

IMPLEMENTASI ARTIFICIAL INTELLIGENCE DAN INTERNET OF THINGS UNTUK MENDETEKSI PENGGUNAAN HELM PROYEK

Helmy*¹, Bilqis Syabilla Rosyada², Yunita Fitriyani³, Thomas Agung Setyawan⁴, Endro Wasito⁵, Catur Budi Waluyo⁶, Dianita Ratna Kusumatuti⁷

^{1,2,3,4,5,6,7}Politeknik Negeri Semarang, Semarang

Email: ¹helmy@polines.ac.id, ²bilqissybl@gmail.com, ³yunitafitriyani00@gmail.com, ⁴thomas@polines.ac.id, ⁵endro.wasito@polines.ac.id, ⁶catur.waluyo@polines.ac.id, ⁷dianita.ratna.kusumatuti@polines.ac.id

*Penulis Korespondensi

(Naskah masuk: 22 Januari 2024, diterima untuk diterbitkan: 27 Agustus 2025)

Abstrak

BPJS Ketenagakerjaan mencatat bahwa jumlah kecelakaan kerja di Indonesia meningkat dari 221.740 kasus pada tahun 2020 menjadi 234.370 kasus pada tahun 2021, dan terus meningkat hingga mencapai 256.334 kasus pada November 2022. Berdasarkan data dari Kementerian Ketenagakerjaan Indonesia, pada tahun 2020, 57,5% dari total 126,51 juta pekerja di Indonesia memiliki tingkat pendidikan rendah, yang berkontribusi pada rendahnya kesadaran akan pentingnya budaya Keselamatan dan Kesehatan Kerja (K3) serta penggunaan Alat Pelindung Diri (APD) yang sesuai standar. Pemantauan penggunaan APD, termasuk helm proyek di area konstruksi, masih dilakukan secara manual, yang dirasa kurang efisien. Penelitian ini bertujuan untuk mengembangkan alat berbasis Artificial Intelligence (AI) dan Internet of Things (IoT) yang dapat memantau penggunaan helm proyek secara real-time dengan akurasi tinggi dan dapat dipantau melalui dashboard. Alat ini terbukti lebih efektif dalam meminimalisir kecelakaan kerja, dengan rata-rata akurasi deteksi sebesar 84,65% untuk pekerja yang memakai helm dan 71,5% untuk yang tidak memakai helm. Penelitian ini menggunakan metode Agile yang melibatkan observasi, identifikasi kebutuhan, perancangan, pembuatan sistem, implementasi, dan pengujian. Hasil penelitian ini menunjukkan bahwa sistem yang dikembangkan mampu memberikan kemudahan bagi petugas K3 dalam melakukan pengawasan, sehingga dapat mengurangi risiko kecelakaan kerja.

Kata kunci: *Artificial Intelligence, Internet of Things, CNN, YOLO, Agile, Helm Proyek*

IMPLEMENTATION OF ARTIFICIAL INTELLIGENCE AND INTERNET OF THINGS FOR DETECTING THE USAGE OF PROJECT HELM

Abstract

The Workers Social Security Agency (BPJS Ketenagakerjaan) recorded that the number of work accidents in Indonesia increased from 221,740 cases in 2020 to 234,370 cases in 2021, and continued to rise, reaching 256,334 cases by November 2022. According to data from the Indonesian Ministry of Manpower, in 2020, 57.5% of the total 126.51 million workers in Indonesia had a low level of education, which contributed to a lack of awareness of the importance of Occupational Safety and Health (OSH) culture and the use of Personal Protective Equipment (PPE) that meets standards. The monitoring of PPE usage, including project helmets in construction areas, is still conducted manually, which is considered inefficient. This study aims to develop a tool based on Artificial Intelligence (AI) and the Internet of Things (IoT) that can monitor the use of project helmets in real-time with high accuracy and can be monitored through a dashboard. This tool has proven to be more effective in minimizing work accidents, with an average detection accuracy of 84.65% for workers wearing helmets and 71.5% for those not wearing helmets. The study utilized the Agile method, involving observation, needs identification, system design, system development, implementation, and testing. The results of this study show that the developed system can provide ease for OSH officers in conducting supervision, thereby reducing the risk of work accidents.

Keywords: *Artificial Intelligence, Internet of Things, CNN, YOLO, Agile, Project Helm*

1. PENDAHULUAN

Seiring dengan pesatnya laju perkembangan pembangunan konstruksi gedung bertingkat di Indonesia, peranan pengendalian risiko kecelakaan

kerja semakin penting. Namun, penerapan Keselamatan dan Kesehatan Kerja (K3) sering kali terabaikan, yang ditunjukkan oleh tingginya angka kecelakaan kerja. Teori Keselamatan dan Kesehatan Kerja (K3) menekankan pentingnya identifikasi dan

mitigasi risiko di lingkungan kerja untuk melindungi pekerja dari potensi bahaya. Menurut Kesai, sektor konstruksi dan manufaktur menyumbang persentase kecelakaan tertinggi sebesar 32%, diikuti sektor transportasi 9%, kehutanan 4%, dan pertambangan 2% (Alfiansah dkk., 2020).

BPJS Ketenagakerjaan melaporkan peningkatan 5% kecelakaan kerja di Indonesia pada tahun 2021 dibandingkan dengan tahun 2020. Data Kementerian Ketenagakerjaan Indonesia tahun 2020 menunjukkan bahwa 57,5% dari total 126,51 juta pekerja memiliki tingkat pendidikan rendah (BPJS Ketenagakerjaan, 2022). Hal ini mempengaruhi rendahnya kesadaran akan pentingnya Keselamatan dan Kesehatan Kerja (K3) dan penggunaan Alat Pelindung Diri (APD) yang memenuhi standar. APD, seperti helm proyek, dirancang untuk melindungi pekerja dari risiko kecelakaan dengan melindungi kepala dari benda jatuh dan menyerap kejutan (Waruwu & Yuamita, 2016).

