

## SISTEM DETEKSI OTOMATIS HELM UNTUK PROYEK KESELAMATAN KERJA BERBASIS YOLOv8

Muhammad Azizul Mustofa<sup>1\*</sup>, Bagus Satrio Waluyo Poetro

Universitas Islam Sultan Agung

[hmmdazizul@std.unissula.ac.id](mailto:hmmdazizul@std.unissula.ac.id)<sup>1\*</sup> [bagusswp@unissula.ac.id](mailto:bagusswp@unissula.ac.id)<sup>2</sup>

Received: 28-07- 2025

Revised: 20-08-2025

Approved: 27-08-2025

### ABSTRAK

Keselamatan kerja di industri konstruksi sangat ditentukan oleh kepatuhan pekerja dalam penggunaan Alat Pelindung Diri (APD), khususnya helm keselamatan. Rendahnya tingkat kedisiplinan pekerja serta keterbatasan pengawasan manual menjadi tantangan serius di lapangan. Penelitian ini bertujuan untuk mengembangkan sistem deteksi otomatis penggunaan helm berbasis algoritma YOLOv8n yang mampu bekerja secara real-time melalui kamera CCTV dengan efisiensi tinggi pada perangkat komputasi terbatas. Metode penelitian meliputi pengumpulan 1.000 gambar teranotasi yang terdiri dari pekerja dengan dan tanpa helm, anotasi data menggunakan Roboflow, pembagian dataset (80% training, 10% validation, 10% testing), serta pelatihan model di Google Colab menggunakan parameter batch size 16, image size 640, epoch 100, dan learning rate 0,0001. Hasil pengujian menunjukkan tingkat akurasi sebesar 74% pada skenario pekerja berhelm dan 81% pada pekerja tanpa helm. Evaluasi melalui confusion matrix juga mengindikasikan kemampuan model dalam mendeteksi kategori helm dengan tingkat presisi 85%, sementara pada kategori non-helmet model mencapai 100% akurasi. Temuan ini membuktikan bahwa YOLOv8n memiliki potensi besar dalam mendukung sistem pengawasan Keselamatan dan Kesehatan Kerja (K3) secara otomatis, sehingga dapat membantu mengurangi angka kecelakaan kerja. Penelitian ini membuka peluang pengembangan lebih lanjut, seperti deteksi APD lain serta integrasi dengan fitur notifikasi otomatis berbasis IoT.

Kata kunci: YOLOv8, deteksi helm, K3, CCTV, deep learning, APD, real-time.

### PENDAHULUAN

Pada proyek pembangunan Mall Pelayanan Publik (MPP) di Kabupaten Indramayu, ditemukan pelanggaran serius terhadap standar keselamatan kerja, terutama dalam hal pemakaian Alat Pelindung Diri (APD) seperti helm keselamatan. Laporan media lokal menyebutkan bahwa banyak pekerja yang tidak mematuhi aturan penggunaan helm, bahkan setelah terjadi kecelakaan kerja yang menelan korban jiwa. Kondisi ini menunjukkan bahwa pengawasan manual di lapangan kurang efektif dalam menjamin kepatuhan terhadap prosedur keselamatan. Sebagai solusi, teknologi berbasis algoritma YOLOv8 dapat dimanfaatkan untuk mendeteksi secara otomatis pekerja yang tidak mengenakan helm melalui kamera CCTV yang terpasang di area proyek. Sistem ini mampu memberikan peringatan langsung kepada petugas K3 dan mencatat pelanggaran secara digital. Dengan penerapan sistem seperti ini, perusahaan dapat meningkatkan pengawasan, menurunkan risiko kecelakaan, serta membangun budaya kerja yang lebih disiplin dan aman, terutama dalam proyek besar seperti MPP yang melibatkan banyak pekerja dan area kerja luas (Alan, 2025).

Dalam dunia konstruksi yang terus berkembang pesat, keselamatan kerja tetap menjadi tantangan utama. Menurut data Badan Pusat Statistik, sektor ini mencatatkan angka kecelakaan yang mengkhawatirkan, dengan ribuan pekerja mengalami cedera setiap tahunnya. Setiap kecelakaan bukan hanya merugikan individu, tetapi juga berpotensi menimbulkan dampak ekonomi yang besar bagi perusahaan dan masyarakat. Oleh karena itu, penerapan alat pelindung diri (APD) seperti helm bukan sekadar kewajiban, melainkan suatu keharusan yang tak bisa diabaikan (Hartono dkk., 2023). Namun, kendala nyata muncul saat penerapan aturan penggunaan helm tidak diikuti oleh kesadaran dan disiplin pekerja. Meskipun peraturan telah ditetapkan, dalam

praktiknya, banyak pekerja yang mengabaikan kewajiban ini, terutama di lokasi proyek yang ramai dan dinamis. Dalam konteks ini, inovasi teknologi menjadi solusi yang sangat dibutuhkan untuk meningkatkan kepatuhan dan menciptakan lingkungan kerja yang lebih aman (Susanti, 2023a).

Di sinilah sistem cerdas berbasis YOLO (*You Only Look Once*) hadir sebagai terobosan revolusioner. Dengan kemampuannya untuk mendeteksi objek secara real-time melalui algoritma pembelajaran mendalam, YOLO menawarkan pendekatan baru dalam memantau kepatuhan penggunaan helm. Teknologi ini tidak hanya cepat dan akurat, tetapi juga mampu bekerja di berbagai kondisi pencahayaan dan latar belakang yang kompleks, menjadikannya ideal untuk diterapkan di lokasi proyek yang beragam. Dengan menerapkan sistem deteksi otomatis helm berbasis YOLO, setiap pekerja yang memasuki area proyek akan secara instan dipantau. Ketika helm tidak terdeteksi, sistem dapat mengeluarkan peringatan yang segera diinformasikan kepada pengawas atau manajer proyek. Hal ini tidak hanya mendorong disiplin, tetapi juga menciptakan budaya keselamatan yang lebih kuat di antara pekerja (Direja dkk., 2024).

