

# Deteksi Kelengkapan APD Keselamatan Pekerja Konstruksi Dengan Menggunakan Metode *Convolutional Neural Network* Dan Model Jaringan *YOLOv5*

Imanuela Gloriagnys Natalia<sup>1</sup>, Asmunin<sup>2</sup>

<sup>1,3</sup>Manajemen Informatika, Fakultas Program Vokasi, Universitas Negeri Surabaya

[imanuela.19032@mhs.unesa.ac.id](mailto:imanuela.19032@mhs.unesa.ac.id)

[asmunin@unesa.ac.id](mailto:asmunin@unesa.ac.id)

**Abstrak**— Pada saat ini, keselamatan dan Kesehatan pekerja dalam industry konstruksi sangatlah penting. Salah satu aspek kunci dalam menjaga keselamatan pekerja adalah dengan memastikan bahwa mereka telah menggunakan peralatan perlindungan diri (APD) seperti helm pelindung kepala, rompi keselamatan, masker dan lainnya dengan benar serta lengkap. Metode You Only Look Once (YOLO) merupakan metode mutakhir deteksi objek dengan tingkat akurasi tinggi yang dapat membantu merealisasikan produk yang dibuat peneliti buat. Untuk mengurangi resiko terjadinya kecelakaan kerja, peneliti menggunakan teknologi kecerdasan buatan untuk membuat model deteksi kelengkapan APD keselamatan pekerja konstruksi berdasarkan citra pekerja konstruksi yang menggunakan APD dengan tingkat performa yang cukup baik, dimana ketika model mendeteksi APD lengkap maka akan muncul keterangan bahwa benar APD lengkap sedangkan jika tidak lengkap maka akan muncul keterangan bahwa APD tidak lengkap, sehingga pekerja dapat melengkapi terlebih dahulu APD yang digunakan.

**Kata Kunci**— You Only Look Once, Kecerdasan Buatan, Model Deteksi APD, Deteksi Objek.

## I. PENDAHULUAN

Pada saat ini, keselamatan dan kesehatan pekerja di industri konstruksi sangat penting. Memastikan bahwa pekerja telah menggunakan peralatan perlindungan diri (APD) seperti helm pelindung kepala, rompi keselamatan, masker, dan lainnya dengan benar dan lengkap adalah bagian penting dari menjaga keselamatan pekerja. Mengacu pada data laporan tahunan Badan Penyelenggara Jaminan Sosial (BPJS) Ketenagakerjaan selama delapan tahun terakhir (2015–November 2022), jumlah kecelakaan kerja terus meningkat setiap tahunnya. Menurut data BPJS Ketenagakerjaan, jumlah kecelakaan kerja pada tahun 2022 mencapai 265.334 kasus [1]. Menteri Ketenagakerjaan Ida Fauziah menegaskan bahwa keselamatan, keamanan, dan kesehatan kerja harus menjadi prioritas utama untuk meningkatkan produktivitas kerja [1]. [1]. Sebaik-baiknya pengaturan keselamatan dan kesehatan kerja akan membuat tempat kerja aman dan nyaman,

meningkatkan produktivitas. Setiap pekerja harus menggunakan alat pelindung diri sebagai bagian dari upaya untuk mengurangi risiko kecelakaan kerja. APD dasar yang harus digunakan oleh pekerja konstruksi termasuk 1) helm, 2) masker, 3) rompi keselamatan, dan 4) sepatu bot pelindung. Menurut Peraturan Menteri Tenaga Kerja dan Transmigrasi Republik Indonesia No. 8 Tahun 2010, Pasal 6 Ayat 1 mengatur bahwa pekerja, pekerja, dan orang lain harus memakai atau menggunakan alat pelindung diri (APD) sesuai dengan potensi bahaya dan risiko yang ada di tempat kerja [2].

Jaminan Kecelakaan Kerja (JKK) BPJS Ketenagakerjaan tahun 2022, masih menunjukkan kecenderungan peningkatan kasus setiap tahunnya. Pada tahun 2021 tercatat sebanyak 234.370 kasus yang menyebabkan kematian pekerja/buruh sebanyak 6.552 orang, meningkat sebesar 5,7 % dibandingkan dengan tahun 2020. Angka tersebut menjadi indikasi bahwa penerapan K3 harus semakin menjadi prioritas bagi dunia kerja di Indonesia [3]. Deteksi alat pelindung diri dalam beberapa tahun terakhir telah mendapatkan perhatian yang cukup signifikan. Hal ini dianggap penting karena sangat diperlukan dalam manajemen keselamatan dan produktivitas para pekerja konstruksi [4]. Untuk mengidentifikasi APD keselamatan pekerja konstruksi, teori dan algoritma Deep Learning (Pembelajaran Mendalam) telah digunakan. Oleh karena itu, peneliti ingin mengembangkan model untuk mendeteksi kelengkapan APD keselamatan pekerja konstruksi. Model ini akan menggunakan gambar pekerja yang mengenakan APD melalui visi komputer dan mengidentifikasi kelasnya dari dataset yang ada. Kelas tersebut termasuk hardhat, mask, vest, dan boots. Untuk mengekstraksi fitur, data citra pekerja yang mengenakan APD dimasukkan ke jaringan saraf. Sistem akan mengidentifikasi APD berdasarkan kelas saat ini. Peneliti akan menguji dan menerapkan produk model deteksi mereka pada aplikasi berbasis web.