Namun, pemantauan APD masih banyak dilakukan secara manual, yang dianggap tidak efisien. Metode otomatisasi adalah solusi kunci untuk memantau kepatuhan penggunaan APD, mengurangi beban kerja supervisor, dan meningkatkan efisiensi. Sistem deteksi objek menggunakan kamera pengawas telah diusulkan untuk memantau area di mana penggunaan APD diwajibkan (Saliha dkk., 2018).

Seiring dengan perkembangan teknologi, *Artificial Intelligence* (AI) telah diterapkan untuk mendeteksi penggunaan APD secara otomatis. Beberapa penelitian sebelumnya telah mengembangkan sistem berbasis AI untuk mendeteksi helm proyek, namun masih memiliki kelemahan seperti keterbatasan dalam kecepatan deteksi dan akurasi (Widodo dkk., 2021).

Artificial Intelligence (AI)/Kecerdasan buatan didefinisikan sebagai suatu mesin atau program yang memiliki kecerdasan di dalamnya guna menyelesaikan suatu pekerjaan. Berbagai penerapan teknologi AI yaitu pengembangan sistem pakar, robotic, pengenalan suara, computer vision, artificial neural network dan sistem fuzzy (Ghosh dkk., 2018).

Penelitian ini bertujuan untuk mengembangkan sistem deteksi objek *Safety Helmet* secara *real-time* menggunakan metode *Convolutional Neural Network* (CNN) dan *You Only Look Once* (YOLO). CNN YOLO adalah salah satu teknik *Deep Learning* (DL) yang terbukti efisien dalam mendeteksi objek dalam citra dengan kecepatan dan akurasi tinggi (Zhang dkk., 2020). Sistem ini terdiri dari tiga tahap utama: *preprocessing* untuk meresize dan melakukan anotasi *labeling* pada dataset citra, training menggunakan metode transfer learning YOLOv3, dan deteksi objek (Wang, 2020)

Tujuan dari sistem yang diusulkan dalam penelitian ini adalah peningkatan akurasi dan efisiensi deteksi dibandingkan dengan penelitian sebelumnya. Sistem ini tidak hanya mendeteksi penggunaan helm proyek tetapi juga mengurangi

keterlibatan manusia dalam proses pengawasan, sehingga lebih efektif dalam meningkatkan keselamatan kerja di industri konstruksi.

2. KAJIAN PUSTAKA

2.1 Landasan Penelitian yang Relevan

Bagian ini secara umum membahas hasil-hasil penelitian sebelumnya yang menjadi sumber pustaka untuk penelitian ini. Dalam Tabel 1, disajikan beberapa penelitian terkait yang telah dilakukan sebelumnya.

Tabel 1. Penelitian yang Relevan

No	Nama Penulis (Tahun), Judul	Deskripsi Singkat	Perbedaan Penelitian
1	Hu, J., Gao, X., Wu, H., & Gao, S. (2019). Detection of Workers Without the Helments ini Videos Based on YOLO V3.	Penelitian ini menggunakan YOLO v3 sebagai algoritma utama untuk menggabungkan fitur pada level dan skala yang berbeda guna memprediksi lokasi dan kategori objek. Selanjutnya, dataset diperluas agar model dapat mengumpulkan lebih banyak data tentang objek, yang kemudian diintegrasikan ke dalam lokasi konstruksi sebagai program komputer yang dapat mendeteksi waktu serta memiliki fungsi alarm. Dataset ini memastikan peningkatan akurasi dan tingkat deteksi.	- Penelitian ini tidak menggunakan Raspberry Pi, Webcam dan Modem 4G. -Hasil <i>inferencing</i> tidak ditampilkan pada sebuah halaman web menggunakan API dan tidak dikirimkan ke cloud/database .
2	Wang, X., Niu, D., Luo, P., Zhu, C., Ding, L., & Huang, K. (2020). A Safety Helmet and Protective Clothing Detection Method based on Improved-Yolo v 3.	Penelitian ini menggunakan metode YOLO v3 yang telah ditingkatkan untuk mendeteksi helm pengaman dan pakaian pelindung dengan menggunakan video pengawasan dari lokasi konstruksi sebagai dataset utama. Jaringan YOLOv3 ditingkatkan melalui modifikasi struktur lapisan konvolusi residual.	- Penelitian ini tidak menggunakan rangkaian perangkat keras Raspberry Pi, Webcam dan Modem 4G. -Hasil <i>inferencing</i> tidak ditampilkan pada sebuah halaman web menggunakan API dan tidak dikirimkan ke cloud/database

2.2. CNN – YOLO

You Only Look Once (YOLO) adalah suatu algoritma yang dikembangkan pertama kali oleh Joseph Redmon pada tahun 2015 untuk deteksi objek berbasis *Deep Learning*. YOLO V3 adalah perkembangan dari algoritma YOLO dengan menggunakan Darknet-53 sebagai ekstraktor fitur, yang terdiri dari 53 lapisan konvolusi. Ini berbeda dengan Darknet-19 yang digunakan dalam YOLO v2, yang hanya memiliki 19 lapisan konvolusi. (Manajang dkk., 2020).

CNN YOLO adalah sebuah jenis arsitektur jaringan saraf yang digunakan untuk deteksi objek dalam *computer vision tasks*. CNN YOLO dirancang untuk mendeteksi objek dalam sebuah gambar dengan membagi gambar ke dalam grid.