Melalui penelitian ini, kami bertujuan untuk merancang dan mengembangkan sistem cerdas berbasis YOLO untuk deteksi otomatis helm proyek yang efektif dan efisien. Diharapkan sistem ini tidak hanya meningkatkan kepatuhan penggunaan helm di lapangan, tetapi juga berkontribusi pada pengurangan angka kecelakaan kerja, menciptakan lingkungan yang lebih aman, dan memberikan nilai tambah bagi seluruh pemangku kepentingan dalam industri konstruksi. Dengan menggabungkan teknologi dan komitmen terhadap keselamatan, kita dapat bersama-sama membangun masa depan yang lebih aman bagi para pekerja di lapangan

## KAJIAN LITERATUR

Keselamatan dan Kesehatan Kerja (K3) adalah aspek yang paling penting bagi karyawan yang bekerja di sektor konstruksi. Cedera kepala merupakan salah satu jenis cedera yang serius dan sering kali berakibat fatal. Oleh karena itu, helm sangat penting bagi pekerja sebagai alat pelindung utama di lingkungan konstruksi. Metode yang digunakan untuk deteksi ini adalah YOLO (*You Only Look Once*), yang terdiri dari tiga tahap: pra-pemrosesan, pelatihan, dan deteksi. Pada tahap deteksi, citra yang telah diproses akan dikenali, dan implementasi ini dilakukan menggunakan bahasa pemrograman Python dengan platform Google Colab. Tujuan dari implementasi ini adalah untuk mendeteksi penggunaan helm dengan metode *TinyYOLOv8*. Dari hasil pengujian terhadap 50 dataset, akurasi tertinggi yang dicapai adalah 93% pada skenario ketiga, dengan waktu pemrosesan selama 9,44 detik (Susanti, 2023).

Pada penelitian selanjutnya yaitu penerapan Keselamatan dan Kesehatan Kerja (K3) di proyek konstruksi bertujuan menciptakan lingkungan kerja yang aman dan bebas dari risiko kecelakaan. Menurut ILO (2019), lebih dari 395 juta pekerja secara global mengalami cedera *non-fatal*. Di Indonesia, BPJS Ketenagakerjaan mencatat 370.747 kasus kecelakaan kerja pada 2023, meningkat 19,7% dari tahun sebelumnya, dengan 32,12% disebabkan oleh kelalaian penggunaan APD seperti helm. Untuk mengatasi hal ini, teknologi berbasis *YOLOv8* digunakan sebagai solusi deteksi otomatis penggunaan helm, dengan hasil evaluasi model menunjukkan presisi 77,7%, recall 60,7%, dan mAP sebesar 68,9%, membuktikan efektivitasnya dalam mendukung pengawasan K3 secara real-time (Fatkhin & Fadjeri, 2024).

Penelitian selanjutnya kemajuan dalam deteksi kepala telah mengalami peningkatan signifikan berkat perkembangan algoritma kecerdasan buatan. Penelitian

ini bertujuan untuk mengevaluasi performa model sistem penghitung jumlah kepala menggunakan algoritma YOLOv8, dengan fokus khusus pada pembuatan model deteksi jumlah orang. Dataset yang digunakan terdiri dari 2.390 gambar yang diperoleh dari Roboflow, dengan pembagian data sebesar 70:20:10 untuk data latih, data uji, dan data validasi. Model dilatih selama 50 epoch. Kinerja model diukur melalui data pelatihan, confusion matrix, dan metrik evaluasi yang meliputi presisi, akurasi, *recall*, dan *F1-score*. Hasil pengujian menunjukkan akurasi mencapai 87,56%, presisi 83,74%, *recall* 100%, dan *F1-score* 91,15%. Selain itu, kurva presisi menunjukkan nilai tertinggi 1 pada tingkat kepercayaan 0,857, sementara *recall* mencatat 0,8 pada tingkat kepercayaan 0. *F1-score* sebesar 0,716 dicapai pada kepercayaan 0,36, dan presisi-*recall* mencapai 0,771 pada 0,5 mAP (Drantantiyas et al., 2023).

Sementara itu pada penelitian nomor polisi kendaraan bermotor berfungsi sebagai identifikasi unik yang diberikan oleh kepolisian dan biasanya tertera pada plat kendaraan. Pengenalan otomatis plat nomor memainkan peran krusial dalam pengaturan lalu lintas dan peningkatan keamanan. Penelitian ini berfokus pada penerapan arsitektur YOLOv8 untuk deteksi plat nomor secara otomatis, diintegrasikan dengan metode PaddleOCR untuk meningkatkan akurasi pengenalan karakter. Model yang dikembangkan menunjukkan performa yang mengesankan, dengan skor presisi mencapai 0,871 dan *recall* sebesar 0,85. Model ini mampu mengidentifikasi plat nomor secara efektif, bahkan pada gambar yang berkualitas rendah atau mengalami distorsi (Cahyanti dkk., 2025).

Selanjutnya pada penelitian ini bertujuan untuk mengembangkan sistem deteksi mata uang Rupiah emisi tahun 2022 dengan memanfaatkan model YOLO V8, yang dikenal karena kemampuannya dalam deteksi objek yang cepat dan akurat. Sistem ini dilengkapi dengan output audio untuk memudahkan penyandang tunanetra dalam mengenali mata uang. Dirancang khusus untuk mendeteksi uang kertas, model ini mampu memberikan label dalam bahasa Inggris dan Indonesia, serta menunjukkan tingkat keberhasilan yang baik dalam membedakan denominasi mata uang. Meskipun fokus utamanya adalah pada mata uang Indonesia, model ini memiliki potensi untuk diperluas guna mendeteksi mata uang dari negara lain. Penelitian ini menjawab tantangan aksesibilitas bagi individu tunanetra, sekaligus meningkatkan keberlanjutan teknologi untuk mendukung inklusivitas dalam sistem keuangan (Steven Immanuel Sihombing dkk., 2024).

## METODE PENELITIAN

### *Convolutional Neural Network (CNN)*

*Convolutional Neural Networks (CNN)* merupakan arsitektur jaringan saraf yang lebih cocok untuk klasifikasi gambar. Tujuan utama CNN adalah kompleksitasnya, dimana setiap fitur diekstraksi dari gambar untuk membuat beberapa pola yang lebih mudah untuk diklasifikasikan. Metode ini membuat pembelajaran gambar lebih efisien untuk diterapkan (Fasounaki et al., 2021).

