

BAB I

PENDAHULUAN

1.1 Latar Belakang

Lingkungan kerja di atas kapal memiliki tingkat bahaya yang tinggi karena melibatkan berbagai aktivitas operasional yang kompleks. Setiap area kapal menyimpan risiko tersendiri yang menuntut penerapan standar keselamatan kerja yang ketat, termasuk kewajiban penggunaan Alat Pelindung Diri (APD). Di ruang mesin, misalnya, keberadaan mesin induk, pompa, dan jaringan perpipaan bertekanan tinggi berpotensi menimbulkan bahaya kebisingan, panas, maupun kontak dengan komponen berputar. Oleh karena itu, kru wajib menggunakan helm keselamatan, pelindung telinga, sarung tangan, dan sepatu pelindung. Sementara itu, di ruang navigasi (*navigation bridge*), meski tidak berhadapan langsung dengan peralatan berat, risiko tetap ada seperti kebakaran akibat instalasi listrik atau benturan saat cuaca ekstrem, sehingga penggunaan APD seperti *safety shoes*, rompi reflektif, dan perlengkapan darurat tetap diperlukan.

Untuk menjawab tantangan tersebut, regulasi keselamatan telah ditetapkan baik di tingkat internasional maupun nasional. Konvensi *International Convention for the Safety of Life at Sea* (SOLAS) serta *International Safety Management Code* (ISM Code) dari *International Maritime Organization* (IMO) menjadi acuan utama dalam penerapan standar keselamatan maritim. Di Indonesia, Undang-Undang Nomor 1 Tahun 1970 tentang Keselamatan Kerja dan Peraturan Menteri Perhubungan Nomor PM 84 Tahun 2013 tentang Keselamatan Pelayaran menegaskan kewajiban penggunaan APD dalam setiap aktivitas berisiko. Namun demikian, implementasi di lapangan sering kali belum optimal. Berdasarkan laporan RRI, Kepala Basarnas Pangkal Pinang menyatakan bahwa sebagian besar korban kecelakaan kapal laut yang berujung fatal terjadi karena tidak menggunakan alat keselamatan seperti *life jacket* (Achmad, 2023). Kasus lain juga dilaporkan ketika Tim SAR menyelamatkan seorang ABK kapal kargo di Perairan Benoa yang mengalami kecelakaan kerja, di mana pihak Basarnas kembali mengingatkan seluruh awak kapal dan nelayan agar selalu menyiapkan dan menggunakan perlengkapan keselamatan selama beraktivitas di laut (Suardika, 2024). Selain itu, Hubla, 2024) melaporkan terjadinya kecelakaan kerja di tongkang ASP Prosperity di mana salah satu *crew* mengalami kecelakaan kerja yang berujung meninggal dunia setelah terjatuh dan tidak segera ditemukan selama beberapa jam kemudian.

Data terbaru juga mendukung hal tersebut. Kegiatan *Awareness* Keselamatan dan Kesehatan Kerja (K3) yang digelar Pelindo menunjukkan bahwa tingginya angka kecelakaan kerja di sektor pelabuhan, diperkirakan sekitar 80-85%, disebabkan oleh tindakan tidak aman atau faktor manusia, bukan kegagalan peralatan. Ditekankan pentingnya penggunaan Alat Pelindung Diri (APD) yang sesuai, seperti helm dan sepatu keselamatan, karena kelalaian dalam mengenakan APD menjadi salah satu penyebab utama terjadinya insiden (Pelindo, 2025). Sebagai tindak lanjut, Pelindo memperkuat implementasi Sistem Manajemen Keselamatan dan Kesehatan Kerja (SMK3) melalui

distribusi APD, kampanye keselamatan, peningkatan pengawasan, serta sanksi disiplin bagi pelanggar (SMPT, 2023).

Seiring berkembangnya teknologi, transformasi digital menjadi solusi strategis untuk memperkuat sistem keselamatan kerja. Pemanfaatan kecerdasan buatan (*Artificial Intelligence*), khususnya metode deteksi objek *You Only Look Once* (YOLO), memungkinkan pengawasan otomatis terhadap kepatuhan penggunaan APD secara *real-time*. Sistem berbasis YOLO dapat diintegrasikan dengan kamera pengawas untuk mendeteksi helm, rompi, sarung tangan, dan sepatu pelindung, sekaligus mengenali wajah kru untuk mendukung absensi berbasis visual. Lebih jauh lagi, sistem ini dapat mengidentifikasi situasi darurat seperti jatuhnya pekerja dan secara otomatis mengaktifkan alarm sebagai bagian dari tanggap darurat.

Dengan kecepatan, akurasi, serta efisiensi yang tinggi, YOLO telah banyak diterapkan dalam sistem deteksi keselamatan, termasuk di kapal dan pelabuhan. Penelitian sebelumnya menunjukkan bahwa metode ini mampu mendukung pencegahan kecelakaan kerja melalui deteksi APD secara otomatis (Barlybayev et al., 2024) dan dapat diimplementasikan secara optimal pada perangkat komputasi ringan, sehingga relevan digunakan pada lingkungan kerja berisiko tinggi (Permatasari et al., 2024).

1.2 Rumusan Masalah

- a. Bagaimana merancang sistem yang mampu mendeteksi kelengkapan Alat Pelindung Diri (APD) secara otomatis dengan penggunaan YOLO?
- b. Bagaimana merancang sistem yang dapat mendeteksi kondisi awak kapal yang jatuh secara otomatis dengan penggunaan YOLO?
- c. Bagaimana merancang sistem absensi otomatis untuk awak kapal dengan integrasi YOLO dan teknologi pengenalan wajah (*face recognition*)?
- d. Bagaimana merancang sistem yang mampu menghasilkan laporan otomatis berdasarkan hasil deteksi APD, deteksi jatuh, dan absensi, serta mengaktifkan alarm apabila ditemukan kondisi berbahaya?

1.3 Tujuan dan Manfaat

- a. Merancang sistem deteksi kelengkapan APD awak kapal berbasis YOLO untuk memastikan penggunaan APD dapat dipantau secara otomatis sebagai bagian dari *preventive safety*.
- b. Merancang sistem deteksi jatuh otomatis berbasis YOLO sebagai bagian dari *early warning system* untuk mengidentifikasi kondisi darurat awak kapal dan memberikan peringatan secara cepat.
- c. Merancang sistem absensi otomatis berbasis kamera dengan integrasi YOLO dan teknologi pengenalan wajah sehingga proses pendataan kehadiran menjadi lebih mudah, cepat, dan efisien.
- d. Mengembangkan sistem penyimpanan laporan dan alarm otomatis yang menyimpan data hasil deteksi, serta mengaktifkan alarm ketika terjadi pelanggaran atau kondisi darurat.

1.4 Batasan Masalah

- a. Keselamatan kerja melepituo sistem hanya mendeteksi beberapa contoh APD yaitu *Safety Helmet*, *Safety Vest* dan *Safety Shoes*.
- b. Sistem mendeteksi kondisi saat terjatuh dan tidak bergerak dalam waktu tertentu agar memberikan responsi darurat yang cepat tanpa memberikan *false alarm*.
- c. Sistem dijalankan pada komputer yang terhubung dengan kamera yang dipasang di pintu masuk ruang kerja menghadap ke *hallway* untuk mendeteksi awak kapal yang melintas, serta alarm audio yang ditempatkan di ruang *sick bay* sebagai peringatan saat terdeteksi kondisi berbahaya.
- d. Sistem beroperasi dengan berbasis terminal tanpa antarmuka grafis (GUI) atau *dashboard*, dengan semua proses ditampilkan langsung melalui layar terminal menggunakan *OpenCV*.

1.5 Tinjauan Pustaka

1.5.1 Definisi Keselamatan dan Kesehatan Kerja

Konsep Keselamatan dan Kesehatan Kerja (K3) adalah pendekatan yang terstruktur dan menyeluruh untuk menjamin lingkungan kerja yang aman dan melindungi pekerja dari berbagai risiko kecelakaan dan gangguan kesehatan. Konsep ini tidak hanya bertujuan untuk mencegah kecelakaan, tetapi juga untuk meningkatkan produktivitas dan kesejahteraan tenaga kerja melalui penerapan prosedur keselamatan yang konsisten dan sistematis.

1.5.2 Alat Pelindung Diri

Alat Pelindung Diri (APD) merupakan perlengkapan wajib yang dirancang untuk melindungi tenaga kerja dari potensi bahaya atau risiko cedera di lingkungan kerja. APD berfungsi sebagai penghalang antara tubuh pekerja dengan sumber bahaya fisik, kimia, biologi, maupun mekanis. Menurut regulasi nasional, seperti yang tertuang dalam Undang-Undang No. 1 Tahun 1970 tentang Keselamatan Kerja dan Peraturan Menteri Ketenagakerjaan No. 8 Tahun 2010, setiap pekerja wajib menggunakan APD sesuai dengan potensi risiko yang ada di tempat kerja. Peralatan yang termasuk dalam kategori APD antara lain helm keselamatan, sepatu pelindung, sarung tangan, rompi reflektif, masker, dan pelindung mata. Penggunaan APD yang tepat dan konsisten tidak hanya melindungi pekerja dari cedera, tetapi juga merupakan indikator kepatuhan terhadap budaya keselamatan kerja yang profesional. Di sektor maritim, terutama pada area dengan risiko tinggi seperti ruang mesin atau dek terbuka, pemakaian APD menjadi krusial dalam mencegah kecelakaan kerja yang dapat berakibat fatal (Maulana Putra et al., 2024).

1.5.3 Regulasi Internasional

- a. ***International Labour Organization (ILO) – Maritime Labour Convention (MLC) 2006***

International Labour Organization (2022) menetapkan bahwa setiap

negara anggotanya wajib mengatur perlindungan kesehatan dan keselamatan kerja bagi pelaut. Standard A4.3 menegaskan bahwa kapal harus menyediakan APD yang sesuai ukuran, mengambil langkah-langkah pencegahan untuk mengurangi risiko kecelakaan, paparan faktor lingkungan berbahaya, serta bahaya dari penggunaan mesin. Selain itu, MLC 2006 mewajibkan adanya program keselamatan kerja di atas kapal, investigasi kecelakaan, serta mekanisme pelaporan dan perbaikan kondisi tidak aman. Ketentuan ini menjadi dasar internasional bagi implementasi keselamatan kerja bagi awak kapal.

b. *International Safety Management (ISM) Code*

ISM Code merupakan komponen penting dari SOLAS, yang menekankan pentingnya pengelolaan risiko dan keselamatan operasional kapal. Dalam Bagian 7 tentang Operasi Kapal, dinyatakan bahwa perusahaan harus menetapkan prosedur kerja yang mencakup penggunaan APD sebagai bagian dari operasi aman di kapal, termasuk dalam situasi darurat.

c. *International Convention for the Safety of Life at Sea (SOLAS)*

Dalam SOLAS, Chapter I Regulation 29 dijelaskan bahwa sistem keselamatan di kapal sekurang-kurangnya harus terdiri dari rencana darurat tertulis yang memuat seluruh situasi darurat yang dapat diperkirakan sebelumnya, termasuk kecelakaan personel. Selain itu, Administrasi dapat menerima penggunaan sistem pendukung keputusan berbasis komputer di anjungan yang memuat seluruh informasi dalam rencana darurat, prosedur, dan daftar periksa, serta mampu menampilkan tindakan yang direkomendasikan untuk menghadapi keadaan darurat yang mungkin terjadi. Selanjutnya, Chapter IX Regulation 1 mewajibkan perusahaan dan kapal untuk mematuhi ketentuan ISM Code, yang dalam konteks regulasi ini harus diperlakukan sebagai ketentuan yang bersifat wajib.

1.5.4 Regulasi Nasional

a. Undang-Undang Dasar Negara Republik Indonesia Tahun 1945

Sebagai undang-undang tertinggi negara, UUD 1945 memberikan dasar untuk perlindungan keselamatan kerja. "*Tiap-tiap warga negara berhak atas pekerjaan dan penghidupan yang layak bagi kemanusiaan,*" kata Pasal 27, ayat 2. Hal ini menunjukkan bahwa hak dasar manusia yang dijamin negara termasuk keselamatan kerja bagi tenaga kerja maritim.

b. Undang-Undang No. 1 Tahun 1970 tentang Keselamatan Kerja

Menurut UU Nomor 1 Tahun 1970 Pasal 9 ayat (1) menetapkan bahwa pengurus berkewajiban memberikan penjelasan kepada setiap tenaga kerja baru mengenai berbagai aspek keselamatan sebelum mereka mulai bekerja. Penjelasan tersebut mencakup kondisi dan potensi bahaya di lingkungan kerja, seluruh bentuk pengamanan serta alat perlindungan

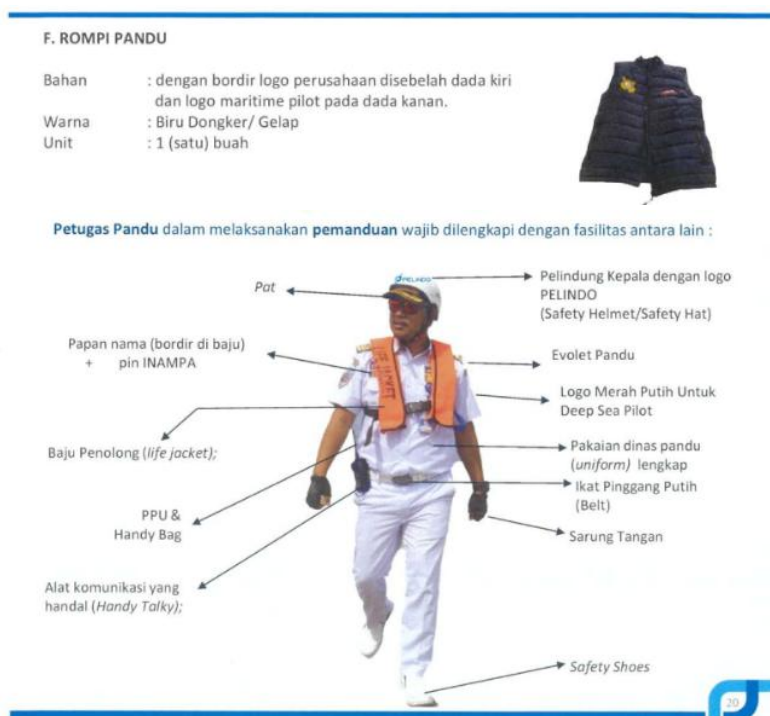
yang wajib digunakan, ketersediaan dan penggunaan alat pelindung diri yang sesuai, serta cara kerja yang aman beserta sikap yang harus diterapkan saat melaksanakan tugas.

c. Peraturan Menteri Ketenagakerjaan Nomor 8 Tahun 2010 tentang Alat Pelindung Diri

Kementerian Ketenagakerjaan Republik Indonesia mengatur secara khusus mengenai penggunaan alat pelindung diri sebagai bagian dari penerapan keselamatan dan kesehatan kerja. Dalam peraturan ini ditegaskan bahwa setiap pemberi kerja memiliki kewajiban untuk menyediakan APD yang sesuai dengan potensi bahaya di lingkungan kerja dan memastikan bahwa alat tersebut memenuhi standar yang berlaku. Di sisi lain, tenaga kerja diwajibkan untuk menggunakan APD yang telah disediakan selama melakukan pekerjaan. Selain itu, peraturan ini juga menjelaskan berbagai jenis APD yang digunakan untuk melindungi bagian tubuh tertentu, seperti kepala, mata, pernapasan, tangan, dan kaki.

1.5.5 Regulasi Perusahaan (PT. Pelabuhan Indonesia)

Dalam Buku Panduan Standarisasi Operasional Pelayanan Kapal 2022 PT. PELINDO, diaturkan di halaman 19 – 28 APD yang wajib digunakan oleh pandu dan kru kapal.



Gambar 1. Aturan APD PT. PELINDO
Sumber : PT PELABUHAN INDONESIA (PERSERO), (2022)

1.5.6 Responsi Darurat

Responsi darurat adalah kemampuan sistem atau individu untuk merespons insiden atau situasi berisiko secara cepat, tepat, dan efektif. Dalam konteks keselamatan kerja, salah satu kejadian yang paling kritis adalah jatuh (*fall incident*), yang dapat menyebabkan cedera serius atau bahkan kematian jika tidak ditangani secara cepat. Oleh karena itu, sistem respons darurat yang efektif menjadi aspek penting dalam manajemen keselamatan kerja.

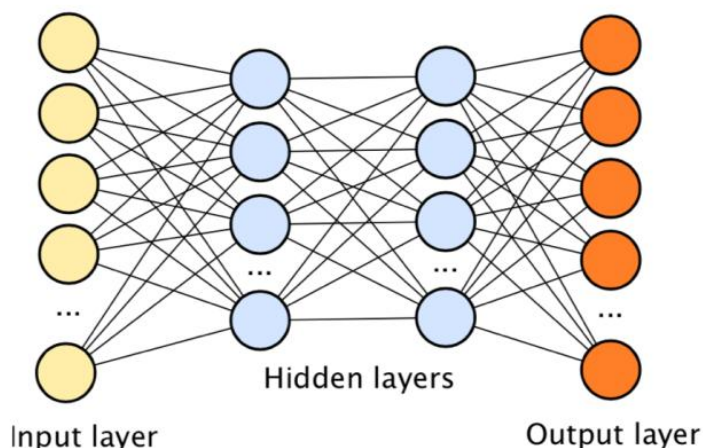
Waktu respon adalah durasi antara terjadinya insiden dan tindakan yang dilakukan untuk menanganinya. Waktu respon yang cepat dapat sangat mengurangi risiko cedera dan meningkatkan keselamatan individu. Beberapa studi menyarankan bahwa sistem deteksi kejadian darurat, seperti jatuh, harus mampu memberikan notifikasi segera (*Immediate Alert*) dalam hitungan detik hingga satu menit. Misalnya, teknologi berbasis sensor atau *smartphone* dapat mengirimkan peringatan secara otomatis ke petugas atau orang terdekat dalam waktu singkat. Selanjutnya, *Intervention Time*, yaitu waktu yang dibutuhkan petugas atau sistem otomatis untuk tiba di lokasi dan memberikan pertolongan pertama, idealnya tidak lebih dari lima menit. Penerapan waktu respon yang cepat ini terbukti dapat menurunkan risiko cedera serius pada individu dewasa dan mempercepat penanganan darurat secara efektif (Newaz and Hanada, 2023).

Danilenka et al., (2023) menjelaskan dalam sistem deteksi jatuh otomatis, penting untuk mengatur waktu deteksi dan konfirmasi agar alarm tidak menyala secara palsu. Sistem modern biasanya membutuhkan 0,5–2 detik untuk mendeteksi perubahan posisi tubuh atau akselerasi yang menandakan jatuh. Setelah deteksi awal, digunakan *hold time* atau konfirmasi beberapa detik berturut-turut sebelum alarm aktif. Misalnya, jika posisi tubuh terlentang atau jatuh terdeteksi lebih dari 1–1,5 detik berturut-turut, baru alarm dinyalakan. Hal ini membedakan antara gerakan normal (misal membungkuk atau duduk tiba-tiba) dengan jatuh yang sebenarnya, sekaligus meminimalkan *false alarm*. Dengan kombinasi *threshold* posisi tubuh, kecepatan jatuh, dan orientasi tubuh, sistem dapat menyalakan alarm dalam total $\pm 1,5$ –3 detik setelah jatuh terjadi, sehingga respons dapat segera dilakukan.