Keunggulan utama CNN YOLO adalah kecepatan dan efisiensinya. YOLO mampu melakukan deteksi objek secara cepat dan efisien. Dengan memproses seluruh citra hanya sekali, YOLO menghindari *overhead* yang terkait dengan metode deteksi berbasis geser-geser (*sliding window*) atau jaringan pemantauan berulang (*repeated monitoring networks*). Ini membuat YOLO sangat cocok untuk aplikasi *real-time*, seperti deteksi objek pada video atau kamera bergerak. YOLO dapat dioptimalkan untuk memanfaatkan pemrosesan paralel pada perangkat keras seperti GPU, yang dapat meningkatkan kinerja deteksi dan memungkinkan penggunaan dalam aplikasi yang lebih kompleks (Nguyen dkk., 2019).

2.3. Raspberry Pi 4

Raspberry Pi merupakan sebuah komputer papan tunggal (*single-board computer*) yang memiliki ukuran seukuran kartu kredit. Fungsinya meliputi menjalankan program perkantoran, permainan komputer, dan berperan sebagai pemutar media untuk video beresolusi tinggi. Python digunakan sebagai bahasa pemrograman utama pada Raspberry Pi, meskipun ada kemungkinan untuk memilih bahasa pemrograman lain. Raspberry Pi dilengkapi dengan komponen serupa seperti komputer konvensional, termasuk CPU, GPU, RAM, Port USB, Audio Jack, HDMI, Ethernet, dan GPIO. Sebagai tempat penyimpanan data dan sistem operasi, Raspberry Pi mengandalkan kartu Micro SD dengan kapasitas minimal 4 GB, sedangkan daya diperoleh melalui micro USB power dengan rekomendasi sumber daya sebesar 5V dan arus minimal 700 mA. Raspberry Pi juga dilengkapi dengan port untuk display seperti TV atau monitor, serta koneksi USB untuk keyboard dan mouse (Hermawan dkk., 2021).

2.4. Kamera CCTV

CCTV atau *Closed Circuit Television* pertama kali ditemukan oleh Walter Brunch, CCTV tersebut digunakan pada tahun 1942 oleh tim pelaksana untuk peluncuran roket V-2. Pada awalnya, perusahaan Siemens AG memproduksi CCTV ini

untuk memonitor proses peluncuran roket V-2 dan memastikan bahwa semuanya berjalan dengan baik (Salawazo dkk., 2019).

CCTV sendiri adalah alat perekam yang menggunakan satu atau lebih kamera video untuk menghasilkan data video dan audio. Sebagai kamera video digital, CCTV mengirimkan sinyal gambar dari suatu ruangan atau area penyiaran ke suatu tujuan tertentu yang dapat dipantau melalui perangkat monitor (Yolanda dkk., 2021).

2.5. OpenCV

OpenCV (*Open Source Computer Vision Library*) adalah pustaka sumber terbuka yang digunakan untuk pengembangan computer vision dan *machine learning*. OpenCV menyediakan infrastruktur yang mempermudah pengembangan dan penyesuaian kode aplikasi computer vision. Dengan lebih dari 2500 algoritma, OpenCV mencakup beragam fungsi computer vision dan machine learning, termasuk deteksi dan pengenalan wajah, identifikasi objek, pelacakan pergerakan, dan banyak lagi. Pustaka ini mendukung antarmuka pemrograman dalam bahasa C, C++, Python, dan Java, serta dapat dijalankan pada berbagai sistem operasi, seperti Windows, Linux, Android, MacOS, dan iOS (Muharram & Suryadi, 2022).

2.6. Application Programming Interface (API)

Antarmuka Pemrograman Aplikasi (API) merupakan konsep fungsi yang menyediakan antarmuka untuk aplikasi, memungkinkan akses dan pemanfaatan aplikasi oleh pihak lain tanpa perlu mengubah struktur kode utama atau database sistem. Hal ini juga memfasilitasi komunikasi antar sistem, bahkan jika beroperasi pada platform yang berbeda. *Web Service*, sebagai salah satu jenis API, berperan dalam memberikan akses pengguna untuk proses pengambilan data. Menggunakan arsitektur *Representational State Transfer* (ReST) yang beroperasi melalui *Hypertext Transfer Protocol* (HTTP), *Web Service* menyajikan data dalam format *Javascript Object Notation* (JSON) kepada pengguna saat mengakses API (Muri dkk., 2019).

2.7. SIM card

SIM card atau disebut juga Kartu SIM (*Subscriber Identity Module*) adalah sebuah kartu pintar (*SmartCard*) seukuran peranko yang benamkan pada telepon genggam serta menyimpan pengenal jasa penyedia telekomunikasi. SIM card menyimpan informasi yang berkaitan dengan jaringan yang digunakan untuk *authentication* dan identifikasi pengguna. Data yang paling penting adalah nomor identitas kartu (ICCID, *Integrated Circuit Card ID*), nomor pengguna internasional (IMSI, *International Mobile Subscriber Identity*), kunci otentikasi (Ki, *Authentication Key*), kode area (LAI, *Local Area Identity*), dan nomor panggilan darurat operator. SIM

card juga menyimpan nomor layanan pusat untuk SMS (SMSC, *Short Message Service Center*), nama penyedia layanan (SPN, *Service Provider Nama*). Ketika SIM card tersebut berorientasi sebagai smartcard, maka memuka kemungkinan keamanan yang beresonansi jauh melampaui dunia yang bersifat mobile (Anwar dkk., 2016).

3. METODE PENELITIAN

3.1. Observasi

Pada tahap pertama dilakukan metode observasi, tahap ini dilakukan pencarian informasi dan data-data yang berhubungan dengan penelitian. Informasi dan data yang sudah terkumpul nantinya sebagai bahan mengerjakan penelitian. Informasi dan data bisa berupa pengamatan langsung maupun dari beberapa jurnal. Kemungkinan yang akan menjadi penghambat pengerjaan juga dapat dicari jalan keluarnya pada tahap ini.