Kecerdasan buatan adalah suatu ciptaan manusia yang memiliki kemampuan diterapkan pada mesin, sehingga mesin dapat memahami sesuatu yang biasanya dilakukan oleh manusia. Kecerdasan buatan (Artificial Intelligence) merupakan topik luas yang mencakup Machine Learning (ML) dan Deep Learning (DL) [5]. Salah satu bidang studi

> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <

kecerdasan buatan adalah pembelajaran mesin, yang juga disebut sebagai pembelajaran mesin. Di mana seseorang dapat membuat program tanpa menuliskannya [6]. Hanya dibutuhkan proses input dan output, mesin akan secara otomatis belajar dan membuat pola yang sesuai dengan program yang akan dibuat. Pembelajaran mesin dilatih menggunakan data (juga dikenal sebagai pembelajaran data), yang kemudian divalidasi dengan data validasi, dan terakhir dilakukan pengujian dengan data pengujian. Inspirasi dari cara otak manusia menjalankan data dari lapisan pertama hingga lapisan terakhir, Jaringan Saraf Tiruan, atau ANN, adalah salah satu algoritma pembelajaran mesin yang paling populer [6]. Dalam cabang ilmu machine learning berbasis Jaringan Saraf Tiruan (JST), deep learning mengajarkan komputer untuk mengklasifikasi gambar atau suara secara langsung [7]. Salah satu kelas deep feed-forward artificial neural networks yang banyak digunakan dalam analisis citra adalah convolutional neural network (CNN), juga dikenal sebagai ConvNet atau CNN, yang memiliki kemampuan untuk mendeteksi dan mengenali objek pada gambar [8]. CNN terdiri dari satu lapisan masukan (input layer), satu lapisan keluaran (output layer), dan beberapa lapisan tersembunyi. Tujuan dari metode supervised learning adalah untuk mengelompokkan data yang berlabel ke dalam data sebelumnya [9].

You Only Look Once adalah jaringan konvolusi terpadu yang dapat memperkirakan probabilitas kelas dan bounding box secara bersamaan. YOLO sangat baik dalam melatih gambar dan mengoptimalkan model deteksi. Model terintegrasi ini memiliki banyak keuntungan dibandingkan dengan metode tradisional yang berfokus pada deteksi objek [10]. Roboflow merupakan sebuah platform yang tersedia di web yang memiliki banyak fungsi yang berhubungan dengan dataset. Dengan menggunakan roboflow ini dapat membagikan dataset sekaligus memproses dataset tersebut. Flask adalah web framework yang ditulis dalam bahasa Python dan memiliki dua lingkungan luar, WSGI toolkit dan Jinja2 template engine. Karena itu, Flask adalah jenis microframework yang tidak memerlukan lingkungan luar tertentu untuk digunakan. Validation Form, Handling Upload, dan Database adalah beberapa fitur dan komponen pihak ketiga yang dapat ditambahkan ke Flask melalui penggunaan ekstensi Flask. Python adalah salah satu bahasa pemrograman komputer yang paling banyak digunakan perusahaan besar di seluruh dunia saat ini. Ini digunakan oleh banyak teknologi canggih seperti pengajaran mesin dan situs web.

Metode You Only Look Once (YOLO) adalah teknik deteksi objek baru dan sangat akurat yang dapat membantu peneliti membuat produk. YOLO adalah algoritma baru yang dikembangkan dari metode Convolutional Neural Network (CNN) dan berfokus pada deteksi objek dengan menggunakan hasil pengklasifikasi ulang atau kembali untuk menjalankan fitur deteksi. Arsitektur terpadu YOLO sangat cepat dan memiliki kemampuan untuk memproses gambar secara real-time dengan 45 FPS (frame per seconds). Algoritma jaringan metode YOLO yang lebih kecil disebut Fast YOLO dan memiliki kemampuan untuk memproses gambar secara real-time dengan 155 FPS [10]. YOLOv5 adalah proyek sumber

terbuka yang diumumkan pada bulan Juni 2020 oleh Ultralytics LLC di GitHub. Meskipun belum ada publikasi dokumen yang relevan, proyek ini telah diterima dan diakui oleh banyak orang yang berhubungan dengannya, dan kodenya terus diperbarui. YOLOv5 menggunakan kerangka kerja PyTorch dan memiliki dukungan yang lebih sederhana, penerapan yang lebih mudah, dan parameter model yang lebih sedikit dibandingkan dengan YOLOv4. Selain itu, YOLOv5 mendukung konversi ke ONNX dan CoreML, membuatnya mudah digunakan di terminal seluler dan perangkat tertanam. Matriks Intersection Over Union digunakan untuk mengukur akurasi model dan mendeteksi objek pada kumpulan data yang terlatih. IOU sangat bermanfaat untuk membandingkan objek nyata atau gambar dengan kotak pembatas yang diprediksi dari model. Parameter ini berfungsi sebagai ambang batas deteksi. Matriks akurasinya disebut mAP 0,5 karena ambang batasnya adalah 0.5. YOLOv5 memiliki sekitar tiga puluh (30) hyperparameter berbeda yang digunakan untuk pengaturan latihan yang berbeda. Jika Anda ingin mendapatkan hasil yang lebih baik, sehingga dapat mengoptimalkannya. Jika optimasi parameter lebih baik, maka akan mendapatkan hasil yang lebih baik. Jika Anda tidak terlalu familiar dengan hal ini, Anda dapat menggunakan nilai default yang digunakan dalam pelatihan pada dataset COCO[11].

YOLO adalah salah satu dari algoritma *Deep Learning* yang prosesnya memanfaatkan jaringan saraf pada CNN dalam deteksi objek. Algoritma YOLO membagi sebuah citra ke dalam bentuk *grid* berukuran (sisi x sisi) yang selanjutnya pada setiap *grid* akan memperkirakan *bounding box* serta peta kelas pada masing-masing *grid*. Apabila pada salah satu *grid* mendeteksi objek, maka pada *grid* tersebut akan diprediksi oleh *bounding box* yang mengelilingi objek yang terdeteksi. Nilai *confidence* akan dihitung pada masing-masing *bounding box* yang kemudian akan diseleksi berdasarkan nilai yang diperoleh.

