

**DETEKSI KELENGKAPAN ALAT PELINDUNG DIRI UNTUK PEKERJA PROYEK
MENGUNAKAN METODE YOU ONLY LOOK ONCE VERSI 5 (YOLOv5)**

SKRIPSI

FAISAL LAZUARDY MUHAMMAD

191402075



**PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2023**

DETEKSI KELENGKAPAN ALAT PELINDUNG DIRI UNTUK PEKERJA PROYEK
MENGUNAKAN METODE YOU ONLY LOOK ONCE VERSI 5 (YOLOv5)

SKRIPSI

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

FAISAL LAZUARDY MUHAMMAD

191402075



PROGRAM STUDI S1 TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2023

PERSETUJUAN

Judul : DETEKSI KELENGKAPAN ALAT PELINDUNG DIRI
UNTUK PEKERJA PROYEK MENGGUNAKAN
METODE YOU ONLY LOOK ONCE VERSI 5 (YOLOv5)

Kategori : SKRIPSI

Nama : FAISAL LAZUARDY MUHAMMAD

Nomor Induk Mahasiswa : 191402075

Program Studi : SARJANA (S1) TEKNOLOGI INFORMASI

Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI UNIVERSITAS SUMATERA UTARA

Medan, 10 Januari 2024


Komisi Pembimbing :

Pembimbing II



Seniman S.Kom., M.Kom.
NIP. 198705252014041001

Pembimbing I




Ainul Hizriadi, S.Kom., M.Sc.
NIP. 198510272017061001

Diketahui/disetujui oleh

Program Studi S1 Teknologi Informasi

Ketua,




Dedy Arisandi S.T., M.Kom.
NIP. 197908312009121002

PERNYATAAN

**DETEKSI KELENGKAPAN ALAT PELINDUNG DIRI UNTUK PEKERJA PROYEK
MENGUNAKAN METODE YOU ONLY LOOK ONCE VERSI 5 (YOLOv5)**

SKRIPSI

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

Medan, 10 Januari 2024

Faisal Lazuardy Muhammad

191402075

UCAPAN TERIMA KASIH

1. Puji syukur kepada Tuhan Yang Maha Esa karena atas berkat dan karunia-Nya penulis dapat mengerjakan dan menyelesaikan skripsi yang berjudul ini sebagai syarat dalam memperoleh gelar Sarjana Komputer di Program Studi S1 Teknologi Informasi Universitas Sumatera Utara. Selama pengerjaan skripsi ini juga penulis mendapat berbagai bantuan, bimbingan dan dorongan semangat dari berbagai pihak secara langsung maupun tidak langsung. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan terima kasih kepada:
2. Keluarga penulis, terutama kedua orangtua penulis ibu Kasmilawati Br. Keliat dan Bapak Muhammad Yunus yang telah membesarkan saya serta memberikan kasih sayang, doa, masukan dan bimbingan pada dimulai dari pendidikan hingga menyelesaikan tugas akhir ini, dan juga buat kakak kakak penulis yaitu Laras Dwi Lestari dan Sasya Chairunnisa yang juga memberikan doa dan dukungan.
3. Bapak Dr. Muryanto Amin, S.Sos, M.Si. yang merupakan Rektor Universitas Sumatera Utara.
4. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
5. Bapak Dedy Arisandi S.T., M.Kom. selaku Ketua Program Studi S1 Teknologi Informasi Universitas Sumatera Utara.
6. Bapak Ainul Hizriadi, S.Kom., M.Sc. selaku Dosen Pembimbing 1 dan Bapak Seniman, S.Kom., M.Kom. selaku Dosen Pembimbing 2 yang telah memberikan ilmu, saran dan kritik yang membangun, motivasi, dan waktunya selama proses pengerjaan skripsi ini.
7. Seluruh Dosen Program Studi S1 Teknologi Informasi yang telah memberikan ilmu yang bermanfaat bagi penulis selama masa perkuliahan
8. Pegawai dan staff Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara yang mendukung segala urusan administrasi dan akademik selama perkuliahan.
9. Teman-teman di Kaum Biru yaitu Rahmat Triolonggo, Tito Surya Eka Laya, Aditya Rizqi, Iqbal Faturrahman, Habib Akbar, Faruja Rizky, Tengku Muhammad Ikilil, Fazri Imran, Heru Ghibran, dan Zaky Arief yang sudah banyak membantu, memberikan semangat tiada henti dan mendengarkan keluh kesah selama masa perkuliahan serta melewati susah senangnya bersama.

10. Teman teman PORE, Nanda Ambiya, Denaya Pramudya, Fadhil Zuhairsyah, Irsyad Fauzi, Huzaifah Lais, Faris Al Falaq, dan Brilian Jonathan yang telah banyak membantu penulis dalam pengerjaan penelitian hingga penulisan skripsi serta teman-teman seperjuangan di lab computer vision yang telah berjuang dan berprogress bersama untuk menyelesaikan tugas akhir ini.
11. Firza Syahrani Harahap yang selalu setia menemani, menyemangati, memberikan bantuan tiada henti, menjadi tempat penulis untuk berbagi cerita dan berdiskusi.
12. Bang Rhama Permadi Ahmad yang sudah banyak membantu saya dalam urusan administrasi dan membimbing saya selama masa perkuliahan saya.
13. kepada teman-teman Kom C, teman-teman satu angkatan dan senior yang telah membantu penulis selama masa perkuliahan.

Penulis menyadari bahwa Skripsi ini masih belum sepenuhnya sempurna dan masih banyak kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang dapat membantu menyempurnakan untuk perbaikan di masa mendatang. Penulis juga berharap Skripsi ini dapat bermanfaat bagi semua pihak yang membacanya.

Medan, 20 Oktober 2023

Penulis

ABSTRAK

Seperangkat alat yang digunakan untuk melindungi diri atau tubuh pekerja dari segala bahaya/kecelakaan kerja yang mungkin terjadi di lingkungan proyek difungsikan untuk mengurangi resiko cedera serius. APD terdiri dari, *Helm (Hard hat helmet)*, *masker (mask)*, *rompi (safety vest)*, dan *sepatu (boots)*. Kelengkapan APD pekerja proyek, sulit dipantau oleh pihak K3 di setiap bagian lingkungan proyek secara bersamaan. Hal ini dikarenakan tenaga kerja dan waktu yang terbatas sehingga, para pekerja bebas melepas APD mereka sebebasnya di lingkungan proyek. Padahal hal inilah yang meminimalisir cedera fatal pada pekerja itu sendiri. Dengan adanya keterbatasan waktu dan tenaga kerja diatas yang membuat tantangan dalam memantau pekerja sehingga dibutuhkan aplikasi yang dapat mendeteksi kelengkapan APD. Oleh karena itu metode You Only Look Once Versi 5 (YOLOv5) digunakan dalam aplikasi ini agar dapat mendeteksi setiap bagian APD. Pada penelitian yang akan dilakukan terdiri dari 7 kelas yaitu, *Gloves, Helmet, Non-helmet, Person, Shoes, Vest, Bare-arms*. Dataset yang dipakai berjumlah 1526 data yang setelah melalui tahap augmentasi akan menjadi 3958 images. Data tersebut akan dibagi menjadi 3628 *data training*, 295 *data validation*, 29 *data testing*. Penulis mendapatkan nilai akurasi sebesar 92,07% dalam pendeteksian APD. Hasil yang didapat menunjukkan bahwa metode YOLOv5 bagus dalam mendeteksi APD pada pekerja.

Kata kunci: Alat Pelindung Diri, pekerja, Keselamatan Kerja, YOLOv5

PERSONAL PROTECTIVE EQUIPMENT DETECTION FOR WORKER USING YOU ONLY LOOK ONCE (YOLOv5) METHOD

ABSTRACT

Toolset used for self-protection from any danger or accident that might happened in the environment. PPE consists of Hard Hat Helmet, Mask, Safety Vest, Boots. PPE is hard to monitored by Occupational Health and Safety (OHS) in every sector of the project environment at the same time. Because of labor and limited time, the workers took off their PPE as they like in the environment. Whereas in this case, it's to minimalized fatal injury of themselves. With this limited time and the amount of labor, the application is needed to detect PPE. You Only Look Once Version 5 (YOLOv5) method will be used in this application to cover every sector of the environment. In this study, there are 7 classes which are *Gloves, Helmet, Non-helmet, Person, Shoes, Vest, Bare-arms*. The dataset consists of 1526 data which will be augmented and become 3958 images. The Data will be divided to 3628 *training data*, 295 *validation data*, and 29 *testing data*. The writer got the accuracy score of 92,07% in PPE detection. The result shows that YOLOv5 method is good in detecting PPE in workers body.

Keyword: Personal Protective Equipment, workers, Safety work, YOLOv5

DAFTAR ISI

| | |
|--------------------------------|-------------|
| PERSETUJUAN | iii |
| PERNYATAAN | iv |
| UCAPAN TERIMAKASIH | v |
| ABSTRAK | viii |
| ABSTRACT | ix |
| DAFTAR ISI | x |
| DAFTAR GAMBAR | xiii |
| DAFTAR TABEL | xiv |
| 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 | 4 |
| 1.7. Sistematika Penulisan | 4 |
| BAB 2 LANDASAN TEORI | 6 |
| 2.1 Alat Pelindung Diri | 6 |
| 2.1.1 Helm (Hard Hat Helmet) | 6 |
| 2.1.2 Sarung Tangan (Gloves) | 7 |
| 2.1.3 Rompi (Safety Vest) | 8 |
| 2.1.4 Sepatu Boots | 8 |
| 2.2 Computer Vision | 9 |
| 2.3 Real Time Object Detection | 9 |
| 2.4 Image Preprocessing | 9 |
| 2.4.1 Resize | 10 |
| 2.4.2 Labelling | 10 |
| 2.4.3 Augmentation | 10 |

| | |
|--|-----------|
| 2.5 Convolutional Neural Network (CNN) | 10 |
| 2.6 You Only Look Once (YOLO) | 11 |
| 2.7 You Only Look Once V 5 | 12 |
| 2.7.1 Backbone: CSPDarknet53 | 13 |
| 2.7.2 Neck | 13 |
| 2.7.3 Head | 14 |
| 2.8 Evaluasi Confusion Matrix | 14 |
| 2.8.1 Akurasi (Accuracy) | 15 |
| 2.8.2 Presisi (Precision) | 15 |
| 2.8.3 Recall | 15 |
| 2.8.4 F1- Score | 15 |
| 2.9 Penelitian Terdahulu | 16 |
| 2.10 Perbedaan Penelitian Terdahulu | 18 |
| BAB 3 ANALISIS DAN PERANCANGAN SISTEM | 19 |
| 3.1 Analisis Sistem | 19 |
| 3.2 Image acquisition | 20 |
| 3.3 Pre-processing | 22 |
| 3.3.1 Resizing | 22 |
| 3.3.2 Labelling | 23 |
| 3.4 Augmentasi | 24 |
| 3.5 Image Classification | 26 |
| 3.6 YOLOv5 Model | 27 |
| 3.6.1 Clone repository | 27 |
| 3.6.2 Install Requirement | 28 |
| 3.6.3 Menambahkan <i>data.names</i> dan <i>data.yaml</i> | 28 |
| 3.6.4 Konfigurasi YOLOv5 | 28 |
| 3.7 Proses Training | 28 |
| 3.8 Proses Testing | 29 |
| 3.9 Perancangan Antarmuka Sistem | 29 |
| 3.9.1 Rancangan Tampilan Utama | 29 |
| 3.9.2 Rancangan Tampilan Open camera | 30 |
| 3.9.3 Rancangan Tampilan upload files | 31 |

| | |
|--|-----------|
| BAB 4 IMPLEMENTASI DAN PENGUJIAN | 32 |
| 4.1 Implementasi Sistem | 32 |
| 4.1.1 Perangkat lunak dan perangkat keras | 32 |
| 4.2 Implementasi Data | 32 |
| 4.3 Pelatihan Sistem | 33 |
| 4.4 Prosedur Operasional | 35 |
| 4.5 Implementasi Perancangan Sistem | 36 |
| 4.5.1 Tampilan Halaman Utama | 36 |
| 4.5.2 Tampilan Mendeteksi dengan webcam | 37 |
| 4.5.3 Tampilan Mendeteksi dengan Upload file | 37 |
| 4.6 Pengujian Sistem | 38 |
| BAB 5 KESIMPULAN DAN SARAN | 52 |
| 5.1 Kesimpulan | 52 |
| 5.2 Saran | 52 |
| DAFTAR PUSTAKA | 53 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2.1 Alat Pelindung Diri Proyek | 6 |
| Gambar 2.2 Hard Hat Helmet | 7 |
| Gambar 2.3 Sarung Tangan | 7 |
| Gambar 2.4 Safety Vest | 8 |
| Gambar 2.5 Sepatu Boots | 8 |
| Gambar 2.6 Tugas Computer Vision | 9 |
| Gambar 2.7 Arsitektur CNN | 10 |
| Gambar 2.8 Ilustrasi proses pada YOLO | 12 |
| Gambar 2.9 Arsitektur umum YOLOv5 | 12 |
| Gambar 2.1 Perbandingan Backbone Network | 13 |
| Gambar 2.11 Ilustrasi PANet | 13 |
| Gambar 3.1 Arsitektur umum | 20 |
| Gambar 3.2 Isi file.txt | 23 |
| Gambar 3.3 Proses membuat Bounding Box | 23 |
| Gambar 3.4 Menentukan Class | 24 |
| Gambar 3.5 Proses Mosaic | 25 |
| Gambar 3.6 Pembagian data | 25 |
| Gambar 3.7 Sebelum NMS | 27 |
| Gambar 3.8 Sesudah NMS | 27 |
| Gambar 3.9 Clone repository | 27 |
| Gambar 3.10 Halaman Utama | 30 |
| Gambar 3.11 Jendela Pop up open webcam | 30 |
| Gambar 3.12 Halaman Upload Files | 31 |
| Gambar 4.1 Gambar pada dataset | 33 |
| Gambar 4.2 Google Colab | 33 |
| Gambar 4.3 Grafik Loss | 35 |
| Gambar 4.4 Precision-Recall Curve | 35 |
| Gambar 4.5 Flask Run | 36 |
| Gambar 4.5 Link dari flask | 36 |
| Gambar 4.6 Tampilan Utama | 36 |
| Gambar 4.7 Jendela Pop-up webcam | 37 |
| Gambar 4.8 Tampilan Upload file | 37 |

DAFTAR TABEL

| | |
|---------------------------------------|----|
| Tabel 2.1 Confusion Matrix | 14 |
| Tabel 2.2 Penelitian Terdahulu | 17 |
| Tabel 3.1 Pembagian data yang didapat | 20 |
| Tabel 3.2 Contoh data yang didapat | 21 |
| Tabel 4.1 Langkah Training | 34 |
| Tabel 4.2 Pengujian Sistem | 38 |
| Tabel 4.3 Confusion matrix | 49 |
| Tabel 4.4 Tabel Hasil Pengujian | 49 |
| Tabel 4.5 Hasil perhitungan | 51 |

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pada masa sekarang, Indonesia sedang melakukan pembangunan infrastruktur secara menyeluruh. Baik pada sektor transportasi, pengembangan wilayah dan Kawasan, Infrastruktur sumber daya air, maupun energi dan Teknologi. Dalam setiap pembangunan ini, pastinya diperlukan pekerja yang ahli dan cekatan. Kecelakaan tidak dapat ditebak kapan terjadi, sehingga pekerja hanya mampu mencegah kemungkinan cedera yang buruk dengan menerapkan Kesehatan dan Keselamatan Kerja (K3). Pada tahun 2020, ada 221.740 kasus kecelakaan kerja di Indonesia. Jumlah itu meningkat dari 234.370 kasus pada tahun 2021 menjadi 265.334 kasus pada November 2022 (Badan Penyelenggara Jaminan Sosial, 2023). 2 tahun terakhir, angka kasus kecelakaan kerja di Indonesia menunjukkan kenaikan.