3.2. Identifikasi Kebutuhan

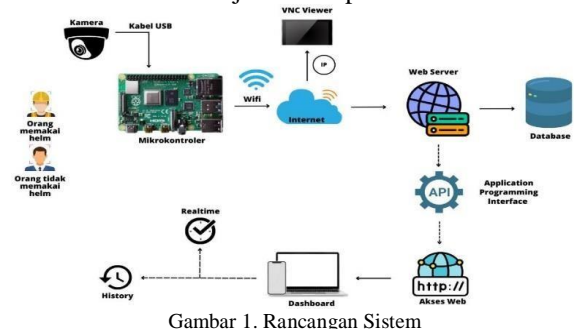
Tahap merekap komponen-komponen apa saja yang dibutuhkan dalam pembuatan alat. Untuk list komponen pembuatan penelitian ini dapat dilihat pada Tabel 2.

No	Komponen
1.	Raspberry Pi ver 4
2.	Camera Logitech 720P
3.	Modem 4G
4.	Memory Card 64GB
5.	Sim Card
6.	Dummy Camera

3.3. Perancangan

Dalam perancangan sistem pemantauan penggunaan helm proyek berbasis *Artificial Intelligence* (AI) dan *Internet of Things* (IoT), diperlukan perangkat seperti *Raspberry Pi* dan *cloud* untuk mengirimkan data ke *dashboard*. *Raspberry Pi* berperan sebagai *microcontroller*, sementara *cloud* digunakan untuk menyimpan data atau sebagai *database*. Berdasarkan rancangan sistem ini, kamera menangkap gambar, kemudian *Raspberry Pi* mendeteksi apakah pengguna mengenakan helm atau tidak. Hasil deteksi tersebut kemudian diolah oleh *Raspberry Pi*. Hasil inferensi dikirimkan ke *cloud* atau *database*. Data deteksi yang sudah disimpan di *cloud* akan dikirimkan ke *dashboard* menggunakan API, namun melalui *web server* terlebih dahulu untuk keamanan data. Hasil yang dikirimkan akan ditampilkan pada bagian *widget realtime* di *dashboard*. Setelah 10 detik, hasil deteksi yang tampil pada *widget realtime* akan masuk ke bagian *history* di *dashboard*, di mana pengguna dapat melihat hasil deteksi sebelumnya. Rancangan sistem secara

keseluruhan ditunjukkan pada Gambar 1.

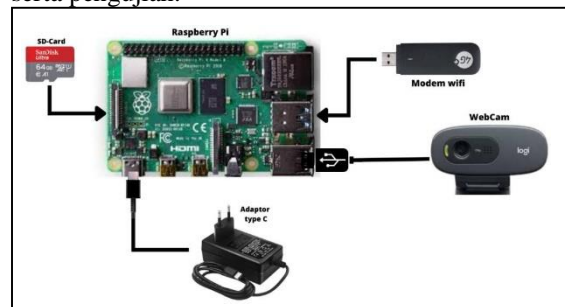


Gambar 1. Rancangan Sistem

3.3.1. Perancangan Perangkat Keras

Gambar 2 menunjukkan desain fisik perangkat keras. Ilustrasi ini menggambarkan bahwa kamera akan aktif dan mendeteksi objek manusia. Hasil deteksi tersebut kemudian akan diproses oleh *Raspberry Pi*, yang akan melakukan *inferensi* atau klasifikasi untuk menentukan apakah seseorang memakai helm atau tidak. Hasil dari *inferensi* ini kemudian dikirimkan ke *cloud* atau *database*.

Raspberry Pi, yang berfungsi sebagai *mikrokontroler*, akan dilengkapi dengan *webcam*, *modem wifi*, dan kartu SD, serta mendapatkan sumber daya dari adaptor *power supply*. *Raspberry Pi* juga akan terhubung ke PC, yang akan digunakan sebagai monitor untuk aktivitas pembuatan kode sistem dan dashboard, pengumpulan dataset, pelabelan dataset, serta pengujian.



Gambar 2. Gambaran desain fisik perangkat

3.3.2. Perancangan Perangkat Lunak

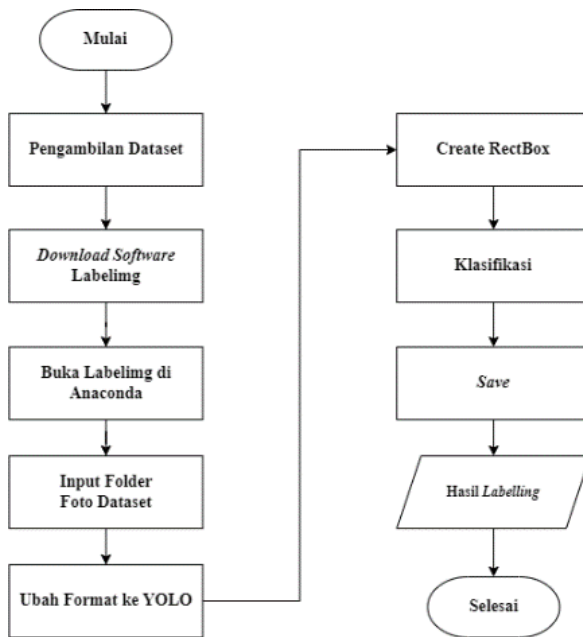
Perancangan perangkat lunak sistem dibagi menjadi beberapa program. Program pertama adalah program untuk mengolah data dengan menormalkan citra. Program kedua adalah pelatihan dataset menggunakan transfer *learning* dengan YOLOv3 untuk melatih model agar dapat mengklasifikasikan apakah seseorang memakai helm atau tidak. Program ketiga digunakan untuk menguji sistem yang telah dilatih sebelumnya. Program keempat adalah program yang mengirimkan hasil *inferensi* ke *cloud* atau *database*, yang kemudian ditampilkan pada halaman web.