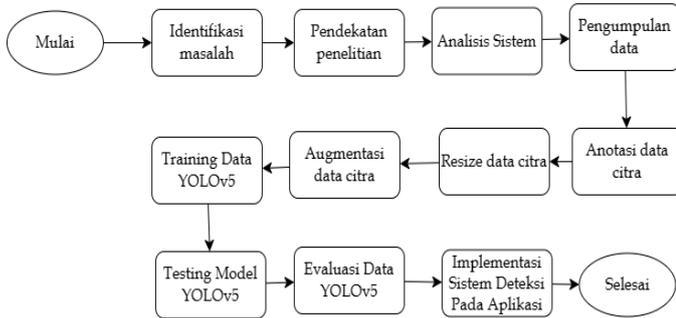
YOLO memproses data dengan cepat karena menggunakan regresi dan tidak menggunakan saluran yang kompleks. YOLO yang dibuat oleh Bochkovskiy hanya menggunakan jaringan saraf Bochkovskiy pada citra gambar baru selama waktu pengujian untuk menebak objek. Pada GPU Titan X, jaringan dasar Bochkovskiy bekerja pada 45 fps tanpa pemrosesan batch, dan versi tercepat YOLO Bochkovskiy buatan bekerja pada 150 fps. Ini berarti Bochkovskiy dapat memproses video streaming secara real-time dengan latensi kurang dari 25 milidetik. YOLO juga lebih presisi dari rata-rata model real-time metode lainnya [10]

Nilai probabilitas bersyarat Kelas C,  $Pr(Kelas | Objek)$ , diprediksi oleh setiap sel grid. Nilai kemungkinan didasarkan pada sel mesh di mana objek berada. Tidak peduli berapa banyak kotak B, Redmon dkk. memprediksi hanya satu set probabilitas kelas yang ada di setiap sel grid. Selama pengujian, mereka mengalikan probabilitas kelas bersyarat dengan nilai keyakinan yang diprediksi untuk setiap kotak, yang menghasilkan skor keyakinan kelas khusus untuk setiap kotak. Skor ini mengkodekan probabilitas kelas yang ada di dalam kotak dan seberapa baik kotak yang diprediksi sama atau sebanding dengan objek lain [10].

> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <

## II. METODE PENELITIAN

Pada bab ini, menjelaskan tentang alur tahapan yang dilakukan peneliti selama proses penelitian berlangsung. Adapun alur tahapan yang dilakukan dapat dilihat pada Gambar 1 di bawah:



Gbr. 1. Alur Penelitian

### A. Identifikasi Masalah

Secara umum identifikasi masalah pada penelitian ini didapatkan berdasar studi literatur yang diakses melalui beberapa situs seperti Academia, Google Scholar, Arxiv, serta sumber-sumber berita aktual terkait tingginya tingkat kecelakaan kerja diakibatkan kurang patuhnya terhadap penggunaan APD ketika bekerja yang diakibatkan oleh human error. Hal ini mendasari dilakukan sebuah penelitian untuk menanggulangi permasalahan tersebut.

### B. Pendekatan Penelitian

Berdasarkan temuan masalah terkait tingginya tingkat kecelakaan kerja yang disebabkan oleh human error, maka penelitian ini dibuat sebuah sistem deteksi kelengkapan APD pada pekerja konstruksi secara real-time yang memproses citra gambar menggunakan metode CNN dan model YOLOv5 dengan tingkat akurasi yang cukup tinggi.

### C. Analisis Sistem

Analisis sistem pada penelitian ini diperlukan untuk mengetahui kebutuhan dan konfigurasi untuk membuat sistem deteksi kelengkapan APD keselamatan pekerja konstruksi. Berikut beberapa spesifikasi, aplikasi dan *library* yang diperlukan.

#### 1) Kebutuhan Spesifikasi Komputer Minimal

- CPU Intel Core i5
- RAM 8 GB
- Storage 1 GB
- OS (Windows / Linux / MacOS)

#### 2) Aplikasi dan Modul

- Visual Studio Code
- Python
- Roboflow
- Flask
- Google Colab

### 3) Input dan output

- *Input*: Citra pekerja konstruksi mengenakan APD.
- *Output*: Hasil deteksi APD pada pekerja konstruksi, apabila APD lengkap true sedangkan jika APD tidak lengkap false.

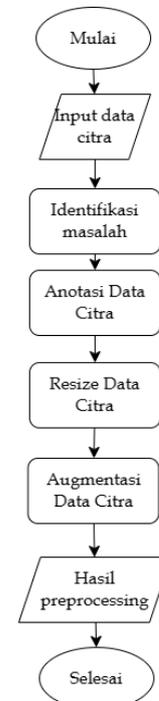
### D. Pengumpulan Data

Pada penelitian ini data dikumpulkan melalui web scrapping dengan kata kunci “PPE for construction worker” yang dikumpulkan memiliki total sebanyak 519 citra. Yang kemudian diolah menggunakan Roboflow. Data akan di anotasi, kemudian dilakukan preprocessing seperti resize gambar menjadi ukuran 640x640, augmentasi gambar, dan split gambar menjadi train, val, dan test.



Gbr. 2. Dataset

### E. Pre-processing Data



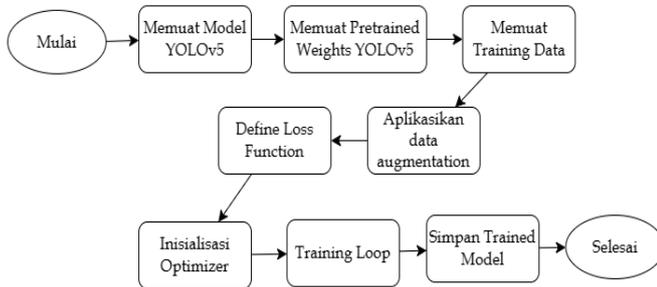
Gbr. 3. Alur Pre-processing Data

Gambar 3 Menjelaskan proses ini data yang telah dikumpulkan diupload ke platform Roboflow, kemudian di anotasikan menjadi 4 kelas yaitu hardhat, mask, vest, dan boots. Setelah dilakukan penganotasian data maka data akan dibagi menjadi 3 yaitu train, val, test, yang kemudian

> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <

dilanjutkan dengan resize data menjadi ukuran 640x640, lalu dilakukan augmentasi pada data gambar. Setelah itu dataset dapat didownload dengan format YOLOv5 Pytorch supaya sesuai dengan model deteksi yang akan digunakan yaitu YOLOv5. Roboflow sendiri menyediakan fitur untuk training pada dataset yang telah diolah dengan hasil akhir menampilkan besar nilai precision, recall, dan mAP yang diperoleh.