Dari kasus kecelakaan kerja tersebut, ada yang bersifat fatal atau sekedar cedera ringan. Tingkat kompetensi pekerja terhadap K3 yang rendah menjadi penyebab kecelakaan yang dapat dihindari menjadi fatal. Untuk mencegah cedera di lingkungan proyek, alat pelindung diri adalah komponen K3 yang harus digunakan. Alat pelindung diri (APD) adalah alat yang digunakan untuk melindungi diri atau tubuh pekerja dari bahaya dan kecelakaan kerja yang mungkin terjadi di proyek. APD terdiri dari, *Helm* (*Hard hat helmet*), *masker* (*mask*), *rompi* (*safety vest*), dan *sepatu* (*boots*).

Kelengkapan APD pekerja proyek, sulit dipantau oleh pihak K3 di setiap bagian lingkungan proyek secara bersamaan. Hal ini dikarenakan tenaga kerja dan waktu yang terbatas sehingga, para pekerja bebas melepas APD mereka sebebasnya di lingkungan proyek. Padahal hal inilah yang meminimalisir cedera fatal pada pekerja itu sendiri. Dengan adanya keterbatasan waktu dan tenaga kerja diatas yang membuat tantangan dalam memantau pekerja sehingga dibutuhkan aplikasi yang dapat mendeteksi kelengkapan APD. Oleh karena itu metode *You Only Look Once* Versi 5 (YOLOv5) digunakan dalam aplikasi ini agar dapat mendeteksi setiap bagian APD.

Beberapa penelitian mengenai deteksi kelengkapan APD yang sebelumnya pernah dilakukan, diantaranya adalah penelitian dengan metode *Mask Region Convolutional*

Neural Network (Mask R-CNN) yang dilakukan oleh (Laily, M.E. et al., 2022) mendapatkan akurasi 95% dimana penelitian ini menggunakan dataset yang terdiri dari *helmet* (pelindung kepala) dan *vest* (rompi). Penelitian lain yang mendeteksi masker menggunakan *ResNet* dan *YOLOv2* oleh (Loey, M. et al., 2020) dan membandingkan *mAP* dan *Loss* pada 4 metode yaitu *SSD*, *Faster R-CNN*, *Mask R-CNN*, dan metode yang digunakan didapatkan bahwa metode yang digunakan menghasilkan akurasi paling tinggi yaitu 87,7% dan *Loss* paling kecil yaitu 0,043.

Penelitian oleh (Zhang, W. et al., 2021) berjudul “*Face mask wearing detection algorithm based on improved YOLOv4*” yang mana penelitian ini mendeteksi masker pada orang di tempat tertentu menghasilkan akurasi 95,2% dengan data training yang berisi 2685 *image*. Selanjutnya, penelitian oleh (Nepal, U. et al., 2021) yang membandingkan model *YOLOv3*, *YOLOv4* dan *YOLOv5* untuk mendeteksi lokasi pendaratan darurat *UAV* yang baik, dan didapatkan bahwa *YOLOv5* merupakan metode yang memiliki akurasi tertinggi.

Kemudian penelitian oleh (Zhang, Y. et al., 2021) yang berjudul “*Real-time Vehicle Detection Based on improved YOLOv5*” mendeteksi kendaraan secara real-time menggunakan metode *YOLOv5* menunjukkan bahwa metode bagus untuk mengenal objek kendaraan yang berbeda-beda seperti bus, minibus, sedan, taxi, truk, SUV dengan akurasi rata-rata diatas 80%. Dan juga penelitian oleh (Putra, I.K. et al., 2022) yang membuat sistem pengawasan K3 pada PT. PLN Indonesia Power Bali GPU juga memiliki dataset yang berupa APD pada lingkungan PT yaitu *helmet* (pelindung kepala) dan *facemask* (Masker). Mereka menggunakan metode *YOLOv5* dan mendapat akurasi deteksi paling besar yaitu 90%.

Penulis mengajukan penelitian untuk membuat aplikasi *web* berdasarkan latar belakang dan temuan penelitian sebelumnya dengan menerapkan metode *You Only Look Once (YOLO)* yang dapat membantu K3 untuk mendeteksi kelengkapan APD pekerja proyek. Penelitian ini diberi judul “**Deteksi Kelengkapan Alat Pelindung Diri (APD) untuk pekerja proyek menggunakan metode You Only Look Once Versi 5 (YOLOv5)**”.

1.2 Rumusan Masalah

Salah satu faktor yang menghambat pihak K3 yaitu keterbatasan waktu dan tenaga kerja dalam memantau kelengkapan APD pekerja tersebut. Hal ini dikarenakan jumlah pekerja

yang lebih banyak dari pengawasnya. Dan juga, dikarenakan lingkungannya yang luas sehingga tidak terawasi dengan baik. Sistem dapat mengenali setiap bagian APD pada pekerja, dengan cara melabel setiap bagian APD yang ditentukan. Dikarenakan hal tersebut, diperlukan sistem yang mampu menggunakan citra bentuk APD yaitu helm, rompi, sarung tangan dan sepatu boots dan juga Non-APD yaitu *Bare-arms* dan *Non-Helmet* dalam mendeteksi kelengkapan APD tersebut pada pekerja.

1.3 Tujuan Penelitian

Tujuan penelitian ini yaitu pengaplikasian metode *You Only Look Once Versi 5 (YOLOv5)* untuk melakukan deteksi kelengkapan APD dan menganalisis kinerja metode yang akan dipakai .

1.4 Batasan Masalah

Adapun batasan masalah untuk mencegah meluasnya ruang lingkup permasalahan dalam penelitian ini adalah:

1. Penelitian ini hanya mengidentifikasi bagian dari APD pada manusia (Pekerja)
2. Data citra APD menggunakan data yang berasal dari *Kaggle* dan kamera *smartphone*.
3. Ekstensi file citra yaitu .jpg atau .jpeg
4. Terdapat empat jenis bagian dari APD yang akan dideteksi yaitu *Helmet*, *Vest*, *Gloves* dan *Shoes*.
5. Terdapat tiga jenis bagian dari Non APD yang akan dideteksi yaitu *Person*, *Non-helmet*, dan *Bare-arms*.
6. Sistem mampu mendeteksi pada batas jarak 18 meter, untuk jarak yang lebih dari angka tersebut maka sistem kurang dapat mendeteksi dengan baik.
7. *Output* yang dihasilkan adalah Status kelengkapan APD pada orang tersebut.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah :

1. Membantu pengawasan K3 pada pekerja proyek
2. Mengetahui kinerja metode *You Only Look Once Versi 5 (YOLOv5)* dalam mendeteksi APD pada manusia.
3. Menjadi refrensi dan bahan pembelajaran bagi penelitian-penelitian selanjutnya untuk menggunakan metode yang sama dalam mendeteksi suatu objek

1.6 Metodologi Penelitian

Berikut merupakan tahapan-tahapan yang akan dilakukan dalam penelitian ini:

1. Studi Literatur

Tahap pembelajaran, pengumpulan Informasi, dan pencarian data yang diperlukan pada penelitian. Adapun referensi seputar Alat Pelindung Diri (APD), *real-time image processing*, dan metode *You Only Look Once (YOLO)* bisa didapati dari banyak sumber seperti artikel, jurnal, skripsi, dan sebagainya.

2. Analisis Permasalahan

Pada bagian ini, informasi dari studi literatur sebelumnya dianalisis untuk meningkatkan pemahaman dan pemahaman tentang metode *YOLO* yang akan digunakan untuk mendeteksi APD.

3. Perancangan Sistem

Pada bagian ini, akan dibuat rancangan arsitektur umum, penentuan pengujian dan melatih *dataset* serta merancang antarmuka hingga hasil akhir berdasarkan dua tahap sebelumnya.

4. Implementasi

Pada tahapan ini, diimplementasikan rancangan sistem sebelumnya dalam bentuk kode program agar menjadi sistem yang sesuai dengan tujuan penelitian.

5. Pengujian Sistem

Pada bagian ini, sistem akan diuji untuk memastikan bahwa aplikasi telah berfungsi dengan baik dan melihat kecocokan metode yang digunakan untuk penelitian ini.

6. Penyusunan Laporan

Pada tahapan ini, peneliti menyusun laporan dan dokumentasi dari setiap proses penelitian yang telah dilakukan untuk dipaparkan dari penelitian ini.

1.7 Sistematika Penulisan

BAB 1: Pendahuluan

Pada bab satu berisi tentang latar belakang, rumusan masalah, tujuan penelitian,

batasan masalah, manfaat penelitian, dan sistematika penulisan.

BAB 2: Landasan Teori

Pada bab dua akan dibahas teori tentang APD, *image processing*, *Computer Vision*, dan Metode *You Only Look Once (YOLO)*

BAB 3: Analisis dan Perancangan Sistem

Pada bab tiga akan dibahas tentang rancangan sistem yang akan dikerjakan dari arsitektur umum yaitu dari fase *input*, *image processing*, *image classification output*, dan desain antarmuka.

BAB 4: Implementasi dan Pengujian

Pada bab empat akan dibahas mengenai hasil dan pengujian analisis dan juga rancangan sistem yang akan dibangun juga hasil dari penerapan model.

BAB 5: Kesimpulan dan Saran

Pada bab lima mengenai hasil dan saran dari peneliti dalam pengembangan penelitian agar menjadi lebih baik di masa depan.

BAB 2

LANDASAN TEORI

2.1 Alat Pelindung Diri

Alat pelindung diri (APD) adalah alat yang dapat melindungi seseorang dengan mengisolasi sebagian atau seluruh tubuhnya dari bahaya yang mungkin terjadi di tempat kerja (Menteri Tenaga Kerja dan Transmigrasi Republik Indonesia, 2010). APD sangat penting karena risiko kecelakaan yang tinggi di bidang konstruksi. Angka kecelakaan kerja masih perlu menjadi perhatian yang lebih bagi pemerintah dan perusahaan yang beroperasi di bidang konstruksi. Kementerian Ketenagakerjaan Republik Indonesia melaporkan 110.285 kasus kecelakaan pada tahun 2015, 105.182 kasus pada tahun 2016, dan 80.392 kasus pada tahun 2017. Dalam hirarki pengendalian resiko, Alat Pelindung Diri (APD) merupakan bentuk terakhir dari pengendalian resiko.

Adapun contoh dari gambar Alat Pelindung Diri Proyek pada gambar 2.1.



Gambar 2.1 Alat Pelindung Diri Proyek
(Sumber : merahputih.com)

2.1.1 Helm (*Hard Hat Helmet*)

Hard Hat Helmet atau alat pelindung kepala merupakan komponen paling penting yang mesti dipakai oleh setiap orang saat memasuki lokasi konstruksi. Alat pelindung kepala dipakai agar kepala terlindungi dari benda yang terjatuh, uap panas atau dingin, sampai potensi kebakaran maupun korosif. Alat pelindung kepala memiliki

beberapa kategori yang telah dibagi atas fungsi dan identitas pegawai.



Gambar 2.2 *Hard Hat Helmet*
(kemkes.go.id)

2.1.2 Sarung Tangan (*Gloves*)

Sarung tangan keselamatan merupakan alat pelindung diri (APD) untuk melindungi tangan dari macam resiko maupun bahaya kemungkinan cedera saat bekerja di lingkungan konstruksi. Bentuk perlindungan yang diberikan yaitu pencegahan terhadap luka sayatan, tertusuk benda tajam, luka yang disebabkan bakaran, benturan, dan bentuk cedera yang lain. Pada umumnya industri atau lingkungan kerja yang memerlukan sarung tangan keselamatan yaitu industri pabrik, sektor otomotif, laboratorium, Rumah Sakit, sektor pertambangan, sektor minyak dan gas, bagian perkebunan, dan lain-lain.



Gambar 2.3 Sarung Tangan
(diptasafety.com)

2.1.3 Rompi (*Safety Vest*)

Safety vest adalah rompi yang terbuat dari berbagai bahan yang dapat dipilih, seperti *nylon*, *drill*, *net* atau jaring, *polyster*, atau plastik, dan beberapa sisinya dilengkapi dengan reflektor atau pemantul cahaya. Seseorang dapat menghindari bahaya terserempet dan tertabrak kendaraan yang beroperasi di dekatnya dengan mengenakan baju pengaman. Selama proses bongkar muat barang dengan alat berat, pekerja dapat menghindari bahaya terjepit barang atau jatuh barang saat alat berat bergerak.



Gambar 2.4 *Safety Vest*
(nindikarya.co.id)

2.1.4 Sepatu (*Shoes*)

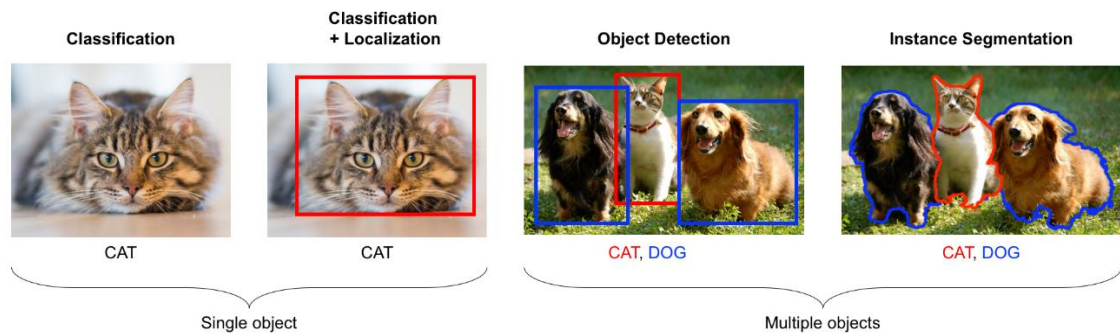
Sepatu keselamatan kerja melindungi kaki pekerja dari bahaya jatuh dari benda berat, percikan cairan, tertusuk oleh benda tajam, atau terpeleset. Alat pelindung kaki termasuk *boots*, sepatu anti listrik, sepatu anti licin, dan sepatu tapak besi. Sepatu tapak besi adalah sepatu khusus yang terbuat dari pelat besi yang melindungi jari-jari kaki pekerja dari jatuh dan benturan dari benda-benda yang terbuat dari bahan bangunan.



Gambar 2.5 Sepatu Boots
(Tokopedia.com)

2.2 Computer Vision

Computer Vision merupakan bagaimana komputer/mesin dapat melihat. Dengan menganalisis gambar atau video, teknik *Computer Vision* dapat memvisualisasikan data. Tujuan utama dari *Computer Vision* adalah agar komputer atau mesin dapat meniru atau bahkan mengungguli kemampuan persepsi mata dan otak manusia. (Wahyu, A.P. , 2016).



Gambar 2.6 Tugas *Computer Vision*
(oreily.com)

2.3 Real Time Object Detection

Object detection adalah teknologi komputer yang termasuk ke dalam *Computer Vision* dan pemrosesan gambar untuk mendeteksi suatu objek dalam suatu citra digital. *Object detection* dapat mendeteksi objek dalam suatu gambar seperti wajah, tumbuhan, kendaraan, atau lainnya dan melacak pergerakan orang untuk meningkatkan automasi pekerjaan. (Zhang, W. et al., 2021).

Real time object detection merupakan proses mendeteksi objek yang mana objek diambil secara langsung (*live*) menggunakan kamera sebagai input. Data yang diambil secara langsung akan diklasifikasikan tergantung dari label yang dipasang. *Real time object detection* mementingkan *Frame Per Second (FPS)* karena kecepatan *frame* pada data memengaruhi hasil deteksi.

2.4 Image Preprocessing

Pada tahapan ini, data citra APD akan diolah agar mendapat hasil yang lebih bagus untuk diproses pada tahapan berikutnya. *Resize*, *Augmentation*, dan *Labelling* adalah tahapan *pre-processing* yang akan dipakai dalam penelitian ini.

2.4.1 *Resize*

Resize citra atau penskalaan adalah mengecilkan ukuran citra yang akan digunakan. Proses ini berguna untuk meringankan sistem agar pemrosesan citra akan lebih cepat dan tidak menghabiskan banyak memori penyimpanan. Pada penelitian ini, *resize* akan dilakukan dengan cara membandingkan ukuran citra dari segmentasi dengan target ukuran citra.

