

BAB 2

TINJAUAN PUSTAKA

2.1 Landasan Teori

Landasan teori merupakan kerangka konseptual yang menopang seluruh rancangan dan implementasi penelitian ini. Bagian ini menyajikan tinjauan pustaka dan konsep-konsep dasar yang relevan dengan topik penelitian, mulai dari pengenalan objek studi hingga teknologi yang digunakan, guna memastikan dasar ilmiah yang kuat untuk pengembangan system.

2.1.1 PT Bumi Siak Pusako

PT. Bumi Siak Pusako (BSP) adalah perusahaan yang bergerak di bidang eksplorasi dan produksi minyak dan gas bumi, yang berpusat di Kabupaten Siak, Provinsi Riau. Sebagai perusahaan yang beroperasi di lingkungan berisiko tinggi, BSP sangat memperhatikan aspek keselamatan kerja untuk melindungi karyawannya dan memastikan kelangsungan operasional.

Proses pengawasan keselamatan kerja di BSP saat ini dilakukan melalui pemantauan menggunakan CCTV yang telah terpasang di area kerja. Operator bertugas untuk memantau karyawan secara langsung melalui rekaman CCTV guna memastikan bahwa mereka mengenakan Alat Pelindung Diri (APD) sesuai dengan standar keselamatan yang ditetapkan. Namun, metode ini seringkali menyulitkan karena operator tidak dapat dengan mudah mengidentifikasi karyawan yang tidak menggunakan APD secara lengkap selama berada di lapangan.



Gambar 2.1 PT Bumi Siak Pusako

Sebagai bagian dari proses pengawasan, ketika sistem mendeteksi pelanggaran APD, operator akan memanggil karyawan yang melanggar untuk dicatat informasi dirinya. Setelah itu, data pelanggaran yang telah dicatat akan dikirimkan ke departemen terkait sesuai dengan karyawan yang bersangkutan. Data ini digunakan sebagai dasar untuk evaluasi dan tindak lanjut sesuai dengan kebijakan perusahaan.

2.1.2 Konsep dan Definisi Alat Pelindung Diri (APD)

Alat Pelindung Diri (APD) merupakan salah satu komponen penting dalam upaya keselamatan dan kesehatan kerja (K3). Menurut Peraturan Menteri Tenaga Kerja dan Transmigrasi Nomor PER.08/MEN/VII/2010 tentang Alat Pelindung Diri yang selanjutnya disingkat APD adalah suatu alat yang mempunyai kemampuan untuk melindungi seseorang yang fungsinya mengisolasi sebagian atau seluruh tubuh dari potensi bahaya di tempat kerja. Penggunaan APD bertujuan untuk meminimalkan risiko kecelakaan kerja dan penyakit akibat kerja yang dapat terjadi akibat paparan langsung terhadap lingkungan kerja yang berbahaya.

2.1.3 Standar APD di PT Bumi Siak Pusako

Di area CLTS (*Centralize Land Treatment Site*) dan area Pit (kolam tempat minyak) yang akan menjadi area deteksi penggunaan Alat Pelindung Diri (APD) seperti helm, baju keselamatan, dan sepatu keselamatan merupakan komponen penting dalam upaya keselamatan dan kesehatan kerja (K3) di PT Bumi Siak Pusako.