3.3.3. Perancangan Data Training

Sebelum melakukan pelabelan dataset, perlu dilakukan pengumpulan dataset terlebih dahulu. Dataset diperoleh dengan mengambil foto langsung saat praktikum menggunakan helm proyek. Foto diambil dari sudut depan, belakang, atas, kanan, dan kiri. Jumlah dataset yang diperoleh adalah 1800 foto. Selain dari pengambilan foto langsung, dataset juga diperoleh dengan mengunduh foto dari website *Kaggle* dengan nama file *safety helmet detection* yang di sampling dan total akhir dataset ada 1900. Pelabelan dilakukan menggunakan software *LabelImg* pada *Anaconda*.

3.3.4. Perancangan Labelling Dataset

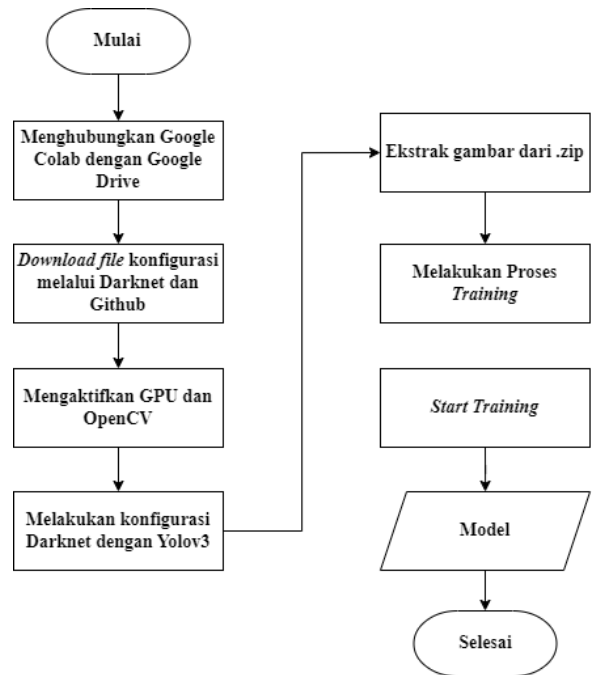
Labelling adalah proses memberikan label atau klasifikasi pada data, memudahkan pengorganisasian dan analisis. Dalam *machine learning*, labelling diperlukan untuk melatih model agar dapat mengenali pola dan membuat prediksi berdasarkan data yang terlabel. *Labelling* memfasilitasi proses pembelajaran model untuk meningkatkan performa dalam tugas-tugas seperti klasifikasi dan prediksi.



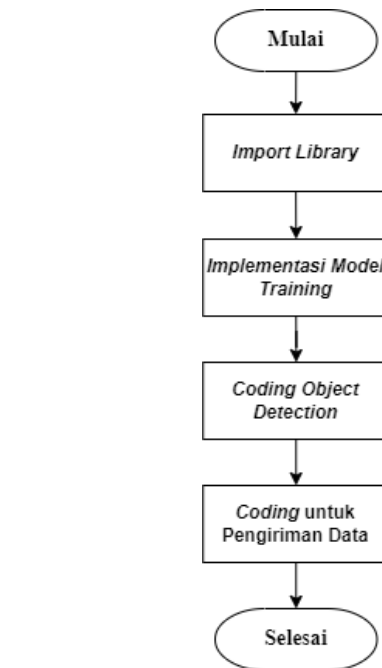
Gambar 3. Diagram alur labelling dataset

Pada Gambar 3 menunjukkan alur program pada tahap *labelling* dataset yang diawali dengan mengakses data *training* yang telah disiapkan, kemudian meng-*install* software *labelling* lalu membukanya di *anaconda*. Setelah itu, dilakukan input folder yang berisi foto dataset lalu mengubah formatnya ke *YOLO*. Selanjutnya dilakukan *Create ReactBox* untuk melabeli masing-masing data dan diklasifikasikan sesuai kelasnya.

3.3.5. Perancangan Training



Gambar 4. Diagram Alur Training



Gambar 5. Diagram Alur Pembuatan Program YOLO *Object Detection*

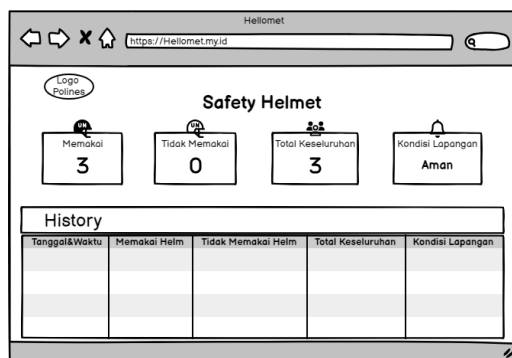
Gambar 4 merupakan diagram alur proses *training*, dimana hasil dari foto yang selesai di *labelling* selanjutnya akan di *training* untuk melatih atau membangun model. *Training* dilakukan menggunakan algoritma *YOLO*. Software yang digunakan untuk *training* adalah *darknet*. Hasil dari *training* dataset adalah model yang akan di implementasikan pada coding *YOLO object detection*.

3.3.6. Perancangan Program YOLO Object Detection

Diagram alir pembuatan Program YOLO *Object Detection* dapat dilihat pada Gambar 5. Pembuatan deteksi objek dilakukan menggunakan aplikasi *PyCharm*. Kode tersebut berisi *library* (OpenCV dan Numpy), model hasil pelatihan, URL API, *source code* untuk frame kamera, *source code* untuk deteksi objek, *source code* untuk perhitungan tingkat kepercayaan/akurasi, serta *source code* untuk pengiriman data.

