

**DETEKSI JUMLAH MANUSIA MENGGUNAKAN METODE
YOLOv4 DALAM SUATU GEDUNG (STUDI KASUS : LAB
ELEKTRO UNISSULA)**

LAPORAN TUGAS AKHIR

Laporan Ini Di susun Untuk Memenuhi Salah Satu Syarat Memperoleh Gelar Stara
Satu (S1) Pada Program Studi Teknik Elektro Universitas Islam Sultan Agung
Semarang



Di susun Oleh:

MOHAMAD SEPRIYANTO UTOMO

NIM: 30601700022

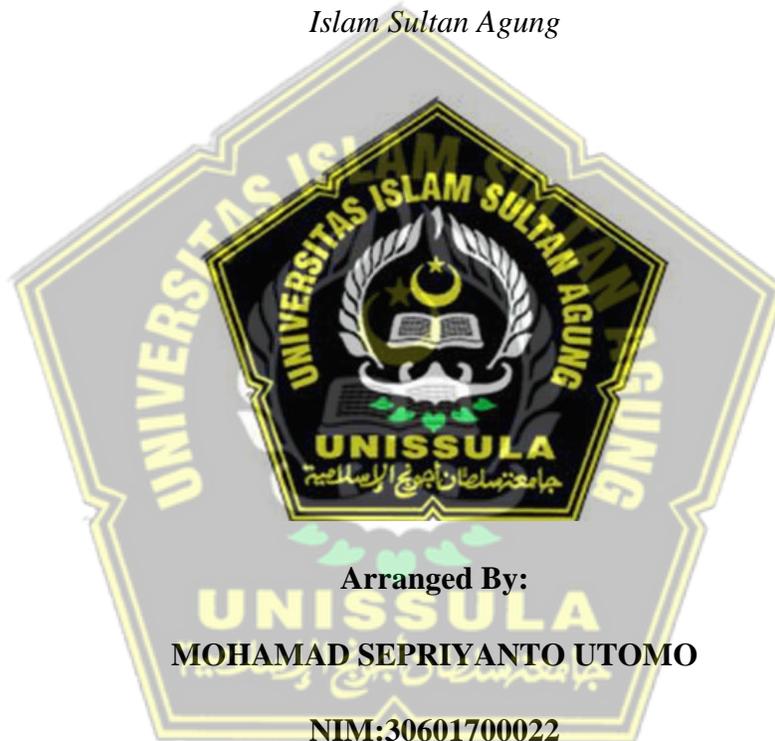
**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG
SEMARANG**

2023

**DETECTION OF THE NUMBER OF PEOPLE USING YOLOv4
METHOD IN A BUILDING (CASE STUDY: UNISSULA
ELECTRICAL LAB)**

FINAL PROJECT

*Proposed to complete the requirement to obtain a bachelor's degree (S1) at
Departement of Industrial Engineering, Faculty of Industrial Technology, Universitas
Islam Sultan Agung*



Arranged By:

MOHAMAD SEPRIYANTO UTOMO

NIM:30601700022

**MAJORING OF ELECTRICAL ENGINEERING
INDUSTRIAL TECHNOLOGY FACULTY
SULTAN AGUNG ISLAMIC UNIVERSITY
SEMARANG**

2023

LEMBAR PENGESAHAN PEMBIMBING

Laporan Tugas Akhir dengan judul “DETEKSI JUMLAH MANUSIA MENGGUNAKAN METODE YOLOv4 DALAM SUATU GEDUNG (STUDI KASUS : LAB ELEKTRO UNISSULA)” ini disusun oleh:

Nama : Mohamad Sepriyanto Utomo

NIM : 30601700022

Program Studi : Teknik Elektro

Telah disahkan dan disetujui oleh dosen pembimbing pada:

Hari : Rabu

Tanggal : 10 Maret 2023

Pembimbing I

Jenny Putri Hapsari ST, MT

NIDN. 0607018501

Pembimbing II

Ir. Budi Pramono Jati, MM, MT.

NIDN. 0623126501

Mengetahui,

Ketua Program Studi Teknik Elektro

Jenny Putri Hapsari, ST, MT

NIDN. 0607018501

LEMBAR PENGESAHAN PENGUJI

Laporan Tugas Akhir dengan judul “DETEKSI JUMLAH MANUSIA MENGGUNAKAN METODE YOLOv4 DALAM SUATU GEDUNG (STUDI KASUS : LAB ELEKTRO UNISSULA)” ini telah dipertahankan di depan Penguji sidang Tugas Akhir pada:

Hari : Rabu
Tanggal : 10 Maret 2023

Penguji I

Penguji II

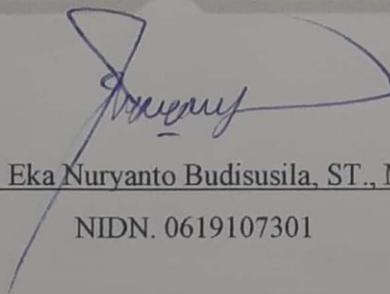

Jenny Putri Hapsari ST, MT


Ir. Budi Pramono Jati, MM, MT.

NIDN. 0607018501

NIDN. 0623126501

Ketua Penguji


Dr. Eka Nuryanto Budisusila, ST., MT

NIDN. 0619107301

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Mohamad Sepriyanto Utomo
NIM : 30601700022
Fakultas : Fakultas Teknologi Industri
Program Studi : Teknik Elektro
Judul Tugas Akhir : DETEKSI JUMLAH MANUSIA MENGGUNAKAN
METODE YOLOv4 DALAM SUATU GEDUNG (STUDI KASUS : LAB
ELEKTRO UNISSULA)

Dengan ini menyatakan bahwa judul Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Elektro tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diaacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila dikemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, Maret 2023

Yang menyatakan



Mohamad Sepriyanto Utomo

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama : Mohamad Sepriyanto Utomo

NIM : 30601700020

Program Studi : Teknik Elektro

Fakultas : Teknologi industri

Alamat Asal : jalan dalurung 01 No.82 RT.03/RW.07 kel. BANTARJATI kec. BOGOR UTARA, KOTA BOGOR

Email : Mohamad Sepriyanto Utomo

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : (DETEKSI JUMLAH MANUSIA MENGGUNAKAN METODE YOLOv4 DALAM SUATU GEDUNG (STUDI KASUS : LAB ELEKTRO UNISSULA) Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiatisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan Agung.

Semarang, Maret 2023

Yang Menyatakan



Mohamad Sepriyanto Utomo

HALAMAN PERSEMBAHAN

Alhamdulillahirabbil'alamin

Rasa syukur kepada Allah SWT yang telah melimpahkan rahmat, hidayah, cinta dan kasih sayang serta telah memberikan kekuatan dan kesabaran yang berlimpah sehingga dapat menyelesaikan tugas akhir ini dengan sebaik-baiknya, Sholawat serta salam selalu terlimpah kepada baginda Nabi besar Nabi Muhammad SAW, semoga kelak akan mendapat syafa'at beliau di yaumul qiamah nanti, amin. Laporan tugas akhir ini yang berjudul “Pengukuran Tingkat Efisiensi Aktivitas Proses Produksi Menggunakan Metode *Data Envelopment Analysis* (DEA) (Studi Kasus: CV. Mandiri Sejahtera Garment)” yang saya persembahkan kepada orang-orang yang sangat saya sayangi dan cintai terutama kedua orang tua saya Bapak dan Ibu tercinta sebagai wujud rasa terima kasih karena telah memberikan semangat, dukungan, motivasi dan mendoakan dalam menyelesaikan tugas akhir saya ini.

Telah selesainya tugas akhir saya ini merupakan capaian awal yang bisa saya persembahkan untuk memulai kehidupan baru. Saya tahu, bahwa tugas akhir ini tidak ada apa-apanya dibandingkan dengan perjuangan orang tua saya dalam mendidik, membimbing serta membiayai saya selama ini, tetapi saya akan selalu berusaha untuk membuat kedua orang tua saya selalu bangga dan bahagia dengan usaha saya semaksimal mungkin. Terima kasih atas seluruh kerja keras Bapak dan juga Ibu, untuk setiap doa yang tak henti-hentinya di ucapkan untuk kesuksesan saya, sampai saat ini saya hanya masih bisa membalasnya dengan ucapan kata terima kasih yang sebesar-besarnya. Semoga Allah SWT membalas semua kebaikan Bapak dan Ibu, Aamiin. Yang terakhir terima kasih kepada semua teman-teman atas semua kebaikan, semangat dan motivasi yang telah diberikan untuk saya dalam mengerjakan Tugas Akhir ini.

HALAMAN MOTO

*“Sesungguhnya Allah tidak akan mengubah keadaan suatu kaum sebelum mereka
mengubah keadaan diri mereka sendiri”*

(QS. Ar-Ra'd : 11)



KATA PENGANTAR

Rasa syukur penulis kepada kehadirat Allah SWT yang telah memberikan nikmat dan rahmat-Nya sehingga masih berkesempatan untuk menuntut ilmu dalam keadaan sehat wal'afiat. Shalawat serta Salam tercurahkan kepada baginda Rasulullah Muhammad SAW, semoga kelak kita mendapatkan syafaatnya. Aamiin ya Yaa Robbaalalamin.

Banyak hambatan yang terjadi dalam penulisan tugas akhir ini tetapi dengan adanya pihak lain yang membantu sehingga penulis dapat menyelesaikannya. Untuk itu dengan segala kerendahan hati, penulis mengucapkan banyak terima kasih kepada :

1. Allah SWT yang telah memberikan Rahmat dan ridho-Nya serta memberikan kelapangan hati dan pikiran selama menuntut ilmu.
2. Kedua orang tua saya, Ibu dan Bapak tercinta yang senantiasa mendoakan, memberikan dukungan, motivasi, dan kasih sayang kepada penulis.
3. Ibu Dr. Ir. Novi Marlyana, S.T., M.T., IPU., ASEAN.Eng selaku Dekan Fakultas Teknologi Industri, Universitas Islam Sultan Agung Semarang.
4. Ibu Jenny Putri Hapsari, ST, MT. selaku Ketua Program Studi Teknik Elektro Universitas Islam Sultan Agung Semarang.
5. Bapak Jenny Putri Hapsari, ST, MT. selaku Dosen Wali serta Pembimbing I dan Bapak Ir. Budi Pramonojati MM., MT. selaku Dosen Pembimbing II yang telah membimbing saya, memberi arahan, saran, masukan, motivasi, serta kesabaran dalam penyusunan Tugas Akhir ini.
6. Semua Dosen dan Karyawan Fakultas Teknologi Industri atas semua ilmu, bimbingan dan bantuannya hingga penulis menyusun tugas akhir ini.

7. Tidak lupa pula kepada semua pihak yang telah terlibat dalam penyusunan Tugas Akhir ini, yang tidak dapat penulis sebutkan satu – persatu.



ABSTRAK

Pemerintah Indonesia telah menetapkan pembatasan kapasitas ruangan sebagai upaya mencegah penyebaran Covid-19, namun masih banyak masyarakat yang mengabaikan peraturan tersebut. Algoritma YOLOv4 (You Only Look Once) merupakan algoritma deep learning yang memanfaatkan convolution neural network (CNN) dalam mendeteksi objek, termasuk manusia dalam ruangan. Penelitian ini bertujuan untuk mengembangkan aplikasi penghitung jumlah manusia pada ruangan menggunakan bahasa pemrograman Python dengan library YOLOv4 untuk meminimalisir penyebaran Covid-19. Metode penelitian tugas akhir ini adalah mengimplementasikan algoritma YOLOv4 untuk mendeteksi manusia serta dapat menghitung manusia yang masuk dan keluar pada ruangan yang diawasi oleh CCTV.

Pada proses CNN, YOLOv4 membagi gambar ke dalam kisi-kisi dengan ukuran tertentu, kemudian memprediksi kotak pembatas dan peta kelas dari setiap kisi. Algoritma YOLOv4 menggunakan bahasa Python dan library yang dapat mendeteksi objek dengan frame rate yang lebih tinggi. Aplikasi ini dapat digunakan untuk mengetahui jumlah orang yang masuk dan keluar dari ruangan, serta dapat mendeteksi manusia dari bagian depan, belakang, dan sisi samping. Pengujian alat pendeteksi manusia dengan metode YOLOv4 ini sudah sesuai target yang diinginkan dan memberikan hasil yang sesuai dengan konsep rancangan awal.

Pengujian dilakukan dengan berbagai cara, seperti pengujian deteksi manusia dengan satu orang masuk secara bergantian, pengujian deteksi manusia dengan dua orang masuk secara bersamaan, dan pengujian deteksi manusia tiga orang masuk secara bersamaan. Pendeteksia sudah bisa menghitung jumlah manusia baik yang masuk maupun keluar jika manusia tersebut melintasi garis pendeteksi. Namun, perhitungan bisa mengalami error jika resolution gambar kurang jelas atau jika terdapat dua objek yang saling menghalangi. Secara keseluruhan, YOLOv4 dapat menjadi solusi efektif untuk mencegah penyebaran Covid-19 dengan mendeteksi jumlah manusia dalam ruangan.

