



## IMPLEMENTASI PERBANDINGAN YOLO V8 DAN YOLO V11 DALAM PENERAPAN TATA TERTIB BERPAKAIAN DI LINGKUNGAN KAMPUS STUDI KASUS UNIVERSITAS ESA UNGGUL KAMPUS BEKASI

Muhammad Febri Yudhi<sup>1</sup>, Nixon Erzed<sup>2</sup>, Yulhendri<sup>3</sup>, Jefry Sunupurwa Asri<sup>4</sup>

<sup>1-4</sup>Teknik Informatika, Universitas Esa Unggul

<sup>1</sup>[mfebriyudhi09@gmail.com](mailto:mfebriyudhi09@gmail.com)

### **Abstract**

*The consistent enforcement of dress code regulations is an important approach to creating an ethical campus environment. This study aims to develop an automated dress code violation detection system utilizing artificial intelligence technology with the YOLOv8 and YOLOv11 algorithm algorithm. The system is designed to detect categories of compliant and non-compliant clothing in real-time through cameras and provide automatic notifications when violations are detected. The model, trained with a dataset of clothing images, is evaluated using metrics such as precision, recall, and mean Average Precision (mAP). Training results show that the model achieves high accuracy in detecting dress code violations, supporting the goal of enhancing monitoring efficiency and reducing the violation rate. At Esa Unggul University, Bekasi Campus, this system also contributes to modernizing campus management through technology, fostering the creation of a disciplined academic environment aligned with the values of educational institutions.*

**Keyword:** *Neat clothing arrangement, artificial intelligence, real-time monitoring.*

### **Article History:**

Received: February 2025

Reviewed: February 2025

Published: February 20254

Plagiarism Checker No 234

Prefix DOI :

10.8734/Kohesi.v1i2.365

Copyright : Author

Publish by : Kohesi



This work is licensed under

a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)

## **I. PENDAHULUAN**

Pendidikan adalah proses pembelajaran yang bertujuan untuk mengembangkan pengetahuan, keterampilan, nilai, dan karakter individu agar dapat berkontribusi secara positif dalam masyarakat. Pendidikan tidak hanya terbatas pada aspek akademik, tetapi juga mencakup pembentukan sikap, etika, dan disiplin yang menjadi bagian penting dalam dunia profesional. Institusi pendidikan memiliki peran krusial dalam membentuk karakter peserta didik, salah satunya melalui penerapan peraturan dan tata tertib yang mendukung lingkungan akademik yang kondusif.



Universitas Esa Unggul (UEU) merupakan salah satu institusi pendidikan tinggi di Indonesia yang berkomitmen untuk menyediakan pendidikan berkualitas serta mendukung pengembangan akademik dan profesionalisme mahasiswa. Dengan berbagai fakultas dan program studi yang ditawarkan, UEU terus berinovasi dalam menerapkan teknologi guna meningkatkan efektivitas operasional serta mendukung tata tertib dan kedisiplinan di lingkungan kampus. Salah satu aspek yang menjadi perhatian adalah pengawasan aturan berpakaian mahasiswa di area formal kampus, termasuk di ruangan administrasi. Universitas Esa Unggul Kampus Bekasi juga menginginkan etika berpakaian yang baik bagi mahasiswanya untuk menjaga citra akademik dan menciptakan suasana yang nyaman, terutama di ruangan administrasi.

Seiring berkembangnya teknologi kecerdasan buatan (Artificial Intelligence) di era revolusi industri 4.0, berbagai solusi inovatif dapat diterapkan untuk mengatasi tantangan ini. Salah satu pendekatan yang menjanjikan adalah deteksi objek berbasis YOLO (You Only Look Once). YOLO adalah algoritma deteksi objek real-time yang mampu mengidentifikasi dan mengenali objek dengan akurasi tinggi melalui pemrosesan data gambar. YOLO merupakan teknologi deteksi objek yang sedang berkembang dan telah banyak digunakan. Teknologi YOLO memiliki daya Tarik yang tinggi karena kompatibilitasnya yang tinggi. Setengah dekade terakhir didominasi dengan hadirnya YOLOv8 pada tahun 2022.[8]Teknologi ini telah menunjukkan keberhasilannya dalam berbagai aplikasi, seperti pengawasan keselamatan kerja, manajemen lalu lintas, dan pelestarian budaya.

YOLOv8 adalah versi terbaru dari algoritma deteksi objek YOLO (You Only Look Once), yang terkenal dengan kemampuan dan akurasi deteksi waktu nyata. YOLOv8 menggunakan mekanisme deteksi tanpa jangkar (anchor-free detection), jaringan piramida fitur yang ditingkatkan (feature pyramid network), dan fungsi kerugian yang dimodifikasi untuk meningkatkan kinerja dibandingkan dengan versi sebelumnya. Algoritma ini dirancang untuk mensegmentasi objek dalam gambar dengan memprediksi kotak batas (bounding box) dan kelas yang terkait menggunakan jaringan saraf konvolusional tunggal (single convolutional neural network).[1] Sejak diluncurkan pada tahun 2024 hadirnya YOLOv11 dimana itu adalah versi terbaru dari algoritma seri YOLO. Versi ini meningkatkan ekstraksi fitur dengan arsitektur backbone dan neck yang lebih canggih, serta meningkatkan kecepatan dan efisiensi melalui desain dan pipeline pelatihan yang dioptimalkan, sehingga menyeimbangkan presisi dan performa YOLOv11.[12]

Sebelum memilih YOLOv8, peneliti mempertimbangkan model-model teratas dalam pengujian segmentasi instance COCO, serta metode lama seperti TensorMask, PolarMask, SipMask, dan Mask R-CNN. Namun, peneliti memilih YOLOv8 sebagai metode dasar karena beberapa alasan. Pertama, algoritma YOLO secara konsisten menunjukkan kinerja yang kuat dalam tugas deteksi objek, menjadikannya kompetitif dan cocok untuk masalah segmentasi instance pejalan kaki. Kemampuan deteksi waktu nyata YOLOv8, khususnya, sangat penting untuk aplikasi yang melibatkan keselamatan publik dan tanggap darurat, sebagaimana dibuktikan oleh keberhasilan versi YOLO sebelumnya dalam berbagai tugas deteksi objek.



