

**PENGENDALIAN PENDINGIN RUANGAN (*AIR CONDITIONER*)
BERDASARKAN DETEKSI JUMLAH MANUSIA
MENGUNAKAN METODE YOLO-V7
PADA ORANGE PI 5**

SKRIPSI

**FARHAN YEHANDA
201402137**



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

PENGENDALIAN PENDINGIN RUANGAN (*AIR CONDITIONER*)
BERDASARKAN DETEKSI JUMLAH MANUSIA
MENGUNAKAN METODE YOLO-V7
PADA ORANGE PI 5

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah
Sarjana Teknologi Informasi

FARHAN YEHANDA
201402137



PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024

PERSETUJUAN

Judul : Pengendalian Pendingin Ruangan (*Air Conditioner*)
 Berdasarkan Deteksi Jumlah Manusia Menggunakan
 Metode YOLOv7 Pada Orange Pi 5.

Kategori : Skripsi

Nama : Farhan Yehanda

Nomor Induk Mahasiswa : 201402137

Program Studi : Teknologi Informasi

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
 INFORMASI UNIVERSITAS SUMATERA UTARA

Medan, 11 Juli 2024

Komisi Pembimbing :

Pembimbing 2

Dr. Mohammad Andri Budiman, S.T.,
 M.Comp.SC., M.E.M

Pembimbing 1

Seniman, S.Kom M.Kom

Diketahui/disetujui oleh

Program Studi S1 Teknologi Informasi

Ketua,



Dedy Arisman, S.Kom.

NIP. 19908312009121002

PERNYATAAN

PENGENDALIAN PENDINGIN RUANGAN (*AIR CONDITIONER*)
BERDASARKAN DETEKSI JUMLAH MANUSIA
MENGUNAKAN METODE YOLO-V7
PADA ORANGE PI 5

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 11 Juli 2024

Farhan Yehanda

201402137

UCAPAN TERIMAKASIH

Segala puji bagi Allah SWT yang telah melimpahkan berkah dan rahmat-Nya kepada penulis sehingga penulis dapat menyelesaikan skripsi ini sebagai persyaratan untuk memperoleh gelar Sarjana Komputer Teknologi pada Program Studi S1 Teknologi Informasi di Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Pada Kesempatan kali ini penulis ingin menyampaikan rasa terima kasih kepada seluruh pihak yang telah memberikan dukungan, doa dan semangat kepada penulis. Penulis ingin menyampaikan terima kasih kepada:

1. Keluarga penulis, Ibu Yenni Adriyani dan Bapak Burhanuddin yang selalu memberikan semangat, doa dan dukungan kepada penulis.
2. Bapak Dr. Muryanto Amin, S.Sos, M.Si., yang merupakan Rektor Universitas Sumatera Utara.
3. Ibu Dr. Maya Silvi Lydia, B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara
4. Bapak Dedy Arisandi, S.T., M.Kom. selaku Ketua Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
5. Bapak Ivan Jaya, S.Si., M.Kom. selaku Sekretaris Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
6. Bapak Seniman, S.Kom, M.Kom. selaku Dosen Pembimbing I penulis yang telah memberikan banyak masukan, bantuan, bimbingan serta meminjamkan alat dalam membantu proses penelitian penulis.
7. Bapak Dr. Mohammad Andri Budiman, S.T., M.Comp.SC., M.E.M selaku Dosen Pembimbing II penulis yang telah memberikan masukkan, saran, dan motivasi serta meluangkan waktunya kepada penulis.
8. Bapak Dr. Muhammad Anggia Muchtar, ST., MMIT selaku Dosen Penguji I yang telah memberikan saran dan masukan untuk perbaikan skripsi penulis.
9. Bapak Fahrurrozi Lubis, B.IT., M.Sc.IT selaku Dosen Penguji II yang telah menguji, memberikan saran, dan masukan untuk perbaikan skripsi penulis.

10. Seluruh Bapak/Ibu Dosen dan staff pegawai Program Studi Teknologi Informasi yang telah membimbing dan membantu selama masa perkuliahan penulis.
11. Siti Khadijah yang sudah menemani penulis dari awal perkuliahan sampai saat ini. Selalu menjadi tempat bercerita tentang keluh kesah dan mendukung penulis dalam keadaan apapun.
12. Teman-teman seperjuangan selama proses penelitian penulis Gideon Tulus Hatta Yudha Siahaan, Reza Yudhistira Ramadhana, Mustafa Kamal Nasution, dan Galileo Gulampati Marapratama Purba, yang selalu membantu penulis ketika kesulitan dalam mengerjakan skripsi, memberikan doa dan semangat pada proses penelitian hingga pada penyelesaian skripsi ini.
13. Teman-teman penulis dari KOM B khususnya Muhammad Iqbal, Gembira Raymondo, Franda Christiano, Muhammad Ridwan Rizki, Raihan Alifya, Hizkia Winter, dan Reinhart Domingo.
14. Kepada rekan-rekan angkatan 2020 dari program studi Teknologi Informasi Universitas Sumatera Utara yang tidak dapat disebutkan satu persatu namanya, yang telah memberikan semangat sehingga penulis dapat menyelesaikan skripsi ini.

Penulis mengakui bahwa skripsi ini masih belum sempurna dan masih terdapat banyak kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang berguna untuk menyempurnakan skripsi ini. Penulis juga berharap bahwa hasil penelitian ini dapat memberikan manfaat kepada semua pihak yang membacanya.

Medan, 11 Juli 2024

Farhan Yehanda

ABSTRAK

Penelitian ini berujuan untuk mengembangkan sistem pengendalian pendingin ruangan (*Air Conditioner*) secara otomatis berdasarkan deteksi jumlah manusia menggunakan metode YOLOv7 pada *single board computer* Orange Pi 5. Latar belakang dari penelitian ini adalah kebutuhan untuk mengoptimalkan penggunaan AC di ruangan umum seperti di ruang kelas, dan ruang meeting yang sering digunakan secara tidak efisien. Metode yang digunakan dalam penelitian ini meliputi penggunaan Model pre-trained YOLOv7 yang telah dilatih dengan dataset COCO dan difilter untuk hanya mendeteksi manusia dari gambar yang diambil setiap 10 menit menggunakan IP Kamera. Sistem mengirimkan perintah dari Orange Pi 5 ke Arduino UNO untuk mengontrol operasi AC, termasuk menghidupkan, mematikan dan mengatur suhu AC berdasarkan deteksi jumlah orang yang terdeteksi di ruangan. Pengaturan suhu dilakukan sesuai dengan standar yang ditetapkan oleh Keputusan Menteri Kesehatan Republik Indonesia Nomor 1405/Menkes/SK/XI/2002, yang merekomendasikan suhu ruangan berkisar antara 23 °C hingga 26 °C. Hasil Penelitian menunjukkan bahwa model pre-trained YOLOv7 dapat mendeteksi manusia dengan baik. Sistem ini juga berhasil mengontrol suhu AC berdasarkan jumlah orang yang terdeteksi dan menyesuaikan suhu kembali ketika suhu ruangan lebih besar dari suhu AC. Selain itu, AC akan dimatikan ketika sistem tidak mendeteksi keberadaan manusia di ruangan dan melewati batas waktu yang ditentukan.

Kata Kunci: AC, Manusia, YOLOv7

ABSTRACT

This study aims to develop an automatic air conditioning (AC) control system based on human detection using the YOLOv7 method on a single-board computer, Orange Pi 5. The background of this research is need to optimize AC usage in public spaces such as classrooms and meeting rooms, which are often used inefficiently. The methods used in this study include the utilization of a pre-trained YOLOv7 model trained on the COCO dataset and filtered specifically to detect humans from images captured every 10 minutes using an IP Camera. The system sends commands from the Orange Pi 5 to an Arduino UNO to control the AC operation, including turning it on, off and adjusting the temperature based on the number of people detected in the room. Temperature settings are made according to the standards set by the Indonesian Minister of Health Decree No. 1405/Menkes/SK/XI/2992, which recommends room temperature ranging from 23 ° to 26 ° C. The research results show that pre-trained YOLOv7 model effectively detects humans. This system also successfully controls the AC automatically based on the detected number of people detected and adjust it when the room temperature exceeds the AC setting. Moreover the AC is turned off when the system detect no people in the room beyond a specified time limit.

Keyword: AC, People, YOLOv7

DAFTAR ISI

PERSETUJUAN	i
PERNYATAAN.....	ii
UCAPAN TERIMAKASIH.....	iii
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah.....	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	3
1.7 Sistematika Penulisan	4
BAB 2 LANDASAN TEORI	6
2.1 <i>Air Conditioner (AC)</i>	6
2.2 Sistem Kerja AC.....	6
2.3 <i>Single Board Computer</i>	7
2.4 Orange Pi 5.....	7
2.5 Arduino	8
2.5.1 Arduino UNO	8
2.6 <i>Universal Asynchronous Receiver Transmitter (UART)</i>	9
2.7 DHT22	9
2.8 IR <i>Receiver</i>	10
2.9 IR <i>Transmitter</i>	10
2.10 Real Time Streaming Protocol (RTSP)	10
2.11 <i>Computer Vision</i>	11
2.12 <i>Object Detection</i>	11

2.13	Deep Learning.....	11
2.14	Convolutional Neural Networks (CNN)	12
2.15	You Only Look Once (YOLO).....	12
2.16	YOLOv7.....	14
2.17	Penelitian Terdahulu.....	15
2.18	Perbedaan Penelitian	17
BAB 3 ANALISIS DAN PERANCANGAN SISTEM.....		19
3.1	Cloning Repository YOLOv7	19
3.2	Penginstallan Requirement	20
3.3	Arsitektur Umum	20
3.3.1	Input	21
3.3.2	Post Processing.....	22
3.3.2.1	Class Filtering	22
3.3.2.2	Counting People.....	22
3.3.3	Deployment	23
3.3.4	Inference.....	23
3.3.5	Decoding	24
3.3.6	Output.....	25
3.4	Rancangan Perangkat Keras.....	29
BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM.....		31
4.1	Perangkat Keras dan Perangkat Lunak	31
4.2	Pengimplementasian Model Pretrained YOLOv7.....	31
4.3	Implementasi Perangkat Keras.....	32
4.3.1	Koneksi Orange Pi 5 dengan Arduino UNO	33
4.3.2	Konfigurasi IP Kamera.....	35
4.3.3	Koneksi Arduino UNO dengan DHT22	38
4.3.4	Koneksi Arduino UNO dengan IR Receiver	40
4.3.5	Koneksi Arduino UNO dengan IR Transmitter.....	41
4.4	Pengujian Sistem.....	41
4.4.1	Pengujian IR Receiver.....	41
4.4.2	Pengujian DHT22.....	42
4.4.3	Pengujian Deteksi Manusia.....	43
4.4.4	Pengujian AC.....	47

BAB 5 KESIMPULAN DAN SARAN.....	57
5.1 Kesimpulan	57
5.2 Saran.....	57
DAFTAR PUSTAKA	59

DAFTAR TABEL

Tabel 2.1 Fungsi Pin DHT22.....	9
Tabel 2.2 Perbandingan Beberapa Model YOLOv7.....	14
Tabel 2.3 Penelitian Terdahulu	16
Tabel 3.1 Tabel Perintah ke Arduino	26
Tabel 4.1 Hasil Pengujian Pendeteksian Manusia	43
Tabel 4.2 Spesifikasi AC	47

DAFTAR GAMBAR

Gambar 2.1 Sistem Kerja AC (Golwa et al., 2023)	6
Gambar 2.2 Orange Pi 5 (Wiki-Orange Pi Contributors, 2023).	8
Gambar 2.3 Arduino UNO	8
Gambar 2.4 DHT22	9
Gambar 2.5 IR <i>Receiver</i>	10
Gambar 2.6 IR Transmitter.....	10
Gambar 2.7 Arsitektur CNN.....	12
Gambar 2.8 Mekanisme Arsitektur YOLO.....	13
Gambar 2.9 Arsitektur YOLOv7	15
Gambar 3.1 Proses <i>Cloning</i> Github WongKinYiu.....	19
Gambar 3.2 Isi Direktori Repository YOLOv7	19
Gambar 3.3 Arsitektur Umum	20
Gambar 3.4 Daftar <i>Class</i> pada Dataset COCO	22
Gambar 3.5 E-LAN	24
Gambar 3.6 Rancangan Perangkat Keras	29
Gambar 4.1 Daftar <i>Class</i> pada Dataset COCO	31
Gambar 4.2 Implementasi Perangkat Keras	33
Gambar 4.3 Daftar GPIO Orange Pi 5 yang Telah Aktif.....	34
Gambar 4.4 Pin Orange Pi 5.....	34
Gambar 4.5 Koneksi Arduino dengan Orange Pi 5	35
Gambar 4.6 EYESEC IP Kamera	35
Gambar 4.7 Konfigurasi pada DVR untuk Mencari Alamat IP.....	36
Gambar 4.8 Memastikan Layanan dan Port pada IP Kamera.....	36
Gambar 4.9 Pengaturan Koneksi Jaringan IP Kamera pada Aplikasi Icsee	37
Gambar 4.10 Pengaturan URL RTSP pada IP Kamera.....	38
Gambar 4.11 Tampilan IP Kamera	38
Gambar 4.12 Koneksi Arduino UNO dengan DHT22	39
Gambar 4.13 Denah Pemasangan Sensor Suhu DHT22	39
Gambar 4.14 Implementasi Pemasangan 2 Sensor DHT22	40
Gambar 4.15 Koneksi Arduino UNO dengan IR Receiver VS1838	40
Gambar 4.16 Koneksi Arduino UNO dengan IR Transmitter	41

Gambar 4.17 Proses Pengambilan Data Remote AC.....	42
Gambar 4.18 Data Pada Tombol 27.....	42
Gambar 4.19 Pengujian DHT22 Menggunakan DHTtester.ino	43
Gambar 4.20 Ruangan dalam Kondisi Sedikit	48
Gambar 4.21 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Sedikit	48
Gambar 4.22 Rata-Rata Suhu Aktual setelah Penyesuaian Suhu pada Ruangan dalam Kondisi Sedikit	49
Gambar 4.23 Ruangan dalam Kondisi Sedang.....	50
Gambar 4.24 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Sedang.....	50
Gambar 4.25 Rata-Rata Suhu Aktual setelah Penyesuaian Suhu pada Ruangan dalam Kondisi Sedang.....	51
Gambar 4.26 Ruangan dalam Kondisi Banyak	51
Gambar 4.27 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Banyak	52
Gambar 4.28 Rata-Rata Suhu Aktual Setelah Penyesuaian Suhu pada Ruangan dalam Kondisi Banyak	53
Gambar 4.29 Ruangan dalam Kondisi Lebih dari 17 Orang	53
Gambar 4.30 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Lebih dari 17 Orang	54
Gambar 4.31 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Lebih dari 17 Orang	55
Gambar 4.32 Pengujian Ketika Waktu Sudah Melebihi Batas yang Ditentukan dan Kamera Tidak Mendeteksi Orang di Ruangan	55

BAB 1

PENDAHULUAN

1.1 Latar Belakang

AC (*Air Conditioner*) merupakan suatu alat yang berfungsi untuk mendinginkan ruangan dengan cara memindahkan panas dari dalam ke luar ruangan. Menurut Keputusan Menteri Kesehatan Republik Indonesia Nomor 1405/Menkes/SK/XI/2002 mengenai Persyaratan Lingkungan Kerja Perkantoran dan Industri, bahwa persyaratan udara ruangan yang baik memiliki *range* suhu berkisar $23^{\circ}\text{ C} - 26^{\circ}\text{ C}$. Ketika suhu udara melebihi 26° C , maka diperlukan penggunaan perangkat penyejuk udara seperti *Air Conditioner* (AC).

Di Indonesia, penggunaan AC sudah sangat umum, tidak hanya di lingkungan pribadi seperti di rumah, tetapi juga di tempat umum seperti ruang kelas, ruang makan, dan area publik lainnya. Penggunaan AC dapat memberikan kenyamanan dan meningkatkan produktivitas bagi penggunanya. (Zakiyah et al., 2018).

Seiring meluasnya penggunaan AC, terdapat kesalahan umum yang sering dilakukan oleh pengguna, salah satunya adalah kurangnya kesadaran untuk mematikan AC setelah ruangan tidak lagi digunakan pada saat diluar jam kerja (Fetra, 2020). Selain itu, pengguna terkadang secara langsung menghidupkan AC pada suhu yang sangat rendah dan membiarkan AC pada suhu tersebut, kebiasaan seperti ini mengakibatkan AC bekerja sangat keras untuk mencapai suhu yang diinginkan pengguna dan dapat menyebabkan pemborosan energi listrik (Widura et al., 2023).

Dalam mengatasi permasalahan ini, diperlukan sebuah sistem yang dapat mengendalikan AC secara otomatis berdasarkan deteksi manusia di ruangan. Sistem juga akan dirancang untuk mengatur jadwal operasional AC, sehingga hanya menyala dan mati sesuai pengaturan jadwal yang telah ditentukan. Sistem ini dapat membantu

meningkatkan efisiensi energi listrik dan mengurangi potensi kesalahan pengguna dalam menggunakan AC.

Untuk mendeteksi manusia, salah satu model yang dapat digunakan adalah YOLO (*You Only Look Once*). Model YOLO merupakan model yang terbukti cepat untuk mendeteksi objek secara cepat (Ardiansyah et al., 2023). Seperti yang dilakukan oleh Mileanasari et al. (2023), peneliti mengembangkan sebuah sistem pengendalian AC yang menggunakan model *You Only Look Once* (YOLO) V5 dalam pengolahan citra untuk menghitung jumlah yang orang yang diterapkan pada SBC Nvidia Jetson Nano. Hasilnya sistem yang dirancang berhasil mengendalikan suhu AC berdasarkan deteksi jumlah orang dan membuat penurunan konsumsi energi listrik di laboratorium sebesar 17.13%.

Orange Pi 5 adalah *single board computer* yang sejenis dengan Raspberry Pi, Nano Pi, dan jenis Pi lainnya (Cepi Muh et al., 2022). Orange Pi 5 menggunakan prosesor ARM 64-bit *octa core*, *Internet of Things* (IOT) dan lain-lain (Wiki-Orange Pi Contributors, 2023). Dengan processor seperti ini orange pi 5 mempunyai spesifikasi yang mumpuni untuk menjalankan *task* seperti kecerdasan buatan, cloud computing dan lain-lain.

Penelitian yang dilakukan oleh Bin Tuan Rashid et al., (2023) membahas tentang performa algoritma deteksi objek pada *Single Board Computer* (SBC) Nvidia Jetson Nano dengan membandingkan tiga algoritma deteksi objek yaitu, Faster R-CNN, SSD dan YOLOv7. Hasil penelitian menunjukkan bahwa YOLOv7 menampilkan FPS yang tinggi, waktu inferensi yang rendah, serta akurasi yang tinggi ketika diterapkan ke dalam SBC.

Berdasarkan penjelasan diatas penulis mengangkat judul “Pengendalian Pendingin Ruangan (*Air Conditioner*) Berdasarkan Deteksi Jumlah Manusia Menggunakan Metode Yolov7 Pada Orange Pi 5”.

1.2 Rumusan Masalah

Penggunaan AC di Indonesia telah menjadi kebutuhan umum untuk menciptakan lingkungan yang nyaman dan produktif. Namun, penggunaan AC yang tidak efisien, menyebabkan pemborosan energi listrik yang sia-sia. Oleh karena itu, diperlukan sebuah sistem pengendalian AC otomatis berdasarkan deteksi jumlah ruangan serta mengatur jadwal operasional AC digunakan untuk mengoptimalkan penggunaan AC.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk membangun sistem pengendalian pendingin ruangan (*Air Conditioner*) secara otomatis berdasarkan deteksi jumlah manusia menggunakan metode YOLOv7 pada single board computer Orange Pi 5.

1.4 Batasan Masalah

1. Data input adalah data real time dengan menggunakan protocol RTSP (*Real Time Streaming Protocol*).
2. Model yang digunakan adalah *pre-trained* YOLO versi 7 yang dilatih menggunakan dataset COCO.
3. Standar penentuan suhu ruangan berdasarkan standar Keputusan Menteri Kesehatan Republik Indonesia Nomor 1405/Menkes/SK/XI/2002.
4. Sistem hanya dapat mengontrol satu *Air Conditioner*.