2.4.2 *Labelling*

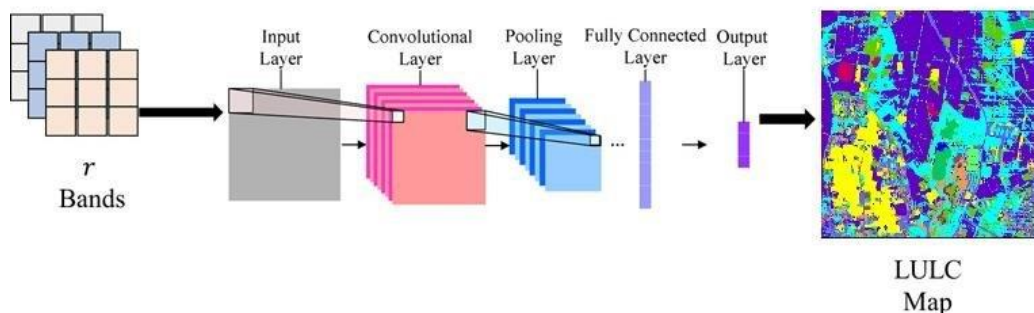
Labelling adalah proses memberikan *label* nama pada gambar yang telah di *resize* sebelumnya untuk pengujian sistem. Proses *Labelling* dilakukan agar komputer dapat menandai citra yang diuji sesuai dengan *label* yang telah diberikan.

2.4.3 *Augmentation*

Augmentation merupakan proses memperbanyak gambar dengan mengubah citra asli ke bentuk yang berbeda-beda dengan cara seperti memutar citra, menambahkan *noise* pada citra, atau membalik gambar. Peneliti akan menambahkan *noise* pada citra yang nantinya akan menghasilkan beberapa citra lainnya.

2.5 *Convolutional Neural Network (CNN)*

Convolutional Neural Network adalah algoritma *deep learning* yang dibuat untuk melangsungkan pengolahan data dua dimensi. *CNN* digunakan untuk mengklasifikasikan data yang sudah dilabel dengan tujuan untuk pengelompokan suatu data *input* kepada data yang sudah tersedia. Hasil dari setiap layer pada *CNN* merupakan sebuah pola pada bagian dari citra yang nantinya akan diklasifikasikan. Berikut merupakan arsitektur umum *CNN*.



Gambar 2.7 Arsitektur *CNN*
(researchgate.net)

Ada 4 tahapan yang terdapat pada *CNN*, yaitu:

1. *Input Layer*

Pada tahapan ini, akan disimpan *pixel* gambar dari citra yang akan menjadi input.

2. *Convolutional Layer*

Tahapan ini memiliki *filter*, parameter yang akan dipelajari pada tahap training nantinya. Ukuran *filter* itu sendiri biasanya lebih kecil dari gambar aslinya. Pada tahap inilah, fitur pada *input layer* sebelumnya akan diekstrak.

3. *Pooling layer*

Tahapan ini membantu mengurangi terjadinya *overfitting* selama training model dengan melakukan pemadatan pada fitur di *feature map*.

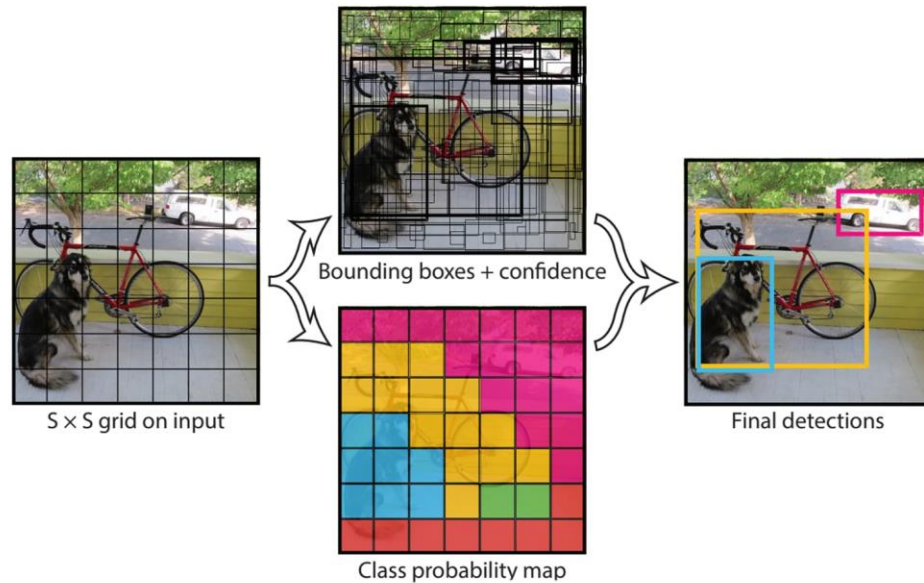
4. *Fully Connected layer*

Disebut juga lapisan *linear*, yaitu menghubungkan setiap *input neuron* kepada *output neuron*. Sehingga, setiap *layer* memiliki koneksi terhadap setiap *layer* sebelumnya.

2.6 You Only Look Once (YOLO)

You Only Look Once (YOLO) adalah algoritma yang dikembangkan untuk melakukan deteksi objek secara *real-time*. Model yang dibuat akan diterapkan pada citra di lokasi dan skala tertentu. Lokasi pada citra dengan skor tertinggi akan dianggap sebagai suatu pendeteksian.

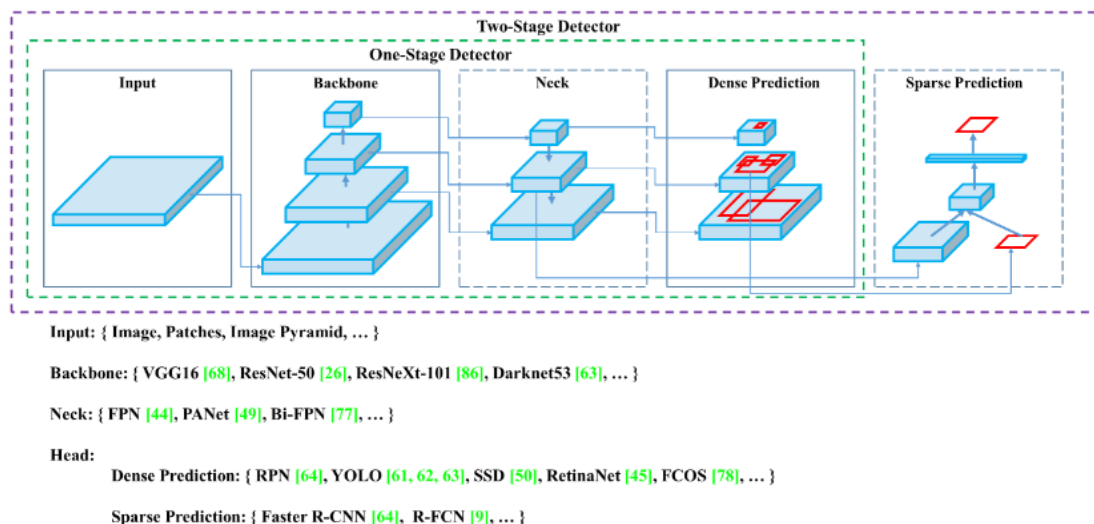
YOLO menemukan objek di foto dengan menggunakan *Artificial Neural Network (ANN)*. Pendekatan ini memecah citra menjadi lebih banyak bagian dan melakukan prediksi setiap bagian *bounding box* dan kemungkinan untuk setiap bagiannya. *Bounding box* ini akan dilakukan perbandingan dengan berbagai kemungkinan yang diprediksi. Arsitektur *YOLO* juga mirip dengan *CNN*. *YOLO* hanya menggunakan *Convolutional Layer* terakhir dan *Pooling Layer*. Cara kerja *YOLO* yaitu dengan melihat seluruh gambar sekali, kemudian melewati jaringan saraf sekali langsung mendeteksi *object* yang ada. Oleh karena itu disebut *You Only Look Once (YOLO)*.



Gambar 2.8 Ilustrasi proses pada *YOLO* (pacmann.io)

2.7 You Only Look Once V 5

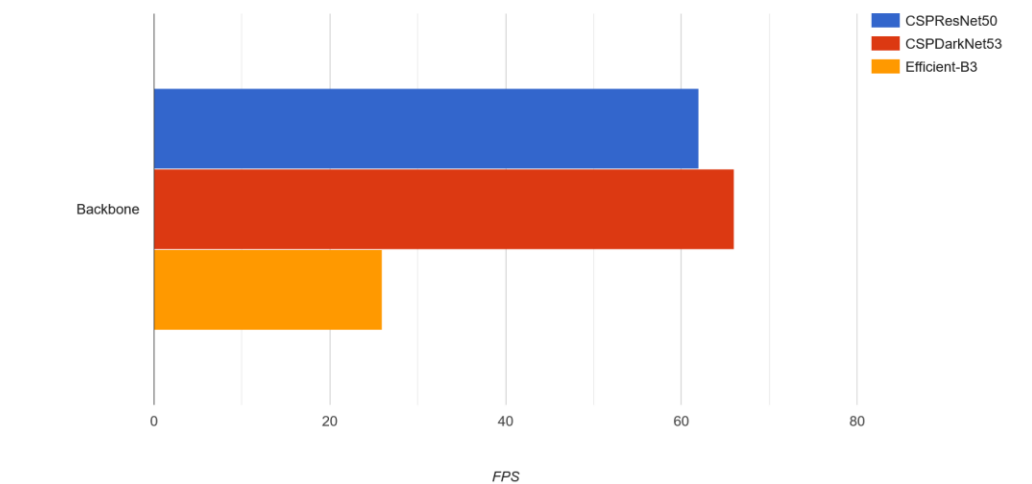
YOLO adalah jaringan deteksi objek, dan *YOLOv5* adalah metode terbaru yang dibuat melalui perkembangan metode *YOLO* (Tan et al. 2021). Arsitektur *YOLOv5* sama dengan *YOLO* versi sebelumnya. *YOLOv5* juga menggunakan *neural network* yang sama dengan versi 4 dan 3 yaitu *CSPDarknet53* pada *backbone*, *PANet* pada *Neck*, dan versi *Head* yang sama pada versi sebelumnya.



Gambar 2.9 Arsitektur umum *YOLOv5* (medium.com)

2.7.1 Backbone: CSPDarknet53

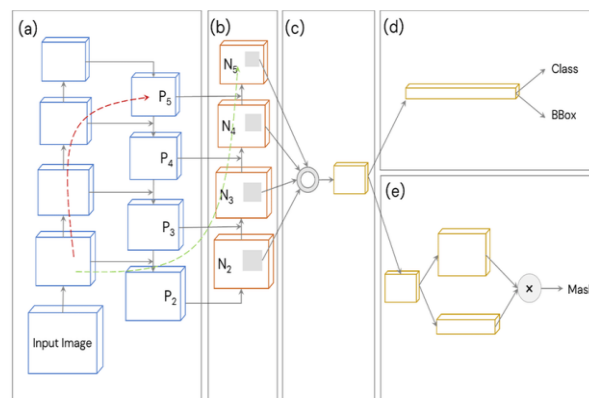
Backbone Network adalah tahapan yang mengarah pada *feature extraction*. Pada tahapan ini *backbone network* yang akan digunakan adalah CSPDarknet53 yang terdiri dari dua yaitu CSP (*Cross Stage Partial*) dan lapisan dasar *Convolutional*. Berikut merupakan perbandingan dari beberapa *backbone network* seperti CSPResNet50, CSPDarkNet53, dan Efficient-B3



Gambar 2.1 Perbandingan *Backbone Network* (researchgate.net)

2.7.2 Neck

Pada tahapan ini, akan dibentuk *pyramid feature* yang nantinya dipakai untuk memprediksi deteksi pada *pixel* yang kecil untuk meningkatkan proses segmentasi. YOLOv5 menggunakan *neck* yang sama dengan versi sebelumnya yaitu PANet (*Path Aggregation Network*) yang sudah dimodifikasi untuk menghasilkan akurasi yang lebih baik.



Gambar 2.11 Ilustrasi PANet (researchgate.net)

2.7.3 Head

Pada tahapan ini, akan dihasilkan kotak deteksi yang digunakan untuk mendapatkan koordinat atau bagian dari *bounding box* objek untuk menentukan *confidence score* pada suatu kelas. Akan ada tujuh vector yang mewakili satu koordinat tertentu. Tahap *Head* yang ada pada *YOLOv5* sama dengan proses pada *YOLOv3* dan *YOLOv4*. Pada tahap ini juga akan diimplementasikan fungsi *IOU* (*Intersection Over Union*) dan *Non max suppression* untuk mencegah hasil kotak yang bertimpahan.

2.8 Evaluasi Confusion Matrix

Cara mengevaluasi performa model yang telah kita kembangkan adalah dengan membandingkan hasil prediksi model dengan keadaan sebenarnya (realita).

Tabel 2.1 Confusion Matrix

| | | Realita | |
|----------|-----|---------------------|---------------------|
| | | Yes | No |
| Prediksi | Yes | True Positive (TP) | False Positive (FP) |
| | No | False Negative (FN) | True Negative (TN) |

ada Confusion matrix, ada empat nilai yang dihasilkan yaitu:

- True Positive* (TP), yaitu jumlah data yang bernilai positif dan diprediksi benar sebagai positif.
- False Positive* (FP), yaitu jumlah data yang bernilai negatif tetapi diprediksi sebagai positif.
- False Negative* (FN), yaitu jumlah data yang bernilai positif tetapi diprediksi sebagai negatif.
- True Negative* (TN), yaitu jumlah data yang bernilai negatif dan diprediksi benar sebagai negatif.

Confusion matrix digunakan untuk menghitung nilai *accuracy*, *presicion*, *recall*, dan *F1-score*. Keempat metode evaluasi tersebut untuk mengukur performa dari *classifier* atau algoritma *machine learning* yang digunakan untuk melakukan prediksi.

Menghitung metode evaluasi dengan menggunakan confusion matrix sebagai berikut:

2.8.1 Akurasi (*Accuracy*)

Akurasi (*accuracy*) adalah persentase prediksi yang benar dari semua pengamatan.

$$\begin{aligned} \text{Akurasi} &= (\text{Prediksi benar}) / (\text{Semua kasus}) \\ \text{Akurasi} &= (TP + TN) / (TP + FP + TN + FN) \end{aligned}$$

(1)

2.8.2 Presisi (*Precision*)

Presisi (*precision*) adalah persentase kasus yang diprediksi positif (PB + PP) yang ternyata positif (PB).

$$\begin{aligned} \text{Presisi} &= (\text{True Positive}) / (\text{Total prediksi positif}) \\ \text{Presisi} &= (TP) / (TP + FP) \end{aligned}$$

(2)

2.8.3 Recall

Recall mengukur pecahan kasus positif (TP + FN) yang diidentifikasi dengan benar (TP).

$$\begin{aligned} \text{Recall} &= (\text{True Positive}) / (\text{Total realita positif}) \\ \text{Recall} &= (TP) / (TP + FN) \end{aligned}$$

(3)

2.8.4 F1-Score

Nilai *F1-Score* atau dikenal juga dengan nama *F-Measure* didapatkan dari hasil *Precision* dan *Recall* antara kategori hasil prediksi dengan kategori sebenarnya.

$$F1 = \frac{2 \times \text{Presisi} \times \text{recall}}{\text{Presisi} + \text{recall}}$$

(4)

2.9 Penelitian Terdahulu

Terdapat penelitian mengenai deteksi kelengkapan APD yang sebelumnya pernah dilakukan, diantaranya adalah penelitian oleh (Putra, I.K. et al., 2022) yang membuat sistem pengawasan K3 pada PT. PLN Indonesia Power Bali GPU juga memiliki dataset yang berupa APD pada lingkungan PT yaitu *helmet* (pelindung kepala) dan *facemask* (Masker). Mereka menggunakan metode YOLOv5 dan mendapat akurasi deteksi paling besar yaitu 90%.

Penelitian yang berhubungan dengan metode *Mask Region Convolutional Neural Network (Mask R-CNN)* yang dilakukan oleh (Laily, M.E. et al., 2022) mendapatkan akurasi 95% dimana penelitian ini menggunakan dataset yang terdiri dari *helmet* (pelindung kepala) dan *vest* (rompi). Penelitian lain yang mendeteksi masker menggunakan *ResNet-50* dan *YOLOv2* oleh (Loey, M. et al., 2020) dan membandingkan *mAP* dan *Loss* pada 4 metode yaitu *SSD*, *Faster R-CNN*, *Mask R-CNN*, dan metode yang digunakan didapatkan bahwa metode yang digunakan menghasilkan akurasi paling tinggi yaitu 87,7% dan *Loss* paling kecil yaitu 0,043.