#### F. Pelatihan Data



Gbr. 4. Alur Pelatihan Data

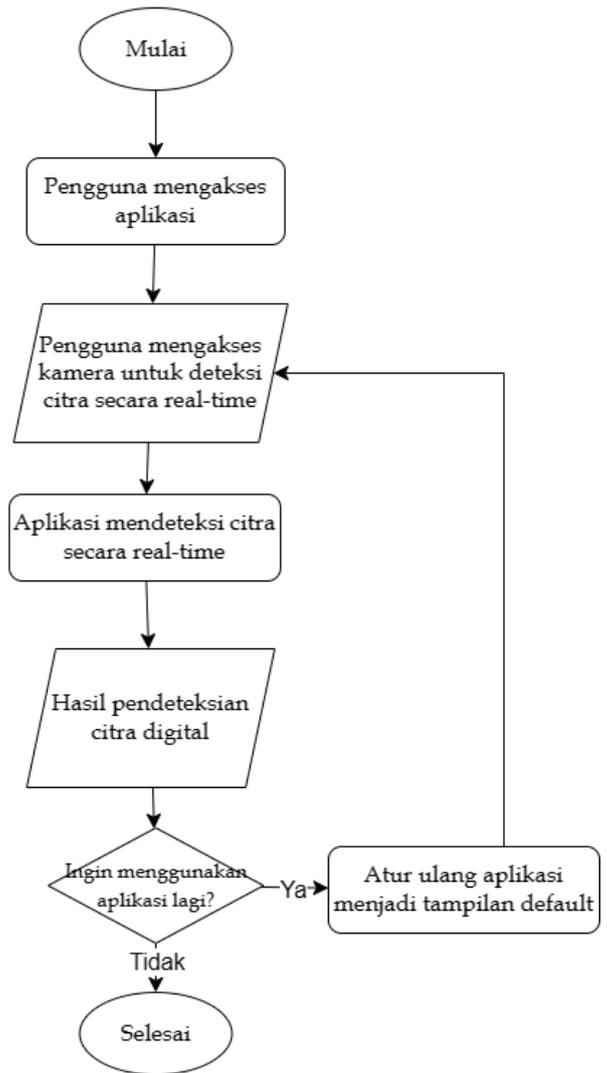
Gambar 4 menjelaskan alur proses training data dengan model YOLOv5 yang dilakukan menggunakan Google Colab yang telah tersambung dengan Google Drive agar hasil dari pemrosesan data tidak hilang dan dapat diakses Kembali ketika membutuhkan. Proses diawali dengan memuat dataset yang telah diolah menggunakan Roboflow sebelumnya kemudian dilanjutkan dengan memuat model YOLOv5 yang telah disediakan oleh Ultralytics. Tahapan selanjutnya memuat pretrained weights model, lalu dilanjutkan dengan define loss function, kemudian inisialisasi optimizer, selanjutnya pada tahapan training loop digunakan data akan diproses secara berkala berdasarkan berapa banyak epochs yang digunakan. Setelah data diproses maka data selanjutnya disimpan dan model akan dievaluasi guna mengetahui besar nilai akurasi model YOLOv5.

#### G. Evaluasi Data

Setelah melakukan beberapa kali proses pelatihan data dengan model YOLOv5 maka selanjutnya evaluasi data tersebut. Pada tahap ini akan menghasilkan nilai evaluasi performa model YOLOv5 yang telah dilatih dan dapat dilihat dari file weight yang terakhir disimpan setelah proses training data selesai. Nilai evaluasi ini berupa nilai precision, recall, mAP dan F1-Score.

#### H. Pengujian Produk Pada Aplikasi Website

Gambar 5 menunjukkan visualisasi dari alur kerja aplikasi pada penelitian, dimana user mengakses aplikasi melalui web browser dengan memasukkan URL aplikasi tersebut. Setelah itu user akan diminta untuk mengakses kamera guna memasukkan citra digital yang akan dideteksi oleh sistem. Maka aplikasi akan memproses dan kemudian menampilkan output pada sesuai dengan citra yang telah diinputkan. Jika user ingin menggunakan aplikasi kembali maka aplikasi akan ter-setting menjadi default awal, jika tidak maka aplikasi telah selesai digunakan.



Gbr. 5. Alur Kerja Sistem Deteksi Kelengkapan APD

### III. HASIL DAN PEMBAHASAN

Pada bab ini, menjelaskan hasil dan pembahasan yang telah dilakukan pada penelitian sistem deteksi kelengkapan APD, yang mencakup *pre-processing* data, pelatihan data, evaluasi data, dan hasil pengujian model pada aplikasi berbasis website.