1.5 Manfaat Penelitian

1. Dapat mengoptimalkan penggunaan *Air Conditioner*.
2. Dapat menjadi referensi penelitian dalam menerapkan YOLOv7 pada *single board computer* Orange Pi 5.

1.6 Metodologi Penelitian

Tahapan-tahapan yang akan dilakukan dalam penelitian ini adalah sebagai berikut:

1. Studi Literatur

Studi literatur merupakan tahap untuk mengumpulkan beberapa referensi yaitu berupa jurnal, artikel dan sumber bacaan lainnya mengenai AC, dan penerapan metode *You Only Look Once* (YOLO) versi 7 pada orange pi 5.

2. Analisis Permasalahan

Tahap ini merupakan tahap melakukan analisis terhadap metode *You Only Look Once* versi 7 dan pengontrolan *Air Conditioner* dari referensi yang telah dikumpulkan untuk mendapatkan pemahaman serta masalah yang terjadi dan cara menyelesaiakannya.

3. Perancangan Sistem

Tahap berikutnya adalah melakukan proses perancangan sistem berdasarkan hasil analisis studi literatur, untuk memastikan bahwa alat yang dikembangkan sesuai sasaran dan mudah digunakan.

4. Implementasi

Selanjutnya akan dilakukan implementasi dalam bentuk kode program, perakitan hardware berdasarkan analisis dan perancangan sistem yang telah dilakukan sebelumnya.

5. Pengujian

Pada tahap ini menguji sistem yang telah dibuat menggunakan metode YOLOv7 untuk mendekripsi manusia, menghitung jumlahnya dan mengatur *Air Conditioner* berdasarkan deteksi jumlah manusia.

6. Penyusunan Laporan

Akhirnya dilakukan dokumentasi dan penyusunan laporan berdasarkan hasil evaluasi dari penelitian yang telah dilakukan.

1.7 Sistematika Penulisan

Berikut di bawah ini terdapat 5 bagian sistematika penulisan skripsi yaitu:

BAB 1. PENDAHULUAN

Pada Bab 1, meliputi latar belakang penelitian, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian serta sistematika penulisan pada penelitian.

BAB 2. LANDASAN TEORI

Pada Bab 2, memuat penjelasan landasan teori yang digunakan pada penelitian. Topik yang dibahas meliputi *Air Conditioner*, Orange Pi 5, dan metode *You Only Look Once* versi 7, serta penelitian terdahulu yang digunakan sebagai acuan di dalam penelitian ini.

BAB 3. ANALISIS DAN PERANCANGAN SISTEM

Pada Bab 3, menjelaskan arsitektur umum dari metode yang digunakan untuk penelitian dan perancangan perangkat keras.

BAB 4. IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada Bab 4, menjelaskan tentang proses implementasi dari sistem yang telah dibangun dan dilakukan analisis dari hasil percobaan yang dilakukan untuk mengevaluasi kinerja sistem.

BAB 5. KESIMPULAN DAN SARAN

Pada Bab 5, menjelaskan tentang kesimpulan dan saran dari penelitian yang telah dilakukan agar bermanfaat untuk penelitian selanjutnya.

BAB 2

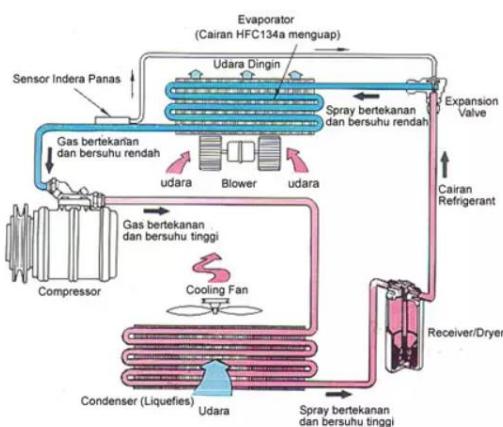
LANDASAN TEORI

2.1 *Air Conditioner (AC)*

Air Conditioner (AC) adalah perangkat yang dirancang untuk mengatur suhu dan kelembapan udara di dalam suatu ruangan. AC digunakan dalam berbagai tempat seperti rumah tinggal, perkantoran, gedung perkuliahan, pusat pembelanjaan dan transportasi. AC dapat menciptakan lingkungan yang nyaman sehingga dapat meningkatkan produktivitas bagi penggunanya. Suhu ruangan dan kelembapan yang optimal membantu pengguna untuk lebih fokus dalam beraktivitas (Widiarto et al., 2022).

2.2 Sistem Kerja AC

Pendingin Ruangan *Air Conditioner* mampu mendinginkan udara di dalam ruangan sesuai keinginan, terutama untuk menurunkan suhu ruangan agar lebih dingin dibandingkan dengan suhu lingkungan sekitarnya. Sistem dalam AC terdiri dari kompresor, kondensor, tabung orifice, evaportator dan katup expansi. Adapun sistem kerja AC terdapat pada Gambar 2.1.



Gambar 2.1 Sistem Kerja AC (Golwa et al., 2023)

Kompresor dalam sistem *Air Conditioner* berfungsi untuk mengatur sirkulasi cairan fluid (*refigerant*). Kompresor berfungsi untuk menekan dan mengubah *refigerant* dari bentuk gas bertekanan rendah menjadi gas bertekanan tinggi. Setelah itu, gas bertekanan tinggi ini dialirkan ke kondensor, dimana gas tersebut didinginkan dan dikondensasikan menjadi cairan bertekanan tinggi.

Kalor yang dilepaskan oleh kondensor merupakan gabungan dari energi yang digunakan oleh kompresor dan kalor yang diserap oleh evaporator dari substansi yang akan didinginkan.

Setelah *refigerant* melewati kondensor dan melepaskan kalor penguapan, cairan *refigerant* berubah dari fase uap menjadi fase cair. Kemudian cairan *refigerant* dialirkan melalui katup ekspansi, dimana tekanannya diturunkan. Penurunan tekanan ini menyebabkan cairan *refigerant* berubah dari fase cair menjadi fase uap.

Selanjutnya, cairan *refigerant* yang bertekanan rendah ini kemudian dialirkan ke evaporator. Di dalam evaporator, cairan *refigerant* mengalami perubahan fase dari cair menjadi uap lagi karena tekanan yang sangat rendah saat melewati katup ekspansi yang menghasilkan efek pendinginan (Golwa et al., 2023).

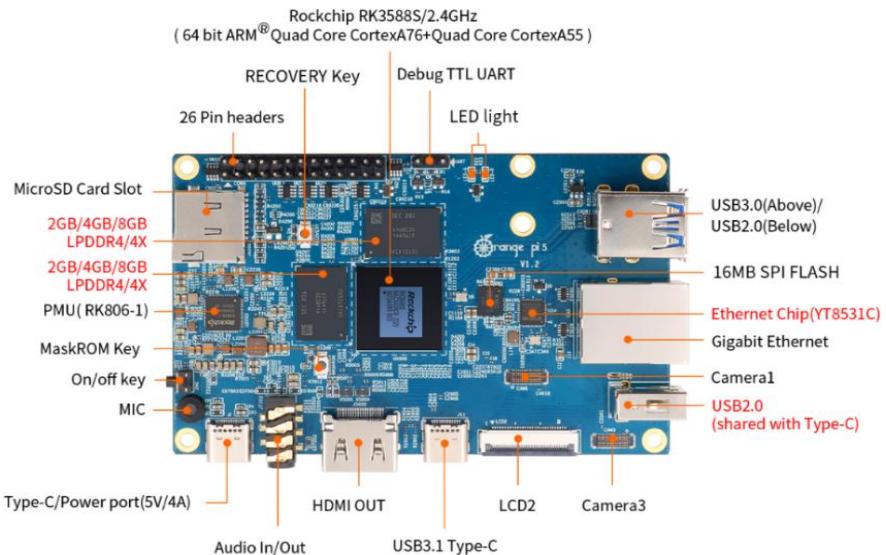
2.3 Single Board Computer

Single Board Computer (SBC) adalah bentuk komputer yang dirangkai pada suatu papan sirkuit yang mencakup CPU, GPU, memori dan port I/O yang merupakan komponen minimal agar SBC dapat berfungsi sepenuhnya. SBC dapat digunakan untuk berbagai keperluan contohnya pengendalian sistem *embedded computer*. Contoh SBC adalah Banana Pi, Raspberry Pi, Orange Pi, Rock Pi dan masih banyak lagi (Arief et al., 2018).

2.4 Orange Pi 5

Orange Pi 5 adalah salah satu single board computer yang sejenis dengan Raspberry Pi, Nano Pi, dan jenis Pi lainnya (Cepi et al., 2022). Orange Pi 5 mengadopsi prosesor ARM RK3588S octa-core 64-bit, memiliki *built-in Neural Processing Unit (NPU) AI accelerator* dengan daya komputasi hingga 6 Tops.

Orange Pi 5 juga dilengkapi dengan GPU ARM Mali-G610 MP4, serta memiliki. Memiliki opsi RAM mulai dari 4GB hingga 32GB (LPDDR4/4x), dan kemampuan pemrosesan tampilan hingga 8K. Orange Pi 5 dapat memenuhi kebutuhan berbagai aplikasi seperti edge computing, artificial intelligence dan lain-lain (Wiki-Orange Pi Contributors, 2023).



Gambar 2.2 Orange Pi 5 (Wiki-Orange Pi Contributors, 2023).

2.5 Arduino

Arduino adalah sebuah sistem perangkat keras dan lunak yang bersifat open source. Arduino memiliki berbagai macam jenis board seperti Arduino UNO, Arduino Nano, Arduino Lilypad, Arduino Mega dan masih banyak lagi. Arduino memiliki IDE (*Integrated Development Environment*) yang merupakan perangkat lunak yang berfungsi untuk menulis, mengompilasi dan mengunggah program ke board Arduino sesuai dengan jenis board yang digunakan.

2.5.1 Arduino UNO

Arduino UNO merupakan platform yang dirancang untuk membantu peneliti dalam mengembangkan perangkat yang menggunakan sensor (Shakirovich, 2022). Arduino UNO berbasis pada datasheet ATmega328P. Arduino UNO dilengkapi dengan 14 pin untuk input/output, 6 input analog, osilator kristal 16 MHz, koneksi USB, jack power, tombol reset dan header ICSP.



Gambar 2.3 Arduino UNO
(Sumber: <https://store.arduino.cc/products/arduino-uno-rev3>)

2.6 Universal Asynchronous Receiver Transmitter (UART)

Universal Asynchronous Receiver Transmitter (UART) adalah salah satu jenis komunikasi serial yang digunakan untuk pertukaran data jarak pendek dan kecepatan rendah, dan biaya rendah antara satu perangkat dengan perangkat yang lain. UART terdiri dari 3 komponen utama yaitu *baud rate*, *transmitter* dan *receiver*.

Baud rate berfungsi sebagai penentu kecepatan komunikasi dan memastikan bahwa data yang dikirim dan diterima berada pada kecepatan yang sama. Untuk melakukan komunikasi dua arah, UART hanya memerlukan dua jalur sinyal yaitu *transmitter* (TXD) dan *receiver* (RXD). TXD adalah jalur pengiriman data dari UART, sedangkan RXD adalah jalur penerimaan data ke UART (Banti et al., 2024).

2.7 DHT22

DHT22 adalah sensor yang dirancang untuk mendeteksi suhu dan kelembapan. DHT22 mampu mengukur suhu dan kelembapan dengan ketelitian +/- 0.5 °C untuk suhu dan +/- 2% untuk kelembapan. Melalui pin-pinnya, DHT22 dapat terhubung dengan perangkat lain seperti Arduino.



Gambar 2.4 DHT22

(Sumber: <https://id.szks-kuongshun.com/uno/uno-sensor/dht22-digital-temperature-humidity-sensor-module.html>)

DHT22 memiliki 3 pin, penulis menyajikan data pada tabel 2.1

Tabel 2.1 Fungsi Pin DHT22

Pin	Fungsi
-	Ground
+	Sumber Daya 5v
Out	Mengirimkan data suhu dan kelembapan

Berdasarkan Tabel 2.1 DHT22 memiliki tiga pin yang masing-masing memiliki fungsi berbeda-beda. Pin (-) digunakan untuk koneksi ground, yang berfungsi sebagai

ground pada rangkaian listrik. Pin (+) digunakan sebagai sumber daya yang disuplai dengan tegangan 5 volt untuk memastikan sensor berfungsi dengan baik. Pin ‘out’ berfungsi untuk mengirimkan data suhu dan kelembapan yang telah diukur oleh sensor.

2.8 IR Receiver

Infrared receiver adalah modul penerima data yang menggunakan gelombang infra merah dengan frekuensi carrier sebesar 38kHz. Modul ini berfungsi sebagai input dalam aplikasi transmisi data nirkabel, salah satunya adalah remote kontrol. (Yusniati, 2018).



Gambar 2.5 IR Receiver

(Sumber: <https://www.botnroll.com/en/infrared/4870-vs1838-ir-receiver.html>)

2.9 IR Transmitter

Infrared transmitter adalah suatu perangkat yang mengirim data menggunakan gelombang infra merah dengan frekuensi bawaan sebesar 38 kHz secara nirkabel. Perangkat ini dapat berfungsi sebagai output pada berbagai aplikasi transmisi data nirkabel, seperti pada sistem robotik dan keamanan. Contoh penggunaan LED inframerah adalah pada remote TV, AC dan remote kontrol lainnya yang menggunakan LED Inframerah (Yusniati, 2018).



Gambar 2.6 IR Transmitter

(Sumber: <https://www.cytron.io/p-infrared-transmitter-led-3mm>)

2.10 Real Time Streaming Protocol (RTSP)

Real Time Streaming Protocol (RTSP) adalah protocol tingkat aplikasi yang

digunakan sebagai pengendali untuk server media streaming. RTSP mendukung kendali pengiriman data dengan karakteristik real-time, terutama dalam konteks streaming media. RTSP memiliki kelebihan untuk menjaga konsistensi hubungan antara server video streaming dan klien, sehingga sesi media dapat diatur dan dikendalikan dengan baik (Fadhli et al., 2019).

RTSP bekerja sama dengan protocol RTP (*Real-time Transport Protocol*) untuk mengirim data real-time. Layanan RTSP umumnya digunakan dalam sistem IP Kamera untuk mengakses dan mengontrol aliran video secara langsung.

Setiap merek IP kamera dapat memiliki format link RTSP yang berbeda-beda. Berikut contoh link RTSP dengan beberapa merek:

1. Arecont: rtsp://ip-addr/h264.sdp
2. Amcrest:
rtsp://[username]:password@IPaddress:554/cam/realmonitor?/channel=1&subtyp
e=0
3. Samsung : rtsp://<DeviceIP>:port/LiveChannel/0/media.smp

2.11 Computer Vision

Computer Vision adalah cabang dari kecerdasan buatan yang berfokus pada pembuatan mesin atau sistem yang mampu melihat yang berarti memberikan kemampuan kepada sistem komputer untuk mengekstraksi informasi dari data visual seperti gambar dan video. Contoh *computer vision* adalah segmentasi, klasifikasi, dan deteksi objek (Faishal et al., 2023).

2.12 Object Detection

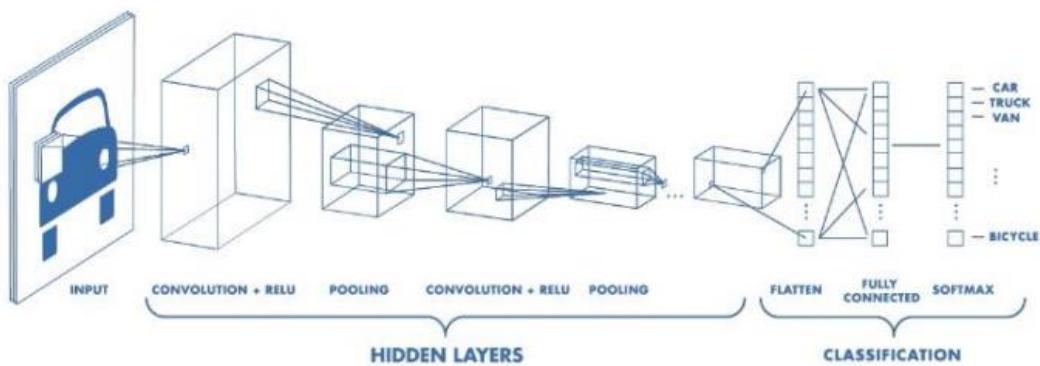
Object Detection adalah suatu proses yang digunakan untuk mengidentifikasi keberadaan objek khusus dalam sebuah gambar digital. Proses deteksi ini melibatkan penggunaan berbagai metode yang pada dasarnya melibatkan pemindaian fitur-fitur dari seluruh objek pada gambar input (Novandra et al., 2021).

2.13 Deep Learning

Deep Learning merupakan cabang dari kecerdasan buatan dan *machine learning* yang memanfaatkan jaringan saraf tiruan (*Neural Networks*) dengan *multiple layer* untuk meningkatkan akurasi dalam tugas-tugas kecerdasan buatan seperti deteksi objek, pengenalan suara, terjemahan bahasa (Raup et al., 2022).

2.14 Convolutional Neural Networks (CNN)

Convolutional Neural Network merupakan algoritma *deep learning* yang umumnya digunakan dalam berbagai aplikasi. CNN sering digunakan untuk klasifikasi gambar, segmentasi, deteksi objek, pemrosesan video, dan *natural language processing*. (Purwono et al., 2022).

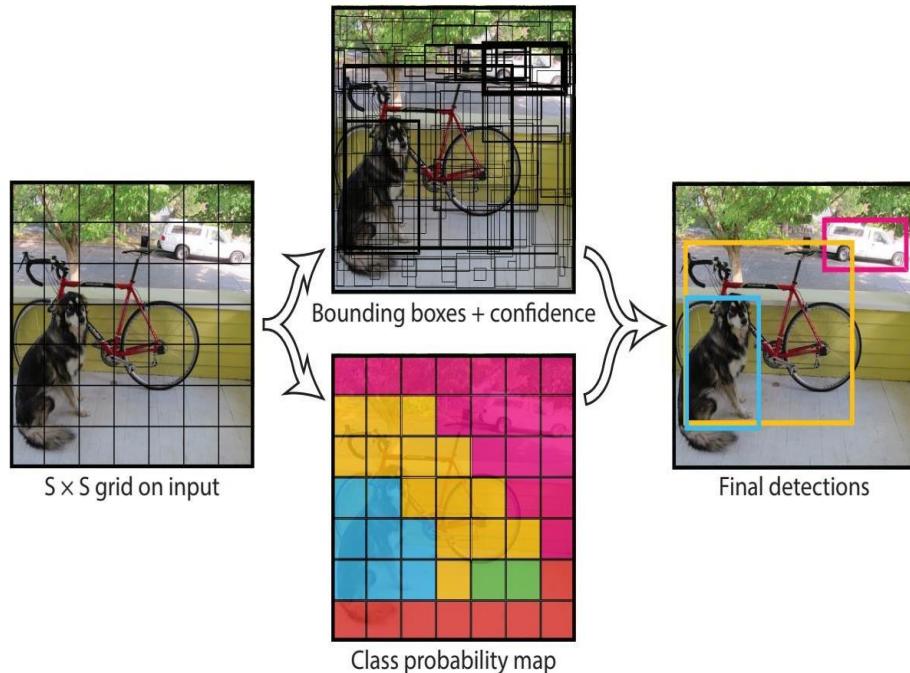


Gambar 2.7 Arsitektur CNN
(Purwono et al., 2022)

Convolutional layer merupakan komponen inti dari CNN yang terdiri dari berbagai layer yang dapat dipelajari. Fungsinya adalah untuk mengekstraksi fitur-fitur penting dari volume input dan meneruskannya ke lapisan berikutnya untuk menghasilkan representasi yang semakin kompleks. Setelah *convolutional layer*, *pooling layer* digunakan untuk mengurangi dimensi data dan jumlah parameter input. *Fully Connected Layer* menghubungkan setiap *neuron* dari lapisan sebelumnya ke setiap *neuron* tunggal.

2.15 You Only Look Once (YOLO)

You Only Look Once (YOLO) merupakan metode untuk mendeteksi objek berbasis *Convolutional Neural Network* (CNN). Metode YOLO bekerja dengan membagi citra ke dalam beberapa *grid* yang lebih kecil. Metode ini menghasilkan *Feature Map* yang mencakup *bounding box*, skor kepercayaan (*confidence*), dan klasifikasi objek. YOLO dikenal sebagai salah satu pendekatan deteksi objek paling cepat yang memberikan kinerja yang baik dan tingkat akurasi yang tinggi (Sarosa et al., 2021).