Penelitian oleh (Zhang, W. et al., 2021) berjudul “*Face mask wearing detection algorithm based on improved YOLOv4*” yang mana penelitian ini mendeteksi masker pada orang di tempat tertentu menghasilkan akurasi 95,2% dengan data training yang berisi 2685 *image*. Selanjutnya, penelitian oleh (Nepal, U. et al., 2021) yang membandingkan model *YOLOv3*, *YOLOv4* dan *YOLOv5* untuk mendeteksi lokasi pendaratan darurat *UAV* yang baik, dan didapatkan bahwa *YOLOv5* merupakan metode yang memiliki akurasi tertinggi.

Kemudian penelitian oleh (Zhang, Y. et al., 2021) yang berjudul “*Real-time Vehicle Detection Based on improved YOLOv5*” mendeteksi kendaraan secara real-time menggunakan metode *YOLOv5* menunjukkan bahwa metode bagus untuk mengenal objek kendaraan yang berbeda-beda seperti bus, minibus, sedan, taxi, truk, SUV dengan akurasi rata-rata diatas 80%. Dan juga penelitian mengenai deteksi kelengkapan APD yang sebelumnya pernah dilakukan, diantaranya adalah penelitian dengan metode *Mask Region Convolutional Neural Network (Mask R-CNN)* yang dilakukan oleh (Laily, M.E. et al., 2022) mendapatkan akurasi 95% dimana penelitian ini menggunakan dataset yang terdiri dari *helmet* (pelindung kepala) dan *vest* (rompi).

Tabel 2.2 Penelitian Terdahulu

| No. | Penulis | Metode | Keterangan |
|------------|--|--|---|
| 1. | Milzamah Elvi Laily, Fathorazi Nur Fajri, dan Gulpi Qorik Oktagalu Pratamasunu (2022) | <i>Mask Region Convolutional Neural Network</i> | Penelitian ini membahas tentang deteksi Alat Pelindung Diri yang terdiri dari Helmet dan Vest menghasilkan tingkat akurasi 95% |
| 2. | Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N.Taha dan Nour Eldeen M. Khalifa (2020) | <i>ResNet- 50 dan You Only Look Once Version 2</i> | Penelitian ini membahas tentang deteksi masker dan membandingkan 4 metode dan didapatkan metode yang digunakan dengan akurasi tertinggi yaitu 87,7% dan Loss terendah yaitu 0,043 |
| 3. | Jimin Yu dan Wei Zhang (2021) | <i>You Only Look Once Version 4</i> | Penelitian ini membahas tentang deteksi masker pada wajah dengan data training sebanyak 2685 image dan didapatkan akurasi sebesar 95,2% |
| 4. | Upesh Nepal <i>et al.</i> (2021) | <i>You Only Look Once Version 3 , 4 dan 5</i> | Penelitian ini membahas mengenai perbandingan ketiga metode dalam mendeteksi lokasi pendaratan UAV terbaik dan disimpulkan YOLOv5 merupakan metode terbaik. |

| No. | Penulis | Metode | Keterangan |
|-----|---|-------------------------------------|---|
| 5. | Yu Zhang <i>et al.</i> (2021) | <i>You Only Look Once Version 5</i> | Penelitian ini membahas tentang deteksi kendaraan yang terdiri dari bus, minibus, sedan, taxi, truk, dan SUV secara real-time dengan akurasi rata-rata diatas 80% |
| 6. | I Kadek Angga Aditya Putra <i>et al.</i> (2022) | <i>You Only Look Once Version 5</i> | Penelitian ini membuat sistem pengawasan K3 pada PT. PLN Indonesia Power Bali GPU juga memiliki dataset yang berupa APD pada lingkungan PT yaitu helmet (pelindung kepala) dan facemask (Masker) dengan akurasi 90% |

2.10 Perbedaan Penelitian Terdahulu

Terdapat beberapa penelitian terdahulu yang menggunakan objek penelitian, waktu penelitian, tempat pelaksanaan penelitian dan batasan berbeda yang digunakan selama penyelidikan. Penulis memakai *You Only Look Once Versi 5 (YOLOv5)* sebagai metode untuk mendeteksi objek yang diamati yaitu komponen Alat Pelindung Diri (APD) , yang mana berbeda dari penelitian sebelumnya seperti penelitian oleh (I Kadek Angga Aditya Putra *et al.*, 2022) dan (Milzamah *et al.*, 2022) yang menggunakan objek yang berbeda yaitu *Helmet* dan *Safety vest* dan juga Facemask. Penulis juga membuat sistem yaitu aplikasi berbasis *web* dan sistem dapat dipakai secara *real-time* dalam penelitian ini agar pengguna lebih mudah dalam penggunaan sistem.

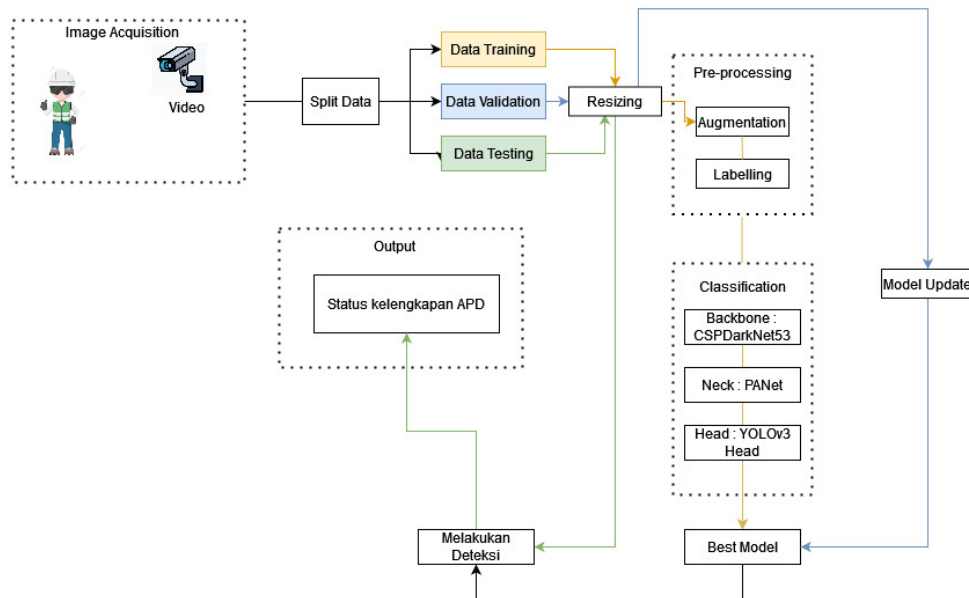
BAB 3

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini, akan disajikan perancangan sistem dalam aplikasi pendeteksian APD. Langkah yang akan dilakukan mengenai pengambilan data, *preprocessing*, pendeteksian menggunakan YOLOv5, dan proses *training* dan *testing* pada metode YOLOv5. Dalam proses pembelajaran algoritma, data pelatihan digunakan untuk melatih tingkat akurasi dengan menyesuaikan parameter saat ini agar sesuai dengan data yang disajikan, ini memungkinkan untuk memahami informasi dalam data. Sementara itu, data uji digunakan untuk mengevaluasi kinerja algoritma setelah dilatih. Data tersebut didistribusikan sebagai berikut: 3648 untuk data pelatihan, 295 untuk data validasi, dan 15 untuk data pengujian, dan seluruh data dengan ekstensi JPG.

3.1 Analisis Sistem

Arstitektur umum penelitian terdapat pada gambar 3.1 untuk mendeteksi APD yang akan melalui beberapa langkah. Proses awal yang akan dilakukan yaitu mengumpulkan data yang nantinya data tersebut akan dibagi menjadi 3 yaitu data *training*, *validation* dan *testing* yang mana seluruh data akan dilakukan *resizing*, dan data *training* dan *validation* akan melewati tahap *preprocessing* sedangkan data *testing* langsung menuju model yang telah di *training*. Untuk langkah pertama yaitu mengambil data citra dari *Closed Circuit Television* (CCTV) lalu dikirimkan ke langkah *preprocesssing* dahulu dimana dilakukan *resizing* agar citra yang didapatkan memiliki ukuran yang sama dan menjadi ukuran yang diperlukan. Kemudian proses *augmentation* dengan menambah *mosaic* untuk mengajarkan model untuk mengenali objek pada lokalisasi yang berbeda tanpa terlalu bergantung pada satu konteks citra tertentu. Lalu proses *labelling* untuk menandai bagian pada citra yang akan dideteksi. Selanjutnya dilakukan identifikasi menggunakan YOLOv5 untuk mendeteksi APD, dimana metode ini digunakan untuk menampilkan *bounding box* dari objek yang dideteksi dengan keterangan jenisnya dan *confidence level* dari objek tersebut dan dihasilkan *learned model*, maka akan dihasilkan *output* dari semua proses diatas yaitu *class* pada *model*.



Gambar 3.1 Arsitektur umum



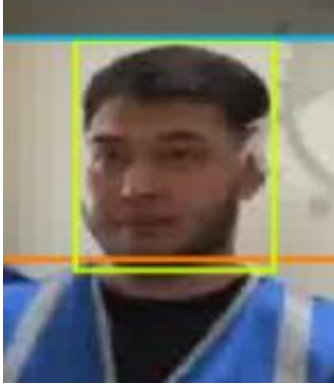

3.2 Image acquisition


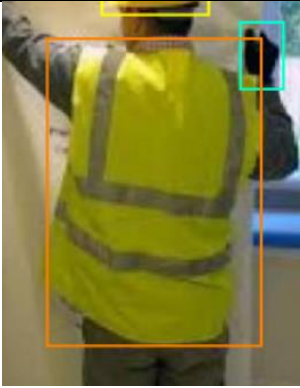

Image acquisition meliputi serangkaian tahapan proses pengumpulan dan validasi data yang diperlukan dalam aplikasi ini. Data yang digunakan berupa gambar (citra) dimana akan dibagi berdasarkan kelasnya. Data yang digunakan berupa gambar pekerja (manusia) proyek menggunakan APD, dan tidak menggunakan APD lalu di label masing-masing ke kelas yang berbeda. Sebagian *dataset* didapat dari *roboflow*, dan sebagian lagi diambil menggunakan *smartphone* dengan kamera 64MP dengan ratio 16:9 dan diambil menggunakan tangan penulis sendiri. Pembagian data dari penelitian ini dapat dilihat pada Table 3.1 dan table 3.2

Tabel 3.1 Pembagian data yang didapat

| No | Objek | Jumlah |
|----|--------------|-------------|
| 1 | Gloves | 1040 |
| 2 | Helmet | 1112 |
| 3 | Non-helmet | 219 |
| 4 | Person | 1280 |
| 5 | Shoes | 913 |
| 6 | Vest | 1322 |
| 7 | Bare-arms | 130 |
| | Total | 6016 |

Tabel 3.2 Contoh data yang didapat

| No | Objek | Class |
|----|---|------------|
| 1 |  | Gloves |
| 2 |  | Helmet |
| 3 |  | Non-helmet |
| 4 |  | Person |

| No | Objek | Class |
|----|---|-----------|
| 5 |  | Shoes |
| 6 |  | Vest |
| 7 |  | Bare-arms |

3.3 *Pre-processing*

Seluruh data citra yang telah diambil akan dilanjutkan ke tahapan *pre-processing* dimana pada penelitian ini digunakan dua tahapan yaitu *resizing*, dan *image labelling*.

3.3.1 *Resizing*

Resize pada citra merupakan tahap penskalaan yaitu merubah ukuran citra menjadi lebih kecil. *Resizing* dilakukan agar proses *training* untuk membuat model dapat menjadi lebih cepat. Disaat kita tidak bisa mengatur ukuran data yang akan digunakan, maka *resize* dilakukan untuk menyesuaikan ukuran dari citra. *Resize* dilakukan di *roboflow*, dan gambar akan *dirresize* menjadi ukuran 416 x 416.

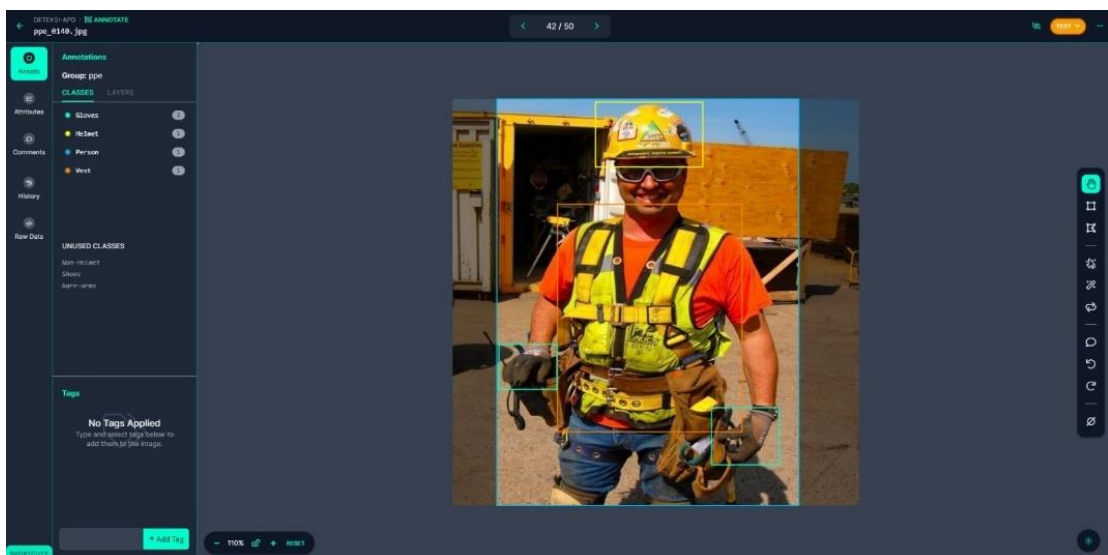
3.3.2 Labelling

Tahap *labeling* dilakukan untuk memberikan *label* nama pada gambar yang telah di *resize* sebelumnya untuk pengujian sistem untuk mendeteksi APD. Pada penelitian ini *labelling* menggunakan aplikasi *roboflow* yang digunakan untuk penamaan label untuk citra dengan setiap komponen APD. Hasil dari labelling akan disimpan dalam bentuk *file.txt*

```
3 0.13341346153846154 0.2247596153846154 0.2668269230769231 0.4495192307692308
5 0.7872596153846154 0.2247596153846154 0.4254807692307692 0.4495192307692308
5 0.028846153846153848 0.7007211538461539 0.057692307692307696 0.15144230769230768
0 0.06850961538461539 0.5192307692307693 0.05889423076923077 0.10576923076923077
3 0.07692307692307693 0.7247596153846154 0.15384615384615385 0.5504807692307693
1 0.9050480769230769 0.5252403846153846 0.1045673076923077 0.12139423076923077
1 0.6742788461538461 0.5973557692307693 0.11057692307692307 0.23557692307692307
3 0.9338942307692307 0.7247596153846154 0.13221153846153846 0.5504807692307693
```

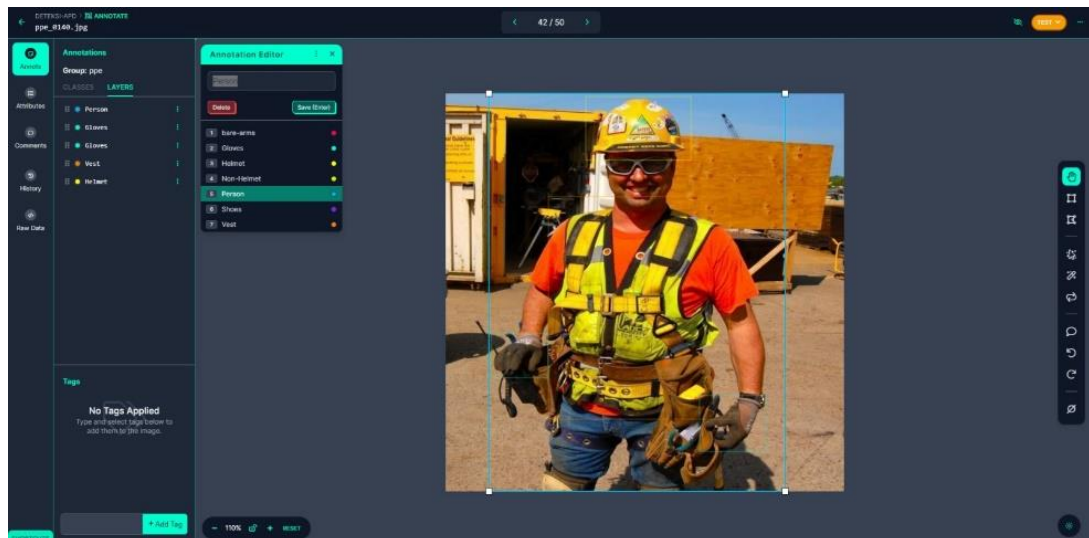
Gambar 3.2 Isi file.txt

Proses *labelling* dilakukan untuk menjadi pelajaran bagi sistem untuk mengenali objek yang akan dideteksi. *Labelling* dilakukan menggunakan roboflow seperti yang ditampilkan pada Gambar 3.3, dan Gambar 3.4 dimana kita membuat *bounding box* menggunakan *tool* yang telah disediakan *roboflow*.