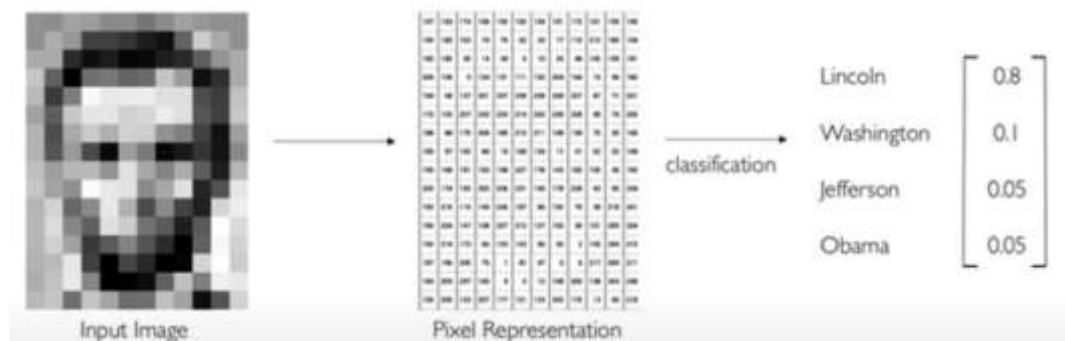


Gambar 2.2 APD Standar PT Bumi Siak Pusako

Helm keselamatan berfungsi melindungi kepala dari benturan benda keras atau material jatuh, terutama di area pengeboran dan pengangkutan. Baju keselamatan, yang terbuat dari bahan tahan api, melindungi tubuh dari paparan panas, api, atau bahan kimia berbahaya. Sementara itu, sepatu keselamatan dengan sol anti-slip dan pelindung kaki (*steel toe*) melindungi pekerja dari risiko tergelincir atau tertimpa benda berat di area kerja yang tidak rata.

2.1.4 Computer Vision

Computer vision adalah bidang ilmu komputer yang berfokus pada pengembangan algoritma dan sistem yang memungkinkan komputer untuk memahami dan memproses gambar atau video. Teknologi ini memiliki aplikasi luas, termasuk dalam deteksi objek, pengenalan wajah, dan analisis video. Dalam konteks keselamatan kerja, *computer vision* dapat digunakan untuk memantau kepatuhan terhadap penggunaan APD di tempat kerja, sehingga meningkatkan keselamatan pekerja (Diwan dkk., 2022; , Wicaksono dkk., 2021).



Gambar 2.3 Cara Kerja *Computer Vision*

2.1.5 Deep Learning

Deep Learning adalah sub-bidang dari Kecerdasan Buatan (*Artificial Intelligence/AI*) yang berfokus pada pengembangan algoritma yang dapat belajar dari data. Pendekatan ini menggunakan jaringan syaraf tiruan (*Neural Networks*) dengan banyak lapisan (*deep layers*) untuk memproses data dalam jumlah besar dan kompleks. Dalam konteks deteksi objek, teknik *deep learning* seperti YOLO telah terbukti lebih efisien dibandingkan metode tradisional, terutama dalam hal akurasi dan kecepatan (Albalawi dkk., 2021; Diwan dkk., 2022). Penggunaan *deep learning* dalam sistem keselamatan kerja dapat meningkatkan efektivitas deteksi dan respons terhadap potensi bahaya (Wicaksono dkk., 2021; Arif dkk., 2022).

Salah satu arsitektur *deep learning* yang paling umum digunakan dalam pemrosesan gambar adalah *Convolutional Neural Networks* (CNN), yang digunakan dalam arsitektur YOLO untuk deteksi APD secara *real-time*. Selain CNN, terdapat *Recurrent Neural Networks* (RNN) yang lebih cocok untuk data sekuensial seperti teks dan suara, serta *Transformers* yang saat ini menjadi standar dalam pemrosesan bahasa alami (NLP) dan visi komputer.

Selain arsitektur utama tersebut, terdapat berbagai model *deep learning* yang dirancang untuk tugas khusus. *Generative Adversarial Networks* (GAN) digunakan untuk membuat data sintesis, sementara *Autoencoders* bermanfaat untuk kompresi dan deteksi anomali. *Capsule Networks* (CapsNet) dikembangkan untuk meningkatkan pemahaman spasial dalam pemrosesan gambar, sedangkan *Spiking Neural Networks* (SNN) meniru cara kerja otak manusia dalam pemrosesan sinyal. Selain itu, *Self-Organizing Maps* (SOM) sering digunakan dalam clustering dan visualisasi data.

Secara umum, tahapan dalam mengembangkan dan melatih model *deep learning* terdiri dari beberapa langkah berikut:

1) *Preprocessing* Data

Preprocessing data adalah tahap awal yang sangat penting dalam *Deep Learning*. Tahap ini meliputi normalisasi, augmentasi data, dan *encoding* kategorikal. *Preprocessing* data adalah tahap kritis yang mempengaruhi performa model. Menurut Shorten & Khoshgoftaar (2019), augmentasi data (*data augmentation*) adalah teknik yang sangat efektif untuk meningkatkan generalisasi model, terutama ketika dataset terbatas. Dalam konteks YOLO, *preprocessing* melibatkan *resizing* gambar, normalisasi piksel, dan augmentasi seperti *flipping* dan *cropping* (Bochkovskiy dkk., 2020).

