integrasi Sistem Penjadwalan

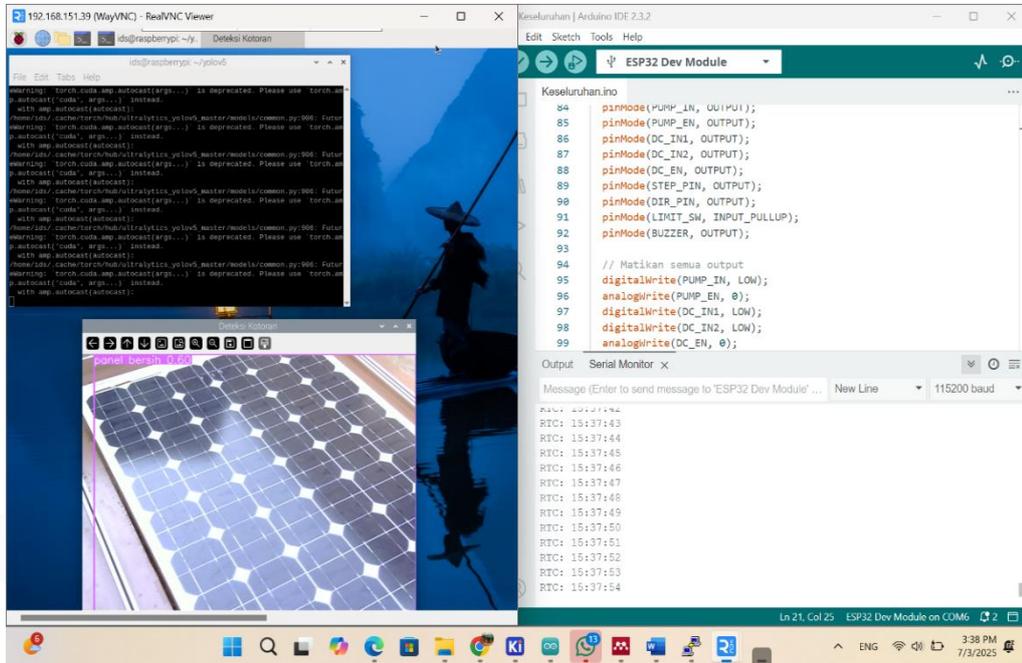
Pada sistem penjadwalan, perlu dilakukan uji coba untuk memastikan bahwa sistem penjadwalan dapat berfungsi sesuai dengan waktu yang ditentukan, serta semua komponen yang telah terintegrasi dapat bekerja secara otomatis tanpa perlu intervensi manual. Modul *RTC DS3231* yang terhubung ke *ESP32* dirancang agar berfungsi sebagai penentu waktu penjadwalan sistem bekerja.

Penetapan jadwal pembersihan dilakukan dengan penetapan frekuensi pembersihan minimum dilakukan sebanyak satu kali dalam satu minggu. Saat mendekati jadwal pembersihan, sistem akan tetap menangkap gambar kondisi permukaan solar panel menggunakan webcam. Gambar yang didapatkan akan dianalisis oleh model *YOLOv5* untuk pengidentifikasian tingkat kekotoran agar dapat menjalankan siklus sistem pembersihan yang harus dilakukan.

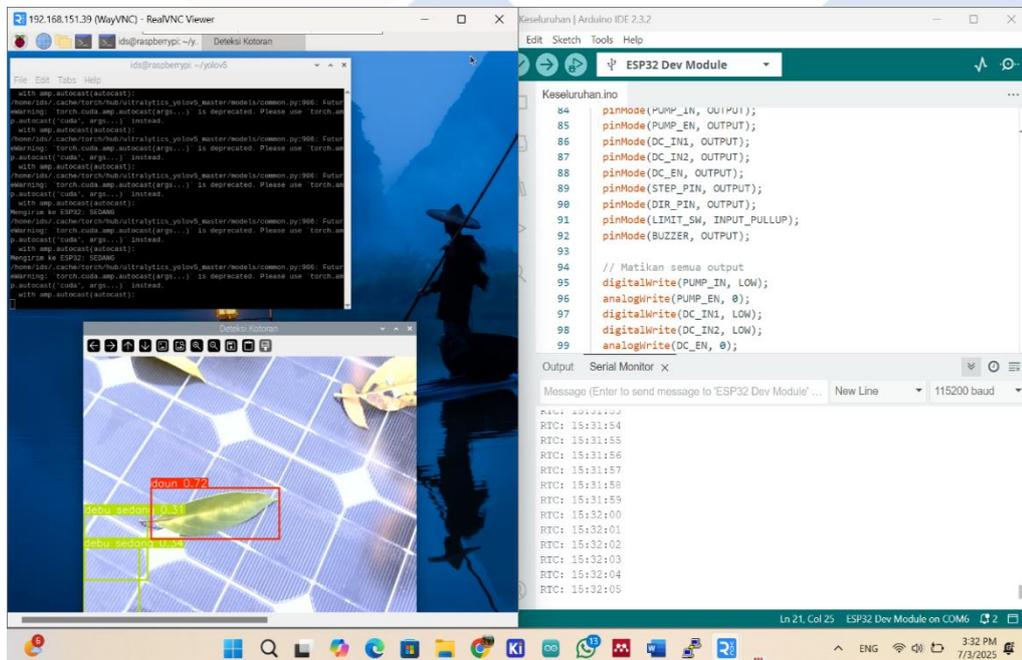
Melalui metode ini diharapkan sistem dapat menjaga performa solar panel agar tetap optimal. Berikut hasil uji coba integrasi sistem penjadwalan dapat dilihat pada tabel 4.2.