Gambar 2.8 Mekanisme Arsitektur YOLO
(Redmon et al., 2016)

Berdasarkan Gambar 2.8, YOLO memiliki tiga tahapan utama dalam proses deteksi objek. Tahap pertama dimulai dengan membagi citra menjadi grid-grid dengan ukuran $s \times s$ untuk mendeteksi objek. Setiap grid akan melakukan prediksi terhadap *bounding box* dan nilai *confidence*. *Confidence* mencerminkan keyakinan bahwa *bounding box* tersebut berisi objek yang sesuai, serta mengukur tingkat akurasi prediksi secara keseluruhan.

Pada tahapan kedua, setiap *bounding box* didefinisikan dengan lima variabel, yaitu x, y, w, h dan c . Variabel x dan y mewakili koordinat titik tengah *bounding box* objek yang terdeteksi, sedangkan w dan h adalah dimensi lebar dan tinggi. Variabel c mencakup nilai *confidence* dari *bounding box*, memberikan informasi tentang sejauh mana model yakin dengan keberadaan objek tersebut.

Tahap terakhir melibatkan prediksi nilai probabilitas kelas oleh setiap grid jika terdapat objek di dalamnya. Nilai probabilitas kelas dan nilai *confidence* dari *bounding box* dikalikan, menghasilkan nilai *confidence* pada setiap *bounding box* untuk setiap kelas secara khusus. Pendekatan ini memungkinkan YOLO untuk memberikan prediksi deteksi objek yang lebih akurat dengan mempertimbangkan *confidence* dan probabilitas kelas pada setiap langkah (Sarosa et al., 2021).

2.16 YOLOv7

YOLOv7 dikembangkan pada tahun 2022 oleh Chien-Yaou et al., yang mengambil konsep dasar dari seri YOLO. Namun, yang membedakan YOLOv7 adalah kecepatan dan akurasinya yang tinggi, melampaui semua detektor objek dalam rentang kecepatan 5 hingga 160 FPS (*Frame per Second*) yang membuatnya menjadi pilihan yang sangat efisien untuk aplikasi yang memerlukan deteksi objek yang cepat dan akurat (Y. Wang et al., 2022).

Pada YOLO versi 7 ada beberapa model YOLOv7 yang dapat digunakan yaitu YOLOv7, YOLOv7-X, YOLOv7-W6, YOLOv7-E6, YOLOv7-D6, YOLOv7-E6E. Adapun perbandingan performa model-model YOLOv7 ketika diuji dengan dataset COCO dapat dilihat pada Tabel 2.2.

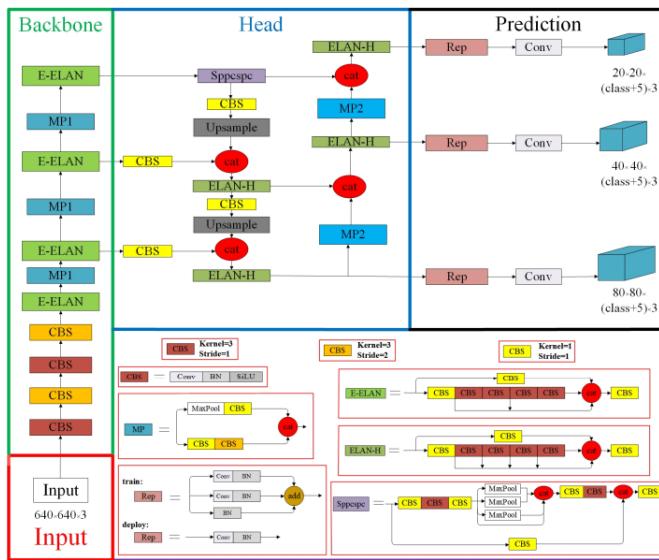
Tabel 2.2 Perbandingan Beberapa Model YOLOv7

Model	Test Size	AP ^{test}	AP ₅₀ ^{test}	AP ₇₅ ^{test}	Batch 1 fps	Batch 32 avg time
YOLOv7	640	51.4%	69.7%	55.9%	161 fps	2.8 ms
YOLOv7-x	640	53.1%	71.2%	57.8%	114 fps	4.3 ms
YOLOv7-W6	1280	54.9%	72.6%	60.1%	84 fps	7.6 ms
YOLOv7-E6	1280	56.0%	73.5%	61.2%	56 fps	12.3 ms
YOLOv7-D6	1280	56.6%	74.0%	61.8%	44 fps	15.0 ms
YOLOv7-E6E	1280	56.8%	74.4%	62.1%	36 fps	18.7 ms

YOLOv7 memiliki 3 bagian utama yaitu, *backbone*, *neck*, dan *head*. Pada bagian *backbone* YOLOv7 menerima input sebesar 640x640 piksel yang berfungsi untuk mengoptimalkan pra-pemrosesan dengan teknik data mosaik dan hibrida, serta memanfaatkan metode perhitungan bingkai anchor dari YOLOv5. Jaringan *backbone* YOLOv7 terdiri dari tiga komponen utama yaitu CBS, E-LAN, dan MP1. CBS melakukan proses konvolusi, normalisasi batch dan fungsi aktivasi SiLU. Lalu, komponen E-ELAN melakukan peningkatan kemampuan pembelajaran jaringan dan MP1 menggabungkan CBS dan MaxPool untuk ekstraksi fitur yang lebih baik.

Pada bagian *Neck*, bagian ini berfungsi untuk mengekstrak dan menggabungkan informasi dari bagian sebelumnya yaitu backbone dan menyiapkannya sebelum diproses ke bagian head. Bagian Neck terdapat modul SPP (*Spatial Pyramid Pooling*), lapisan ini berguna untuk menghilangkan batas ukuran tetap pada jaringan.

Bagian *Head* pada YOLOv7 berfungsi untuk memproses lebih lanjut fitur yang diekstrak oleh *neck*. *Head* berfungsi untuk menyesuaikan jumlah saluran citra dan memprediksi informasi objek seperti *confidence*, kategori, dan bingkai *anchor*. Pada bagian *Head*, bagian ini menggunakan NMS atau *non maximum suppression*. NMS merupakan komponen yang sangat penting untuk menghindari hasil deteksi yang bertindihan dan memastikan untuk mengeluarkan hasil deteksi dengan nilai confident yang paling tinggi.



Gambar 2.9 Arsitektur YOLOv7
(Liu et al., 2023)

2.17 Penelitian Terdahulu

Beberapa penelitian sebelumnya telah berhasil untuk membuat sistem pengendali AC otomatis berdasarkan jumlah manusia. Penelitian oleh Mileanasari et al. (2023) mengembangkan sebuah sistem pengendalian AC yang menggunakan model *You Only Look Once* (YOLO) V5 dalam pengolahan citra untuk menghitung jumlah yang orang yang diterapkan pada SBC Nvidia Jetson Nano. Hasilnya sistem yang dirancang dapat mengendalikan suhu AC berdasarkan deteksi jumlah orang dan membuat penurunan konsumsi energi listrik di laboratorium sebesar 17.13%.

Penelitian yang dilakukan oleh Natsir et al. (2019), menggunakan IOT untuk mengimplementasikan sistem kendali AC otomatis pada ruang kelas di Universitas Serang Raya. Sistem ini memanfaatkan Arduino UNO sebagai perangkat utama, lalu sensor *ultrasonic* untuk menghitung jumlah orang yang masuk ruangan, dan sensor suhu DS18B20 untuk memonitor suhu ruangan. Hasilnya sistem dapat mengatur suhu

secara otomatis berdasarkan jumlah orang dan dapat dimonitor dari web server melalui web browser.

Penelitian oleh Afandi et al., (2021) menggunakan sensor passive infrared untuk mendeteksi keberadaan orang diruangan dan membuat perangkat pengendali otomatis berbasis IOT. Hasilnya sistem mampu mengendalikan AC dengan membuat mode *Cold*, *Medium*, dan *Cool* melalui web dan mematikan otomatis jika sensor ultrasonic tidak mendeteksi orang di dalam ruangan.

Pella et al., (2019) membuat implementasi otomasi pengaturan suhu ruangan berdasarkan deteksi jumlah orang menggunakan SBC Raspberry Pi, Tensorflow, dan OpenCV. Sistem ini juga melibatkan mikrokontroller ESP8266, Pi kamera, dan modul infra merah untuk mengontrol *Air Cooling System* (ACS) dengan tingkatan suhu berkisar antara 18°C hingga 24°C. Pengujian dilakukan pada ruangan kosong dan memiliki kapasitas 50 orang, menghasilkan kemampuan sistem dalam mendeteksi jumlah orang dan mengatur ACS. Namun, beberapa kendala terjadi akibat posisi kemiringan kamera yang belum optimal.

Widura et al., (2023) mengembangkan sistem pengendalian AC otomatis menggunakan mikrokontroller ESP32 yang mendeteksi pergerakan manusia dan terintegrasi dengan Internet Of Things berbasis Blynk. Menggunakan pendekripsi gerakan dari sensor gerak, sistem pengendalian AC otomatis dapat beroperasi dengan baik, menghasilkan penghematan energi harian hingga 15% dibandingkan dengan pengendalian manual.

Tabel 2.3 Penelitian Terdahulu

No	Peneliti	Judul	Keterangan
1	Mileanasari <i>et al.</i> , (2023)	Pemantauan dan Pengendalian Suhu <i>Air Conditioner</i> untuk Penghematan Energi Berbasis Pengolahan Citra	Mengembangkan sistem pengendalian AC otomatis menggunakan YOLOv5 berdasarkan jumlah manusia pada SBC Jetson Nano. Hasilnya dapat menghemat konsumsi energi listrik di laboratorium sebesar 17.13%.
2	Natsir <i>et al.</i> , (2019)	Implementasi IOT untuk Sistem Kendali AC	Membuat sistem kendali AC otomatis berdasarkan jumlah manusia menggunakan sensor ultrasonic.

		Otomatis Pada Ruang Kelas di Universitas Serang Raya	Hasilnya AC dapat dikontrol otomatis berdasarkan jumlah orang dan dapat dikontrol manual melalui web.
3	Pella et al., (2019)	Disain Sistem Otomasi Suhu Ruangan Pertemuan dengan Penerapan Teknik Machine Learning	Menggunakan Raspberry Pi, <i>Tensorflow</i> , dan OpenCV untuk mengendalikan suhu AC berdasarkan jumlah orang. Hasilnya, sistem berhasil mengatur suhu berdasarkan jumlah orang, namun sedikit terhambat dalam mendekripsi karena kemiringan kamera yang belum optimal.
4	Afandi <i>et al.</i> , (2021)	Automated Air Conditioner Controller and Monitoring Based on Internet of Things	Menggunakan sensor passive infrared untuk mendekripsi keberadaan orang diruangan dan membuat perangkat pengendali otomatis berbasis IOT. Hasilnya sistem mampu mengendalikan AC melalui web dan mematikan otomatis jika tidak terdeteksi orang di dalam ruangan.
5	Widura et al., (2023)	IoT-Based Air Conditioning Control System for Energy Saving	Mengembangkan sistem pengendalian AC otomatis menggunakan ESP32 yang teintergrasi dengan IoT dan menggunakan sensor gerak. Hasilnya sistem pengendalian otomatis menghasilkan penghematan energi hingga 15%.

2.18 Perbedaan Penelitian

Terdapat perbedaan penelitian antara penelitian yang telah dijelaskan sebelumnya dengan penelitian ini, yaitu pada penelitian ini penulis menggunakan model pre-trained YOLO-V7 untuk mendekripsi manusia. Selain itu pengaturan suhu juga akan disesuaikan kembali 5 menit setelah pengaturan suhu awal, jika suhu lebih besar atau lebih kecil dari pengaturan suhu awal maka suhu akan diatur sebesar

selisihnya. Penelitian ini menggunakan *single board computer* Orange Pi 5 sebagai perangkat utama untuk pendekslan manusia dan pengriman perintah ke Arduino UNO. Sedangkan Arduino UNO akan menerima perintah dari Orang Pi 5 untuk mengontrol AC ataupun mendekksi suhu aktual berdasarkan perintah yang diterimanya.

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

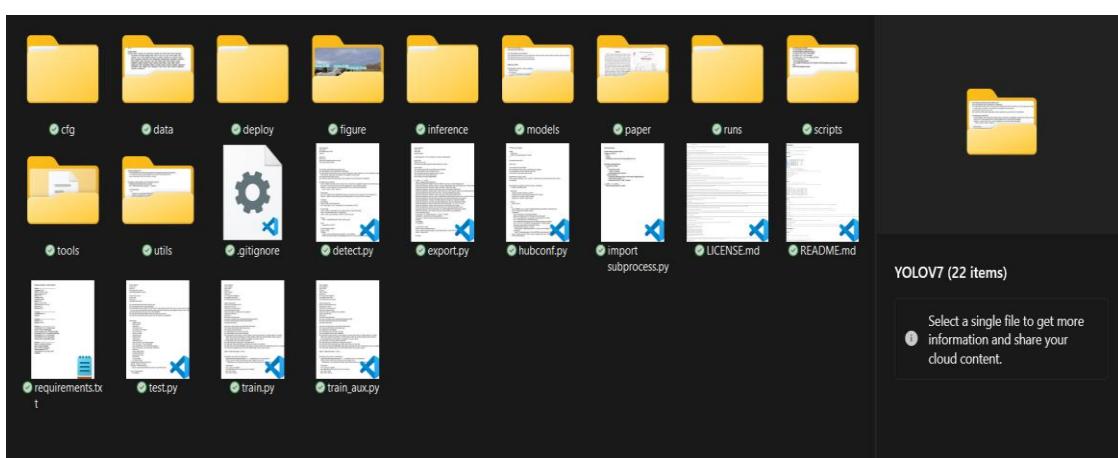
3.1 Cloning Repository YOLOv7

Direktori YOLOv7 diperoleh dengan mengkloning repository Github milik WongKinYiu dari link <https://github.com/WongKinYiu/yolov7>.

```
HP@LAPTOP-LFL3FUJS MINGW64 ~/Downloads/baru_skripsi
$ git clone https://github.com/WongKinYiu/yolov7.git
Cloning into 'yolov7'...
remote: Enumerating objects: 1197, done.
remote: Total 1197 (delta 0), reused 0 (delta 0), pack-reused 1197
Receiving objects: 100% (1197/1197), 74.23 MiB | 1010.00 KiB/s, done.
Resolving deltas: 100% (519/519), done.
```

Gambar 3.1 Proses Cloning Github WongKinYiu

Berikut adalah tampilan isi direktori YOLOv7 pada Gambar 3.2.



Gambar 3.2 Isi Direktori Repository YOLOv7

Di dalam repository YOLOv7 terdapat file ‘detect.py’ yang digunakan sebagai file utama untuk melakukan pendekripsi objek. File ‘detect.py’ memerlukan weight

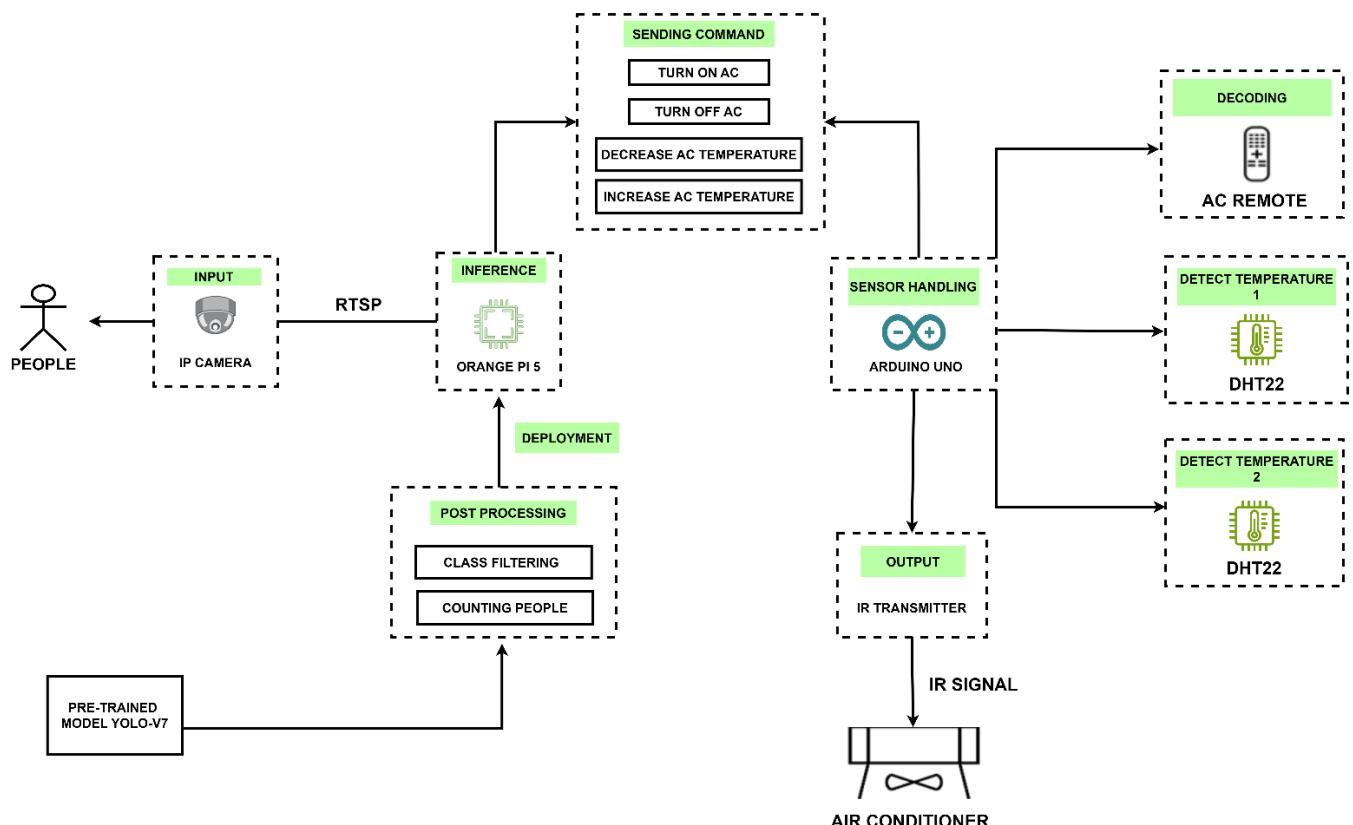
‘yolov7.pt’ sebagai acuan untuk pendekslsian objek ‘person’. Model YOLOv7 di dalam repository ini, telah dievaluasi menggunakan dataset COCO 2017 sebanyak 143.575 data gambar. Rincian alokasi data dapat ditemukan dalam file ‘coco.yaml’. Berdasarkan informasi dari file ‘coco.yaml’, data ini dibagi menjadi 3 bagian, data pelatihan (training) sebanyak 118.287 gambar, data pengujian (testing) sebanyak 20.288 dan data validasi (validation) sebanyak 5000 gambar.

3.2 Penginstallan Requirement

Tahapan ini merupakan tahapan untuk menginstalasi paket-paket yang diperlukan untuk menggunakan YOLOv7. Karena repository YOLOv7 menggunakan python, maka komponen yang diperlukan seperti matplotlib, numpy, opencv harus diinstall terlebih dahulu. Selain itu, instalasi Pytorch dan CUDA juga harus diinstall untuk memulai proses deteksi menggunakan YOLOv7.

3.3 Arsitektur Umum

Penelitian ini melibatkan beberapa tahapan memiliki beberapa tahapan yang dimulai dari proses input, deployment, inference, sampai pada pengiriman karakter yang berisi perintah dari Orange Pi 5 ke Arduino UNO untuk pengontrolan *Air Conditioner*.



Gambar 3.3 Arsitektur Umum

3.3.1 Input

Data input adalah data real time dari IP kamera dengan menggunakan protokol RTSP (Real Time Streaming Protocol). Sistem akan melakukan pengambilan gambar setiap 10 menit untuk mendeteksi perubahan jumlah orang di ruangan.

URL RTSP yang akan digunakan adalah ‘rtsp://admin:qwerty192.168.0.189:554’, yang merupakan URL yang berasal dari IP kamera yang digunakan. Berikut penjelasan format url RTSP:

1. **rtsp://** : Merupakan skema protokol yang menunjukan bahwa protokol RTSP adalah protokol untuk mengakses IP Kamera.
2. **admin:qwerty180845@**: Merupakan kredensial login untuk mengakses perangkat. “admin” dalam nama pengguna dan “qwerty170845” adalah kata sandi.
3. **192.168.0.189** : Merupakan alamat IP dari perangkat yang digunakan untuk mengakses IP kamera. Alamat ini menunjukkan bahwa perangkat berada di jaringan LAN (*Local Area Network*).
4. **:554** : Merupakan nomor port default yang digunakan oleh protokol RTSP.