2) Pelatihan Model

Pelatihan model melibatkan proses optimasi di mana model belajar dari data dengan meminimalkan fungsi *loss*. Proses ini biasanya dilakukan menggunakan algoritma optimasi seperti *Stochastic Gradient Descent* (SGD) atau Adam. Menurut Smith dkk. (2020), penggunaan *learning rate scheduling* dan teknik optimasi seperti AdamW dapat meningkatkan kecepatan konvergensi dan performa model. Dalam YOLO, pelatihan melibatkan penyesuaian bobot pada

jaringan CNN untuk meminimalkan kesalahan prediksi *bounding box* dan kelas objek (Jocher dkk., 2020).

3) Evaluasi Model

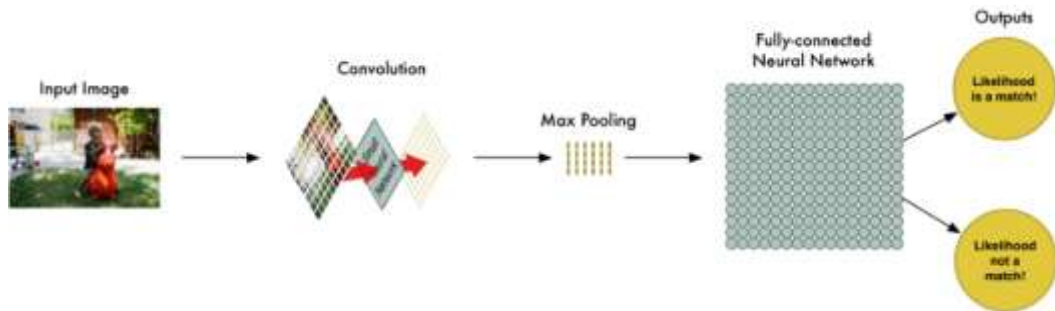
Evaluasi model dilakukan untuk mengukur performa model pada data yang belum pernah dilihat sebelumnya. Menurut Padilla dkk. (2021), matrik evaluasi yang umum digunakan termasuk akurasi, presisi, *recall*, dan *F1-score* untuk klasifikasi, serta *Intersection over Union* (IoU) untuk deteksi objek. YOLO menggunakan mAP sebagai metrik utama untuk mengevaluasi akurasi deteksi objek (Bochkovskiy dkk., 2020).

4) Fine-tuning dan Deployment

Fine-tuning adalah proses penyesuaian model yang telah dilatih pada dataset tertentu untuk meningkatkan performa pada dataset baru. Menurut Howard & Gugger (2020), *transfer learning* dan *fine-tuning* adalah teknik yang sangat efektif ketika dataset terbatas. Setelah model mencapai performa yang diinginkan, model dapat di-*deploy* ke lingkungan produksi menggunakan *framework* seperti TensorFlow Lite atau ONNX (Jocher dkk., 2020).