Tabel 4.2 Hasil Uji Coba Integrasi Sistem Penjadwalan

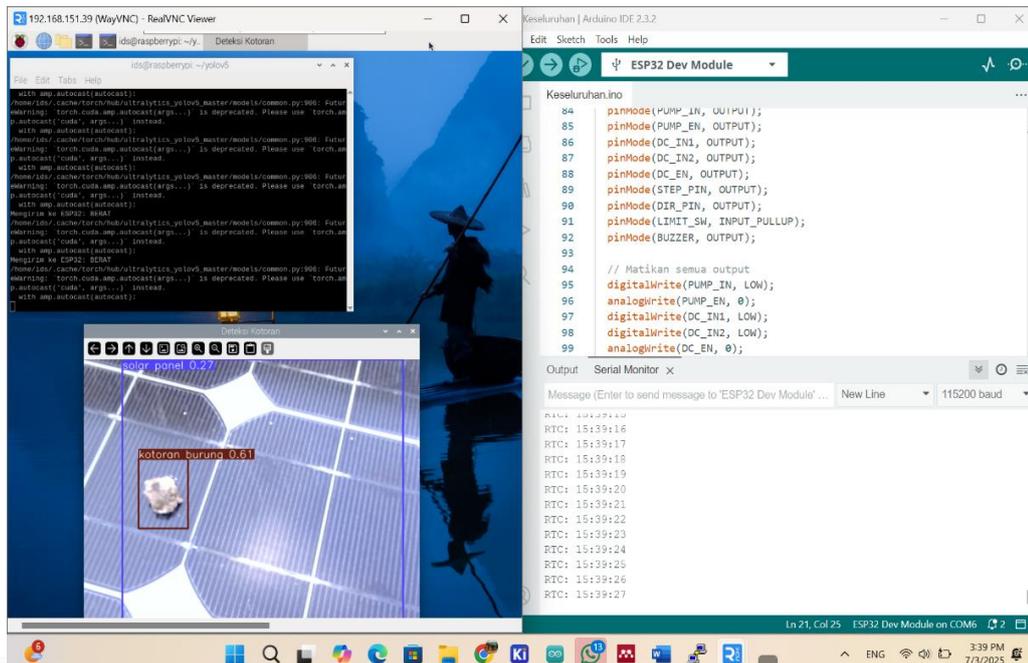
No.	Label Deteksi	Jarak Deteksi	Confidence Score	Respon Sistem	Referensi Gambar
1.	Panel bersih	1 m	0.60	Sistem hanya mendeteksi tanpa ada pembersihan	Lihat gambar 4.3
2.	Daun	60 cm	0.72	Sistem mendeteksi dan melakukan pembersihan sebanyak 2 siklus	Lihat gambar 4.4
3.	Kotoran burung	39 cm	0.61	Sistem mendeteksi kondisi solar panel dan melakukan pembersihan sebanyak 3 siklus	Lihat gambar 4.5



Gambar 4.3 panel bersih



Gambar 4.4 daun



Gambar 4.5 Kotoran Burung

Pada pengujian di atas, dapat diambil kesimpulan bahwa sistem dapat melakukan sebuah pendeteksian objek menggunakan *YOLOv5*, dengan output dalam bentuk label objek dan *confidence score*. Pada saat kondisi panel bersih, sistem mendeteksi label dengan nilai *confidence score* sebesar 0.60. Lantaran tidak terdeteksi adanya objek kotoran pada permukaan panel, maka sistem tidak mengirimkan sinyal pembersihan. Hal ini menunjukkan bahwa hasil sudah sesuai dengan rancangan logika kontrol.

Selanjutnya pada kondisi daun yang berada di atas permukaan *solar panel*. Sistem dapat mendeteksi objek dengan nilai *confidence score* mencapai 0.72. nilai ini menunjukkan nilai yang cukup tinggi sehingga secara otomatis sistem mengirimkan perintah untuk melakukan pembersihan sebanyak 2 siklus karena termasuk ke dalam kategori tingkat kekotoran sedang.

Adapun pada kondisi terdeteksi adanya kotoran burung di atas permukaan *solar panel* dengan nilai *confidence score* sebesar 0.61. Walaupun nilainya lebih rendah dari kondisi pada saat terdapat daun di atas permukaan *solar panel*, sistem tetap mengenali objek kotoran dan mengirimkan perintah pembersihan ke sistem

untuk melakukan 3 kali siklus pembersihan karena jenis kotoran ini umumnya lebih sulit dibersihkan dibanding daun.

Dari ketiga hasil pengujian dapat disimpulkan bahwa sistem penjadwalan sudah terintegrasi dengan baik. Sistem tidak asal bekerja berdasarkan waktu, tetapi juga mendeteksi kondisi panel secara *real-time* untuk menentukan Tindakan pembersihan yang sesuai. Artinya, walaupun sistem memiliki jadwal pembersihan otomatis, sistem akan mendahulukan kondisi aktual panel sebelum adanya penentuan untuk menjalankan sistem pembersihan atau tidak.