Berikut adalah cara untuk mengakses url RTSP pada python.

```
rtsp_url = 'rtsp://admin:qwerty192.168.0.189:554'
cap = cv2.VideoCapture(rtsp_url)
ret, frame = cap.read()
if not ret:
    print("Error: Unable to Capture photo")
cap.release()
```

Script ini bertujuan untuk mengambil gambar dari stream video menggunakan protokol RTSP (*Real Streaming Protocol*). Pertama, URL RTSP akan disimpan ke dalam variabel. Kemudian metode ‘cap.read’ digunakan untuk membaca frame pertama dari stream video.

Metode ini mengembalikan dua nilai yaitu ‘ret’ yang merupakan sebuah boolean yang menunjukkan apakah pembacaan frame berhasil dan ‘frame’ yang berisi data frame jika pembacaan berhasil. Jika ‘ret’ bernilai ‘False’ ini menunjukkan bahwa frame tidak berhasil dibaca dan akan menampilkan pesan ‘Error: Unable to capture photo’. Selanjutnya OpenCV yang diimpor sebagai ‘cv2’, yang merupakan *library* untuk pengolahan gambar dan video, digunakan untuk

membuka koneksi ke stream video. URL RTSP akan dimasukkan ke dalam objek ‘VideoCapture’ dari library ‘cv2’. Selanjutnya ‘cap.release’ akan melepasakan sumber daya menjalankan objek ‘VideoCapture()’ untuk memastikan koneksi ke stream video ditutup dengan benar.

3.3.2 Post Processing

Pada tahap ini dilakukan *post processing* untuk menyesuaikan pre-trained model YOLOv7 *bounding box* untuk hanya menyertakan kelas ‘person’ sehingga, hanya objek dengan kelas ‘person’ yang akan diproses lebih lanjut dan menghitung jumlah orangnya berdasarkan *bounding box* yang terdeteksi.

3.3.2.1 Class Filtering

```
# number of classes
nc: 80

# class names
names: [ 'person', 'bicycle', 'car', 'motorcycle', 'airplane', 'bus', 'train', 'truck', 'boat', 'traffic light',
'fire hydrant', 'stop sign', 'parking meter', 'bench', 'bird', 'cat', 'dog', 'horse', 'sheep', 'cow',
'elephant', 'bear', 'zebra', 'giraffe', 'backpack', 'umbrella', 'handbag', 'tie', 'suitcase', 'frisbee',
'skis', 'snowboard', 'sports ball', 'kite', 'baseball bat', 'baseball glove', 'skateboard', 'surfboard',
'tennis racket', 'bottle', 'wine glass', 'cup', 'fork', 'knife', 'spoon', 'bowl', 'banana', 'apple',
'sandwich', 'orange', 'broccoli', 'carrot', 'hot dog', 'pizza', 'donut', 'cake', 'chair', 'couch',
'potted plant', 'bed', 'dining table', 'toilet', 'tv', 'laptop', 'mouse', 'remote', 'keyboard', 'cell phone',
'microwave', 'oven', 'toaster', 'sink', 'refrigerator', 'book', 'clock', 'vase', 'scissors', 'teddy bear',
'hair drier', 'toothbrush' ]
```

Gambar 3.4 Daftar *Class* pada Dataset COCO

Model Pre trained YOLOv7 telah dilatih sebelumnya mampu mengenali 80 class yang berbeda, seperti yang ditunjukkan pada Gambar 3.4, menunjukkan isi dari file ‘coco.yaml’ yang terdapat daftart class pada dataset COCO yang disimpan di dalam array dengan nama names.

Class ‘person’ terdapat pada elemen pertama dalam indeks 0, yang dapat digunakan untuk melakukan penyaringan class dengan menggunakan argumen ‘--class 0’. Argumen ini mengarahkan model untuk mendeteksi class pada indeks 0 yaitu class ‘person’.

3.3.2.2 Counting People

Untuk menghitung jumlah orang, dilakukan perhitungan dari jumlah *bounding box* yang telah terdeteksi. Berikut adalah script python untuk menghitung jumlah class ‘person’.

```
for c in det[:, -1].unique():
    n = (det[:, -1] == c).sum()
    s += f"{n} {names[int(c)]}{('s' * (n>1))},"
    if names[int(c)] == 'person':
```

```

    person_count += n
else: person_count=0

```

Script diatas merupakan script untuk meghitung jumlah deteksi untuk setiap kelas objek yang tercatat di dalam array ‘det’. Jika kelas objek adalah ‘person’, jumlah deteksinya ditambahkan ke dalam ‘person_count’. Selanjutnya, hasil dari perhitungan jumlah deteksi ‘person’ tersebut akan disimpan ke dalam sebuah file teks dengan nama ‘jumlah.txt’ menggunakan script berikut.

```

path_file = "jumlah.txt"
if person_count != 0:
    person_count = person_count.item()
with open(path_file, 'w') as f:
    f.write(str(person_count))
    print(f'Jumlah deteksi person: {person_count}')

```

Script di atas akan menyimpan nilai dari variabel ‘person_count’ ke dalam file text ‘jumlah.txt’ dan mencetak jumlah deteksi kelas ‘person’ tersebut ke layar.

3.3.3 Deployment

Repository dan model *pre-trained* YOLOv7 akan *di-deploy* ke *single board computer* Orange pi 5. Deployment dilakukan untuk menjalankan script dan model untuk mendeteksi manusia di Orange Pi 5.

3.3.4 Inference

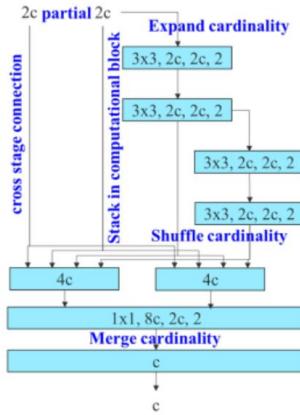
Inference adalah proses model YOLOv7 dalam menghasilkan output pendektsian seperti menghasilkan bounding box beserta confidencenya (Bin Tuan Rashid & Fadzil, 2023). Proses model YOLOv7 dalam mendekksi objek melibatkan 4 tahapan yaitu *Input*, *Backbone*, *Neck* dan *Head*.

a. *Input*

Input pada YOLOv7 menerima input sebesar 640x64 piksel yang berfungsi untuk mengoptimalkan pra-pemrosesan dengan teknik data mosaik dan hibrida, serta memanfaatkan metode perhitungan bingkai anchor dari YOLOv5.

b. *Backbone*

Backbone memiliki jaringan yang bernama E-ELAN (Exftended Efficient Layer Aggregation Network, jaringan E-ELAN berfungsi melakukan peningkatan kemampuan pembelajaran jaringan dan MP1 menggabungkan CBS dan MaxPool untuk ekstraksi fitur yang lebih baik.



Gambar 3.5 E-LAN
(C.-Y. Wang et al., 2022)

c. Neck

Pada bagian Neck, bagian ini berfungsi untuk mengekstrak dan menggabungkan informasi dari bagian sebelumnya yaitu backbone dan menyiapkannya sebelum diproses ke bagian head. Bagian Neck terdapat modul SPP (*Spatial Pyramid Pooling*), lapisan ini berguna untuk menghilangkan batas ukuran tetap pada jaringan.

d. Head

Bagian Head pada YOLOv7 berfungsi untuk memproses lebih lanjut fitur yang diekstrak oleh neck. Head berfungsi untuk menyesuaikan jumlah saluran citra dan memprediksi informasi objek seperti confidence, kategori, dan bingkai anchor. Pada bagian Head, bagian ini menggunakan NMS atau *Non Maximum Supression*. NMS merupakan komponen yang sangat penting untuk menghindari hasil deteksi yang bertindihan dan memastikan untuk mengeluarkan hasil deteksi dengan nilai confident yang paling tinggi.

3.3.5 Decoding

Tahapan ini merupakan proses pengambilan data tombol dari remote AC yang akan ditangkap oleh IR receiver dan digunakan untuk pengendalian AC otomatis sesuai suhu yang ditentukan. Data remote yang diambil adalah data yang dibutuhkan untuk pengaturan suhu AC seperti tombol untuk mematikan AC, menghidupkan AC, menaikkan suhu dan menurunkan suhu.

Berikut adalah contoh data remote yang dihasilkan oleh remote AC.

```
uint16_t rawData[137] = {4280,4420, 480,1670, 480,620, 430,1670, 480,1670,
480,570, 480,570, 480,1670, 480,620, 480,570, 480,1670, 480,570, 480,570,
480,1670, 480,1670, 480,620, 430,1670, 480,570, 480,620, 430,1670, 480,1670,
480,1670, 480,1670, 430,1720, 430,1670, 480,1670, 430,1720, 430,620, 430,620,
480,620, 430,620, 430,620, 480,620, 430,1720, 430,620, 430,1720, 430,1670,
480,620, 430,620, 430,620, 480,620, 430,620, 430,1720, 430,620, 430,620,
480,1670, 430,1720, 430,1720, 430,1670, 430,5320, 4280,4420, 480,1670,
430,670, 430,1670, 430,1720, 380,720, 380,670, 380,1770, 380,670, 430,620,
430,1770, 330,720, 230,920, 180,1920, 230,1870, 330,2920, 280,720, 380,620,
430};
```

`uint16_t rawData[137]` merupakan informasi dari salah satu tombol, yang menunjukkan isi dari array bernama ‘`rawData`’, sebanyak 137 nilai 16-bit yang dapat digunakan dalam pengiriman sinyal inframerah dari remote. Setiap tombol memiliki pola nilai yang berbeda-beda. Dalam remote AC, setiap data yang diberikan, merupakan kode yang ditampilkan pada remote. Misalnya, jika ditekan pada tombol 17 pada mode cool, dan fan 3. Maka kode AC yang akan keluar adalah kode yang merepresentasikan pada tampilan remote tersebut.

3.3.6 Output

Tahapan ini merupakan tahapan untuk mengirim perintah dari Orange Pi 5 ke Arduino UNO. Pada tahap ini, Orange Pi 5 berkomunikasi dengan Arduino UNO secara serial untuk mengatur pengiriman perintah ke AC. Orange Pi 5 mengirimkan intruksi berupa karakter yang akan diproses oleh Arduino UNO untuk mengontrol AC berdasarkan analisis kondisi suhu ruangan dan jumlah orang.

Intruksi yang dikirimkan mencakup perintah untuk menghidupkan AC, mematikan AC, menaikkan, menurunkan suhu. Informasi tentang karakter yang berisi instruksi yang diterima oleh Arduino disajikan pada Tabel 3.1.

Tabel 3.1 Tabel Perintah ke Arduino

NO	Karakter	Isi Perintah
1.	'B'	Mematikan AC
2.	'T'	Cek Suhu 1
3.	'R'	Cek Suhu 2
4.	'P'	Set Suhu 29
5.	'Q'	Set Suhu 30
6.	'O'	Set Suhu 28
7.	'C'	Set Suhu 27
8.	'D'	Set Suhu 26
9.	'F'	Set Suhu 25
10.	'G'	Set Suhu 24
11.	'H'	Set Suhu 23
12.	'I'	Set Suhu 22
13.	'J'	Set Suhu 21
14.	'K'	Set Suhu 20
15.	'L'	Set Suhu 19
16.	'M'	Set Suhu 18
17.	'N'	Set Suhu 17

Sistem akan mengambil gambar di ruangan dan menghitung jumlah orang yang terdeteksi setiap 10 menit dan akan mengatur suhu AC ke nilai yang sudah ditentukan. Pengaturan suhu ini, berdasarkan dengan Keputusan Menteri Kesehatan Republik Indonesia Nomor 1405/Menkes/SK/XI/2002, yang

menyatakan bahwa persyaratan udara ruangan yang baik memiliki range suhu berkisar $23^{\circ}\text{C} - 26^{\circ}\text{C}$.

Pengujian dilakukan pada sebuah ruangan dengan luas $74,44 \text{ m}^2$, namun hanya memiliki ruang yang dapat digunakan sebesar $27,68 \text{ m}^2$. Berdasarkan Keputusan Menteri Pekerjaan Umum Nomor 30 Tentang Pedoman Teknis Fasilitas Dan Aksesibilitas Pada Bangunan Gedung Dan Lingkungan, standar ruang gerak per orang adalah $1,6 \text{ m}^2$ maka kapasitas maksimal ruangan ini adalah ± 17 orang. Dari informasi kapasitas ruangan ini, pengaturan suhu AC akan dibagi menjadi 4 kondisi yaitu:

1. Kondisi sedikit (0-5 orang): AC akan dinyalakan pada suhu 25°C .
2. Kondisi sedang (6-10 orang): AC akan dinyalakan pada suhu 24°C .
3. Kondisi banyak (11-17 orang): AC akan dinyalakan pada suhu 23°C .
4. Kondisi lebih dari 17 orang AC akan dinyalakan pada suhu 22°C

Sensor DHT22 akan diletakkan di 2 tempat pada ruangan, yaitu bagian terdekat dan terjauh dari AC. Sensor ini akan mendekripsi suhu aktual selama 5 menit setelah perhitungan jumlah orang. Nilai suhu yang diperoleh dari kedua sensor ini kemudian dirata-ratakan untuk mendapatkan suhu aktual ruangan. Jika suhu ruangan lebih besar atau lebih kecil dari suhu yang diinginkan dengan selisih sebesar n derajat, maka sistem akan menurunkan atau menaikkan suhu sebesar n derajat.

Arduino UNO akan menerima intruksi yang dikirimkan dari Orange Pi 5. Intruksi tersebut meliputi perintah untuk menghidupkan AC, mematikan AC, menaikkan, menurunkan suhu. Arduino UNO kemudian akan menerima intruksi berupa karakter yang merepresentasikan pengontrolan AC. Berikut adalah pseudocode sistem untuk pengiriman perintah ke Arduino UNO untuk mengontrol AC.

While (true) :

```

current_time = get_current_time()
if (current_time >= 7 and current_time < 17):
    number_of_people = detect_people()
    if(number_of_people >= 0 and number_of_people <=
5):
        temp_set('F') // set temp 25

```

```

elif(number_of_people >= 6 and number_of_people <=
10):
    temp_set('G') // set temp 24
elif(number_of_people >= 11 and number_of_people
<=17):
    temp_set('H') // set temp 23
else:
    temp_set('I') // set temp 22
// adjust temperature setting
avg_actual_temp = (read_temp1('T')+read_temp2('R'))/2
if (avg_actual_temp < temp_set):
    temp_set += (temp_set - avg_actual_temp)
elif (avg_actual_temp > temp_set):
    temp_set -= (avg_actual_temp - temp_set)
temp_set = round(temp_set)
avg_actual_temp = (read_temp1('T')+read_temp2('R'))/2
elif(current_time > END_TIME):
    number_of_people = detect_people()
    if (number_of_people == 0):
        AC_OFF('B') // Turned Off The AC
        time.sleep(current_time - next_7am)
    else:
        if(number_of_people >= 0 and number_of_people <= 5):
            temp_set('F') // set temp 25
        elif(number_of_people >= 6 and number_of_people <=
10):
            temp_set('G') // set tem 24
        elif(number_of_people >= 11 and number_of_people <=
17):
            temp_set('H') // set temp 23
        else:
            temp_set('H') // set temp 22
// adjust temperature setting

```

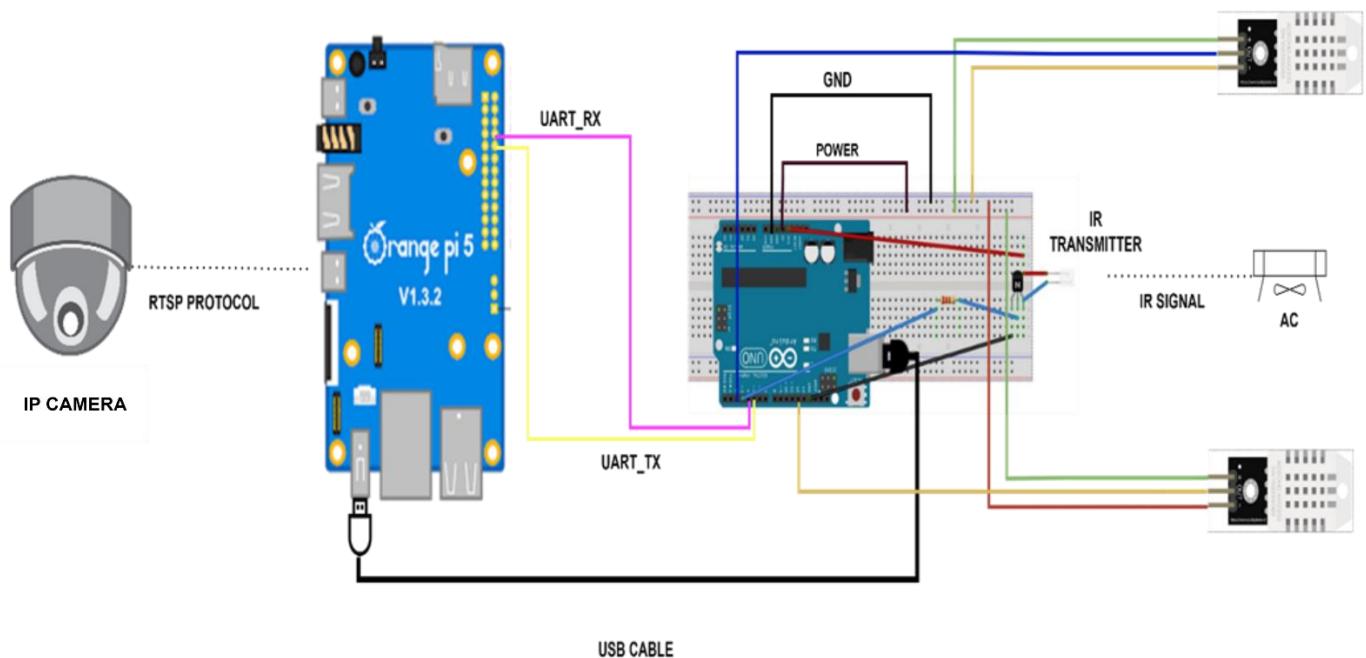
```

avg_actual_temp = (read_temp1('T')+read_temp2('R'))/2
if (avg_actual_temp < temp_set):
    temp_set += (temp_set - avg_actual_temp)
elif (avg_actual_temp > temp_set):
    temp_set -= (avg_actual_temp - temp_set)
temp_set = round(temp_set)
avg_actual_temp = (read_temp1('T')+read_temp2('R'))/2

```

Sistem kontrol AC ditatur untuk menghidupkan, mematikan dan mengontrol suhu AC pada jam yang telah ditentukan, yaitu dari jam 07:00 hingga jam 17:00. Setelah jam 17:00, sistem tetap melakukan pendeksi keberadaan orang. Jika pada jam tersebut tidak ada orang yang terdeteksi, AC akan otomatis dimatikan dan sistem akan menunda pendeksi hingga pukul 07:00 pagi keesokan harinya, sebelum kembali mendeksi dan mengatur suhu AC.

3.4 Rancangan Perangkat Keras



Gambar 3.6 Rancangan Perangkat Keras

Berikut beberapa penjelasan dari beberapa perangkat keras yang digunakan peneliti:

1. IP Camera

IP Kamera EYESEC yang beresolusi 5MP digunakan sebagai perangkat untuk memantau kondisi ruangan. IP Kamera dihubungkan melalui link RTSP (*Real Time*

Streaming Protocol) pada jaringan *Local Area Network* (LAN) dengan Orange Pi 5 agar dapat mengirimkan aliran video secara real time.

2. Orange Pi 5

Perangkat Orange Pi 5 digunakan sebagai perangkat yang diterapkan model YOLOv7 untuk pendekripsi manusia dari kamera yang terhubung oleh link RTSP yang mengambil gambar setiap 10 menit dan mengirim perintah karakter ke Arduino UNO yang akan mengontrol pendingin ruangan (AC) sesuai jumlah orang yang telah terdeteksi.