2.1.6 Convolutional Neural Network (CNN)

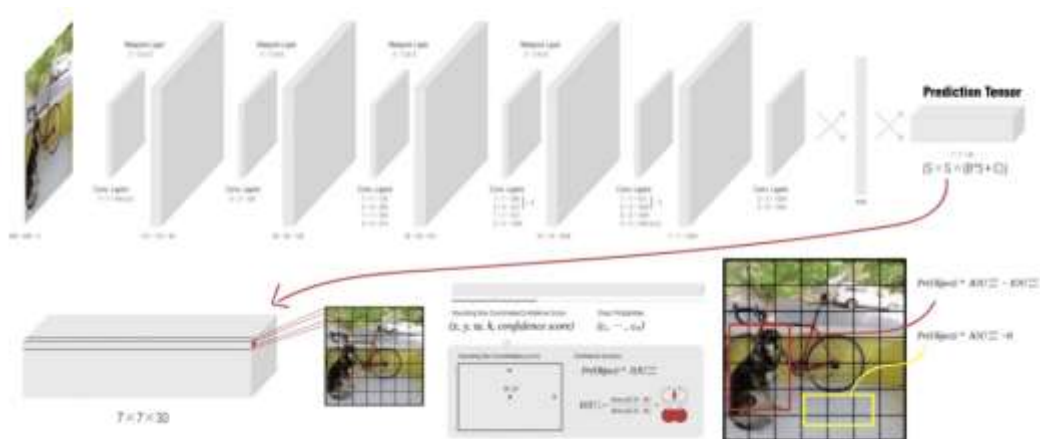
CNN adalah arsitektur *Deep Learning* yang dirancang khusus untuk memproses data yang memiliki struktur *grid*, seperti gambar. CNN menggunakan lapisan *convolution* untuk mengekstrak fitur dari gambar dan lapisan *pooling* untuk mengurangi dimensi data. Menurut Krizhevsky dkk. (2012), CNN telah mencapai performa yang luar biasa dalam kompetisi pengenalan gambar *ImageNet*, yang menjadi tonggak penting dalam perkembangan *Deep Learning*. CNN juga menjadi dasar bagi banyak metode deteksi objek, termasuk YOLO (*You Only Look Once*). YOLO adalah turunan dari CNN yang dirancang untuk melakukan deteksi objek secara *real-time* dengan menggabungkan proses deteksi dan klasifikasi dalam satu tahap (Redmon dkk., 2016).



Gambar 2.4 Cara Kerja CNN

2.1.7 You Only Look Once (YOLO)

You Only Look Once (YOLO) adalah arsitektur sistem berfokus pada deteksi objek yang terkenal karena kemampuannya untuk melakukan deteksi secara *real-time* dengan akurasi tinggi yang menggunakan arsitektur CNN sebagai dasarnya. Arsitektur ini bekerja dengan membagi gambar menjadi *grid* dan memprediksi *bounding boxes* serta probabilitas kelas untuk setiap *grid*. Penelitian menunjukkan bahwa YOLOv4 dan versi lebih baru dapat mencapai akurasi deteksi yang sangat baik dalam berbagai aplikasi, termasuk dalam konteks keselamatan kerja untuk mendeteksi penggunaan APD (Utami, 2024; , Husnan, 2023). Implementasi YOLO dalam sistem pemantauan dapat membantu mengidentifikasi pelanggaran penggunaan APD secara otomatis, yang berpotensi mengurangi risiko kecelakaan kerja (Yusup, 2024).



Gambar 2.5 Proses Arsitektur YOLO

Untuk memahami evolusi dan cara kerja arsitektur pendeteksi objek ini, berikut adalah rincian tahapan perkembangan dari algoritma YOLO:

1) YOLOv1

YOLOv1 adalah versi pertama dari arsitektur ini yang memperkenalkan pendekatan deteksi objek yang cepat dan efisien dengan satu jaringan tunggal. Meskipun demikian, YOLOv1 memiliki masalah dalam mendeteksi objek kecil dan akurasi yang lebih rendah dibandingkan dengan arsitektur lain pada saat itu. Penelitian oleh Redmon dkk. (2016) menunjukkan bahwa meskipun YOLOv1 dapat mendeteksi objek dengan kecepatan tinggi, akurasinya masih perlu ditingkatkan untuk aplikasi yang lebih kompleks.

2) YOLOv2 (YOLO9000)

YOLOv2, yang dikenal sebagai YOLO9000, meningkatkan akurasi deteksi dengan menggunakan teknik anchor boxes dan pelatihan pada dataset yang lebih besar, termasuk COCO dan ImageNet. Namun, versi ini masih mengalami kesulitan dalam mendeteksi objek kecil dan dalam kondisi pencahayaan yang buruk. Redmon & Farhadi (2017) mencatat bahwa YOLOv2 menunjukkan peningkatan signifikan dalam akurasi dibandingkan dengan versi sebelumnya, tetapi masih memiliki keterbatasan dalam situasi tertentu.