Di bidang *deep learning*, *Convolutional Neural Network (CNN)* menjadi sangat populer dan metode ini tidak mengubah kualitas gambar tetapi mengubah dimensinya. Klasifikasi tambahan menggunakan CNN dengan metode pembelajaran *transfer*. Keunggulan metode ini adalah kemampuan untuk melakukan pra-pelatihan menggunakan informasi yang telah dipelajari dari model CNN yang telah dilatih sebelumnya. Informasi ini kemudian dapat disesuaikan dan diterapkan pada tugas klasifikasi yang baru. Teknik ini tidak hanya dapat meningkatkan kinerja model secara

keseluruhan, tetapi juga dapat mengurangi waktu yang diperlukan untuk pelatihan model dan menghasilkan kinerja yang lebih baik (Satrio et al., 2024.).



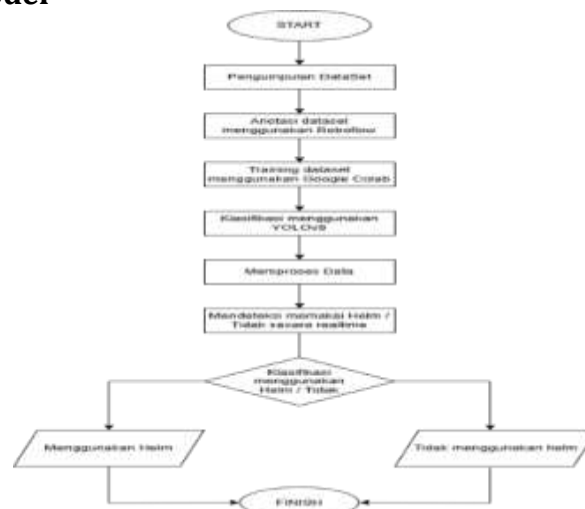
Gambar Arsitektur CNN

Dalam implementasinya, metode *Convolutional Neural Network* (CNN) dikembangkan melalui beberapa tahapan yaitu pembuatan model, pelatihan model, evaluasi model dan prediksi model akhir. Pembangunan CNN terdiri dari dua proses yaitu ekstraksi dan klasifikasi. Proses ekstraksi citra pada CNN terdiri dari beberapa lapisan tersembunyi yaitu lapisan konvolusi, fungsi aktivasi, dan lapisan agregasi. CNN bekerja secara berurutan, sehingga keluaran lapisan konvolusional pertama digunakan sebagai masukan pada lapisan konvolusional berikutnya. Proses klasifikasi menggunakan seluruh lapisan koneksi. Kedua lapisan ini dapat digabungkan dengan fungsi aktivasi (*softmax*) untuk mengoptimalkan proses CNN (Nugroho & Puspaningrum, 2021).

## YOLOv8

Sistem yang dikembangkan dalam penelitian ini bertujuan untuk meningkatkan keselamatan kerja di lingkungan proyek konstruksi dengan memanfaatkan teknologi deteksi objek berbasis deep learning. Sistem ini menggunakan algoritma YOLOv8 (*You Only Look Once* versi 8) untuk secara otomatis mendeteksi keberadaan helm pada kepala para pekerja. Data berupa gambar pekerja yang mengenakan dan tidak mengenakan helm dikumpulkan lalu diberi label melalui platform seperti Roboflow. Dataset ini dibagi menjadi data pelatihan, validasi, dan pengujian. Selanjutnya, model YOLOv8 dilatih menggunakan Google Colab untuk mengenali objek helm dengan parameter yang telah disesuaikan. Setelah proses pelatihan selesai, model digunakan untuk mendeteksi penggunaan helm secara *real-time* dari gambar atau video yang diambil oleh kamera pengawas (*CCTV*) di lokasi proyek. Jika sistem mendeteksi seorang pekerja tidak mengenakan helm, maka secara otomatis akan mengirimkan peringatan kepada petugas K3. Selain itu, pelanggaran tersebut akan dicatat untuk keperluan evaluasi dan pelaporan. Dengan adanya sistem ini, proses pengawasan menjadi lebih efisien, dan budaya keselamatan kerja dapat dibangun dengan lebih kuat melalui pemanfaatan teknologi yang cerdas dan responsif.

## Rancangan Alur Model



Gambar *Flowchart* Alur Sistem

Diagram alur (*flowchart*) tersebut secara keseluruhan memvisualisasikan proses kerja dari sistem pendeteksi helm secara real-time, mulai dari tahap awal hingga menghasilkan output klasifikasi. Prosedur dimulai dengan tahap inisiasi sistem, ditandai dengan simbol “Start”, yang mengindikasikan dimulainya seluruh proses kerja. Langkah pertama yang dilakukan adalah pengumpulan data, di mana sistem menghimpun berbagai gambar atau cuplikan video yang memuat individu yang mengenakan dan tidak mengenakan helm. Proses ini merupakan tahapan dasar yang sangat penting karena kualitas dan variasi data yang dikumpulkan akan sangat mempengaruhi kinerja model ke depannya.

Setelah dataset dikumpulkan, data tersebut harus diberi anotasi. Anotasi merupakan proses pelabelan terhadap objek yang terdapat dalam gambar, dan dalam hal ini dilakukan menggunakan platform Roboflow. Melalui Roboflow, setiap objek dalam gambar diberi bounding box serta label kategori seperti “Helmet” atau “No Helmet”, sehingga nantinya model dapat mempelajari ciri visual dari masing-masing kategori secara akurat. Selanjutnya, dataset yang sudah dianotasi dimasukkan ke dalam proses pelatihan model. Pelatihan ini dilakukan di lingkungan Google Colab, yang menyediakan akses terhadap GPU untuk mempercepat proses komputasi, serta sangat fleksibel bagi pengguna dalam menjalankan kode Python.

Tahapan pelatihan menggunakan model deteksi objek YOLOv8, yang merupakan versi terbaru dari arsitektur YOLO yang dikenal memiliki performa unggul dalam deteksi objek secara cepat dan presisi tinggi. Setelah proses *training* selesai, model yang dihasilkan siap digunakan untuk proses klasifikasi. Input berupa gambar atau video kemudian dianalisis oleh model untuk mendeteksi keberadaan objek manusia dan menilai apakah objek tersebut mengenakan helm atau tidak. Proses ini bekerja secara otomatis dan berlangsung secara *real-time*, artinya deteksi dilakukan seketika pada saat input masuk, tanpa perlu jeda waktu yang lama.