Kedua, YOLOv8 adalah metode yang sudah mapan dengan komunitas pengguna yang luas, menyediakan sumber daya implementasi yang dapat diakses. Ketiga, algoritma ini mudah diterapkan dan digunakan.[1] Penelitian sebelumnya menunjukkan potensi besar dari algoritma YOLO dalam berbagai bidang. Sebagai contoh, penerapan YOLOv5 dalam mendeteksi penggunaan alat pelindung diri (APD) di proyek konstruksi berhasil mencapai akurasi 83%. Penelitian tersebut menyoroti pentingnya kualitas dataset dan spesifikasi perangkat keras yang memadai untuk meningkatkan akurasi deteksi.[2] Selain itu, YOLOv8 berhasil mendeteksi kendaraan secara otomatis dalam penelitian manajemen lalu lintas, dengan akurasi hingga 86% serta F1 Score sebesar 85%. Hasil ini dicapai dengan integrasi algoritma DeepSORT untuk pelacakan objek dan efisiensi pengumpulan data menggunakan platform seperti Roboflow.[3] Selain itu penelitian sebelumnya yang berfokus pada klasifikasi jenis kebaya tradisional menggunakan YOLOv8 dengan data augmentasi mencatat akurasi rata-rata 94%, menunjukkan kemampuan algoritma ini dalam mendeteksi perbedaan halus antarjenis kebaya untuk melestarikan budaya Indonesia.[4]

Penelitian ini bertujuan untuk menerapkan algoritma YOLOv8 dalam pengawasan tata tertib berpakaian di Universitas Esa Unggul Kampus Bekasi, khususnya di ruangan administrasi. Dengan melatih model deteksi objek menggunakan dataset gambar pakaian yang dilabeli secara cermat, sistem ini diharapkan mampu mendeteksi pelanggaran tata tertib secara real-time dengan tingkat akurasi yang tinggi. Melalui evaluasi menggunakan metrik seperti precision, recall, dan F1 Score, penelitian ini diharapkan dapat memberikan solusi inovatif untuk menciptakan lingkungan kampus yang lebih profesional, efisien, dan modern.

Penegakan tata tertib berpakaian yang konsisten menjadi tantangan tersendiri di lingkungan kampus Universitas Esa Unggul Kampus Bekasi, khususnya di ruang administrasi. Pengawasan manual terhadap tata tertib berpakaian sering kali membutuhkan waktu dan tenaga yang signifikan, serta rentan terhadap kesalahan manusia. Hal ini menyebabkan pelanggaran sering kali tidak terdeteksi secara konsisten, sehingga mengurangi efektivitas penerapan peraturan tersebut. [5]

Tantangan lain yang dihadapi adalah kurangnya standar yang jelas dalam proses verifikasi kepatuhan berpakaian. Ketergantungan pada pengamatan manual dan subjektivitas penilaian kerap menimbulkan ketidakadilan dan kebingungan bagi mahasiswa. Selain itu, beban kerja petugas administrasi yang tinggi memperparah situasi ini, menjadikannya sulit untuk memberikan peringatan atau tindakan yang tepat.

Dalam era revolusi industri 4.0, teknologi telah memengaruhi berbagai aspek kehidupan manusia, termasuk cara berkomunikasi, bekerja, belajar, dan menyelesaikan tugas-tugas yang sebelumnya sulit dilakukan tanpa bantuan teknologi. Salah satu kemajuan teknologi yang terkenal adalah kecerdasan buatan (Artificial Intelligence), yang memungkinkan penyelesaian masalah kompleks dengan efektivitas dan efisiensi tinggi dalam pengolahan data dan waktu[6] sehingga tidak hanya mengurangi beban kerja petugas tetapi juga meningkatkan konsistensi dan keadilan dalam penegakan aturan.



Melalui penelitian ini, solusi berbasis YOLOv8 dikembangkan untuk mendeteksi pelanggaran tata tertib berpakaian secara otomatis, dengan memberikan notifikasi real-time kepada petugas administrasi. Hal ini diharapkan dapat meningkatkan efisiensi pengawasan, menurunkan tingkat pelanggaran, dan menciptakan lingkungan kampus yang lebih profesional.

## II. METODE

Bagian ini memaparkan secara detail hasil penelitian dan analisis yang telah dilakukan terkait akurasi deteksi objek, khususnya dalam mendeteksi kepatuhan tata tertib berpakaian mahasiswa saat memasuki ruang administrasi Universitas Esa Unggul Kampus Bekasi. Penelitian ini berfokus pada identifikasi pakaian yang sesuai atau tidak sesuai dengan peraturan yang berlaku, yang mencakup 6 kelas data dengan variasi dan jumlah berbeda. Secara keseluruhan, dataset yang digunakan terdiri dari 1162 gambar yang telah diproses dan dioptimalkan menggunakan Roboflow.

Roboflow merupakan platform web yang memiliki fungsi berhubungan dengan kumpulan dataset. Roboflow adalah kerangka kerja pengembang computer vision untuk pengumpulan data yang lebih baik hingga prapemrosesan, dan teknik pelatihan model. Dengan menggunakan Roboflow dapat membagikan dataset sekaligus memproses dataset tersebut melakukan annotate atau menandai objek yang akan di deteksi menggunakan bounding box, selain itu dapat digunakan juga pre-processing pada dataset misalnya melakukan grayscale, dan juga augmentasi dengan menggunakan Roboflow. Untuk melakukan pengujian deteksi objek diperlukan sebuah dataset pada tahap akuisisi data (pengumpulan data), dimana tahap tersebut merupakan sebuah tantangan bagi peneliti untuk mengumpulkan sekumpulan data citra kendaraan yang baik untuk melakukan pendeteksian objek[3]

Kaggle merupakan salah satu platform terkemuka di bidang Data Science dan Machine Learning secara global, dengan lebih dari 6000 dataset dan komunitas ilmuwan terbesar saat ini [7]. Kaggle menyediakan platform bagi pengguna untuk bekerja sama dengan orang lain, menemukan dan membagikan kumpulan data, menggunakan notebook berbasis GPU, serta berkompetisi dengan data scientist lainnya dalam menyelesaikan tantangan data science.

Langkah – langkah penelitian oleh adalah sebagai berikut:



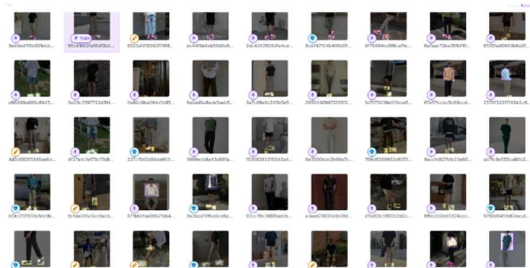
Gambar 1. Langkah-Langkah Penelitian

## 1 Pembuatan Model

### 1). Pengambilan Dataset

Proses pengambilan dataset dilakukan dengan mencari dan mengumpulkan gambar melalui media online dari berbagai sumber terpercaya. Dataset yang dikumpulkan mencakup gambar mahasiswa dengan pakaian yang sesuai aturan dan tidak sesuai aturan, berdasarkan kategori berikut:

- 1) Atasan sesuai aturan
- 2) Atasan tidak sesuai aturan
- 3) Bawahan sesuai aturan
- 4) Bawahan tidak sesuai aturan
- 5) Alas kaki sesuai aturan
- 6) Alas kaki tidak sesuai aturan



Gambar 2. Pengambilan Dataset

Tabel Jumlah Data

Nama Kelas	Jumlah Data
1) Atasan sesuai aturan	308
2) Atasan tidak sesuai aturan	323
3) Bawahan sesuai aturan	331
4) Bawahan tidak sesuai aturan	344
5) Alas kaki sesuai aturan	250
6) Alas kaki tidak sesuai aturan	250

Setelah semua data terkumpul, data tersebut akan dibagi menjadi tiga bagian: 70% untuk data train, 15% untuk data validasi, dan 15% data test. Pembagian ini dilakukan agar pengujian dapat menghasilkan nilai akurasi yang tinggi.