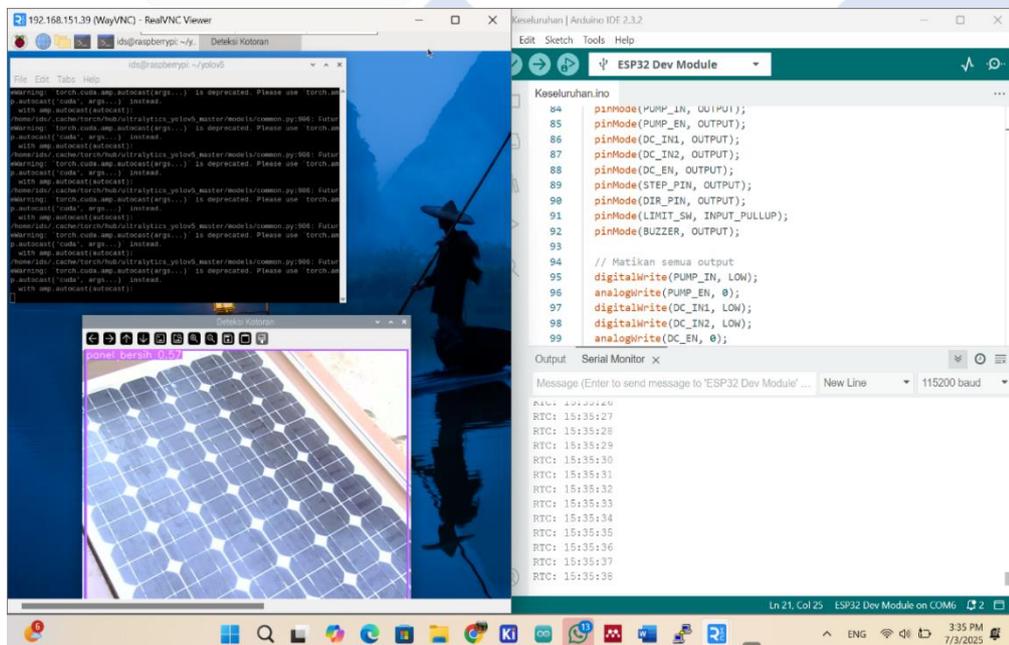
4.4. Uji Coba Integrasi Sistem Otomatis

Proses pengujian integrasi sistem otomatis dilakukan untuk memastikan bahwa seluruh komponen yang telah diintegrasikan dapat bekerja secara terpadu tanpa intervensi manual. Sistem ini dirancang agar dapat mendeteksi tingkat kekotoran solar panel secara otomatis, kemudian memberikan respon sesuai dengan logika pemrograman, termasuk juga pembersihan secara mandiri.

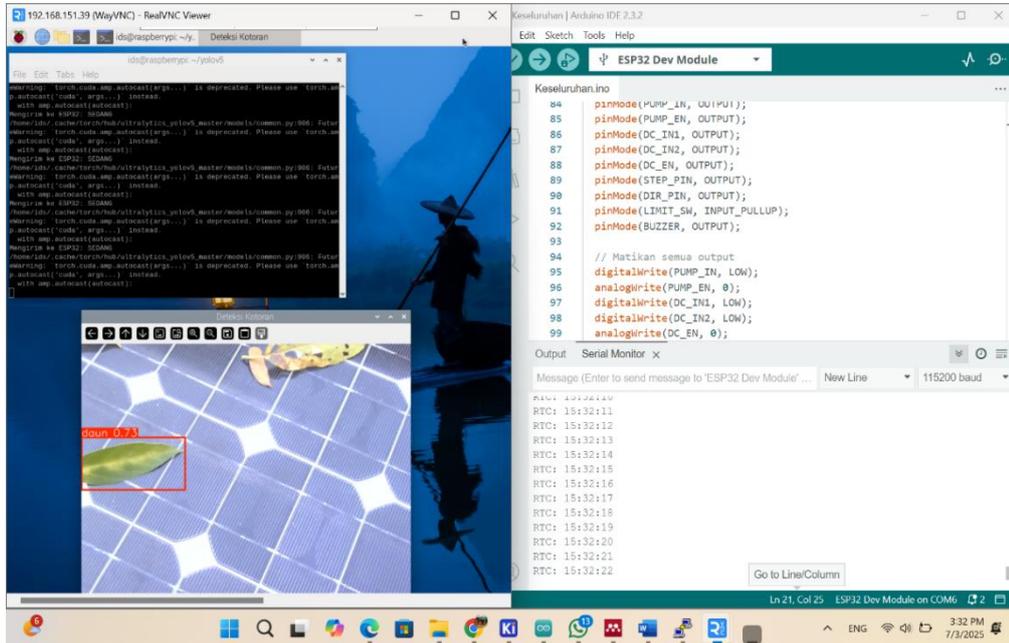
Berbeda pada tahap yang sebelumnya, tahap ini uji coba dilakukan langsung di lapangan dan diuji langsung di atas permukaan solar panel untuk melihat kinerja sistem yang telah dirancang apakah sesuai atau tidak. Hasil uji coba integrasi sistem secara otomatis dapat dilihat pada tabel 4.3 di bawah ini.

Tabel 4.3 Hasil Uji Coba Integrasi Sistem Otomatis

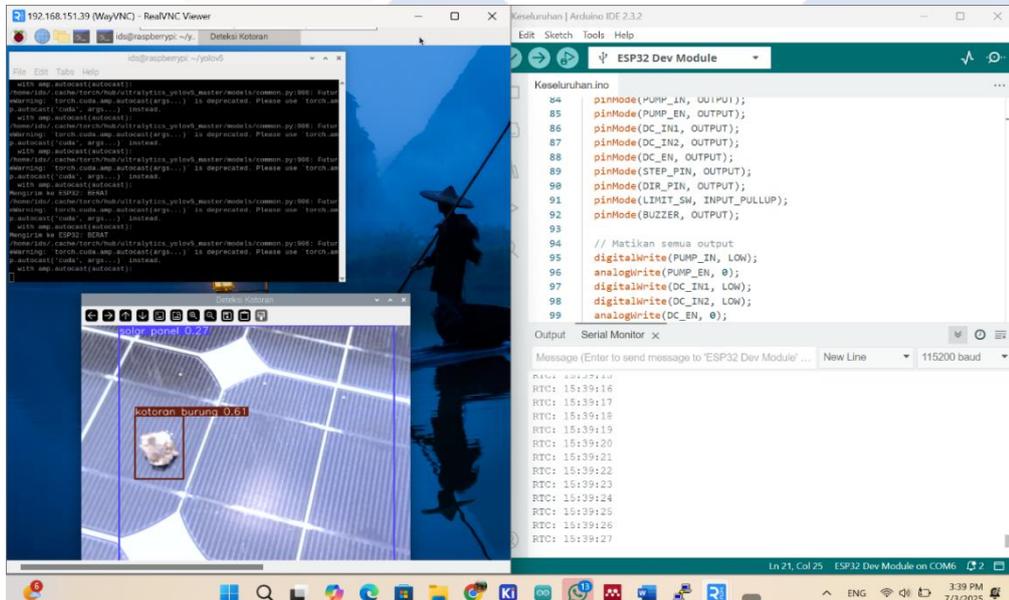
No.	Label Deteksi	Jarak Deteksi	Confidence Score	Respon Sistem	Referensi Gambar
1.	Panel bersih	1 m	0.57	Sistem hanya mendeteksi tanpa ada proses pembersihan	Lihat gambar 4.6
2.	Daun	60 cm	0.73	Sistem mendeteksi kondisi solar panel dan melakukan pembersihan 2 siklus	Lihat gambar 4.7
3.	Kotoran burung	40 cm	0.61	Sistem mendeteksi dan melakukan pembersihan sebanyak 3 siklus	Gambar 4.8