1.5.7 Deep Learning

Kecerdasan buatan (*Artificial Intelligence*) merupakan bidang dalam ilmu komputer yang berfokus pada pengembangan sistem yang mampu meniru kemampuan berpikir dan pengambilan keputusan manusia. AI mencakup berbagai pendekatan, mulai dari sistem berbasis aturan hingga algoritma pembelajaran yang lebih adaptif. Salah satu cabang utama dari AI adalah *Machine Learning*, yaitu pendekatan yang memungkinkan sistem belajar dari data untuk membuat prediksi atau keputusan tanpa diprogram secara eksplisit. ML bekerja dengan membangun model matematis berdasarkan pola yang diekstraksi dari data, sehingga dapat digunakan dalam berbagai aplikasi seperti industri, transportasi, dan sistem keselamatan kerja. Perkembangan lebih lanjut dari ML melahirkan *deep learning*, yang menggunakan jaringan saraf tiruan bertingkat (*Deep Neural Networks*) untuk memproses data kompleks, terutama data tidak terstruktur seperti citra dan video. DNN memiliki banyak lapisan tersembunyi yang mampu mengekstraksi representasi data secara bertahap, dari fitur sederhana hingga fitur

kompleks, sehingga menjadi fondasi berbagai arsitektur modern seperti CNN, RNN, dan Transformer (Choi et al., 2020).



Gambar 2. *Deep Neural Network*

Sumber: <https://blog.spheron.network/deep-learning-basics-a-clear-overview>

Deep Neural Network (DNN) adalah jenis jaringan saraf tiruan yang terdiri dari beberapa lapisan tersembunyi antara lapisan input dan output. Setiap lapisan berisi sejumlah neuron yang terhubung secara penuh (*fully connected*) dengan lapisan sebelumnya dan sesudahnya. Model ini dirancang untuk mempelajari representasi data yang kompleks melalui proses pelatihan yang melibatkan propagasi maju (*forward propagation*) dan pembaruan bobot menggunakan algoritma seperti *backpropagation* (Choi et al., 2020).

- **Lapisan Input (*Input Layer*)** menerima data mentah, seperti fitur gambar, teks, atau sinyal lainnya. Setiap neuron pada lapisan ini mewakili satu fitur dari data yang akan diproses.
- **Lapisan Tersembunyi (*Hidden Layers*)** melakukan transformasi non-linear terhadap input melalui fungsi aktivasi seperti ReLU, *sigmoid*, atau *tanh*. Setiap neuron mengalikan input dengan bobotnya dan menambahkan bias sebelum dilewatkan ke fungsi aktivasi, memungkinkan jaringan untuk mengekstraksi representasi fitur yang kompleks secara bertahap.
- **Lapisan Output (*Output Layer*)** menghasilkan prediksi akhir, seperti kelas objek dalam klasifikasi citra. Fungsi aktivasi yang digunakan pada lapisan output tergantung pada jenis tugas: *softmax* untuk klasifikasi multi-kelas atau *sigmoid* untuk klasifikasi biner.

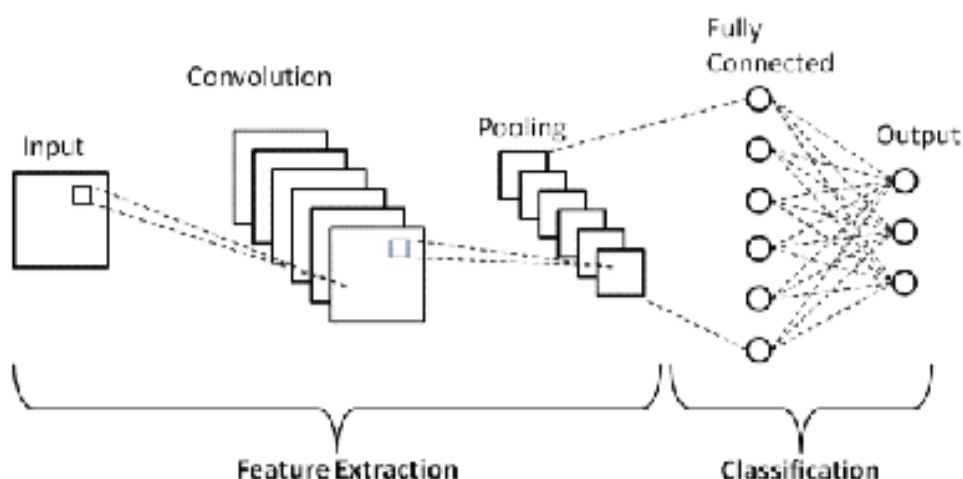
1.5.8 *Convolutional Neural Network (CNN)*

Convolutional Neural Network merupakan salah satu arsitektur *deep learning* yang paling banyak digunakan dalam bidang *computer vision*, terutama untuk tugas seperti klasifikasi gambar, deteksi objek, dan segmentasi. CNN didesain untuk meniru cara kerja

sistem visual manusia dalam mengenali pola visual secara bertahap dari yang sederhana hingga kompleks (Arnita et al., 2022).

CNN merupakan varian khusus dari DNN yang dirancang untuk memproses data dengan struktur *grid*, seperti gambar. Meskipun keduanya merupakan jenis jaringan saraf tiruan yang dalam, terdapat perbedaan mendasar dalam arsitektur dan penerapannya. Perbedaan utama antara CNN dan DNN terletak pada cara mereka memanfaatkan struktur spasial data. CNN dirancang untuk memanfaatkan keterkaitan spasial dalam data, menjadikannya lebih efisien dan efektif untuk tugas-tugas seperti klasifikasi gambar, deteksi objek, dan segmentasi. Sementara itu, DNN lebih umum dan dapat diterapkan pada berbagai jenis data, tetapi mungkin kurang efisien dalam menangani data berdimensi tinggi seperti citra.

Pada CNN, lapisan konvolusi (*convolutional layer*) berfungsi sebagai inti utama yang melakukan ekstraksi fitur. Lapisan ini menggunakan filter atau *kernel* yang digeser melintasi citra untuk menangkap pola lokal seperti tepi, garis, tekstur, atau bentuk sederhana. Fitur-fitur yang terdeteksi pada lapisan awal kemudian dipadukan pada lapisan yang lebih dalam untuk mengenali pola yang lebih kompleks, misalnya wajah manusia atau objek tertentu. Setelah itu, lapisan *pooling* digunakan untuk mereduksi dimensi peta fitur sambil mempertahankan informasi penting. Teknik *pooling* yang paling umum adalah *max pooling*, yang memilih nilai maksimum dalam suatu area, sehingga jaringan menjadi lebih efisien dalam komputasi dan lebih *robust* terhadap translasi atau pergeseran objek dalam gambar. Dengan kombinasi lapisan konvolusi dan *pooling* ini, CNN mampu mengekstraksi representasi hierarkis dari citra, mulai dari fitur sederhana hingga kompleks, tanpa kehilangan informasi penting (Arnita et al., 2022).



Gambar 3. Arsitektur CNN

Sumber: Arnita et al., 2022

Arnita et al., 2022 menjelaskan CNN terdiri dari beberapa tahapan utama, yang secara luas dikategorikan menjadi ekstraksi fitur dan klasifikasi.

a. Ekstraksi Fitur

Pada arsitektur CNN, proses pertama yang dilakukan adalah ekstraksi fitur. Tahapan ini dimulai dari lapisan *input* yang menerima citra mentah untuk kemudian diproses pada lapisan konvolusional. Lapisan konvolusional menggunakan kernel atau filter yang bergerak melintasi citra untuk mendeteksi pola lokal, seperti tepi (*edges*), garis, atau tekstur. Setiap hasil konvolusi membentuk *feature map*, yang kemudian dilewatkan pada fungsi aktivasi, umumnya *ReLU (Rectified Linear Unit)*, guna menambahkan sifat non-linear sehingga jaringan dapat mempelajari pola yang lebih kompleks.

Selanjutnya, CNN menggunakan lapisan *pooling*, seperti *max pooling* atau *average pooling*, untuk mereduksi dimensi data tanpa kehilangan informasi penting. *Pooling* berfungsi memperkecil ukuran representasi, mengurangi kompleksitas komputasi, serta meningkatkan ketahanan jaringan terhadap pergeseran (*translation invariance*) pada citra. Dengan demikian, proses ekstraksi fitur memungkinkan CNN mengubah citra mentah menjadi representasi yang lebih sederhana namun tetap kaya informasi, yang siap diproses pada tahap klasifikasi.

b. Klasifikasi

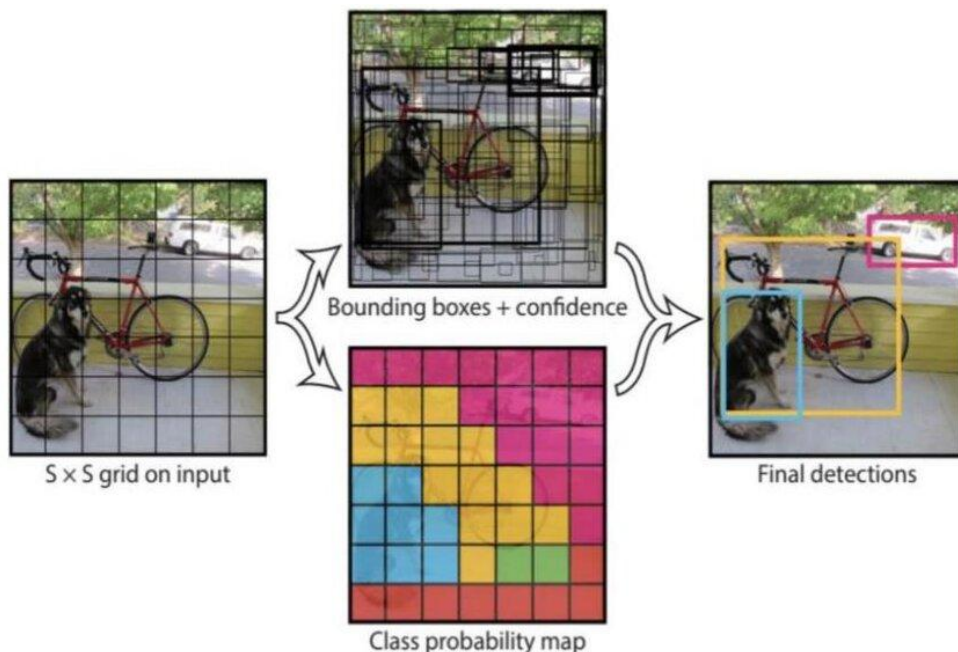
Setelah melalui beberapa lapisan konvolusional dan *pooling*, *feature map* yang dihasilkan diubah menjadi vektor satu dimensi melalui proses *flattening*. Vektor ini berperan sebagai masukan pada lapisan *fully connected (dense layer)*, yang berfungsi sama seperti lapisan tersembunyi pada *Deep Neural Network (DNN)*. Pada tahap ini, kombinasi fitur yang telah diekstraksi digunakan untuk mengenali pola yang lebih abstrak dan menghubungkannya dengan kelas target.

Tahap akhir adalah lapisan output, yang umumnya menggunakan fungsi aktivasi *softmax*. Fungsi ini mengubah nilai keluaran menjadi distribusi probabilitas untuk setiap kelas. Misalnya, sebuah citra dapat diprediksi dengan probabilitas 0,7 untuk “helm”, 0,2 untuk “tanpa helm”, dan 0,1 untuk “vest”. Probabilitas tertinggi kemudian ditetapkan sebagai hasil prediksi akhir. Dengan mekanisme klasifikasi ini, CNN mampu menghubungkan fitur visual dari citra dengan label yang sesuai, sehingga dapat digunakan dalam aplikasi seperti deteksi objek, pengenalan wajah, maupun sistem keselamatan kerja berbasis visi komputer.

1.5.9 You Only Look Once (YOLO)

Sejak pertama kali diperkenalkan oleh Joseph Redmon dan rekan-rekannya pada tahun 2016, YOLO terus mengalami perkembangan signifikan. YOLOv1 menitikberatkan pada kecepatan, sementara YOLOv2 dan YOLOv3 yang dikembangkan meningkatkan akurasi dengan penggunaan *anchor boxes* dan arsitektur yang lebih dalam. YOLOv4 memperkenalkan teknik optimasi seperti *mosaic augmentation* dan *backbone CSPDarknet53* agar performa lebih baik. Selanjutnya, YOLOv5 menghadirkan model-

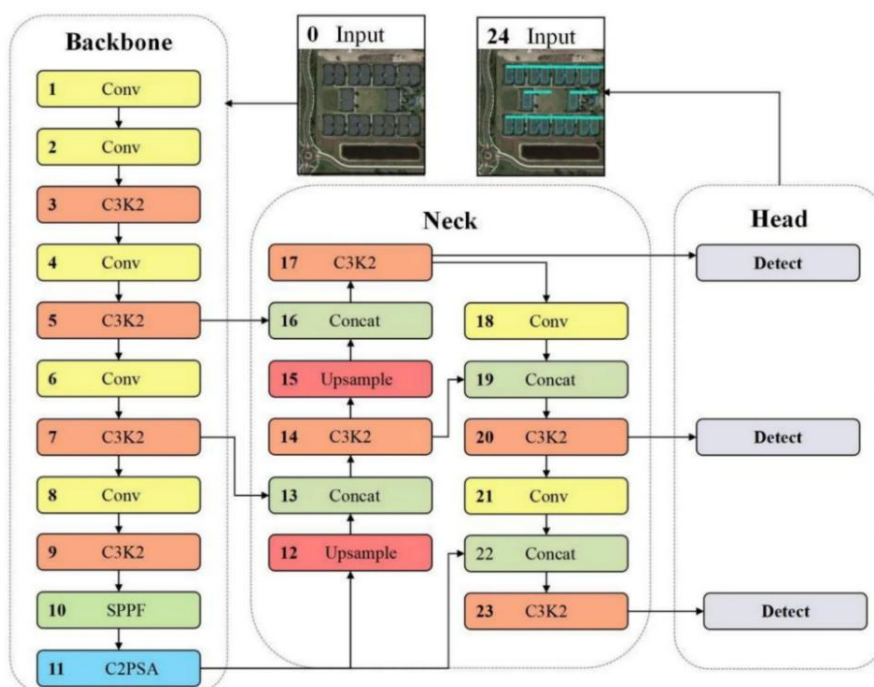
model dengan variasi ukuran berbasis *PyTorch* yang lebih fleksibel untuk berbagai kebutuhan. Versi berikutnya seperti YOLOv6, YOLOv7, dan YOLOv8 terus memperbaiki efisiensi, akurasi, serta dukungan untuk perangkat dengan sumber daya terbatas (Barlybayev et al., 2024).



Gambar 4. Tahapan Deteksi Objek
Sumber: Redmon et al., 2016

Prinsip dasar YOLO yang diperkenalkan oleh Redmon dapat diilustrasikan melalui gambar di atas. Alur kerjanya dimulai dengan membagi gambar input menjadi grid $S \times S$. Setiap sel dalam *grid* tersebut bertanggung jawab untuk memprediksi beberapa *bounding boxes* beserta *confidence score* (tingkat kepercayaan keberadaan objek), sekaligus menghasilkan *class probability map* (peta probabilitas kelas) untuk klasifikasi. Seluruh prediksi lokal ini kemudian secara langsung digabungkan dan diproses dalam satu tahap inferensi tunggal untuk menghasilkan *final detections*. Pendekatan *single-stage* inilah yang menjadi fondasi kecepatan tinggi YOLO, dan konsep *grid* serta prediksi terpadu ini terus menjadi dasar yang dikembangkan dan *dioptimasi* secara signifikan pada setiap iterasi berikutnya, seperti yang tercantum dalam perkembangan historisnya.

Perkembangan ini terus berlanjut hingga versi terkini seperti YOLOv11, yang memfokuskan pada optimalisasi efisiensi komputasi dan stabilitas pelatihan tanpa mengorbankan kecepatan inferensi *real-time*. YOLOv11 mengadopsi desain jaringan yang lebih ringan serta memperkuat mekanisme ekstraksi fitur *multiskala* pada arsitekturnya, sehingga meningkatkan konsistensi deteksi objek dari ukuran kecil hingga besar. Selain itu, penyempurnaan pada strategi pelatihan dan struktur jaringan membuatnya lebih adaptif terhadap berbagai skenario aplikasi, baik dalam lingkungan terkontrol maupun kondisi lapangan yang kompleks (Khanam and Hussain, 2024).



Gambar 5. Diagram Struktur Kerja YOLOv11

Sumber: <https://www.nature.com/articles/s41598-025-96314-x/figures/2>

(Khanam and Hussain, 2024) menjelaskan alur kerja YOLOv11 mengikuti arsitektur modular yang terdiri dari tiga komponen utama, yaitu:

a. *Backbone*

Backbone berfungsi sebagai *feature extractor* utama yang bertugas mengekstrak ciri atau fitur visual dari gambar *input*. Proses diawali dengan lapisan *konvolusional* untuk mengekstrak fitur dasar dan mengurangi resolusi gambar. Inovasi kunci YOLOv11 pada *backbone* adalah pengenalan blok C3k2 yang lebih cepat dan efisien, serta penggunaan SPPF (*Spatial Pyramid Pooling - Fast*) dan modul baru C2PSA. Modul C2PSA menambahkan mekanisme *spatial attention* yang memungkinkan model secara otomatis lebih fokus pada wilayah penting dalam gambar, sehingga meningkatkan akurasi untuk objek kecil atau kompleks.

b. *Neck*

Neck bertindak sebagai jembatan yang mengagregasi dan memperkuat fitur dari *backbone*. Komponen ini menggabungkan *feature maps* dari berbagai tingkat (skala) yang berbeda melalui proses *upsampling* dan *concatenation*. Tujuannya adalah untuk menciptakan representasi fitur yang kaya dan multi-skala, sehingga model dapat mendeteksi objek dengan ukuran beragam—dari sangat kecil hingga besar—secara efektif. Di YOLOv11, neck juga mengadopsi blok C3k2 dan memperkuat mekanisme perhatian dari modul C2PSA untuk seleksi wilayah yang lebih baik.

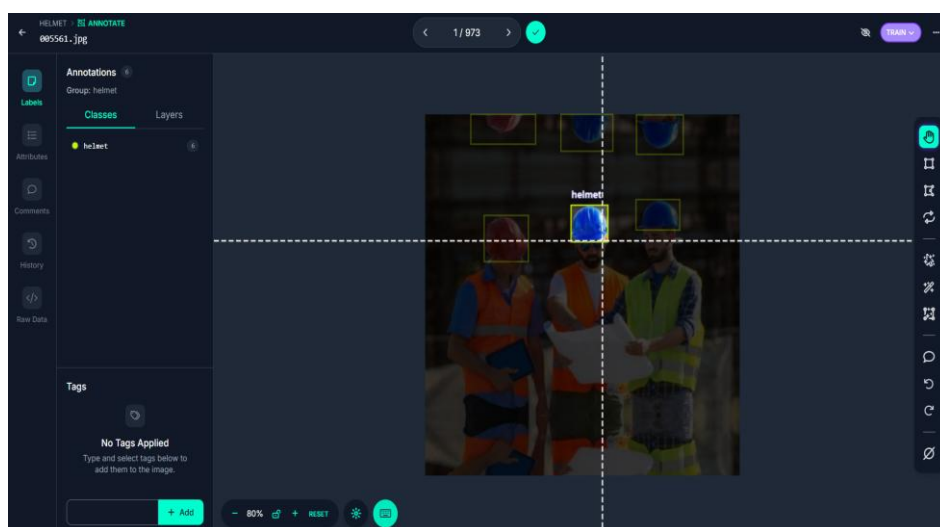
c. Head

Head adalah komponen akhir yang bertanggung jawab untuk menghasilkan prediksi. Bagian ini mengambil fitur yang telah diproses dari *neck* dan mentransformasikannya menjadi *output* praktis: koordinat *bounding box* untuk lokalisasi, skor keberadaan objek (*objectness*), dan skor klasifikasi. Di dalamnya, terdapat beberapa blok C3k2 dan lapisan CBS (*Convolution-BatchNorm-Silu*) untuk penyempurnaan fitur. Proses diakhiri dengan lapisan konvolusi akhir dan lapisan *Detect* yang mengkonsolidasi semua prediksi dari berbagai skala menjadi *output* terpadu untuk setiap objek yang terdeteksi.