Tahap berikutnya adalah pemrosesan data input. Model akan menganalisis gambar secara menyeluruh untuk mendeteksi objek target. Setelah objek dikenali, sistem kemudian menjalankan proses klasifikasi berdasarkan hasil analisis model. Jika sistem menemukan bahwa seseorang dalam gambar memakai helm, maka hasil klasifikasinya adalah “Menggunakan Helm”. Sebaliknya, jika tidak terdeteksi adanya helm, maka individu tersebut akan diklasifikasikan sebagai “Tidak Menggunakan Helm”. Proses

klasifikasi ini bersifat otomatis dan hasilnya dapat divisualisasikan dalam bentuk output teks, warna bounding box berbeda, atau notifikasi lainnya.

Setelah hasil klasifikasi ditentukan, sistem memasuki tahap akhir yang ditandai dengan simbol "Finish". Namun, karena sistem dirancang untuk berjalan secara real-time, proses ini bersifat siklikal atau berulang. Sistem akan terus menerima input baru, memprosesnya, dan menghasilkan klasifikasi secara terus menerus selama sistem dijalankan. Dengan pendekatan alur kerja seperti ini, sistem mampu memberikan deteksi helm secara cepat, akurat, dan konsisten. Hal ini sangat mendukung penerapan teknologi ini di berbagai sektor yang memerlukan kepatuhan terhadap penggunaan alat pelindung diri, seperti area konstruksi, pabrik, dan sistem pemantauan lalu lintas.

### Pengumpulan Data

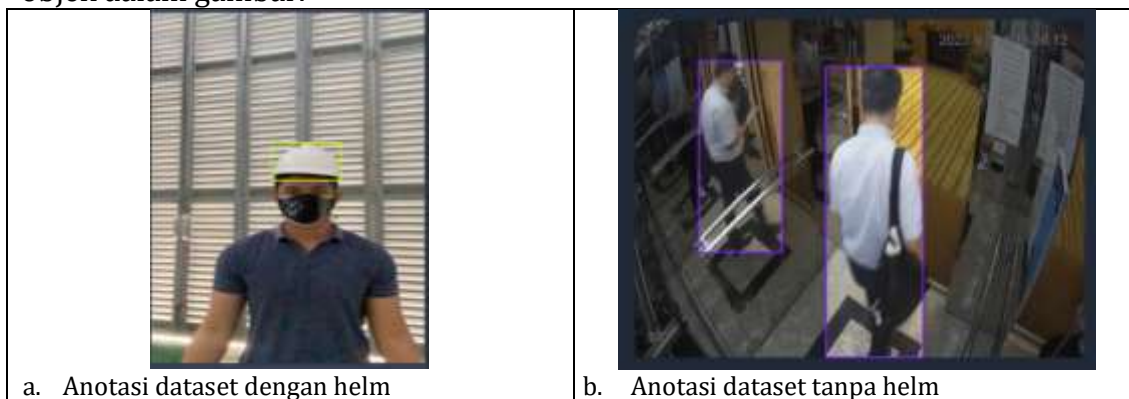
Pada penelitian ini, proses pengumpulan data yang berupa gambar diperoleh dari Roboflow (<https://drive.google.com/drive/folders/1-h0rrtyvJkMstPkQOV7u-06ElvvBTFiVI?usp=sharing>). Sehingga jumlah gambar yang akan digunakan pada penelitian ini berjumlah 1.000 gambar.

Tabel Pembagian Dataset



### Anotasi Data

Anotasi adalah pemberian tanda dengan menentukan *bounding box* yang berisi nama kelas sesuai dengan masing-masing objek, *Bounding box* ini berisi label atau nama kelas yang sesuai dengan objek tersebut. Proses ini penting dalam pelatihan model deteksi objek, karena memberikan informasi kepada model tentang lokasi dan jenis objek dalam gambar.



### Pre-processing Data

Tahap awal dalam proses preprocessing gambar adalah melakukan penyesuaian ukuran gambar (*resize*). Langkah ini bertujuan untuk menyamakan resolusi seluruh gambar dalam dataset yang sebelumnya memiliki dimensi berbeda-beda. Dengan menyatukan resolusi, sistem dapat memproses data secara konsisten dan lebih efisien.

Selain itu, ukuran gambar yang lebih kecil dapat membantu mengurangi beban komputasi, sehingga mempercepat waktu pelatihan model secara keseluruhan. Umumnya, gambar akan diubah ke resolusi tertentu, misalnya 640x640 piksel, agar sesuai dengan kebutuhan arsitektur model deteksi objek seperti YOLOv8.

### Augmentasi Data

Dalam penelitian ini, dilakukan beberapa augmentasi data pada dataset, seperti rotasi gambar, penyesuaian kecerahan, pengaturan *exposure*, *mosaic* gambar, dan penambahan efek *noise*. Rotasi gambar bertujuan membantu model mendeteksi objek saat orientasi objek berubah. Penyesuaian kecerahan dan *exposure* dilakukan agar model dapat mendeteksi objek dengan baik meskipun ada perubahan pencahayaan.