Gambar 3.3 Proses membuat *Bounding Box*

Lalu ditentukan label pada *class* yang telah dibuat seperti pada gambar 3.4, dan menentukan *class* untuk *bounding box* yang dibuat.



Gambar 3.4 Menentukan *Class*

3.4 Augmentasi

Setelah melakukan *labelling*, akan dilanjutkan ke tahapan augmentasi untuk menambah data yang berfungsi untuk meningkatkan nilai akurasi. Augmentasi juga dilakukan di *roboflow*, dan menggunakan 1 augmentasi yaitu *mosaic*. Augmentasi *pseudo-code* seperti berikut:

```
def augment_data(image):

    image <--- random_flip_horizontal(image)
    image <--- random_flip_vertical(image)
    image <--- normalize_image(image)

    return image

def random_flip_horizontal(image):

    return flipped_image

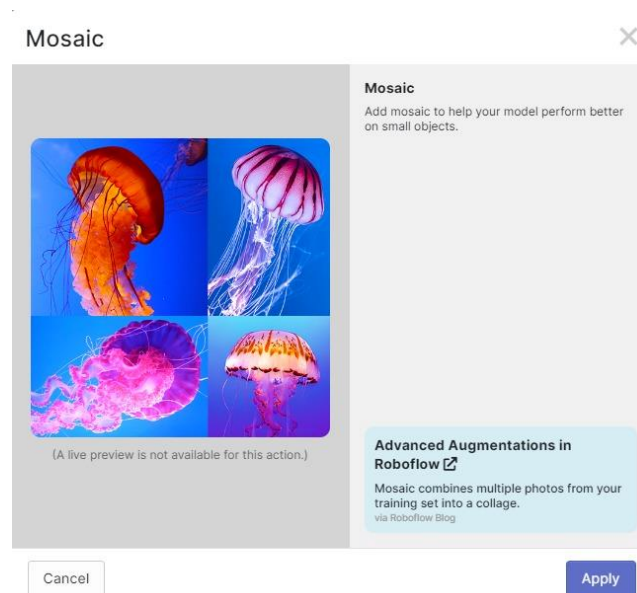
def random_flip_vertical(image):

    return flipped_image
```

```
def normalize_image(image):
    return normalized_image

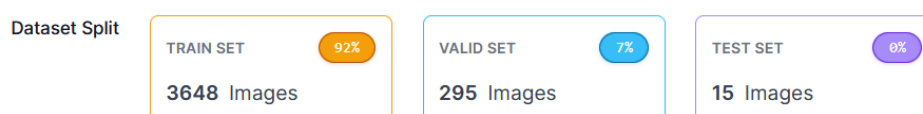
image <--- load_image(image_path)
augmented_image <--- augment_data(image)
```

Pada tahapan augmentasi ini, *mosaic* akan mengajarkan model untuk mengenali objek di lokasi yang berbeda tanpa bergantung pada satu konteks tertentu. Akan diambil 4 citra dan *resize* menjadi ukuran yang sama dan menyatukannya menjadi proporsi yang acak.



Gambar 3.5 Proses Mosaic

Setelah melakukan augmentasi, maka data akan bertambah banyak 2x lipat dari data yang sudah ada sebelumnya. Data yang bertambah banyak ini akan dipecah menjadi data *training*, *validation* dan data *testing*.



Gambar 3.6 Pembagian data

3.5 *Image Classification*

Setelah melalui tahap *pre-processing*, citra akan di klasifikasi menggunakan metode YOLOv5. Metode ini mampu mengidentifikasi objek dengan *bounding box* lalu mengklasifikasikan nya berdasarkan kelas yang sudah dibuat. Pendeteksian objek dengan YOLOv5 dapat digambarkan sebagai berikut:

| | |
|----------------------------------|--|
| <i>Input</i> | : YOLOv5 mengambil input data berupa gambar. |
| <i>Backbone</i> | :Data input akan melalui jaringan saraf konvolusi (CNN), kemudian input gambar akan dipetakan sekumpulan fiturnya. |
| <i>Neck</i> | :Sekumpulan fitur yang dipetakan sebelumnya akan disempurnakan dan digabungkan pada <i>neck</i> untuk mengambil lebih banyak konteks data dan Informasi semantik. |
| <i>Head</i> | :Sekumpulan fitur yang telah disempurnakan akan memprediksi objek dengan kotak pembatas dan kemungkinan kelas pada serangkaian kotak <i>anchor</i> dengan skala dan rasio yang berbeda. |
| Deteksi Objek | :Merupakan hasil akhir dari metode ini, yang akan menampilkan kotak pembatas hasil prediksi sebelumnya dan kemungkinan kelas untuk objek yang terdapat dalam <i>input</i> gambar. |
| <i>Loss Function</i> | :Meminimalisir <i>minimize loss function</i> . Akan dilakukan beberapa iterasi untuk menghindari <i>bounding box</i> yang bertimpahan dan akan menghasilkan deteksi yang akurat. |
| <i>Non maximum Suppression</i> : | Merupakan tahap akhir dari mendeteksi citra yang mana algoritma akan menghapus <i>bounding box</i> dengan skor terendah dan hanya akan menampilkan <i>bounding box</i> dengan skor tertinggi |



Gambar 3.7 Sebelum *Non Maximum Suppression*



Gambar 3.8 Sesudah *Non Max Suppression*

Learned Model :Merupakan hasil dari proses training dan berekstensi .pt menggunakan library PyTorch. Pada proses *training* akan terbentuk model terbaik (*best*) dan model terakhir (*last*), dan akan digunakan model terbaik (*best*).

3.6 YOLOv5 Model

Sebelum melakukan training, perlu diperhatikan beberapa parameter terlebih dahulu yaitu menentukan *weight* yang akan digunakan, melakukan konfigurasi model YOLOv5 dan parameter lainnya

3.6.1 Clone repository

Pada tahapan ini, akan dilakukan clone repository metode YOLOv5 dari *github*. Untuk melakukan *clone repository* YOLOv5 terdapat pada gambar 3.9.

```
[ ] # clone YOLOv5 repository
!git clone https://github.com/ultralytics/yolov5 # clone repo
%cd /content/drive/MyDrive/yolov5_ta/yolov5
!git reset --hard 064365d8683fd002e9ad789c1e91fa3d021b44f0
```

Gambar 3.9 Clone repository

3.6.2 Install Requirement

Pada tahapan ini, akan diinstall beberapa *packages* yang diperlukan untuk menggunakan metode YOLOv5. Pada *github* terdapat *requirements.txt* yang berisi *packages* yang akan diinstall nantiya.

3.6.3 Menambahkan *data.names* dan *data.yaml*

Pada *data.names* akan berisi nama kelas yang digunakan dalam deteksi, dan *data.yaml* berisi path folder untuk *training*, *validation*, nama kelas yang dibuat, dan jumlah kelas yang dibuat.

3.6.4 Konfigurasi YOLOv5

Konfigurasi YOLOv5 perlu diperhatikan jumlah kelas yang digunakan (*classes*) dan *setting filter* dalam YOLOv5. Jumlah kelas memperlihatkan berapa kelas yang digunakan, dan setting filter harus disesuaikan dengan nilai kelas.

3.7 Proses Training

Sebanyak 92% dari total data yaitu 3648 data akan melalui *training*, dan akan ditentukan *batchsize*, *epoch*, dan *size* hingga akhirnya akan dihasilkan *learned* model dari konfigurasi diatas. Jumlah data *training* harus lebih banyak dari data *testing* agar algoritma pembelajaran menjadi lebih kompleks dan pemetaan pada gambar menjadi beragam. Dapat dilihat *pseudo-codenya* sebagai berikut.

```

DEFINE dataset_location
DEFINE model_config_path
DEFINE weights_path

FUNCTION train_model():
    SET img_size <--- 416
    SET batch_size <--- 64
    SET num_epochs <--- 250

    SET data_path <--- dataset_location + "/data.yaml"
    SET config_path <--- "./models/custom_yolov5s.yaml"
    SET weights <--- '

```

```

SET model_name <--- "yolov5s_results"
SET cache_enabled <--- True

EXECUTE train.py script with:
    image <--- img_size
    batch size <--- batch_size
    epochs <--- num_epochs

    data <--- path_data
    cfg <--- config_path
    weights <--- weights

    name <--- model_name
    cache <--- cache_enabled

CALL train_model() function

```

3.8 Proses Testing

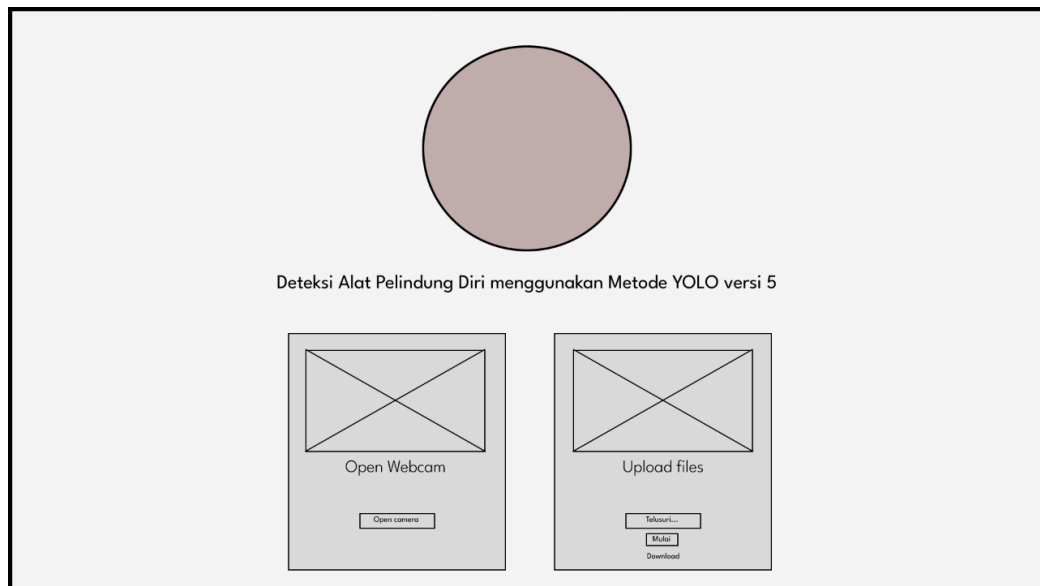
Setelah melakukan *training*, akan dihasilkan *learned model* dan akan dilakukan uji coba pada orang yang menggunakan APD untuk mengetahui apakah mampu mendeteksi komponen APD lalu akan dihitung akurasi yang didapatkan dari hasil *testing* dengan metode YOLOv5 ini agar dapat diketahui kinerja dari model yang dihasilkan.

3.9 Perancangan Antarmuka Sistem

Merancang antarmuka sistem pendeteksian APD ini berguna untuk memudahkan penggunaan sistem yang telah dibuat.

3.9.1 Rancangan Tampilan Utama

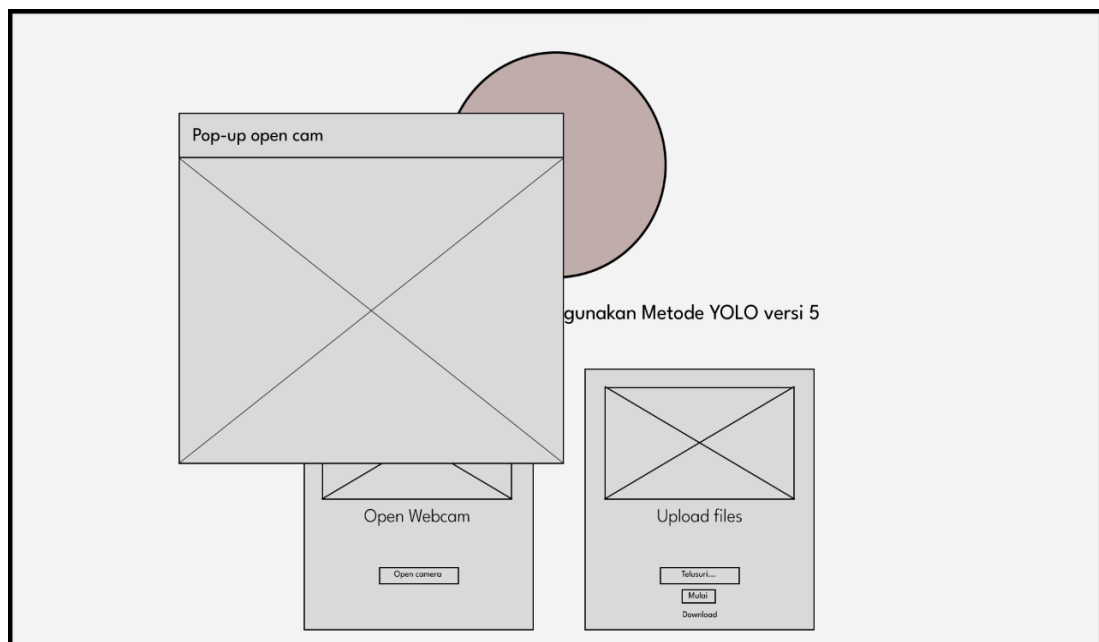
Pada halaman utama, user akan dihadapkan pada 2 opsi yaitu menggunakan sistem pendeteksian secara *real-time* yaitu dengan membuka kamera (*Open cam*), atau dengan mengupload file. Untuk tampilan utama dapat dilihat pada gambar 3.18.



Gambar 3.10 Halaman Utama

3.9.2 Rancangan Tampilan *Open camera*

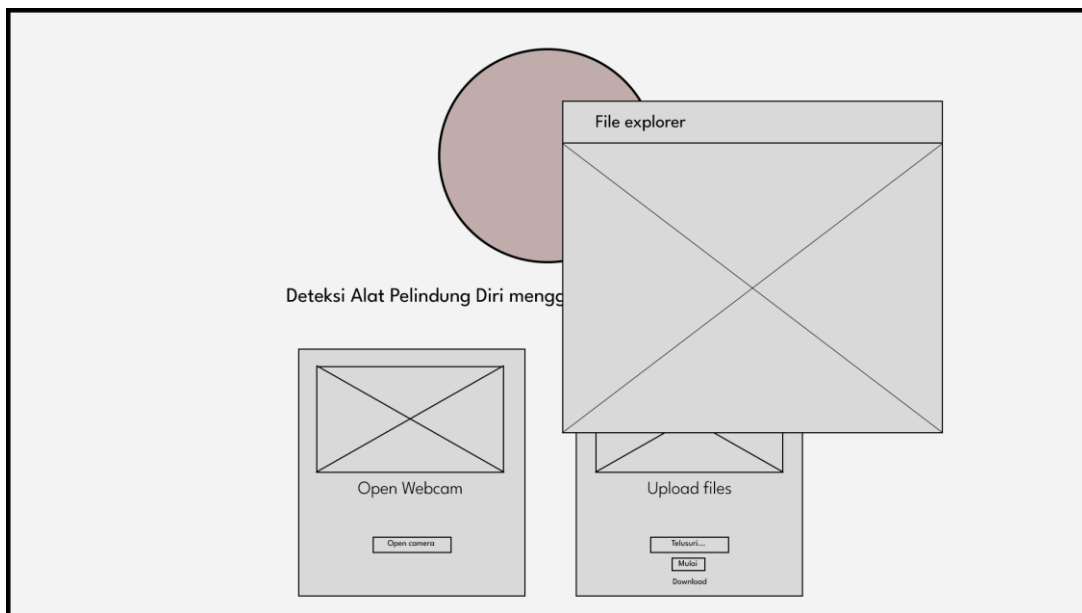
Pada bagian ini, akan ditampilkan kotak pop up yang merupakan tempat untuk menampilkan yang direkam camera input. Jendela *pop up* ini muncul setelah pilihan “*Open Camera*” ditekan pada halaman utama. Dapat dilihat pada gambar 3.19.