3. Arduino UNO

Arduino UNO berguna sebagai pengolahan sensor suhu *IR Transmitter* dan akan menerima perintah dari koneksi kabel USB dan UART (*Universal Asynchronous Receiver Transmitter*). Perintah akan dikirim dalam bentuk karakter yang mengandung intruksi untuk mengontrol AC seperti menyalakan AC, mematikan AC, menurunkan suhu, menaikkan suhu dan mendekripsi suhu.

4. DHT22

Dua buah sensor suhu DHT22 digunakan untuk mengukur suhu aktual di dalam ruangan. Kedua sensor tersebut akan dipasang pada jarak terdekat dari AC dan pada jarak terjauh dari AC. Data suhu yang diperoleh dari DHT22 akan digunakan dalam pengaturan suhu AC. Ketika suhu melebihi atau di bawah batas yang ditentukan, sistem akan mengatur suhu AC sesuai dengan selisih suhu aktual.

5. Breadboard

Breadboard digunakan untuk merakit dan menghubungkan sensor DHT22, IR Transmitter dan IR Receiver. Penggunaan breadboard memungkinkan tata letak pengkabelan Arduino UNO menjadi lebih teratur dan efisien.

6. IR Receiver

IR Receiver digunakan sebagai sensor penerima data remote AC untuk pengontrolan AC.

7. IR Transmitter

IR transmitter berfungsi untuk mengirimkan sinyal infrared ke AC untuk melakukan pengontrolan *Air Conditioner*. IR transmitter juga dihubungkan dengan resistor dan transistor hal ini bertujuan untuk mengatur dan memastikan daya listrik yang diterima oleh IR Transmitter sesuai dengan yang dibutuhkannya.

BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan disajikan hasil dari analisis sistem yang telah dirancang dan dilakukan pada Bab 3. Hasil yang akan didapatkan meliputi pendekripsi manusia dan pengendalian sistem pendingin ruangan (*Air Conditioner*).

4.1 Perangkat Keras dan Perangkat Lunak

A. Orange Pi 5

1. Processor : 4xCortex-A76 + 4xCortex-A55
2. Sistem Operasi : Ubuntu
3. GPU : ARM Mali-G
4. NPU : 6 Tops
5. Kapasitas Memori : 64 Gb
6. Sumber Daya : Type-C 5v, 4A
7. Thonny IDE versi 4.14

B. Arduino UNO

1. Sumber Daya : 5V
2. 14 Pin IO
3. Kapasitas memori : 32KB
4. Arduino IDE versi 18.1.103

4.2 Pengimplementasian Model Pretrained YOLOv7

```
# number of classes
nc: 80

# class names
names: ["person", "bicycle", "car", "motorcycle", "airplane", "bus", "train", "truck", "boat", "traffic light",
"fire hydrant", "stop sign", "parking meter", "bench", "bird", "cat", "dog", "horse", "sheep", "cow",
"elephant", "bear", "zebra", "giraffe", "backpack", "umbrella", "handbag", "tie", "suitcase", "frisbee",
"skis", "snowboard", "sports ball", "kite", "baseball bat", "baseball glove", "skateboard", "surfboard",
"tennis racket", "bottle", "wine glass", "cup", "fork", "knife", "spoon", "bowl", "banana", "apple",
"sandwich", "orange", "broccoli", "carrot", "hot dog", "pizza", "donut", "cake", "chair", "couch",
"potted plant", "bed", "dining table", "toilet", "tv", "laptop", "mouse", "remote", "keyboard", "cell phone",
"microwave", "oven", "toaster", "sink", "refrigerator", "book", "clock", "vase", "scissors", "teddy bear",
"hair drier", "toothbrush"]
```

Gambar 4.1 Daftar Class pada Dataset COCO

Model Pre trained YOLOv7 telah dilatih sebelumnya mampu mengenali 80 *class* yang berbeda, seperti yang ditunjukkan pada Gambar 4.1, menunjukkan isi dari file ‘coco.yaml’ yang terdapat daftar *class*s pada dataset COCO yang disimpan di dalam array dengan nama *names*. *Class* pada ‘person’ terdapat pada elemen pertama dalam index 0.

Dalam melakukan penyaringan *class*, digunakan argumen ‘--class 0’. Argumen ini membuat model untuk hanya mendeteksi *class* yang berbeda pada indeks 0 dalam array ‘names’, yaitu *class* ‘*person*’.

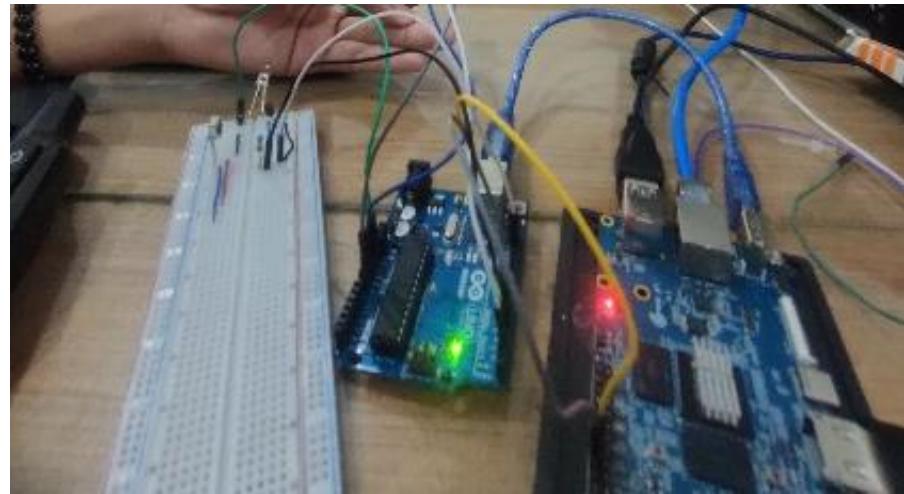
```
detect_command = f"python detect.py --class 0 --weights yolov7.pt --conf 0.25 --img-size 640 --source {photo_path}"
```

Script python tersebut merupakan script untuk menjalankan file ‘detect.py’ dengan menggunakan YOLOv7. Penggunaan –class 0 untuk memfilter class agar yang objek atau kategori yang terdeteksi adalah hanya class ‘*person*’, menggunakan weight dari yolov7.pt, *threshold confidence* deteksi sebesar 0,25 yang artinya jika hasil akurasi kurang dari 0,25 maka *bounding box* tidak akan ditampilkan. Input yang diterima adalah 640*640 piksel dan sumber gambar dari ‘photo_path’ yaitu alamat dari gambar yang sudah di-*capture*.

4.3 Implementasi Perangkat Keras

Melakukan implementasi perangkat keras yaitu Orange Pi 5 dan Arduino UNO. Orange Pi 5 bertugas untuk mengirim perintah ke Arduino UNO untuk menjalankan pendekripsi suhu dan mengontrol *Air Conditioner* berdasarkan kondisi yang didapatkan oleh Orange Pi 5 seperti pendekripsi manusia, pendekripsi suhu dan pengaturan jam operasional Air Conditioner.

2 buah sensor DHT22 masing-masing dihubungkan pada breadboard dengan power 5v dan ground yang dihubungkan pada arduino. Sensor IR transmitter dan IR Receiver juga akan dipasang ke breadboard lalu dihubungkan ke Arduino UNO. Berikut adalah implementasi perangkat keras yang dapat dilihat pada Gambar 4.2.



Gambar 4.2 Implementasi Perangkat Keras

Berdasarkan Gambar 4.2, terlihat implementasi perangkat keras yang digunakan. Perangkat keras diposisikan sekitar 3 meter dari AC dengan IR *transmitter* yang secara langsung mengarah ke AC.

4.3.1 Koneksi Orange Pi 5 dengan Arduino UNO

Dalam penelitian ini menggunakan dua jenis komunikasi serial yaitu menggunakan kabel USB dan kabel UART untuk menghubungkan Orange Pi 5 ke Arduino UNO. Kabel USB dipasang ke port Arduino UNO yang berfungsi untuk mengirimkan data dari Arduino UNO ke Orange Pi 5, sedangkan kabel UART berfungsi untuk mengirimkan data dari Orange Pi 5 ke Arduino UNO.

Untuk dapat mengaktifkan pin UART dari Orange Pi 5 perlu adanya konfigurasi menggunakan library dari WiringOP. Berikut langkah-langkahnya:

1. Mengunduh kode dari library wiringOP

Sudo apt update

Sudo apt install -y git

Git clone <https://github.com/orangepi-xunlong/wiringOP.git-b> next

2. Menjalankan dan menginstall wiringOP

cd wiringOP

sudo ./build clean

sudo ./build

3. Menggunakan perintah ‘gpio readall’ untuk memastikan bahwa pinout siap digunakan.



Gambar 4.5 Koneksi Arduino dengan Orange Pi 5

4.3.2 Konfigurasi IP Kamera

Penelitian ini menggunakan IP kamera EYESEC dalam melakukan pendekripsi manusia. Kamera ini membutuhkan kabel sebagai sumber daya untuk dihidupkan. Kamera ini memiliki frame rate sebesar 30 fps dengan resolusi 5 megapiksel (2560*1920 FHD Resolution).



Gambar 4.6 EYESEC IP Kamera

Untuk mennggunakan IP Kamera ini, diperlukan beberapa langkah-langkah konfigurasi. Berikut langkah-langkah konfigurasi pada IP Kamera untuk mendapatkan link RTSP:

1. Melakukan konfigurasi pada DVR untuk mencari alamat IP kamera

Langkah pertama adalah mengkonfigurasi DVR (Digital Video Recorder) untuk mencari dan mengenali alamat IP kamera yang terhubung. Proses ini

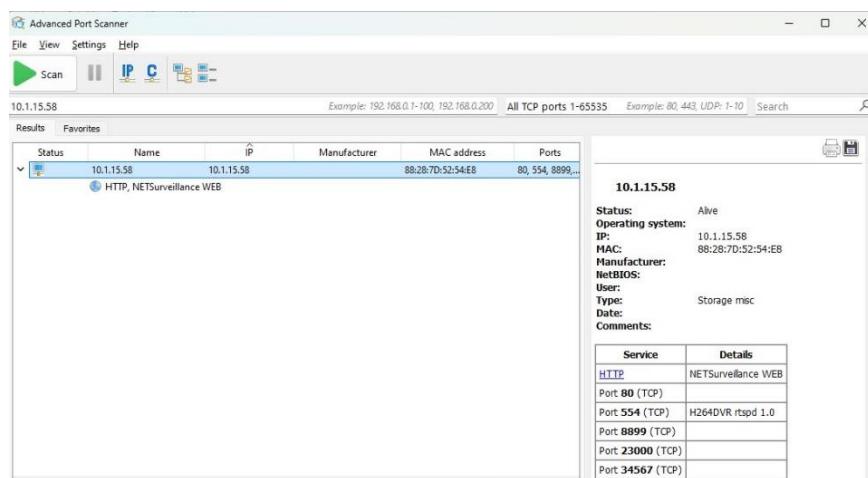
juga mencakup pengaturan username dan password untuk memastikan akses yang aman dan terautentikasi ke kamera.



Gambar 4.7 Konfigurasi pada DVR untuk Mencari Alamat IP

2. Memastikan Layanan dan Port pada IP kamera

Setelah alamat IP kamera ditemukan, pastikan bahwa layanan yang diperlukan untuk komunikasi dengan kamera, seperti RTSP tersedia. Layanan RTSP (*Real Time Streaming Protocol*) menggunakan port ‘554’ sebagai port standar.

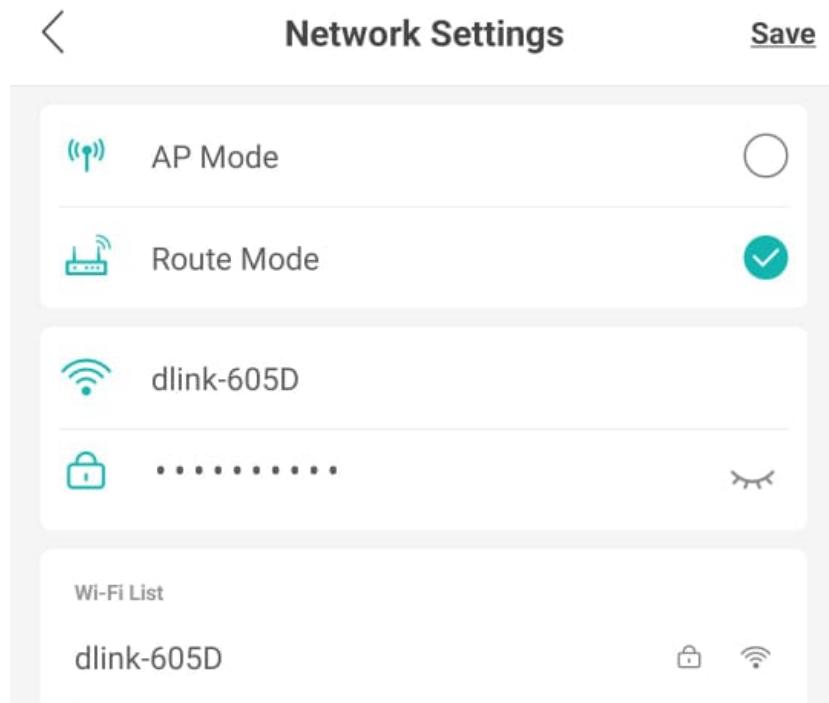


Gambar 4.8 Memastikan Layanan dan Port pada IP Kamera

3. Pengaturan Koneksi Jaringan IP Kamera

Untuk dapat menghubungkan kamera dengan link RTSP, Orange Pi 5 dihubungkan wifi yang sama dengan IP kamera. Akses koneksi jaringan IP kamera dapat dilakukan melalui aplikasi iCSee, dengan syarat kamera harus dihubungkan dengan wifi yang sama dengan Orange Pi 5. Dalam konteks ini,

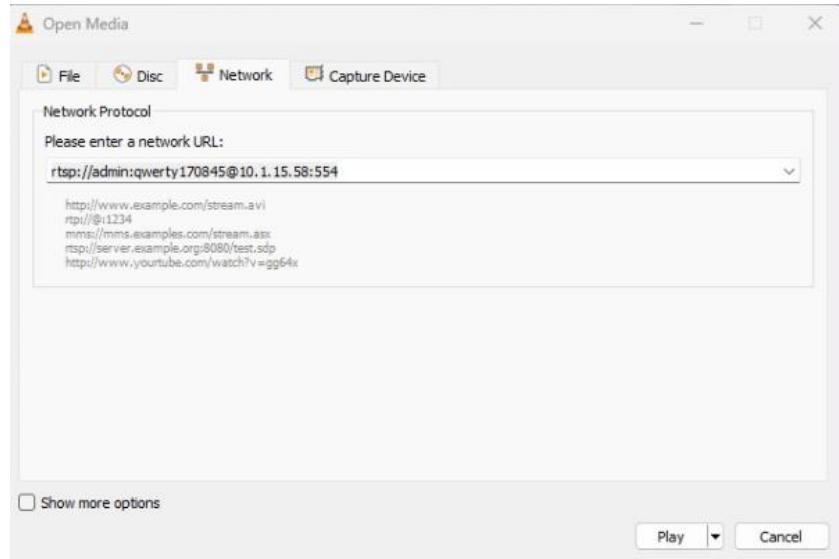
penulis menggunakan wifi yang tersedia yaitu dlink-605D untuk mengkoneksikan antara Orange Pi 5 dan IP Kamera.



Gambar 4.9 Pengaturan Koneksi Jaringan IP Kamera pada Aplikasi Icsee

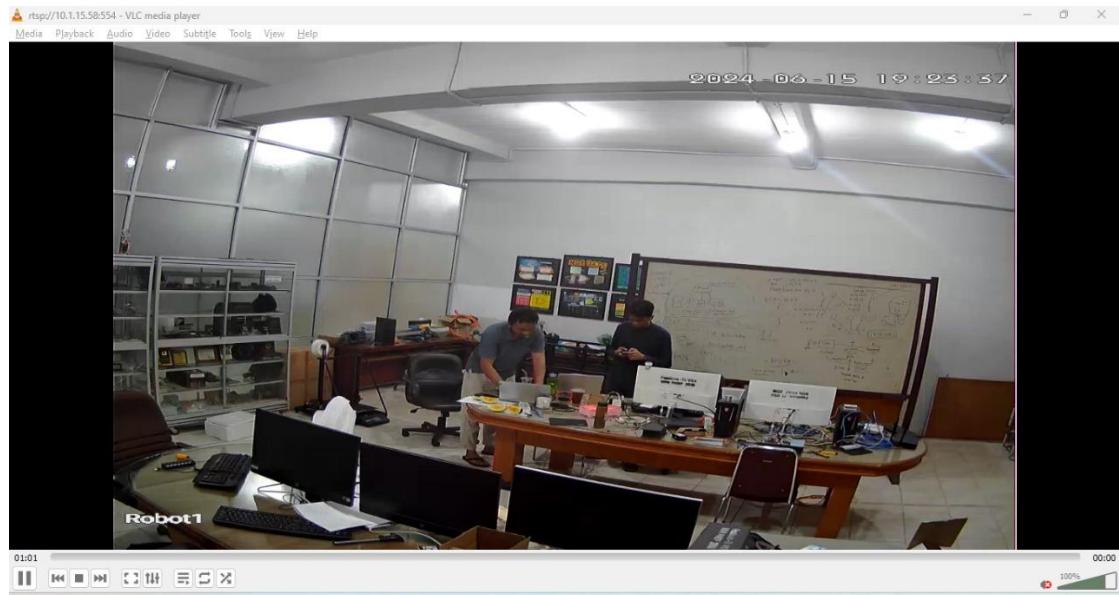
4. Pengaturan URL RTSP pada IP kamera

Setelah memastikan layanan dan port dari IP Kamera, langkah selanjutnya adalah memasukkan data yang didapat seperti nama pengguna yaitu ‘admin’, lalu qwerty170845 sebagai *password*, alamat IP 10.15.58, dan port RTSP 554 ke dalam format URL RTSP, sehingga URL RTSP nya adalah rtsp://admin:qwerty170845@10.1.15.58:554. URL kamera yang telah didapatkan akan dapat diakses dan dikonfigurasi menggunakan VLC media player pada menu network.



Gambar 4.10 Pengaturan URL RTSP pada IP Kamera

Setelah semua konfigurasi selesai, berikut adalah tampilan RTSP kamera pada Gambar 4.11.

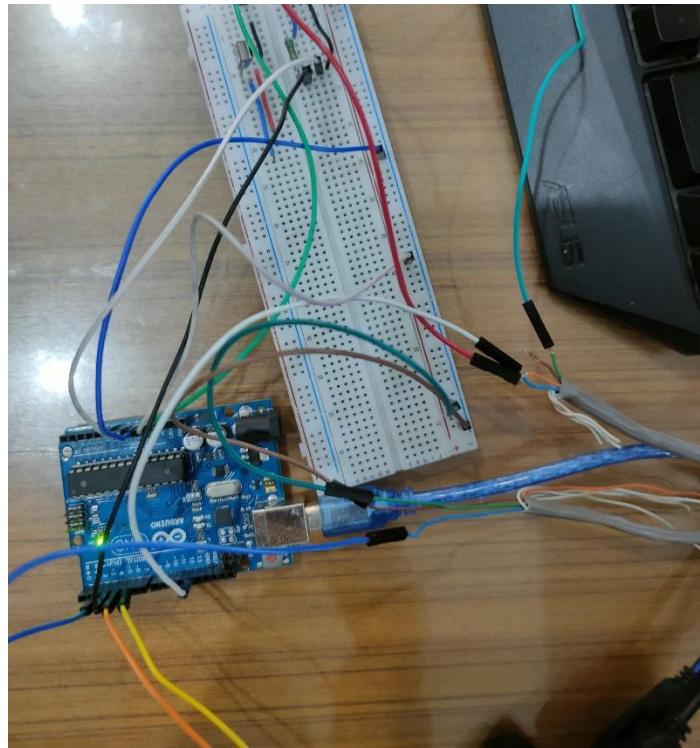


Gambar 4.11 Tampilan IP Kamera

4.3.3 Koneksi Arduino UNO dengan DHT22

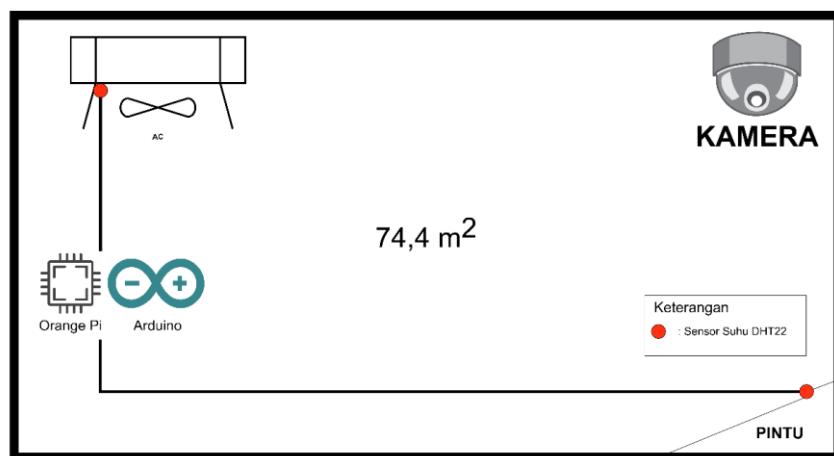
Menghubungkan Arduino UNO dengan dua sensor suhu DHT22 dengan cara menghubungkan pin (+) dari kedua sensor DHT22 ke pin 5v Arduino UNO melalui breadboard, pin (-) dari kedua sensor DHT22 ke pin GND Arduino UNO melalui breadboard, pin OUT dari sensor suhu pertama dihubungkan ke pin 2 Arduino UNO, dan pin OUT dari sensor suhu kedua dihubungkan ke pin 12

Arduino UNO. Adapun gambar koneksi Arduino UNO dengan DHT22 dapat dilihat pada Gambar 4.12



Gambar 4.12 Koneksi Arduino UNO dengan DHT22

Dua sensor suhu akan digunakan dalam pengujian ini yang bertujuan untuk mengukur 2 input suhu aktual pada jarak terdekat dan terjauh dari AC. Denah pemasangan sensor suhu DHT22 terdapat pada Gambar 4.13.