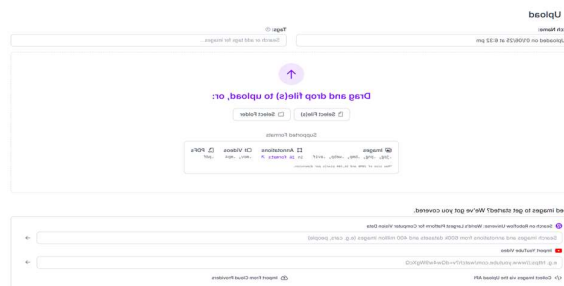
Berikut teks yang diekstrak dari gambar:

## 2). Pelabelan Gambar

Pada tahap ini, gambar-gambar diberi label untuk memungkinkan sistem mengenali nama objek yang akan dideteksi. Proses pelabelan dilakukan menggunakan aplikasi Roboflow, dengan tujuan menambahkan label pada gambar-gambar untuk mengidentifikasi apakah ada Peternak atau tidak, sesuai dengan pembagian dan kategori yang telah ditentukan sebelumnya.

### a. Upload Dataset

Data gambar yang telah dikumpulkan kemudian dimasukkan ke dalam Roboflow sebelum proses pelabelan dilakukan.



Gambar 3. Upload Dataset

### b. Pembuatan Kelas Dataset

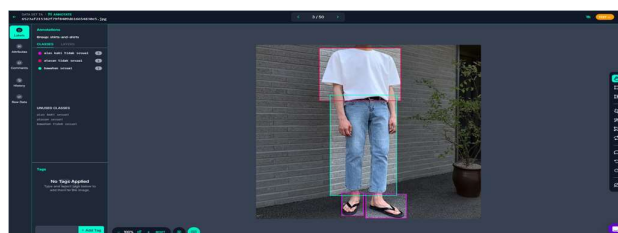
Tujuan pembuatan kelas ini adalah untuk mempermudah proses penentuan dataset gambar yang telah diberi *bounding box*. Kelas ini secara otomatis akan menghasilkan dua kelas yang sudah ditetapkan sebelumnya, di mana dataset akan dikelompokkan sesuai dengan kategori tersendiri.

COLOR	CLASS NAME	COUNT
Yellow	ada sapi sesuai	401
Pink	ada sapi tidak sesuai	401
Purple	tidak ada sapi sesuai	391
Red	tidak ada sapi tidak sesuai	391
Light Blue	bayangan sapi sesuai	391
Dark Blue	bayangan sapi tidak sesuai	391

Gambar 4. Pembuatan Kelas Dataset

### c. Proses Pelabelan

Proses pelabelan gambar dilakukan menggunakan Roboflow dengan menambahkan *bounding box* atau bingkai di sekitar objek dalam gambar. Setelah *bounding box* dibuat, kelas dengan kategori yang telah ditentukan sebelumnya akan muncul secara otomatis.



Gambar 5. Proses Pelabelan



### 3). Pembagian Dataset

Dalam dataset yang diupload ini menggunakan beberapa pengujian augmentasi, yaitu: Yang pertama pengelompokan dataset dilakukan menggunakan alat Roboflow untuk membaginya menjadi data train, validasi, dan test. Sebelum dilakukan augmentasi, dataset yang terdiri dari 356 gambar akan dibagi menjadi 60% untuk data train (215 gambar), 20% untuk data validasi (71 gambar), dan 20% untuk data test (70 gambar).



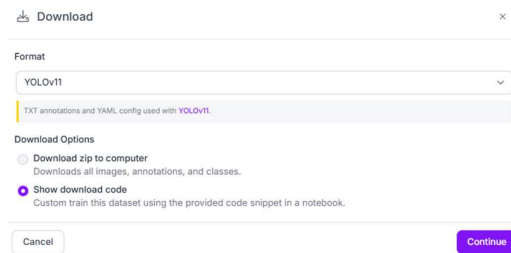
Gambar 6. Split Dataset Sebelum Augmentasi

### 4). Training

Data yang telah diproses sesuai kebutuhan akan diekspor untuk digunakan dalam pelatihan di Google Colab. Hasil ekspor dataset ini akan menghasilkan API yang akan digunakan di Google Colab untuk melatih model YOLOv8.

#### a. Export Dataset Roboflow

Setelah dataset dianotasi, dilakukan ekspor dataset dalam format yang kompatibel dengan YOLOv8.



Gambar 7. Proses Pembuatan API

Setelah proses ekspor dataset, Roboflow akan menyediakan API yang berisi dataset yang telah dianotasi. API ini selanjutnya akan digunakan untuk melatih model di Google Colab.

#### b. Training Kaggle

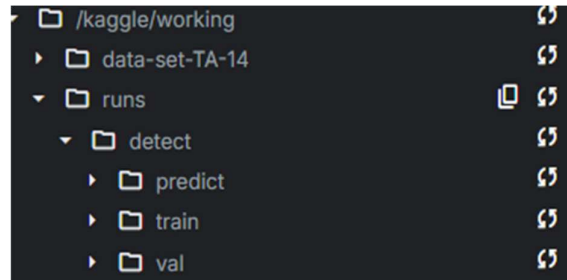
Pelatihan model dilakukan dengan memanfaatkan API yang diperoleh dari Roboflow. API ini akan diakses di Kaggle untuk melatih model. Dataset yang diunduh sudah terintegrasi dengan anotasi yang diperlukan untuk melatih model deteksi objek menggunakan YOLOv11 dan YOLOv8. Program ini memudahkan pengembang untuk memulai pelatihan model deteksi objek dengan dataset yang siap digunakan di lingkungan pengembangan seperti Kaggle.

Setelah API dari Roboflow bisa dan berhasil diakses, API tersebut akan digunakan untuk melatih model. Training dilakukan dengan dilakukan selama 100 epoch perbandingan. Gambar-gambar yang digunakan memiliki ukuran 640x640 piksel. Jumlah worker yang digunakan ditentukan oleh opsi workers. Perangkat dengan indeks 0 (GPU) digunakan untuk

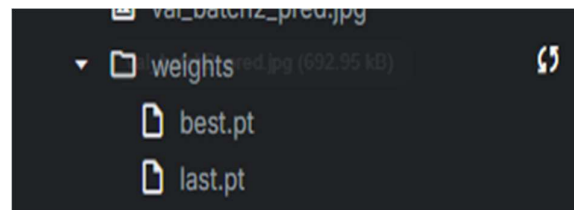




mempercepat pelatihan. Setelah melakukan pelatihan di Google Colab dengan menggunakan 70% data train, 15% data valid, dan 15% data test, diperoleh akurasi sebesar 84.8%. Setelah proses pelatihan selesai, akan ada berkas yang dihasilkan, yaitu `best.pt`. Berkas tersebut akan dipakai untuk menguji modelnya. Di dalam train ada folder `weight` yg berisi `bst.pt` hasil dari training kita.