3) YOLOv3

YOLOv3 memperkenalkan arsitektur yang lebih dalam dengan penggunaan fitur dari beberapa skala, yang meningkatkan kemampuan deteksi objek kecil. Meskipun lebih akurat, YOLOv3 lebih lambat dibandingkan dengan versi sebelumnya dan memerlukan lebih banyak sumber daya komputasi. Redmon & Farhadi (2018) melaporkan bahwa YOLOv3 mencapai akurasi yang lebih baik dalam mendeteksi objek kecil dan kompleks, tetapi dengan *trade-off* dalam kecepatan.

4) YOLOv4

YOLOv4 mengoptimalkan kecepatan dan akurasi dengan menggunakan teknik augmentasi data dan arsitektur yang lebih efisien. YOLOv4 dapat mencapai kecepatan deteksi yang sangat tinggi dengan akurasi yang baik, meskipun memerlukan perangkat keras yang lebih kuat untuk mencapai performa optimal. Bochkovskiy dkk. (2020) menunjukkan bahwa YOLOv4 dapat mencapai mAP (*mean Average Precision*) yang tinggi dalam berbagai pengujian, menjadikannya pilihan yang sangat baik untuk aplikasi *real-time*.

5) YOLOv5

YOLOv5, meskipun bukan versi resmi dari pengembang asli YOLO, menawarkan kemudahan penggunaan dan implementasi yang lebih baik, serta kecepatan deteksi yang sangat cepat. Namun, versi ini masih dalam pengembangan dan tidak memiliki dukungan resmi dari pengembang YOLO asli. Glenn Jocher (2020) mengembangkan YOLOv5 dengan fokus pada kemudahan penggunaan dan efisiensi dalam pelatihan, menjadikannya populer di kalangan pengembang.

6) YOLOv6 dan YOLOv7

YOLOv6 dan YOLOv7 terus meningkatkan akurasi dan kecepatan deteksi lebih lanjut dengan arsitektur yang lebih canggih dan teknik pelatihan yang lebih baik. Meskipun memerlukan pemahaman yang lebih dalam tentang pengaturan dan parameter untuk mencapai hasil optimal, kedua versi ini menunjukkan performa yang sangat baik dalam berbagai aplikasi deteksi objek. Wang dkk. (2022) menunjukkan bahwa YOLOv7 dapat mencapai performa yang sangat baik dalam berbagai pengujian, menjadikannya salah satu arsitektur terdepan dalam deteksi objek.

7) YOLOv8

YOLOv8 menunjukkan peningkatan dalam hal akurasi dan kecepatan, serta kemampuan untuk mendeteksi objek dalam kondisi yang lebih menantang. Meskipun masih dalam tahap pengembangan dan mungkin memerlukan lebih banyak pengujian untuk aplikasi di dunia nyata, YOLOv8 telah menunjukkan performa tinggi dalam mendeteksi berbagai jenis objek. Rohatul Qolby (2024) melaporkan bahwa YOLOv8 menunjukkan performa tinggi dalam mendeteksi berbagai jenis APD dengan precision yang sangat baik.

2.1.8 Metode Waterfall

Metode *Waterfall* adalah salah satu model pengembangan perangkat lunak yang paling awal dan paling sederhana, yang mengikuti pendekatan linier dan berurutan. Dalam model ini, setiap fase pengembangan harus diselesaikan sepenuhnya sebelum melanjutkan ke fase berikutnya, yang mencakup analisis kebutuhan, desain sistem, implementasi, pengujian, dan pemeliharaan (Khan, 2023). Metode ini sangat cocok digunakan ketika persyaratan proyek sudah jelas

dan tidak ada perubahan yang diharapkan selama proses pengembangan (Khan, 2023).