Gambar 4.6 Panel Bersih



Gambar 4.7 Daun



Gambar 4.8 Kotoran Burung

Pada kondisi panel bersih, sistem hanya menjalankan proses deteksi tanpa adanya proses pembersihan. Hasil menunjukkan berupa nilai *confidence score*

sebesar 0.57. Hal ini menunjukkan bahwa sistem dapat membedakan antara panel bersih dan panel kotor dengan cukup baik.

Selanjutnya pengujian dengan kondisi daun menutupi sebagian permukaan *solar panel*. Sistem mendeteksi objek sebesar 0.73 yang sudah termasuk tinggi nilai suatu *confidence score*-nya. Bentuk respons sistem terhadap objek daun berupa pengambilan keputusan pembersihan sebanyak 2 siklus. Jumlah siklus pembersihan ini sudah sesuai karena daun termasuk suatu kotoran ringan yang dapat dibersihkan hanya dengan beberapa siklus pembersihan.

Terakhir adalah kondisi permukaan panel yang terdeteksi adanya kotoran burung. Nilai *confidence score* yang didapatkan sebesar 0.61 yang kemudian sistem mengirimkan perintah untuk pembersihan sebanyak 3 siklus karena kotoran burung termasuk dalam kondisi kotor berat.

Dari 3 kondisi pengujian di atas, dapat disimpulkan bahwa sistem dapat bekerja secara otomatis berdasarkan kondisi *real-time* panel tanpa ketergantungan dengan penjadwalan pembersihan. Deteksi berjalan dengan baik dan sistem dapat menentukan kapan kondisi tersebut harus dibersihkan dan berapa kali siklus yang harus dipakai. Sistem dapat berjalan apabila nilai *confidence score* diatas 0.50.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil dan pembahasan pengujian proyek akhir yang berjudul “Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*” dapat diperoleh kesimpulan:

1. Sistem pembersih *solar panel* berhasil dirancang dan diimplementasikan secara otomatis yang dilengkapi dengan sistem pengkondisian. Sistem ini dapat mendeteksi kondisi permukaan *solar panel* menggunakan *webcam* dan memberikan perintah untuk menjalankan sistem pembersihan dari hasil deteksi.
2. Berhasil membangun sistem kontrol menggunakan *Raspberry Pi 4* yang terintegrasi *ESP32* dengan komunikasi *MQTT* melalui jaringan WiFi.
3. Penerapan *image processing* berhasil dengan menggunakan model deteksi objek *YOLOv5*. *Webcam* akan menangkap kondisi permukaan *solar panel*, yang kemudian akan dideteksi berdasarkan nama label dan nilai *confidence score*. Hasil deteksi tersebut kemudian akan dianalisa untuk memilih siklus pembersihan yang akan dilakukan sehingga sistem pembersihan akan dilakukan apabila kondisi panel dalam kondisi kotor.

5.2. Saran

Berdasarkan proses perancangan, pengimplementasian, serta pengujian sistem yang telah dilakukan, penulis memberikan beberapa saran sebagai berikut:

1. Peningkatan akurasi deteksi gambar dengan menambahkan jumlah dataset yang lebih banyak dan beragam agar sistem dapat mengenali pola jenis kotoran dengan lebih akurat.
2. Penggunaan kamera atau webcam dengan resolusi kualitas gambar yang lebih tinggi.
3. Peningkatan versi *Raspberry Pi* dan kapasitas penyimpanannya serta peningkatan versi *YOLO* yang digunakan.