Gambar 3.11 Jendela *Pop up open webcam*

3.9.3 Rancangan Tampilan *upload files*

Pada bagian ini, akan ditampilkan jendela *pop up file explorer* yang merupakan tempat untuk memilih *file* dengan ekstensi .jpg untuk gambar atau .mp4 untuk video . *Pop up* ini muncul setelah pilihan “Telusuri...” pada halaman utama ditekan. Setelah *file* dipilih, lalu tekan tombol “Mulai” untuk melakukan deteksi. Dan setelah aplikasi selesai melakukan deteksi, *file* baru dapat diunduh dengan menekan pilihan “Download”, lalu *file* akan diunduh ke penyimpanan lokal. Dapat dilihat pada gambar 3.20.



Gambar 3.12 Halaman *Upload Files*

BAB 4

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dibahas proses implementasi deteksi APD pada manusia menggunakan metode YOLOv5 dari rancangan yang sudah dipaparkan dan hasil dari pengujian sistem.

4.1 Implementasi Sistem

Metode *You Only Look Once* (YOLO) akan diimplementasikan untuk mengkalsifikasikan komponen APD pada manusia di tahapan ini. Maka, diperlukan perangkat keras dan perangkat lunak yang mendukung yaitu :

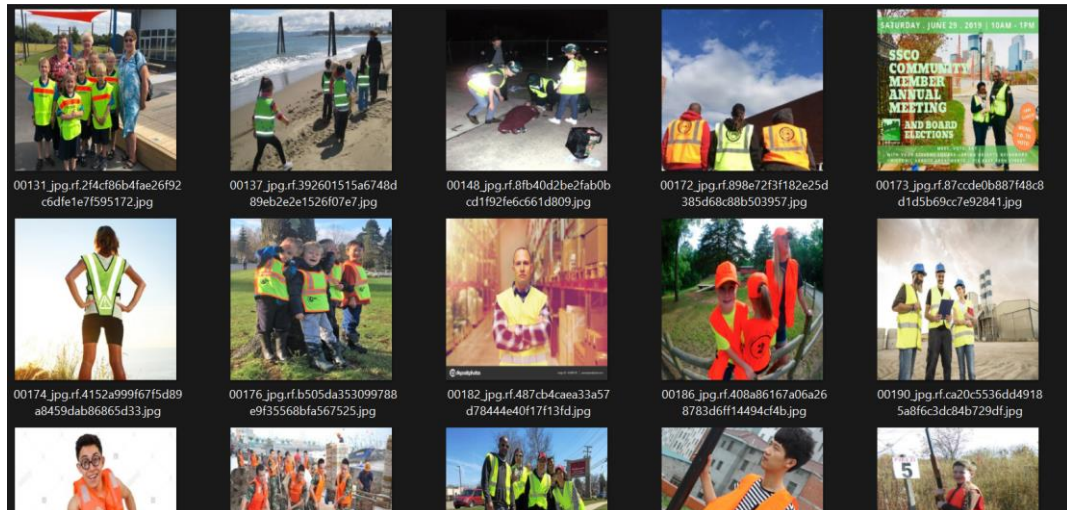
4.1.1 Perangkat lunak dan perangkat keras

Untuk sistem pendektasian, spesifikasi dari perangkat keras dan perangkat lunak sebagai berikut :

1. Processor AMD Ryzen 5 3500U CPU 2.1 GHz
2. RAM berkapasitas 8 GB
3. SSD dengan kapasitas 500 GB
4. OS Windows 11 Home Single Language
5. Google Colab GPU Nvidia Cuda V.11.2
6. *Visual Studio Code*
7. *Google Colab*
8. *Roboflow (Labelling)*

4.2 Implementasi Data

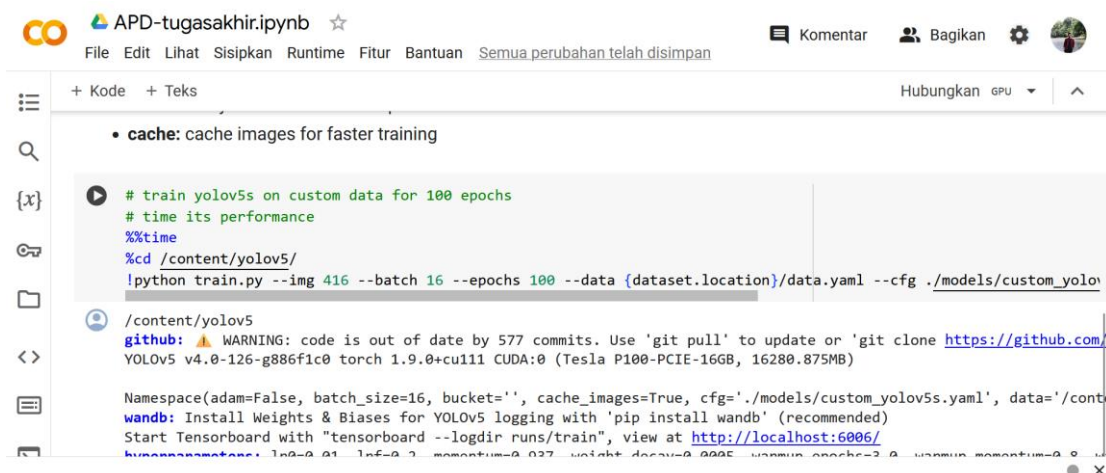
Data gambar yang akan dipakai adalah gambar setiap komponen APD, yang terdiri dari *Gloves, Helmet, Shoes, Vest, Non-helmet, Bare-arms* dan *Person* yang diambil menggunakan kamera smartphone dan dataset dari roboflow yang totalnya 1526. Penulis mengambil data uji pada lingkungan proyek perpustakaan USU, dan panglong menggunakan *webcam* dan *Closed Circuit Television (CCTV)*. Contoh dari gambar citra yang nantinya dipakai seperti pada gambar 4.1.



Gambar 4.1 Gambar pada dataset

4.3 Pelatihan Sistem

Sesudah tahap *pre-processing*, *training* data dilakukan agar model mampu mempelajari ciri dari setiap objek menggunakan metode YOLOv5. Proses *training* menggunakan platform *Google Colab* pada gambar 4.2. *Google Colab* menyediakan *framework* *pytorch* dan *runtime* GPU NVIDIA yang sudah mendukung CUDA secara gratis dengan runtime 12 jam per harinya dan dapat diupgrade menjadi *Colab pro*. *Output* model akan berekstensi .py. Untuk proses ini akan menggunakan repository YOLOv5 yang tersedia dan dapat di clone melalui *Github ultralytics*. Ukuran gambar yang digunakan adalah 416 dengan weight pretrained YOLOv5s.



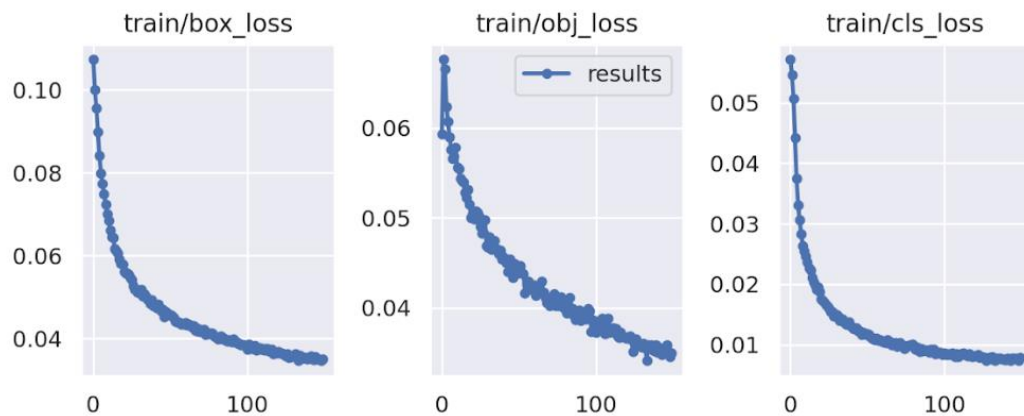
Gambar 4.2 Google Colab

Pada langkah *training* dilakukan konfigurasi pada *epoch* dan *batch size* untuk menemukan akurasi yang paling baik, hasil dari percobaan terdapat pada tabel 4.1.

Tabel 4.1 Langkah *Training*

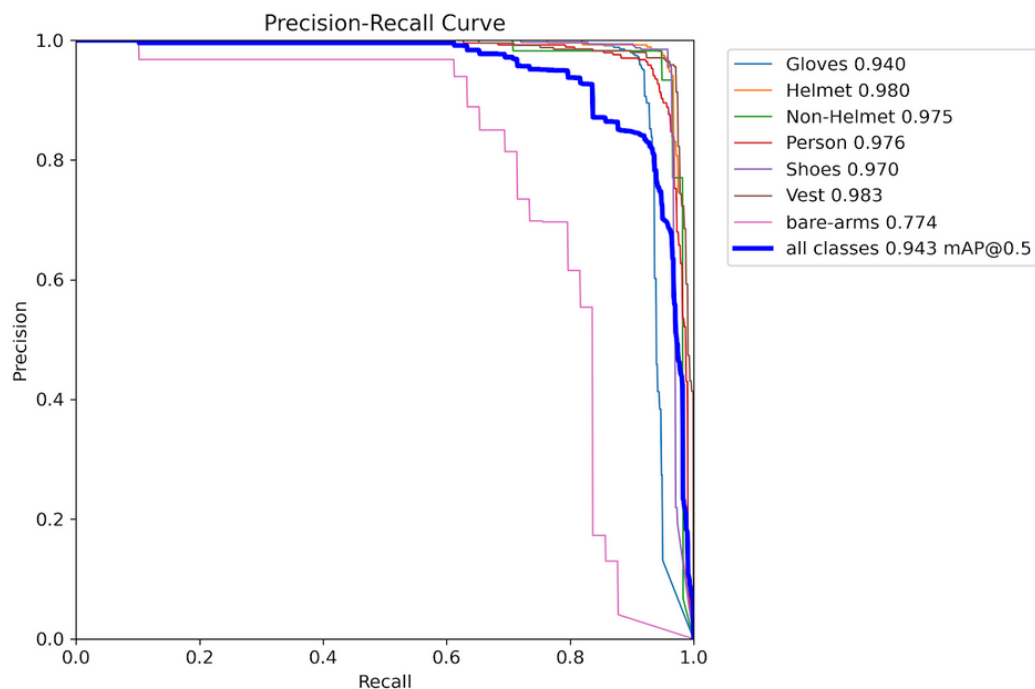
| Epoch | Batch Size | mAP | Precision | Recall | Box_loss |
|------------|------------|--------------|--------------|--------------|----------------|
| 100 | 8 | 0.92 | 0.938 | 0.879 | 0.03392 |
| 150 | 8 | 0.936 | 0.952 | 0.905 | 0.03148 |
| 200 | 8 | 0.941 | 0.955 | 0.901 | 0.02978 |
| 250 | 8 | 0.946 | 0.948 | 0.912 | 0.02787 |
| 100 | 16 | 0.913 | 0.909 | 0.875 | 0.03362 |
| 150 | 16 | 0.933 | 0.943 | 0.905 | 0.03111 |
| 200 | 16 | 0.945 | 0.957 | 0.9 | 0.02964 |
| 250 | 16 | 0.95 | 0.947 | 0.912 | 0.02771 |
| 100 | 32 | 0.92 | 0.927 | 0.885 | 0.03354 |
| 150 | 32 | 0.929 | 0.945 | 0.89 | 0.03111 |
| 200 | 32 | 0.939 | 0.966 | 0.903 | 0.02932 |
| 250 | 32 | 0.945 | 0.953 | 0.915 | 0.2766 |
| 100 | 64 | 0.913 | 0.934 | 0.882 | 0.03327 |
| 150 | 64 | 0.935 | 0.957 | 0.897 | 0.03125 |
| 200 | 64 | 0.945 | 0.956 | 0.906 | 0.02951 |
| 250 | 64 | 0.948 | 0.959 | 0.915 | 0.02743 |
| 100 | 128 | 0.917 | 0.943 | 0.876 | 0.03349 |
| 150 | 128 | 0.929 | 0.947 | 0.899 | 0.31101 |
| 200 | 128 | 0.937 | 0.963 | 0.9 | 0.29102 |
| 250 | 128 | 0.942 | 0.974 | 0.91 | 0.26607 |

Hasil yang ditunjukkan pada Tabel 4.1 dapat diambil kesimpulan yaitu dengan epoch 250 dan batch size 64 menghasilkan model dengan performa paling baik. Dapat dilihat juga performa dari model dengan grafik *loss* pada gambar 4.3, yaitu grafik *object loss*, *objectness loss*, dan *class loss*. Yang mana *object loss* menampilkan sebagaimana bagusnya algoritma mampu melokasikan titik tengah objek dan performa dari *bounding box* nya. *Objectness* yaitu nilai kemungkinan suatu objek pada ROI (*Region Of Interest*) yang didapat. *Classification loss* menampilkan performa algoritma dalam melakukan prediksi *class* dari objek yang diberikan.



Gambar 4.3 Grafik *Loss*

Untuk *precision Recall curve* ditampilkan pada gambar 4.4, yang menunjukkan tarik ulur antara *precision* dan *recall*. Hasil model pada proses training akan tersimpan sebagai dua output yaitu *best.pt* dan *last.pt*.



Gambar 4.4 *Precision-Recall Curve*

4.4 Prosedur Operasional

Interface pada aplikasi, menampilkan halaman menu untuk mendeteksi Alat Pelindung Diri (APD) secara *realtime* menggunakan *camera input*, melakukan deteksi dengan file berbentuk gambar atau video dengan model yang telah dibuat.

Untuk menjalankan aplikasi, *user* memakai basis web dengan flask, yang akan diunduh kedalam file user dan menjalankan perintah “flask run” di terminal *VSCode*.

```
(env) PS C:\Users\lazua\666\SKRIPSI\faishal\YOLOv5-Flask-master3> flask run
```

Gambar 4.5 Flask Run

```
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [17/Nov/2023 23:47:05] "GET / HTTP/1.1" 200 -
```

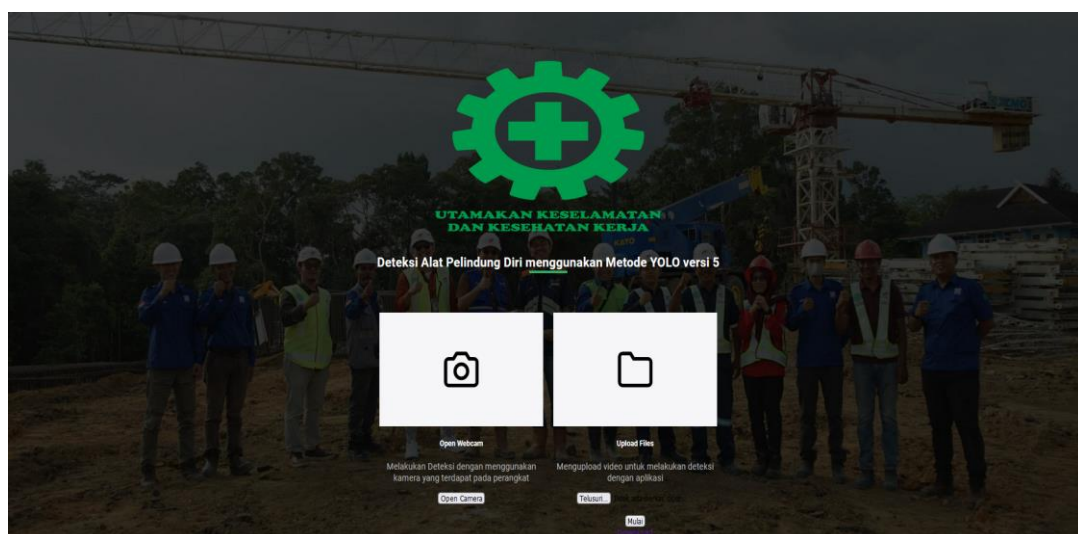
Gambar 4.5 Link dari flask

Alamat yang didapat dari terminal yaitu 127.0.0.1:5000 sesuai dengan gambar 4.5, *user* dapat menggunakan aplikasi dengan membuka alamat tersebut pada *browser*.