Setelah itu, YOLO membagi citra *input* menjadi *grid* berukuran $s \times s$. Setiap sel *grid* bertanggung jawab untuk memprediksi apakah ada objek yang pusatnya berada di dalam sel tersebut. Dari setiap sel, YOLO menghasilkan sejumlah *bounding box* yang dilengkapi dengan nilai *confidence score*, yaitu probabilitas bahwa kotak tersebut benar-benar berisi objek. Selain itu, sistem juga menghitung *class probability* untuk menentukan jenis objek di dalam *bounding box*, misalnya helm, orang, atau kendaraan. Karena dalam praktiknya satu objek dapat menghasilkan banyak prediksi kotak, maka digunakan tahap *Non-Maximum Suppression* (NMS) untuk menyaring kotak yang tumpang tindih. Hanya *bounding box* dengan *confidence* tertinggi yang dipertahankan. Hasil akhirnya berupa final *detection*, yaitu posisi *bounding box* dengan label kelas dan skor probabilitas yang paling akurat (Permatasari et al., 2024).

1.5.10 Training Dataset

Dalam pembangunan model berbasis YOLO maupun CNN lainnya, dataset berperan sebagai elemen fundamental. Dataset biasanya berupa sekumpulan citra yang telah dianotasi, baik dengan *bounding box* untuk deteksi objek maupun label identitas untuk pengenalan wajah. Kualitas dan kelengkapan *dataset* sangat menentukan akurasi dan kemampuan generalisasi model (Barlybayev et al., 2024).



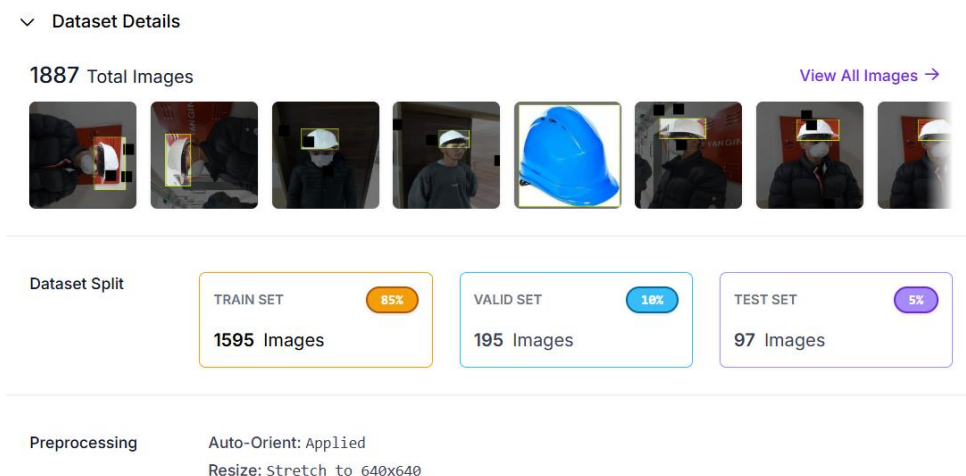
Gambar 6. Labeling

Proses pengumpulan data dilakukan dengan mengambil citra dari *dataset* publik seperti COCO dan Pascal VOC, maupun dengan membuat *custom dataset* sesuai kebutuhan aplikasi. Tahap ini harus mempertimbangkan variasi kondisi pencahayaan, sudut pandang, serta latar belakang agar model mampu melakukan generalisasi dengan baik. Setelah citra terkumpul, dilakukan proses *labeling* yaitu memberi anotasi pada setiap objek dengan menggunakan *bounding box* yang mendefinisikan area objek berdasarkan koordinat pixel pada citra semakin presisi anotasi *bounding box*, semakin baik pula model dalam mengenali pola visual objek saat proses pelatihan (Yusup Efendi et al., 2024).

Dalam proses *training*, dataset tidak hanya dikumpulkan tetapi juga diproses agar model dapat belajar secara optimal. Tahapan pelatihan umumnya meliputi:

a. *Preprocessing Data*

Citra yang digunakan dalam proses pelatihan biasanya diseragamkan ukurannya agar model dapat belajar secara konsisten tanpa bias dimensi. Misalnya, citra asli yang memiliki resolusi berbeda-beda (besar maupun kecil) diubah ke ukuran standar tertentu seperti 320×320 pixel, 416×416 pixel, atau 640×640 pixel sesuai kebutuhan dan kapasitas model. Selain itu, citra juga dinormalisasi (misalnya dengan membagi nilai pixel 0–255 menjadi rentang 0–1) agar stabil dalam proses pelatihan. Setelah itu, *dataset* dibagi menjadi data latih, validasi, dan uji sehingga model dapat belajar dari data latih, dievaluasi kinerjanya pada data validasi, dan diuji generalisasinya pada data uji untuk menghindari *overfitting*.

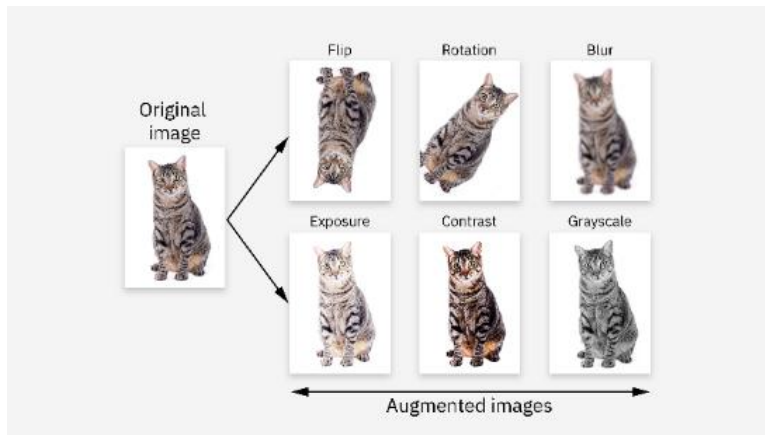


Gambar 7. Pembagian Dataset

Sumber: <https://app.roboflow.com/k-oaniq/helmet-evxi3-ewfpw>)

b. *Data Augmentation*

Teknik seperti rotasi, *flipping*, *scaling*, *cropping*, hingga perubahan pencahayaan diterapkan untuk memperluas variasi data, sehingga model lebih tahan terhadap kondisi nyata yang bervariasi.

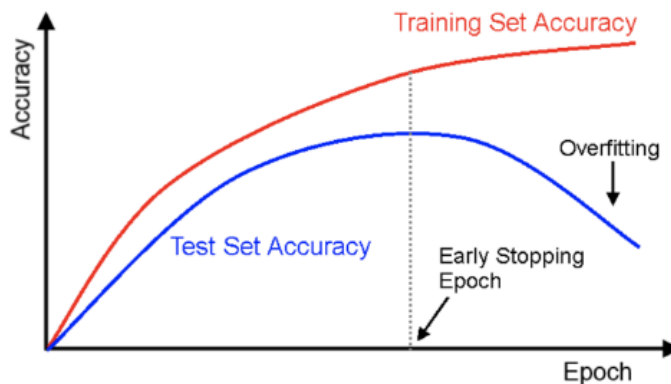


Gambar 8. Augmentasi Data

Sumber: <https://www.ibm.com/think/topics/data-augmentation>

c. *Training Iteratif (Epochs)*

Proses pelatihan model dilakukan secara bertahap dalam satuan iterasi yang disebut *epoch*, yaitu satu siklus penuh melewati seluruh *dataset* latih. Pada setiap *epoch*, model melakukan prediksi, kemudian hasil prediksi dibandingkan dengan label sebenarnya untuk menghitung *loss function* yang menunjukkan besar kesalahan. Semakin banyak *epoch* yang dijalankan, semakin besar peluang model untuk mengenali pola yang lebih kompleks, namun jumlah *epoch* yang terlalu banyak juga dapat menyebabkan *overfitting*, di mana model terlalu menyesuaikan diri dengan data latih dan gagal melakukan generalisasi pada data baru.

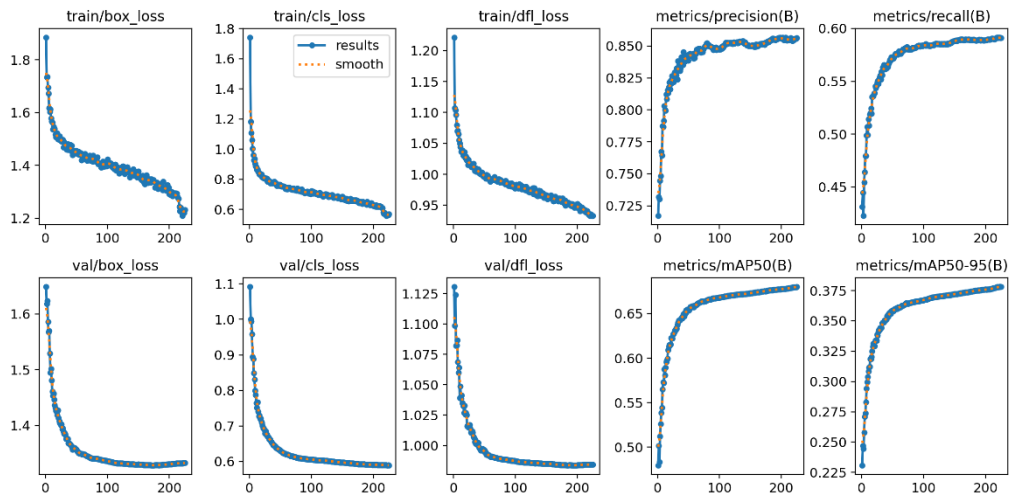
Gambar 9. Kurva Hubungan *Epoch* dan Akurasi

Sumber: <https://forum.knime.com/t/how-to-generate-accuracy-curve-by-epoch-to-detect-overfitting-with-training-validation-data/>

d. Validasi dan Evaluasi

Performa model dipantau menggunakan data validasi dengan metrik seperti *precision*, *recall*, dan *mean Average Precision* (mAP) untuk

memastikan bahwa model tidak hanya bekerja baik pada data latih, tetapi juga mampu melakukan generalisasi pada data baru. Validasi dilakukan pada setiap epoch pelatihan sehingga perbedaan antara *training loss* dan *validation loss* dapat dianalisis. Jika *training loss* terus menurun tetapi *validation loss* meningkat, kondisi ini disebut *overfitting*.



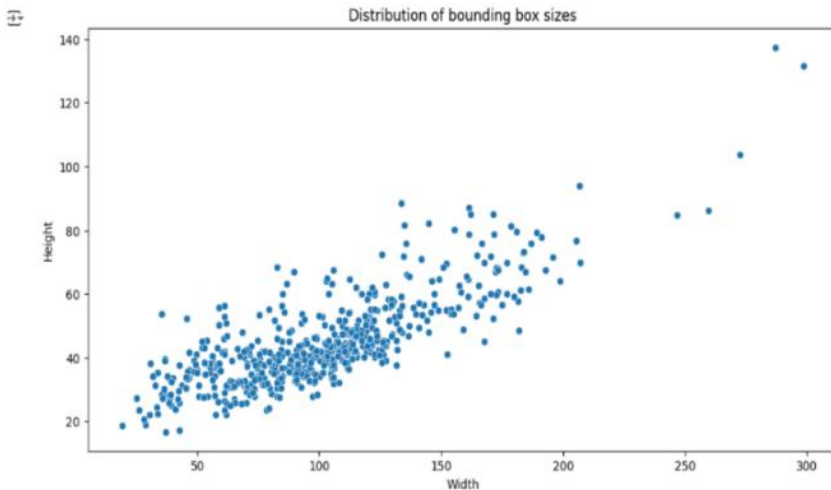
Gambar 10. Kurva *Train*, *Val* dan *Metrics*

Sumber: <https://huggingface.co/AdamCodd/YOLOv11n-face-detection>

e. *Testing Akhir*

Model diuji dengan data baru untuk mengukur kemampuan generalisasi sebelum diterapkan di dunia nyata.

Dalam proses pelatihan, distribusi *bounding box* dalam dataset memiliki peran penting karena memengaruhi kualitas deteksi model. *Bounding box* merepresentasikan posisi dan ukuran objek dalam citra melalui koordinat pusat (x , y) lebar (w), dan tinggi (h). Analisis kurva distribusi *bounding box* umumnya dilakukan untuk memahami variasi ukuran, rasio aspek, maupun persebaran posisi objek di dalam citra. Informasi ini digunakan untuk menentukan *anchor box* yang sesuai melalui metode seperti *k-means clustering*, sehingga model dapat lebih efisien dalam memprediksi objek dengan ukuran dan bentuk yang bervariasi. Misalnya, jika dataset didominasi oleh objek kecil seperti helm atau masker, *anchor box* yang dipilih cenderung berukuran kecil. Sebaliknya, dataset dengan objek besar seperti kendaraan akan menghasilkan *anchor box* dengan dimensi lebih besar. Dengan demikian, kurva distribusi *bounding box* tidak hanya memberikan gambaran karakteristik *dataset*, tetapi juga menjadi dasar penting untuk meningkatkan akurasi deteksi objek pada sistem YOLO (Bochkovskiy et al., 2020).



Gambar 11. Kurva Distribusi *Bounding Box*

1.5.11 Model Hasil (*Weights*)

Dalam proses pelatihan model deteksi objek seperti YOLO, *weights* (bobot) menjadi hasil utama yang diperbarui secara bertahap berdasarkan *loss function*. Selain *loss*, evaluasi hasil model juga menggunakan berbagai metrik seperti *confusion matrix*, kurva evaluasi, serta visualisasi distribusi label.

a. *Mean Average Precision (mAP)*

Salah satu metrik evaluasi yang umum dipakai adalah *mean Average Precision (mAP)*, yang menggabungkan aspek *precision* dan *recall* dalam satu angka yang representatif. Nilai mAP digunakan sebagai metrik evaluasi utama untuk mengetahui seberapa baik *weights* tersebut telah mengoptimalkan kemampuan model dalam mendeteksi objek (Kalambia et al., 2024).

$$AP = \sum_n (R_n - R_{n-1}) \times P_n \dots\dots\dots(1)$$

$$mAP = \frac{1}{C} \sum_{i=1}^C AP_i \dots\dots\dots(2)$$

Dimana:

P_n : *Precision* pada titik ke- n

R_n : *Recall* pada titik ke- n

C : Jumlah kelas objek yang dievaluasi

- **mAP dan IoU:** Metrik mAP merupakan rata-rata dari *Average Precision (AP)* untuk semua kelas objek dalam dataset. AP diperoleh dari grafik *precision–recall curve*, hasil dari variasi *confidence threshold*. Untuk menentukan hasil prediksi *bounding box* model disebut *True Positive (TP)* atau bukan, digunakan ambang *Intersection*

over Union (IoU), yaitu ukuran tingkat tumpang tindih antara prediksi *bounding box* dan *ground truth* menurut:

$$IoU = \frac{\text{Area of Intersection}}{\text{Area of Union}} \dots\dots\dots(3)$$

Suatu prediksi dikatakan benar (TP) apabila $IoU \geq$ ambang yang ditetapkan. Jika tidak, ia dikategorikan sebagai *False Positive* (FP). Selain itu, objek yang tidak terdeteksi dihitung sebagai *False Negative* (FN). Perhitungan mAP mensyaratkan penilaian AP pada setiap kelas, kemudian dirata-rata (Gde and Janapriya, 2023).

- **mAP50:** mAP@0.5 merupakan mAP yang dihitung pada ambang $IoU = 0,5$ (50%). Prediksi *bounding box* dianggap benar jika *overlap* antara prediksi dan *ground truth* minimal 50%. mAP@0.5 memberikan gambaran kemampuan model dalam mendeteksi objek dengan toleransi lokasi yang moderat.
- **mAP50-95:** merupakan metrik yang lebih komprehensif dan ketat dibandingkan mAP@0.5. Metrik ini dihitung dengan mengambil rata-rata AP pada beberapa ambang IoU dari 0,50 sampai 0,95, kemudian dirata-ratakan menjadi skor akhir. Dengan memasukkan ambang IoU yang lebih tinggi, metrik ini menilai seberapa baik model memosisikan *bounding box* secara presisi tinggi terhadap lokasi objek

b. Loss Function

Loss function berfungsi untuk mengukur sejauh mana prediksi model menyimpang dari nilai sebenarnya (*ground truth*) dan menjadi acuan dalam memperbaiki parameter jaringan. Secara umum, komponen *loss* pada YOLO terdiri atas:

- **Localization Loss (Bounding Box Regression Loss):** Bagian ini mengukur seberapa akurat prediksi posisi dan ukuran *bounding box* dibandingkan dengan *ground truth*. Parameter yang dihitung meliputi koordinat pusat (x, y), lebar (w) dan tinggi (h). YOLO menggunakan fungsi *error* seperti *Mean Squared Error* (MSE) atau *Generalized IoU* untuk menghitung perbedaan bentuk dan posisi kotak.

$$Loss_{box} = \frac{\sum error_{box}}{\text{jumlah prediksi}} \dots\dots\dots(4)$$

- **Distribution Focal Loss:** Mengukur seberapa akurat model memprediksi distribusi posisi *bounding box* terhadap objek yang sebenarnya. Jika prediksi distribusi *box* sesuai dengan *ground truth*, maka model “yakin” bahwa *box* tersebut memang menggambarkan objek yang benar. Sebaliknya, jika distribusi tersebut salah atau melenceng jauh, model akan menganggap bahwa kotak tersebut

kurang mewakili objek dan akan memberikan nilai yang lebih rendah.

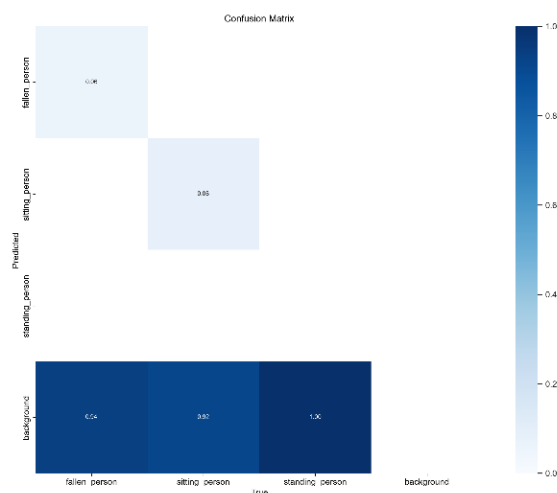
$$Loss_{dfl} = \frac{\sum error_{dfl}}{jumlah\ prediksi} \dots\dots\dots(5)$$

- **Classification Loss:** Setelah objek terdeteksi, YOLO juga harus menentukan kelas dari objek tersebut (misalnya orang, helm, kendaraan, dan lain-lain). Untuk menghitung kesalahan klasifikasi, biasanya digunakan *cross-entropy loss* yang membandingkan distribusi probabilitas kelas hasil prediksi dengan label sebenarnya.

$$Loss_{cls} = \frac{\sum error_{cls}}{jumlah\ prediksi} \dots\dots\dots(6)$$

c. Confusion Matrix

Confusion matrix digunakan untuk mengevaluasi performa klasifikasi dengan membandingkan hasil prediksi terhadap label sebenarnya. Matriks ini terdiri dari:



Gambar 12. Confusion Matrix

- **True Positive (TP):** objek benar terdeteksi sesuai kelasnya,
- **False Positive (FP):** model mendeteksi objek yang salah atau tidak ada,
- **False Negative (FN):** objek nyata tidak terdeteksi,
- **True Negative (TN):** *background* dikenali dengan benar.

d. Kurva Evaluasi

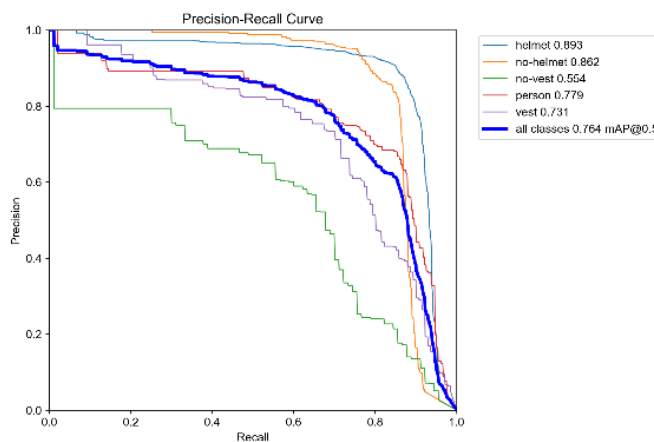
Confusion matrix membandingkan prediksi dengan *ground truth* untuk menghitung TP, FP, FN, dan TN. Dari sini diturunkan *precision*, *recall*,

dan *F1-score*.