4. Pengintegrasian sistem dengan *IoT* agar dapat melakukan kontrol pembersihan dari jarak jauh.



DAFTAR PUSTAKA

- [1] B. P. Statistik, Kapasitas Terpasang PLN menurut Jenis Pembangkit Listrik (MW) [Online], diakses pada 25 Januari 2025, Available : <https://www.bps.go.id/id>.
- [2] Nurlis Elsa Effendi, Heru Abrianto, A. Darmawan Sidik, “Analisa Pengaruh Kondisi Panel Surya Kotor dengan Panel Surya Bersih Terhadap Energi Listrik yang dihasilkan”, *Jurnal Cahaya Mandalika*, vol. 4, no.1, pp. 1025–1040, 2020.
- [3] Muhammad Rizal Wira Kusuma, Esa Apriaskar, Djunaidi, “Rancang Bangun Sistem Pembersih Otomatis Pada Solar Panel Menggunakan Wiper Berbasis Mikrokontroler”, *Techné: Jurnal Ilmiah Elektroteknika*, vol. 19, no. 01, pp. 23–32, 2020, doi: 10.31358/techne.v19i01.220.
- [4] Fathan Farisi Danial, “Rancang Bangun Sistem Monitoring Kebersihan Lantai Berbasis Pengolahan Citra Menggunakan *Raspberry Pi* yang Terintegrasi Dengan Robot Pembersih Lantai melalui Komunikasi *MQTT*”, Skripsi, Institut Teknologi Nasional, Malang pp. 13–14, 2023.
- [5] Sonya Widyawati Putri, Gaguk Marausna, Erwan Eko Prasetyo, “Analisis Pengaruh Intensitas Cahaya Matahari Terhadap Daya Keluaran Pada Panel Surya”, *Teknika STTKD : Jurnal Teknik, Elektronik. Engine*, vol. 8, no. 1, pp. 29–37, 2022, doi: 10.56521/teknika.v8i1.442.
- [6] Budiyanto dan Hery Setiawan, “Analisa Perbandingan Kinerja Panel Surya Vertikal Dengan Panel Surya Fleksibel pada Jenis *Monocrystalline*”, *Resistor (Elektronika Kendali Telekomunikasi Tenaga Listrik Komputer)*, vol. 4, no. 1, pp. 77-86, 2021, doi: 10.24853/resistor.4.1.77-86.
- [7] Trio Ade Mulyanto, *et al.*, “*Home Automation System* dengan menggunakan *Raspberry Pi 4*”, *Jurnal Ilmiah Digital of Information Technology*, vol. 11, no. 1, pp. 60–73, 2021, doi: 10.55041/ijisrem32505.
- [8] Muhammad Kukuh Isnaen dan Arnisa Stefanie, “Implementasi *Raspberry Pi* dalam Alat Klasifikasi Penyakit Mata dengan Arsitektur *Yolov8* Menggunakan Oftalmoskop”, *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 3, pp. 1885–1889, 2023, doi: 10.36040/jati.v7i3.6950.
- [9] Yosua D. Widiarto, Meicsy E. I. Najoan, Muhammad Dwisnanto Putro, “Sistem Penggerak Robot Beroda *Vacuum Cleaner* Berbasis *Mini Computer Raspberry pi*”, *Jurnal Teknik Elektro dan Komputer*, vol. 7, no. 1, pp. 25–32, 2018.
- [10] Sahwari dan Untung Jasuli, “Pembuatan Pola Tulisan Tangan Melalui Webcam Dengan Metode $L^*a^*b^*$ Color Space,” *JUSTIFY Jurnal Sistem Informasi Ibrahimy*, vol. 1, no. 1, pp. 8–16, 2022, doi: 10.35316/justify.v1i1.1993.
- [11] Julfekar Ali Andre, “Sistem Security Webcam dengan Menggunakan Microsoft Visual Basic (6.0),” *RABIT ; Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 1, no. 2, pp. 48–60, 2016, [Online]. Available: <http://jurnal.univrab.ac.id/index.php/rabit/article/view/23/10>

- [12] Rinaldy, Risa Farrid Christianti, Didi Supriyadi, "Pengendalian Motor Servo yang Terintegrasi dengan Webcam Berbasis Internet dan Arduino", *Jurnal Infotel*, vol. 5, no. 2, pp. 17–23, 2013.
- [13] RD. Kusumanto dan Alan Novi Tomponu, "Pengolahan Citra Digital untuk Mendeteksi Obyek Menggunakan Pengolahan Warna Model Normalisasi RGB", *Seminar Nasional Teknologi Informasi & Komunikasi Terapan*, vol. 17, no. C, pp. 329–332, 2011, doi: 10.1016/S0166-1116(08)71924-1.
- [14] Faris Mushlihul Amin, "Identifikasi Citra Daging Ayam Berformalin Menggunakan Metode Fitur Tekstur dan K-Nearest Neighbor (K-NN)", *Jurnal Matematika "MANTIK"*, vol. 4, no. 1, pp. 68–74, 2018, doi: 10.15642/mantik.2018.4.1.68-74.
- [15] Getsa Novandra Rizkatama, Anan Nugroho, Alfa Faridh Suni, "Sistem Cerdas Penghitung Jumlah Mobil untuk Mengetahui Ketersediaan Lahan Parkir berbasis Python dan YOLO v4", *Edu Komputika Jurnal*, vol. 8, no. 2, pp. 91–99, 2021, doi: 10.15294/edukomputika.v8i2.47865.
- [16] Faizal Indaryanto, Anan Nugroho, Alfa Faridh Suni, "Aplikasi Penghitung Jarak dan Jumlah Orang Berbasis YOLO Sebagai Protokol Kesehatan Covid-19", *Edu Komputika Jurnal*, vol. 8, no. 1, pp. 31–38, 2021, doi: 10.15294/edukomputika.v8i1.47837.
- [17] Primasdika Yunia Putra, *et al.*, "Deteksi Kendaraan Truk pada Video Menggunakan Metode Tiny-YOLO v4", *JIP (Jurnal Informatika Polinema)*, vol. 9, no. 2, pp. 215–222, 2023, doi: 10.33795/jip.v9i2.1243.
- [18] Rudi Kurniawan, Ahmad Taqwa Martadinata, Sandy Dwi Cahyo, "Klasifikasi Tingkat Kematangan Buah Sawit Berbasis Deep Learning dengan Menggunakan Arsitektur Yolov5", *Jurnal of Information System Research (JOSH)*, vol. 5, no. 1, pp. 302–309, 2023, doi: 10.47065/josh.v5i1.4408.
- [19] Fadhli Dzil Ikram, *et al.*, "Deteksi Objek di Lapangan pada Robot Sepakbola Beroda Menggunakan Metode YOLOV5", *Jurnal elkolind*, vol. 11, no. 2, pp. 604–611, 2024, doi: 10.33795/elkolind.v11i2.5235.
- [20] Prince Richard Setiono, Sherwin R. U. A Sompie, Meicsy E. I Najoan, "Aplikasi Pengenalan Wajah untuk Sistem Absensi Kelas berbasis Raspberry Pi", *Jurnal Teknik Informatika*, vol. 15, no. 3, pp. 179–188, 2020.
- [21] Haris Isyanto, Muhammad Azra Komara Batubara, Deni Almada, "Perancangan Alat Pembersih Panel Surya Berbasis Internet of Things", *RESISTOR (Elektronika Kendali Telekomunikasi Tenaga Listrik Komputer)*, vol. 6, no. 2, pp. 125–132, 2023, [Online]. Available: <https://jurnal.umj.ac.id/index.php/resistor/article/view/18823>
- [22] Annisa Zikri, "Rancang Bangun Sistem Penyiraman Tanaman Otomatis berbasis Raspberry Pi 3 dengan Memanfaatkan Thingspeak dan Interface Android sebagai Kendali", Skripsi, UIN Syarif Hidayatullah, Jakarta, 2020.
- [23] Axl Hanuebi, Aherwin Sompie, Feisy Kambey, "Aplikasi Pengenalan Wajah untuk Membuka Pintu berbasis Raspberry Pi", *Jurnal Teknik Informatika*, vol. 14, no. 2, pp. 243–252, 2019.
- [24] Lazuardi Arsy, Oky Dwi Nurhayati, Kurniawan Teguh Martono, "Aplikasi Pengolahan Citra Digital Meat Detection dengan Metode Segmentasi K-