4.5 Implementasi Perancangan Sistem

4.5.1 Tampilan Halaman Utama

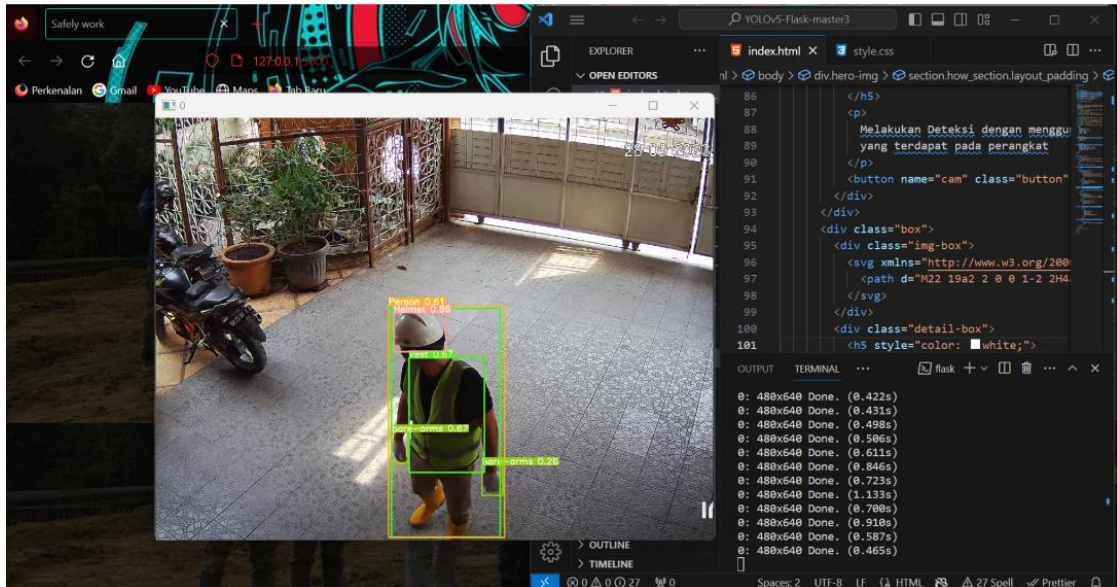
Disajikan Logo Kesehatan dan Keselamatan Kerja (K3) dan terdapat judul “Deteksi Alat Pelindung Diri menggunakan YOLO versi 5”. Pada Tampilan ini terdapat 2 menu yang disajikan yaitu “*Open Webcam*” dan “*Upload Files*”. Pada opsi “*Open Webcam*” terdapat ikon kamera beserta deskripsi dibawahnya tentang deteksi secara *real-time*, dan terdapat tombol *Open Camera*. Pada opsi “*Upload Files*” terdapat ikon folder beserta deskripsi dibawahnya tentang deteksi secara *non real-time*, dan terdapat tombol “Telusuri...”, tombol “Mulai” dan opsi download dibawahnya. Dapat dilihat pada gambar 4.6



Gambar 4.6 Tampilan Utama

4.5.2 Tampilan Mendeteksi dengan *webcam*

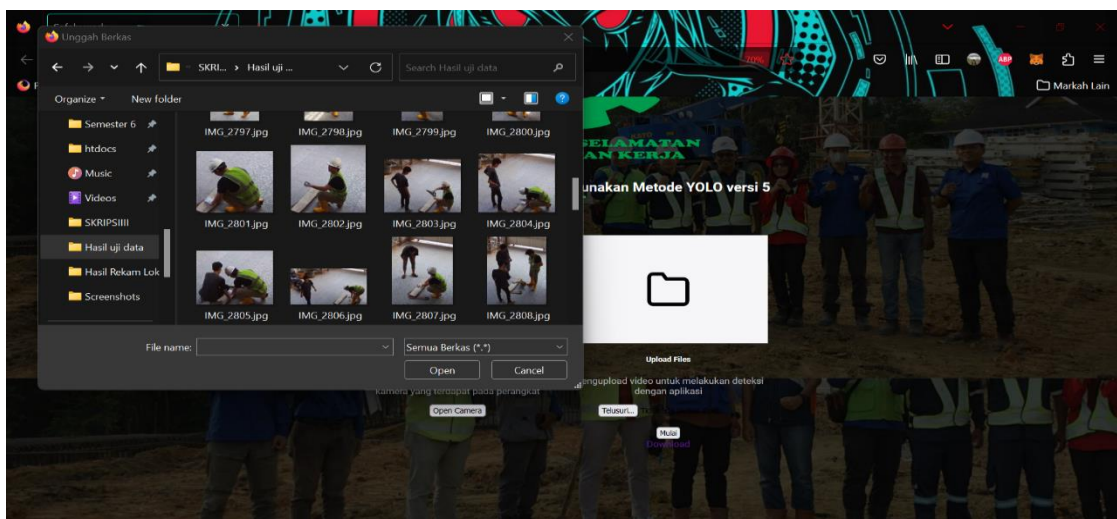
Setelah menekan tombol “*Open Camera*” pada halaman utama, jendela *pop up* akan ditampilkan setelah beberapa saat untuk menampilkan *webcam* yang terdapat pada perangkat user. Jendela *pop up* menampilkan video realtime hasil pendeteksian dari *webcam user* yang dapat dilihat pada gambar 4.7



Gambar 4.7 Jendela Pop-up webcam

4.5.3 Tampilan Mendeteksi dengan *Upload file*

Setelah menekan tombol “Telusuri...” pada halaman utama, jendela *pop up* akan ditampilkan untuk menampilkan direktori file yang terdapat pada perangkat user. Pengguna diminta untuk memilih file yang akan dilakukan deteksi dengan aplikasi, baik itu gambar ataupun video. Dapat dilihat pada gambar 4.8


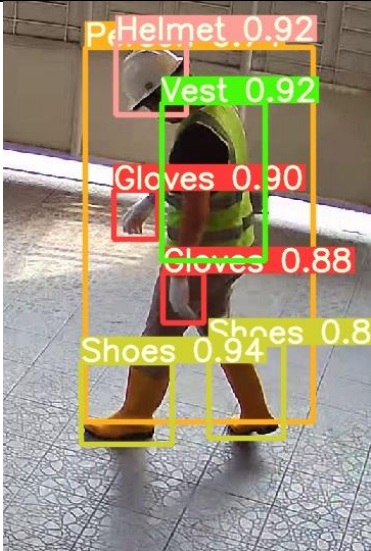


Gambar 4.8 Tampilan Upload file



4.6 Pengujian Sistem



Pada tahapan ini akan ditampilkan hasil dari pengujian dengan model yang telah dihasilkan dari proses pelatihan sebelumnya untuk mendapatkan kemampuan model dalam melakukan deteksi Alat Pelindung Diri (APD). Pada tahap uji digunakan kamera *Closed Circuit Television* (CCTV) dengan spesifikasi 1080p dan *Webcam* dengan spesifikasi 1080p sebagai input pada aplikasi.




Tabel 4.2 Pengujian Sistem




| No | Gambar | Aktual | Hasil Deteksi | Status |
|----|---|--|--|--|
| 1. |  | Person Helmet Vest Gloves Gloves Shoes Shoes | Person Helmet Vest Gloves Gloves Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil |
| 2. |  | Person Helmet Vest Gloves Gloves Shoes Shoes | Person Helmet Vest Gloves Gloves Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil |




[illegible]




| No | Gambar | Aktual | Hasil Deteksi | Status |
|----|--|--|---|--|
| | | Shoes | Shoes | Berhasil |
| 6. |  | Person Person Helmet Non-Helmet Vest Gloves Gloves Bare-arms Shoes Shoes Shoes Shoes | Person Person Helmet Non-Helmet Vest Gloves Tidak terdeteksi Bare-arms Shoes Shoes Shoes Tidak terdeteksi | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Gagal Berhasil Berhasil Berhasil Gagal |
| 7. |  | Person Person Helmet Non-Helmet Vest Gloves Gloves Bare-arms Bare-arms Shoes Shoes Shoes Shoes | Person Person Helmet Non-Helmet Vest Gloves Tidak terdeteksi Bare-arms Bare-arms Shoes Tidak terdeteksi Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Gagal Berhasil Berhasil Gagal Berhasil Berhasil |

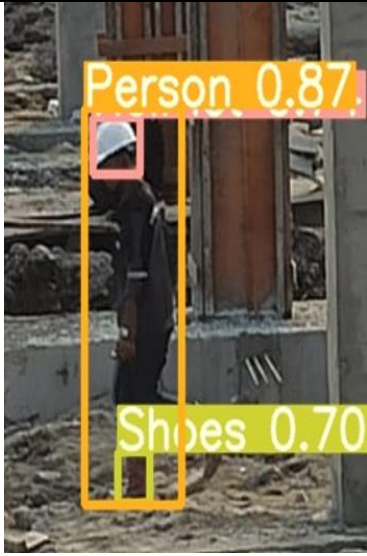


| No | Gambar | Aktual | Hasil Deteksi | Status |
|----|---|---|--|--|
| 8. |  | Person Person Person Non-Helmet Non-Helmet Non-Helmet Vest Gloves Gloves Bare-arms Bare-arms Bare-arms Shoes Shoes Shoes Shoes | Person Person Person Non-Helmet Non-Helmet Non-Helmet Vest Gloves Gloves Bare-arms Bare-arms Tidak terdeteksi Shoes Shoes Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil |
| 9. |  | Person Helmet Vest Gloves Gloves Shoes Shoes | Person Helmet Vest Gloves Tidak terdeteksi Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Gagal Berhasil Berhasil |




| No | Gambar | Aktual | Hasil Deteksi | Status |
|-----|---|--|---|---|
| 10. |  | Person Helmet Vest Gloves | Person Helmet Vest Gloves | Berhasil Berhasil Berhasil Berhasil |
| 11. |  | Person Helmet Vest Gloves Gloves | Person Helmet Vest Gloves Tidak terdeteksi | Berhasil Berhasil Berhasil Berhasil Gagal |
| 12. |  | Person Non-Helmet Vest Gloves Gloves Shoes Shoes | Person Non-Helmet Vest Gloves Gloves Shoes Tidak terdeteksi | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Gagal |



| No | Gambar | Aktual | Hasil Deteksi | Status |
|-----|---|---|---|--|
| 13. |  | Person Non-Helmet Vest Gloves Gloves Shoes | Person Non-Helmet Vest Gloves Gloves Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil |
| 14. |  | Person Non-Helmet Vest Gloves Gloves Shoes | Person Non-Helmet Vest Gloves Gloves Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil |
| 15. |  | Person Non-Helmet Vest Gloves Gloves Shoes | Person Non-Helmet Vest Gloves Tidak terdeteksi Shoes | Berhasil Berhasil Berhasil Berhasil Gagal Berhasil |

| No | Gambar | Aktual | Hasil Deteksi | Status |
|-----|---|--|--|---|
| 16. |  | Person Helmet Vest Gloves Shoes | Person Helmet Vest Gloves Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil |
| 17. |  | Person Helmet Vest Shoes | Person Helmet Vest Tidak terdeteksi | Berhasil Berhasil Berhasil Gagal |
| 18. |  | Person Helmet Vest Gloves Gloves Shoes Shoes | Person Helmet Vest Gloves Tidak terdeteksi Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Gagal Berhasil Berhasil |

| No | Gambar | Aktual | Hasil Deteksi | Status |
|-----|---|---|--|--|
| 19. |  | Person Helmet Vest Gloves Gloves Shoes | Person Helmet Vest Tidak terdeteksi Tidak terdeteksi Tidak terdeteksi | Berhasil Berhasil Berhasil Gagal Gagal Gagal |
| 20. |  | Person Person Helmet Helmet Vest Vest Bare-arms Gloves Shoes Shoes | Person Person Helmet Helmet Vest Vest Tidak terdeteksi Tidak terdeteksi Tidak terdeteksi Tidak terdeteksi | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Gagal Gagal Gagal Gagal |
| 21. |  | Person Person Helmet Helmet Vest Vest | Person Person Helmet Helmet Vest Tidak terdeteksi | Berhasil Berhasil Berhasil Berhasil Berhasil Gagal |

| No | Gambar | Aktual | Hasil Deteksi | Status |
|-----|---|--|--|--|
| 22. |  | Person Helmet Bare-arms Shoes Shoes | Person Helmet Tidak terdeteksi Shoes Tidak terdeteksi | Berhasil Berhasil Gagal Berhasil Gagal |
| 23. |  | Person Person Helmet Helmet Vest Vest Bare-arms Bare-arms Shoes Shoes Shoes Shoes | Person Person Helmet Helmet Vest Vest Tidak terdeteksi Tidak terdeteksi Shoes Shoes Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Gagal Gagal Berhasil Berhasil Berhasil Berhasil |
| 24. |  | Person Person Person Helmet Helmet Helmet Bare-arms Bare-arms Shoes Shoes | Person Person Person Helmet Helmet Helmet Tidak terdeteksi Tidak terdeteksi Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Gagal Gagal Berhasil Berhasil |

| No | Gambar | Aktual | Hasil Deteksi | Status |
|-----|---|---|--|--|
| | | Shoes | Shoes | Berhasil |
| 25. |  | Person Person Person Helmet Helmet Helmet Vest Vest Bare-arms Bare-arms Bare-arms Shoes Shoes Shoes Shoes | Person Person Person Helmet Helmet Tidak terdeteksi Vest Vest Tidak terdeteksi Tidak terdeteksi Tidak terdeteksi Shoes Shoes Shoes Shoes | Berhasil Berhasil Berhasil Berhasil Berhasil Gagal Berhasil Berhasil Gagal Gagal Gagal Berhasil Berhasil Berhasil Berhasil |
| 26. |  | Person Person Helmet Helmet Vest Vest Bare-arms Shoes | Person Person Helmet Helmet Vest Vest Tidak terdeteksi Tidak terdeteksi | Berhasil Berhasil Berhasil Berhasil Berhasil Berhasil Gagal Gagal |
| 27. |  | Person Person Helmet Helmet Vest Vest | Person Person Helmet Tidak terdeteksi Vest Vest | Berhasil Berhasil Berhasil Gagal Berhasil Berhasil |

| No | Gambar | Aktual | Hasil Deteksi | Status |
|-----|--|--|---|---|
| 28. |  | Person Helmet Vest Gloves Gloves Shoes Shoes | Person Helmet Vest Bare-arms Bare-arms Shoes Tidak terdeteksi | Berhasil Berhasil Berhasil Gagal Gagal Berhasil Gagal |
| 29. |  | Person Helmet Vest Gloves Gloves Shoes | Person Gloves Vest Gloves Tidak terdeteksi Shoes | Berhasil Gagal Berhasil Berhasil Gagal Berhasil |

Setelah dilakukan pengujian untuk penelitian mendeteksi Alat Pelindung Diri (APD) dengan metode *You Only Look Once Versi 5* (YOLOv5) terdapat beberapa kesalahan, yaitu kesalahan tidak terdeteksi salah satu *class*. Faktor terbesar dalam kesalahan tidak terdeteksi, yaitu dikarenakan kualitas kamera *Closed Circuit Television* (CCTV) dan juga angle camera yang tidak maksimal untuk memperlihatkan *class* pada *object* secara keseluruhan. Pada sarung tangan dan *bare-arms* juga ditemukan kesulitan deteksi dikarenakan bentuk yang sama, maka model akan mendeteksi kedua kelas tersebut berdasarkan warna saja. Dapat dilihat *Confusion matrix* pada tabel 4.3.