- **Precision–Recall (PR) Curve:** PR Curve menampilkan *trade-off* antara *precision* dan *recall* pada berbagai ambang *confidence*. Area di bawah kurva PR (*Average Precision* / AP) sering digunakan untuk menghitung mAP (*mean Average Precision*), yang merupakan metrik utama dalam evaluasi YOLO.

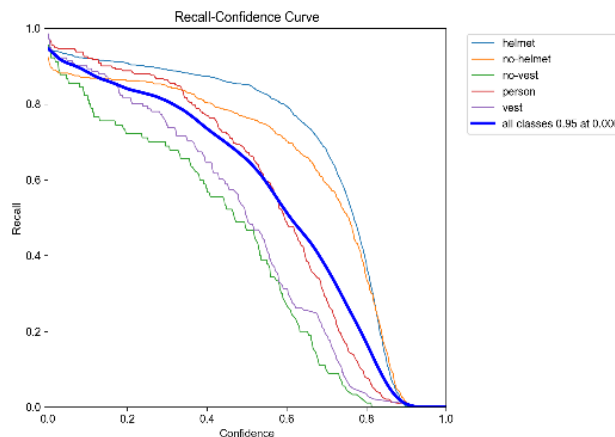
$$Precision = \frac{TP}{TP + FP} \dots\dots\dots(7)$$

$$Recall = \frac{TP}{TP + FN} \dots\dots\dots(8)$$



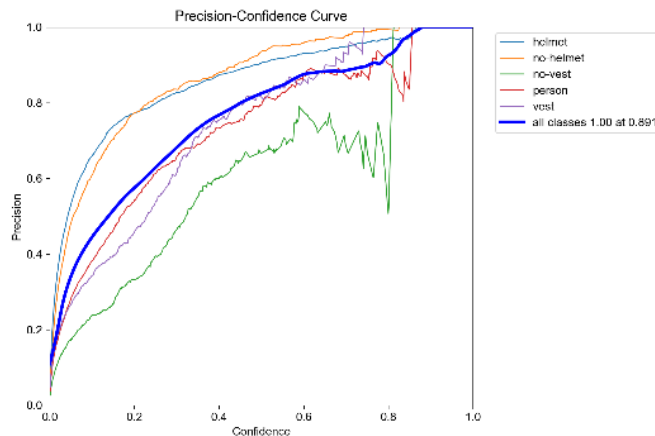
Gambar 13. Precision–Recall (PR) Curve

- **Recall–Confidence Curve:** Kurva ini menunjukkan hubungan antara nilai *confidence threshold* dengan *recall*. Jika ambang batas *confidence* rendah, *recall* cenderung tinggi karena lebih banyak objek terdeteksi, tetapi kemungkinan menghasilkan banyak FP.



Gambar 14. Recall–Confidence Curve

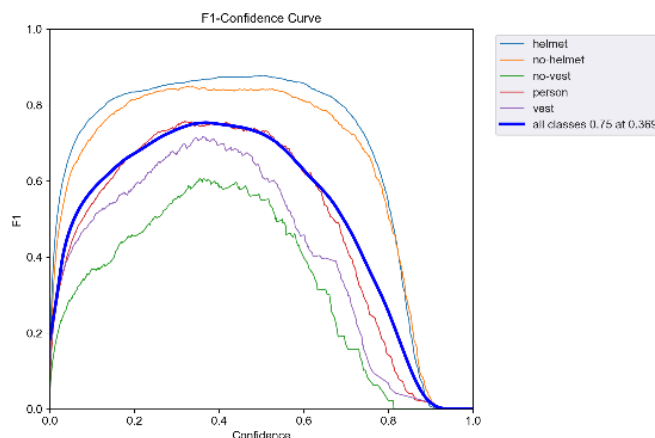
- **Precision–Confidence Curve:** Kurva ini menggambarkan bagaimana *precision* berubah seiring dengan peningkatan nilai *confidence threshold*. *Precision* meningkat jika hanya prediksi dengan *confidence* tinggi yang dipertahankan, meski dapat mengurangi *recall*.



Gambar 15. *Precision–Confidence Curve*

- **F1–Confidence Curve:** *F1-score* merupakan rata-rata harmonis dari *precision* dan *recall*. Kurva ini digunakan untuk menentukan nilai *confidence* optimal yang memberikan keseimbangan terbaik antara *precision* dan *recall*.

$$F1 = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \dots\dots\dots(9)$$

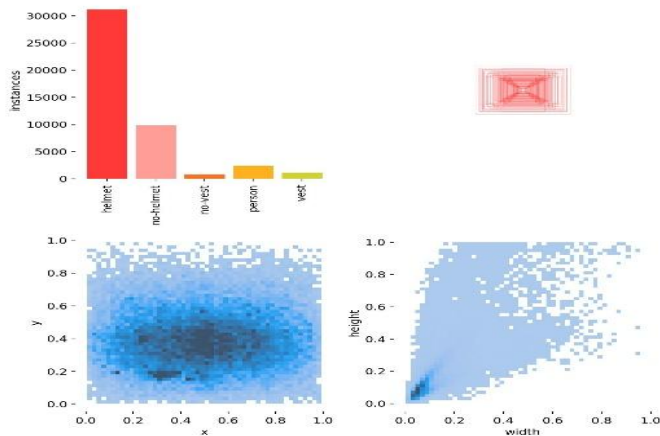


Gambar 16. *F1–Confidence Curve*

e. Grafik Label

Selain metrik numerik, YOLO juga menyediakan visualisasi label untuk mengevaluasi *dataset*. Grafik distribusi jumlah anotasi per kelas. Visualisasi ini membantu melihat apakah *dataset* seimbang (*balanced*) atau

terdapat ketidakseimbangan kelas (*class imbalance*), yang dapat memengaruhi performa model.



Gambar 17. Grafik Label

1.5.12 YOLO Pose Estimation

YOLO *Pose Estimation* merupakan model deteksi yang secara khusus dirancang untuk memperkirakan posisi titik-titik tubuh manusia (*keypoints*) dalam suatu citra atau video. Model ini bekerja dengan mendeteksi keberadaan manusia sekaligus memprediksi koordinat sendi-sendi tubuh seperti kepala, bahu, siku, lutut, dan pergelangan kaki secara langsung (Ding et al., 2024).

Tabel 1. *Keypoints* dalam YOLO *Pose Estimation*

Index	Keypoints
0	Nose
1	Left Eye
2	Right Eye
3	Left Ear
4	Right Ear
5	Left Shoulder
6	Right Shoulder
7	Left Elbow
8	Right Elbow
9	Left Wrist
10	Right Wrist
11	Left Hip
12	Right Hip
13	Left Knee
14	Right Knee
15	Left Ankle
16	Right Ankle

Setiap titik tubuh direpresentasikan dalam bentuk koordinat dua dimensi yang membentuk struktur kerangka (*skeleton model*) manusia. YOLO Pose menggunakan pendekatan *keypoint regression*, di mana jaringan mempelajari hubungan spasial antara bagian tubuh untuk menghasilkan estimasi pose yang presisi. Keunggulan YOLO Pose terletak pada kecepatan dan efisiensinya dalam melakukan estimasi pose secara *real-time* tanpa memerlukan tahapan kompleks. Model dapat diterapkan dalam sistem pengawasan keselamatan, deteksi aktivitas, serta pemantauan posisi tubuh manusia untuk menganalisis postur atau mendeteksi kejadian tertentu seperti jatuh. Dengan kemampuan mendeteksi hingga 17 titik tubuh manusia, YOLO Pose memberikan keseimbangan antara akurasi tinggi dan waktu inferensi yang cepat.



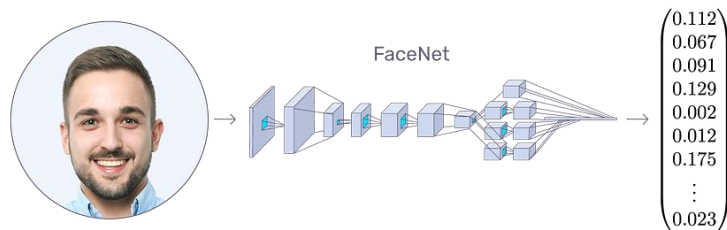
Gambar 18. Kerangka YOLO Pose Estimation

Sumber : <https://so-development.org/how-to-use-yolo11-for-pose-estimation/>

1.5.13 Face Recognition

Face recognition adalah teknologi yang mampu mengidentifikasi dan memverifikasi identitas wajah manusia dengan tingkat akurasi yang tinggi. Sistem ini sangat penting dalam berbagai bidang seperti keamanan, pengawasan, dan sistem absensi otomatis. Salah satu pendekatan terbaru yang banyak digunakan adalah *FaceNet*, sebuah model *deep learning* yang dikembangkan oleh Google. *FaceNet* bekerja dengan mengubah citra wajah menjadi vektor fitur berdimensi tinggi (128-dimensi) yang merepresentasikan wajah tersebut secara unik. Dalam sistem ini, pengukuran kemiripan antara dua wajah dilakukan menggunakan jarak *Euclidean*, di

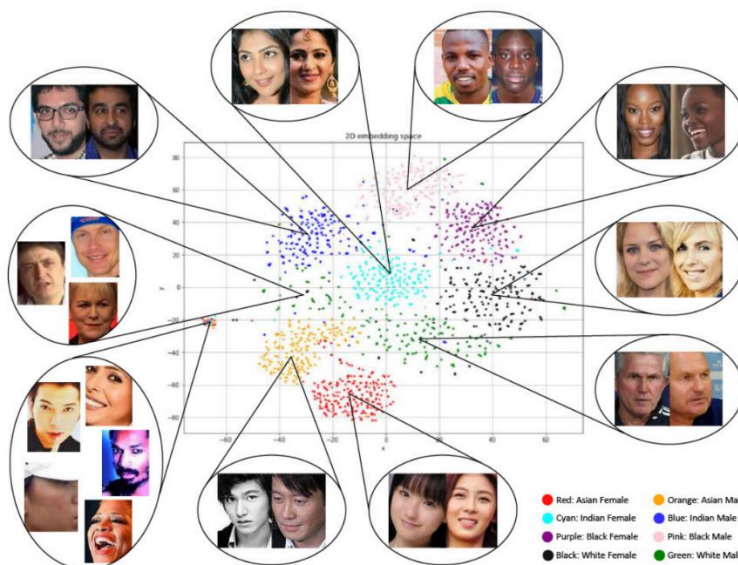
mana jarak yang lebih kecil menunjukkan kemiripan yang lebih tinggi. Keunggulan *FaceNet* terletak pada kemampuannya melakukan *face embedding* langsung dari citra tanpa memerlukan proses alignasi yang rumit, sehingga meningkatkan kecepatan dan akurasi pengenalan wajah secara *real-time* (Rutuja et al., 2022).



Gambar 19. *Embedded Face*

Sumber : <https://medium.com/@culuma/face-recognition-with-facenet-and-mtcnn-11e77240adb6>

Untuk membantu memahami struktur vektor fitur berdimensi tinggi yang dihasilkan oleh *FaceNet*, sering digunakan teknik visualisasi t-SNE. t-SNE merupakan metode reduksi dimensi non-linear yang memetakan data berdimensi tinggi ke dalam ruang dua atau tiga dimensi dengan tetap mempertahankan kedekatan antar data yang saling bertetangga. Melalui visualisasi ini, sebaran dan pengelompokan *embedding* wajah dapat diamati secara intuitif, sehingga memudahkan analisis apakah fitur wajah dari individu yang sama membentuk kluster yang rapat dan terpisah dari individu lain (Frisella et al., 2022).



Gambar 20. Grafik t-SNE

Sumber : <https://www.mdpi.com/2813-0324/3/1/6>

BAB II

METODE PENELITIAN

2.1 Tempat dan Waktu Penelitian

Penelitian dilakukan di Ruang Laboratorium Listrik dan Kendali Departemen Teknik Sistem Perkapalan Fakultas Teknik Universitas Hasanudin.

2.2 Perangkat Keras dan Lunak

2.2.1 Hardware

Tabel 2. *Hardware*

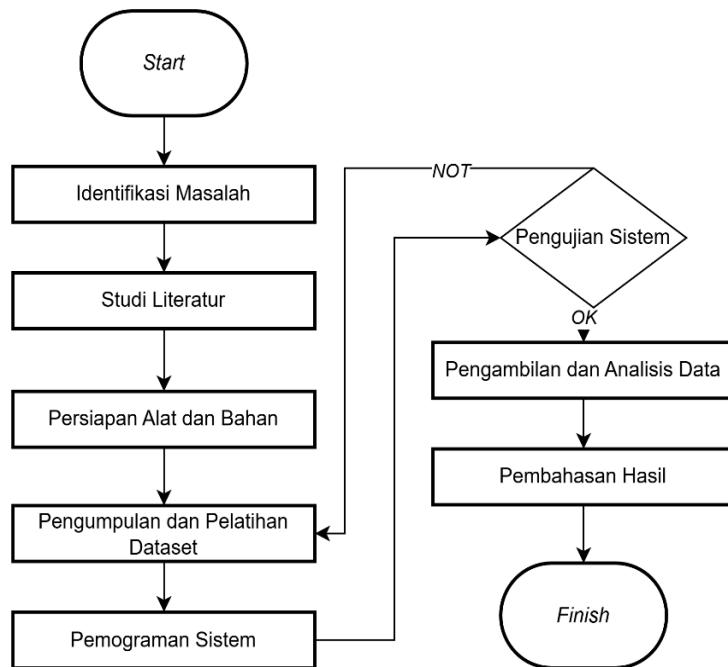
Spesifikasi	Fungsi
MSI Thin GeForce 1650 Laptop	Sebagai laptop utama yang menjalankan sistem deteksi YOLO, pengenalan wajah, dan pengolahan laporan.
HP 245 G7 Notebook PC	Sebagai laptop pembanding performa sistem.
Logitech Brio 500 Web Camera	<i>Input</i> visual untuk proses deteksi APD, jatuh, dan wajah secara <i>real-time</i> .
Speaker Bluetooth	Sistem alarm audio saat terdeteksi pelanggaran atau kejadian jatuh.

2.2.2 Software

Tabel 3. *Software*

Program	Fungsi
Python (PyLibrary)	Bahasa pemrograman utama untuk menjalankan seluruh sistem dan pustaka pendukung.
Ultralytics YOLO	Mendeteksi APD (helm/rompi/sepatu), estimasi pose & <i>keypoint</i> , deteksi wajah, inferensi <i>real-time</i>
OpenCV	Menangkap <i>input</i> dari kamera, memproses citra, dan menampilkan hasil deteksi secara <i>real-time</i> .
FaceNet	Ekstraksi <i>embedding</i> wajah, <i>encoding</i> fitur wajah, perbandingan <i>similarity</i> untuk <i>recognition</i>
NumPy	Operasi komputasi, perhitungan matriks untuk pencocokan wajah, pemrosesan koordinat <i>keypoint</i>
Terminal	Menjalankan skrip utama dan menampilkan log hasil proses sistem secara langsung.

2.3 Tahapan Penelitian



Gambar 21. Diagram Alir Penelitian

2.4 Persiapan Model

Dalam penelitian ini, akan diterapkan tiga model deep learning untuk mencapai tujuan deteksi dan pengenalan. Model YOLOv11m (Ultralytics, 2024) akan dimanfaatkan secara multifungsi, yaitu untuk deteksi wajah (*face detection*) menggunakan model pra-pelatihan yang tersedia, estimasi postur tubuh (*pose estimation*), serta sebagai arsitektur dasar untuk melatih model deteksi Alat Pelindung Diri (APD) secara khusus. Selanjutnya, arsitektur *FaceNet* akan diadopsi untuk membangun sistem pengenalan wajah (*face recognition*) melalui mekanisme *embedding* dan *encoding* yang dikembangkan secara tersendiri.

2.4.1 Dataset Deteksi APD

Digunakan dataset dengan total 2.109 gambar yang diklasifikasikan ke dalam empat kelas objek utama, yaitu *Person*, *Helmet*, *Vest* dan *Shoes*.

Tabel 4. *Detail Dataset APD*

Class	Images
Helmet	1.459
Vest	1.859
Shoes	1.463
Person	1.505

Sebelum proses pelatihan, *dataset* dibagi menjadi tiga subset dengan komposisi proporsional: 70% untuk pelatihan (*train set*) sebanyak 1.477 gambar, 20% untuk validasi (*validation set*) sebanyak 421 gambar, dan 10% untuk pengujian (*test set*) sebanyak 211 gambar.

2.4.2 Keypoints Pose Estimation

Berdasarkan model YOLOv11m *Pose Estimation* (Ultralytics, 2024), terdapat beberapa *keypoints* tubuh yang relevan untuk dianalisis dalam penelitian ini. *Keypoint Nose* (indeks 0) mewakili posisi kepala, yang berguna untuk memperkirakan lokasi helm. *Keypoints Shoulders* dan *Hips* (indeks 5, 6, 11, 12) membentuk vektor torso yang dapat digunakan untuk menganalisis orientasi dan postur tubuh. Sementara itu, *keypoints Ankles* (indeks 15, 16) menunjukkan titik kontak dengan tanah, yang membantu dalam penilaian stabilitas dan posisi kaki terkait deteksi sepatu keselamatan

Tabel 5. *Keypoints* Relevan Model *Pose-Estimation*

Description	Keypoints	Index
Head Position	Nose	0
Vector of Torso	Shoulders and Hips	5, 6, 11, 12
Ground Contact	Ankles	15, 16

2.4.3 Dataset Pengenalan Wajah

Dataset wajah yang digunakan untuk membangun sistem pengenalan terdiri dari tiga subjek dengan jumlah gambar sebagai berikut: Amanda, Ernando, Gazali, Londongallo, Mirna, Pepri, Sri, Syamsidar, Teguh, dan Thessa.

Tabel 6. Detail *Dataset* Wajah

Nama	Jumlah Gambar
Amanda	39
Ernando	37
Gazali	38
Londongallo	35
Mirna	34
Pepri	38
Sri	34
Syamsidar	35
Teguh	36
Thessa	34

Kemudian dilakukan teknik augmentasi yang digunakan meliputi penyesuaian kontras, perubahan kecerahan, perubahan sudut, pembalikan horizontal, dan rotasi gambar akan dijadikan *embedding* dengan model *FaceNet*

2.5 Teknik Pengumpulan dan Analisis Data

Teknik pengumpulan data dalam penelitian ini dilakukan melalui pengujian langsung sistem secara *real-time* dan video rekaman. Data dikumpulkan berdasarkan hasil deteksi sistem yang berjalan dalam skenario pengujian terkontrol, yang didokumentasikan dalam bentuk laporan data.

Analisis data dilakukan dengan tiga pendekatan utama. Pertama, analisis performa sistem dan respons waktu deteksi. Kedua, analisis deteksi dilakukan dengan mengukur konsistensi ukuran, posisi, dan *confidence score* dari objek yang terdeteksi seperti helm, rompi, dan sepatu, serta akurasi *keypoints* pose tubuh. Ketiga, analisis keberhasilan dihitung secara kuantitatif dengan membandingkan jumlah deteksi benar terhadap kondisi aktual. Hasil analisis kemudian dideskripsikan secara kualitatif untuk menilai keandalan sistem dalam beroperasi secara *real-time*.