Gambar 2.6 *Software Development Life Cycle Metode Waterfall*

Kelebihan dari metode *Waterfall* adalah kesederhanaannya yang membuatnya mudah dipahami dan dikelola, serta dokumentasi yang jelas di setiap tahap (Khan, 2023). Namun, metode ini juga memiliki kelemahan, seperti kurangnya fleksibilitas untuk mengakomodasi perubahan kebutuhan yang mungkin muncul setelah fase awal, sehingga tidak cocok untuk proyek yang kompleks atau berisiko tinggi (Khan, 2023). Meskipun demikian, metode *Waterfall* tetap banyak digunakan dalam pengembangan perangkat lunak, terutama untuk proyek-proyek kecil dan menengah yang memiliki spesifikasi yang stabil (Khan, 2023).

2.1.9 *Roboflow*

Roboflow adalah platform yang memfasilitasi pengembangan model deteksi objek dengan menyediakan alat untuk anotasi, augmentasi, dan manajemen dataset. Dalam beberapa tahun terakhir, berbagai penelitian di Indonesia telah memanfaatkan Roboflow untuk berbagai aplikasi. Misalnya, penelitian oleh Saepudin dkk. (2024) mengevaluasi kinerja YOLOv8 yang dioptimalkan dengan Roboflow untuk deteksi ekspresi wajah emosional, menunjukkan peningkatan akurasi dan efisiensi dalam pengenalan ekspresi wajah. Roboflow menjadi alat yang efektif dalam pengembangan berbagai aplikasi deteksi objek, membantu peneliti dalam meningkatkan akurasi dan efisiensi model yang dikembangkan.

2.1.10 *FastAPI*

FastAPI merupakan *framework* web modern berbasis Python yang dirancang

khusus untuk membangun Application Programming Interface (API) dengan performa tinggi dan efisien (Azhari, 2022). *Framework* ini dibangun di atas library Starlette dan Pydantic, serta mengimplementasikan Asynchronous Server Gateway Interface (ASGI) yang memungkinkan pemrosesan asinkronus dengan performa setara Node.js dan Go.

Berbeda dengan Flask yang bersifat general purpose, FastAPI fokus pada pembangunan web service API dengan keunggulan automatic code completion, dokumentasi API otomatis berbasis OpenAPI, dan validasi data ketat menggunakan Python type hints (Azhari, 2022). FastAPI diklaim mampu meningkatkan kecepatan pengembangan hingga 200-300% dan mengurangi bug hingga 40%, menjadikannya pilihan populer untuk pengembangan web service modern yang membutuhkan performa tinggi dan skalabilitas.

2.1.11 Python

Python adalah bahasa pemrograman tingkat tinggi yang dikembangkan oleh Guido van Rossum pada tahun 1991. Dikenal karena sintaks yang sederhana dan mudah dibaca, Python banyak digunakan untuk berbagai aplikasi, mulai dari pengembangan web hingga *Artificial Intelligence* (AI) dan analisis data. Keunggulan Python terletak pada desainnya yang mengutamakan keterbacaan, serta pustaka yang sangat luas, membuatnya ideal untuk pengembang pemula dan profesional. Python juga digunakan secara luas dalam pendidikan untuk mengenalkan konsep dasar pemrograman. (Romzi & Kurniawan, 2020).

2.2 Penelitian Terdahulu

Penelitian terdahulu merupakan salah satu landasan untuk merumuskan dan mengembangkan ide penelitian pada proyek akhir yang akan dilaksanakan. Penelitian mengenai deteksi objek dengan menggunakan *computer vision* sebelumnya telah banyak dirumuskan oleh penelitian terdahulu. Berikut adalah beberapa penelitian terdahulu yang dijadikan acuan untuk pengembangan sistem deteksi APD:

Penelitian pertama dilakukan oleh Nirvana dkk. (2023) dengan judul “Sistem Pendeteksi Alat Pelindung Diri (APD) pada Pekerja Konstruksi Berbasis *Convolutional Neural Network*” yang mengembangkan sistem deteksi APD

menggunakan arsitektur YOLOv7 pada pekerja konstruksi. Penelitian ini memanfaatkan dataset beragam yang mencakup kelas seperti helm dengan beberapa warna, kacamata, rompi, sarung tangan, dan sepatu keselamatan. Hasil pengujian menunjukkan bahwa YOLOv7 mencapai akurasi tinggi, dengan nilai mAP mencapai 0,947 pada jarak tertentu. Sistem ini dapat mendeteksi APD dengan baik, meskipun performanya menurun dalam kondisi pencahayaan rendah atau hujan, serta saat ada objek yang menghalangi seperti tubuh pekerja.