Kata kunci : Pembatasan kapasitas ruangan, pencegahan COVID-19, algoritma YOLOv4, pendeteksian manusia

DAFTAR ISI

LAPORAN TUGAS AKHIR	i
FINAL PROJECT	ii
LEMBAR PENGESAHAN PEMBIMBING	Error! Bookmark not defined.
LEMBAR PENGESAHAN PENGUJI.....	Error! Bookmark not defined.
SURAT PERNYATAAN KEASLIAN TUGAS AKHIR	Error! Bookmark not defined.
PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH.....	Error! Bookmark not defined.
HALAMAN PERSEMBAHAN.....	vii
HALAMAN MOTO	viii
KATA PENGANTAR	ix
ABSTRAK.....	xi
DAFTAR ISI	xii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvi
BAB I PENDAHULUAN	Error! Bookmark not defined.
1.1 Latar Belakang	Error! Bookmark not defined.
1.2 Rumusan Masalah	Error! Bookmark not defined.
1.3 Pembatasan Masalah	Error! Bookmark not defined.
1.4 Tujuan	Error! Bookmark not defined.
1.5 Manfaat	Error! Bookmark not defined.
1.6 Sistematika Penulisan	Error! Bookmark not defined.
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI	Error! Bookmark not defined.
2.1 Tinjuan Pustaka	Error! Bookmark not defined.
2.2 YOLO	Error! Bookmark not defined.
2.3 YOLOv4	Error! Bookmark not defined.
2.4 Convolution Layer	Error! Bookmark not defined.
2.5 Open CV	Error! Bookmark not defined.
2.6 Object Detection	Error! Bookmark not defined.

2.7	<i>Bahasa Python</i>	Error! Bookmark not defined.
2.8	<i>Image Processing</i>	Error! Bookmark not defined.
2.9	WEBCAM.....	Error! Bookmark not defined.
2.10	CUDA	Error! Bookmark not defined.
2.11	cuDNN.....	Error! Bookmark not defined.
2.12	Pycharm	Error! Bookmark not defined.
2.13	<i>Tracking object</i>	Error! Bookmark not defined.
2.14	Citra	Error! Bookmark not defined.
2.15	<i>Convolutional Neural Network</i>	Error! Bookmark not defined.
2.16	PIXEL	Error! Bookmark not defined.
BAB III METODE PENELITIAN		Error! Bookmark not defined.
3.1	Metode Penelitian	Error! Bookmark not defined.
3.2	Pendeteksia Manusia.....	Error! Bookmark not defined.
3.2.1	Datasheet	Error! Bookmark not defined.
3.2.2	Anotasi citra.....	Error! Bookmark not defined.
3.2.3	<i>Training</i>	Error! Bookmark not defined.
3.2.4	<i>Preprocessing</i>	Error! Bookmark not defined.
3.2.5	Hasil Pendeteksian	Error! Bookmark not defined.
3.3	Pendeteksi	Error! Bookmark not defined.
3.4	Perancangan Program	Error! Bookmark not defined.
3.5	Alat dan Bahan.....	Error! Bookmark not defined.
3.6	Perancangan Alat	Error! Bookmark not defined.
3.7	Perancangan <i>Software</i>	Error! Bookmark not defined.
3.7.1	Python 3.11.1.....	Error! Bookmark not defined.
3.7.2	Pycharm	Error! Bookmark not defined.
3.7.3	OpenCV.....	Error! Bookmark not defined.
3.7.4	Pillow	Error! Bookmark not defined.
3.7.5	CUDA toolkit 12.0	Error! Bookmark not defined.
3.7.6	cuDNN.....	Error! Bookmark not defined.
3.8	<i>Flowchart</i> Pemrograman YOLOv4.....	Error! Bookmark not defined.

3.9	Pengujian dan Pengambilan Data	Error! Bookmark not defined.
3.9.1	Pengujian Software	Error! Bookmark not defined.
3.9.2	Pengujian Keseluruhan	Error! Bookmark not defined.
BAB IV PENGUJIAN DAN ANALISA		1
4.1	Pengujian Deteksi Manusia Satu Orang Secara Bergantian	32
4.2	Pengujian Deteksi Dua Orang Masuk Secara Bersama dengan Posisi Sejajar dan Berurutan	34
4.2.1	Pengujian Deteksi Dua Orang Masuk Secara Bersama dengan Posisi Sejajar	34
4.2.2	Pengujian Deteksi Dua Orang Masuk Secara Bersama dengan Posisi Berurutan	35
4.3	Pengujian Deteksi Tiga Orang Masuk Secara Bersama dengan Posisi Sejajar dan Berurutan	37
4.3.1	Pengujian Deteksi Tiga Orang Masuk Secara Bersama dengan Posisi Sejajar	37
4.3.2	Pengujian Deteksi Tiga Orang Masuk Secara Bersama dengan Posisi Berurutan	38
4.4	Analisa	40
BAB V PENUTUP		42
5.1	Kesimpulan	42
5.2	Saran	43
DAFTAR PUSTAKA		44

DAFTAR GAMBAR

Gambar 2. 1 Arsitektur YOLOv4.....	9
Gambar 2. 2 Pemisahan R G B.....	9
Gambar 2. 3 WebCAM.....	12
Gambar 2. 4 Arsitektur <i>Covolutional Neural Network</i>	15
Gambar 3. 1 Blok Aplikasi Penghitung Manusia.....	16
Gambar 3. 2 <i>Flowchart</i> Alur Penelitian.....	17
Gambar 3. 3 Proses Pendeteksia Manusia Menggunakan YOLOv4.....	18
Gambar 3. 4 <i>Dataseet</i>	19
Gambar 3. 5 Proses Anotasi Gambar.....	19
Gambar 3. 6 Hasil Setelah Anotasi Gambar.....	20
Gambar 3. 7 Proses Training menjadi YOLOv4.....	21
Gambar 3. 8 Terdeteksi Dari Depan.....	22
Gambar 3. 9 Terdeteksi Dari Samping.....	22
Gambar 3. 10 Terdeteksi Dari Belakang.....	23
Gambar 3. 11 Tampilan Python Pada Website.....	25
Gambar 3. 12 Tampilan Gambar Pycharm.....	25
Gambar 3. 13 Install OpenCV Pada Pycharm.....	26
Gambar 3. 14 Install Pillow Pada Pycharm.....	26
Gambar 3. 15 Tampilan CUNDA.....	27
Gambar 3. 16 Tampilan cuDNN.....	28
Gambar 3. 17 <i>Flowchart</i> Pemrograman YOLOv4.....	29

DAFTAR TABEL

Tabel 4. 1 Pengujian Deteksi Satu Orang Secara Bergantian	32
Tabel 4. 2 Hasil Pengujian Deteksi Dua Orang Masuk Secara Berasma Dengan Posisi Sejajar.....	34
Tabel 4. 3 Hasil Pengujian Deteksi Dua Orang Masuk Secara Berasma Dengan Posisi Berurutan.....	35
Tabel 4. 4 Hasil Pengujian Deteksi Tiga Orang Masuk Secara Berasma Dengan Posisi Sejajar	37
Tabel 4. 5 Hasil Pengujian Deteksi Tiga Orang Masuk Secara Berasma Dengan Posisi Berurutan	39



BAB I

PENDAHULUAN

1.1 Latar Belakang

Pembatasan kapasitas ruangan di tetapkan oleh pemerintah Indonesia untuk mencegahnya penyebaran covid 19 yang terus terjadi , banyak sekali masyarakat yang mengabaikan peraturan pembatasan kapasitas ruangan seperti ruang kelas, ruangan kantor , dan ruangan bioskop. Pemerintah hanya menetapkan isi ruangan di isi sebanyak 50% - 75% untuk menghindari pencegahan covid 19 ini. Di mana sekarang memasuki era revolusi industri 4.0. di era ini, digitalisasi dan efisensi sangat di perlukan untuk perkembangan teknologi. Salah satu cara untuk mengetahui jumlah manusia yang berada didalam ruangan untuk mencegah penyebaran covid 19. dengan menggunakan WEBCAM yang di beri algoritma Yolov4 untuk memudahkan pengawasan jumlah manusia dalam ruangan. YOLOv4 menggunakan bahasa *Python* dan library yang dapat mendeteksi objek dengan *frame rate* yang lebih tinggi.

Algoritma YOLO (You Only Look Once) adalah Algoritma deep learning yang memanfaatkan *convolution neura network* (CNN) dalam mendeteksi objek. Pada proses CNN ini terdapat 3 proses yaitu yang pertama proses pre - processing , processing dan classifying. Algoritma YOLO membagi gambar ke dalam kisi kisi ukuran $s \times s$, kemudian memprediksi kotak pembatas dan peta kelas dari setiap kisi. Apabila pada satu *grid* terprediksi objek, maka pada *grid* tersebut akan di prediksi *bounding box* yang mengelilingi objek tersebut. Nilai *confidence* akan di hitung pada masing masing *bounding box* yang kemudian akan di seleksi berdasarkan nilai yang di dapat. Sistem deteksi yang di terapkan adalah melakukan deteksi menggunakan *reuse calassifier* atau *locator*. Penelitian ini bertujuan untuk mengembangkan aplikasi penghitung jumlah manusia pada ruangan menggunakan bahasa pemrograman python dengan libary YOLOv4. Aplikasi ini digunakan untuk mendeteksi jumlah manusia / orang agar dapat meminimalisir penyebaran covid 19. Aplikasi ini dapat digunakan untuk mengetahui jumlah orang yang masuk dan keluar dari ruangan.

1.2 Rumusan Masalah

Bedasarkan Latar belakang tersebut di kemukakan rumusan masalah sebagai berikut:

1. Bagaimana cara YOLOv4 mendeteksi objek tersebut sebagai manusia?
2. Bagaimana cara menghitung jumlah manusia dengan menggunakan YOLOv4?
3. Bagaimana kinerja YOLOv4 dalam mendeteksi manusia dan menghitung manusia?

1.3 Pembatasan Masalah

Agar penelitian ini menjadi fokus, pembatasan masalah perlu dilakukan. Batasan pada penelitian ini adalah :

1. Objek yang di deteksi hanyalah manusia yang masuk dalam ruangan.
2. Algoritma/ bahasa pemrograman yang dipakai hanya bahasa *python*.
3. Pengambilan citra dan pengujian objek akan dilakukan menggunakan WEBCAM.
4. Citra akan menjadi input adalah sebuah video secara real time.
5. Ruangan hanya memiliki satu pintu masuk dan keluar untuk memudahkan perhitungan.
6. Gedung yang digunakan untuk penelitian hanya gedung LAB elektro UNISSULA.

1.4 Tujuan

Tujuan dari penelitian ini adalah :

1. Mengetahui cara mendeteksi manusia menggunakan Algoritma YOLOv4
2. Mengetahui cara menghitung manusia yang masuk dalam ruangan menggunakan YOLOv4

1.5 Manfaat

Manfaat dari penelitian ini adalah:

1. Memahami cara penelitian dan pengumpulan data untuk penelitian.
2. Memahami cara kerja YOLOv4 untuk deteksi manusia dalam ruangan melalui WEBcam.
3. Mempermudah pengawasan orang yang masuk dan keluar dalam ruangan.

1.6 Sistematika Penulisan

Adapun penulisan laporan tugas akhir ini, menggunakan sistematika penulisan terbagi menjadi 5 bab. Sistematika penulisan tersebut adalah sebagai berikut :

BAB I : PENDAHULUAN

Bab ini berisikan latar belakang, perumusan masalah, pembatasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA DAN LANDASAN TEORI

Bab ini membahas rujukan - rujukan dari penelitian sebelumnya yang berkaitan dengan tugas akhir yang dibuat. Selain itu bab ini juga berisikan berbagai teori dan konsep yang mendukung dalam pembuatan tugas akhir ini.

BAB III : METODOLOGI PENELITIAN

Bab ini menguraikan dan memaparkan mengenai metode penelitian yang digunakan, tempat dan teknik pengumpulan data, rancangan penelitian dan tahapan untuk dapat merancang alat deteksi jumlah manusia menggunakan metode yolov4 dalam suatu gedung (studi kasus : LAB ELEKTRO UNISSULA).

BAB IV : PENGUJIAN DAN ANALISA

Bab ini berisikan pemaparan mengenai pengujian dan analisa hasil pada aplikasi deteksi jumlah manusia menggunakan metode yolov4 dalam suatu gedung (studi kasus : LAB ELEKTRO UNISSULA)

BAB V : PENUTUP

Bab ini berisikan kesimpulan dan saran. Dari pembahasan dan hasil analisa, maka ditarik suatu kesimpulan dari penelitian. Selain itu di berikan juga beberapa saran untuk pengembangan selanjutnya.



BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian tentang penghitungan objek melalui metode YOLO ini sudah dilakukan dalam beberapa penelitian terdahulu, di antaranya adalah:

1. Penelitian dengan judul “Penggunaan *lift* pada gedung - gedung deteksi dan menghitung manusia menggunakan YOLO - CNN” penelitian ini menggunakan metode YOLO - CNN untuk mengetahui jumlah manusia yang berada di dalam *lift* untuk membatasi jumlah manusia untuk mencegah penularan covid 19.(Pamungkas, 2021)
2. Penelitian dengan judul “Aplikasi penghitung jarak dan jumlah orang berbasis YOLO sebagai protokol kesehatan” penelitian ini menggunakan metode YOLO sebagai pendeteksi manusia dan menghitung jarak antara manusia yang sebagai protokol yang telah ditetapkan oleh pemerintah.(Indaryanto, 2021)
3. Penelitian dengan judul “Deteksi plat nomor kendaraan bergerak berbasis metode YOLO(*you only look once*)” penelitian ini menggunakan metode YOLO sebagai penghitung kendaraan dan mendeteksi plat nomor kendaraan. Pendeteksi ini dimulai dari input video, akurasi, input ROI, proses YOLO, *tracking*, memilih *confidence* tertinggi dan *cropping*. (Shianto, Gunadi and Setyati, 2019)
4. Penelitian dengan judul “Sistem deteksi pelanggaran sosial distancing di ruang terbuka menggunakan algoritma YOLO (*you once look only*)” penelitian ini menggunakan metode YOLO sebagai pendeteksi untuk manusia dan mengukur jarak antar manusia yang terjadi diruangan terbuka untuk pencegahan covid 19 dan serta dapat di akses menggunakan website untuk mempermudah penggunaan.(Rahman, 2021)

2.2 YOLO

Saat ini penelitian sedang berlomba lomba untuk mengembangkan sebuah model algoritma pengenalan dan deteksi objek dengan sangat cepat serta akurat di karenakan dampak dari semakin berkembang teknologi dan juga semakin banyak kebutuhan manusia. Model - model pengenalan dan deteksi objek seperti CNN (*Convulitonal Neural Network*), R-CNN (*Regional CNN*), *Faster R-CNN*, dan DPM (*Deformable part model*) memiliki kompleksitas yang tinggi dan biaya penggunaan yang tidak murah. Namun, model - model ini memiliki akurasi yang cukup tinggi walaupun sangat kompleks. Ternyata menghasilkan algoritma yang sangat cepat melebihi model - model yang ada saat itu. Namun, kecepatan pemrosesannya ini belum sebandi ng dengan akurasi karena saat itu YOLOv1 hanya mampu memiliki akurasi sebesar 88% pada top-5 *accuracy* dengan waktu cukup lama pelatihan sekitar kurang lebih satu minggu.(Aini *et al.*, 2021)

Pada proses pelatihan YOLOv1 fokus pada penggunaan *multi-part loss function* yang dapat merespon langsung terhadap performa dan keseluruhan model deteksi secara bersamaan. Selain itu, YOLOv1 juga mengalami kesulitan dalam mengenali objek - objek kecil yang berdekatan yang di akibatkan oleh adanya *spatial constraints* pada prediksi *bounding box* -nya. Yolov1 juga mengalami kesulitan mengenali objek - objek baru dari gambar yang memiliki konfigurasi maupun aspek rasio yang berbeda karena YOLOv1 hanya mempelajari prediksinya dari data - data yang di berikan saja.

Untuk memperbaiki batas batasan bata YOLOv1 tersebut, Redmon, dkk., akhirnya merilis YOLOv2 dan YOLO9000 pada paper keduanya pada tahun 2016. pada dasarnya YOLOv2 memfokuskan pengembangan pada perbaikan *recall* dan lokalisasi bersamaan dengan mempertahankan akurasi kalsifikasinya. YOLOv2 menggunakan kustomisasi jaringan berbasis arsitektur dari VGG-16, namun dengan akurasi yang yang lebih rendah. Selain itu, penggunaan algoritma *Open-source draknet-19* juga mampu meningkatkan kinerja dan performa dari YOLOv2. Draknet-19 memiliki prinsip kerja seperti *Network in Network* (NIN) yang memanfaatkan *global average pooling* untuk melakukan prediksi. Berdasarkan YOLOv2 ini, Redmon,

dkk., juga mengajukan model yang lebih kuat yang diberi nama YOLO9000. YOLO9000 menggunakan *Hierarchical view of objek classification* untuk dapat menyatukan dataset - dataset yang berbeda. Selain itu, YOLO9000 juga menggunakan algoritma pelatihan gabungan yang mendeteksi dan klasifikasi pada detektor, YOLO9000 menggunakan dataset label dari WordNet yang mampu mengenali 9000 kelas objek. Karena itu WordNet berbentuk grafik berarah yang susunannya cukup rumit, maka untuk menyederhanakannya dibentuklah pohon hirarki yang diberinama WordTree yang konsepnya berasal dari ImageNet. kemudian WordTree diisi dengan dataset gabungan dari COCO dan ImageNet. YOLO9000 mampu mempelajari jenis objek baru namun masih kesulitan untuk dapat mempelajari objek baru.(Aini *et al.*, 2021)

Selanjutnya, pada tahun 2018, Redmon, dkk., berhasil meluncurkan versi terbaru dari YOLO. Berdasarkan laporan teknologi yang di rilisnya, YOLOv3 memiliki ukuran yang lebih besar dari versi - versi sebelumnya. Namun, YOLOv3 ini masih memiliki performa yang lebih cepat dan akurasi yang lebih baik dibandingkan dari versi - versi yang sudah ada sebelumnya. YOLOv3 menerapkan regresi logistik pada *bounding box*-nya untuk mendeteksi keobjekan lebih baik. Selain itu penggunaan softmax digantikan oleh *Independent Logistic Classifier* karena softmax dinyatakan tidak berpengaruh langsung terhadap performa. Selain itu juga *binary cross-entropy loss* juga dapat digunakan saat training untuk memprediksi *class objek*. dibandingkan dengan draknet-19, draknet-53 digunakan pada versi ini dikarenakan memiliki ukuran tertinggi pada *floating point operation* per detik yang berarti dapat memanfaatkan GPU lebih baik dan lebih efisien dan lebih cepat. Namun, di balik itu performa yang lebih tinggi dari versi sebelumnya ini, YOLOv3 masih mengalami beberapa kesulitan untuk mengenali sebuah objek yang berukuran medium dan besar. Selain itu, YOLOv3 juga sulit untuk dapat mensejajarkan *bounding box* dengan objek pada gambarnya. Oleh karena itu, walaupun YOLOv3 memiliki kecepatan tinggi daripada versi - versi sebelumnya. Dan juga YOLOv3 disarankan untuk menjalankan pada matriks deteksi lama dengan 0,5 IOU.(Aini *et al.*, 2021)

YOLOv4 selanjutnya dikembangkan oleh Alexey Bochkovskiy, dkk., pada tahun 2020. Alexey Bochkovskiy melanjutkan pengembangan YOLO karena Redmon menghentikan penelitian dibidang *Computer Vision* (CV) karena mengetahui bahwa karya ilmiahnya dapat disalahgunakan oleh pihak - pihak tertentu. YOLOv4 lebih di menekankan pada pengoptimisasian komputasi paralel dan kecepatan operasi. YOLOv4 ditujukan untuk dapat menggunakan GPU (*Graphical Processing unit*) konvensional tunggal (Contoh NVIDIA GeForce GTX 1080 Ti dan RTX 2080 Ti). Hal yang berbeda dari versi sebelumnya yaitu YOLOv4 menggunakan teknik pengembangan baru yang dinamakan *Bag-of-Freebies* (BoF) dan *Bag-of-special* (BoS) pada tahap *training*-nya dengan tujuan dapat meningkatkan performa model serta akurasi tanpa mempengaruhi lama waktu pemrosesan saat *training*.(Aini *et al.*, 2021)

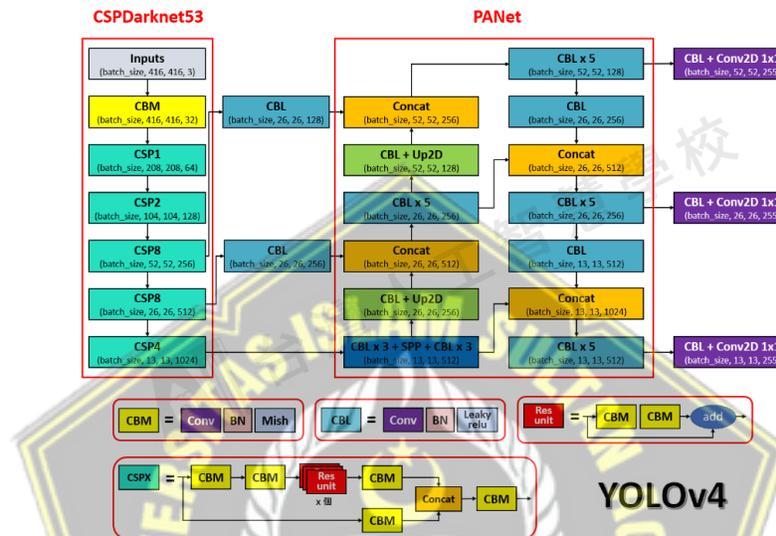
Oleh karena itu, Alexey, dkk., berhasil memodifikasi YOLOv4 sehingga tidak lagi memerlukan GPU mahal untuk dapat melakukan pemrosesan pengenalan objek lagi. Alexey, dkk., menambahkan beberapa metode yang dimodifikasi untuk melengkapi kemampuan YOLOv4, seperti PAN (*Path Aggregation Network*), CBN (*Cross mini-Batch Normalization*), SAM (*Spatial Attention Module*), dan lain sebagainya.(Aini *et al.*, 2021)

2.3 YOLOv4

YOLOv4 (*You only look once*) adalah salah satu metode deteksi objek menggunakan single convolutional neural network yang memprediksikan bounding box serta probability class secara langsung dalam sekali evaluasi. Setelah mendapatkan citra input, sistem melakukan resize terhadap citra menjadi 416 x 416 yang kemudian diproses dengan single *convolutional neural network*.(Kusuma, 2021)

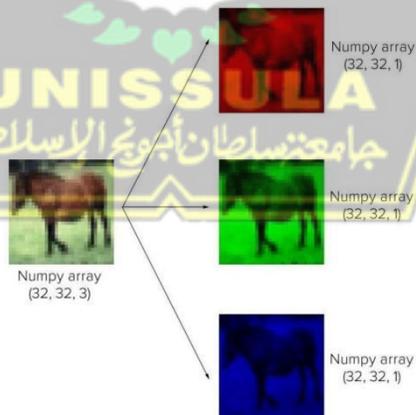
Yolov4 merupakan pengembangan dari versi sebelumnya,yaitu YOLOv3. yang membedakan YOLOv3 dari versi - versi sebelumnya adalah arsitektur darknet yang digunakannya, yaitu Darknet-53. Arsitektur ini berperan sebagai fitur pengekstraksi citra input yang terdiri atas 53 *convolutional layers* sebagai *featur extraction* dan 53

layers lain yang berperan sebagai pendeteksi objek. YOLOv4 juga masih menggunakan Draknet-53, tepatnya CSPDarknet-53 atau dapat disebut sebagai CSPNet, yaitu *backbone* baru yang dapat meningkatkan *learning capability* dari CNN. (Kusuma, 2021)



Gambar 2. 1 Arisitektur YOLOv4

2.4 Convolution Layer



Gambar 2. 2 Pemisahan R G B

Gambar di atas merupakan RGB (*Red*, *Green*, dan *Blue*) image berukuran 32 x32 pixels yang sebenarnya adalah multidimensional array dengan ukuran 32 x 32 x 3. Convolution layer terdiri dari neuron yang tersusun sedemikian rupa sehingga

membentuk sebuah filter dengan panjang dan tinggi (pixel). Sebagai contoh, layer pertama pada feature extraction layer biasanya adalah convolution layer dengan ukuran $5 \times 5 \times 3$. Panjang 5 pixel, tinggi 5 pixel, dan tebal/jumlah 3 buah sesuai dengan channel gambar tersebut. (Pujoseno, 2018)

Untuk menghitung dimensi dari feature map bisa menggunakan rumus seperti di bawah ini :

$$Output = \frac{W - N + 2P}{s} + 1 \quad (1)$$

- W = Panjang/Tinggi Input
- N = Panjang/Tinggi Filter
- P = Padding
- S = Stride

Stride adalah parameter yang menyatukan beberapa jumlah pergeseran filter. Jika nilai stride adalah 1, maka convolutional filter akan bergeser sebanyak 1 pixel secara *horizontal* lalu *vertical*. Semakin kecil stride maka akan semakin detail informasi yang akan didapatkan dari sebuah input, namun membutuhkan komputasi yang lebih jika dibandingkan dengan stride yang besar

Padding atau zero padding adalah parameter yang membutuhkan jumlah pixel (berisi nilai 0) yang akan di tambahkan di setiap sisi input. Hal ini digunakan dengan tujuan memanipulasi dimensi output dari *convolutional layer* (*feature map*). Dimensi output dari *convolutional layer* selalu lebih kecil dari input (kecuali penggunaan 1×1 filter dengan stride 1). Padding dapat mengatur dimensi output agar tetap sama seperti dimensi input atau setidaknya tidak berkurang secara drastis.