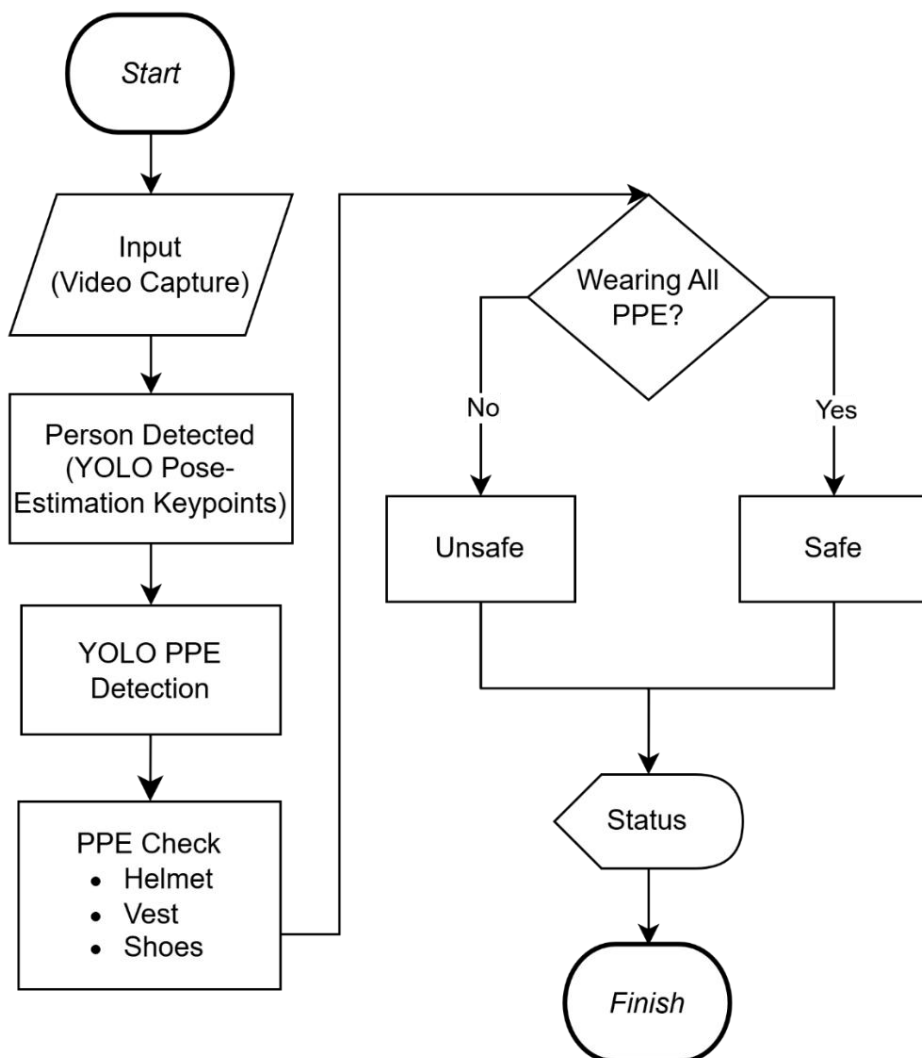
BAB III

HASIL DAN PEMBAHASAN

3.1 Diagram Alir Sistem

3.1.1 Diagram Alir Sistem Deteksi APD

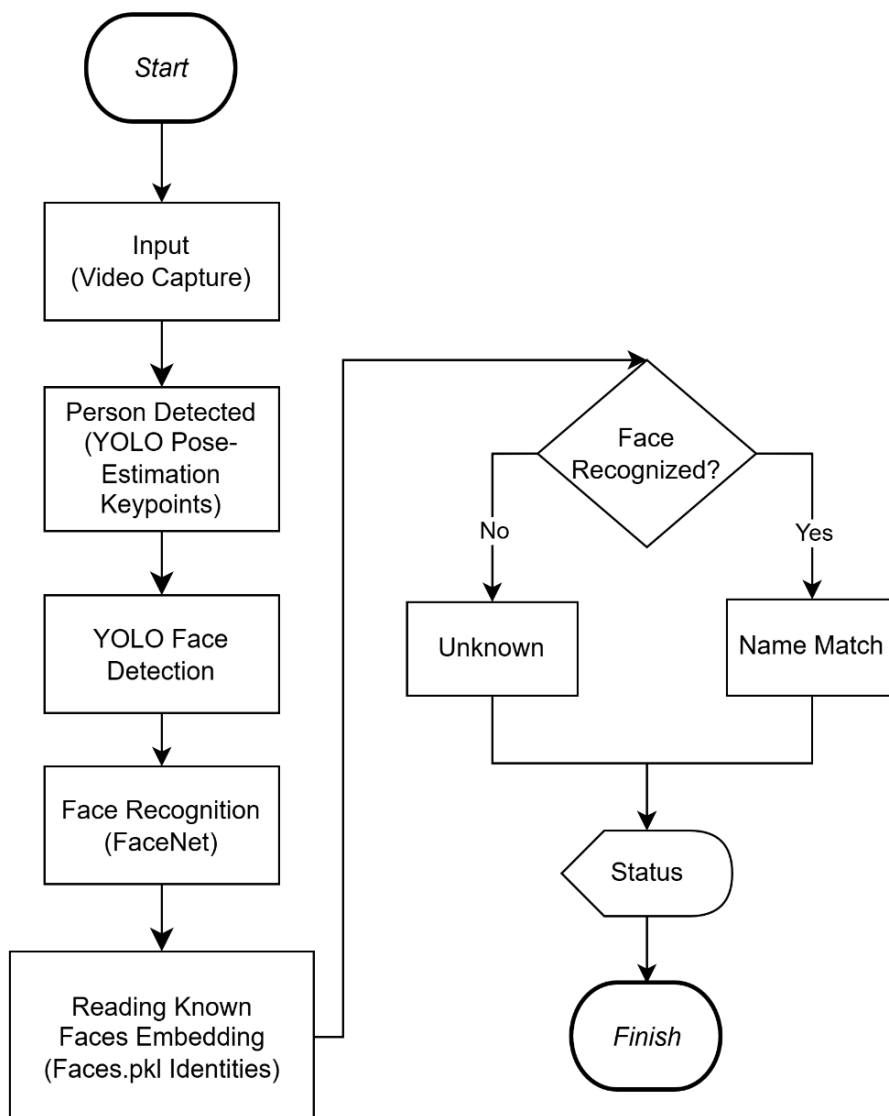
Diagram alir pada Gambar 22 menunjukkan proses deteksi APD yang dimulai dari *input* video. Sistem mendeteksi keberadaan manusia menggunakan YOLO *pose estimation*, kemudian melakukan deteksi APD untuk mengidentifikasi helm, rompi, dan sepatu. Hasil pemeriksaan APD digunakan untuk menentukan status keselamatan, yaitu aman jika seluruh APD terdeteksi dan tidak aman jika terdapat APD yang tidak terdeteksi.



Gambar 22. Diagram Alir Sistem Deteksi APD

3.1.2 Diagram Alir Sistem Deteksi dan Pengenalan Wajah

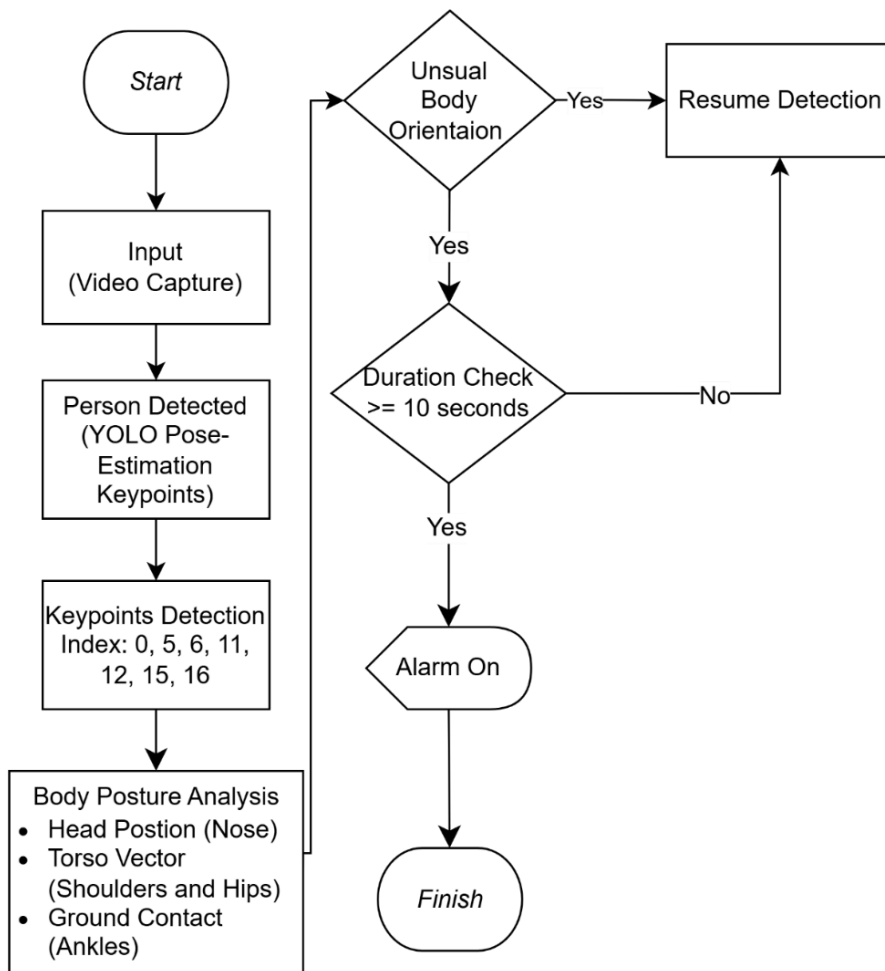
Diagram alir pada Gambar 23 menunjukkan alur kerja sistem untuk bagian deteksi dan pengenalan wajah yang dimulai dari *input* video. Sistem mendeteksi keberadaan manusia menggunakan YOLO pose *estimation*, kemudian melakukan deteksi wajah dengan YOLO. Wajah yang terdeteksi diproses menggunakan *FaceNet* untuk menghasilkan *embedding*, yang selanjutnya dibandingkan dengan data *embedding* wajah yang tersimpan pada file *faces.pkl* yang berisi identitas Amanda, Ernando, Gazali, Londongallo, Mirna, Pepri, Sri, Syamsidar, Teguh dan Thessa. Hasil perbandingan digunakan untuk menentukan status pengenalan, yaitu dikenali dengan kecocokan nama atau tidak dikenali dan diberi label *unknown*, sebelum proses diakhiri.



Gambar 23. Diagram Alir Sistem Pengenalan Wajah

3.1.3 Diagram Alir Sistem Deteksi Jatuh

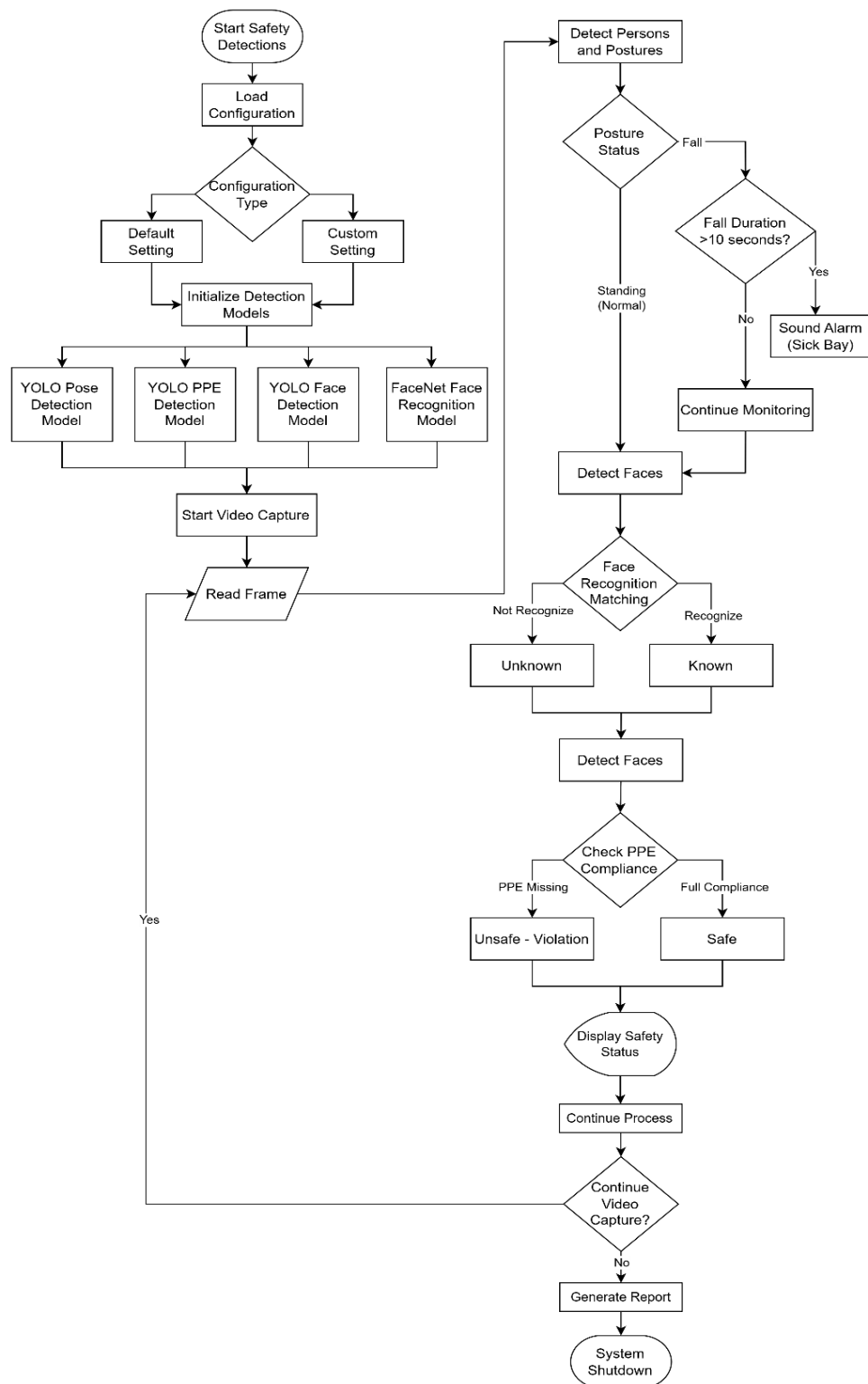
Diagram alir pada Gambar 24 menunjukkan alur kerja sistem untuk bagian deteksi jatuh yang dimulai dari *input* video. Sistem mendeteksi keberadaan manusia menggunakan YOLO *pose estimation* untuk memperoleh titik-titik kunci tubuh, kemudian menganalisis orientasi tubuh. Jika terdeteksi orientasi tubuh yang tidak normal, sistem melakukan pemeriksaan durasi selama minimal 10 detik untuk memastikan kondisi tersebut bersifat berkelanjutan. Jika kondisi terpenuhi, sistem mengaktifkan alarm sebagai indikasi kejadian jatuh, sedangkan jika tidak terpenuhi, proses deteksi dilanjutkan kembali hingga kondisi normal.



Gambar 24. Diagram Alir Sistem Deteksi Jatuh

3.1.4 Diagram Alir Sistem Deteksi Full

Diagram alir pada Gambar 25 menggambarkan alur kerja sistem deteksi full yang mengintegrasikan deteksi postur tubuh, pengenalan wajah, dan pemeriksaan APD dalam satu proses.

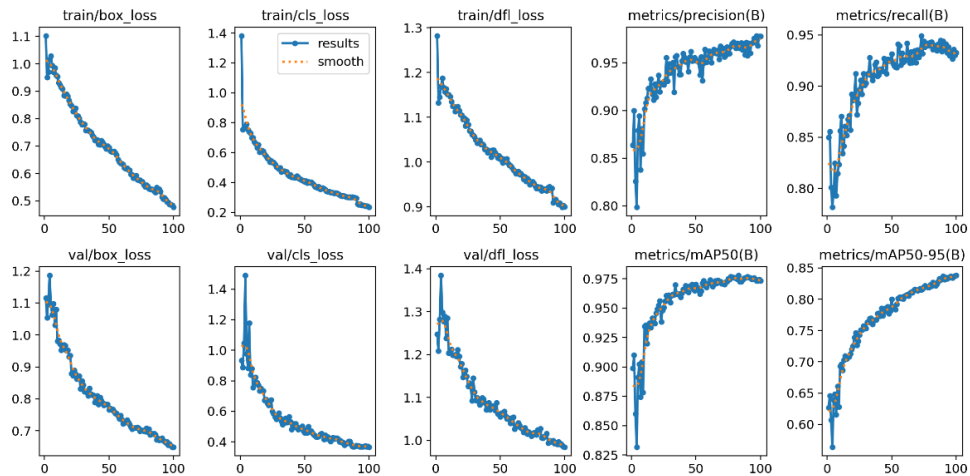


Gambar 25. Diagram Alir Sistem Deteksi Full

3.2 Data Model Deteksi APD

3.2.1 Hasil Training Dataset

Gambar 26 menunjukkan hasil proses pelatihan model deteksi APD hingga *epoch* ke-100.



Gambar 26. Hasil *Training Model Deteksi APD*

Selama proses *training*, terlihat bahwa nilai *train loss* dan *validation loss* untuk *box*, klasifikasi, dan *distribution focal loss* (DFL) mengalami penurunan yang konsisten seiring bertambahnya *epoch*. Pola ini menunjukkan bahwa model mampu mempelajari fitur objek APD dengan baik tanpa mengalami lonjakan *loss* yang signifikan pada data validasi. Stabilitas kurva *training* dan validasi mengindikasikan bahwa proses pembelajaran berjalan dengan baik dan tidak menunjukkan *overfitting* yang berlebihan. Hasil terbaik dapat dilihat dengan nilai yang ditunjukkan di Tabel 7 berikut untuk metrik *best.pt* (*Epoch 100*)

Tabel 7. Metrik *best.pt* (*epoch 100*)

Metrik	Value
<i>train/box_loss</i>	0,47704
<i>train/cls_loss</i>	0,23485
<i>train/dfl_loss</i>	0,89998
<i>metrics/precision(B)</i>	0,9778
<i>metrics/recall(B)</i>	0,93261
<i>metrics/mAP50(B)</i>	0,97316
<i>metrics/mAP50-95(B)</i>	0,83811
<i>val/box_loss</i>	0,64962
<i>val/cls_loss</i>	0,36639
<i>val/dfl_loss</i>	0,98492

3.2.2 Distribusi *Bounding Box* Deteksi APD

Dalam Gambar 27, ditunjukkan model objek tersebut dilatih menggunakan bentuk label standar dengan *bounding box* dan koordinat pusat. Ukuran gambar yang digunakan adalah 640x640. Berikut adalah isi dari *file label .txt* dari gambar 28 yang kemudian dikonversikan:



Gambar 27. Objek yang Diberikan Label dan Diidentifikasi

YOLO format: *class_id x_center y_center width height* yang kemudian konversi dengan kali ukuran gambar sehingga:

$$(x, y, w, h) = (x_n, y_n, w_n, h_n) \times \text{img size} \dots \dots \dots (10)$$

- Helmet: (0 0,603906 0 89844 0,154688 0,123438)

$$x = 0,603906 \times 640 = 386,5$$

$$y = 0 89844 \times 640 = 57,5$$

$$w = 0,154688 \times 640 = 99$$

$$h = 0,123438 \times 640 = 79$$

Posisi pusat dan dimensi *bounding box* adalah (386,5 x 57,5 & 99 x 79)

- Shoes 1: (3 0,451562 0,935156 0,156250 0,129688)

$$x = 0,451562 \times 640 = 289$$

$$y = 0,935156 \times 640 = 598,5$$

$$w = 0,156250 \times 640 = 100$$

$$h = 0,129688 \times 640 = 83$$

Posisi pusat dan dimensi *bounding box* adalah (289 x 598,5 & 100 x 83)

- Shoes 2: (3 0,736719 0,941406 0,167188 0,104688)

$$x = 0,736719 \times 640 = 471,5$$

$$y = 0,941406 \times 640 = 602,5$$

$$w = 0,167188 \times 640 = 107$$

$$h = 0,104688 \times 640 = 67$$

Posisi pusat dan dimensi *bounding box* adalah (471,5 x 602,5 & 107 x 67)

- Vest: (4 0,600000 0,356250 0,268750 0,334375)

$$x = 0,600000 \times 640 = 384$$

$$y = 0,356250 \times 640 = 228$$

$$w = 0,268750 \times 640 = 172$$

$$h = 0,334375 \times 640 = 214$$

Posisi pusat dan dimensi *bounding box* adalah (384 x 228 & 172 x 214)

- Person: (2 0,490625 0,509375 0,862500 0,981250)

$$x = 0,490625 \times 640 = 314$$

$$y = 0,509375 \times 640 = 326$$

$$w = 0,862500 \times 640 = 552$$

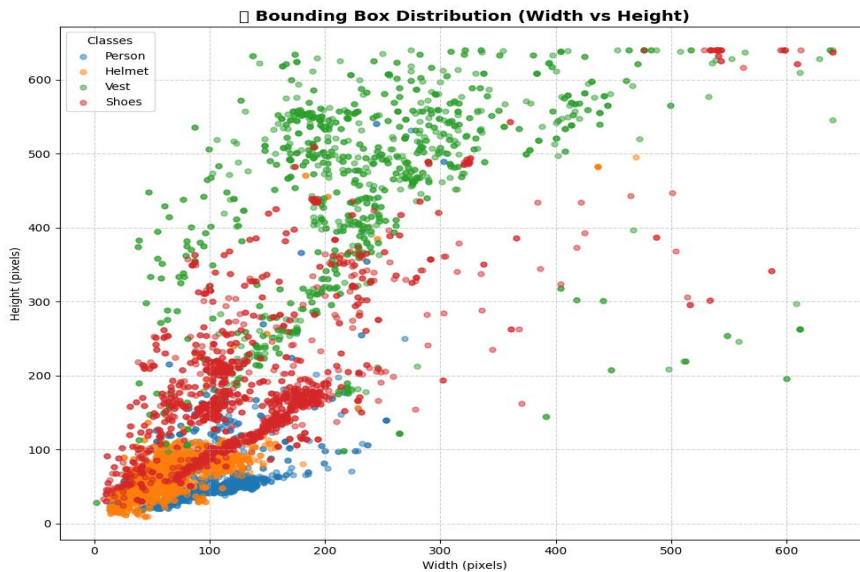
$$h = 0,981250 \times 640 = 628$$

Posisi pusat dan dimensi *bounding box* adalah (314 x 326 & 552 x 628)

Berdasarkan konversi koordinat *bounding box* pada salah satu gambar *dataset*, dapat diamati karakteristik ukuran dan posisi setiap objek yang menjadi dasar untuk analisis distribusi visual. Variasi dimensi yang signifikan antara kelas objek, seperti perbandingan ukuran person yang mendominasi area dengan *helmet* yang lebih kompak, merefleksikan kompleksitas objek dalam lingkungan nyata.