Tabel 4.3 *Confusion matrix*

| | | Aktual | | | | | | | Total |
|----------|------------------|--------|--------|------------|------|--------|-----------|-------|-------|
| | | Person | Helmet | Non-Helmet | Vest | Gloves | Bare-arms | Shoes | |
| Prediksi | Person | 43 | 0 | 0 | 0 | 0 | 0 | 0 | 43 |
| | Helmet | 0 | 30 | 0 | 0 | 0 | 0 | 0 | 30 |
| | Non-Helmet | 0 | 0 | 10 | 0 | 0 | 0 | 0 | 10 |
| | Vest | 0 | 0 | 0 | 32 | 0 | 0 | 0 | 32 |
| | Gloves | 0 | 1 | 0 | 0 | 25 | 0 | 0 | 25 |
| | Bare-arms | 0 | 0 | 0 | 0 | 2 | 14 | 0 | 16 |
| | Shoes | 0 | 0 | 0 | 0 | 0 | 0 | 51 | 51 |
| | Tidak terdeteksi | 0 | 2 | 0 | 1 | 8 | 7 | 7 | 25 |
| Total | | 43 | 33 | 10 | 33 | 38 | 25 | 61 | 227 |

Pada tahap pengujian juga terdapat kesalahan pada *output* yang dihasilkan. Untuk penelitian ini kekurangan sistem untuk mendeteksi objek dapat dilihat pada tabel 4.4.

Tabel 4.4 Tabel Hasil Pengujian

| | TP | FP | FN |
|------------|-----|----|----|
| Person | 43 | 0 | 0 |
| Helmet | 30 | 1 | 0 |
| Non-Helmet | 10 | 0 | 0 |
| Vest | 32 | 0 | 0 |
| Gloves | 25 | 2 | 1 |
| Bare-arms | 14 | 0 | 2 |
| Shoes | 51 | 0 | 0 |
| Total | 209 | 3 | 3 |

Perhitungan untuk menentukan Akurasi, Presisi, *Recall* dan *F-1 Score*nya, dapat dilihat sebagai berikut:

a) *Accuracy*

Rumus menghitung *Accuracy* dapat dilihat pada persamaan (1)

$$\begin{aligned}
 \text{Person} &= \frac{43}{43+0+0} \times 100\% = 100\% \\
 \text{Helmet} &= \frac{30}{30+1+0} \times 100\% = 96,7\% \\
 \text{Non-Helmet} &= \frac{10}{10+0+0} \times 100\% = 100\% \\
 \text{Vest} &= \frac{32}{32+0+0} \times 100\% = 100\% \\
 \text{Gloves} &= \frac{25}{25+2+1} \times 100\% = 89,2\% \\
 \text{Bare-arms} &= \frac{14}{14+0+2} \times 100\% = 87,5\% \\
 \text{Shoes} &= \frac{51}{51+0+0} \times 100\% = 100\%
 \end{aligned}$$

b) *Precision*

Rumus menghitung *precision* dapat dilihat pada persamaan (1)

$$\begin{aligned}
 \text{Person} &= \frac{43}{43+0} \times 100\% = 100\% \\
 \text{Helmet} &= \frac{30}{30+1} \times 100\% = 96,7\% \\
 \text{Non-Helmet} &= \frac{10}{10+0} \times 100\% = 100\% \\
 \text{Vest} &= \frac{32}{32+0} \times 100\% = 100\% \\
 \text{Gloves} &= \frac{25}{25+2} \times 100\% = 92,5\% \\
 \text{Bare-arms} &= \frac{14}{14+0} \times 100\% = 100\% \\
 \text{Shoes} &= \frac{51}{51+0} \times 100\% = 100\%
 \end{aligned}$$

c) *Recall*

Rumus menghitung *Recall* dapat dilihat pada persamaan (1)

$$\begin{aligned}
 \text{Person} &= \frac{43}{43+0} \times 100\% = 100\% \\
 \text{Helmet} &= \frac{30}{30+0} \times 100\% = 100\% \\
 \text{Non-Helmet} &= \frac{10}{10+0} \times 100\% = 100\% \\
 \text{Vest} &= \frac{32}{32+0} \times 100\% = 100\%
 \end{aligned}$$

$$\text{Gloves} = \frac{25}{25+1} \times 100\% = 96,1\%$$

$$\text{Bare-arms} = \frac{14}{14+2} \times 100\% = 87,5\%$$

$$\text{Shoes} = \frac{51}{51+0} \times 100\% = 100\%$$

d) *F-1 score*

Rumus menghitung *F-1 score* dapat dilihat pada persamaan (1)

$$\text{Person} = 2 \times \frac{(100\% \times 100\%)}{(100\% + 100\%)} = 100\%$$

$$\text{Helmet} = 2 \times \frac{(96,7\% \times 100\%)}{(96,7\% + 100\%)} = 98,3\%$$

$$\text{Non-Helmet} = 2 \times \frac{(100\% \times 100\%)}{(100\% + 100\%)} = 100\%$$

$$\text{Vest} = 2 \times \frac{(100\% \times 100\%)}{(100\% + 100\%)} = 100\%$$

$$\text{Gloves} = 2 \times \frac{(92,5\% \times 96,1\%)}{(92,5\% + 96,1\%)} = 94,2\%$$

$$\text{Bare-arms} = 2 \times \frac{(100\% \times 87,5\%)}{(100\% + 87,5\%)} = 93,3\%$$

$$\text{Shoes} = 2 \times \frac{(100\% \times 100\%)}{(100\% + 100\%)} = 100\%$$

Hasil dari perhitungan masing-masing class dapat disimpulkan pada tabel 4.5

Tabel 4.5 Hasil perhitungan

| | <i>Precision</i> | <i>Recall</i> | <i>F-1 score</i> |
|------------|------------------|---------------|------------------|
| Person | 100% | 100% | 100% |
| Helmet | 96,7% | 100% | 98,3% |
| Non-Helmet | 100% | 100% | 100% |
| Vest | 100% | 100% | 100% |
| Gloves | 92,5% | 96,1% | 94,2% |
| Bare-arms | 100% | 87,5% | 93,3% |
| Shoes | 100% | 100% | 100% |

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Sesudah melalui Langkah-langkah pengujian untuk penelitian Alat Pelindung Diri (APD) dengan metode *You Only Look Once Versi 5* (YOLOv5) dapat disimpulkan sebagai berikut :

1. Metode *You Only Look Once* mampu melakukan deteksi Alat Pelindung Diri. Penelitian ini mendapatkan akurasi sebesar 92,07%
2. Untuk mendapatkan model yang terbaik, *epoch* dan *batch size* dapat memengaruhi performa dari model dalam mendeteksi
3. *Angle* kamera yang jauh merupakan faktor yang mampu membuat model tidak dapat melakukan deteksi secara sempurna, dikarenakan bentuk APD yang tidak terlalu jelas jika *class* yang melekat pada objek terlihat kecil
4. Untuk *class bare-arms* dan *Gloves* terdapat tantangan dalam deteksi karena bentuk nya yang sama-sama berbentuk tangan manusia, maka faktor yang paling membedakan yaitu warna pada objek
5. Pada proses *training* dihasilkan model terbaik pada 250 *epoch* dan menggunakan 64 *batch size*.

5.2 Saran

Adapun saran yang bisa digunakan untuk referensi pada penelitian berikutnya sebagai berikut :

1. Dalam mengambil data *testing*, perlu diambil dari beberapa *angle* yang mampu mendapatkan objek untuk tidak terlalu jauh dari kamera untuk mendapatkan akurasi yang lebih besar.
2. Menambah fitur rekapitulasi data untuk pekerja yang tidak menggunakan APD dengan mendata pekerja yang terdeteksi Non-APD.
3. Untuk penelitian berikutnya, dataset dan hyperparameter dapat diubah-ubah untuk mendapatkan angka akurasi yang lebih dari penulis.
4. Melakukan penelitian yang sama dengan menggunakan algoritma yang lain agar dapat dibandingkan dengan performa menggunakan metode *You Only Look Once Versi 5*.

DAFTAR PUSTAKA

- Amerigo, M. (2022). *Alat Pelindung Diri di Bidang Konstruksi*. Bandung: <https://depobeta.com/magazine/artikel/alat-pelindung-diri-di-bidang-konstruksi/>.
- Cheng, Y.-H., Tseng, C.-Y., Nguyen, D.-M., & Lin, Y.-D. (2022). YOLOv4-Driven Appearance Grading Filing Mechanism: Towards a High-Accuracy Tomato Grading Model through a Deep-Learning Frameworks. *Mathematics Journal*, 10.
- Gai, R., Chen, N., & Yuan, H. (2023). A detection algorithm for cherry fruits based on improved yolo-v4 model. *Neural Computing and Applications Journal*, 13895-13906.
- Gerald, C., & Lubis, C. (2022). Pendeteksian dan Pengenalan Jenis Mobil menggunakan algoritam You Only Look Once dan Convolutional Neural Network. *Jurnal Ilmu Komputer dan Sistem Informasi*, 197-199.
- Hanh, L. D., & Bao, D. N. (2022). Autonomous lemon grading system by using machine learning and traditional image processing. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 445-452.
- Holleman, M. (2017). *Real-Time Object Detection with YOLO*. California: Matthijs Holleman.
- Inspektorat Jenderal. (2021). *Mengenal APD dan APK*. Jakarta: Kementerian Pekerjaan Umum dan Perumahan Rakyat.
- Laily, M. E., Fajri, F. N., & Pratamasunu, G. Q. (2022). Deteksi Penggunaan Alat Pelindung Diri (APD) Untuk Keselamatan dan Kesehatan Kerja (K3) Menggunakan Metode Mask Region Convolutional Neural Network (Mask R-CNN). *Jurnal Politeknik Caltex Riau*, 279-288.
- Li, J., Liu, H., Wang, T., Jiang, M., Wang, S., Li, K., & Zhuo, X. (2017). Safety helmet wearing detection based on image processing and machine learning. *2017 Ninth International Conference on Advanced Computational Intelligence (ICACI)*, 201-205.
- Loey, M., Manogaran, G., Taha, M. H., & Khalifa, N. E. (2020). Fighting against COVID-19: A novel deep Learning model based on YOLO-v2 with ResNet-50 for medical face mask detection. *Sustainable Cities and Society Journal*, Vol.65.
- Mailoa, R. M., & Santoso, L. W. (2022). Deteksi Rompi dan Helm Keselamatan Menggunakan Metode YOLO dan CNN. *Jurnal Infra*, 49-55.
- Nepal, U., & Eslamiat, H. (2021, February 2). Comparing YOLOv3, YOLOv4, and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *MDPI AG*, 464.

- Putra, I. K., Giovani, I. D., Naufal, M. Z., Rahayu, N. K., Sangka, I. G., & Purnama, I. B. (2022). Sistem Pengawasan K3 pada PT. PLN Indonesia Power Bali PGU. *Journal of Applied Mechanical Engineering and Green*, 110-112.
- Sisco, J., Saniputra, F. R., Pratama, Y., Dharmawan, M. R., & Cholissodin, I. (2020). Pengembangan Deteksi Citra Mobil Untuk Mengetahui Jumlah Tempat Parkir Menggunakan Cuda Dan Modified YOLO. *Jurnal Teknologi Informasi dan Ilmu Komputer*, Vol 6 No 4.
- Song, H., Han, X.-y., Montenegro-Marin, C. E., & Krishnamoorthy, S. (2021). Secure prediction and assessment of sports injuries using deep learning based convolutional neural network. *Journal of Ambience & Human Computing*, 3399-3410.
- Syaikhoni, A., & Ariyadi, A. (2018). Deteksi Objek dengan Tensorflow Object Detection API. *Master of Computer Science Journal*, 31-35.
- Viktoria, P. (2018). Vehicle Detection Based on Convolutional Neural Networks. *Institute of Computer Science Journal*, 64-107.
- Wahyu, A. P., & Nugraha, U. (2018). Weight Measurement and Identification of Cow Type Using Computer Vision Method. *International Journal of Engineering & Technology*, 78-107.
- Yu, J., & Zhang, W. (2021). Face mask wearing detection algorithm based on improved YOLOv4. *Journal of Physics: Conference Series*, 22-58.
- Zhang, Y., Guo, Z., Wu, J., Tian, Y., Tang, H., & Guo, X. (2021). Real-Time Vehicle Detection Based on Improved YOLOv5. *MDPI AG*, 14-26.
- Zhong, M., & Meng, F. (2019). A YOLOv3-based non-helmet-use detection for seafarer safety aboard merchant ships. *Journal of Physics: Conference Series*, 012096.



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

UNIVERSITAS SUMATERA UTARA
FAKULTAS ILMU KOMPUTER
DAN TEKNOLOGI INFORMASI

Jalan Universitas No. 9A Gedung A, Kampus USU Medan 20155, Telepon: (061) 821007
Laman: <http://Fasilkomti.usu.ac.id>

KEPUTUSAN
DEKAN FAKULTAS ILMU KOMPUTER
DAN TEKNOLOGI INFORMASI
NOMOR :236/UN5.2.1.14/SK/SPB/2024
DEKAN FAKULTAS ILMU KOMPUTER
DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA

- Membaca : Surat Permohonan Mahasiswa Fasilkom-TI USU tanggal 8 Januari 2024 perihal permohonan ujian skripsi.
Nama : FAISAL LAZUARDY M
NIM : 191402075
Program Studi : Sarjana (S-1) Teknologi Informasi
Judul Skripsi : Deteksi Kelengkapan Alat Pelindung Diri Untuk Pekerja Proyek Menggunakan Metode You Only Look Once Versi 5 (Yolov5)
- Memperhatikan : Bahwa Mahasiswa tersebut telah memenuhi kewajiban untuk ikut dalam pelaksanaan Meja Hijaui Skripsi Mahasiswa pada Program Studi Sarjana (S-1) Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara TA 2023/2024.
- Menimbang : Bahwa permohonan tersebut diatas dapat disetujui dan perlu ditetapkan dengan surat keputusan
- Mengingat : 1. Undang-undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional
2. Peraturan Pemerintah Nomor 17 tahun 2010 tentang pengelolaan dan penyelenggara pendidikan.
3. Keputusan Rektor USU Nomor 03/UN5.1.R/SK/SPB/2021 tentang Peraturan Akademik Program Sarjana Universitas Sumatera Utara.
4. Surat Keputusan Rektor USU Nomor 1876/UN5.1.R/SK/SDM/2021 tentang pengangkatan Dekan Fasilkom-TI USU Periode 2021-2026
- Menetapkan :
Pertama : MEMUTUSKAN
: Membentuk dan mengangkat Tim Penguji Skripsi mahasiswa sebagai berikut:
Ketua : Dedy Arsandi ST., M.Kom.
NIP: 197908312009121002
Sekretaris : Fahrurrozi Lubis B.IT., M.Sc.IT
NIP: 198610122018052001
Anggota Penguji : Anul Hiziandi S.Kom, M.Sc
NIP: 198510272017061001
Anggota Penguji : Seniman S.Kom., M.Kom.
NIP: 198705252014041001
Moderator : -
Penitera : -
- Kedua : Segala biaya yang diperlukan untuk pelaksanaan kegiatan ini dibebankan pada Dana Penerimaan Bukan Pajak (PNPB) Fasilkom-TI USU Tahun 2024.
- Ketiga : Keputusan ini berlaku sejak tanggal ditetapkan dengan ketentuan bahwa segala sesuatunya akan diperbaiki sebagaimana mestinya apabila dikemudian hari terdapat kekeliruan dalam surat keputusan ini.

Ditetapkan di : Medan
Pada Tanggal 11 Januari 2024
Dekan,

MAYA SILVI LYDIA
NIP 197401272002122001

- Tembusan :
1. Ketua Program Studi Sarjana (S-1) Teknologi Informasi
 2. Yang bersangkutan
 3. Arsip