Gambar 8. Model Dataset



Gambar 9. Model Dataset

### c. Evaluasi Model

Evaluasi akan dilakukan menggunakan metrik-metrik seperti *precision*, *recall*, dan *mean Average Precision* (mAP). Metrik *precision* mengukur sejauh mana model dapat mengidentifikasi objek dengan benar dari seluruh objek yang ditemukan. Metrik *recall* mengukur sejauh mana model dapat menemukan semua objek yang ada dalam dataset. mAP merupakan metrik yang menggabungkan *precision* dan *recall* untuk memberikan gambaran keseluruhan performa model. Hasil *training* dapat dilihat dari tabel 1 dan 2.

Tabel Hasil Validasi

Hasil Traing YOLOv11s

Ultralytics 8.3.40 Python-3.10.12 torch-2.4.1+cu121 CUDA:0 (Tesla T4, 15099MiB)  
YOLO11s summary (fused): 238 layers, 9,415,122 parameters, 0 gradients, 21.3 GFLOPs  
val: Scanning /kaggle/working/data-set-TA-2-1/valid/labels.cache... 177 images,  
class Images Instances Box(P) R mAP50 m

	all					
	177	484	0.892	0.844	0.886	0.619
alas kaki sesuai	74	148	0.797	0.676	0.783	0.414
alas kaki tidak sesuai	47	89	0.963	0.876	0.935	0.625
atasan sesuai	63	64	0.748	0.875	0.862	0.691
atasan tidak sesuai	51	52	0.976	0.783	0.886	0.685
bawahan sesuai	59	62	0.894	0.956	0.908	0.628
bawahan tidak sesuai	69	69	0.973	0.899	0.941	0.671

Speed: 2.0ms preprocess, 11.1ms inference, 0.0ms loss, 3.1ms postprocess per image  
Results saved to runs/detect/val  
Learn more at <https://docs.ultralytics.com/modes/val>

Hasil Traing YOLOv8s





```

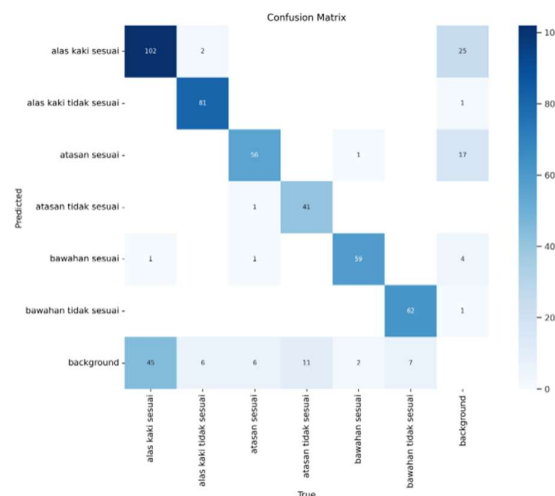
Ultralytics YOLOv8.2.103 Python-3.10.12 torch-2.4.1+cu121 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 168 layers, 11,127,906 parameters, 0 gradients, 28.4 GFLOPs
val: Scanning /kaggle/working/data-set-TA-2-1/valid/labels.cache... 177 images,
      Class      Images  Instances  Box(P)      R      mAP50  m
      all         177        484      0.834      0.859      0.902  0.599
      alas kaki sesuai      74        148      0.82      0.649      0.83      0.457
      alas kaki tidak sesuai  47         89      0.896      0.899      0.944      0.568
      atasan sesuai         63         64      0.743      0.904      0.906      0.721
      atasan tidak sesuai    51         52      0.785      0.788      0.83      0.604
      bawahan sesuai        59         62      0.807      0.984      0.948      0.609
      bawahan tidak sesuai   69         69      0.954      0.928      0.951      0.633
Speed: 1.0ms preprocess, 11.2ms inference, 0.0ms loss, 2.6ms postprocess per image
Results saved to runs/detect/val
Learn more at https://docs.ultralytics.com/modes/val
    
```

Model pada Gambar 1 menggunakan YOLOv8 versi 8.3.40 dengan 238 layer, 9,4 juta parameter, dan 21,3 GFLOPS sehingga cocok untuk sistem dengan keterbatasan komputasi, menghasilkan presisi (P) 0.892, recall (R) 0.844, dan mAP50 0.886. Sementara itu, Gambar 2 menggunakan YOLOv8 versi 8.2.103 dengan 168 layer, 11,1 juta parameter, dan 28,4 GFLOPS sehingga lebih cocok untuk tugas deteksi yang membutuhkan akurasi tinggi, memberikan presisi 0.834, recall 0.859, dan mAP50 0.902.

Perbedaan utama terletak pada arsitektur model, di mana Gambar 1 lebih ringan dan efisien, sementara Gambar 2 memiliki akurasi rata-rata lebih baik dalam mendeteksi seluruh kelas. Selain itu, kecepatan pemrosesan Gambar 2 sedikit lebih tinggi (1.9 ms vs. 2.0 ms per gambar). YOLOv8 adalah model deteksi objek satu tahap yang dikembangkan dengan PyTorch, menggunakan jaringan konvolusi penuh yang dirancang untuk memprediksi bounding boxes dan probabilitas kelas objek secara real-time dengan kecepatan lebih tinggi dibandingkan YOLOv11. Meskipun YOLOv11 memiliki arsitektur backbone dan neck yang lebih canggih untuk meningkatkan ekstraksi fitur, model ini tetap mengutamakan keseimbangan antara presisi dan performa melalui desain serta pipeline pelatihan yang dioptimalkan[12].