Gambar 28 menggambarkan grafik sebaran ukuran *bounding box* (lebar dan tinggi) dari setiap kelas objek pelatihan model. Titik biru (person) tersebar cukup luas karena ukuran manusia berubah sesuai jarak kamera, titik hijau (vest) cenderung besar karena mengikuti area torso, titik oranye (*helmet*) lebih kecil dan rapat karena hanya mencakup kepala, sedangkan titik merah (*shoes*) memiliki variasi lebih tinggi akibat

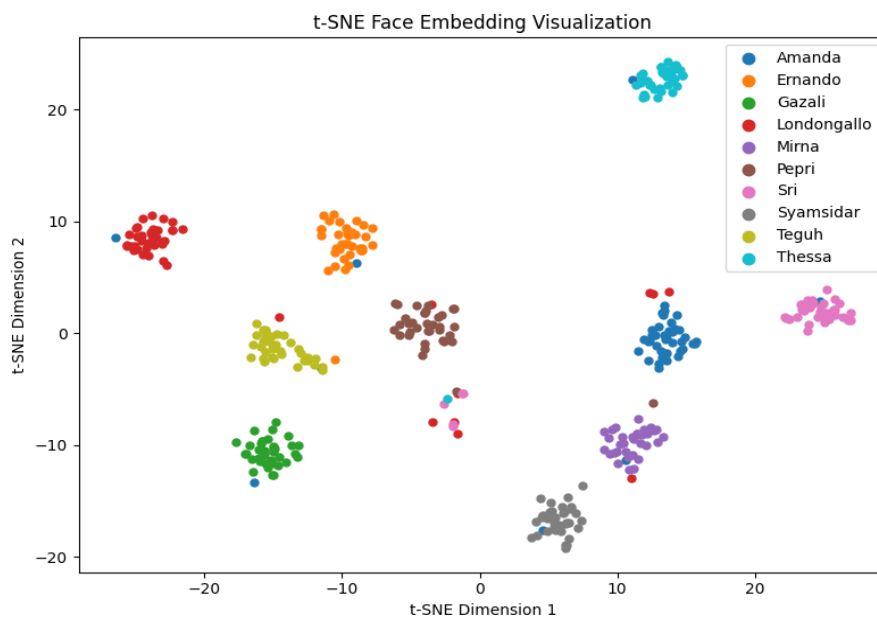
ukuran sepatu yang kecil namun rentan berubah karena sudut kamera. Pola ini menunjukkan bahwa *dataset* memiliki variasi ukuran objek yang cukup baik.



Gambar 28. Distribusi *Bounding Box* (W x H) Model Hasil *Training*

3.3 Data Model Face Recognition

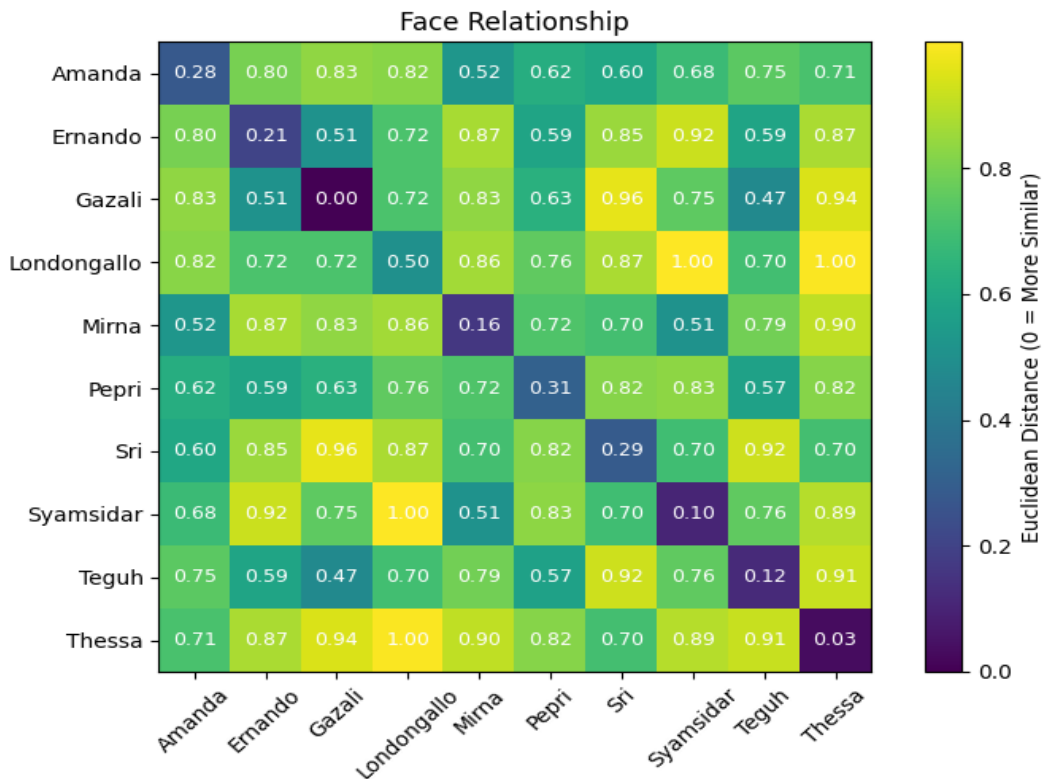
Hasil visualisasi embedding wajah menggunakan grafik t-SNE (Gambar 29) dan matriks *Euclidean Distance* (Gambar 29) menunjukkan pola yang saling mendukung dalam menggambarkan hubungan antar identitas.



Gambar 29. Hasil *Encoding* Wajah

Pada grafik t-SNE, setiap identitas membentuk *cluster* yang terpisah dan relatif kompak, di mana *embedding* dari individu yang sama cenderung mengelompok pada area yang berdekatan, sedangkan *embedding* dari individu yang berbeda berada pada posisi yang berjauhan. Jarak antar *cluster* pada visualisasi ini mencerminkan tingkat perbedaan karakteristik wajah antar identitas, sementara kedekatan beberapa *cluster* menunjukkan adanya kemiripan relatif fitur wajah tanpa adanya tumpang tindih yang signifikan.

Pola Visual tersebut diperkuat dengan nilai *euclidean distance* yang ditampilkan dalam Gambar 30. Nilai jarak yang lebih kecil umumnya muncul pada pasangan *embedding* dari identitas yang sama, sedangkan nilai yang lebih besar ditemukan pada pasangan antar identitas yang berbeda. Beberapa pasangan identitas menunjukkan nilai jarak yang relatif lebih rendah, yang sejalan dengan posisi *cluster* yang berdekatan pada grafik t-SNE. Keselarasan antara visualisasi t-SNE dan nilai *euclidean distance* ini menunjukkan bahwa *embedding* wajah yang dihasilkan memiliki konsistensi dan kemampuan diskriminatif yang baik dalam merepresentasikan perbedaan antar identitas.



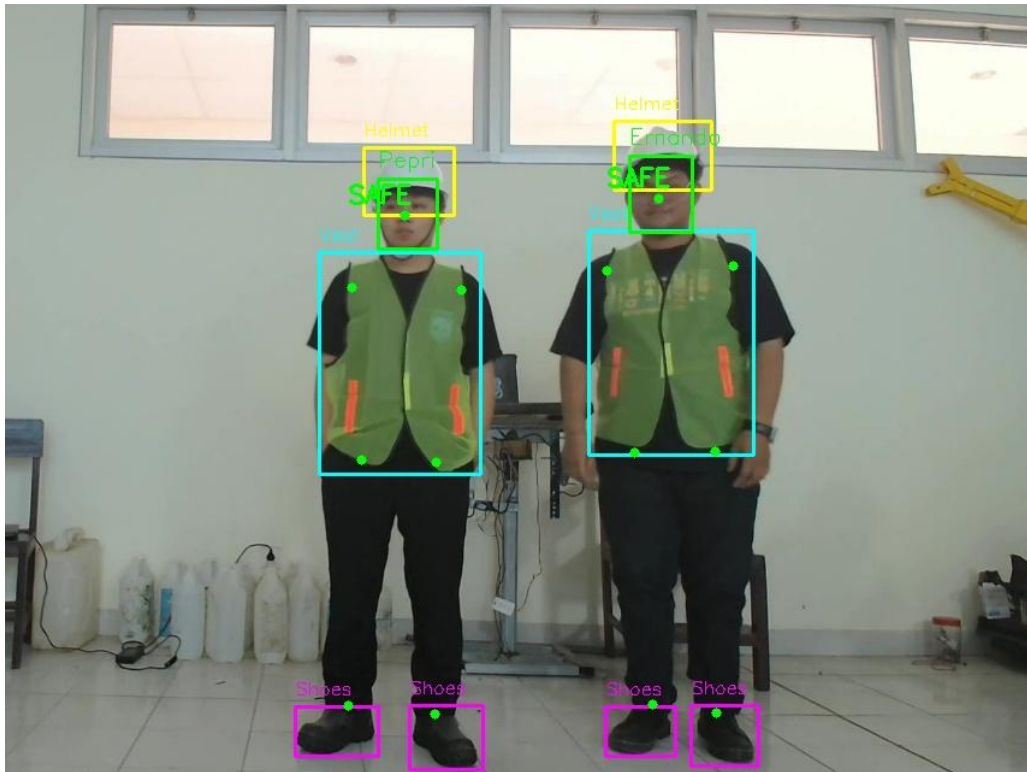
Gambar 30. *Euclidean Distance*

3.4 Pengujian Sistem Secara *Real-Time*

Pengujian dilakukan untuk menilai kinerja YOLO dalam mendeteksi APD pada berbagai kondisi, termasuk variasi kelengkapan helm, rompi, sepatu, serta situasi dengan banyak orang dalam satu frame. Hasil representatif disajikan dengan

membandingkan kondisi aktual dan deteksi sistem menggunakan tanda ☒ untuk benar dan ☐ untuk salah. Evaluasi mencakup kategori *Status*, *Face*, *Vest*, *Helmet*, *Left Shoe*, *Right Shoe*, *Standing*, dan *Fall*. Berikut 5 Pengujian dan selanjutnya berada di Lampiran.

3.4.1 Pengujian 1



Gambar 31. Pengujian dengan Kondisi APD Lengkap (2 Orang)

Berdasarkan hasil Pengujian 1 (Gambar 31) dengan kondisi APD lengkap pada dua orang, sistem berhasil mendeteksi kedua individu yaitu Pepri dan Ernando dengan status SAFE. Seluruh elemen APD yang meliputi helm, rompi, serta sepatu kiri dan kanan terdeteksi dengan benar sesuai kondisi sebenarnya, sehingga kedua subjek dinyatakan berada dalam kondisi aman. Nilai *confidence* yang diperoleh berada pada rentang 67% hingga 93%, dengan nilai tertinggi pada deteksi rompi (91%–93%), sedangkan sepatu kiri memiliki nilai *confidence* terendah yaitu 67%–68%, namun masih dapat terdeteksi dengan baik.

Durasi deteksi wajah berada pada kisaran 100 hingga 103 ms, sementara waktu deteksi untuk *helm*, *vest*, dan sepatu relatif stabil yaitu sekitar 54 ms pada kedua subjek. Ukuran *bounding box* menunjukkan variasi sesuai objek, dengan wajah berada pada kisaran 49 × 58 hingga 52 × 63 pixel, helm pada kisaran 74 × 56 hingga 81 × 56 pixel, dan *vest* berada pada ukuran terbesar yaitu sekitar 134 × 184 hingga 136 × 185 pixel. Sementara itu, sepatu kiri dan kanan berada pada rentang 56 × 49 hingga 61 × 53 pixel dan 57 × 41 hingga 59 × 47 pixel.

Tabel 8. Detail Deteksi Pengujian 1

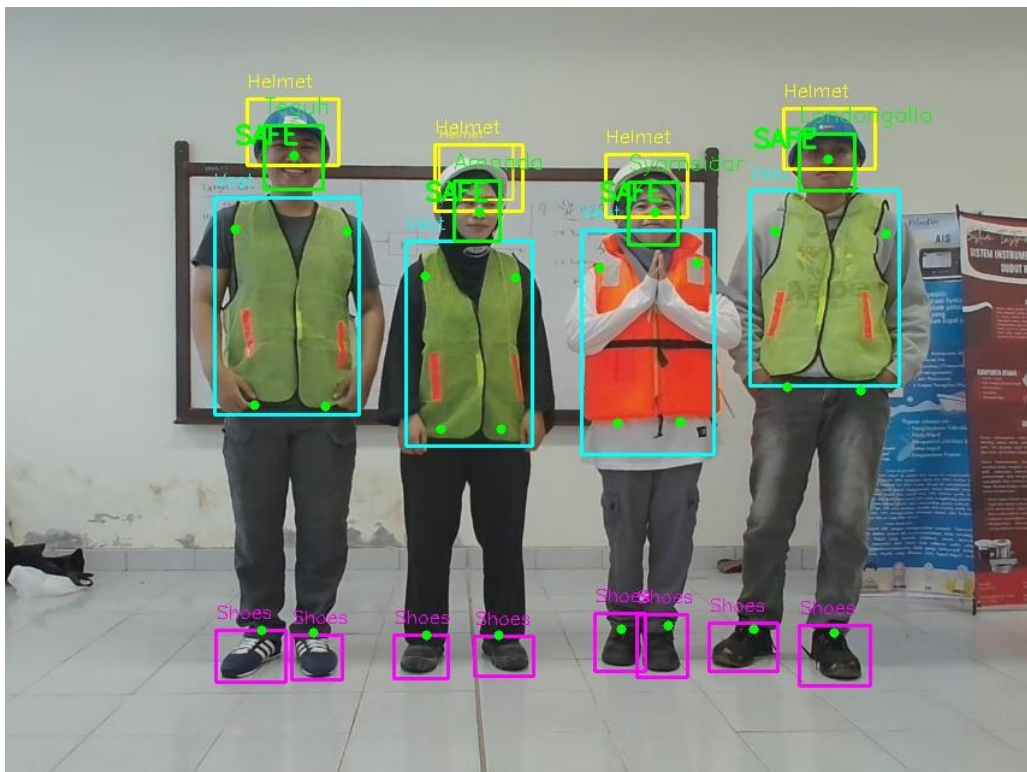
	Person	Pepri	Ernando
	Status	SAFE	SAFE
	Result	☑	☑
Face	Result	☑	☑
	Conf (%)	92	89
	Dur (ms)	103	100
	BBox W x H (pixel)	49 x 58	52 x 63
Helmet	Result	☑	☑
	Conf (%)	84	86
	Dur (ms)	54	54
	BBox W x H (pixel)	74 x 56	81 x 56
Vest	Result	☑	☑
	Conf (%)	91	93
	Dur (ms)	54	54
	BBox W x H (pixel)	134 x 184	136 x 185
Shoe (L)	Result	☑	☑
	Conf (%)	68	67
	Dur (ms)	54	54
	BBox W x H (pixel)	61 x 53	56 x 49
Shoe (R)	Result	☑	☑
	Conf (%)	76	78
	Dur (ms)	54	54
	BBox W x H (pixel)	59 x 47	57 x 41

Pada bagian pengujian postur tubuh, sistem juga berhasil mendeteksi posisi berdiri (*standing*) untuk Pepri dan Ernando sesuai kondisi sebenarnya dengan *confidence* sebesar 92% dan durasi pemrosesan 19 ms. Pola sebaran koordinat *keypoints* dari bagian kepala hingga kaki menunjukkan struktur tubuh yang konsisten, di mana titik bahu, pinggul, dan pergelangan kaki berada pada posisi yang proporsional, sehingga mendukung identifikasi postur berdiri dan penetapan status SAFE pada kedua subjek.

Tabel 9. Detail Deteksi Posture Tubuh Pengujian 1

Person		Pepri	Ernando
Pose		Standing	Standing
Result		☒	☒
Duration (ms)		19	19
Confidence (%)		92	92
Keypoints Coordinate X x Y (Pixel)	Nose	333 x 176	543 x 162
	(L) Shoulder	379 x 238	605 x 217
	(R) Shoulder	289 x 236	501 x 221
	(L) Hip	358 x 380	590 x 371
	(R) Hip	297 x 379	523 x 372
	(L) Ankle	357 x 589	592 x 589
	(R) Ankle	285 x 582	538 x 582

3.4.2 Pengujian 2



Gambar 32. Pengujian dengan Kondisi Safety Vest dan Life Vest (4 Orang)

Berdasarkan hasil Pengujian 2 (Gambar 32), sistem berhasil mendeteksi keempat individu yaitu Teguh, Amanda, Syamsidar, dan Londongallo dengan status SAFE. Seluruh elemen APD terdeteksi dengan baik sesuai kondisi sebenarnya. Nilai *confidence* berada pada rentang 57% hingga 94%, dengan nilai tertinggi pada deteksi wajah dan rompi, sedangkan sepatu memiliki nilai lebih rendah namun tetap terdeteksi. Durasi deteksi wajah berkisar 87–96 ms, sementara deteksi APD stabil pada 51 ms. Ukuran *bounding box* menunjukkan variasi sesuai objek, dengan *vest* sebagai objek terbesar dan sepatu sebagai yang terkecil.