Gambar 4.13 Denah Pemasangan Sensor Suhu DHT22

Implementasi pemasangan 2 sensor DHT22 terdapat pada gambar 4.14

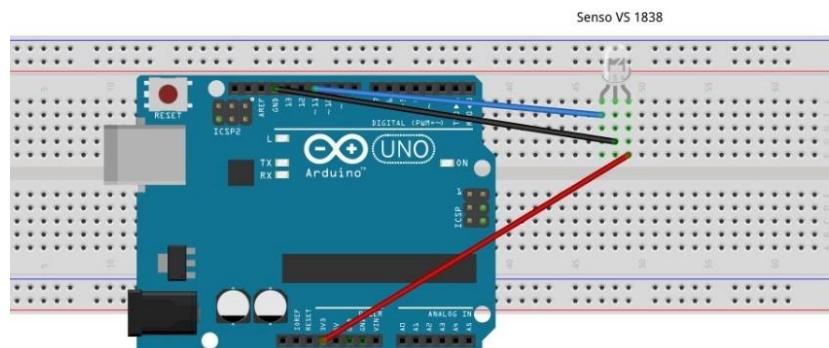


Gambar 4.14 Implementasi Pemasangan 2 Sensor DHT22

Gambar 4.14(a) menunjukkan pemasangan sensor suhu DHT22 terdekat dari AC, sedangkan Gambar 4.14(b) menunjukkan pemasangan sensor suhu DHT22 terjauh dari AC. Dua sensor DHT22 dihubungkan dengan kabel LAN yang masing-masing panjangnya 3 meter dan 10 meter. Sensor suhu terdekat dari AC akan direpresentasikan dengan karakter ‘T’ dan sensor suhu terjauh dari AC akan direpresentasikan dengan karakter ‘R’.

4.3.4 Koneksi Arduino UNO dengan IR Receiver

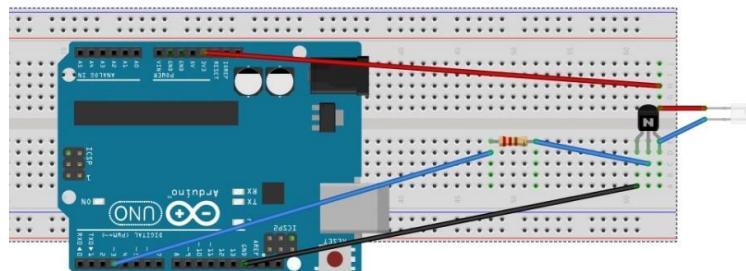
Arduino UNO dapat terkoneksi dengan sensor IR Receiver menggunakan pin-pin yang tersedia pada IR Receiver. Dalam penelitian ini, breadboard digunakan sebagai pengganti komponen elektronik, penelitian ini menggunakan breadboard untuk menghubungkan sensor VS1838 ke Arduino UNO. Penataan hubungan pin seperti pin VCC terhubung pada 3,3 v pinout Arduino UNO, lalu pin GND dihubungkan pada pinout GND Arduino UNO dan pin signal dihubungkan pada pin 11 Arduino UNO. Adapun gambar koneksi Arduino UNO dengan IR Receiver VS1838 dapat dilihat pada Gambar 4.15.



Gambar 4.15 Koneksi Arduino UNO dengan IR Receiver VS1838

4.3.5 Koneksi Arduino UNO dengan IR Transmitter

Seperti halnya dengan IR Receiver, IR Transmitter menggunakan breadboard untuk terhubung dengan Arduino UNO. Penataan pin-pin IR *Transmitter* dihubungkan dengan cara pin signal dihubungkan pada pin 3 Arduino UNO, lalu pin vcc dihubungkan dengan 3,3 v pinout Arduino UNO dan pin GND dihubungkan dengan pin GND Arduino UNO. Penggunaan IR Receiver terpisah dengan IR Transmitter karena terbatasnya pinout 3,3 v Arduino UNO. Setelah data remote AC didapatkan, IR transmitter tidak digunakan lagi. Dalam penelitian ini juga menggunakan transistor dan resistor sebagai penguat dan penahan arus listrik, sekaligus mencegah terjadinya konsleting pada IR *Transmitter*.



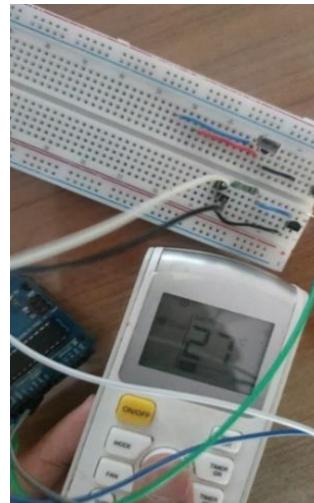
Gambar 4.16 Koneksi Arduino UNO dengan IR Transmitter

4.4 Pengujian Sistem

Sistem yang telah dibuat akan dilakukan implementasi untuk mendapatkan hasil dari proses pendekripsi manusia untuk mengontrol pendingin ruangan (*Air Conditioner*). Pada pengujian ini meliputi pengujian IR *receiver* untuk pengambilan data remote, pengujian DHT22, pengujian deteksi manusia hingga pengujian AC berdasarkan deteksi jumlah manusia.

4.4.1 Pengujian IR Receiver

Pengujian dilakukan dengan mendekatkan remote AC ke sensor IR *Receiver* VS1838, lalu menekan tombol suhu yang diperlukan. Hal ini bertujuan untuk memastikan bahwa sensor IR *Receiver* dapat menerima dan merespons sinyal inframerah yang dikirimkan oleh remote AC.



Gambar 4.17 Proses Pengambilan Data Remote AC

Setelah tombol AC di tekan, serial monitor pada arduino IDE akan mengeluarkan output data yang merepresentasikan tampilan dari remote AC. Dilakukan pengujian pada tombol tombol 27, lalu pada tampilan remote AC terdapat mode cool dan fan Auto. Ini berarti tombol ini merepresentasikan tombol suhu 27, mode cool dan fan auto. Data pada tombol 27 terdapat pada Gambar 4.18.

```
Output Serial Monitor ×
Message (Enter to send message to 'Arduino Uno' on 'COM3')
+ 400,- 600 + 400,- 600 + 400,- 600 + 400,- 600
+ 450,-1700 + 450,-1650 + 500,-1650 + 500,- 550
+ 500,-1650 + 500,-1650 + 450,-1700 + 450,-1650
+ 500,-5200 +4300,-4450 + 500,-1650 + 450,- 600
+ 500,-1650 + 450,-1700 + 450,- 600 + 450,- 600
+ 500,-1650 + 450,- 600 + 500,- 600 + 450,-1650
+ 500,- 600 + 450,- 600 + 500,-1650 + 450,-1650
+ 500,- 600 + 450,-1700 + 450,-1650 + 500,- 600
+ 450,- 600 + 450,-1700 + 450,-1650 + 500,-1650
+ 450,-1700 + 450,-1650 + 500,- 600 + 450,-1650
+ 500,-1650 + 500,- 550 + 500,- 600 + 450,- 600
+ 500,- 600 + 450,- 600 + 450,- 600 + 500,- 600
+ 450,- 600 + 450,-1700 + 450,- 600 + 450,- 600
+ 500,- 600 + 450,- 600 + 450,-1700 + 450,-1650
+ 500,-1650 + 500,- 550 + 500,-1650 + 500,-1650
+ 450,-1700 + 450,-1650 + 500,- 550
Result as internal 8bit ticks (50 us) array - compensated with MARK_EXCESS_MICROS=20
uint8_t rawTicks[199] = {86,89,10,33,9,12,10,33,9,34,9,12,9,12,10,33,9,12,10,12,9,33,10,12,9,12,10,33,9,33,10,12,9,33,10,
Result as microseconds array - compensated with MARK_EXCESS_MICROS=20
uint16_t rawData[199] = {4280,470,480,1670,430,620,480,1670,430,1720,430,620,430,620,480,1670,430,620,480,620,490,1670,480,620
Fronto Hex as string
char prontoData[] = "0000 006D 0064 0000 00A6 00AA 0014 003F 0012 0016 0014 003F 0012 0041 0012 0016 0012 0016 0014 003F 0012 0016 0014 003F
```

Gambar 4.18 Data Pada Tombol 27

4.4.2 Pengujian DHT22

Pengujian DHT22 dilakukan menggunakan file DHTTester.ino yang berfungsi untuk menampilkan output *humidity* atau kelembapan, *temperature* atau suhu dalam *celsius* dan *farenheit*, lalu juga menampilkan *Heat Index*. Pengujian ini dilakukan untuk memastikan bahwa DHT22 berjalan dengan baik.

```

[{"id": 1, "label": "person", "bbox_2d": [298, 124, 778, 328], "confidence": 0.77}, {"id": 2, "label": "person", "bbox_2d": [500, 124, 778, 328], "confidence": 0.56}, {"id": 3, "label": "person", "bbox_2d": [600, 124, 778, 328], "confidence": 0.31}], [{"id": 1, "label": "person", "bbox_2d": [298, 124, 778, 328], "confidence": 0.77}, {"id": 2, "label": "person", "bbox_2d": [500, 124, 778, 328], "confidence": 0.56}, {"id": 3, "label": "person", "bbox_2d": [600, 124, 778, 328], "confidence": 0.31}]

```

Gambar 4.19 Pengujian DHT22 Menggunakan DHTtester.ino

4.4.3 Pengujian Deteksi Manusia

Pada tahap ini dilakukan pengujian pendekripsi manusia menggunakan IP kamera. Sistem akan mengambil gambar setiap 10 menit untuk melakukan pendekripsi manusia di ruangan. Pengujian ini dilakukan untuk melihat hasil kemampuan model YOLOv7 dalam mendekripsi manusia. Hasil pendekripsi manusia terdapat pada Tabel 4.1.

Tabel 4.1 Hasil Pengujian Pendekripsi Manusia

NO	Hasil Pendekripsi dan Tampilan Terminal
1	 <p>Hasil Pendekripsi Menampilkan 3 Bounding Box</p>

	<pre>orangeipi@orangeipi5: ~/Documents/YOLOV7_fix orangeipi@orangeipi5: ~/Documents/YOLOV7_fix 80x24 RepConv.fuse_repvgg_block /usr/local/lib/python3.10/dist-packages/torch/functional.py:512: UserWarning: torch.meshgrid: in an upcoming release, it will be required to pass the indexing argument. (Triggered internally at /pytorch/aten/src/ATen/native/TensorShape.cpp:3587.) return _VF.meshgrid(tensors, **kwargs) # type: ignore[attr-defined] Model Summary: 306 layers, 36905341 parameters, 6652669 gradients, 104.5 GFLOPS 3 persons, Done. (2927.9ms) Inference, (2.0ms) NMS Jumlah deteksi person: 3 The image with the result is saved in: runs/detect/exp210/captured_photo.jpg Jumlah orang pada ruangan adalah: 3 orang</pre> <p>Terminal menampilkan pesan ‘jumlah orang pada ruangan adalah : 3 orang’</p>
2	 <p>Hasil pendekripsi menunjukkan 5 bounding box dengan kondisi ruangan 5 orang</p>

3.	 <p>Hasil pendeksi menunjukkan 2 bounding box dengan kondisi ruangan 2 orang</p>
	<pre>orangepi@orangepi5: ~/Documents/YOLOV7_fix orangepi@orangepi5: ~/Documents/YOLOV7_fix 80x24 return _VF.meshgrid(tensors, **kwargs) # type: ignore[attr-defined] Model Summary: 306 layers, 36905341 parameters, 6652669 gradients, 104.5 GFLOPS 2 persons, Done. (3180.6ms) Inference, (4.5ms) NMS Jumlah deteksi person: 2 The image with the result is saved in: runs/detect/exp393/Hasil_Deteksi.jpg Jumlah orang pada ruangan adalah: 2 orang</pre>
4.	 <p>Terminal menampilkan pesan 'Jumlah orang pada ruangan adalah: 2 orang'</p>

	<p>Hasil pendekstian menunjukkan 3 bounding box dengan kondisi ruangan 3 orang</p> <pre>orangepi@orangepi5: ~/Documents/YOLOV7_fix orangepi@orangepi5: ~/Documents/YOLOV7_fix 80x24 RepConv.fuse_repvgg_block /usr/local/lib/python3.10/dist-packages/torch/functional.py:512: UserWarning: torch.meshgrid: in an upcoming release, it will be required to pass the indexing argument. (Triggered internally at /pytorch/aten/src/ATen/native/TensorShape.cpp:3587.) return _VF.meshgrid(tensors, **kwargs) # type: ignore[attr-defined] Model Summary: 306 layers, 36905341 parameters, 6652669 gradients, 104.5 GFLOPS 3 persons, Done. (2927.9ms) Inference, (2.0ms) NMS Jumlah deteksi person: 3 The image with the result is saved in: runs/detect/exp210/captured_photo.jpg Jumlah orang pada ruangan adalah: 3 orang</pre> <p>Terminal menampilkan pesan ‘Jumlah orang pada ruangan adalah: 3 orang’</p>
5.	 <p>Hasil pendekstian menunjukkan 0 bounding box dengan kondisi ruangan 0 orang</p> <pre>RepConv.fuse_repvgg_block /usr/local/lib/python3.10/dist-packages/torch/functional.py:512: UserWarning: torch.meshgrid: in an upcoming release, it will be required to pass the indexing argument. (Triggered internally at /pytorch/aten/src/ATen/native/TensorShape.cpp:3587.) return _VF.meshgrid(tensors, **kwargs) # type: ignore[attr-defined] Model Summary: 306 layers, 36905341 parameters, 6652669 gradients, 104.5 GFLOPS Done. (3122.1ms) Inference, (1.5ms) NMS Jumlah deteksi person: 0 The image with the result is saved in: runs/detect/exp396/Hasil_Deteksi.jpg Jumlah orang pada ruangan adalah: 0 orang</pre> <p>Terminal menampilkan pesan ‘Jumlah orang pada ruangan adalah: 0 orang’</p>

Berdasarkan Tabel 4.1 menunjukkan kemampuan model *pretrained* YOLOv7 dalam mendekksi manusia diruangan yang terdeteksi dari IP kamera. Terminal melakukan

pembacaan dari file ‘jumlah.txt’ untuk dapat mengambil nilai perhitungan jumlah *bounding box*. Sebagai contoh ketika, pendekripsi menunjukkan 3 bounding box, selanjutnya sistem akan menulis pada file ‘jumlah.txt’ dengan angka 3, lalu terminal akan menampilkan pesan ‘Jumlah orang pada ruangan adalah: 3 orang’ dan hasil serupa terjadi dengan deteksi 5, 2 dan 0 bounding box yang masing-masing mencerminkan jumlah orang di ruangan yang terdeteksi oleh kamera.

4.4.4 Pengujian AC

Pengujian dilakukan di ruangan Laboratorium Sistem Komputer dan Keamanan Jaringan, Program Studi Ilmu Komputer, Universitas Sumatera Utara. Ruangan ini memiliki luas 74,4 m² yang memiliki AC dengan spesifikasi AC yang disajikan pada Tabel 4.2.

Tabel 4.2 Spesifikasi AC

Nama AC	Panasonic CS-ZN9WKP
Tipe AC	Standard
Daya Listrik	824 watt
Berat	8 kg
Daya PK	1 PK
Kapasitas Pendinginan	9000 BTU/h
Tipe Refigerant	R-32

Dilakukan pengujian pengontrolan *Air Conditioner* bertujuan untuk memastikan apakah IR *transmitter* berhasil mengontrol AC sesuai hasil pendekripsi oleh IP kamera. Pengujian ini dilakukan dengan 5 kondisi yaitu:

- Kondisi sedikit (0-5 orang): AC akan dinyalakan pada suhu 25 °C.
- Kondisi sedang (6-10 orang): AC akan dinyalakan pada suhu 24 °C.
- Kondisi banyak (11-17 orang): AC akan dinyalakan pada suhu 23 °C.
- Kondisi lebih dari 17 orang AC akan dinyalakan pada suhu 22 °C
- Kondisi sudah melewati jam 5 dan hasil pendekripsi tidak mendekripsi orang maka AC akan dimatikan dan akan hidup kembali pada pukul 7.

1. Kondisi sedikit (0-5 orang) : AC akan dinyalakan pada suhu 25 °C.

The left terminal window shows the execution of a script named 'YOLOV7_fix' which performs object detection. It outputs the following text:

```
Jumlah deteksi manusia: 0
The image with the result is saved in: hasil_deteksi/Hasil_Deteksi.jpeg
Results saved to hasil_deteksi
Jumlah orang pada ruangan adalah: 0 orang
Data 'F' dikirim ke Arduino
Balasan dari Arduino:
suhu setting awal = 25
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.30
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 27.10
suhu 1 = 23.3, suhu 2 = 27.1, suhu rata-rata
= 25.200000000000003
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.40
suhu 1 = 22.9, suhu 2 = 26.4, suhu rata-rata
= 24.65
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.40
suhu 1 = 22.9, suhu 2 = 26.4, suhu rata-rata
= 24.65
```

The right terminal window shows the serial communication interface with the Arduino. It displays a series of timestamped messages:

```
07:00:16.603 -> F
07:00:16.768 -> suhu 25
07:00:20.622 -> T
07:00:20.622 -> 23.30
07:00:24.602 -> R
07:00:24.602 -> 27.10
07:00:28.697 -> T
07:00:28.697 -> 22.90
07:00:32.707 -> R
07:00:32.707 -> 26.40
07:00:36.817 -> T
07:00:36.817 -> 22.90
07:00:40.803 -> R
07:00:40.803 -> 26.40
07:00:44.883 -> T
07:00:44.883 -> 22.90
```

Gambar 4.20 Ruangan dalam Kondisi Sedikit

Pada jam 07.00, sistem hidup dan memulai pendekripsi dan mendekripsi jumlah manusia pada ruangan sebanyak 0 orang dan sistem mengirimkan karakter ‘F’ untuk mengontrol suhu AC sebesar 25 °C. Selanjutnya sistem mengirimkan karakter ‘T’ dan ‘R’ untuk mendapatkan suhu aktual dari rata-rata suhu jarak terdekat dan terjauh dari AC yaitu sebesar 25.20 °C.