## Perbandingan Confusion Matrix

### 1.Yolov11s

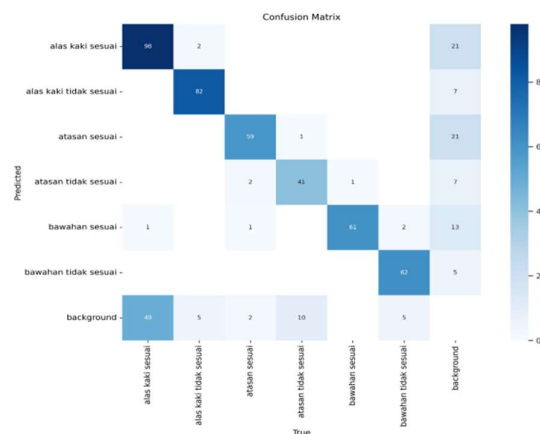


Gambar 10. Confusion Matrix Yolov11s



Confusion matrix adalah alat visualisasi yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan merepresentasikan hubungan antara prediksi model (baris) dan label sebenarnya (kolom). Dalam matrix ini, terdapat 7 kelas: "alas kaki sesuai", "alas kaki tidak sesuai", "atasan sesuai", "atasan tidak sesuai", "bawahan sesuai", "bawahan tidak sesuai", dan "background". Sel pada diagonal utama menunjukkan jumlah data yang diprediksi benar (True Positive/TP), sedangkan sel di luar diagonal mencerminkan kesalahan prediksi (False Positive/FP atau False Negative/FN). Akurasi model dihitung dengan membagi jumlah TP dengan total data, sementara presisi, recall, dan F1-score memberikan analisis mendalam terhadap kinerja tiap kelas. Misalnya, pada kelas "alas kaki sesuai", model memiliki TP sebesar 102, FP sebesar 2, dan FN sebesar 1, yang menunjukkan kinerja baik untuk kelas ini. Namun, terdapat kebingungan yang cukup tinggi antara kelas "background" dan beberapa kelas lainnya. Dengan analisis mendetail pada metrik seperti presisi dan recall, kita dapat memahami area yang perlu ditingkatkan untuk memperbaiki performa model.

## 2.Yolov8s



Gambar 11. Confusion Matrix Yolov8s

Confusion matrix adalah alat evaluasi performa model klasifikasi yang memvisualisasikan hubungan antara prediksi model (baris) dan label sebenarnya (kolom), dengan angka di setiap sel menunjukkan jumlah instance pada kombinasi prediksi dan label tertentu. Pada matriks ini, model mengklasifikasikan data ke dalam tujuh kategori: "alas kaki sesuai", "alas kaki tidak sesuai", "atasan sesuai", "atasan tidak sesuai", "bawahan sesuai", "bawahan tidak sesuai", dan "background". True Positive (TP) dicatat pada diagonal utama, seperti angka 96 yang menunjukkan data "alas kaki sesuai" diprediksi benar. Sebaliknya, angka di luar diagonal mencerminkan kesalahan, seperti False Positive (FP) atau False Negative (FN), yang misalnya terjadi ketika 2 data "alas kaki tidak sesuai" salah diprediksi sebagai "alas kaki sesuai". Analisis metrik seperti akurasi (rasio total TP terhadap seluruh data), presisi (proporsi prediksi positif yang benar), recall (proporsi data sebenarnya positif yang terprediksi benar), dan F1-score membantu mengevaluasi kekuatan dan kelemahan model. Dalam kasus ini, model menunjukkan performa baik pada beberapa kategori seperti



"alas kaki sesuai", "atasan sesuai", dan "bawahan sesuai", namun kesulitan membedakan "background" dari beberapa kategori lain. Dengan memanfaatkan insight dari matriks ini, area perbaikan dapat diidentifikasi untuk meningkatkan akurasi model secara keseluruhan.

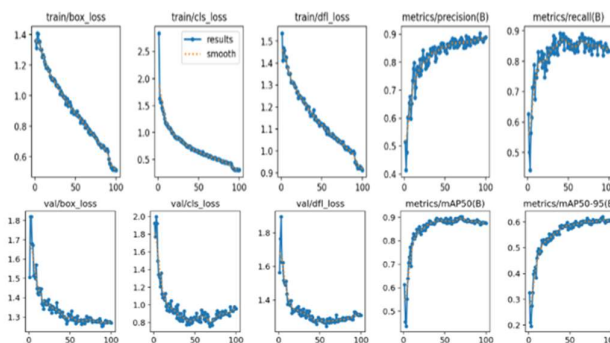
**Tabel Perbandingan confusion matrix**

Aspek	Gambar 1	Gambar 2
"alas kaki sesuai"	102 prediksi benar.	96 prediksi benar. Sedikit penurunan.
"background"	Banyak kebingungan dengan kelas lain, terutama "alas kaki sesuai" dan "bawahan tidak sesuai".	Kebingungan dengan kelas lain berkurang secara signifikan. Peningkatan signifikan.
Kelas lain	Beberapa kesalahan klasifikasi. Perlu analisis presisi dan recall untuk detail lebih lanjut.	Beberapa kesalahan klasifikasi. Perubahan perlu dianalisis dengan presisi dan recall, tetapi secara umum, model tampaknya sedikit lebih baik dalam membedakan kelas-kelas pakaian, terutama karena perbaikan pada "background".
Kesimpulan Umum	Model menunjukkan kinerja yang cukup baik, tetapi masih ada kebingungan, terutama dengan "background".	Model menunjukkan peningkatan yang signifikan dalam membedakan "background" dan sedikit perbaikan pada beberapa kelas lain, meskipun ada sedikit penurunan pada prediksi benar untuk "alas kaki sesuai".

Model yang direpresentasikan oleh Gambar 2 umumnya *lebih baik* daripada model pada Gambar 1, terutama karena kemampuannya yang lebih baik dalam mengklasifikasikan "background". Meskipun ada sedikit penurunan pada prediksi benar "alas kaki sesuai", peningkatan pada "background" secara keseluruhan lebih signifikan. Untuk analisis yang lebih mendalam, disarankan untuk menghitung metrik presisi, recall, dan F1-score untuk setiap kelas pada kedua *confusion matrix*.

## Evaluasi Pelatihan Model Deteksi Objek

### 1.Yolov11s



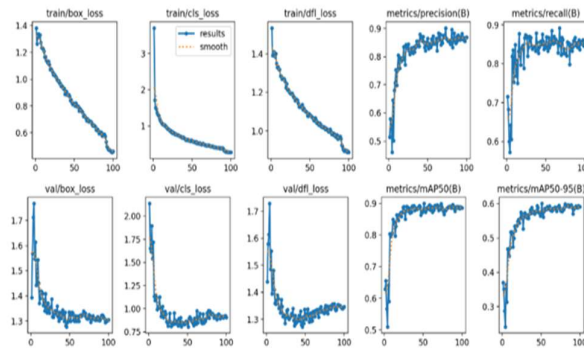
**Gambar 12. Grafik Pelatihan Model Yolov11s**

Gambar tersebut menunjukkan grafik metrik pelatihan dan validasi untuk model object detection, kemungkinan besar menggunakan arsitektur YOLO. Grafik pelatihan seperti *train/box\_loss*, *train/cls\_loss*, dan *train/dfl\_loss* menunjukkan penurunan loss yang stabil, mengindikasikan model semakin akurat dalam memprediksi bounding box,



mengklasifikasikan objek, dan menangani ketidakseimbangan kelas. Metrik  $precision(B)$  dan  $recall(B)$  juga meningkat, menunjukkan model memiliki false positive rendah dan mendeteksi sebagian besar objek dengan baik. Pada grafik validasi,  $val/box\_loss$ ,  $val/cls\_loss$ , dan  $val/df\_loss$  menurun dan mendekati nilai loss pelatihan, menandakan tidak ada indikasi overfitting yang signifikan. Metrik evaluasi seperti  $mAP50(B)$  dan  $mAP50-95(B)$  meningkat, menunjukkan akurasi yang baik pada berbagai ambang IoU. Meskipun ada fluktuasi pada recall dan beberapa loss di awal pelatihan, grafik secara keseluruhan menunjukkan bahwa model telah dilatih dengan baik. Untuk evaluasi lebih lanjut, penting membandingkan hasil ini dengan baseline atau model lain pada dataset yang sama