Tabel 10. Detail Deteksi Pengujian 2

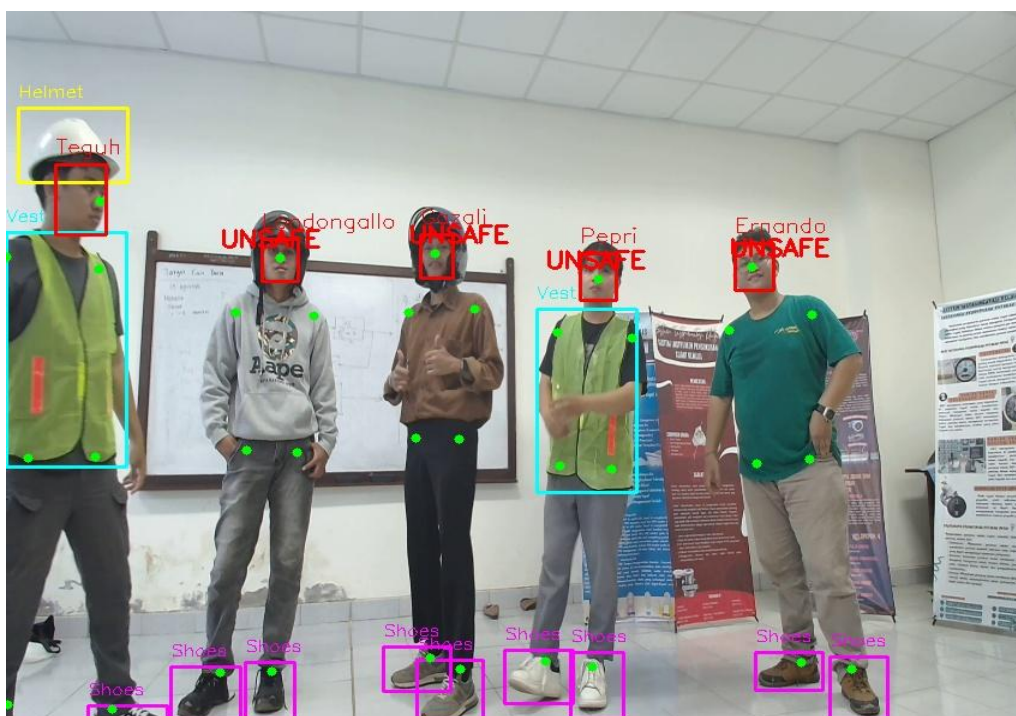
Person		Teguh	Amanda	Syamsidar	Londong
Results		☑	☑	☑	☑
Status		SAFE	SAFE	SAFE	SAFE
Face	Result	☑	☑	☑	☑
	Conf (%)	94	80	80	90
	Dur (ms)	96	87	89	93
	BBox W x H (pixel)	49 x 53	46 x 47	38 x 50	41 x 52
Helmet	Result	☑	☑	☑	☑
	Conf (%)	86	88	76	64
	Dur (ms)	51	51	51	51
	BBox W x H (pixel)	75 x 54	75 x 50	72 x 54	68 x 51
Vest	Result	☑	☑	☑	☑
	Conf (%)	92	92	91	84
	Dur (ms)	51	51	51	51
	BBox W x H (pixel)	120 x 179	124 x 162	105 x 169	110 x 185
Shoe (L)	Result	☑	☑	☑	☑
	Conf (%)	83	66	59	86
	Dur (ms)	51	51	51	51
	BBox W x H (pixel)	41 x 37	58 x 50	48 x 33	41 x 52
Shoe (R)	Result	☑	☑	☑	☑
	Conf (%)	85	57	80	80
	Dur (ms)	51	51	51	51
	BBox W x H (pixel)	49 x 53	46 x 47	38 x 50	41 x 52

Pada pengujian postur tubuh, sistem berhasil mendeteksi posisi berdiri pada seluruh subjek dengan *confidence* 92% dan durasi 22 ms. Sebaran koordinat *keypoints* menunjukkan struktur tubuh yang konsisten dan proporsional, sehingga mendukung identifikasi postur berdiri dan penetapan status SAFE tanpa indikasi kesalahan deteksi.

Tabel 11. Detail Deteksi Postur Tubuh Pengujian 2

Person		Teguh	Amanda	Syamsidar	Londongallo
Pose		Standing	Standing	Standing	Standing
Result		✓	✓	✓	✓
Duration (ms)		22	22	22	22
Confidence (%)		92	92	92	92
Koordinat Keypoints X x Y (Pixel)	Nose	240 x 124	395 x 171	540 x 170	684 x 126
	(L) Shoulder	284 x 187	424 x 224	576 x 213	732 x 189
	(R) Shoulder	192 x 184	350 x 224	495 x 216	641 x 186
	(L) Hip	266 x 331	412 x 351	562 x 344	711 x 318
	(R) Hip	208 x 330	363 x 351	509 x 346	651 x 315
	(L) Ankle	256 x 520	410 x 521	551 x 514	689 x 519
	(R) Ankle	213 x 518	350 x 521	512 x 516	623 x 516

3.4.3 Pengujian 3



Gambar 33. Pengujian dengan Berbagai Kondisi *Helmet* (5 Orang)

Berdasarkan hasil Pengujian 3 (Gambar 33), sistem mendeteksi lima individu dengan status UNSAFE akibat ketidaklengkapan APD. Pada aspek helm, hanya Teguh yang terdeteksi menggunakan helm proyek (87%), sementara Londongallo dan Gazali menggunakan *helmet* motor yang tidak dikenali sebagai APD, dan Pepri serta Ernando tidak menggunakan helm. Selain itu, rompi hanya terdeteksi pada Pepri dan Teguh, serta beberapa sepatu tidak terdeteksi. Nilai *confidence* wajah berada pada 72%–90% dengan durasi 98–112 ms, sedangkan deteksi APD lainnya stabil di 52 ms.

Tabel 12 Detail Deteksi Pengujian 3

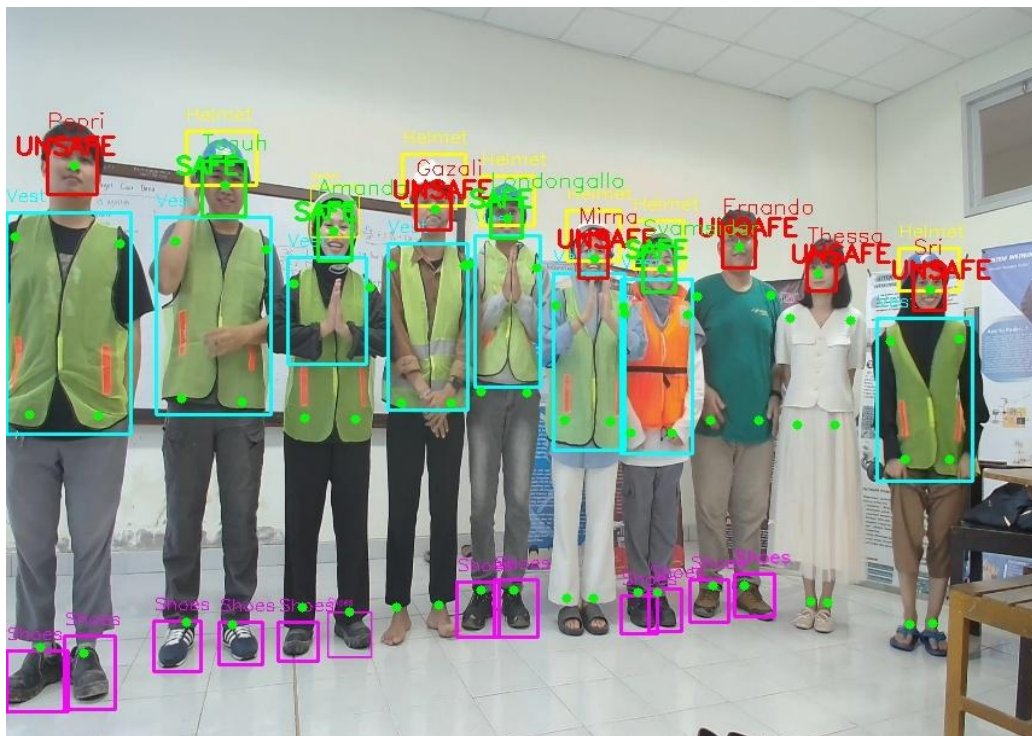
Person		Ernando	Londong	Pepri	Teguh	Gazali
Results		☒	☒	☒	☒	☒
Status		UNSAFE	UNSAFE	UNSAFE	UNSAFE	UNSAFE
Face	Result	☒	☒	☒	☒	☒
	Conf (%)	90	72	78	79	77
	Dur (ms)	98	105	112	100	104
	BBox W x H (pixel)	32 x 42	30 x 38	30 x 42	42 x 63	27 x 39
Helmet	Result	☒	☒	☒	☒	☒
	Conf (%)	-	-	-	87	-
	Dur (ms)	52	52	52	52	52
	BBox W x H (pixel)	-	-	-	90 x 67	-
Vest	Result	☒	☒	☒	☒	☒
	Conf (%)	-	-	91	93	-
	Dur (ms)	52	52	52	52	52
	BBox W x H (pixel)	-	-	82 x 165	100 x 210	-
Shoe (L)	Result	☒	☒	☒	☒	☒
	Conf (%)	81	85	60	45	65
	Dur (ms)	52	52	52	52	52
	BBox W x H (pixel)	47 x 56	41 x 52	43 x 59	65 x 14	55 x 40
Shoe (R)	Result	☒	☒	☒	☒	☒
	Conf (%)	86	87	87	-	35
	Dur (ms)	52	52	52	52	52
	BBox W x H (pixel)	55 x 34	59 x 34	55 x 42	-	57 x 48

Pada pengujian postur tubuh, seluruh subjek terdeteksi dalam posisi berdiri dengan *confidence* 92% dan durasi 26 ms. Sebaran koordinat *keypoints* menunjukkan struktur tubuh yang masih utuh dan proporsional, meskipun terdapat beberapa titik yang kurang stabil pada kondisi tertentu.

Tabel 13. Detail Deteksi Postur Tubuh Pengujian 3

Person		Ernando	Londong	Pepri	Teguh	Gazali
Pose		Standing	Standing	Standing	Standing	Standing
Results		✓	✓	✓	✓	✓
Duration (ms)		26	26	26	26	26
Confidence (%)		92	92	92	92	92
Keypoints Coordinate X x Y (Pixel)	Nose	622 x 230	228 x 222	494 x 241	78 x 171	357 x 218
	(L) Shoulder	672 x 274	256 x 275	522 x 295	77 x 232	391 x 269
	(R) Shoulder	603 x 287	192 x 271	459 x 290	0 x 222	336 x 271
	(L) Hip	670 x 403	243 x 397	505 x 413	75 x 401	378 x 385
	(R) Hip	626 x 408	200 x 396	462 x 409	18 x 403	342 x 384
	(L) Ankle	705 x 593	222 x 595	489 x 590	88 x 628	379 x 593
	(R) Ankle	662 x 587	179 x 596	450 x 585	1 x 624	354 x 583

3.4.4 Pengujian 4



Gambar 34. Pengujian dengan 10 Orang dalam Berbagai Kondisi

Tabel 14. Detail Deteksi Pengujian 4

	Person	Pepri	Teguh	Amanda	Ernando	Mirna
	Results	☒	☒	☒	☒	☒
	Status	UNSAFE	SAFE	SAFE	UNSAFE	SAFE
Face	Result	☒	☒	☒	☒	☒
	Conf (%)	80	88	78	91	84
	Dur (ms)	76	79	76	78	76
	BBox W x H (pixel)	41 x 48	37 x 48	28 x 48	27 x 37	24 x 38
Helmet	Result	☒	☒	☒	☒	☒
	Conf (%)	-	86	84	-	86
	Dur (ms)	64	64	64	64	64
	BBox W x H (pixel)	-	58 x 47	44 x 40	42 x 42	44 x 44
Vest	Result	☒	☒	☒	☒	☒
	Conf (%)	92	90	57	46	28
	Dur (ms)	64	64	64	64	64
	BBox W x H (pixel)	102 x 193	94 x 172	64 x 93	59 x 151	58 x 153
Shoe (L)	Result	☒	☒	☒	☒	☒
	Conf (%)	74	76	0	73	74
	Dur (ms)	64	64	64	64	64
	BBox W x H (pixel)	41 x 63	36 x 36	34 x 32	30 x 33	36 x 49
Shoe (R)	Result	☒	☒	☒	☒	☒
	Conf (%)	86	69	0	74	-
	Dur (ms)	64	64	64	64	64
	BBox W x H (pixel)	38 x 44	33 x 34	33 x 35	35 x 47	-

Tabel 15. Lanjutan tabel 14

Person		Gazali	Sri	Thessa	Londong	Syamsidar
Results		✓	✓	✓	✓	✓
Status		UNSAFE	SAFE	SAFE	UNSAFE	SAFE
Face	Result	✓	✓	✓	✓	✓
	Conf (%)	84	80	90	79	88
	Dur (ms)	78	78	75	76	75
	BBox W x H (pixel)	30 x 38	26 x 39	24 x 32	26 x 34	27 x 40
Helmet	Result	✓	✓	✓	✓	✓
	Conf (%)	86	78	-	85	79
	Dur (ms)	64	64	64	64	64
	BBox W x H (pixel)	52 x 45	50 x 37	-	43 x 42	38 x 40
Vest	Result	✓	✓	✓	✓	✓
	Conf (%)	90	91	-	74	83
	Dur (ms)	64	64	64	64	64
	BBox W x H (pixel)	65 x 144	79 x 141	-	59 x 138	67 x 156
Shoe (L)	Result	✓	✓	✓	✓	✓
	Conf (%)	-	-	-	68	64
	Dur (ms)	64	64	64	64	64
	BBox W x H (pixel)	-	-	-	31 x 40	28 x 32
Shoe (R)	Result	✓	✓	✓	✓	✓
	Conf (%)	30 x 35	30 x 35	30 x 35	30 x 35	30 x 35
	Dur (ms)	30 x 35	30 x 35	30 x 35	30 x 35	30 x 35
	BBox W x H (pixel)	30 x 35	30 x 35	30 x 35	30 x 35	30 x 35

Berdasarkan hasil Pengujian 4 (Gambar 34), sistem berhasil mendeteksi 10 individu dengan status yang bervariasi, yaitu Teguh, Amanda, Mirna, Sri, Londongallo, dan Syamsidar berstatus SAFE, sedangkan Pepri, Ernando, Gazali, dan Thessa berstatus UNSAFE akibat ketidaklengkapan APD. Helm tidak terdeteksi pada Pepri, Ernando, dan Thessa, rompi tidak terdeteksi pada Thessa, serta beberapa sepatu tidak

terdeteksi terutama pada Gazali dan Sri. Nilai confidence wajah berada pada kisaran 78% hingga 91% dengan durasi sekitar 75–79 ms, sedangkan deteksi APD lainnya stabil pada 64 ms. Variasi bounding box menunjukkan vest sebagai objek terbesar, sementara wajah dan sepatu berukuran lebih kecil. Pengujian ini menunjukkan sistem mampu menangani banyak subjek dalam satu frame dengan kondisi campuran SAFE dan UNSAFE secara simultan.

Pada pengujian postur tubuh, seluruh subjek terdeteksi dalam posisi berdiri dengan durasi sekitar 20 ms. Sebaran koordinat keypoints menunjukkan struktur tubuh yang konsisten, sehingga sistem tetap mampu mempertahankan akurasi deteksi postur meskipun jumlah subjek dalam satu frame meningkat.

Tabel 16. Detail Deteksi Postur Tubuh Pengujian 4

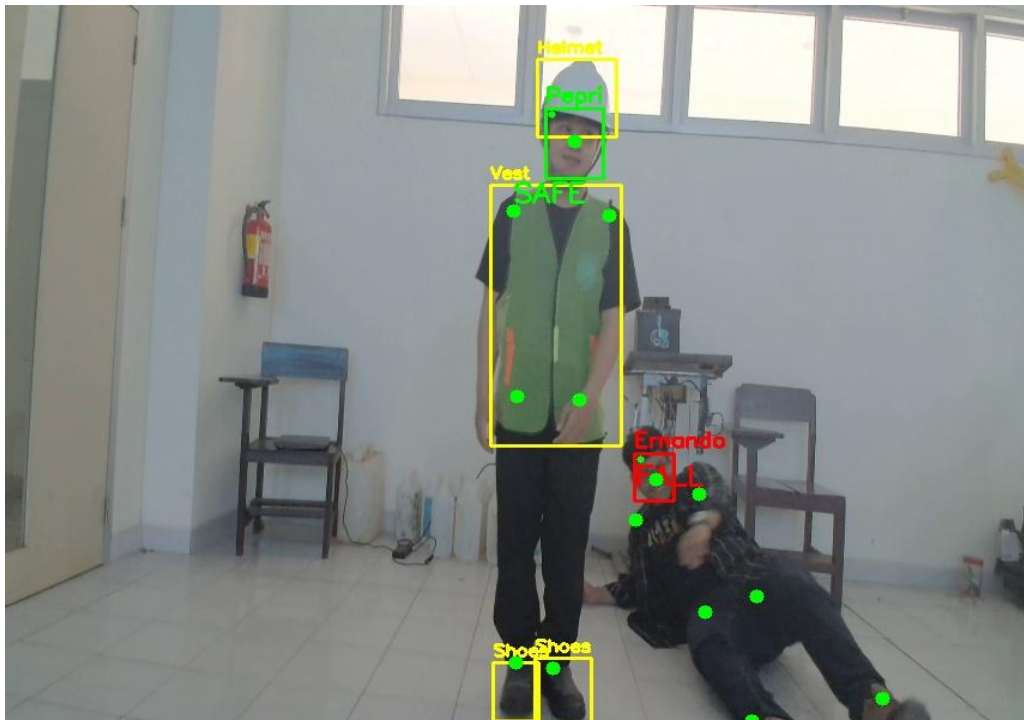
Person		Pepri	Teguh	Amanda	Ernando	Mirna
Pose		Standing	Standing	Standing	Standing	Standing
Result		☒	☒	☒	☒	☒
Duration (ms)		20	20	20	20	20
Keypoints Coordinate X x Y (Pixel)	Nose	56 × 138	181 × 156	271 × 196	608 × 209	487 × 219
	(L) Shoulder	95 × 206	215 × 212	302 × 245	634 × 252	512 × 262
	(R) Shoulder	8 × 201	141 × 204	238 × 242	573 × 252	455 × 260
	(L) Hip	77 × 357	195 × 345	290 × 360	623 × 360	500 × 361
	(R) Hip	20 × 356	145 × 342	248 × 358	583 × 361	463 × 360
	(L) Ankle	64 × 564	188 × 539	285 × 527	611 × 504	487 × 515
	(R) Ankle	28 × 558	150 × 536	245 × 523	586 × 505	464 × 515

Tabel 17. Lanjutan Tabel 16

Person		Gazali	Sri	Thessa	Londongallo	Syamsidar
Pose		Standing	Standing	Standing	Standing	Standing
Results		☒	☒	☒	☒	☒
Duration (ms)		20	20	20	20	20
Keypoints Coordinate X x Y (Pixel)	Nose	355 × 176	765 × 247	673 × 233	416 × 185	544 × 228
	(L) Shoulder	380 × 226	790 × 290	700 × 273	440 × 228	565 × 269
	(R) Shoulder	319 × 226	733 × 292	650 × 272	383 × 224	512 × 267
	(L) Hip	369 × 346	781 × 394	687 × 365	432 × 336	552 × 371
	(R) Hip	328 × 346	744 × 395	655 × 364	393 × 335	517 × 371
	(L) Ankle	358 × 521	767 × 538	680 × 519	420 × 509	540 × 510
	(R) Ankle	324 × 524	748 × 539	667 × 518	394 × 509	524 × 511

Selain itu, peningkatan jumlah individu dalam satu *frame* juga memberikan variasi posisi, jarak, dan ukuran objek yang lebih kompleks, sehingga mempengaruhi hasil deteksi pada beberapa kelas seperti sepatu dan rompi yang memiliki ukuran relatif kecil. Perbedaan jarak terhadap kamera menyebabkan variasi ukuran *bounding box* dan nilai *confidence*, terutama pada objek yang berada lebih jauh atau tertutup sebagian. Kondisi ini menunjukkan bahwa sistem tetap mampu melakukan deteksi secara bersamaan pada banyak subjek, meskipun terdapat variasi kondisi lingkungan dan posisi antar individu

3.4.5 Pengujian 5



Gambar 35. Pengujian dengan Kondisi Berdiri dan Jatuh

Berdasarkan hasil Pengujian 5 (Gambar 35), sistem mampu membedakan dua kondisi postur yaitu berdiri dan jatuh pada dua individu, Pepri dan Ernando. Pepri terdeteksi dalam posisi *standing* dengan *confidence* 92%, sedangkan Ernando terdeteksi dalam posisi *fall* dengan *confidence* yang sama, yaitu 92%, dengan durasi pemrosesan masing-masing 18 ms. Perbedaan ini terlihat jelas dari pola koordinat *keypoints*, di mana posisi tubuh Pepri tersusun vertikal dan proporsional, sementara pada Ernando titik-titik tubuh cenderung menyebar secara horizontal dan berada pada posisi lebih rendah, yang mengindikasikan kondisi jatuh.