2.5 Open CV

OpenCV (*Open source computer vision library*) adalah perpustakaan perangkat lunak untuk memproses gambar waktu nyata yang dinamis, dibuat oleh intel, dan sekarang didukung oleh Willow Garage dan Itseez. OpenCV dirilis di bawah lisensi BSD yang lebih liberal daripada GPL, dan memberikan kebebasan penuh untuk

pemakai yang komersial tanpa mengungkapkan kode sumbernya. Ini juga memiliki antarmuka yang mendukung bahasa pemrograman seperti C+, C, python dan java, termasuk sistem operasi windows, Linux, MacOS, iOS, dan Android. OpenCV digunakan untuk efisiensi komputasi, dengan fokus pada aplikasi nyata.(Budi arjo, 2020)

2.6 Object Detection

Object detection atau deteksi objek merupakan bagian dari computer vision. Objek detection mengacu pada kemampuan komputer untuk mendeteksi sejumlah objek pada suatu citra. Hal ini dapat dilakukan dengan cara mengambil sudut *image feature* seperti garis, sudut, kontur, dan warna dari sebuah citra. Deteksi objek merupakan bagian dari *Object recognition* atau identifikasi objek. Oleh karena itu dapat disimpulkan bahwa deteksi objek harus diidentifikasi terlebih dahulu objek tersebut. Sedangkan penelitian ini dilakukan deteksi objek dan kemudian diimplementasikan untuk menghitung jumlah.(Faizal Indaryanto, 2021)

2.7 Bahasa Python

Python ialah bahasa pemrograman tingkat tinggi yang dikembangkan oleh Guido van Rossum pada tahun 1989 dan diperkenalkan pertama kali pada tahun 1991. Python lahir atas dasar keinginan untuk mempermudah programmer dalam menyelesaikan tugas - tugas dengan cepat. Python dirancang untuk memberikan kemudahan yang sangat luar biasa kepada programmer baik dari masalah waktu, maupun kemudahan dalam pengembangan program dalam hal kompatibilitas dengan sistem. (Qutsiah, 2016)

2.8 Image Processing

Image processing atau pengolahan citra merupakan suatu metode atau teknik yang dapat digunakan untuk memproses citra atau gambar dengan jalan memanipulasinya

menjadi suatu data gambar yang diisikan untuk mendapatkan suatu informasi tertentu mengenai obyek yang sedang diamati.(Putri, 2016)

2.9 WEBCAM

Webcam ialah sebuah kamera yang terhubung dengan computer. Webcam menangkap gambar diam dan juga gambar yang bergerak atau video. Dengan bantuan perangkat lunak, webcam dapat mengirimkan videonya keinternet secara realtime. Webcam tidak seperti camera digital dan juga camrecorder digital yang memiliki penyimpanan komputer untuk menyimpan gambar ataupun video yang telah di rekam.



Gambar 2. 3 WebCAM

2.10 CUDA

CUDA (*Compute Unified Device Architecture*) adalah sebuah arsitektur pemrosesan paralelle yang diimplementasikan ke dalam GPU buatan Nvidia. CUDA di kembangkan untuk kebutuhan umum (non-grafis). Arsitektur CUDA ini memungkinkan pengembang perangkat lunak untuk membuat program yang berjalan pada GPU buatan Nvidia dengan syntax yang mirip dengan syntax C yang sudah banyak di kenal.(Budi arjo, 2020)

Sejak awal kemunculannya sampai saat ini, CUDA mengalami perkembangan yang cukup pesat. Hal ini bisa di lihat dari perbandingan banyaknya riset berbasis CUDA yang di lakukan dan banyak CUDA SDK yang didownload.

2.11 cuDNN

Berdasarkan dari situs website nvidia, cuDNN (CUDA Deep Neural Network library) merupakan library yang dikembangkan oleh nvidia yang dikhususkan untuk deep neural network. cuDNN sangat berguna untuk meningkatkan performa GPU untuk keperluan deep neural network atau jaringan saraf tiruan. cuDNN banyak digunakan diaplikasi untuk deep learning diantaranya ada Tensorflow, Keras, Matlab, Mxnet dan PyTorch.(Budi arjo, 2020)

2.12 Pycharm

IDE lain yang cukup terkenal di kalangan developer python ialah pycharm. Pycharm ini mempunyai dua versi yaitu professional edition dan community edition. Pychar professional edition ini merupak pycharm yang berbayar dan community edition merupaka pycharm yang gratis yang tersedia untuk komunitas python dengan lisensi apache 2. Pycharm digunakan sebagai tools dalam membangun aplikasi untuk penelitian ini.

2.13 Tracking object

Tracking object pada citra bergerak merupakan sebuah cara untuk mengikut elemen citra yang sudah dilokalisasi menjadi objek yang bergerak siring waktu secara otomatis dalam citra yang bergerak siring waktu secara otomatis dalam citra bergerak. Tracking ini memiliki peran yang penting dalam penelitian ini di karenakan dalm pemrosesan video karena dapat menjadi satu patikan untuk mengekstrak beberapa perkiraan property dari objek bergerak dalam video.

2.14 Citra

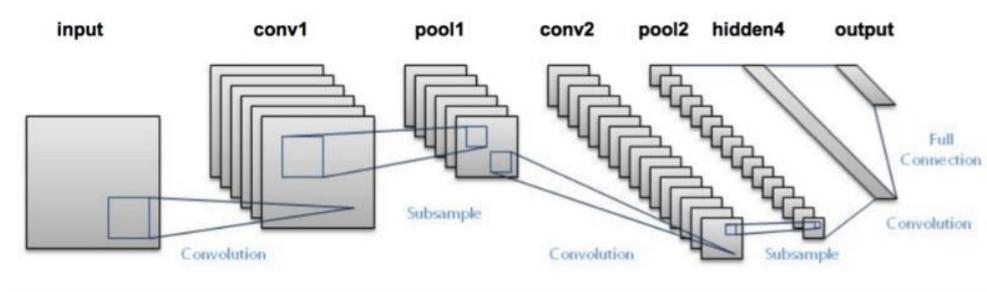
Citra merupaka gambar pada bidang dua dimensi di tinjau dari sudut pandang matematis, citra merupakan fungsi menerus dari intensitas cahaya pada bidang dwimarta. Sumber cahaya yang menerangi objek, kemudian objek memantulkan

kembali sebagaimana dari berkas cahaya tersebut. Pantulan cahaya tersebut di tangkap oleh alat alat topic, misalnya pada mata manusia kamera, dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam.

2.15 Convolutional Neural Network

CNN merupakan operasi yang menggabungkan lapisan-lapisan yang beroperasi secara paralel, hal ini terinspirasi dari sistem saraf biologis manusia. CNN mempresentasikan setiap neuronnya kedalam bentuk 2 dimensi, sehingga metode ini cocok digunakan dengan pemrosesan input berupa citra. CNN merupakan jenis jaringan saraf tiruan yang berfungsi untuk memproses data yang memiliki jaringan seperti grid. Contohnya data deret waktu yang termasuk ke dalam grid 1D (satu dimensi) yang mengambil sampel pada interval waktu yang teratur, dan data gambar yang termasuk ke dalam grid 2D pixel (dua dimensi). CNN juga menggunakan operasi matematika yang disebut konvolusi. Konvolusi adalah jenis operasi linear khusus yang menggunakan konvolusi sebagai pengganti matriks umum dan jika di lihat dari jumlah layer yang digunakan adalah berlapis-lapis, maka CNN tersebut di namakan dengan Deep Convolutional Neural Network atau biasa disingkat DCNN. (Fauzi, Eosina and Primasari, 2021)

CNN sudah banyak digunakan dalam aplikasi pengenalan gambar dan video, sistem pemberi rekomendasi, klasifikasi gambar, analisis gambar medis, dan pemrosesan bahasa alami. CNN mempelajari filter dalam algoritma gambar biasa. Algoritma di sini memiliki tujuan untuk mengelompokkan data berdasarkan data yang sudah ada. Arsitektur CNN terdiri dari *convolution layers*, *pooling layers*, dan *full connection layers*. Arsitektur *Convolutional Neural Network* terlihat pada Gambar 2.4



Gambar 2. 4 Aristektur *Covolutional Neural Network*

2.16 PIXEL

Pixel merupakan singkatan dari picture element atau elemen citra. Umumnya, orang mengetahui pixel sebagai persegi yang sangat kecil. Hal tersebut memang benar untuk layar monitor pada komputer. Tetapi pixel perangkat output digital lain, seperti printer, dapat berupa persegi ataupun lingkaran. Pada akhirnya, dapat dikatakan bahwa pixel adalah sebuah titik matematis yang tidak berdimensi.

Untuk menghitung jumlah pixel pada suatu gambar, Anda dapat mengalikan dimensi lebar dengan dimensi tinggi gambar. Misalnya, jika sebuah gambar memiliki dimensi 800x600 piksel, maka jumlah pixel pada gambar tersebut adalah $800 \times 600 = 480.000$ pixel.

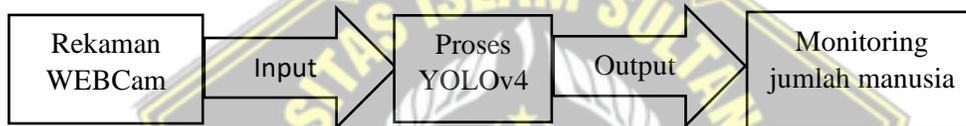
Untuk mengukur resolusi suatu gambar, Anda dapat membagi jumlah pixel dengan luas fisik gambar (dalam satuan meter persegi atau inci persegi). Semakin banyak pixel yang digunakan dalam satu unit area, semakin tinggi pula resolusi gambar yang dapat dicapai. Resolusi gambar yang tinggi akan menghasilkan gambar yang lebih tajam dan detail, namun akan membutuhkan penyimpanan yang lebih besar dan memerlukan waktu pemrosesan yang lebih lama.

BAB III

METODE PENELITIAN

3.1 Metode Penelitian

Metode penelitian tugas akhir ini adalah bagaimana mengimplementasikan algoritma YOLOv4 untuk mendeteksi manusia serta dapat menghitung manusia yang masuk dan keluar pada ruangan yang diawasi oleh webcam. Pendeteksi manusia ini melalui rekaman WEBCAM yang dipasang di area pintu masuk yang datanya diambil dan berupa sebuah citra gambar yang akan diproses oleh *pycharm*. Blok diagram aplikasi penghitung manusia seperti pada gambar 3.1.



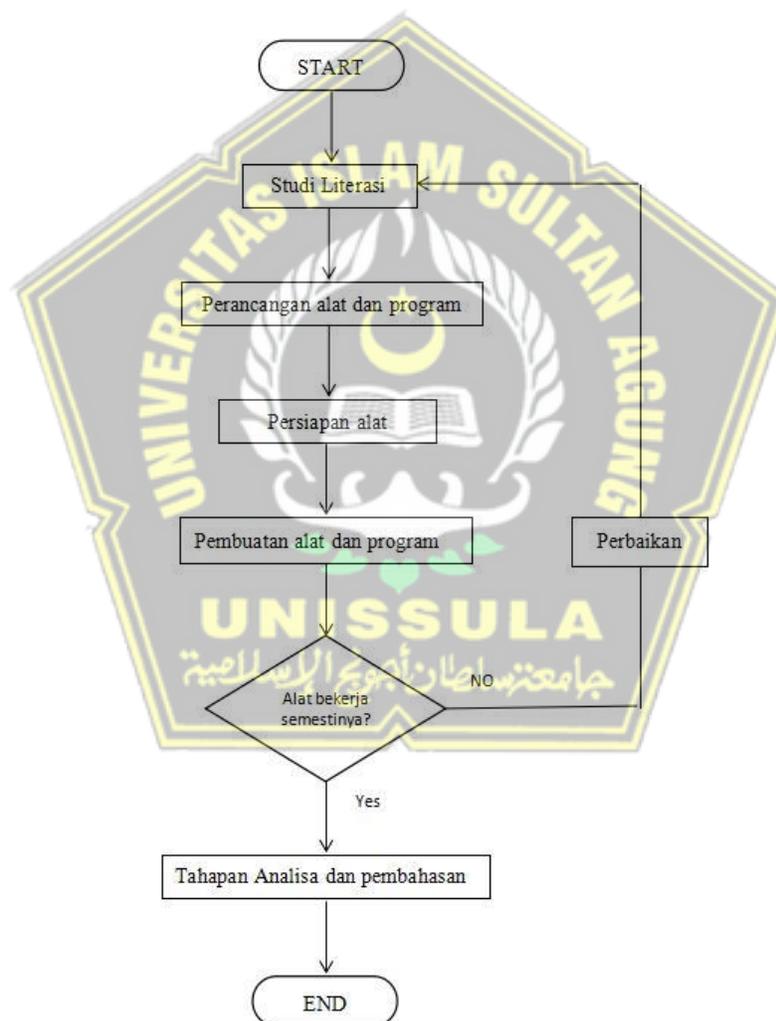
Gambar 3.1 Blok Aplikasi Penghitung Manusia

Gambar 3.1, dapat memperoleh visual yang sudah diambil dari webcam yang akan diproses oleh *image processing* pada *pycharm* menggunakan metode YOLOv4 untuk menghitung berapa banyak orang yang masuk dalam ruangan. Program YOLOv4 hanya akan menghitung manusia jika manusia itu melewati garis yang sudah dibuat oleh program dan berfungsi menghitung manusia jika melewatinya.