## 2.Yolov8s

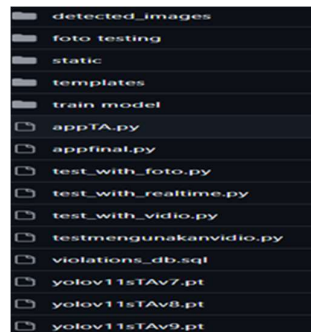


Gambar 13. Grafik Pelatihan Model YOLOv8s

Gambar tersebut menampilkan grafik metrik pelatihan dan validasi untuk model deteksi objek, kemungkinan besar menggunakan arsitektur seperti YOLO, yang menunjukkan perubahan performa model selama pelatihan. Grafik loss pelatihan ( $train/box\_loss$ ,  $train/cls\_loss$ ,  $train/df\_loss$ ) menurun secara konsisten, menandakan model semakin baik dalam memprediksi bounding box, mengklasifikasikan objek, dan menangani ketidakseimbangan kelas. Metrik  $precision(B)$  dan  $recall(B)$  meningkat, menunjukkan model memiliki deteksi positif yang akurat dan mampu menemukan sebagian besar objek. Grafik validasi ( $val/box\_loss$ ,  $val/cls\_loss$ ,  $val/df\_loss$ ) juga mengikuti tren serupa, mengindikasikan model dapat generalisasi dengan baik tanpa tanda overfitting signifikan. Metrik evaluasi seperti  $mAP50(B)$  dan  $mAP50-95(B)$  meningkat, menunjukkan akurasi yang membaik pada berbagai ambang batas tumpang tindih antara prediksi dan ground truth. Meskipun ada sedikit fluktuasi pada beberapa metrik validasi di awal pelatihan, ini masih dalam batas wajar. Secara keseluruhan, grafik ini mencerminkan pelatihan model yang baik dengan performa yang terus meningkat.

### III. HASIL DAN PEMBAHASAN

Langkah penting pada tahap pengujian adalah menggunakan model yang telah dilatih sebelumnya untuk deteksi objek secara *real-time*. Model 'best.pt', yang menunjukkan hasil terbaik selama pelatihan, akan dimuat ke dalam skrip Python menggunakan YOLOv8. Skrip ini akan mengambil video langsung dari kamera, memproses setiap *frame*, dan melakukan inferensi menggunakan model tersebut. Hasil deteksi akan ditampilkan dengan menggambar kotak pembatas dan label pada objek yang terdeteksi. Langkah ini memastikan bahwa sistem deteksi objek *real-time* menggunakan model yang optimal untuk memberikan hasil yang akurat dan andal saat diuji dengan data langsung.



Gambar 14. Testing Model

Model YOLO yang telah dilatih digunakan untuk deteksi objek secara *real-time* melalui webcam. Setelah membuka webcam dan memuat model best(2).pt, program memasuki *loop* utama di mana setiap *frame* dari webcam diambil. Setiap detik, model memeriksa *frame* untuk deteksi objek, menghasilkan kotak pembatas dan tingkat kepastian. Jika tingkat kepastian melebihi 50%, program menggambar kotak di sekitar objek dan menambahkan label kelas serta tingkat kepastian pada objek yang terdeteksi.

1) Hasil dari pengujian *real-time* menggunakan Yolov8s:

Gambar Deteksi	Gambar Hasil Deteksi	Nilai Akurasi
		Atasan sesuai : 0.76 Bawahan sesuai: 0.76 Alas kaki tidak : 0.70
		Atasan sesuai: 0.75 Bawahan sesuai: 0.80 Alas kaki sesuai: 0.25



		Atasan sesuai :0.90 Bawahan : tidak sesuai : 0.89 Alas kaki : tidak sesuai: 0.74
		Atasan tidak sesuai standar : 0.88 Bawahan tidak sesuai standar :0.91 Alas kaki sesuai: 0.29
		Atasan tidak sesuai standar :0.89 Bawahan : sesuai standar:0.86 Alas kaki sesuai:0.30
		Atasan tidak sesuai standar :0.90 Bawahan : sesuai standar:0.84 Alas tidak sesuai:0.80

Gambar 15. Hasil Deteksi Pengujian Model YOLOv8s

2) Hasil dari pengujian *real-time* menggunakan YOLOv11s:

Gamabr Deteksi	Gambar Hasil Deteksi	Nilai Akurasi
		Atasan sesuai : 0.26 Bawahan sesuai: 0.43 Alas kaki tidak : 0.75
		Atasan sesuai: 0.46 Bawahan sesuai: Tidak terbaca Alas kaki sesuai: 0.56





		Atasan sesuai : 0.43 Bawahan tidak sesuai : Tidak terbaca Alas kaki tidak sesuai: 0.83
		Atasan tidak sesuai aturan : 0.73 Bawahan tidak sesuai aturan : 0.77 Alas kaki sesuai: 0.77
		Atasan tidak sesuai aturan : 0.78 Bawahan sesuai aturan: 0.82 Alas kaki sesuai: 0.53
		Atasan tidak sesuai aturan : 0.74 Bawahan sesuai aturan: 0.70 Alas kaki tidak sesuai: 0.79

Gambar 16. Hasil Deteksi Pengujian Model Yolov11s

Gambar di atas menunjukkan hasil pengujian deteksi pakaian secara real-time menggunakan model YOLO yang telah dilatih untuk mendeteksi kesesuaian pakaian mahasiswa dengan aturan tata tertib berpakaian di lingkungan kampus. Dalam pengujian ini, webcam digunakan untuk menangkap video langsung, dan setiap frame dari video tersebut diproses oleh model YOLO untuk mendeteksi dan mengklasifikasikan pakaian.

Pengujian real-time dengan YOLOv8s menunjukkan bahwa model ini dapat mendeteksi kesesuaian pakaian dengan nilai akurasi yang cukup tinggi. Pada beberapa gambar, model ini mendeteksi atasan sesuai dengan tingkat akurasi berkisar antara 0.75 hingga 0.90, sementara untuk bawahan sesuai mencapai 0.78 hingga 0.89. Namun, deteksi alas kaki sesuai memiliki variabilitas yang lebih besar, dengan nilai kepercayaan mulai dari 0.25 hingga 0.80. Sebaliknya, pengujian real-time dengan YOLOv11s menunjukkan hasil yang lebih bervariasi. Model ini mengalami kesulitan dalam mendeteksi beberapa kategori, terutama bawahan sesuai, di mana beberapa gambar tidak dapat terbaca. Akurasi deteksi atasan sesuai berkisar antara 0.26 hingga





0.78, sedangkan bawahan sesuai memiliki nilai yang lebih tidak stabil, dengan beberapa data tidak terbaca. Namun, model ini menunjukkan performa lebih baik dalam mendeteksi alas kaki tidak sesuai, dengan tingkat akurasi yang lebih tinggi dibandingkan YOLOv8s, mencapai 0.75 hingga 0.83..