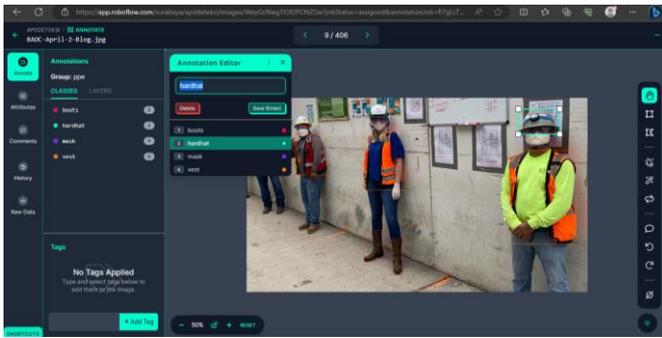
#### A. Pre-processing Data

Pada proses *pre-processing* data, peneliti memanfaatkan fitur yang disediakan oleh platform *Roboflow* membagi tahapan menjadi 4 bagian yaitu:

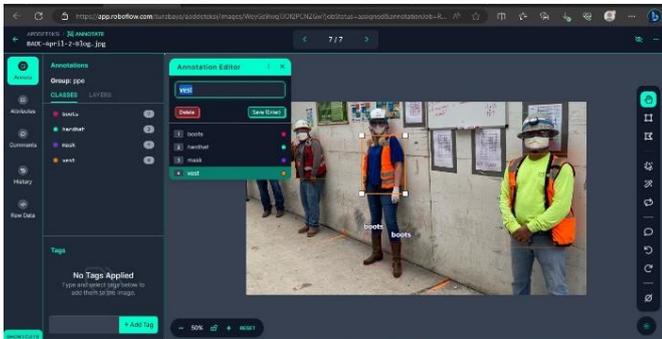
##### 1) Anotasi Data

Pada proses ini citra akan di-anotasi secara satu persatu yang dibagi menjadi 4 kelas berbeda. Kelas ini terbagi menjadi kelas untuk *hardhat*, kelas untuk *mask*, kelas untuk *vest* dan kelas untuk *boots*, dilakukan dengan memberi label berdasarkan kelas pada semua citra pada dataset.

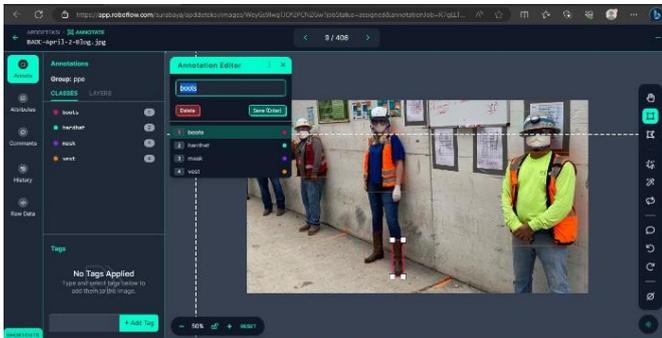
> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <



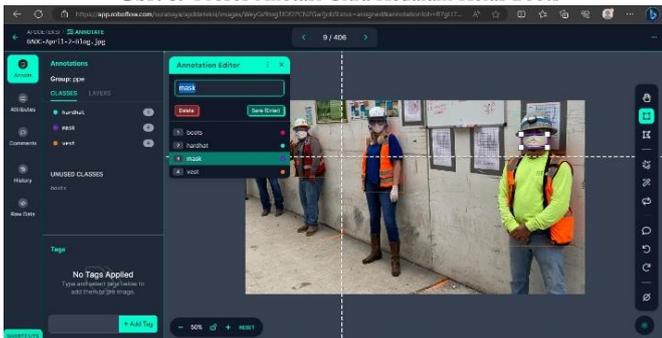
Gbr. 6. Proses Anotasi Citra Kedalam Kelas *Hardhat*



Gbr. 7. Proses Anotasi Citra Kedalam Kelas *Vest*



Gbr. 8. Proses Anotasi Citra Kedalam Kelas *Boots*



Gbr. 9. Proses Anotasi Citra Kedalam Kelas *Mask*

## 2) Split Data

Pada proses ini *Roboflow* akan membagi data citra yang telah dianotasi ke dalam masing-masing kelas menjadi *training set*, *validation set*, dan *testing set* dengan perbandingan pada data *training set* sebanyak 70% dengan total citra 362, *validation set* sebanyak 20% dengan total citra 105, dan *testing set* sebanyak 10% dengan total citra 52.

## 3) Pre-processing Data

### Preprocessing

What can preprocessing do?

Decrease training time and increase performance by applying image transformations to all images in this dataset.

Auto-Orient	Edit	×
Resize Stretch to 640x640	Edit	×
Tile 2 rows x 2 columns	Edit	×
+ Add Preprocessing Step		

Continue

Gbr. 11. Pre-processing Data Pada *Roboflow*

Gambar 11 menjelaskan Pada proses ini *Roboflow* melakukan pre-processing dengan resize data menjadi ukuran 640x640, dan memilih fitur tile guna membagi citra kedalam ukuran 2 baris x 2 kolom untuk memvisualisasikan gambar dalam bentuk grid terstruktur sehingga semakin mempermudah ketika melihat variasi yang ada pada dataset.

## 4) Augmentasi Data

Gambar 12 menjelaskan proses ini menggunakan fitur augmentasi pada *Roboflow* yang pertama dengan *flip* secara *horizontal* dan *vertical*, kemudian lakukan *rotation* antara  $-15^\circ$  dan  $+15^\circ$ , lalu pilih *blur up to 3px*, *noise up to 5% of pixels*, *bounding box: blur up to 2.5px*, *bounding box: noise up to 5% of pixels*.

### Augmentation

What can augmentation do?

Create new training examples for your model to learn from by generating augmented versions of each image in your training set.