3.3.7. Perancangan Mockup Dashboard

Tampilan *mockup dashboard* Rancang Bangun Alat Pendeteksi Penggunaan Helm Proyek Berbasis *Artificial Intelligence – Internet of Things* dapat dilihat pada Gambar 6.



Gambar 6. Mockup Dashboard

Gambar 6 merupakan *mockup dashboard* yang akan digunakan untuk menampilkan hasil dari sistem. *Front-End dashboard* menggunakan *framework* *tailwindCSS* dan *reactJS*. Untuk icon – icon yang menggunakan *heroiconweb*.

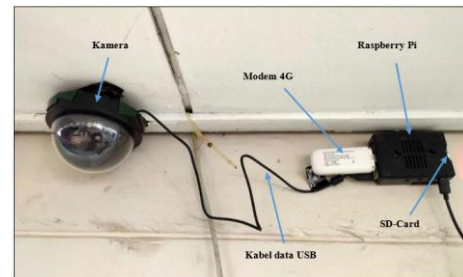
Dashboard memiliki 2 bagian, yaitu *widget* dan tabel. Pada bagian *widget* dibagi menjadi 4 yaitu memakai helm, tidak memakai helm, total keseluruhan dan kondisi lapangan. Fungsinya untuk menampilkan hasil *realtime*. Bagian *widget* kondisi lapangan akan muncul kata aman, peringatan dan tidak aman. Dinyatakan aman, apabila semua orang memakai helm. Dinyatakan peringatan, apabila 1 orang tidak memakai helm serta dinyatakan tidak aman, apabila semua orang tidak menggunakan helm. Pada bagian tabel berfungsi untuk menampilkan data history dari hasil *realtime*. Jarak waktu dari hasil *realtime* ke *history* memiliki *delay* selama per 10 detik.

3.4. Implementasi

Setelah tahap perancangan, kemudian akan dilakukan proses implementasi sistem. Proses implementasi dibagi menjadi dua yaitu penyusunan konfigurasi *hardware* dan pembangunan sistem.

3.4.1. Penyusunan Konfigurasi Hardware

Pada tahap ini Raspberry Pi, Webcam dan Modem 4G akan disusun menjadi suatu rangkaian perangkat keras. Proses penyusunan perangkat disesuaikan dengan rancangan pada tahap rancangan sistem. Pada Raspberry Pi akan dilakukan pemrograman pada *command prompt* agar dapat bekerja sesuai dengan fungsi yang diharapkan. Implementasi konfigurasi *Hardware* ditunjukkan pada Gambar 7.



Gambar 7. Implementasi Sistem

3.4.2. Pembangunan Sistem

Pada tahap pembangunan sistem akan mencakup pembangunan sistem *dashboard* sebagai media monitoring yang akan berkaitan dengan database. Selain itu, akan diimplementasikan API pada proses komunikasi data dengan perangkat keras sesuai dengan rancangan pada tahapan perancangan perangkat lunak sistem. Perangkat keras yang berperan sebagai mikrokontroler untuk pendeteksi akan diletakkan di bengkel sipil Politeknik Negeri Semarang bagian atas bengkel.

4. PENGUJIAN DAN ANALISIS

4.1. Pengujian Sistem dan Dashboard Hari Pertama

Pengujian sistem dan dashboard pertama dilakukan pada hari Selasa, 13 Juni 2023 pukul 14.00 - 17.00 WIB di bengkel sipil Politeknik Negeri Semarang. Sebelum melakukan pemasangan alat dilakukan terlebih dahulu memasang *webcam* ke *dummy CCTV*, setting IP wifi, dan menyambungkan Raspberry Pi ke *VNC Viewer* menggunakan IP. Fungsi *VNC Viewer* sendiri yaitu untuk meremote sistem. Pada pengujian hari pertama koneksi internet menggunakan modem 4G. Dalam melakukan pemasangan alat terlebih dahulu menentukan letak posisi kamera agar mendapatkan *coverage* yang luas sehingga mahasiswa yang melakukan praktikum terdeteksi seluruhnya.

Pada pengujian sistem di hari pertama mendapatkan akurasi yang belum sesuai dan sistem bekerja kurang maksimal sehingga menyebabkan tidak semua mahasiswa terdeteksi. Rata-rata *delay* pada pengujian hari pertama adalah 7 detik. Hasil deteksi dapat dilihat pada Gambar 8.



Gambar 8. Hasil Pengujian Hari Pertama

Untuk pengujian *dashboard* sudah berjalan dengan baik. Pada bagian *dashboard* widget realtime sudah bisa menerima data hasil deteksi dengan delay 3-5 detik. Pada bagian *dashboard* tabel *history* setiap 10 detik akan menerima data yang masuk pada bagian widget realtime sebelumnya. Hasil pengujian *dashboard* hari pertama dapat dilihat pada Gambar 9.



Gambar 9. Hasil Pengujian Dashboard Hari Pertama

4.2. Pengujian Sistem dan Dashboard Hari Kedua

Pengujian sistem dan dashboard kedua dilakukan pada hari Kamis, 15 Juni 2023 pukul 08.30 - 14.00 WIB di bengkel sipil Politeknik Negeri Semarang. Pada pengujian hari kedua kamera dipindahkan letaknya dikarenakan praktikum dilakukan di posisi yang berbeda dari hari pertama. Pada pengujian hari kedua koneksi internet menggunakan wifi dari *handphone*. *Delay* pada pengujian hari kedua lebih sedikit dibandingkan hari pertama yaitu 4 detik. Hasil deteksi dapat dilihat pada Gambar 10.