The left terminal window shows the execution of a script named 'YOLOV7_fix' which performs object detection. It outputs the following text:

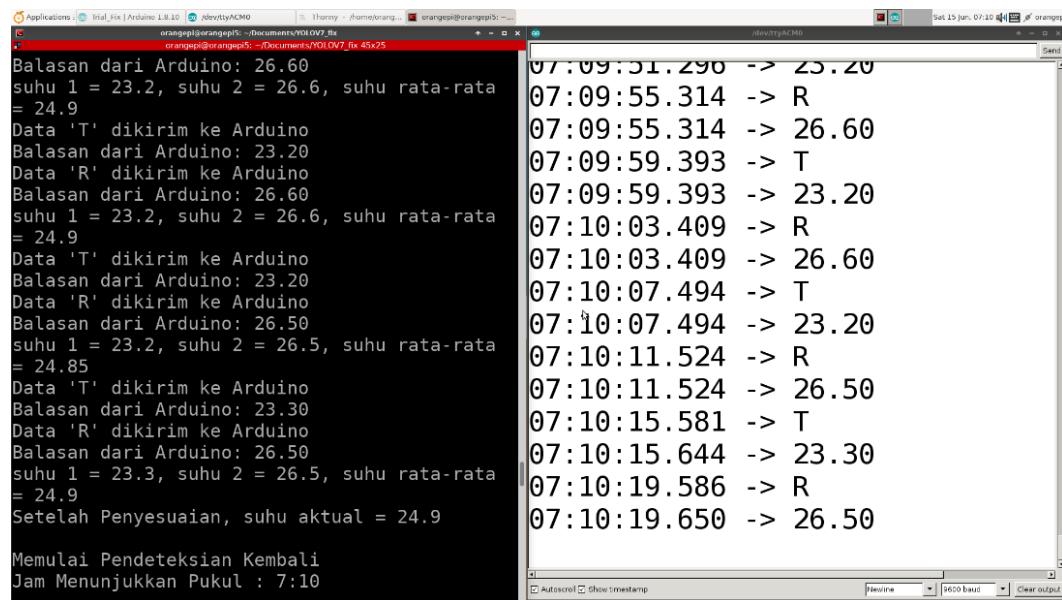
```
Balasan dari Arduino: 24.30
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.70
suhu 1 = 24.3, suhu 2 = 26.7, suhu rata-rata
= 25.5
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 24.40
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.70
suhu 1 = 24.4, suhu 2 = 26.7, suhu rata-rata
= 25.5499999999997
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 24.40
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.70
suhu 1 = 24.4, suhu 2 = 26.7, suhu rata-rata
= 25.5499999999997
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 24.40
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 24.4, suhu 2 = 26.8, suhu rata-rata
= 25.6
suhu_setting = 24
Mengirim Data: G ke Arduino
Data 'G' dikirim ke Arduino
```

The right terminal window shows the serial communication interface with the Arduino. It displays a series of timestamped messages:

```
07:04:51.777 -> R
07:04:51.882 -> 26.70
07:04:55.880 -> T
07:04:55.936 -> 24.40
07:04:59.893 -> R
07:04:59.893 -> 26.70
07:05:03.999 -> T
07:05:03.999 -> 24.40
07:05:07.974 -> R
07:05:08.008 -> 26.70
07:05:12.077 -> T
07:05:12.077 -> 24.40
07:05:16.086 -> R
07:05:16.086 -> 26.80
07:05:20.091 -> G
07:05:20.294 -> suhu 24
```

Gambar 4.21 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Sedikit

5 menit setelah pengaturan suhu awal, pada jam 07.05 rata-rata suhu aktual terdeteksi sebesar 25.6 °C. Hal ini menunjukkan bahwa suhu aktual lebih besar dari suhu setting awal, yaitu 25 °C. Suhu aktual dan suhu setting memiliki selisih sebesar 0.6 °C yang dibulatkan menjadi 1 . Hasilnya suhu setting yang awalnya 25 °C dikurangi dengan 1 °C menjadi 24°C yang akan direpresentasikan oleh karakter G. Selanjutnya sistem kembali mendeteksi suhu rata-rata aktual selama 5 menit.



```

Applications: trial_fix | Arduino: 1.8.10 | /dev/ttyACM0 | Thony - /home/orangepi | orangepi@orangepi: ~/Documents/YOLOV7_fix_45x25
orangepi@orangepi:~/Documents/YOLOV7_fix_45x25

Balasan dari Arduino: 26.60
suhu 1 = 23.2, suhu 2 = 26.6, suhu rata-rata
= 24.9
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.20
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.60
suhu 1 = 23.2, suhu 2 = 26.6, suhu rata-rata
= 24.9
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.20
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 23.2, suhu 2 = 26.5, suhu rata-rata
= 24.85
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.30
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 23.3, suhu 2 = 26.5, suhu rata-rata
= 24.9
Setelah Penyesuaian, suhu aktual = 24.9

Memulai Pendekripsi Kembali
Jam Menunjukkan Pukul : 7:10

07:09:51.290 -> 25.20
07:09:55.314 -> R
07:09:55.314 -> 26.60
07:09:59.393 -> T
07:09:59.393 -> 23.20
07:10:03.409 -> R
07:10:03.409 -> 26.60
07:10:07.494 -> T
07:10:07.494 -> 23.20
07:10:11.524 -> R
07:10:11.524 -> 26.50
07:10:15.581 -> T
07:10:15.644 -> 23.30
07:10:19.586 -> R
07:10:19.650 -> 26.50

```

Gambar 4.22 Rata-Rata Suhu Aktual setelah Penyesuaian Suhu pada Ruangan dalam Kondisi Sedikit

5 menit setelah penyesuaian suhu ke 24 °C, terjadi penurunan rata-rata suhu aktual sebesar 0.7 °C. Sebelum penyesuaian, rata-rata suhu aktual terdeteksi 25.6 °C dan setelah 5 menit turun menjadi 24.9 °C. Selanjutnya terminal akan memulai kembali proses pendekripsi manusia di ruangan.

2. Kondisi sedang (6-10 orang): AC akan dinyalakan pada suhu 24 °C.

```

Jumlah orang pada ruangan adalah: 7 orang
Data 'G' dikirim ke Arduino
Balasan dari Arduino:
suhu setting awal = 24
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.80
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.70
suhu 1 = 22.8, suhu 2 = 26.7, suhu rata-rata = 24.75
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 25.80
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 25.8, suhu 2 = 26.5, suhu rata-rata = 26.15
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 25.80
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 25.8, suhu 2 = 26.5, suhu rata-rata = 26.15
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 25.70
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.60
suhu 1 = 25.7, suhu 2 = 26.6, suhu rata-rata = 26.15
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 25.60

```

```

10:20:56.832 -> G
10:20:57.032 -> suhu 24
10:21:00.851 -> T
10:21:00.851 -> 22.80
10:21:04.831 -> R
10:21:04.865 -> 26.70
10:21:08.942 -> T
10:21:08.942 -> 25.80
10:21:12.938 -> R
10:21:12.938 -> 26.50
10:21:17.038 -> T
10:21:17.038 -> 25.80
10:21:21.027 -> R
10:21:21.060 -> 26.50
10:21:25.133 -> T
10:21:25.212 -> 25.70

```

Gambar 4.23 Ruangan dalam Kondisi Sedang

Pada jam 10.20 sistem mendeteksi manusia di ruangan sebanyak 7 orang dan mengirim karakter ‘G’ untuk mengontrol suhu AC sebesar 24 °C. Selanjutnya, sistem mengirimkan karakter ‘T’ dan ‘R’ untuk mendapatkan rata-rata suhu aktual dan mendapatkan rata-rata suhu aktual sebesar 24,7 °C.

```

Balasan dari Arduino: 26.70
suhu 1 = 23.1, suhu 2 = 26.7, suhu rata-rata = 24.9
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.70
suhu 1 = 23.1, suhu 2 = 26.7, suhu rata-rata = 24.9
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.1, suhu 2 = 26.8, suhu rata-rata = 24.95000000000003
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.70
suhu 1 = 23.1, suhu 2 = 26.7, suhu rata-rata = 24.9
suhu_setting = 23
Mengirim Data: H ke Arduino
Data 'H' dikirim ke Arduino

```

```

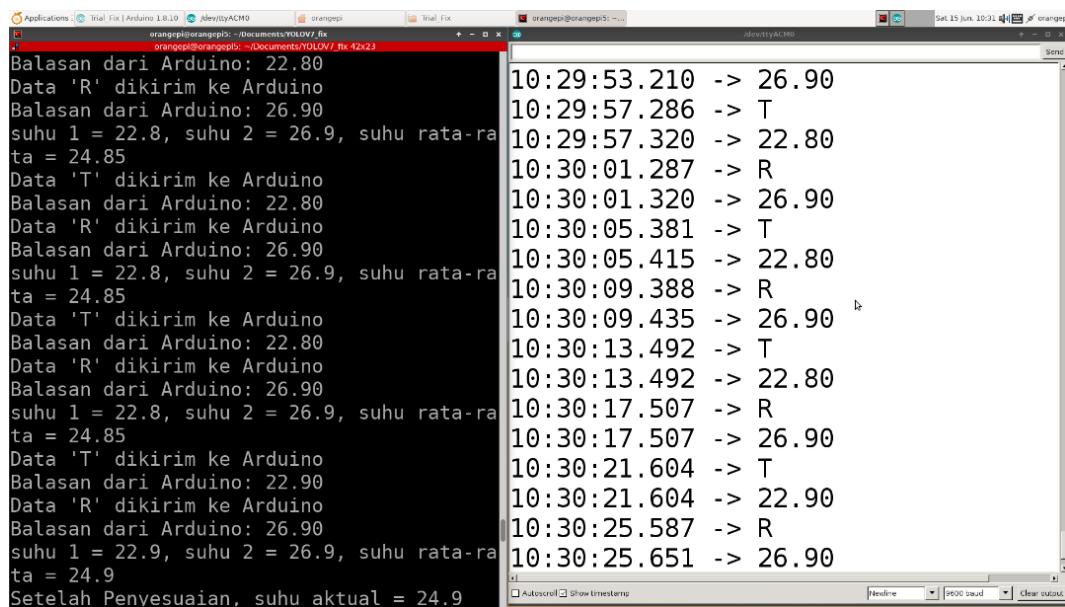
10:24:53.765 -> 23.10
10:24:57.786 -> R
10:24:57.786 -> 26.70
10:25:01.879 -> T
10:25:01.879 -> 23.10
10:25:05.867 -> R
10:25:05.902 -> 26.70
10:25:09.979 -> T
10:25:09.979 -> 23.10
10:25:13.976 -> R
10:25:13.976 -> 26.80
10:25:18.052 -> T
10:25:18.159 -> 23.10
10:25:22.061 -> R
10:25:22.135 -> 26.70
10:25:26.076 -> H
10:25:26.279 -> suhu 23

```

Gambar 4.24 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Sedang

5 menit setelah pengaturan suhu awal, pada jam 10.25 rata-rata suhu aktual terdeteksi sebesar 24,9 °C. Hal ini menunjukkan bahwa suhu aktual lebih besar dari suhu setting awal, yaitu 24 °C. Suhu aktual dan suhu setting memiliki selisih

sebesar 0.9°C yang dibulatkan menjadi 1°C . Hasilnya suhu setting yang awalnya 24°C dikurangi dengan 1°C menjadi 23°C yang direpresentasikan dengan karakter ‘H’. Selanjutnya, sistem kembali melakukan pendekripsi suhu selama 5 menit.

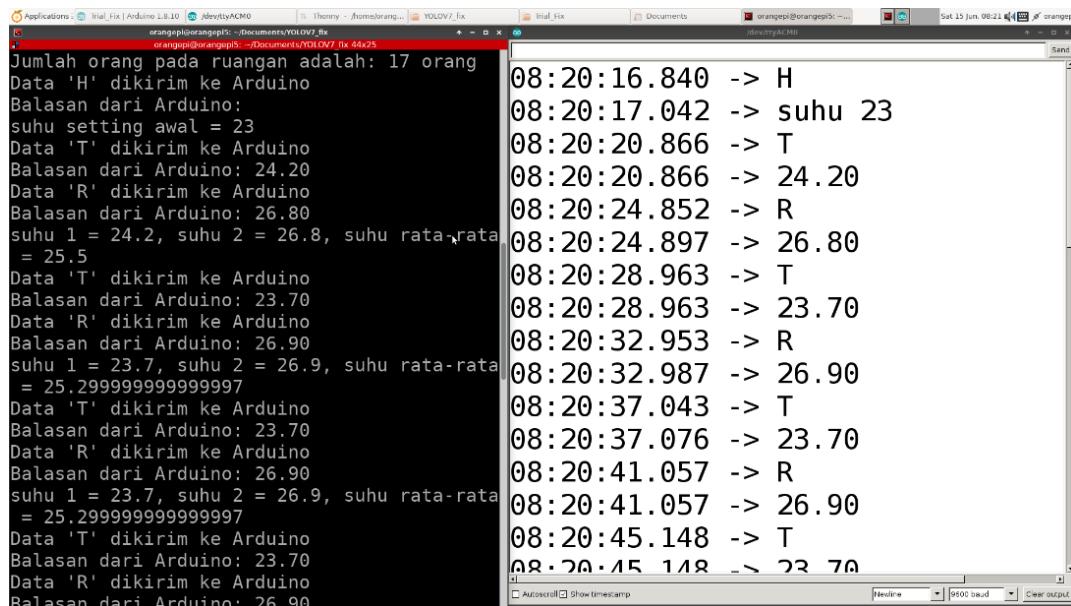


The terminal window shows the Arduino serial output. It reads data from the Arduino, including 'R' for reading temperature, 'T' for sending temperature, and 'H' for setting the temperature to 23. The log viewer next to it shows the received data with timestamps: 10:29:53.210, 10:29:57.286, 10:29:57.320, 10:30:01.287, 10:30:01.320, 10:30:05.381, 10:30:05.415, 10:30:09.388, 10:30:09.435, 10:30:13.492, 10:30:13.492, 10:30:17.507, 10:30:17.507, 10:30:21.604, 10:30:21.604, 10:30:25.587, and 10:30:25.651, all followed by values 26.90, T, R, 22.80, R, 26.90, T, 22.80, R, 22.80, R, 26.90, T, 22.90, R, 26.90, and 26.90 respectively. The final output is 'Setelah Penyesuaian, suhu aktual = 24.9'.

Gambar 4.25 Rata-Rata Suhu Aktual setelah Penyesuaian Suhu pada Ruangan dalam Kondisi Sedang

5 menit setelah penyesuaian suhu ke 23°C , rata-rata suhu aktual menjadi 24.9°C .

3. Kondisi banyak (11-17 orang): AC akan dinyalakan pada suhu 23°C .



The terminal window shows the Arduino serial output. It reads data from the Arduino, including 'H' for setting the temperature to 23. The log viewer next to it shows the received data with timestamps: 08:20:16.840, 08:20:17.042, 08:20:20.866, 08:20:20.866, 08:20:24.852, 08:20:24.897, 08:20:28.963, 08:20:28.963, 08:20:32.953, 08:20:32.987, 08:20:37.043, 08:20:37.076, 08:20:41.057, 08:20:41.057, 08:20:45.148, and 08:20:45.148, all followed by values H, suhu 23, T, 24.20, R, 26.80, T, 23.70, R, 26.90, T, 23.70, R, 26.90, T, 23.70, R, 26.90, and 23.70 respectively. The final output is 'Jumlah orang pada ruangan adalah: 17 orang'.

Gambar 4.26 Ruangan dalam Kondisi Banyak

Pada jam 08.20 sistem mendeteksi jumlah manusia sebanyak 17 orang dan sistem mengirimkan karakter ‘H’ untuk mengontrol suhu sebesar 23 °C. Sistem mengirimkan karakter ‘T’ dan ‘R’ untuk mendapatkan rata-rata suhu aktual dan mendapatkan rata-rata suhu aktual sebesar 25.5 °C.

```

Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.90
suhu 1 = 23.1, suhu 2 = 26.9, suhu rata-rata
= 25.0
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.1, suhu 2 = 26.8, suhu rata-rata
= 24.95000000000003
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.1, suhu 2 = 26.8, suhu rata-rata
= 24.95000000000003
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.1, suhu 2 = 26.8, suhu rata-rata
= 24.95000000000003
suhu_setting = 21
Mengirim Data: J ke Arduino
Data 'J' dikirim ke Arduino
08:24:52.061 -> R
08:24:52.061 -> 26.90
08:24:56.142 -> T
08:24:56.142 -> 23.10
08:25:00.162 -> R
08:25:00.162 -> 26.80
08:25:04.231 -> T
08:25:04.323 -> 23.10
08:25:08.236 -> R
08:25:08.297 -> 26.80
08:25:12.341 -> T
08:25:12.341 -> 23.10
08:25:16.343 -> R
08:25:16.343 -> 26.80
08:25:20.341 -> J
08:25:20.543 -> suhu 21

```

Gambar 4.27 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Banyak

Pada jam 08.25 sistem mendeteksi rata-rata suhu aktual sebesar 24.95 °C. Hal ini menunjukkan bahwa rata-rata suhu aktual lebih besar dari suhu setting awal yaitu 23 °C. Rata-rata suhu aktual dan suhu setting memiliki selisih sebesar 1.95 °C yang dibulatkan menjadi 2 °C. Hasilnya suhu setting yang awalnya 23 °C dikurangi dengan 2 °C menjadi 21°C yang direpresentasikan dengan karakter ‘J’. Selanjutnya, sistem kembali melakukan pendekripsi suhu selama 5 menit.

The terminal window displays sensor readings from an Arduino connected via serial port. The data includes temperature measurements (T) and relative humidity (R) from two sensors, along with a calculated average temperature. The serial monitor on the right shows the same data being transmitted at 9600 baud.

```

Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.00
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 23.0, suhu 2 = 26.5, suhu rata-rata
= 24.75
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 22.9, suhu 2 = 26.5, suhu rata-rata
= 24.7
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 22.9, suhu 2 = 26.5, suhu rata-rata
= 24.7
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.50
suhu 1 = 22.9, suhu 2 = 26.5, suhu rata-rata
= 24.7
Setelah Penyesuaian, suhu aktual = 24.7

```

Gambar 4.28 Rata-Rata Suhu Aktual Setelah Penyesuaian Suhu pada Ruangan dalam Kondisi Banyak

5 menit setelah penyesuaian suhu ke 21 °C, terjadi penurunan rata-rata suhu aktual sebesar 0.25 °C. Sebelum penyesuaian, rata-rata suhu aktual terdeteksi 24.95 °C dan setelah 5 menit turun menjadi 24.7 °C.

4. Kondisi lebih dari 17 orang AC akan dinyalakan pada suhu 22 °C

The terminal window displays sensor readings and a YOLOv7 detection result. The data includes temperature measurements (T) and relative humidity (R) from two sensors, along with a calculated average temperature. The serial monitor on the right shows the same data being transmitted at 9600 baud.

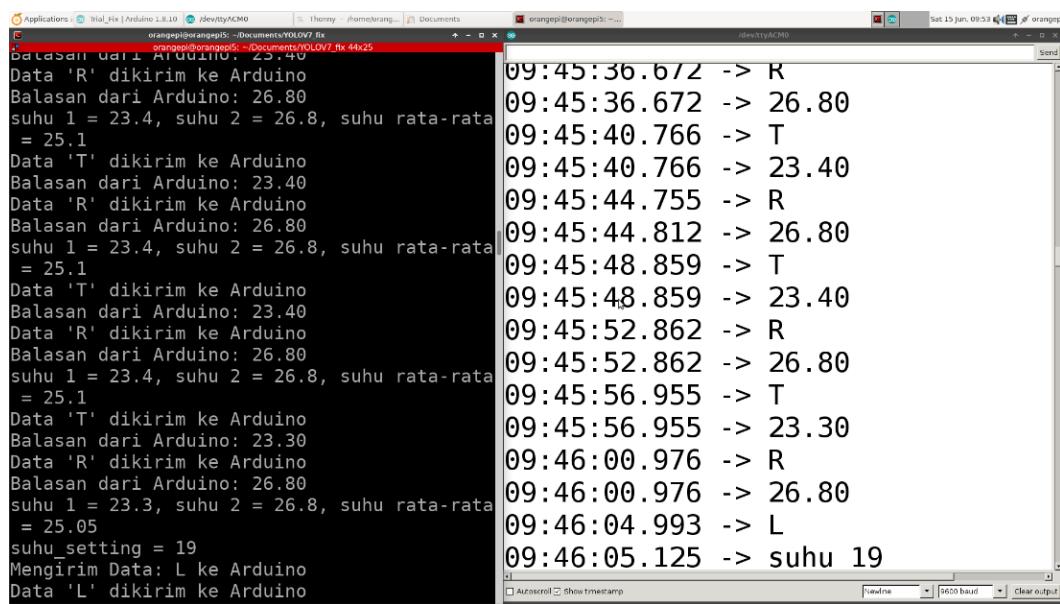
```

Jumlah deteksi manusia: 19
The image with the result is saved in: hasil_deteksi/Hasil_Deteksi.jpg
Results saved to hasil_deteksi
Jumlah orang pada ruangan adalah: 19 orang
Data 'I' dikirim ke Arduino
Balasan dari Arduino:
suhu setting awal = 22
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.10
suhu 1 = 23.1, suhu 2 = 26.1, suhu rata-rata
= 24.6
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 24.00
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.30
suhu 1 = 24.0, suhu 2 = 26.3, suhu rata-rata
= 25.15
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 24.10
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.40
suhu 1 = 24.1, suhu 2 = 26.4, suhu rata-rata
= 25.25

```

Gambar 4.29 Ruangan dalam Kondisi Lebih dari 17 Orang

Pada jam 09.41 sistem mendeteksi manusia di ruangan sebanyak 19 orang dan sistem mengirimkan karakter ‘I’ untuk mengontrol suhu AC sebesar 22 °C. Sistem mendeteksi suhu aktual sebesar 24.6 °C.