Penelitian kedua dilakukan Taufiqurrochman dan Februariyanti (2024) dengan judul “Rancang Bangun Aplikasi Deteksi Alat Pelindung Diri (APD) untuk Pekerja Proyek dengan Menggunakan Algoritma YOLOv5” yang merancang aplikasi deteksi APD untuk pekerja proyek dengan menggunakan YOLOv5. Dataset yang digunakan mencakup empat jenis APD, yaitu helm, masker, rompi, dan sepatu, yang dilatih sebanyak 100 epoch. Hasil deteksi menunjukkan tingkat akurasi tinggi dengan masing-masing persentase: helm 96%, rompi 96%, masker 95%, dan sepatu 92%. Aplikasi berbasis web ini bertujuan untuk meningkatkan kedisiplinan pekerja dalam menggunakan APD dan mengurangi risiko kecelakaan kerja di area proyek.

Penelitian ketiga dilakukan oleh Agustin Nurfirmansyah dan Rohman Dijaya (2022) dengan judul “Deteksi kelalaian alat pelindung diri (APD) pada pekerja konstruksi bangunan” yang mengembangkan sistem deteksi kelalaian penggunaan Alat Pelindung Diri (APD) pada pekerja konstruksi menggunakan arsitektur YOLOv4. Sistem ini menggunakan 500 citra dari 10 kelas APD dan non-APD untuk melatih model deteksi. Setelah pelatihan, model diuji menggunakan Webcam dan CCTV untuk mendeteksi penggunaan APD secara *real-time*. Hasil pengujian menunjukkan bahwa model berhasil mencapai akurasi 80% pada pengujian dengan Webcam dan lebih bervariasi pada pengujian dengan CCTV tergantung pada kualitas perangkat yang digunakan. Model YOLOv4 menghasilkan nilai mAP sebesar 99,99% dengan rata-rata loss 1,305%. Penelitian ini menyoroti pentingnya teknologi dalam meningkatkan pengawasan keselamatan kerja, meskipun masih diperlukan peningkatan pada kualitas perangkat keras dan jumlah data untuk mendapatkan hasil yang lebih optimal.

Penelitian keempat dilakukan oleh Rohatul Qolby (2024) dengan judul “Deteksi Alat Pelindung Diri (APD) Pada Pekerja Konstruksi Dengan Metode

YOLOv8 (Studi Kasus Di PT. Ondosere)” yang mengembangkan sistem deteksi penggunaan Alat Pelindung Diri (APD) pada pekerja konstruksi menggunakan arsitektur YOLOv8. Sistem ini dirancang untuk mendeteksi delapan kelas, yaitu *Glasses, Gloves, Helmet, Mask, NoGlasses, NoGloves, NoHelmet, dan NoMask*, dengan memanfaatkan dataset yang relevan. Model menunjukkan performa tinggi dengan precision rata-rata 91,88%, di mana deteksi masker mencapai precision tertinggi sebesar 97%, dan deteksi non-APD seperti NoHelmet mencapai precision hingga 99%. Sistem diimplementasikan pada aplikasi berbasis web yang diuji di PT. ONDOSERE dan berhasil mendeteksi berbagai APD dengan confidence maksimal 94% untuk helm dan sarung tangan. Meskipun terdapat tantangan dalam mendeteksi masker dan kacamata, penelitian ini berhasil menghadirkan solusi efektif untuk meningkatkan kepatuhan terhadap penggunaan APD di lingkungan kerja.