- Mean Clustering* berbasis OpenCV dan Eclipse", *Jurnal Teknologi dan Sistem Komputer*, vol. 4, no. 2, pp. 322-331, 2016, doi: 10.14710/jtsiskom.4.2.2016.322-332.
- [25] Kurniawan Dwi Irianto, "Pendeteksi Gerak berbasis Kamera Menggunakan OpenCV pada Ruangan", *KomuniTi*, vol. 2, no. 1, pp. 52–59, 2010.
- [26] Moh. Wildan Habibi, Adhitya Bhawiyuga, Achmad Basuki, "Rancang Bangun IOT *Cloud Platform* berbasis Protokol Komunikasi MQTT", *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 2, pp. 479–485, 2018, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [27] Periyaldi, Arief Bramanto W.P, Agusma Wajiansyah, "implementasi Sistem Monitoring Suhu Ruang Server Satnetcom Berbasis *Internet Of Things* (IOT) Menggunakan Protokol Komunikasi *Message Queue Telemetry Transport* (MQTT)", *Jurnal Teknologi Terpadu*, vol. 6, no. 1, pp. 23-29, 2018.
- [28] Irna Tri Yuniahastuti, *et al.*, "Optimasi Perancangan *Solar Cell Cleaner* Menggunakan *Wiper*", *PROSIDING: Seminar Nasional Hasil Penelitian dan Pengabdian Kepada Masyarakat LPPM UNIPMA (SNAPMA) Tahun 2023*, pp. 38–45, 2023.
- [29] Jamaluddin Purba, *et al.*, "Perancangan Prototipe Alat Pembersih Panel Surya dengan Sistem Gerak Otomasi", *Journal Kajian Teknik Mesin (JKTM)*, vol. 7, no. 1, pp. 1–8, 2022.
- [30] Muhammad Yusuf Efendi dan Muhammad Haqi Faisal Abidin, "Implementasi Klasifikasi Jenis Kendaraan di Indonesia Menggunakan YOLO", *SNESTIK Seminar Nasional Teknik Elektro, Sistem Informasi, dan Teknik Informatika*, pp. 219–224, 2021.
- [31] Rangga Gelar Guntara, "Deteksi Atap Bangunan Berbasis Citra Udara Menggunakan Google Colab dan Algoritma *Deep Learning YOLOv7*", *Jurnal Manajemen Sistem Informasi (JMASIF)*, vol. 2, no. 1, pp. 9–18, 2023, doi: 10.59431/jmasif.v2i1.156.



LAMPIRAN

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama lengkap : Muhammad Ihsan Solih
Tempat & tanggal lahir : Pangkalpinang, 17 Juli 2004
Alamat rumah : Jl. Jendral Sudirman, Belinyu-
Bangka
Telp: -
Hp: 0813-6808-4228
Email:
muhammadihsansolih17@gmail.com
ail.com
Jenis kelamin : Laki-laki
Agama : Islam



2. Riwayat Pendidikan

SD	MI Al-Huda Belinyu	2016
SMP	SMPN 1 Belinyu	2019
SMA	SMKS YPN Belinyu	2022
D-IV	Politeknik Manufaktur Negeri Bangka Belitung	2022 - sekarang

3. Pendidikan Non-Formal

MBKM Kemendikbudristek Kampus Mengajar Angkatan 8	2024
---	------

Sungailiat, 04 Juli 2025

Muhammad Ihsan Solih

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama lengkap : Syair Aulia
Tempat & tanggal lahir : Belinyu, 11 Desember 2004
Alamat rumah : Desa Gunung Muda, Belinyu
Telp: -
Hp: 0823-7570-4177
Email:
syairaulia127@gmail.com



Jenis kelamin : Perempuan
Agama : Islam

2. Riwayat Pendidikan

SD	SDN 10 Belinyu	2016
SMP	SMPN 2 Belinyu	2019
SMA	SMAN 1 Belinyu	2022
D-IV	Politeknik Manufaktur Negeri Bangka Belitung	2022- sekarang

3. Pendidikan Non-Formal

MBKM Mandiri Asistensi Mengajar	2024
---------------------------------	------

Sungailiat, 04 Juli 2025



Syair Aulia

Lampiran 2

Program Pengiriman Data dari Raspberry ke ESP32 menggunakan Metode MQTT

```
import cv2
import torch
import paho.mqtt.client as mqtt
# Load YOLOv5 model
model = torch.hub.load('ultralytics/yolov5', 'custom', path='best.pt') # Ganti path
model kamu
# MQTT setup
mqtt_broker = "192.168.151.39" # Ganti dengan IP broker MQTT kamu
client = mqtt.Client()
client.connect(mqtt_broker, 1883, 60)
# Fungsi klasifikasi
def klasifikasi(label):
    if label in ['kotoran burung', 'debu berat']:
        return "BERAT"
    elif label in ['daun', 'debu sedang']:
        return "SEDANG"
    elif label == 'debu ringan':
        return "RINGAN"
    return None
# Jalankan webcam
cap = cv2.VideoCapture(0)
while True:
    ret, frame = cap.read()
    if not ret:
        continue
    results = model(frame)
    labels = results.names
```

```

kelas_terdeteksi = results.xyxy[0][:-1].tolist()
hasil_klasifikasi = set()
for idx in kelas_terdeteksi:
    label = labels[int(idx)]
    klas = klasifikasi(label)
    if klas:
        hasil_klasifikasi.add(klas)
# Prioritaskan klasifikasi: BERAT > SEDANG > RINGAN
prioritas = ["BERAT", "SEDANG", "RINGAN"]
dikirim = None
for p in prioritas:
    if p in hasil_klasifikasi:
        dikirim = p
        break

if dikirim:
    print("Mengirim ke ESP32:", dikirim)
    client.publish("panel/kotoran", dikirim)
# Tampilkan hasil deteksi
frame = results.render()[0]
cv2.imshow("Deteksi Kotoran", frame)
if cv2.waitKey(1) == ord('q'):
    break
cap.release()
cv2.destroyAllWindows()
client.disconnect()