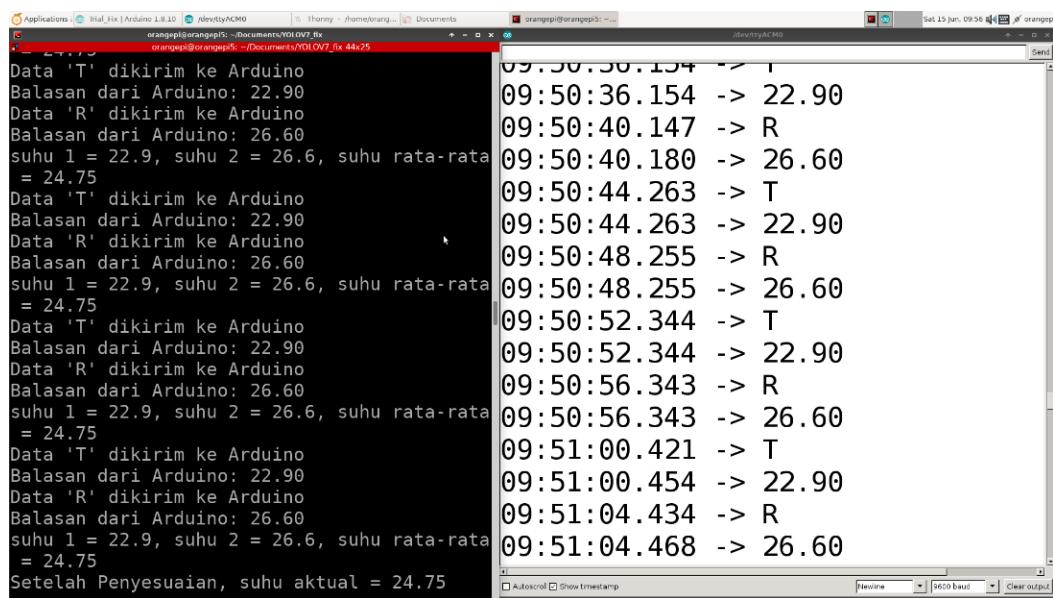
```

Applications: Trial_Fix | Arduino 1.8.10 | /dev/ttyACM0 | Theory - /home/orange... | Documents
orangegepi@orangegepi5: ~Documents/YOLOV7_fix
orangegepi@orangegepi5: ~Documents/YOLOV7_fix 44x25
Balasan dari Arduino: 23.40
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.4, suhu 2 = 26.8, suhu rata-rata
= 25.1
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.40
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.4, suhu 2 = 26.8, suhu rata-rata
= 25.1
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.40
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.4, suhu 2 = 26.8, suhu rata-rata
= 25.1
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 23.30
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.80
suhu 1 = 23.3, suhu 2 = 26.8, suhu rata-rata
= 25.05
suhu_setting = 19
Mengirim Data: L ke Arduino
Data 'L' dikirim ke Arduino
09:45:36.672 -> R
09:45:36.672 -> 26.80
09:45:40.766 -> T
09:45:40.766 -> 23.40
09:45:44.755 -> R
09:45:44.812 -> 26.80
09:45:48.859 -> T
09:45:48.859 -> 23.40
09:45:52.862 -> R
09:45:52.862 -> 26.80
09:45:56.955 -> T
09:45:56.955 -> 23.30
09:46:00.976 -> R
09:46:00.976 -> 26.80
09:46:04.993 -> L
09:46:05.125 -> suhu 19

```

Gambar 4.30 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Lebih dari 17 Orang

5 menit setelah pengaturan suhu awal, sistem mendeteksi rata-rata suhu aktual sebesar 25.05. Hal ini menunjukkan bahwa rata-rata suhu aktual lebih besar dari suhu setting awal yaitu 22 °C. Rata-rata suhu aktual dan suhu setting memiliki selisih sebesar 3.05 °C yang dibulatkan menjadi 3 °C Hasilnya suhu setting yang awalnya 22 °C dikurangi dengan 3 °C menjadi 19°C yang direpresentasikan dengan karakter ‘L’. Selanjutnya, sistem kembali melakukan pendekripsi suhu selama 5 menit.



```

Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.60
suhu 1 = 22.9, suhu 2 = 26.6, suhu rata-rata
= 24.75
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.60
suhu 1 = 22.9, suhu 2 = 26.6, suhu rata-rata
= 24.75
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.60
suhu 1 = 22.9, suhu 2 = 26.6, suhu rata-rata
= 24.75
Data 'T' dikirim ke Arduino
Balasan dari Arduino: 22.90
Data 'R' dikirim ke Arduino
Balasan dari Arduino: 26.60
suhu 1 = 22.9, suhu 2 = 26.6, suhu rata-rata
= 24.75
Setelah Penyesuaian, suhu aktual = 24.75

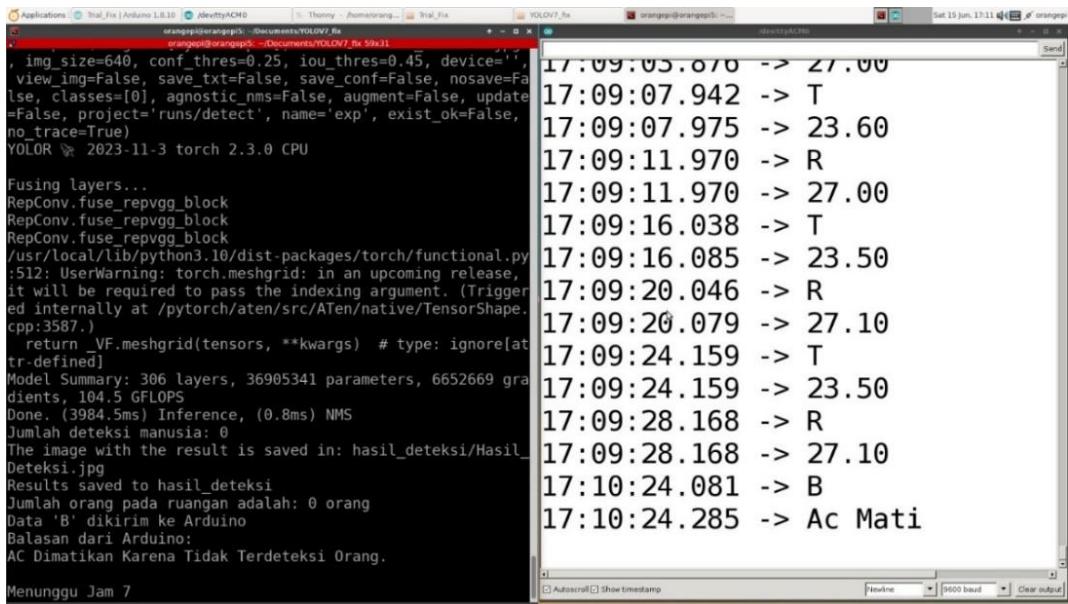
09:50:30.154 -> T
09:50:36.154 -> 22.90
09:50:40.147 -> R
09:50:40.180 -> 26.60
09:50:44.263 -> T
09:50:44.263 -> 22.90
09:50:48.255 -> R
09:50:48.255 -> 26.60
09:50:52.344 -> T
09:50:52.344 -> 22.90
09:50:56.343 -> R
09:50:56.343 -> 26.60
09:51:00.421 -> T
09:51:00.454 -> 22.90
09:51:04.434 -> R
09:51:04.468 -> 26.60

```

Gambar 4.31 Penyesuaian Suhu Setting pada Ruangan dalam Kondisi Lebih dari 17 Orang

5 menit setelah penyesuaian suhu ke 21 °C, terjadi penurunan rata-rata suhu aktual sebesar 0.3 °C. Sebelum penyesuaian, rata-rata suhu aktual terdeteksi 25.05 °C dan setelah 5 menit turun menjadi 24.75 °C

- Kondisi sudah melewati jam 5 dan hasil pendekstian tidak mendekksi orang maka AC akan dimatikan dan akan hidup kembali pada pukul 7.



```

, img_size=640, conf_thres=0.25, iou_thres=0.45, device='',
view_img=False, save_txt=False, save_conf=False, nosave=False,
classes=[0], agnostic_nms=False, augment=False, update=False,
project='runs/detect', name='exp', exist_ok=False,
no_trace=True)
YOLOR > 2023-11-3 torch 2.3.0 CPU

Fusing layers...
RepConv.fuse_repgg_block
RepConv.fuse_repgg_block
RepConv.fuse_repgg_block
/usr/local/lib/python3.10/dist-packages/torch/functional.py:512: UserWarning: torch.meshgrid: in an upcoming release,
it will be required to pass the indexing argument. (Triggered internally at /pytorch/aten/src/ATen/native/TensorShape.cpp:3587.)
    return _VF.meshgrid(tensors, **kwargs) # type: ignore[at-tr-defined]
Model Summary: 306 layers, 36905341 parameters, 6652669 gradients, 104.5 GFLOPS
Done. (3984.5ms) Inference, (0.8ms) NMS
Jumlah deteksi manusia: 0
The image with the result is saved in: hasil_deteksi/Hasil_Deteksi.jpg
Results saved to hasil_deteksi
Jumlah orang pada ruangan adalah: 0 orang
Data 'B' dikirim ke Arduino
Balasan dari Arduino:
AC Dimatikan Karena Tidak Terdeteksi Orang.

Menunggu Jam 7

```

```

17:09:05.070 -> ZT.00
17:09:07.942 -> T
17:09:07.975 -> 23.60
17:09:11.970 -> R
17:09:11.970 -> 27.00
17:09:16.038 -> T
17:09:16.085 -> 23.50
17:09:20.046 -> R
17:09:20.079 -> 27.10
17:09:24.159 -> T
17:09:24.159 -> 23.50
17:09:28.168 -> R
17:09:28.168 -> 27.10
17:10:24.081 -> B
17:10:24.285 -> Ac Mati

```

Gambar 4.32 Pengujian Ketika Waktu Sudah Melebihi Batas yang Ditentukan dan Kamera Tidak Mendekksi Orang di Ruangan

Pada Gambar 4.32 AC ketika sudah melewati jam 5 sistem akan tetap melakukan pendektsian. Kemudian, hasil pendektsian selanjutnya menunjukkan manusia di ruangan sebanyak 0 dan terminal menampilkan pesan ‘Jumlah orang pada ruangan adalah: 0 orang’. Selanjutnya terminal mengirim karakter ‘B’ ke Arduino UNO untuk mematikan AC. Terminal menampilkan pesan ‘AC Dimatikan Karena Tidak Terdeteksi Orang’ dan sistem akan melakukan pendektsian kembali pada pukul jam 7 pagi.

BAB 5

KESIMPULAN DAN SARAN

Pada Bab ini akan dijelaskan kesimpulan yang telah didapatkan dari penelitian pendingin ruangan (*Air Conditioner*) berdasarkan deteksi jumlah manusia, dan beberapa saran yang dibuat penulis agar menjadi acuan untuk penelitian selanjutnya.

5.1 Kesimpulan

Kesimpulan yang didapat dari keseluruhan implementasi metode YOLOv7 berdasarkan hasil penelitian yang telah diterapkan adalah:

1. Model Pre-trained YOLOv7 telah berhasil mendekripsi kelas ‘person’ dengan baik.
2. Sistem terhubung dengan baik antara Orange Pi 5 dan Arduino UNO.
3. Sistem yang telah dibuat dapat mengendalikan AC berdasarkan deteksi jumlah manusia.
4. Sistem mampu melakukan penyesuaian suhu secara otomatis. Saat suhu aktual lebih tinggi dari suhu setting awal, sistem akan menurunkan suhu setting.

5.2 Saran

Beberapa saran yang dijadikan sebagai acuan untuk penelitian selanjutnya adalah:

1. Studi lebih lanjut diharapkan dapat menggunakan metode You Only Look Once versi terbaru.
2. Penelitian selanjutnya diharapkan dapat menggunakan model yang berfokus pada pendekripsi kepala.
3. Penelitian selanjutnya diharapkan dapat memanfaatkan fitur-fitur dari remote AC lainnya.
4. Studi lebih lanjut diharapkan untuk dapat mengendalikan lebih dari satu AC.

5. Studi lebih lanjut diharapkan untuk dapat menerapkan ke CCTV.

DAFTAR PUSTAKA

- Afandi, M. A., Nurandi, S., & Enriko, I. K. A. (2021). Automated Air Conditioner Controller and Monitoring Based on Internet of Things. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 11(1), 83. <https://doi.org/10.22146/ijeis.64563>
- Ardiansyah, A., & Hasan, N. F. (2023). Deteksi dan Klasifikasi Penyakit Pada Daun Kopi Menggunakan Yolov7. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, 12(1), 30–35. <https://doi.org/10.32736/sisfokom.v12i1.1545>
- Arief, L., Akbar, F., Novani, N. P., & Saputra, I. (2018). Jurnal Nasional Teknologi dan Sistem Informasi PENGUJIAN KINERJA SERVER PORTABLE BERBASIS SINGLE BOARD COMPUTER (SBC) DALAM MENDUKUNG KEGIATAN PEMBELAJARAN. *Jurnal Nasional Teknologi Dan Sistem Informasi*. <https://doi.org/10.25077/TEKNOSI.v4i2.2018.098-106>
- Bin Tuan Rashid, T. M. N., & Fadzil, L. M. (2023). Comparative Review of Object Detection Algorithms in Small Single-Board Computers. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(7), 244–252. <https://doi.org/10.17762/ijritcc.v11i7.7904>
- Cepi Muh, U., Rusli, M., Mulya, M. F., & Widyaningsih, T. W. (2022). *Applikasi Sistem Monitoring Server Menggunakan Device Orange Pi Berbasis Web Service Studi Kasus PT. MNC Televisi Indonesia – MNC Group*.
- Fadhli, M., Ketua Dewan Editor Zulfan, Mk., Editor Pelaksana Munawir, M., Baihaqi, M., Sekretaris Yeni Yanti, M., Mitra Bestari Ir Yuwaldi Away, M., Gani, T. A., & Melinda, Me. (2019). *Analisis Performansi Video Streaming Dengan Menggunakan Protokol RTSP Pada Jaringan IEEE 802.11n*.
- Faishal Rahaman, M., & Md Faishal Rahaman, st. (2023). *The Current Trends of Object Detection Algorithms: A Review*. <https://doi.org/10.13140/RG.2.2.19067.49442>
- Fetra, R. (2020). *JTEV (JURNAL TEKNIK ELEKTRO DAN VOKASIONAL) Sistem Otomasi Penyalakan Lampu dan AC (Air Conditioner) pada Ruang Dosen Berbasis Arduino UNO*. <http://ejournal.unp.ac.id/index.php/jtev/index>
- Golwa, G. V., Anggara, F., & Hidayat, I. (2023). Studi potensi listrik dari hembusan angin luaran air conditioner. *Jurnal Teknik Mesin Indonesia*, 18(2), 53–58. <https://doi.org/10.36289/jtmi.v18i2.464>
- Liu, K., Sun, Q., Sun, D., Peng, L., Yang, M., & Wang, N. (2023). Underwater Target Detection Based on Improved YOLOv7. *Journal of Marine Science and Engineering*, 11(3). <https://doi.org/10.3390/jmse11030677>
- Mileanasari, F., Asba, I., Fatih, M., Izza, F., & Rama Annata, A. (2023). Pemantauan dan Pengendalian Suhu Air Conditioner untuk Penghematan Energi Berbasis

- Pengolahan Citra. *Jurnal Teknologi Rekayasa*, 8(2).
<https://doi.org/10.31544/jtera.v8.i2.2023.209-216>
- Ms. Banti Kumari, Ms. Kanika Jindal, & Mr. Amit Bindal. (2024). Design and Implementation of Verilog Based High Speed Low Power UART. *International Research Journal on Advanced Engineering Hub (IRJAEH)*, 2(05), 1468–1477.
<https://doi.org/10.47392/IRJAEH.2024.0203>
- Natsir, M., Bayu Rendra, D., & Derby Yudha Anggara, A. (2019). *IMPLEMENTASI IOT UNTUK SISTEM KENDALI AC OTOMATIS PADA RUANG KELAS DI UNIVERSITAS SERANG RAYA*. 6(1).
<https://www.arduino.cc/en/Products/Counterfeit>
- Novandra Rizkatama, G., Nugroho, A., & Alfa Faridh Suni, dan. (2021). Edu Komputika Journal Sistem Cerdas Penghitung Jumlah Mobil untuk Mengetahui Ketersediaan Lahan Parkir berbasis Python dan YOLO v4. *Edu Komputika*, 8(2).
<http://journal.unnes.ac.id/sju/index.php/edukom>
- Pella, S. I., & Lami, H. (2019). DISAIN SISTEM OTOMASI SUHU RUANGAN PERTEMUAN DENGAN PENERAPAN TEKNIK MACHINE LEARNING. *Jurnal Media Elektro*, VIII(2).
- Purwono, Ma’arif, A., Rahmiani, W., Fathurrahman, H. I. K., Frisky, A. Z. K., & Haq, Q. M. U. (2022). Understanding of Convolutional Neural Network (CNN): A Review. *International Journal of Robotics and Control Systems*, 2(4), 739–748. <https://doi.org/10.31763/ijrcs.v2i4.888>
- Raup, A., Ridwan, W., Khoeriyah, Y., Yuliati Zaqiah, Q., & Islam Negeri Sunan Gunung Djati Bandung, U. (2022). *Deep Learning dan Penerapannya dalam Pembelajaran*. <http://Jiip.stkipyapisdompu.ac.id>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You Only Look Once: Unified, Real-Time Object Detection*. <http://arxiv.org/abs/1506.02640>
- Sarosa, M., & Muna, N. (2021). *IMPLEMENTASI ALGORITMA YOU ONLY LOOK ONCE (YOLO) UNTUK DETEKSI KORBAN BENCANA ALAM*. 8(4).
<https://doi.org/10.25126/jtiik.202184407>
- Shakirovich Ismailov Zafar Botirovich Jo, A. (2022). *Study of arduino microcontroller board*. www.openscience.uz
- Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2022). *YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*. <http://arxiv.org/abs/2207.02696>
- Wang, Y., Wang, H., & Xin, Z. (2022). Efficient Detection Model of Steel Strip Surface Defects Based on YOLO-V7. *IEEE Access*, 10, 133936–133944.
<https://doi.org/10.1109/ACCESS.2022.3230894>

- Widiarto, H., & Darma Kusuma, P. D. (2022). *OTOMATISASI DAN MONITORING AIR CONDITIONER (AC) BERBASIS ARDUINO UNO RUANG SEMINAR GEDUNG TEKNIK PENERBANGAN BARU*.
- Widura, A., Hadiatna, F., & Syamsul Fikri, R. (2023). *IoT-Based Air Conditioning Control System for Energy Saving*. 13(1).
- Wiki-Orange Pi Contributors. (2023). *Orange Pi 5*. Wiki-Orange Pi, .
http://www.orangepi.org/orangepiwiki/index.php?title=Orange_Pi_5&oldid=2348
- Yusniati. (2018). Penggunaan Sensor Infrared Switching Pada Motor DC Satu Phasa. In *Cetak Journal of Electrical Technology* (Vol. 3, Issue 2).
- Zakiyah, A., Lomi, A., & Handoko, F. (2018). Manajemen Energi Penggunaan Pendingin Udara Pada Gedung Perkantoran Universitas Islam Malang. In *Jurnal Teknologi dan Manajemen Industri* (Vol. 4, Issue 2).