Gambar 10. Hasil Deteksi Hari Kedua

Pada pengujian hari kedua, mahasiswa selaku objek yang akan di deteksi melakukan praktikum membuat kerangka menggunakan besi beton yang cukup tinggi sehingga menutupi bagian kepala beberapa mahasiswa sehingga ada yang tidak terdeteksi sebagian. Untuk pengujian *dashboard* hasilnya lebih baik dari hari pertama. Pada bagian *dashboard* widget realtime sudah bisa menerima data hasil deteksi dengan *delay* yang lebih kecil yaitu 2-4 detik. Pada bagian *dashboard* widget history setiap 10 detik akan menerima data yang masuk pada bagian widget realtime sebelumnya. Hasil pengujian *dashboard* hari kedua dapat dilihat pada Gambar 11.



Gambar 11. Hasil Pengujian Dashboard Hari Kedua

Rumus perhitungan akurasi berdasarkan jumlah dataset adalah sebagai berikut.

$$\text{Akurasi} = \frac{\text{Jumlah prediksi benar}}{\text{Total jumlah dataset}} \times 100 \%$$

Keterangan:

Jumlah prediksi benar = Prediksi jumlah dataset yang sesuai dengan klasifikasi
 Total jumlah dataset = Jumlah total dataset.
 Berdasarkan hasil pengujian yang dilakukan, berikut rata-rata akurasi dapat dilihat pada tabel 3. untuk pengujian hari pertama dan tabel 4. untuk pengujian hari kedua.

Tabel 3. Rata-Rata Akurasi Pengujian Hari Pertama

No.	Skenario	Akurasi Memakai Helm	Akurasi Tidak Memakai Helm
1.	3 orang memakai helm, 1 orang tidak memakai helm	71%	68%
		76%	
		81%	
2.	4 orang memakai helm, 1 orang tidak memakai helm	76%	73%
		76%	
		79%	
		80%	
3.	2 orang memakai helm, 2 orang tidak memakai helm	77%	70%
		78%	68%
4.	3 orang memakai helm, 2 orang tidak memakai helm	76%	72%
		77%	71%
		72%	
Rata-Rata		76,6%	70,3%

Tabel 4. Rata-Rata Akurasi Pengujian Hari Kedua

No.	Skenario	Akurasi Memakai Helm	Akurasi Tidak Memakai Helm
1.	3 orang memakai helm, 1 orang tidak memakai helm	94%	72%
		96%	
		90%	
2.	4 orang memakai helm, 1 orang tidak memakai helm	90%	73%
		93%	
		92%	
3.	2 orang memakai helm, 2 orang tidak memakai helm	91%	72%
		88%	
		77%	
4.	3 orang memakai helm, 2 orang tidak memakai helm	91%	79%
		99%	
		96%	
	Rata-Rata	92,7%	72,65%

Jadi, rata-rata akurasi pengujian hari pertama dan hari kedua adalah sebagai berikut.

$$\text{Memakai Helm} = \frac{76,6\% + 92,7\%}{2} = 84,65\%$$

$$\text{Tidak Memakai Helm} = \frac{70,3\% + 72,65\%}{2} = 71,5\%$$

Variasi dan jumlah dataset mempengaruhi tingkat akurasi. Algoritma YOLO mendeteksi bentuk sehingga warna helm apa saja dapat terdeteksi oleh sistem apabila memiliki bentuk yang sama seperti dataset yang di training. Dataset yang di training memiliki bentuk helm proyek yang bervariasi sehingga semua warna helm dapat terdeteksi. Untuk pendeteksian warna, sistem sudah bisa mendeteksi 5 warna helm proyek berdasarkan pengujian yang dilakukan yaitu putih, biru, kuning, merah dan hijau.

4.3. Analisis Hasil

Berdasarkan hasil pengujian sistem dan *dashboard* yang sudah dilakukan, ada beberapa faktor yang memengaruhi hasil deteksi, akurasi, dan *delay* sebagai berikut :

4.3.1. Faktor Koneksi Internet

Delay yang terdapat pada hasil deteksi *Raspberry Pi* juga disebabkan karena koneksi internet. Pada pengujian hari pertama menggunakan modem 4G koneksi internetnya kurang baik sehingga menyebabkan *delay* yang cukup banyak yaitu 4- 6 detik. Pada pengujian hari kedua mengganti koneksi internet menggunakan wifi handphone dan didapatkan hasil deteksi dan akurasi yang baik. *Delay* hasil deteksi hari kedua pada *Raspberry Pi* lebih sedikit yaitu 3-5 detik.

Selain memengaruhi *delay* pada hasil deteksi, koneksi internet juga berpengaruh pada kecepatan data hasil deteksi yang masuk ke *dashboard*. Pada hari pertama *delay* data yang masuk ke *dashboard* bagian *widget realtime* yaitu 3-5 detik dan pada hari kedua *delay* menjadi 2-4 detik.

4.3.2. Faktor Intensitas Cahaya

Pada pengujian hari pertama didapatkan akurasi dan hasil deteksi yang kurang maksimal dibandingkan hari kedua karena cahaya dari luar dipantulkan melalui jendela ruangan sehingga menyebabkan silau dan kamera blur.

4.3.3. Faktor Coverage

Coverage juga memengaruhi hasil deteksi. Semakin jauh jarak antara kamera dengan objek, maka ada beberapa objek yang tidak terdeteksi oleh kamera. Jarak maksimum objek terdeteksi oleh kamera sekitar 2,5-3 meter. Spesifikasi kamera juga memengaruhi hasil deteksi karena pada penelitian ini menggunakan webcam Logitech yang memiliki spesifikasi belum terlalu bagus sehingga jika *coverage* nya terlalu jauh, kamera belum bisa mendeteksi.