Pada metodologi penelitian merupakan salah satu syarat dalam mendapatkan studi – studi yang dilakukan sehingga penelitian sebagai dapat di selesaikan. Adapun metodologi penelitian perancangan ini yaitu sebagai berikut :

1. Mempelajari konsep, ilmu dan teori – teori mengenai rancangan bangun deteksi jumlah manusia menggunakan YOLOv4 dalam suatu Gedung (studi kasus : LAB Elektro UNISSULA) yang dapat mendukung tugas akhir. Sumber referensi diambil melalui jurnal ilmiah, *paper* maupun laporan penelitian
2. Tahapan ini mempersiapkan perancangan sedemikian rupa untuk mendukung penelitian tugas akhir.

3. Mempersiapkan komponen dan bahan untuk mendukung penelitian tugas akhir.
4. Tahapan pembuatan dan perakitan mulai dari membuat mekanik kerja dan pemrograman.
5. Tahapan pengujian alat untuk mengetahui supaya alat sudah bekerja sebagai mana semestinya atau tidak berdasarkan program yang telah dikonfigurasi.
6. Pengelohan data dan menganalisa hal pengujian apakah sesuai hipotesa penulis dan apa sebabnya, serta apakah teknologi dapat dikembangkan.



Gambar 3. 2 Flowchart Alur Penelitian

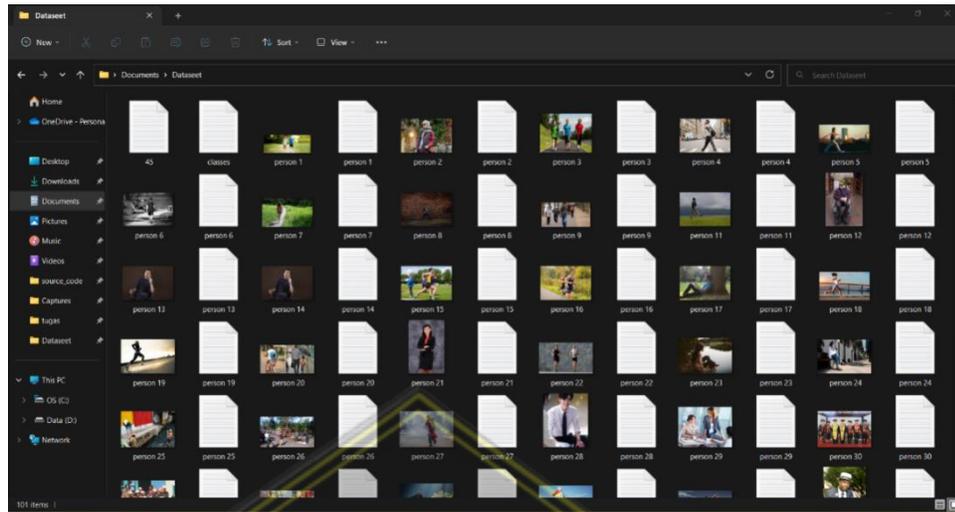
3.2 Pendeteksia Manusia



Gambar 3. 3 Proses Pendeteksian Manusia Menggunakan YOLOv4

3.2.1 Datasheet

Proses untuk memulai deteksi objek yang harus disiapkan adalah menyiapkan data citra yang berisikan objek – objek yang ingin dideteksi. Datasheet ini bisa dilakukan dengan cara pengambilan gambar manusia yang berada di google, datasheet yang di kumpulan untuk penelitian ini sebanyak 50 datasheet dan untuk format yang dipakai ialah format j.pg, pengumpulan datasheet manusia berbagai posisi yang dipakai seperti menghadap belakang, samping dan depan.



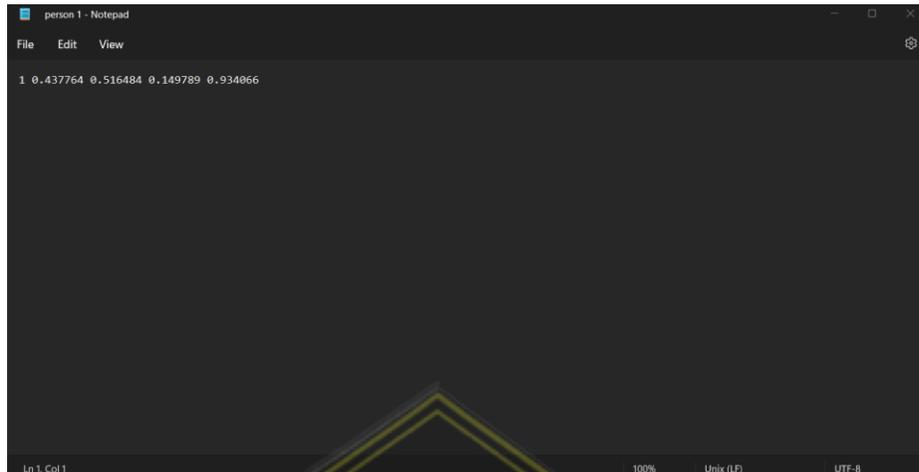
Gambar 3.4 Datasheet

3.2.2 Anotasi citra

Setelah mendapatkan datasheet berupa gambar manusia selanjutnya membuat label dengan cara memberikan kotak pembatas (*bounding box*). Hasil dari anotasi tersebut adalah sebuah data yang berisikan informasi letak kotak pembatas beserta labelnya dalam bentuk `.txt/file`. Proses anotasi citra ini memberi kotak seluruh tubuh manusia supaya bisa mendeteksi manusia.



Gambar 3.5 Proses Anotasi Gambar



Gambar 3. 6 Hasil Setelah Anotasi Gambar

3.2.3 Training

Data yang sudah di anotasi pada sebelumnya akan di latih sehingga membentuk sebuah pola yang di hasilnya berbentuk bobot. Bobot ini akan digunakan untuk mendeteksi objek di dalam citra. Training akan dilakukan menggunakan *you only look once* (YOLO) dimana ini berbasis CNN. Perangkat yang digunakan dalam proses pengambilan data training diproses oleh *google colab* dan sedangkan YOLO yang dipakai itu adalah YOLOv4. Sebelum di latih datasheet yang sudah di labeling untuk ubah foldernya menjadi format zip dan di upload di google drive, setelah mengupload datasheet di google drive tahap selanjutnya membuka googlecolaboraty untuk melakukan training menggunakan YOLOv4, setelah berhasil nanti akan berbentuk file YOLOv4.cfg dan YOLOv4.weight.



```

[ ] #download the newly released yolov4 Convlet weights
%cd /content/darknet
!wget https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v3_optimal/yolov4.conv.137

/content/darknet
--2021-10-11 17:42:34-- https://github.com/AlexeyAB/darknet/releases/download/darknet_yolo_v3_optimal/yolov4.conv.137
Resolving github.com (github.com)... 192.30.255.112
Connecting to github.com (github.com)|192.30.255.112|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://github-releases.githubusercontent.com/75388965/480fe508-889d-11ea-839e-cd4182f7f0b0?ac=Algorithm-MS4-wMC-QmJ2IjE5MkAm-Credential-ACIAW07Y
--2021-10-11 17:42:34-- https://github-releases.githubusercontent.com/75388965/480fe508-889d-11ea-839e-cd4182f7f0b0?ac=Algorithm-MS4-wMC-QmJ2IjE5MkAm-Cred
Resolving github-releases.githubusercontent.com (github-releases.githubusercontent.com)... 185.199.108.154, 185.199.109.154, 185.199.110.154, ...
Connecting to github-releases.githubusercontent.com (github-releases.githubusercontent.com)|185.199.108.154|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 170038676 (162M) [application/octet-stream]
Saving to: 'yolov4.conv.137'

yolov4.conv.137 100%[=====] 162.16M 58.09M/s In 2.8s
2021-10-11 17:42:37 (58.6 MB/s) - 'yolov4.conv.137' saved [170038676/170038676]

#we build config dynamically based on number of classes
#we build iteratively from base config files. This is the same file shape as cfg/yolo-obj.cfg
def file_len(fname):
    with open(fname) as f:
        for i, l in enumerate(f):
            pass
    return i + 1

```

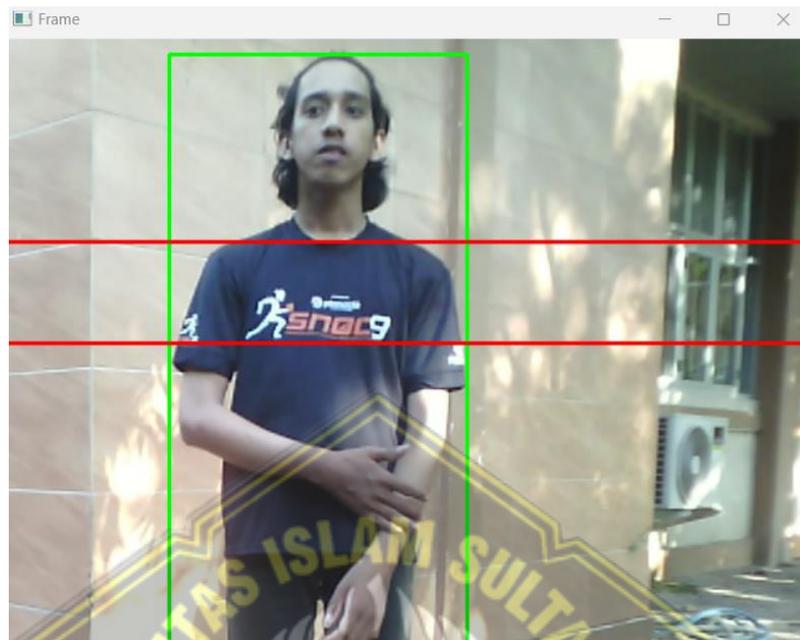
Gambar 3. 7 Proses Training menjadi YOLOv4

3.2.4 Preprocessing

Agar citra dapat di olah secara maksimal membutuhkan tahap preprocessing, tahap ini mengubah citra awal menjadi citra lain dengan cara menambahkan atau mengurangi bagian hal yang dianggap penting maupun tidak untuk dapat diproses selanjutnya. Datasheet yang telah di dapatkan sebelumnya perlu di beri label kelas pada setiap citranya, proses ini dapat disebut juga sebagai anotasi citra. Pada penelitian ini, citra akan di preprocessing resize. Resize yang dilakukan ini sebesar 608 pixel.