Flip Horizontal, Vertical	Edit	×
Rotation Between $-15^\circ$ and $+15^\circ$	Edit	×
Blur Up to 3px	Edit	×
Noise Up to 5% of pixels	Edit	×
Bounding Box: Blur Up to 2.5px	Edit	×
Bounding Box: Noise Up to 5% of pixels	Edit	×
+ Add Augmentation Step		

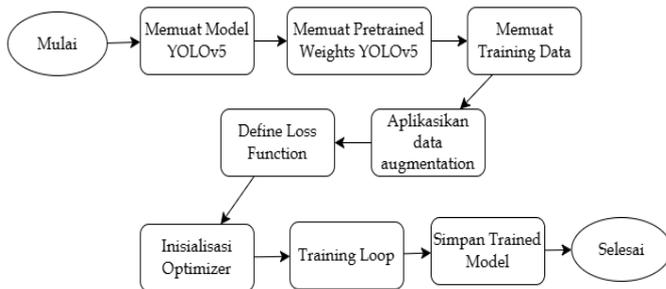
Continue

Gbr. 12. Pre-processing Data Pada *Roboflow*

> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <

## B. Pelatihan Data

Gambar 13 menjelaskan alur proses training data dengan model YOLOv5 yang dilakukan menggunakan Google Colab yang telah tersambung dengan Google Drive agar hasil dari pemrosesan data tidak hilang dan dapat diakses Kembali ketika membutuhkan. Proses diawali dengan memuat dataset yang telah diolah menggunakan Roboflow sebelumnya kemudian dilanjutkan dengan memuat model YOLOv5 yang telah disediakan oleh Ultralytics. Tahapan selanjutnya memuat pretrained weights model, lalu dilanjutkan dengan define loss function, kemudian inialisasi optimizer, selanjutnya pada tahapan training loop digunakan data akan diproses secara berkala berdasarkan berapa banyak epochs yang digunakan. Setelah data diproses maka data selanjutnya disimpan dan model akan dievaluasi guna mengetahui besar nilai akurasi model YOLOv5.



Gbr. 13. Alur Pelatihan Data

## C. Evaluasi Data

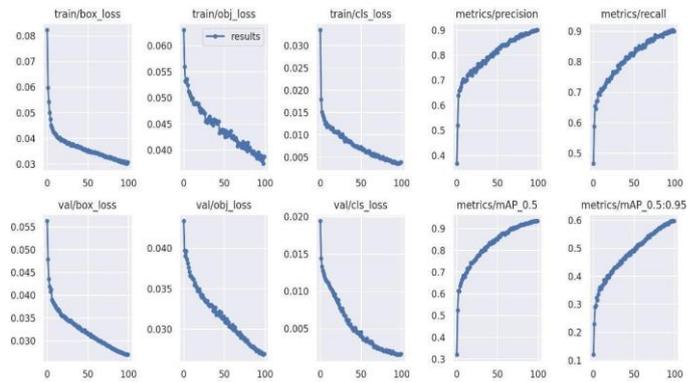
Setelah melakukan beberapa kali proses pelatihan data dengan model YOLOv5 maka selanjutnya evaluasi data tersebut. Pada tahap ini akan menghasilkan nilai evaluasi performa model YOLOv5 yang telah dilatih dan dapat dilihat dari file weight yang terakhir disimpan setelah proses training data selesai. Nilai evaluasi ini berupa nilai precision, recall, mAP dan F1-Score. Proses evaluasi data berlangsung bersamaan dengan proses pelatihan data. Hasil dari evaluasi performa model didapatkan data nilai precision sebesar 90%, nilai recall sebesar 90%, nilai mAP sebesar 93% dan nilai F1-Score sebesar 90%. Nilai ini tidak dapat mencapai akurasi yang cukup tinggi dikarenakan kurangnya spesifikasi dari laptop yang digunakan.

TABEL I  
HASIL PERBANDINGAN PELATIHAN DATA

Batch	Epochs	Precision	Recall	mAP	F1-Score
16	3	45%	40%	38%	42%
16	20	60%	55%	57%	57%
16	50	64%	54%	58%	58%
16	100	90%	90%	93%	90%

Tabel 1 menunjukkan bahwa hasil pelatihan yang paling baik adalah dengan menggunakan batch sebanyak 16 dan epochs sebanyak 100. Hasil dari evaluasi performa model didapatkan data nilai precision sebesar 90%, nilai recall

sebesar 90%, nilai mAP sebesar 93% dan nilai F1-Score sebesar 90%. Nilai ini tidak dapat mencapai akurasi yang cukup tinggi dikarenakan kurangnya spesifikasi dari laptop yang digunakan.



Gbr. 14. Hasil Evaluasi Data Pada Model YOLOv5

## D. Hasil Pengujian Produk Pada Aplikasi Website

Setelah hasil evaluasi berupa nilai *precision*, *recall*, mAP, dan F1-Score didapatkan, maka model dapat diterapkan dengan pengujian deteksi kelengkapan APD pada aplikasi berbasis website.

### 1) Halaman Index

Tampilan antarmuka indeks website deteksi APD didesain dengan tampilan yang intuitif dan mudah digunakan. Pada halaman utama ini terdapat fitur header yang akan mengarahkan pada halaman lain seperti beranda, tentang aplikasi, berita terkini, galeri deteksi, dan juga halaman deteksi. Dengan antarmuka indeks yang user-friendly ini, pengguna dapat dengan cepat menemukan APD yang sesuai dengan kebutuhan mereka dalam waktu singkat.



Gbr. 15. Halaman Index

### 2) Halaman Tentang Aplikasi

Menurut penjelasan di halaman ini, aplikasi deteksi kelengkapan APD pada pekerja bengkel las memainkan peran penting dalam meningkatkan keamanan dan perlindungan mereka. Dengan menggunakan aplikasi ini, pekerja bengkel las dapat memastikan bahwa mereka memiliki semua APD yang diperlukan saat melakukan tugas yang dibutuhkan di tempat kerja mereka. Secara keseluruhan, aplikasi ini meningkatkan keamanan dan keselamatan kerja pekerja bengkel las.