4.3.4 Faktor Mikrokontroler

Menggunakan mikrokontroler *Raspberry Pi* ver 4 dengan ram 4 GB. Performa *Raspberry Pi* dengan ram 4GB belum terlalu bagus untuk menjalankan sistem ini sehingga *delay* hasil deteksi masih cukup besar.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian terhadap alat pendeteksi penggunaan helm proyek berbasis *artificial intelligence* dan *internet of things*, dapat disimpulkan bahwa monitoring penggunaan helm oleh mahasiswa yang melakukan praktik di bengkel sipil berbasis teknologi ini dapat berjalan secara efektif dan efisien karena hasilnya dapat langsung dilihat pada *dashboard*. Akurasi alat ini berdasarkan pengujian menunjukkan rata-rata 84,65% untuk deteksi pemakaian helm dan 71,5% untuk deteksi ketidakpemakaian helm.

Saran untuk penelitian selanjutnya adalah meningkatkan performa sistem dan meminimalisir *delay* dengan mengganti *microcontroller* yang memiliki spesifikasi lebih baik daripada *Raspberry Pi* versi 4. Disarankan juga menggunakan kamera dengan spesifikasi lebih baik agar cakupan area yang lebih luas dapat terdeteksi dengan jelas. Menambahkan tombol on/off pada *dashboard* juga akan memudahkan pengguna dalam menghidupkan alat tanpa perlu melakukan coding melalui terminal. Selain itu, untuk meningkatkan efektivitas peringatan, disarankan untuk menambahkan *buzzer* pada alat.

DAFTAR PUSTAKA

ALFIANSYAH, Y., KURNIAWAN, B. and EKAWATI, E., 2020. Analisis Upaya Manajemen K3 Dalam Pencegahan dan

- Pengendalian Kecelakaan Kerja Pada Proyek Konstruksi PT.X Semarang. *Jurnal Kesehatan Masyarakat*.
- ANWAR, N., RIALDI, I., & LUTHFI, A. (2016). Analisis SIM Card Cloning Terhadap Algoritma Random Number Generator. *Jurnal Buana Informatika*, 7(2). <https://doi.org/10.24002/jbi.v7i2.493>
- GHOSH, A., CHAKRABORTY, D., & LAW, A. 2018. Artificial intelligence in Internet of things. Dalam *CAAI Transactions on Intelligence Technology* (Vol. 3, Nomor4). <https://doi.org/10.1049/trit.2018.1008>
- HERMAWAN, I., ARNALDY, D., AGUSTIN, M., WIDYONO, M.F., NATHANAEL, D. and MULYANI, M.T., 2021. Sistem Pengenalan Benih Padi Menggunakan Metodelight Convolutional Neural Networkpada Raspberry Pi 4B. *Jurnal Teknologi Terpadu*.
- MANANJANG, D.J.P., SOMPIE, S.R.U. and JACOBUS, A., 2020. Implementasi Framework Tensorflow Object Detection API Dalam Mengklasifikasi Jenis Kendaraan Bermotor. *Jurnal Teknik Informatika*.
- MUHARRAM, R.F. and SURYADI, A., 2022. Implementasi artificial intelligence untuk deteksi masker secara realtime dengan tensorflow dan ssdmobilenet Berbasis python. *Jurnal Widya*.
- MURI, M.F.A., UTOMO, H.S. and SAYYIDATI, R., 2019. Application Programming Interface. *Jurnal Sains dan Informatika*.
- NGUYEN, D. T., NGUYEN, T. N., KIM, H., & LEE, H. J., 2019. A high-throughput andpower-efficient fpga implementation of yolo cnn for object detection. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 27(8). <https://doi.org/10.1109/TVLSI.2019.2905242>
- SALAWAZO, V.M.P., GEA, D.P.J., GEA, R.F. and AZMI, F., 2019. Implementasi Metode Convolutional Neural Network (CNN) pada Penegalan Objek Video CCTV. Indonesia Jalan Sekip Sikambang.
- SALIHA, J., JOSEPH, W.B.S. and KALESARAN, A.F.C., 2018. Hubungan antara Pengetahuan dan Sikap dengan Kepatuhan Penggunaan Alat Pelindung Diri pada Pekerja PT. Utama Karya Proyek Pembangunan Jalan Tol Manado-Bitung Tahun 2018. *KESMAS: Jurnal Kesehatan Masyarakat Universitas Sam Ratulangi*.
- WANG, X. N. (2020). A Savety Helmet and Protective Clothing Detection Method based on Improved-Yole v3. *Chinese Automation Congress, CAC 2020*. <https://doi.org/10.1109/CAC51589.2020.00076>.
- WARUWU, S. and YUAMITA, F., 2016. Analisis faktor kesehatan dan keselamatan kerja (K3) yang signifikan mempengaruhi kecelakaan kerja pada proyek pembangunan apartement student castle. *Spektrum Industri*.
- WIDODO, B., ARMANTO, H. and SETYATI, E., 2021. Deteksi Pemakaian Helm Proyek Dengan Metode Convolutional Neural Network. *Journal of Intelligent System and Computation*.
- YOLANDA, D., HERSYAH, M.H. and MAROZI, E., 2021. Implementasi Metode Unsupervised Learning Pada Sistem Keamanan Dengan Optimalisasi Penyimpanan Kamera IP. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*.
- ZHANG, W., YANG, C.F., JIANG, F., GAO, X.Z. and ZHANG, X., 2020. Safety Helmet Wearing Detection Based on Image Processing and Deep Learning. *Proceedings - 2020 International Conference on Communications, Information System and Computer Engineering, CISCE 2020*.

Halaman ini sengaja dikosongkan