3.2.5 Hasil Pendeteksian

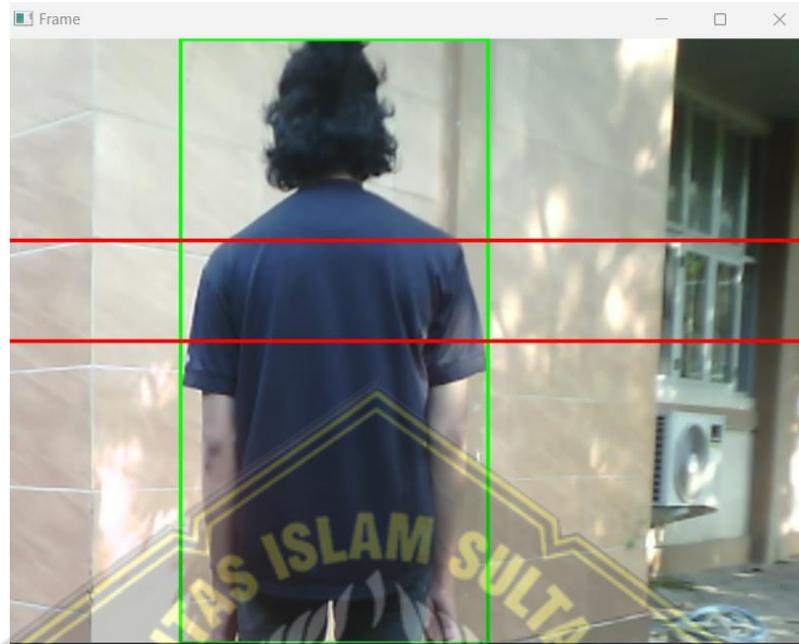
Hasil pendeteksia di lakukan secara realtime untuk mengetahui bagaimana hasil dari data yang sudah di training oleh google colab yang menggunakan metode YOLOv4. Untuk hasil pendeteksia sebagai berikut:



Gambar 3. 8 Terdeteksi Dari Depan



Gambar 3. 9 Terdeteksi Dari Samping



Gambar 3. 10 Terdeteksi Dari Belakang

3.3 Pendeteksi

Proses penghitungan manusia untuk mengetahui manusia yang masuk dan keluar dari Lab elektro UNISSULA dibuat garis pendeteksi untuk menghitung manusia jika melewati garis pendeteksi jika manusia melewati garis pertama maka counter masuk akan +1 dan jika melewati garis ke dua maka counter keluar akan +1.

3.4 Perancangan Program

Untuk penghitungan manusia di rancang pada kondisi ruangan kondisi cahaya dan berapa manusia yang melewati garis. Dalam sistem yang digunakan, sistem menerima dari video kemudian memberikan kota penghitung pada area di dekat pintu masuk untuk mendeteksi manusia bergerak dan melewatinya.

3.5 Alat dan Bahan

Penelitian ini dilakukan menggunakan alat penelitian sebagai berikut :

1. Laptop ASUS dengan spesifikasi intel core i5-9300H, CPU2.40GHz dan memory ram 8.00GB
2. WEB CAM
3. Pycharm
4. Bahasa Python

3.6 Perancangan Alat

Perancangan alat deteksi manusia menggunakan metode YOLOv4 untuk mengetahui banyaknya manusia yang masuk ruangan lab elektro UNISSULA meliputi tahapan - tahapan sebagai berikut:

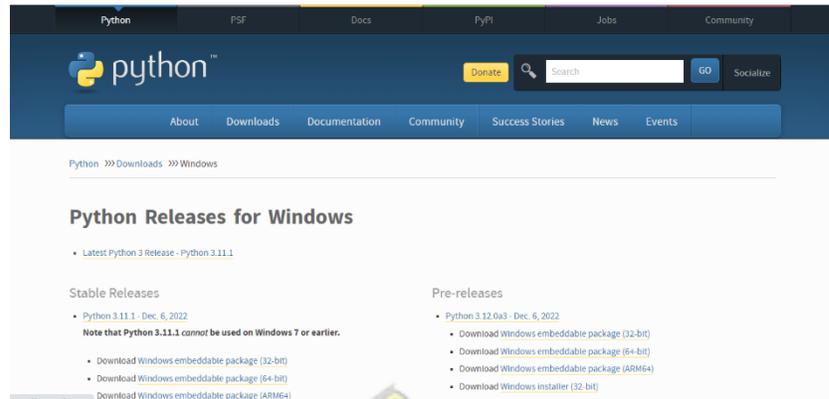
1. Penentuan spesifikasi alat.
2. Pembuatan diagram blok sistem keseluruhan.
3. Perancangan perangkat keras masing - masing blok yang meliputi perancangan dan pembuatan rangkaian dari masing - masing blok.
4. Menggabungkan beberapa blok menjadi keseluruhan sistem yang di rencanakan.

3.7 Perancangan Software

Dalam perancangan software untuk menjalankan pendeteksia manusia dan penghitungan manusia keluar masuk membutuhkan beberapa software untuk mendukung jalannya program. Berikut ini software yang harus di install untuk mendukung menjalan pendeteksian manusia sebagai berikut :

3.7.1 Python 3.11.1

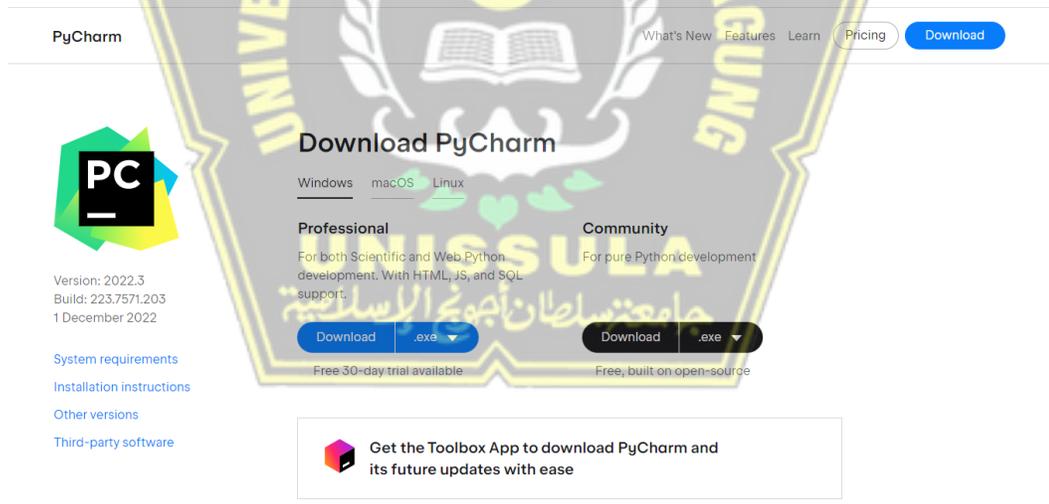
Aplikasi python untuk windows 11 dapat di akses di website <https://www.python.org/downloads/>. Penulisan menggunakan python versi 3.11.1 untuk menjalan program. Halaman download pada website python ditunjukan pada gambar 3.11.



Gambar 3. 11 Tampilan Python Pada Website

3.7.2 Pycharm

Penulisan pemrograman bahasa python menggunakan pycharm community versi 2022.3. tools ini dapat diunduh melalui website <https://www.jetbrains.com/pycharm/download/#section=windows>. Halaman download website pyscharm ditunjukkan pada gambar 3.12 :



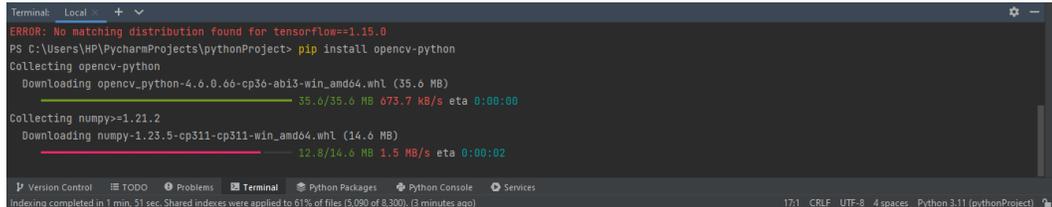
Gambar 3. 12 Tampilan Gambar Pycharm

3.7.3 OpenCV

Proses mengolah citra menggunakan openCV dan untuk menjalankan video dan menggambar garis area perhitungan. Berikut cara untuk menginstal OpenCV di tools pycharm:

1. Membuka tools pycharm yang sudah didownload.

2. Mengetik pip install opencv-python pada bagian terminal pycharm
3. Jika berhasil akan muncul tampilan seperti layar yang ditunjukkan pada oleh gambar 3.13 berikut:



```
Terminal: Local +
ERROR: No matching distribution found for tensorflow==1.15.0
PS C:\Users\HP\PycharmProjects\pythonProject> pip install opencv-python
Collecting opencv-python
  Downloading opencv_python-4.6.0.66-cp36-abi3-win_amd64.whl (35.6 MB)
    35.6/35.6 MB 673.7 kB/s eta 0:00:00
Collecting numpy==1.21.2
  Downloading numpy-1.23.5-cp311-cp311-win_amd64.whl (14.6 MB)
    12.8/14.6 MB 1.5 MB/s eta 0:00:02
Version Control | TODO | Problems | Terminal | Python Packages | Python Console | Services
Indexing completed in 1 min, 51 sec. Shared indexes were applied to 61% of files (5,090 of 8,300). (3 minutes ago)
171 CRLF UTF-8 4 spaces Python 3.11 (pythonProject)
```

Gambar 3. 13 Install OpenCV Pada Pycharm

3.7.4 Pillow

Proses resize menggunakan pillow untuk manipulasi gambar dengan cukup cepat. Berikut cara untuk menginstal pillow pada pycharm:

1. Membuka tools pycharm yang sudah didownload.
2. Mengetik pip install pillow pada bagian terminal pycharm.
3. Jika berhasil menginstal akan muncul tampilan seperti layar ditunjukkan pada gambar 3.14 berikut:



```
Terminal: Local +
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\HP\PycharmProjects\pythonProject\source_code> pip install pillow
collecting pillow
  Downloading Pillow-9.3.0-cp311-cp311-win_amd64.whl (2.5 MB)
    2.5/2.5 MB 741.2 kB/s eta 0:00:00
Installing collected packages: pillow
Successfully installed pillow-9.3.0
PS C:\Users\HP\PycharmProjects\pythonProject\source_code>
Version Control | Python Packages | TODO | Python Console | Problems | Terminal | Services
1:1 CRLF UTF-8 4 spaces <No interpreter>
```

Gambar 3. 14 Install Pillow Pada Pycharm

3.7.5 CUDA toolkit 12.0

CUDA toolkit 12.0 ini agar supaya pendeteksian objek berjalan pada GPU sehingga pendeteksia objek dapat berjalan dengan cukup cepat. CUDA toolkit 12.0 dapat di unduh [CUDA Toolkit 12.0 Downloads | nvidia Developer](#). Berikut ini cara instalasi CUDA toolkit:

1. Membuka website CUDA Toolkit 12.0 dan unduh diwebsite tersebut.
2. Memilih *Agree dan continue*.

3. Memilih *Custom* untuk instalasi komponen yang dibutuhkan kemudian klik next.
4. Centang pada bagian runtime dan development, jika drive gpu nvidia belum terinstall dan centang semua pada bagian drive component kemudian klik next seperti gambar berikut
5. pada bagian installation location biarkan saja default kemudia klik next untuk menjalankan instalasi.
6. setelah instalasi selesai akan muncul informasi komponen apa saja yang telah terinstal. Jika ada beberapa komponen yang tidak terinstal ulangi langkah instalasi dari pertama.

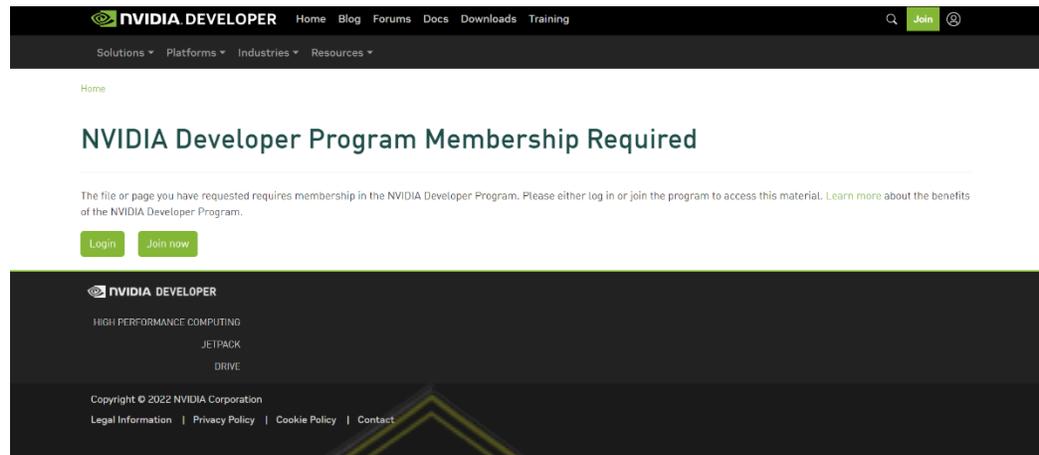


Gambar 3. 15 Tampilan CUNDA

3.7.6 cuDNN

YOLO juga memerlukan cuDNN (CUDA *Deep Neural Network Library*) untuk proses pendeteksia objek melalui GPU. cuDNN dapat diunduh diwebsite [NVIDIA Developer Program Membership Required | NVIDIA Developer](#). Berikut langkah untuk instalasi cuDNN:

1. mendownload cuDNN untuk CUDA 12.0 dari website [Developer Download Centers | NVIDIA Developer](#)
2. kemudia ekstrak berkas dan salin semua isinya ke direktori instalasi CUDA toolkit.