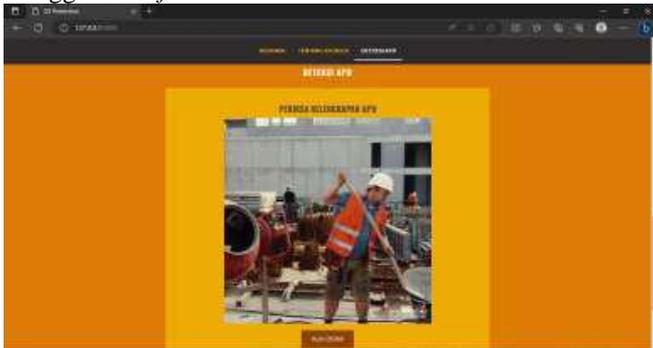
> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <



Gbr. 16. Halaman Tentang Aplikasi

### 3) Halaman Deteksi APD

Halaman ini akan mengarahkan ke halaman deteksi APD secara *real-time* menggunakan webcam yang dideploykan menggunakan *flask*.



Gbr. 17. Halaman Tentang Aplikasi

Gambar 18 menunjukkan halaman deteksi yang tersedia fitur untuk mendeteksi lengkap atau tidaknya APD yang digunakan, dimanaketika mengklik tombol berwarna coklat nanti akan diarahkan ke halaman deteksi kelengkapan APD.



Gbr. 18 Halaman Tentang Aplikasi

### E. Hasil Deteksi APD Pada Aplikasi Website Secara Real-Time

Tabel II menunjukkan perbandingan dari uji coba deteksi APD keselamatan pekerja konstruksi dengan metode Convolutional Neural Network dengan model jaringan YOLOv5 pada Aplikasi berbasis website menggunakan webcam secara *real-time* yang dideploymentkan menggunakan *flask*.

TABEL II  
HASIL DETEKSI APD PADA APLIKASI WEBSITE SECARA REAL-TIME

Gambar	Hasil
	<p>Terdeteksi: Tidak menggunakan APD</p> <p>Keterangan: - Terdeteksi false</p>
	<p>Terdeteksi: - Hardhat - Mask - Vest - Boots</p> <p>Confidence - 84.97 - 84.89 - 80.78 - 78.68</p> <p>Keterangan: - Menghadap ke depan - Terdeteksi true karena APD lengkap</p>

> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <

 <ul style="list-style-type: none"> <li>• ○ Name: vest</li> <li>○ Confidence: 84.85</li> <li>• ○ Name: hardhat</li> <li>○ Confidence: 82.49</li> <li>• ○ Name: mask</li> <li>○ Confidence: 79.74</li> <li>• ○ Name: boots</li> <li>○ Confidence: 75.43</li> <li>• ○ Safe Or Not: true</li> </ul>	<p><b>Terdeteksi:</b></p> <ul style="list-style-type: none"> <li>- Hardhat</li> <li>- Mask</li> <li>- Vest</li> <li>- Boots</li> </ul> <p><b>Confidence</b></p> <ul style="list-style-type: none"> <li>- 84.97</li> <li>- 84.89</li> <li>- 80.78</li> <li>- 78.68</li> </ul> <p><b>Keterangan:</b> Menghadap ke samping kiri terdeteksi true karena APD lengkap</p>	 <ul style="list-style-type: none"> <li>• ○ Name: hardhat</li> <li>○ Confidence: 74.47</li> <li>• ○ Name: vest</li> <li>○ Confidence: 73.82</li> <li>• ○ Name: boots</li> <li>○ Confidence: 72.72</li> <li>• ○ Safe Or Not: false</li> </ul>	<p><b>Terdeteksi:</b></p> <ul style="list-style-type: none"> <li>- Hardhat</li> <li>- Vest</li> <li>- Boots</li> </ul> <p><b>Confidence</b></p> <ul style="list-style-type: none"> <li>- 74.47</li> <li>- 73.82</li> <li>- 72.72</li> </ul> <p><b>Keterangan:</b> - Tidak menggunakan mask/ditutup dengan tangan - Terdeteksi false karena APD tidak lengkap</p>
 <ul style="list-style-type: none"> <li>• ○ Name: hardhat</li> <li>○ Confidence: 83.95</li> <li>• ○ Name: vest</li> <li>○ Confidence: 83.34</li> <li>• ○ Name: boots</li> <li>○ Confidence: 80.29</li> <li>• ○ Name: mask</li> <li>○ Confidence: 69.04</li> <li>• ○ Safe Or Not: true</li> </ul>	<p><b>Terdeteksi:</b></p> <ul style="list-style-type: none"> <li>- Hardhat</li> <li>- Mask</li> <li>- Vest</li> <li>- Boots</li> </ul> <p><b>Confidence</b></p> <ul style="list-style-type: none"> <li>- 83.95</li> <li>- 83.34</li> <li>- 80.29</li> <li>- 69.04</li> </ul> <p><b>Keterangan:</b> - Menghadap ke samping kanan - Terdeteksi true karena APD lengkap</p>	 <ul style="list-style-type: none"> <li>• ○ Name: vest</li> <li>○ Confidence: 86.77</li> <li>• ○ Name: mask</li> <li>○ Confidence: 85.23</li> <li>• ○ Name: boots</li> <li>○ Confidence: 80.38</li> <li>• ○ Safe Or Not: false</li> </ul>	<p><b>Terdeteksi:</b></p> <ul style="list-style-type: none"> <li>- Mask</li> <li>- Vest</li> <li>- Boots</li> </ul> <p><b>Confidence</b></p> <ul style="list-style-type: none"> <li>- 86.77</li> <li>- 85.23</li> <li>- 80.38</li> </ul> <p><b>Keterangan:</b> - Hardhat diganti dengan topi yang berbentuk mirip - Terdeteksi false karena topi bukan APD</p>