Proses pengujian dilakukan dengan menggunakan 6 kelas data, yaitu atasan sesuai (308 data), atasan tidak sesuai (323 data), bawahan sesuai (331 data), bawahan tidak sesuai (344 data), alas kaki sesuai (250 data), dan alas kaki tidak sesuai (250 data). Meskipun hasil pengujian menunjukkan tingkat kepastian yang cukup tinggi, terdapat beberapa data dengan tingkat kepastian rendah. Kesulitan ini dapat disebabkan oleh kompleksitas objek dalam frame, adanya atribut pakaian yang beragam, serta banyaknya benda dalam satu area, yang menyebabkan deteksi lebih dari satu objek secara bersamaan.

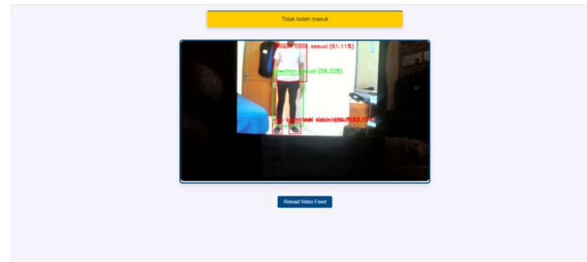
Hasil pengujian menunjukkan bahwa model YOLO dapat mendeteksi kesesuaian pakaian mahasiswa dengan tingkat akurasi yang tinggi. Dalam pengujian keseluruhan dengan 100 epoch, YOLOv8s memiliki mAP50 sebesar 90.2% dan mAP50-95 sebesar 59.9%, sedangkan YOLOv11s memiliki mAP50 sebesar 88.6% dan mAP50-95 sebesar 61.9%. Hal ini menunjukkan bahwa setelah 100 epoch, YOLOv8s lebih unggul dalam mendeteksi objek dengan akurasi tinggi pada threshold 50%, sementara YOLOv11 lebih stabil pada berbagai threshold.

Berdasarkan performa per kelas, YOLOv8s lebih unggul dalam mendeteksi kategori atasan sesuai (mAP50: 90.6%), bawahan sesuai (mAP50: 94.8%), dan alas kaki sesuai (mAP50: 83%), menunjukkan kemampuannya yang baik dalam mengklasifikasikan pakaian yang memenuhi aturan. Sementara itu, YOLOv11s lebih unggul dalam mendeteksi kategori atasan tidak sesuai (mAP50: 88.6%), bawahan tidak sesuai (mAP50: 94.1%), dan alas kaki tidak sesuai (mAP50: 93.5%), yang menunjukkan kemampuannya lebih baik dalam mendeteksi pelanggaran tata tertib berpakaian.

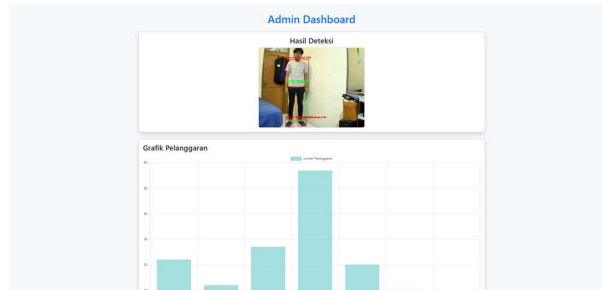
Secara keseluruhan, YOLOv8s lebih unggul dalam mendeteksi pakaian yang sesuai aturan, sedangkan YOLOv11s lebih baik dalam mendeteksi pelanggaran tata tertib berpakaian. Pengujian real-time menunjukkan bahwa kedua model memiliki performa yang cukup baik, tetapi masih terdapat beberapa kendala dalam mendeteksi objek tertentu, terutama alas kaki dan bawahan. Pada pengujian real-time menggunakan YOLOv8s, model menunjukkan akurasi tertinggi 0.90 untuk atasan sesuai, 0.89 untuk bawahan sesuai, dan 0.80 untuk alas kaki tidak sesuai. Sementara itu, pada YOLOv11s, akurasi terbaik yang dicapai adalah 0.78 untuk atasan tidak sesuai, 0.82 untuk bawahan sesuai, dan 0.83 untuk alas kaki tidak sesuai.

### **Hasil Implementasi Sistem dengan Menampilkan Notifikasi di Layar**

Aktivitas mahasiswa dideteksi secara real-time menggunakan kamera, kemudian dianalisis oleh algoritma YOLO. Jika terdeteksi adanya pelanggaran tata tertib berpakaian, sistem akan menampilkan notifikasi berupa teks di layar, dilengkapi dengan gambar yang telah diambil beserta bounding box yang menandai pelanggaran tersebut.



Gambar 17. Dashboard User



Gambar 18. Dashboard admin

Pada Gambar .menunjukkan hasil deteksi pelanggaran tata tertib berpakaian mahasiswa, di mana sistem berhasil mengidentifikasi pelanggaran dan menampilkan notifikasi otomatis di layar. Notifikasi tersebut mencakup teks informasi pelanggaran beserta gambar yang menampilkan area pelanggaran dengan bounding box.

#### IV. KESIMPULAN

Berdasarkan perancangan dan pengujian yang telah dilakukan dalam penelitian ini, diperoleh kesimpulan sebagai berikut:

- 1) Sistem berbasis **YOLOv8** telah berhasil diimplementasikan untuk mendeteksi pelanggaran tata tertib berpakaian di ruang layanan administrasi **Universitas Esa Unggul Kampus Bekasi**. Dengan menggunakan model deep learning, sistem mampu mengidentifikasi apakah seseorang mengenakan pakaian yang sesuai dengan ketentuan atau tidak. Hasil deteksi ditampilkan secara real-time, dan jika ditemukan pelanggaran, sistem secara otomatis merekam bukti visual serta mengirimkan pemberitahuan ke platform seperti **Telegram** untuk pengawasan lebih lanjut.
- 2) Framework **YOLOv8** bekerja dengan membagi gambar atau frame video menjadi grid dan menganalisis setiap bagian menggunakan **bounding box** serta **confidence score** untuk menentukan kelas pakaian yang dikenakan. Sistem dilatih menggunakan dataset yang berisi kategori pakaian sesuai dan tidak sesuai aturan, sehingga dapat mengklasifikasikan objek secara akurat. Proses ini dilakukan dalam waktu nyata (real-time), memungkinkan pemantauan langsung terhadap orang yang masuk ke area administrasi.
- 3) Berdasarkan hasil Pengujian real-time dengan YOLOv8s menunjukkan bahwa model ini dapat mendeteksi kesesuaian pakaian dengan nilai akurasi yang cukup tinggi. Pada beberapa gambar, model ini mendeteksi atasan sesuai dengan tingkat akurasi berkisar