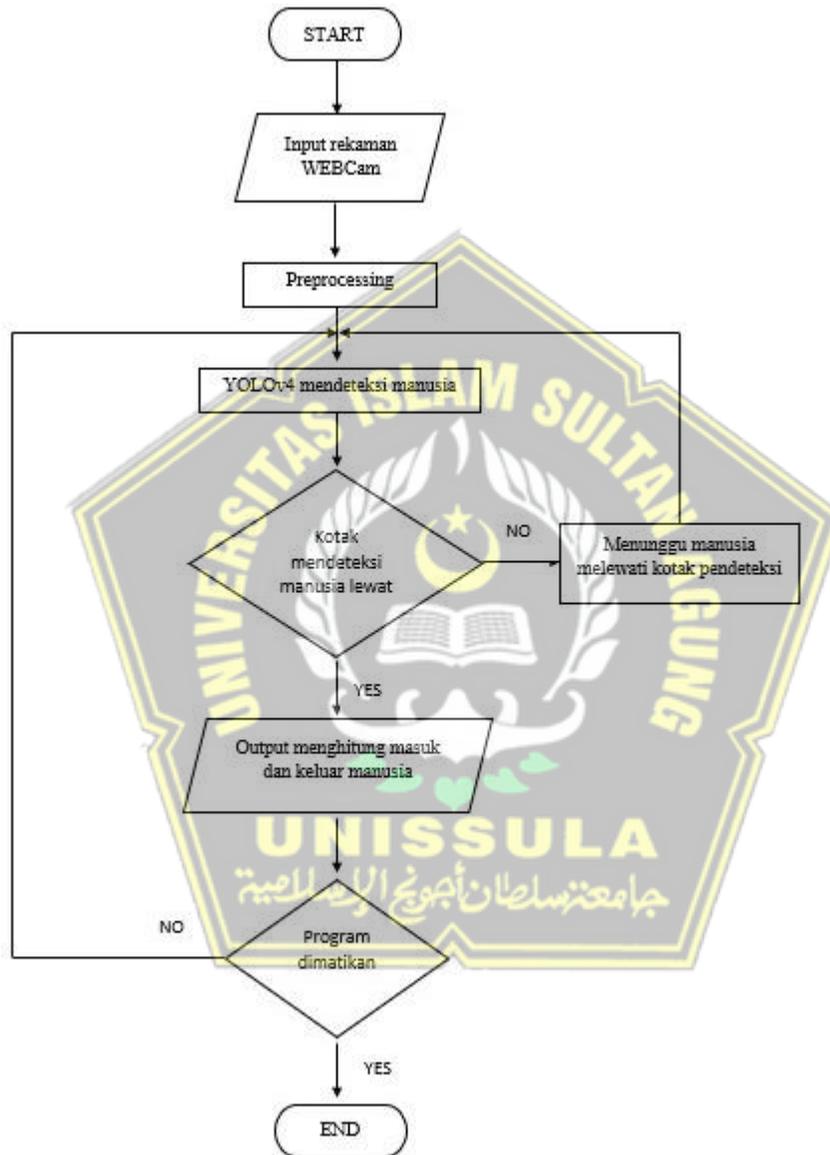


Gambar 3. 16 Tampilan cuDNN



3.8 Flowchart Pemrograman YOLOv4

Untuk mempermudah pembuatan pemrograman YOLO di perlukan berbagai tahap yaitu dengan membuat alur flowchart sistem kerja sebagai berikut.



Gambar 3. 17 Flowchart Pemrograman YOLOv4

Berikut ini adalah penjelasan dari gambar 3.17 flowchart di atas program YOLOv4 sebagai pendeteksi manusia:

1. Mulai ditunjukkan bahwa membuka atau memulai aplikasi.

2. Processing adalah proses untuk mengklasifikasi objek gambar secara manual menggunakan software.
3. YOLOv4 ialah ketika program sudah memulai berkerja mendeteksi manusia dan akan muncul garis hitung manusia yang melewati garis tersebut.
4. Kotak mendeteksi manusia ketika program sudah berhasil mendeteksi manusia yang akan melintas.
5. Menghitung jumlah manusia, jika manusia yang sudah terdeteksi melewati garis dan counter akan otomatis bertambah
6. Menunggu manusia melewati kotak pendeteksi ialah ketika garis masih menunggu manusia yang akan melewatinya.

3.9 Pengujian dan Pengambilan Data

Pengujian pendeteksia manusa dan menghitung jumlah keluar dan masuk dilakukan secara realtime yang berlokasi di lab elektro UNISSULA. Proses pendeteksia hanya manusia saja.

3.9.1 Pengujian Software

Pengujian software ini bertujuan untuk mengamati respon dari program Bahasa python yang telah dibuat. Pengujian di lakukan dengan cara menginstal pycharm untuk menjalan program untuk mendeteksi manusia dan menghitung jumlah orang masuk dan keluar pada gedung LAB elektro UNISSULA.

3.9.2 Pengujian Keseluruhan

Pengujian ini di lakukan untuk mengetahui respon dari kesalahan keseluruhan system dan apakah sudah bekerja sesuai dengan semestinya atau belum. Berikut ini beberapa yang jadi parameter yang di ujikan sebagai berikut:

1. Pengujian alat ini hanya untuk mendeteksi manusia.
2. Pengujian alat ini hanya menghitung jumlah manusia yang terdeteksi dan melewati garis.

3. Pengujian pertama di lakukan dengan satu orang sebanyak sepuluh kali percobaan.
4. Pengujian kedua di lakukan oleh dua yang masuk secara bersama dengan posisi berjajar dan berantrian.
5. Pengujian tiga di lakukan oleh tiga yang masuk secara bersama dengan posisi berjajar dan berantrian.



BAB IV

PENGUJIAN DAN ANALISA

Hasil dari pengujian dan analisa dari pengamatan pada pendeteksi manusia menggunakan metode YOLOv4 untuk mengetahui tingkat keberhasilan terhadap perancangan system yang telah di buat. Pengujian ini di lakukan meliputi, pengujian deteksi manusia dan menghitung manusia yang telah melewati garis pendeteksi.

4.1 Pengujian Deteksi Manusia Satu Orang Secara Bergantian

Pengujian deteksi manusia menggunakan metode YOLOv4 ini yang pertama ialah dengan cara mendeteksi satu orang masuk secara bergantian, di mana pengujian ini di lakukan sebanyak sepuluh kali percobaan untuk mengetahui hasil dari program yang telah di buat. Pada percobaan pertama ini di dapat data hasil percobaan Tabel 4.1

Tabel 4. 1 Pengujian Deteksi Satu Orang Secara Bergantian

Pengujian Ke	Gambar	Terdeteksi	Program	Real
1		Berhasil terdeteksi	Terhitung 1 orang masuk	1 orang masuk
2		Berhasil terdeteksi	Terhitung 1 orang masuk	1 orang masuk
3		Berhasil terdeteksi	Terhitung 1 orang masuk	1 orang masuk
4		Berhasil terdeteksi	Terhitung 1 orang masuk	1 orang masuk

Pengujian Ke	Gambar	Terdeteksi	Program	Real
5		Berhasil terdeteksi	Terhitung 1 orang keluar	1 orang keluar
6		Berhasil terdeteksi	Terhitung 1 orang keluar	1 orang keluar
7		Berhasil terdeteksi	Terhitung 1 orang masuk	1 orang keluar
8		Berhasil terdeteksi	Terhitung 1 orang keluar	1 orang keluar
9		Berhasil terdeteksi	Terhitung 1 orang masuk	1 orang masuk
10		Berhasil terdeteksi	Terhitung 1 orang masuk	1 orang masuk

Dari hasil pengujian deteksi satu orang masuk secara bergantian dapat di tarik *respawn* yang di berikan untuk mendeteksi manusia cukup cepat ketika manusia sudah terekam oleh WEBcam YOLOv4 itu langsung mendeteksi manusia dan untuk perhitungan jumlah manusia yang melintasi pendeteksi berhasil menghitung manusia yang melintas pada garis pendeteksi. Tingkat keberhasilan pada pengujian satu orang masuk secara bergantian sebesar 100%

4.2 Pengujian Deteksi Dua Orang Masuk Secara Bersama dengan Posisi Sejajar dan Berurutan

4.2.1 Pengujian Deteksi Dua Orang Masuk Secara Bersama dengan Posisi Sejajar

Pengujian deteksi dengan dua orang masuk secara bersamaan dengan posisi sejajar di mana kondisi ini orang yang akan masuk keruangan LAB elektro secara bersama dengan posisi bersejajar dan masuk secara bersama. Berikut hasil pengujian data pada tabel 4.2.

Tabel 4. 2 Hasil Pengujian Deteksi Dua Orang Masuk Secara Berasma Dengan Posisi Sejajar

Pengujian Ke	Gambar	Terdeteksi	Program	Real
1		Berhasil terdeteksi 2 orang	Terhitung 2 orang masuk	2 orang masuk
2		Berhasil terdeteksi 2 orang	Terhitung 2 orang masuk	2 orang masuk
3		Berhasil terdeteksi 2 orang	Terhitung 2 orang keluar	2 orang keluar
4		Berhasil terdeteksi 2 orang	Terhitung 2 orang keluar	2 orang keluar
5		Berhasil terdeteksi 2 orang	Terhitung 2 orang masuk	2 orang masuk
6		Berhasil terdeteksi 2 orang	Terhitung 2 orang masuk	2 orang keluar

Pengujian Ke	Gambar	Terdeteksi	Program	Real
7		Berhasil terdeteksi 2 orang	Terhitung 1 orang keluar	2 orang keluar
8		Berhasil terdeteksi 2 orang	Terhitung 1 orang keluar	2 orang keluar
9		Berhasil terdeteksi 2 orang	Terhitung 2 orang masuk	2 orang masuk
10		Berhasil terdeteksi 2 orang	Terhitung 2 orang masuk	2 orang masuk

Dari hasil pengujian deteksi dua orang masuk secara bersama dengan posisi sejajar dapat di tarik kesimpulan *respawn* yang di berikan untuk mendeteksi manusia cukup cepat ketika manusia sudah terekam oleh WEBCam itu langsung mendeteksi manusia dan untuk perhitungan jumlah manusia yang melintasi pendeteksi berhasil menghitung manusia yang melintas pada garis pendeteksi. Tingkat keberhasilan pengujian dua orang masuk secara bersama dengan posisi sejajar sebesar 80%.

4.2.2 Pengujian Deteksi Dua Orang Masuk Secara Bersama dengan Posisi Berurutan

Pengujian deteksi dengan dua orang masuk secara bersamaan dengan posisi berurutan di mana kondisi ini orang yang akan masuk keruang LAB elektro secara bersama dengan posisi dua orang berurutan dan masuk secara bersama dengan jarak cukup berdekatan. Berikut hasil pengujian data pada tabel 4.3

Tabel 4. 3 Hasil Pengujian Deteksi Dua Orang Masuk Secara Berasma Dengan Posisi Berurutan

Pengujian Ke	Gambar	Terdeteksi	Terhitung	Real
1		Hanya terdeteksi 1 orang	Terhitung 1 orang masuk	2 orang masuk

Pengujian Ke	Gambar	Terdeteksi	Terhitung	Real
2		Hanya terdeteksi 1 orang	Terhitung 1 orang masuk	2 orang masuk
3		Hanya terdeteksi 1 orang	Tidak terhitung	2 orang keluar
4		Hanya terdeteksi 1 orang	Tidak terhitung	2 orang keluar
5		Hanya terdeteksi 1 orang	Terhitung 1 orang masuk	2 orang masuk
6		Hanya terdeteksi 1 orang	Terhitung 1 orang masuk	2 orang masuk
7		Hanya terdeteksi 1 orang	Tidak terhitung	2 orang keluar
8		Hanya terdeteksi 1 orang	Terhitung 1 orang keluar	2 orang keluar
9		Hanya terdeteksi 1 orang	Tidak terhitung	2 orang masuk
10		Hanya Terdeteksi 1 orang	Tidak terhitung	2 orang masuk

Dari hasil pengujian deteksi dua orang masuk secara bersama dengan posisi berurutan dapat di tarik kesimpulan pada saat pengujian hanya mendeteksi satu orang

yang terdeteksi di karena orang pertama menghalangi orang kedua jadi YOLOv4 hanya mendeteksi satu orang saja dan menghitung hanya satu orang di karena yang terdeteksi oleh YOLOv4 hanya satu orang saja. Tingkat keberhasilan pengujian dua orang masuk secara Bersama dengan posisi berurutan sebesar 0%.