```

Program kontrol ESP32

```
#include <Wire.h>
#include <RTCLib.h>
#include <WiFi.h>
#include <PubSubClient.h>
RTC_DS3231 rtc;
// Water pump & DC motor (L298N #2)
const int PUMP_IN = 32;
const int PUMP_EN = 19;
const int DC_IN1 = 25;
const int DC_IN2 = 26;
const int DC_EN = 18;
// Stepper via A4988
const int STEP_PIN = 13;
const int DIR_PIN = 12;
// Limit switch & buzzer
const int LIMIT_SW = 4;
const int BUZZER = 23;
// Durasi
const unsigned long PUMP_DURATION = 5000;
const unsigned long STEPPER_BACK_TIME = 17500;
const unsigned long BUZZER_DURATION = 2000;

unsigned long lastDisplay = 0;
// MQTT
const char* ssid = "OPPOA58";
const char* password = "113333555555";
const char* mqtt_server = "192.168.151.39";
WiFiClient espClient;
PubSubClient client(espClient);
String klasifikasi = "";
void setup_wifi() {
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }
}
void callback(char* topic, byte* message, unsigned int length) {
  klasifikasi = "";
  for (int i = 0; i < length; i++) {
    klasifikasi += (char)message[i];
  }
  Serial.println("Klasifikasi diterima: " + klasifikasi);
}
```

```

void reconnect() {
  while (!client.connected()) {
    Serial.print("Menghubungkan MQTT...");
    if (client.connect("ESP32Client")) {
      Serial.println(" TERHUBUNG!");
      client.subscribe("panel/kotoran");
      Serial.println("Subscribe ke: panel/kotoran");
    } else {
      Serial.print("Gagal, rc=");
      Serial.print(client.state());
      Serial.println(" coba lagi dalam 5 detik");
      delay(5000);
    }
  }
}

void setup() {
  Serial.begin(115200);
  delay(2000);
  Wire.begin();
  if (!rtc.begin()) {
    Serial.println("RTC tidak ditemukan!");
    while (1);
  }
  setup_wifi();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
  pinMode(PUMP_IN, OUTPUT);
  pinMode(PUMP_EN, OUTPUT);
  pinMode(DC_IN1, OUTPUT);
  pinMode(DC_IN2, OUTPUT);
  pinMode(DC_EN, OUTPUT);
  pinMode(STEP_PIN, OUTPUT);
  pinMode(DIR_PIN, OUTPUT);
  pinMode(LIMIT_SW, INPUT_PULLUP);
  pinMode(BUZZER, OUTPUT);
  // Matikan semua output
  digitalWrite(PUMP_IN, LOW);
  analogWrite(PUMP_EN, 0);
  digitalWrite(DC_IN1, LOW);
  digitalWrite(DC_IN2, LOW);
  analogWrite(DC_EN, 0);
  digitalWrite(STEP_PIN, LOW);
  digitalWrite(DIR_PIN, LOW);
  digitalWrite(BUZZER, LOW);
  Serial.println("Sistem siap. Menunggu data MQTT (RINGAN/SEDANG/BERAT)...");
}

```

```

}
void loop() {
  if (!client.connected()) reconnect();
  client.loop();
  DateTime now = rtc.now();
  tampilkanWaktu(now);
  if (klasifikasi == "RINGAN") {
    Serial.println("Klasifikasi RINGAN, tidak menjalankan sistem.");
    klasifikasi = "";
  } else if (klasifikasi == "SEDANG") {
    jalankanSistem(2);
    klasifikasi = "";
  } else if (klasifikasi == "BERAT") {
    jalankanSistem(3);
    klasifikasi = "";
  }
}
}
void tampilkanWaktu(const DateTime &t) {
  if (millis() - lastDisplay >= 1000) {
    lastDisplay = millis();
    Serial.printf("RTC: %02d:%02d:%02d\n", t.hour(), t.minute(), t.second());
  }
}
}
void jalankanSistem(int jumlahSiklus) {
  unsigned long start;
  const int STEP_DELAY_US = 800;
  // 1) Pompa
  Serial.println("Pompa ON");
  digitalWrite(PUMP_IN, HIGH);
  analogWrite(PUMP_EN, 255);
  start = millis();
  while (millis() - start < PUMP_DURATION) {
    tampilkanWaktu(rtc.now());
    delay(50);
  }
  analogWrite(PUMP_EN, 0);
  digitalWrite(PUMP_IN, LOW);
  Serial.println("Pompa OFF");
  // 2) Siklus stepper + DC
  for (int c = 1; c <= jumlahSiklus; c++) {
    Serial.printf("=== Siklus ke-%d ===\n", c);

    // MAJU
    Serial.println("Stepper & DC MAJU...");
    digitalWrite(DIR_PIN, HIGH);
    digitalWrite(DC_IN1, HIGH);
  }
}
}