Hasil ini menunjukkan bahwa sistem mampu mengenali perbedaan postur secara akurat meskipun dalam satu *frame* terdapat kondisi yang berbeda. Deteksi jatuh pada Ernando juga sesuai dengan kondisi sebenarnya, sehingga menunjukkan bahwa pendekatan *pose estimation* mampu menangkap perubahan posisi tubuh secara jelas. Selain itu, perbedaan distribusi *keypoints* terlihat pada penurunan posisi titik kepala, bahu, dan pinggul pada kondisi jatuh. Pola ini memperlihatkan perubahan orientasi tubuh

dari vertikal menjadi lebih mendatar. Variasi ini menjadi indikator utama dalam membedakan kondisi berdiri dan jatuh pada setiap individu.

Tabel 18. Detail Deteksi Pengujian 5

Person		Pepri	Ernando
Status		SAFE	FALL
Result		☒	☒
Face	Result	☒	☒
	Conf (%)	79	60
	Dur (ms)	73	76
	BBox W x H (pixel)	49×58	52×63
Helmet	Result	☒	☒
	Conf (%)	83	-
	Dur (ms)	64	64
	BBox W x H (pixel)	74×56	-
Vest	Result	☒	☒
	Conf (%)	90	-
	Dur (ms)	64	64
	BBox W x H (pixel)	134×184	-
Shoe (L)	Result	☒	☒
	Conf (%)	36	-
	Dur (ms)	64	64
	BBox W x H (pixel)	61×53	-
Shoe (R)	Result	☒	☒
	Conf (%)	65	-
	Dur (ms)	64	64
	BBox W x H (pixel)	57×41	-

Perbedaan orientasi tersebut juga terlihat dari jarak antar *keypoints* yang menjadi lebih rapat pada sumbu vertikal dan lebih melebar pada sumbu horizontal saat kondisi jatuh. Posisi titik kaki yang tidak lagi berada tepat di bawah pinggul menunjukkan perubahan keseimbangan tubuh yang signifikan. Pada kondisi berdiri, hubungan antar titik tubuh tetap terjaga dengan pola yang lebih simetris dan stabil. Variasi ini memperlihatkan bagaimana perubahan posisi tubuh dapat terdeteksi melalui pola spasial

keypoints secara menyeluruh. Kondisi ini menunjukkan bahwa analisis *keypoints* mampu merepresentasikan perubahan postur tanpa bergantung pada faktor eksternal seperti lingkungan atau objek di sekitar.

Tabel 19. Detail Deteksi Posture Tubuh Pengujian 5

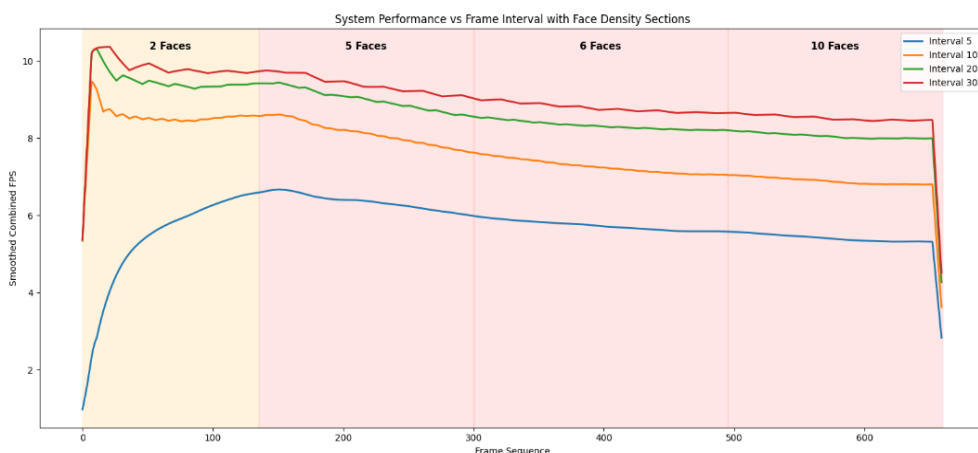
Person		Pepri	Ernando
Pose		Standing	Fall
Result		☒	☒
Duration (ms)		18	18
Confidence (%)		92	92
Keypoints Coordinate X x Y (Pixel)	Nose	475 × 122	543 × 422
	(L) Shoulder	505 × 188	580 × 436
	(R) Shoulder	424 × 183	527 × 458
	(L) Hip	479 × 352	628 × 526
	(R) Hip	428 × 349	585 × 540
	(L) Ankle	457 × 590	732 × 617
	(R) Ankle	427 × 586	623 × 638

3.5 Analisis Data Hasil Pengujian

3.5.1 Analisa Performa

a. Performa Beban *Face Recognition*

Hasil pengujian menunjukkan bahwa interval pemrosesan wajah berpengaruh signifikan terhadap performa sistem yang diukur dalam FPS. Pada grafik, empat konfigurasi interval (5, 10, 20, dan 30 *frame*) diuji pada skenario dengan jumlah wajah bervariasi, yaitu 2, 5, 6, dan 10 wajah per *frame*.



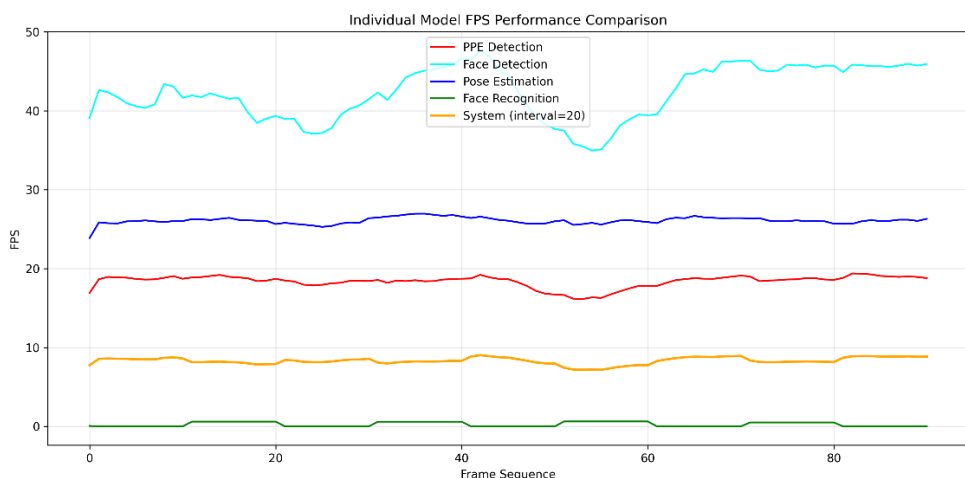
Gambar 36. Grafik Performa FPS di Berbagai Interval *Frame Face Recognition*

Hasil pengujian menunjukkan bahwa interval pemrosesan wajah berpengaruh signifikan terhadap performa sistem yang diukur dalam FPS. Pada grafik, empat konfigurasi interval (5, 10, 20, dan 30 *frame*) diuji pada skenario dengan jumlah wajah bervariasi, yaitu 2, 5, 6, dan 10 wajah per *frame*. Semakin besar interval yang digunakan, semakin tinggi FPS yang dicapai, karena frekuensi ekstraksi *embedding* dan pencocokan dengan *FaceNet* berkurang, sehingga membebaskan sumber daya komputasi untuk tugas deteksi PPE dan estimasi pose.

Sebaliknya, peningkatan jumlah wajah menyebabkan penurunan FPS pada semua interval akibat bertambahnya operasi komputasi, namun penurunan tersebut relatif landai, misalnya pada interval 30 FPS turun dari sekitar 10 menjadi 8,5, sedangkan pada interval 5 turun dari 6 menjadi 5,3. Secara umum, sistem menunjukkan skalabilitas yang baik karena tidak mengalami penurunan performa drastis meskipun jumlah wajah meningkat hingga 10 orang. Interval 20 hingga 30 *frame* terbukti paling optimal karena mampu mempertahankan FPS di atas 8 *frame* per detik pada kondisi padat wajah, sekaligus tetap memberikan frekuensi pengenalan yang memadai untuk menjaga konsistensi identitas pekerja.

b. Performa Sistem Pada Laptop Utama Vs Laptop Pemandang

Berdasarkan pengujian performa sistem pada laptop utama yang ditampilkan pada Gambar 37, terlihat bahwa sistem mampu mencapai performa yang lebih tinggi pada seluruh model. Model *Face Detection* mencatatkan performa tertinggi dengan nilai FPS yang relatif stabil pada kisaran 38–46 FPS. Sementara itu, model *Pose Estimation* berada pada kisaran 25–27 FPS, dan *PPE Detection* pada kisaran 17–19 FPS. Ketiga model ini menunjukkan stabilitas yang baik dengan variasi FPS yang relatif sempit, sehingga proses inferensi dapat berjalan secara konsisten.

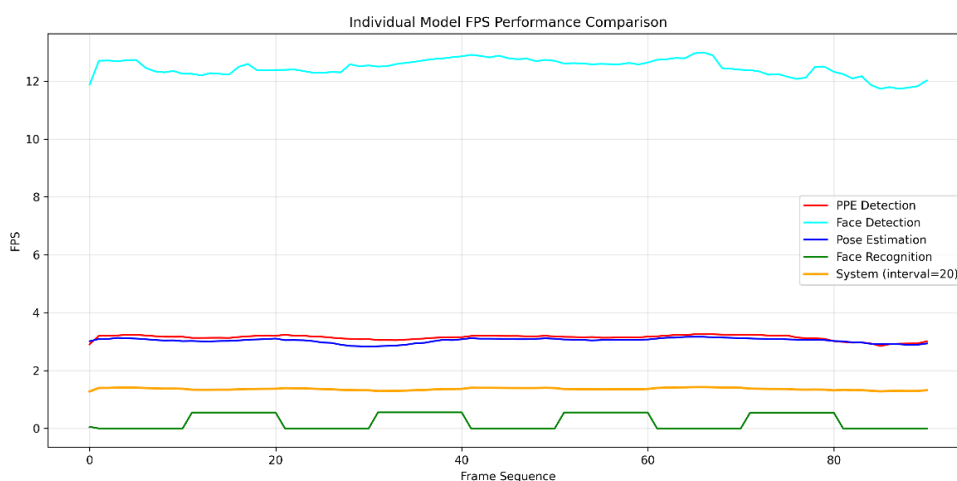


Gambar 37. Grafik Sistem *Running* pada Laptop Utama

Model *Face Recognition* memiliki performa yang lebih rendah dan bersifat fluktuatif karena hanya dijalankan pada interval tertentu, yaitu setiap 20 *frame*. Hal ini menyebabkan proses *face recognition* tidak aktif pada setiap *frame*, sehingga nilai FPS terlihat kecil dan hanya muncul pada waktu tertentu. Pada pengujian sistem terintegrasi yang menggabungkan seluruh model, rata-rata FPS sistem berada pada kisaran 7–9 FPS, dengan fluktuasi yang dipengaruhi oleh jumlah wajah yang terdeteksi. Dalam kondisi tertentu, FPS dapat meningkat hingga sekitar 20 FPS ketika tidak ada wajah terdeteksi, dan menurun hingga 3–4 FPS saat proses deteksi dan pengenalan wajah berjalan secara bersamaan.

Sebaliknya, pada laptop pembanding yang ditampilkan pada Gambar 38, performa sistem mengalami penurunan yang cukup signifikan. Model *Face Detection* hanya mencapai kisaran 12–13 FPS, sedangkan model *PPE Detection* dan *Pose Estimation* berada pada kisaran 2,9–3,2 FPS. Ketiga model masih menunjukkan kestabilan dalam proses inferensi, namun dengan performa yang jauh lebih rendah dibandingkan laptop utama. Pada sistem terintegrasi, rata-rata FPS hanya mencapai sekitar 1,3–1,4 FPS, yang menunjukkan keterbatasan sumber daya komputasi pada perangkat tersebut.

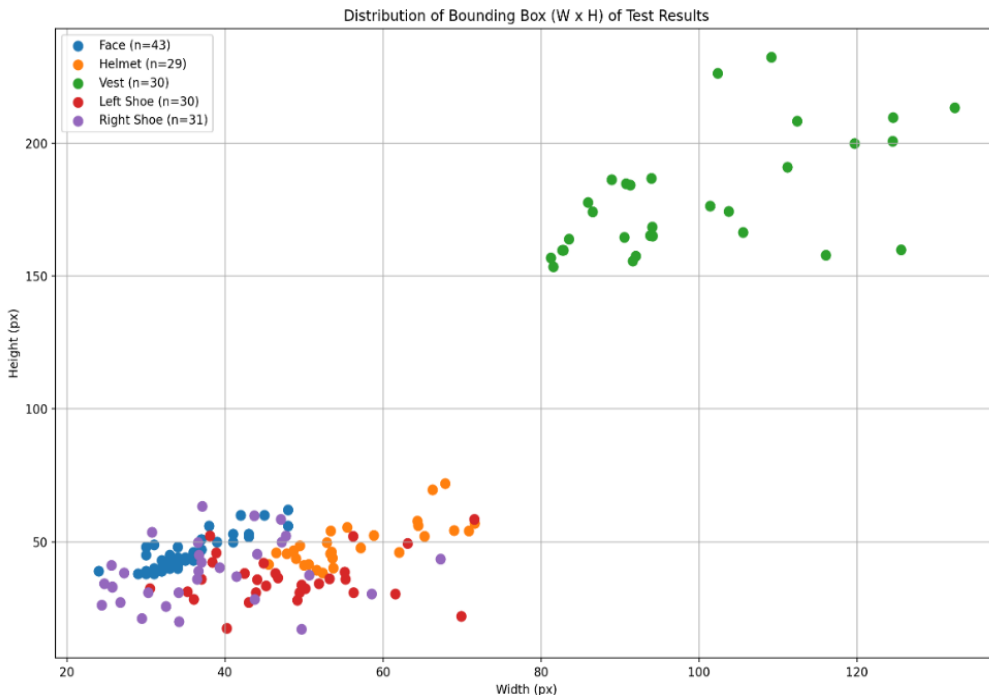
Perbandingan antara kedua perangkat menunjukkan bahwa spesifikasi perangkat keras memiliki pengaruh yang sangat signifikan terhadap performa sistem. Laptop dengan spesifikasi lebih tinggi mampu menjalankan seluruh model dengan lebih optimal dan mendekati kondisi *real-time*, sedangkan perangkat dengan spesifikasi lebih rendah mengalami *bottleneck* pada proses komputasi, terutama ketika seluruh model dijalankan secara bersamaan dalam satu *pipeline* dan sistem terintegrasi mengindikasikan bahwa beban komputasi sangat bergantung pada jumlah objek yang terdeteksi, sehingga diperlukan optimasi seperti pengaturan interval pemrosesan atau pengurangan resolusi untuk menjaga stabilitas FPS.



Gambar 38. Grafik Sistem *Running* pada Laptop Pembanding

3.5.2 Analisis *Bounding Box*

Berdasarkan grafik distribusi *bounding box* dari 10 pengujian (Gambar 39), pola ukuran sebaran tiap kelas yang konsisten dengan jenis objek yang terdeteksi.



Gambar 39. Distribusi *Bounding Box* Hasil Pengujian

Representasi W dan H digunakan untuk melihat ukuran objek hasil deteksi sehingga dapat dianalisis apakah objek berada pada skala kecil, sedang, atau besar, serta berkaitan langsung dengan jarak objek terhadap kamera dan kinerja model deteksi. Kelas *face* yang berjumlah 43 sampel tersebar pada rentang lebar sekitar 24 sampai 48 pixel dan tinggi 38 sampai 62 pixel dengan konsentrasi utama pada lebar 30 sampai 40 pixel dan tinggi 40 sampai 50 pixel. Kelas *helmet* dengan 29 sampel memiliki rentang lebar sekitar 46 sampai 71 pixel dan tinggi 38 sampai 72 pixel, serta masih dijumpai beberapa data bernilai nol yang menunjukkan kondisi tidak terdeteksinya objek pada *frame* tertentu. Kelas *vest* dengan 30 sampel berada pada rentang ukuran paling besar dengan lebar sekitar 81 sampai 132 pixel dan tinggi 153 sampai 232 pixel sehingga titik-titiknya tampak terpisah jelas dari kelas lain pada grafik. Kelas *left shoe* dan *right shoe* masing-masing terdiri dari sekitar 30 sampel dengan rentang lebar 25 sampai 71 pixel dan tinggi 17 sampai 63 pixel serta cukup banyak variasi ukuran akibat perubahan posisi kaki dan jarak subjek terhadap kamera. Sebaran yang lebih tidak beraturan pada kelas sepatu memperlihatkan karakter objek kecil yang lebih sensitif terhadap perubahan sudut pandang dan kondisi lingkungan. Keberadaan sejumlah data dengan ukuran nol pada beberapa kelas juga menggambarkan adanya kasus ketika sistem tidak menghasilkan *bounding box* yang valid pada *frame* tertentu. Seluruh pola ini menggambarkan bagaimana distribusi ukuran objek hasil deteksi muncul pada pengujian

3.5.3 Analisis Kebenaran

Tabel 20. Analisis Keberhasilan Deteksi Sistem 10 Pengujian

Detections	Status	Face	Helmet	Vest	Left Shoes	Right Shoes	Posture	Total
Success	46	51	55	55	51	50	52	308
Failure	9	4	0	0	4	5	3	22
Success Rate (%)	84%	93%	100%	100%	93%	91%	95%	93%

Berdasarkan Tabel 20 hasil analisis keberhasilan deteksi pada 10 pengujian, total deteksi yang tercatat untuk seluruh kelas adalah 330 dengan jumlah keberhasilan sebesar 308 dan kegagalan sebanyak 22, sehingga rata-rata keberhasilan keseluruhan berada pada angka 93%. Pada kelas *Face* tercatat 46 deteksi berhasil dan 9 gagal dengan rata-rata 84%. Kelas *Helmet* mencatat 51 deteksi berhasil dan 4 gagal dengan rata-rata 93%. Kelas *Vest* menunjukkan 55 deteksi berhasil tanpa kegagalan dengan rata-rata 100%. Kelas *Left Shoes* mencatat 55 deteksi berhasil dan 4 gagal dengan rata-rata 93%. Kelas *Right Shoes* mencatat 51 deteksi berhasil dan 5 gagal dengan rata-rata 91%. Kelas *Posture* mencatat 50 deteksi berhasil dan 3 gagal dengan rata-rata 95%. Tabel tersebut menunjukkan bahwa setiap kelas memiliki jumlah keberhasilan dan kegagalan yang berbeda pada 10 kali pengujian. Kelas dengan jumlah keberhasilan tertinggi tanpa kegagalan adalah *Vest* yang memiliki 55 keberhasilan. Kelas yang memiliki jumlah kegagalan tertinggi adalah *Face* dengan 9 kegagalan.