### Training Data

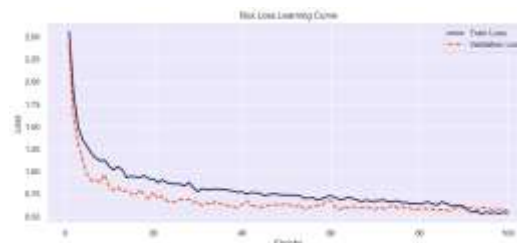
Proses training pada penelitian ini dilakukan menggunakan total 1.000 citra yang diperoleh dari platform Roboflow. Seluruh data tersebut telah melalui proses anotasi secara manual di Roboflow untuk memberikan label posisi dan kelas objek yang akan dikenali. Data kemudian dibagi menjadi tiga subset dengan proporsi 80% untuk training set, 10% untuk validation set, dan 10% untuk testing set. Pembagian ini bertujuan untuk memastikan model dapat belajar secara optimal sekaligus menguji kemampuannya pada data yang tidak pernah digunakan selama proses pelatihan. Parameter pelatihan yang digunakan pada model antara lain:

Tabel 1. Konfigurasi Training

Nama Paramater	Keterangan
Epoch	100
Image Size	640 piksel
Patience	50
Batch Size	16
Optimizer	auto
Learning Rate	0,0001
Drop Out	0,1
Seed	0

## HASIL DAN PEMBAHASAN

### Hasil evaluasi Model



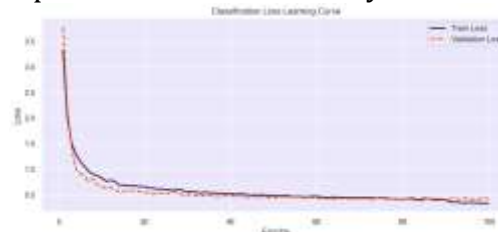
Gambar Box Loss Learning Curve

Grafik pertama yang ditampilkan merupakan *Box Loss Learning Curve*, yaitu kurva pembelajaran yang merepresentasikan performa model dalam hal *bounding box regression* selama proses pelatihan (training) dan validasi. Pada grafik ini, sumbu horizontal (x) menunjukkan jumlah *epoch* dari 0 hingga 100, sedangkan sumbu vertikal (y) merepresentasikan nilai *loss*. Garis berwarna biru tua menandakan nilai *box loss* saat pelatihan, sementara garis putus-putus berwarna jingga menunjukkan

nilai *box loss* pada data validasi.

Dari pengamatan terhadap grafik, dapat dilihat bahwa pada awal pelatihan, nilai *box loss* sangat tinggi, berada di kisaran lebih dari 2.5. Hal ini adalah sesuatu yang wajar karena pada awalnya model belum memiliki parameter yang optimal dalam memprediksi posisi dan ukuran kotak deteksi objek. Seiring bertambahnya *epoch*, terjadi penurunan *loss* yang signifikan, menunjukkan bahwa model mulai belajar mengenali struktur dan pola pada data latih dengan lebih baik.

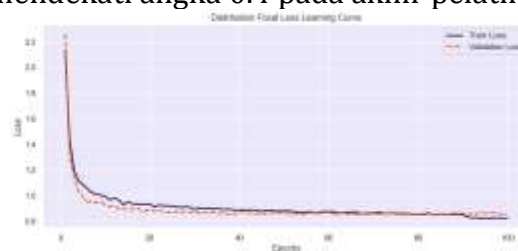
Menjelang pertengahan proses pelatihan (sekitar *epoch* ke-40), penurunan nilai *loss* mulai melambat dan cenderung stabil, yang menandakan bahwa model telah mencapai titik konvergensi, yaitu ketika pembelajaran menjadi lebih matang dan pembaruan bobot (*weight updates*) tidak lagi memberikan peningkatan signifikan terhadap kinerja. Stabilitasnya perbedaan antara kurva pelatihan dan validasi menunjukkan bahwa model tidak mengalami gejala *overfitting* yang berat. Ini merupakan indikator bahwa model mampu melakukan generalisasi yang baik terhadap data yang belum pernah dilihat sebelumnya.



Gambar *Classification Loss Learning Curve*

Grafik kedua menggambarkan *Classification Loss Learning Curve*, yaitu metrik kesalahan (*loss*) dalam proses pengklasifikasian objek yang terdeteksi. Sumbu x tetap menunjukkan *epoch*, sedangkan sumbu y merepresentasikan besarnya *classification loss*. Kurva biru solid adalah nilai *classification loss* pada data latih, sedangkan kurva jingga putus-putus adalah pada data validasi.

Nilai awal *loss* pada grafik ini terpantau sangat tinggi, mendekati 3.8. Hal ini menggambarkan bahwa pada tahap awal, model mengalami kesulitan dalam menentukan label klasifikasi dari objek yang terdeteksi. Seiring pelatihan berlangsung, *classification loss* secara konsisten menurun, dengan penurunan paling tajam terjadi pada 20 *epoch* pertama. Setelah fase ini, kurva mulai melandai dan menunjukkan tren yang cenderung stabil mendekati angka 0.4 pada akhir pelatihan.



Gambar *Distribution Focal Loss learning curve*

Grafik di atas menampilkan *Distribution Focal Loss learning curve* dari sebuah model yang menggunakan fungsi kerugian *Distribution Focal Loss learning curve*. Terdapat dua jenis kurva pada grafik ini, yaitu *Train Loss* yang ditunjukkan dengan garis berwarna biru solid, dan *Validation Loss* yang digambarkan dalam bentuk garis merah putus-putus. Keduanya merepresentasikan perubahan nilai kerugian (*loss*) selama proses pelatihan berlangsung hingga 100 *epoch*. Penurunan yang tajam pada kurva *Train Loss* di awal *epoch* mencerminkan bahwa model mampu belajar dengan

cepat dari data pelatihan. Proses ini terjadi karena parameter-parameter awal dalam model mulai menemukan pola-pola dasar dari data input, sehingga terjadi efisiensi dalam proses minimisasi kerugian.

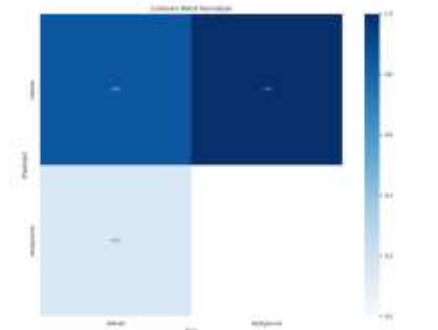
Kurva *Validation Loss* juga mengalami tren penurunan yang serupa, meskipun sedikit lebih lambat jika dibandingkan dengan *Train Loss*. Hal ini menunjukkan bahwa performa model terhadap data validasi, yaitu data yang tidak digunakan secara langsung selama proses pelatihan, juga meningkat. Artinya, model tidak hanya mampu menyesuaikan diri terhadap data pelatihan, namun juga menunjukkan kemampuan generalisasi terhadap data baru yang belum pernah dilihat sebelumnya. Setelah melalui sejumlah *epoch*, baik *Train Loss* maupun *Validation Loss* mulai mengalami pelandaian (*flattening*), khususnya setelah *epoch* ke-80. Fase ini mengindikasikan bahwa model mulai mencapai konvergensi, yakni kondisi di mana proses pembelajaran tidak lagi menghasilkan penurunan kerugian yang signifikan, atau dengan kata lain, model telah berada pada titik stabil secara performa.