4.3 Pengujian Deteksi Tiga Orang Masuk Secara Bersama dengan Posisi Sejajar dan Berurutan

4.3.1 Pengujian Deteksi Tiga Orang Masuk Secara Bersama dengan Posisi Sejajar

Pengujian deteksi dengan tiga orang masuk secara bersamaan dengan posisi sejajar di mana kondisi ini orang yang akan masuk keruang LAB elektro secara bersama dengan posisi tiga orang sejajar dan masuk secara bersama dengan jarak cukup berdekatan. Berikut hasil pengujian data pada tabel 4.4

Tabel 4. 4 Hasil Pengujian Deteksi Tiga Orang Masuk Secara Berasma Dengan Posisi Sejajar

PengujianKe	Gambar	Terdeteksi	Terhitung	Real
1		Berhasil terdeteksi 3 orang	Terhitung 3 orang masuk	3 orang masuk
2		Berhasil terdeteksi 3 orang	Terhitung 3 orang masuk	3 orang masuk
3		Berhasil terdeteksi 2 orang	Terhitung 2 orang keluar	3 orang keluar
4		Berhasil terdeteksi 3 orang	Terhitung 3 orang keluar dan 1 orang masuk	3 orang keluar

PengujianKe	Gambar	Terdeteksi	Terhitung	Real
5		Berhasil terdeteksi orang 3	Terhitung 3 orang masuk	3 orang masuk
6		Berhasil terdeteksi orang 3	Terhitung 3 orang masuk	3 orang masuk
7		Berhasil terdeteksi orang 3	Terhitung 2 orang keluar	3 orang keluar
8		Berhasil terdeteksi orang 3	Terhitung 3 orang keluar	3 orang keluar
9		Berhasil terdeteksi orang 3	Terhitung 3 orang masuk	3 orang masuk
10		Berhasil terdeteksi orang 3	Terhitung 3 orang masuk	3 orang masuk

Dari hasil pengujian deteksi tiga orang masuk secara bersama dengan posisi sejajar dapat di tarik kesimpulan *respawn* yang di berikan untuk mendeteksi manusia cukup cepat ketika manusia sudah terekam oleh WEBcam itu langsung mendeteksi manusia dan untuk perhitungan jumlah manusia yang melintasi pendeteksi berhasil menghitung manusia yang melintas pada garis pendeteksi. Tingkat keberhasilan pada pengujian tiga orang masuk secara Bersama dengan posisi sejajar sebesar 70%.

4.3.2 Pengujian Deteksi Tiga Orang Masuk Secara Bersama dengan Posisi Berurutan

Pengujian deteksi dengan tiga orang masuk secara bersamaan dengan posisi berurutan di mana kondisi ini orang yang akan masuk keruang LAB elektro secara

bersama dengan posisi tiga orang sejajar dan masuk secara bersama dengan jarak cukup berdekatan. Berikut hasil pengujian data pada tabel 4.5

Tabel 4. 5 Hasil Pengujian Deteksi Tiga Orang Masuk Secara Berasma Dengan Posisi Berurutan

PengujianKe	Gambar	Terdeteksi	Terhitung	Real
1		Berhasil terdeteksi 1 orang	Terhitung 1 keluar	3 orang keluar
2		Berhasil terdeteksi 1 orang	Tidak ada yang terhitung	3 orang masuk
3		Berhasil terdeteksi 1 orang	Terhitung 1 orang masu	3 orang masuk
4		Berhasil terdeteksi 1 orang	Terhitung 1 orang keluar	3 orang masuk
5		Berhasil terdeteksi 1 orang	Tidak terhitung	3 orang masuk
6		Berhasil terdeteksi 1 orang	Tidak terhitung	3 orang keluar
7		Berhasil terdeteksi 1 orang	Terhitung 1 orang masuk	3 orang masuk

PengujianKe	Gambar	Terdeteksi	Terhitung	Real
8		Berhasil terdeteksi 1 orang	Terhitung 1 orang masuk	3 orang masuk
9		Berhasil terdeteksi 1 orang	Terhitung 1 orang masuk	3 orang masuk
10		Berhasil terdeteksi 1 orang	Tidak terhitung	3 orang masuk

Dari hasil pengujian deteksi tiga orang masuk secara bersama dengan posisi berurutan dapat ditarik kesimpulan pada saat pengujian hanya mendeteksi satu orang yang terdeteksi di karena orang pertama menghalangi orang kedua jadi YOLOv4 hanya mendeteksi satu orang saja dan menghitung hanya satu orang di karena yang terdeteksi oleh YOLOv4 hanya satu orang saja. Tingkat keberhasilan pada pengujian deteksi tiga orang masuk secara bersama dengan posisi berurutan sebesar 0%.

4.4 Analisa

Pengujian deteksi menggunakan metode YOLOv4 untuk mendeteksi manusia sudah berjalan dengan baik dan cepat jika manusia sudah terrekam oleh Webcam. YOLOv4 dapat mendeteksi manusia dari bagian depan manusia, belakang manusia dan sisi samping manusia, YOLOv4 akan terjadi error jika dua objek saling menghalangi sehingga akan mendeteksi satu objek saja.

Pengujian alat pendeteksi manusia dengan metode YOLOv4 ini sudah sesuai target yang di inginkan. Pendeteksia sudah bisa menghitung jumlah manusia baik yang masuk maupun keluar jika manusia tersebut melintasi garis pendeteksi. Perhitungan bisa error terjadi di karenakan resolution yang kurang jelas sehingga marking modelnya kurang sempurna, dan tidak sama dataset yang di *training* sebelumnya.

Metode YOLOv4 secara keseluruhan sudah memberikan pendeteksian hasil yang sesuai dengan konsep rancangan awal. Dari seluruh kegiatan pengujian cobaan alat menggunakan metode ini sudah berjalan dengan baik dan lancar meskipun pada kenyataan beberapa kali terjadi nya error.

Beberapa cara yang di lakukan dalam uji coba yaitu, pengujian deteksi manusia dengan satu orang masuk secara bergantian, pengujian deteksi manusia dengan dua orang masuk secara bersamaan dan posisi berurutan, dan pengujian deteksi manusia tiga orang masuk secara bersamaan dan posisi berurutan.

Pada pengujian deteksi satu orang masuk secara bergantian, data yang di hasilkan hampur seluruhnya terdeteksi sesuai dengan jumlah orang yang masuk dan keluar. Sistem sudah mendeteksi manusia benar sesuai dengan yang melintas pada garis pendeteksi.

Pengujian dua orang masuk bersamaan dan posisi berurutan, sistem terkadang mendeteksi kurang atau lebih, di karenakan posisi manusia yang akan masuk atau keluar tidak secara jelas terbaca di sistem sehingga beberapa kali terjadi nya error dan kesalahan perhitungan. Namun di beberapa perhitungan juga jelas mendeteksi sesuai dengan manusia yang melintas pada garis pendetesi. Begitu juga pada pengujian tiga orang masuk atau keluar secara bersama dan berurutan.

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil penelitian yang telah di laksanakan dapat di tarik beberapa kesimpulan sebagai berikut

1. YOLOv4 mendeteksi objek sebagai manusia dengan menggunakan konsep *deep learning dan convolutional neural networks* (CNN). Model ini di latih dengan menggunakan dataset gambar yang mengandung manusia sebagai objek dan membuat prediksi pada setiap sisi gambar untuk memidentifikasi keberadaan manusia dan melokalisasi objek. YOLOv4 memiliki arsitektur yang memungkinkan untuk mengenali objek dengan akurasi yang cukup tinggi dan menghitung objek dengan cepat.
2. YOLOv4 Menghitung jumlah manusia dengan mengestrak fitur objek manusia dari gambar dan memperkirakan titik tengah objek manusia. Kemudian, mengikuti objek manusia melalui frame berikutnya dan memperbaharui lokasi objek manusia.
3. YOLOv4 memiliki tingkat keberhasilan yang tinggi dalam mendeteksi manusia dan menghitung jumlah manusia yang melintas pendeteksi pada beberapa kasus, seperti satu orang masuk bergantian dan dua orang masuk sejajar dengan akurasi 100% dan 80% masing – masing. Namun, pada kasus dua atau tiga orang masuk secara berurutan, YOLOv4 memiliki tingkat keberhasilan yang rendah atau bahkan nol. Kinerja YOLOv4 dalam mendeteksi manusia cukup cepat, namun masih perlu ditingkatkan untuk kasus – kasus yang lebih kompleks.

5.2 Saran

Saran yang di tarik dari hasil penelitian untuk pengembangan selanjutnya adalah sebagai berikut :

1. YOLOv4 ini bisa mendeteksi objek lebih dari 1 objek dan Metode YOLO ini masih terus berkembang. Di sarankan menggunakan kamera atau WEBCam yang mempunyai kualitas yang baik.
2. Dalam pengembangan alat pendeteksi manusia, perlu diperhatikan bahwa metode YOLOv4 masih memiliki keterbatasan dalam mendeteksi objek ketika dua objek yang saling menghalangi. Oleh karena itu, perlu dilakukan peningkatan pada metode ini agar dapat lebih akurat dalam mendeteksi objek, terutama dalam situasi kompleks.
3. Menambahkan kamera berkualitas dan memperhatikan penempatan yang strategis untuk meningkatkan akurasi deteksi dan perhitungan jumlah manusia yang melewati garis pendeteksi. Hal ini akan memperbesar peluang deteksi objek manusia, menghasilkan informasi visual yang lebih banyak dan berbeda sudut pandangnya. Dengan begitu, dapat mengurangi kemungkinan terjadinya error dan kesalahan perhitungan serta meningkatkan kinerja sistem secara keseluruhan.
4. Untuk memperoleh hasil yang optimal dalam penggunaan metode YOLOv4, perlu memperhatikan faktor – faktor seperti ukuran dan kualitas datasheet yang digunakan serta spesifikasi hardware yang digunakan. Oleh karena itu, perlu dilakukan penelitian lebih lanjut dalam mengoptimalkan penggunaan metode ini agar dapat memberikan hasil yang lebih baik.
5. Menambahkan variable didalam dan logika jika manusia yang didalam sudah melebihi kapasitas ruangan yang telah ditetapkan muncul peringatan dilayar monitoring atau buzzer yang berbunyi.

DAFTAR PUSTAKA

- Aini, Q. *et al.* (2021) 'Deteksi dan Pengenalan Objek Dengan Model Machine Learning: Model Yolo', *CESS (Journal of Computer Engineering, System and Science)*, 6(2), p. 192. doi:10.24114/cess.v6i2.25840.
- Budi arjo, D.D. (2020) *IMPLEMENTASI SISTEM CERDAS PADA OTOMATISASI PENDETEKSIAN JENIS KENDARAAN DI JALAN RAYA, Laporan Tugas Akhir Jurusan Sistem Informasi USM*. Available at: <https://doi.org/10.1016/j.jnc.2020.125798><https://doi.org/10.1016/j.smr.2020.02.002><http://www.ncbi.nlm.nih.gov/pubmed/810049><http://doi.wiley.com/10.1002/anie.197505391><http://www.sciencedirect.com/science/article/pii/B9780857090409500205>
- Faizal Indaryanto, A.N. (2021) *Aplikasi Penghitung Jarak dan Jumlah Orang Berbasis YOLO Sebagai Protokol Kesehatan Covid-19*.
- Fauzi, M.R., Eosina, P. and Primasari, D. (2021) 'Deteksi Coronavirus Di sease Pada X-Ray Dan CT-Scan Paru Menggunakan Convolutional Neural Network', *JUSS (Jurnal Sains dan Sistem Informasi)*, 3(2), pp. 17–27. doi:10.22437/juss.v3i2.10888.
- Indaryanto, F. (2021) 'Aplikasi Penghitung Jarak dan Jumlah Orang Berbasis YOLO Sebagai Protokol Kesehatan Covid-19'.
- Kusuma, T.A.A.H. (2021) 'PEOPLE COUNTING FOR PUBLIC TRANSPORTATIONS USING YOU ONLY LOOK ONCE METHOD'.
- Pamungkas, B.P.G. (2021) 'DETEKSI DAN MENGHITUNG MANUSIA MENGGUNAKAN YOLO-CNN'.
- Putri, A.R. (2016) 'Pengolahan Citra Dengan Menggunakan Web Cam Pada Kendaraan Bergerak Di Jalan Raya', *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 1(01), pp. 1–6. doi:10.29100/jupi.v1i01.18.
- Qutsiah, S.A. (2016) 'APLIKASI PEMBELAJARAN MATEMATIKA DASAR BANGUN DATAR MENGGUNAKAN PYTHON PADA PERANGKAT

BERGERAK’.

Rahman, D. (2021) ‘SISTEM DETEKSI PELANGGARAN SOCIAL DI STANCING DI RUANG TERBUKA MENGGUNAKAN ALGORITMA YOU ONLY LOOK ONCE (YOLO)’.

Shianto, K.A., Gunadi , K. and Setyati, E. (2019) *Deteksi Jenis Mobil Menggunakan Metode YOLO Dan Faster R-CNN, Jurnal Infra.*