antara 0.75 hingga 0.90, sementara untuk bawahan sesuai mencapai 0.78 hingga 0.89. Namun, deteksi alas kaki sesuai memiliki variabilitas yang lebih besar, dengan nilai kepercayaan mulai dari 0.25 hingga 0.80. Sebaliknya, pengujian real-time dengan YOLOv11s menunjukkan hasil yang lebih bervariasi. Model ini mengalami kesulitan dalam mendeteksi beberapa kategori, terutama bawahan sesuai, di mana beberapa gambar tidak dapat terbaca. Akurasi deteksi atasan sesuai berkisar antara 0.26 hingga 0.78, sedangkan bawahan sesuai memiliki nilai yang lebih tidak stabil, dengan beberapa data tidak terbaca. Namun, model ini menunjukkan performa lebih baik dalam mendeteksi alas kaki tidak sesuai, dengan tingkat akurasi yang lebih tinggi dibandingkan YOLOv8s, mencapai 0.75 hingga 0.83, yang dimana Hasil pengujian menunjukkan bahwa model YOLO dapat mendeteksi kesesuaian pakaian mahasiswa dengan tingkat akurasi yang tinggi. Dalam pengujian keseluruhan dengan 100 epoch, YOLOv8s memiliki mAP50 sebesar 90.2% dan mAP50-95 sebesar 59.9%, sedangkan YOLOv11s memiliki mAP50 sebesar 88.6% dan mAP50-95 sebesar 61.9%. Hal ini menunjukkan bahwa setelah 100 epoch, YOLOv8s lebih unggul dalam mendeteksi objek dengan akurasi tinggi pada threshold 50%, sementara YOLOv11s lebih stabil pada berbagai threshold. Perbedaan ini mengindikasikan bahwa arsitektur dan optimasi model berperan penting dalam meningkatkan akurasi deteksi, di mana YOLOv8s lebih efektif dalam mengekstraksi fitur dan mengenali pola dari dataset yang sama dibandingkan YOLOv11s. Oleh karena itu, dalam konteks implementasi sistem pengawasan kepatuhan berpakaian, YOLOv8s lebih direkomendasikan karena memiliki tingkat keandalan yang lebih tinggi dalam mendeteksi berbagai jenis pakaian secara real-time.

- 4) Sistem ini tidak hanya berfungsi untuk mendeteksi pelanggaran, tetapi juga sebagai kontrol akses ke ruangan administrasi. Jika seseorang terdeteksi mengenakan pakaian yang sesuai dengan peraturan, sistem akan secara otomatis memberikan izin masuk. Sebaliknya, jika terjadi pelanggaran, akses masuk tidak akan diberikan, dan individu tersebut harus mengganti pakaian sesuai dengan ketentuan yang berlaku sebelum dapat memasuki ruangan. Selain itu, sistem akan mengirimkan notifikasi kepada petugas atau staf administrasi untuk memantau dan memastikan kepatuhan terhadap aturan yang telah ditetapkan. Dengan integrasi ini, sistem tidak hanya meningkatkan disiplin berpakaian tetapi juga memudahkan petugas dalam melakukan pengawasan serta memastikan lingkungan kampus tetap tertib dan profesional.



## DAFTAR PUSTAKA

- [1] Firdaus, A. R., Kharisma, O. B., Ismaredah, E., Sains, A. F., Teknologi, D., Elektro, T., ... Pekanbaru, I. (2024). Deteksi Kode Etik Berpakaian pada Area Kampus Menggunakan YoloV8. *Journal of Information System Research (JOSH)*, 5(2), 450–458.
- [2] Jamaludin, F., Pambudi, A., Informatika, T., Sukabumi, U. M., Busana, D., Data, A., & Model, E. (2024). PENERAPAN YOLO ( YOU ONLY LOOK ONCE ) UNTUK DETEKSI ETIKA, 8(5), 10623–10629.
- [3] Ismail Arianto, B., & Zuliarso, E. (2023). Implementasi Metode YOLO pada Deteksi Pakaian Keselamatan yang Lengkap di Proyek Kontruksi. *Ranah Research%: Journal of Multidisciplinary Research and Development*, 6(1), 56–63. <https://doi.org/10.38035/rrj.v6i1.795>
- [4] Hayati, N. J., Singasatia, D., & Muttaqin, M. R. (2023). Object Tracking Menggunakan Algoritma You Only Look Once (YOLO)v8 untuk Menghitung Kendaraan. *Komputa%: Jurnal Ilmiah Komputer Dan Informatika*, 12(2), 91–99. <https://doi.org/10.34010/komputa.v12i2.10654>
- [5] [5]Mulyana, D. I., & Sahroni. (2024). Optimasi Penerapan Algoritma Yolo dan Data Augmentasi dalam Klasifikasi Pakaian Tradisional Kebaya. *Jurnal JTik (Jurnal Teknologi Informasi Dan Komunikasi)*, 8(1), 188–193. <https://doi.org/10.35870/jtik.v8i1.1446>.
- [6] Wulanningrum, R., Handayani, A. N., & Wibawa, A. P. (2024). Perbandingan Instance Segmentation Image Pada Yolo8. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(4), 753–760. <https://doi.org/10.25126/jtiik.1148288>
- [7] Hafid, M. A., & Kurniawan, M. A. (2024). Deteksi Akun Kaggle Bot Menggunakan Linear Regression, 6(2), 449–459. <https://doi.org/10.33650/jecom.v4i2>
- [8] Mira, S., Gudiato, C., & Chan, B. (2023). Computer Vision dan YOLO Menggali Potensi Computer Vision dan Implementasi YOLO untuk Pertanian Pintar. *Uwais Inspirasi Indonesia*.
- [9] B. M. Wantania, B., R.U.A. Sompie, S., & D. Kambey, F. (2020). Penerapan Pendeteksian Manusia Dan Objek Dalam Keranjang Belanja Pada Antrian Di Kasir. *Jurnal Teknik Informatika*, 15(2), 101–108.
- [10] Pressman, R. S., & Maxim, B. R. (2015). *Software engineering: A practitioner's approach* (Ninth ed.). McGraw-Hill.
- [11] B, A. A., Amin, A., & Kasrani, M. W. (2021). Penerapan Metode Yolo Object Detection V1 Terhadap Proses Pendeteksian Jenis Kendaraan Di Parkiran. *Jurnal Teknik Elektro Uniba (JTE UNIBA)*, 6(1), 194–199. <https://doi.org/10.36277/jteuniba.v6i1.130>
- [12] Sharma, A., Kumar, V., & Longchamps, L. (2024). Smart Agricultural Technology Faster R-CNN models for detection of multiple weed species, 9(October).