```

```

digitalWrite(DC_IN2, LOW);
analogWrite(DC_EN, 255);
while (digitalRead(LIMIT_SW) == HIGH) {
  digitalWrite(STEP_PIN, HIGH);
  delayMicroseconds(STEP_DELAY_US);
  digitalWrite(STEP_PIN, LOW);
  delayMicroseconds(STEP_DELAY_US);
  tampilkanWaktu(rtc.now());
}
analogWrite(DC_EN, 0);
digitalWrite(DC_IN1, LOW);
digitalWrite(DC_IN2, LOW);
Serial.println("Limit switch ditekan → STOP maju");
// MUNDUR
Serial.println("Stepper & DC MUNDUR (17.5 detik)...");
digitalWrite(DIR_PIN, LOW);
digitalWrite(DC_IN1, LOW);
digitalWrite(DC_IN2, HIGH);
analogWrite(DC_EN, 255);
start = millis();
while (millis() - start < STEPPER_BACK_TIME) {
  digitalWrite(STEP_PIN, HIGH);
  delayMicroseconds(STEP_DELAY_US);
  digitalWrite(STEP_PIN, LOW);
  delayMicroseconds(STEP_DELAY_US);
  tampilkanWaktu(rtc.now());
}
analogWrite(DC_EN, 0);
digitalWrite(DC_IN1, LOW);
digitalWrite(DC_IN2, LOW);
Serial.println("Stepper & DC OFF");
}
// 3) Buzzer
Serial.println("Buzzer ON (2 detik)...");
digitalWrite(BUZZER, HIGH);
start = millis();
while (millis() - start < BUZZER_DURATION) {
  tampilkanWaktu(rtc.now());
  delay(50);
}
digitalWrite(BUZZER, LOW);
Serial.println("Buzzer OFF, selesai.");
}

```

Lesta Lesta

SISTEM_PEMBERSIH_SOLAR_PANEL_BERBASIS_MICROCON...

 Mikroba

 Mikroba

 Politeknik Manufaktur Negeri Bangka Belitung

Document Details

Submission ID

trn:oid::1:3289912720

Submission Date

Jul 4, 2025, 9:32 AM GMT+7

Download Date

Jul 4, 2025, 9:35 AM GMT+7

File Name

SISTEM_PEMBERSIH_SOLAR_PANEL_BERBASIS_MICROCONTROLLER_DAN_IMAGE_PROCESSING.docx

File Size

9.6 MB

47 Pages

6,756 Words

43,684 Characters

11% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Top Sources

- 10%  Internet sources
- 4%  Publications
- 3%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Top Sources

- 10% Internet sources
- 4% Publications
- 3% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	download.garuda.kemdikbud.go.id	<1%
2	Internet	jip.polinema.ac.id	<1%
3	Internet	repository.polman-babel.ac.id	<1%
4	Internet	core.ac.uk	<1%
5	Internet	docplayer.info	<1%
6	Internet	repository.dinamika.ac.id	<1%
7	Student papers	Universitas Brawijaya	<1%
8	Internet	123dok.com	<1%
9	Publication	Petronela Kurniati Kondang, Nika Wirana, Defi Veronika, Dealmus, Thesa Anan...	<1%
10	Internet	repo.undiksha.ac.id	<1%
11	Internet	repository.its.ac.id	<1%

12	Internet	id.123dok.com	<1%
13	Internet	openlibrarypublications.telkomuniversity.ac.id	<1%
14	Internet	journal.um-surabaya.ac.id	<1%
15	Internet	text-id.123dok.com	<1%
16	Internet	www.scribd.com	<1%
17	Student papers	Sriwijaya University	<1%
18	Internet	eprints.upnjatim.ac.id	<1%
19	Internet	ejournal.upbatam.ac.id	<1%
20	Internet	es.scribd.com	<1%
21	Student papers	Universitas Islam Indonesia	<1%
22	Internet	otodriver.com	<1%
23	Internet	www.labelektronika.com	<1%
24	Student papers	SDM Universitas Gadjah Mada	<1%
25	Internet	afirianto.lecture.ub.ac.id	<1%

26	Internet	repo.darmajaya.ac.id	<1%
27	Internet	de.slideshare.net	<1%
28	Publication	Dedi Tri Laksono, Rien Afrianti, Mira Wellya Fatma, Maresa Prasafitri, Hamdi Alch...	<1%
29	Publication	Yori Saepul Barki, Teguh Ikhlas Ramadhan, Agus Supriatman. "Design of A Soil M...	<1%
30	Internet	dspace.uii.ac.id	<1%
31	Internet	habibsaputro.blogspot.com	<1%
32	Internet	mahfudage.wordpress.com	<1%
33	Internet	pdfcoffee.com	<1%
34	Internet	www.coursehero.com	<1%
35	Internet	carirp.blogspot.com	<1%
36	Internet	eprints.uty.ac.id	<1%
37	Internet	publikasi.dinus.ac.id	<1%
38	Internet	putuadisusanta.wordpress.com	<1%
39	Internet	repository.ummy.ac.id	<1%

40	Internet	repository.usd.ac.id	<1%
41	Internet	www.forensor.com	<1%
42	Internet	ejournal.unsrat.ac.id	<1%
43	Internet	mardaelyshinta.blogspot.com	<1%
44	Internet	media.neliti.com	<1%
45	Internet	www.asrinbau.com	<1%
46	Internet	www.ilmuskripsi.com	<1%
47	Internet	www.slideshare.net	<1%
48	Internet	ojs.umrah.ac.id	<1%
49	Publication	Hotma Pangaribuan, Sunarsan Sitohang. "Peningkatan Kualitas Deteksi Tepi den...	<1%
50	Publication	Yudha Arya Prasaja. "PERBANDINGAN METODE GLCM DAN LBP DALAM KLASIFIKA...	<1%

SURAT PERNYATAAN

Saya/Kami yang bertandatangan dibawah ini telah menyelesaikan Proyek Akhir yang berjudul:
Sistem Pembersih *Solar Panel* berbasis *Microcontroller* dan *Image Processing*

Oleh :

1. Muhammad Ihsan Solih /NPM 1052250
2. Syair Aulia /NPM 1052258

Dengan ini menyatakan bahwa isi laporan akhir proyek akhir sama dengan *hardcopy*.

Demikian surat pernyataan ini dibuat dengan sebenar-benarnya.

Sungailiat, 15 September 2025

1. Muhammad Ihsan Solih ()
2. Syair Aulia ()

Mengetahui,

Pembimbing 1


(Zanu Saputra, M.Tr.T.)

Pembimbing 2


(Lesta, S.P, M.Si.)