Walaupun terdapat sedikit selisih antara *Train Loss* dan *Validation Loss*, perbedaan tersebut masih dalam batas yang wajar. Adanya gap tersebut tidak serta-merta menandakan *overfitting*, namun tetap menjadi aspek penting yang perlu diperhatikan, terutama jika perbedaan itu semakin membesar. *Overfitting* sendiri merupakan kondisi ketika model terlalu menyesuaikan diri terhadap data pelatihan, sehingga kemampuannya dalam menangani data yang belum pernah dilihat menurun. Oleh karena itu, pemantauan terhadap *Validation Loss* sangat penting sebagai indikator untuk menilai seberapa baik model dapat menggeneralisasi.

Penggunaan *Distribution Focal Loss* dalam penelitian ini dimaksudkan untuk menangani masalah ketidakseimbangan kelas (*class imbalance*), yang sering terjadi dalam permasalahan klasifikasi. Fungsi kerugian ini merupakan pengembangan dari *focal loss* konvensional yang bekerja dengan memberikan penalti lebih besar terhadap kesalahan pada kelas minoritas atau kelas yang sulit diprediksi. Dengan pendekatan ini, model diarahkan untuk fokus pada sampel-sampel yang sulit, sehingga menghasilkan representasi pembelajaran yang lebih adil dan seimbang terhadap seluruh distribusi kelas. Hasil yang tergambar pada grafik menunjukkan bahwa strategi ini berhasil diterapkan dengan baik, karena terlihat adanya stabilitas pada *loss function* baik selama pelatihan maupun validasi, serta tidak ditemukan fluktuasi ekstrem yang biasanya mengindikasikan ketidakseimbangan pembelajaran.

Secara keseluruhan, kurva ini mencerminkan bahwa model bekerja secara efisien dan efektif. Ia tidak hanya mampu menurunkan nilai kerugian secara signifikan di awal pelatihan, tetapi juga menjaga stabilitas performa hingga tahap akhir. Hal ini menunjukkan bahwa proses pelatihan berjalan dengan optimal dan penggunaan *Distribution Focal Loss* berhasil membantu model dalam mencapai kemampuan generalisasi yang baik, terutama dalam konteks dataset yang mengandung distribusi kelas yang tidak seimbang. Jika diperlukan perbaikan lanjutan, aspek-aspek seperti penyesuaian *hyperparameter*, penggunaan strategi augmentasi data, atau tuning lebih lanjut terhadap fungsi kerugian dapat dipertimbangkan untuk mengoptimalkan kinerja model ke level yang lebih tinggi lagi.

## Visualisasi Hasil Evaluasi



Gambar matriks evaluasi

Gambar yang ditampilkan merupakan visualisasi dari *confusion matrix* yang telah dinormalisasi. Matriks ini adalah alat penting dalam evaluasi model klasifikasi karena memberikan informasi detail tentang kinerja prediksi model terhadap masing-masing kelas. Dalam konteks ini, model diujikan pada dua kategori utama, yaitu "Helmet" dan "Background". Sumbu mendatar (*horizontal*) menunjukkan label asli dari data (*true labels*), sedangkan sumbu tegak (*vertikal*) merepresentasikan hasil prediksi model. Normalisasi pada matriks ini berarti bahwa angka-angka yang ditampilkan menunjukkan persentase, bukan jumlah absolut, sehingga memudahkan dalam menilai akurasi relatif meskipun distribusi kelas tidak seimbang.

Dalam matriks tersebut, nilai 0.85 muncul pada perpotongan baris dan kolom untuk kelas *Helmet*, yang mengindikasikan bahwa sebesar 85% dari data yang benar-benar termasuk kategori *Helmet* berhasil diklasifikasikan dengan tepat oleh model. Ini merupakan sinyal bahwa model memiliki kemampuan yang cukup baik dalam mengenali objek helm. Namun demikian, masih terdapat 15% kasus di mana objek *Helmet* diprediksi sebagai *Background*, seperti terlihat pada nilai 0.15 di sel baris *Background* dan kolom *Helmet*. Kesalahan ini tergolong dalam kategori *false negative*, karena model gagal mendeteksi *Helmet* yang sebenarnya ada.

Di sisi lain, performa model pada kelas *Background* sangat memuaskan. Nilai 1.00 atau 100% pada sel baris *Background* dan kolom *Background* menunjukkan bahwa seluruh data yang benar-benar merupakan *Background* berhasil diprediksi dengan benar oleh model. Tidak ditemukan kesalahan klasifikasi untuk kelas ini, yang berarti *false positive* dan *false negative* untuk *Background* adalah nol. Hal ini menunjukkan bahwa model memiliki kepekaan dan ketepatan tinggi dalam mengenali latar belakang, yang kemungkinan besar karena ciri visual *Background* yang lebih konsisten dan berbeda signifikan dari *Helmet*.

Perbedaan akurasi antara dua kelas ini menunjukkan bahwa model lebih unggul dalam mengenali latar belakang dibandingkan mengenali helm. Kelemahan dalam mendeteksi *Helmet* bisa jadi disebabkan oleh beberapa faktor, seperti pencahayaan yang buruk, objek helm yang tertutupi sebagian, atau kemiripan warna antara helm dan elemen latar belakang. Selain itu, distribusi data yang tidak seimbang juga bisa membuat model cenderung lebih 'percaya diri' dalam memprediksi *Background*, karena frekuensinya yang mungkin lebih dominan di dalam data pelatihan.

Normalisasi dalam *confusion matrix* ini memungkinkan kita untuk menganalisis performa model secara lebih adil dan proporsional. Dengan melihat rasio daripada

jumlah absolut, kita dapat mengetahui secara akurat seberapa besar proporsi data yang berhasil atau gagal diklasifikasikan oleh model. Ini sangat penting terutama ketika jumlah sampel antar kelas tidak sama, karena model bisa saja tampak memiliki akurasi tinggi hanya karena didominasi oleh satu kelas yang mudah dikenali.