> REPLACE THIS LINE WITH YOUR MANUSCRIPT ID NUMBER (DOUBLE-CLICK HERE TO EDIT) <

 <ul style="list-style-type: none"> <li>• ○ Name: vest</li> <li>○ Confidence: 89.38</li> <li>• ○ Name: mask</li> <li>○ Confidence: 84.20</li> <li>• ○ Name: boots</li> <li>○ Confidence: 72.30</li> <li>• ○ Safe Or Not: false</li> </ul>	<p><b>Terdeteksi:</b></p> <ul style="list-style-type: none"> <li>- Mask</li> <li>- Vest</li> <li>- Boots</li> </ul> <p><b>Confidence</b></p> <ul style="list-style-type: none"> <li>- 89.38</li> <li>- 84.20</li> <li>- 72.30</li> </ul> <p><b>Keterangan:</b></p> <ul style="list-style-type: none"> <li>- Hardhat diganti dengan topi berbentuk lain</li> <li>- Terdeteksi false karena topi bukan APD</li> </ul>
 <ul style="list-style-type: none"> <li>• ○ Name: hardhat</li> <li>○ Confidence: 81.49</li> <li>• ○ Safe Or Not: false</li> </ul>	<p><b>Terdeteksi:</b></p> <ul style="list-style-type: none"> <li>- Hardhat</li> </ul> <p><b>Confidence</b></p> <ul style="list-style-type: none"> <li>- 81.49</li> </ul> <p><b>Keterangan:</b></p> <ul style="list-style-type: none"> <li>- Hanya mengenakan salah satu dari perlengkapan APD</li> <li>- Terdeteksi false karena APD tidak lengkap</li> </ul>

disertai *username*, tanggal dan waktu ketika subjek terdeteksi dalam kondisi lelah selama jangka waktu yang sudah ditentukan.

#### IV. KESIMPULAN DAN SARAN

Kesimpulan yang didapatkan setelah semua proses yang dikerjakan yaitu sistem deteksi kelengkapan APD keselamatan pekerja konstruksi berbasis website berjalan dengan baik dan menampilkan keterangan APD yang dideteksi benar lengkap

atau tidak. Metode Convolutional Neural Network dengan model jaringan YOLOv5 dapat berjalan dengan baik. Performa ketepatan sistem deteksi kelengkapan APD keselamatan pekerja konstruksi dengan model jaringan YOLOv5 mencakup nilai precision sebesar 90%, nilai recall sebesar 90%, nilai mAP sebesar 93% dan nilai F1-Score sebesar 90%. Nilai performa model deteksi ini kurang maksimal karena spesifikasi laptop kurang mumpuni.

Dari beberapa kesimpulan yang ada saran yang dapat peneliti sampaikan untuk pengembangan selanjutnya yaitu terbatasnya ketersediaan penggunaan GPU pada Google Colab yang hanya dapat dipakai selama 6 jam, alangkah baiknya untuk menggunakan Google Colab versi premium. Dataset yang digunakan kurang banyak sehingga pada proses training data mendapatkan hasil performa yang kurang maksimal. Karena keterbatasan dana dan ilmu yang dimiliki peneliti, maka produk hanya dapat diimplementasikan dalam bentuk website. Harapan kedepannya penelitian ini dapat memberi manfaat lebih luas pada pengembangan ilmu pengetahuan.

#### REFERENSI

- [1] M. Syaharani, "Jumlah Kecelakaan Kerja Indonesia dalam 8 Tahun Terakhir," *data.goodstats.id*, 2023. <https://data.goodstats.id/statistic/melasyhrm/jumlah-kecelakaan-kerja-indonesia-dalam-8-tahun-terakhir-sjo5X>.
- [2] Menteri Tenaga Kerja dan Transmigrasi, "Peraturan Menteri Tenaga Kerja dan Transmigrasi Republik Indonesia," *Peratur. Menteri tenaga Kerja dan Transm.*, vol. VII, no. 8, pp. 1–69, 2010, [Online]. Available: <https://indolabourdatabase.files.wordpress.com/2018/03/permenaker-no-8-tahun-2010-tentang-apd.pdf>.
- [3] K. K. R. INDONESIA, "PROFIL KESELAMATAN DAN KESEHATAN KERJA NASIONAL INDONESIA TAHUN 2022," *satudata.kemnaker.go.id*, 2022. [https://satudata.kemnaker.go.id/satudata-public/2022/10/files/publikasi/1675652225177\\_Profil%2520K3%2520Nasional%25202022.pdf](https://satudata.kemnaker.go.id/satudata-public/2022/10/files/publikasi/1675652225177_Profil%2520K3%2520Nasional%25202022.pdf).
- [4] H. Seong, H. Choi, H. Cho, S. Lee, H. Son, and C. Kim, "Vision-based safety vest detection in a construction scene," *ISARC 2017 - Proc. 34th Int. Symp. Autom. Robot. Constr.*, no. Isarc, pp. 288–293, 2017, doi: 10.22260/isarc2017/0039.
- [5] S. Singh, "Cousins of Artificial Intelligence," *towardsdatascience.com*, 2018. <https://towardsdatascience.com/cousins-of-artificial-intelligence-dda4edc27b55>.
- [6] R. K. Dinata and N. Hasdyna, "Machine Learning," 2020.
- [7] S. Ilahiyah and A. Nilogiri, "Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network," *JUSTINDO (Jurnal Sist. dan Teknol. Inf. Indones.)*, vol. 3, no. 2, pp. 49–56, 2018.
- [8] C. Geraldly and C. Lubis, "Jurnal Ilmu Komputer dan Sistem Informasi PENDETEKSIAN DAN PENGENALAN JENIS MOBIL MENGGUNAKAN ALGORITMA YOU ONLY LOOK ONCE DAN CONVOLUTIONAL NEURAL NETWORK."
- [9] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.
- [10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection." [Online]. Available: <http://pjreddie.com/yolo/>.
- [11] S. Yohanandan, "mAP (mean Average Precision) might confuse you!," *Towards Data Science*, 2020. <https://towardsdatascience.com/map-mean-average-precision-might-confuse-you-5956f1bfa9e2>.