Penelitian kelima dilakukan oleh Krolkral dkk. (2025) dengan judul “*Enhancing YOLOv8n for Improved Small Object Detection on Custom Datasets*” yang meningkatkan arsitektur YOLOv8 khusus untuk mendeteksi objek berukuran kecil. Penelitian ini memodifikasi model dengan menambahkan skala deteksi baru yaitu lapisan P2 (resolusi spasial tinggi P2/4). Penambahan lapisan P2 ini terbukti sangat efektif dalam mempertahankan detail fitur halus seperti tekstur dan tepi objek yang seringkali hilang pada lapisan P3 hingga P5 bawaan YOLOv8. Hasil pengujian menunjukkan bahwa model dengan modifikasi lapisan P2 ini mampu melampaui performa arsitektur standar dalam mengenali target berukuran sangat kecil pada berbagai kondisi visual

Tabel 2.1 Perbandingan variabel penelitian

| Kategori | Peneliti 1 | Peneliti 2 | Peneliti 3 | Peneliti 4 | Peneliti 5 |
|----------|------------------------------------|---|---|---|--|
| Metode | YOLO v7 | YOLO v5 | YOLO v4 | YOLO v8 | YOLOv8 (Modifikasi lapisan deteksi P2) |
| Dataset | CHVG, Gloves, Boots, PPE-Detection | Dataset dari Saravana Alagar (4 objek APD: helm, masker, rompi, sepatu) | 500 citra dari 10 kelas APD dan non-APD | Glasses, Gloves, Helmet, Mask, NoGlasses, NoGloves, NoHelmet, | Dua custom dataset berskala besar dengan fokus target objek bervariasi |

| Kategori | Peneliti 1 | Peneliti 2 | Peneliti 3 | Peneliti 4 | Peneliti 5 |
|----------|--|--|---|--|--|
| | | | | <i>dan NoMask.</i> | |
| Tools | Roboflow | Google Colab, Visual Studio Code | Python, OpenCV, YOLOv4 | Teknologi <i>computer vision</i> (model YOLOv8) dengan implementasi <i>real-time</i> . | PyTorch, GPU Acceleration |
| Hasil | mAP 0,947 pada jarak ideal; akurasi 100% pada jarak 1,5-3 meter; performa menurun dalam kondisi rendah cahaya/hujan dan saat mendeteksi sarung tangan/kacamata | Akurasi deteksi: helm 96%, rompi 96%, masker 95%, sepatu 92%; bertujuan meningkatkan kedisiplinan pemakaian APD di proyek konstruksi | Akurasi deteksi 80% pada Webcam; akurasi bervariasi dengan CCTV. mAP 99,99%, rata-rata <i>loss</i> 1,305%. Menunjukkan pentingnya teknologi dalam pengawasan APD. | <i>Precision</i> rata-rata sebesar 91,88%, dengan <i>precision</i> tertinggi 97% untuk deteksi masker dan 99% untuk deteksi non-APD seperti <i>NoHelmet</i> . Pada pengujian aplikasi berbasis web, <i>confidence</i> tertinggi mencapai 94% untuk deteksi helm dan sarung tangan. | Modifikasi lapisan P2 secara signifikan meningkatkan performa deteksi objek berukuran kecil. Model mencapai mAP ₅₀ hingga 98,9% dan secara konsisten mengungguli arsitektur YOLO versi standar maupun yang lebih baru dalam mendeteksi detail fitur halus pada objek. |

Berdasarkan kajian terhadap beberapa penelitian terdahulu pada Tabel 2.1, diketahui bahwa penelitian-penelitian tersebut telah membahas topik yang relevan dengan penelitian ini, namun memiliki keterbatasan dari sisi pengenalan objek berskala kecil, penurunan akurasi pada tangkapan visual CCTV yang dinamis, maupun belum adanya penyesuaian arsitektur khusus untuk jarak pandang yang luas.

Penelitian ini memiliki perbedaan dengan penelitian sebelumnya karena mengkaji permasalahan deteksi atribut APD berukuran kecil (seperti sepatu keselamatan) pada sudut pandang luas (*wide-view angle*) CCTV dengan

menyesuaikan pendekatan penelitian terhadap kebutuhan dan karakteristik objek penelitian di PT Bumi Siak Pusako, yaitu melalui modifikasi penambahan *detection head* P2 pada arsitektur YOLOv8. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi pada optimalisasi pengawasan Keselamatan dan Kesehatan Kerja (K3) secara *real-time* dalam menjawab perumusan masalah serta melengkapi hasil-hasil penelitian yang telah ada sebelumnya.