Secara keseluruhan, dapat disimpulkan bahwa meskipun model sudah menunjukkan performa yang sangat baik dalam mengklasifikasikan *Background*, masih terdapat ruang perbaikan dalam mendeteksi kelas *Helmet*. Untuk meningkatkan akurasi tersebut, beberapa pendekatan dapat diterapkan, seperti penambahan jumlah data kelas *Helmet*, penerapan data *augmentation* untuk memperkaya variasi visual, serta pemanfaatan metode pembelajaran yang lebih fokus terhadap kelas minoritas, misalnya dengan menggunakan fungsi kerugian seperti *focal loss*. Model juga dapat dioptimalkan lebih lanjut melalui tuning parameter dan pengembangan arsitektur jaringan yang lebih adaptif terhadap variasi objek pada kelas *Helmet*.

Selain itu, apabila model ini digunakan untuk aplikasi nyata seperti sistem pemantauan penggunaan helm, maka penting untuk meminimalkan kesalahan klasifikasi *Helmet* sebagai *Background*. Hal ini penting karena kegagalan mendeteksi helm dapat berdampak besar dalam konteks keselamatan kerja atau penerapan peraturan. Oleh karena itu, evaluasi melalui *confusion matrix* seperti ini sangat membantu dalam mengidentifikasi kelemahan model, sekaligus menjadi dasar pengambilan keputusan untuk proses pengembangan lebih lanjut.

### Pengujian Aplikasi

Pada tahap ini, dilakukan serangkaian pengujian terhadap aplikasi percobaan yang dirancang untuk mendeteksi penggunaan helm melalui sistem klasifikasi berbasis YOLOv8. Tujuan dari pengujian ini adalah untuk mengevaluasi performa dan akurasi dari sistem dalam mengidentifikasi keberadaan helm pada pengguna dalam berbagai kondisi. Hasil yang diperoleh menunjukkan bahwa sistem mampu mendeteksi objek helm dengan status klasifikasi berhasil pada seluruh percobaan yang dilakukan, meskipun dengan tingkat akurasi yang bervariasi.

#### a. Pengujian subjek mengenakan helm

Tabel 2 Pengujian pada subjek mengenakan helm

Percobaan Deteksi Menggunakan Helm	Hasil Klasifikasi	Hasil
	Berhasil	74% dan 62%
	Berhasil	51% dan 36%

Berdasarkan hasil yang dihimpun dari lima skenario uji coba yang berbeda, diperoleh tingkat keberhasilan sebagai berikut: 74% dan 62% pada percobaan pertama, 51% dan 36% pada percobaan kedua. Ketidak konsistenan nilai klasifikasi ini dapat

disebabkan oleh berbagai faktor eksternal maupun internal yang mempengaruhi proses deteksi, seperti pencahayaan yang kurang merata, kualitas dan sudut pengambilan gambar, variasi bentuk dan warna helm, serta posisi subjek dalam *frame*. Misalnya, pencahayaan rendah atau bayangan yang menutupi sebagian helm bisa menyebabkan sistem mengalami kesulitan dalam membedakan antara pengguna yang memakai helm dan yang tidak. Selain itu, variasi atribut visual helm (seperti helm setengah wajah atau tertutup sebagian oleh objek lain) juga turut memengaruhi hasil klasifikasi.

### b. Pengujian subjek tidak mengenakan helm

Tabel 3 Pengujian subjek tanpa helm

Percobaan subjek tanpa helm	Hasil Klasifikasi	Hasil
	Berhasil	81%
	Berhasil	70%

Dari dua percobaan yang telah dilakukan dalam skenario ini, seluruh hasil menunjukkan keberhasilan klasifikasi, dengan tingkat akurasi yang terbilang tinggi dan cukup stabil. Secara rinci, sistem berhasil mengidentifikasi subjek tanpa helm dengan akurasi sebesar 81% pada percobaan pertama, diikuti oleh 70% pada percobaan kedua. Dapat disimpulkan bahwa sistem memiliki sensitivitas yang baik dalam mengenali kondisi di mana helm tidak digunakan oleh individu yang terekam dalam gambar atau video.

Tingkat keberhasilan yang relatif tinggi ini menunjukkan bahwa sistem memiliki kecenderungan yang kuat dalam mengenali fitur visual khas dari kepala manusia yang tidak tertutupi oleh pelindung kepala. Deteksi terhadap area wajah, rambut, telinga, dan bentuk kepala tampaknya lebih mudah dikenali oleh algoritma, mengingat objek-objek tersebut memiliki konsistensi bentuk dan pola visual yang lebih seragam dibandingkan dengan keberagaman bentuk dan warna helm yang sering kali menjadi tantangan dalam klasifikasi. Kondisi ini memberi keuntungan tersendiri bagi sistem, karena variabel visual pada kepala manusia cenderung memiliki kemiripan yang tinggi antarsubjek, sehingga model dapat melakukan generalisasi dengan lebih efektif.

### KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem deteksi otomatis helm berbasis algoritma YOLOv8 yang mampu bekerja secara *real-time* menggunakan kamera CCTV. Sistem ini dirancang untuk mendukung penerapan Keselamatan dan Kesehatan Kerja (K3) dengan tujuan utama meningkatkan kepatuhan pekerja dalam penggunaan helm keselamatan di proyek konstruksi.

Berdasarkan hasil pengujian, sistem mampu mendeteksi pekerja berhelm dengan akurasi 74% dan pekerja tanpa helm dengan akurasi 81%. Nilai presisi pada kategori

*Helmet* mencapai 85%, sedangkan kategori *No Helmet* memperoleh presisi sempurna (100%). Hasil ini membuktikan bahwa YOLOv8 memiliki kinerja yang baik dalam membedakan pekerja yang patuh maupun yang melanggar aturan penggunaan APD.

Dari penelitian ini dapat disimpulkan bahwa:

1. Sistem deteksi otomatis helm yang dikembangkan telah memenuhi tujuan penelitian, yaitu menghasilkan model berbasis YOLOv8 yang mampu bekerja secara *real-time* dalam mendeteksi kepatuhan penggunaan helm.
2. Kinerja sistem menunjukkan efektivitas yang lebih tinggi pada kategori *No Helmet*, sedangkan akurasi pada kategori *Helmet* masih dapat ditingkatkan dengan menambah variasi dataset dan strategi augmentasi.
3. Implementasi sistem ini dapat menjadi solusi praktis untuk membantu pengawas proyek dalam meningkatkan efektivitas pengawasan K3, sekaligus menurunkan risiko kecelakaan kerja.

Penelitian ini juga membuka peluang pengembangan lanjutan, seperti perluasan deteksi pada jenis APD lainnya (misalnya rompi, sepatu keselamatan, atau sarung tangan), serta integrasi dengan sistem peringatan otomatis berbasis *IoT* agar penerapan K3 dapat berjalan lebih komprehensif.

#### DAFTAR PUSTAKA

- Cahyanti, M., Ridwan, M., & Septian, D. (2025). *A Real-Time Helmet Detection System Based on YOLOv8 to Support Traffic Law Enforcement Sistem Deteksi Penggunaan Helm Secara Real-Time Berbasis YOLOv8 untuk Mendukung Penegakan Hukum Lalu Lintas*. 29(1), 1–10. <https://doi.org/10.46984/sebatik.v29i1.2585>
- Direja, A. F., Cahyana, Y., Baihaqi, K. A., Engineering, I., Buana, U., & Karawang, P. (2024). *IMPLEMENTATION OF THE YOLOV8 METHOD TO DETECT WORK SAFETY*. 5(3), 865–871.
- Drantantiyas, N. D. G., Yulita, W., Ridwan, N. T., Ramadhani, U. A., Kesuma, R. I., Rakhman, A. Z., Bagaskara, R., Miranto, A., & Mufidah, Z. (2023). Performasi Deteksi Jumlah Manusia Menggunakan YOLOv8. *JASIEK (Jurnal Aplikasi Sains, Informasi, Elektronika Dan Komputer)*, 5(2), 63–68. <https://doi.org/10.26905/jasiek.v5i2.11605>
- Fasounaki, M., Yüce, E. B., Öncül, S., & Ince, G. (2021). CNN-based Text-independent Automatic Speaker Identification Using Short Utterances. *Proceedings - 6th International Conference on Computer Science and Engineering, UBMK 2021, 01*, 413–418. <https://doi.org/10.1109/UBMK52708.2021.9559031>
- Fatkhin, N., & Fadjeri, A. (2024). Pembelajaran Mesin Untuk Deteksi Helm Keselamatan Menggunakan Algoritma YOLOv8. *Jurnal Ilmiah SINUS*, 22(2), 77. <https://doi.org/10.30646/sinus.v22i2.843>
- Hartono, W., Handayani, D., & Sabrina, F. (2023). *Analisis Manajemen Risiko Dampak Kecelakaan Kerja Pada Proyek Konstruksi Jembatan Beton dengan Standar AS / NZS 4360 : 2004*. 11(4), 384–390.
- Korban, S. M. (2025). *Sudah Menelan Korban , Pekerja Proyek MPP Indramayu Masih Membandel Tak Gunakan APD*. November 2022, 2–5.
- Nugroho, B., & Puspaningrum, E. Y. (2021). Kinerja Metode CNN untuk Klasifikasi Pneumonia dengan Variasi Ukuran Citra Input. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 8(3), 533–538. <https://doi.org/10.25126/jtiik.2021834515>
- Satrio, B., Poetro, W., Mulyono, S., & Pramesti, V. A. (n.d.). *Prediksi Penyakit Batu Ginjal dengan Menerapkan Convolutional Neural Network*. 153–162.

- Steven Immanuel Sihombing, R., Abadi Harahap, W., & Kurnia Rahman, W. (2024). Implementasi Yolo V8 Untuk Mendeteksi Mata Uang Rupiah Emisi Tahun 2022 Ber-Output Audio. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4), 5900–5905. <https://doi.org/10.36040/jati.v8i4.10099>
- Susanti, S. (2023a). Deteksi Helm Otomatis Untuk Keselamatan Kerja di Tempat Proyek Berbasis Yolo. 9(1), 28–32.
- Susanti, S. (2023b). Deteksi Helm Otomatis Untuk Keselamatan Kerja di Tempat Proyek Berbasis Yolo. *Proceeding of Applied Science*, 9(1), 28–32.
- Ali, A., Hussain, M., & Khan, F. (2023). Intelligent helmet detection using YOLOv8: A case study on construction safety. *IEEE Access*, 11, 47659–47670. <https://doi.org/10.1109/ACCESS.2023.3274561>
- Cheng, L., Zhou, Y., & Liu, P. (2025). Advances in real-time PPE detection using YOLOv8 in smart construction. *Journal of Safety Research*, 82, 45–58. <https://doi.org/10.1016/j.jsr.2025.04.006>
- Huang, J., Wang, X., & Zhao, L. (2024). Smart construction site safety monitoring using YOLOv8 and IoT-based cameras. *Automation in Construction*, 165, 105024. <https://doi.org/10.1016/j.autcon.2024.105024>
- Kurniawan, R., Lestari, I., & Putri, S. (2022). Implementasi deep learning dalam pengawasan keselamatan kerja berbasis video. *Jurnal Rekayasa Sistem Informasi*, 14(2), 99–110.
- Rahman, A., Prasetyo, B., & Widodo, T. (2023). Penerapan YOLOv8 untuk deteksi objek pada sistem cerdas berbasis video pengawasan. *Jurnal Teknologi Informasi Terapan*, 9(2), 144–153.
- Wijayanto, A., & Fitriani, D. (2024). Analisis implementasi computer vision untuk mendukung K3 pada industri konstruksi. *Jurnal Sistem Cerdas*, 5(1), 22–31.
- Zhang, Y., Li, W., & Chen, H. (2022). Real-time object detection for industrial safety monitoring using deep learning. *International Journal of Computer Vision and Applications*, 18(3), 211–225. <https://doi.org/10.1007/s41095-022-0198-7>