

## INTEGRASI ALGORITMA YOLOV8 DAN STREAMLIT UNTUK VISUALISASI REAL-TIME DAN AKURAT DALAM PENGHITUNGAN KERUMUNAN DI KAWASAN STASIUN BEKASI

<sup>1</sup>Prihandoko, <sup>2</sup>Sri Agustina Rumapea✉, <sup>1</sup>Abdul Hanif Pratama

<sup>1</sup>Departemen of Informatics, Gunadarma University, Depok, Indonesia

<sup>2</sup>Faculty of Computer Science, Universitas Methodist Indonesia, Medan, Indonesia

Email: [srirumapea78@gmail.com](mailto:srirumapea78@gmail.com)

DOI: <https://doi.org/10.46880/jmika.Vol9No1.pp179-187>

### ABSTRACT

Crowd management in public transportation areas has become a critical challenge with the rise of urban populations. This study develops a real-time web-based people detection and counting system by integrating the YOLOv8 algorithm with the Streamlit framework. A case study was conducted at the entrance of Bekasi Station. The model was developed using the AI Project Life Cycle approach, and the system was built following the Waterfall methodology. Data were obtained from video recordings, which were extracted into images, annotated, and processed into training and testing datasets. The YOLOv8 model was trained for 50 epochs, yielding strong performance with an mAP@0.5 of 91.7%, a maximum precision of 93.6%, and an F1-score of 87%. Tests on 15 images showed an average accuracy of 80.37% and an error rate of 19.63%. The model's performance declined on out-of-dataset images due to variations in lighting and extreme crowd density. The system was tested using black-box testing and demonstrated that all main features—image upload, object detection, visualization, and result download—functioned correctly. The system has been successfully deployed on Streamlit Cloud. These results indicate that the system offers a practical, lightweight, and responsive solution to support crowd monitoring in public areas. In future development phases, the system can be extended to support real-time video stream processing and integrated with an object tracking and classification module to accurately identify and differentiate the ingress and egress flow of individuals within a defined surveillance area.

**Keyword:** Crowd Counting, Object Detection, Public Transportation, Streamlit, YOLOv8.

### ABSTRAK

Manajemen kerumunan di area transportasi publik menjadi tantangan penting seiring meningkatnya populasi urban. Penelitian ini mengembangkan sistem deteksi dan penghitungan jumlah orang secara real-time berbasis web, dengan mengintegrasikan algoritma YOLOv8 dan framework Streamlit. Studi kasus dilakukan di pintu masuk Stasiun Bekasi. Model dikembangkan menggunakan pendekatan AI Project Life Cycle dan sistem dibangun dengan metode Waterfall. Data diperoleh dari rekaman video yang diekstrak menjadi gambar, kemudian dianotasi dan diproses menjadi dataset pelatihan dan pengujian. Model YOLOv8 dilatih selama 50 epoch, menghasilkan performa yang baik dengan mAP@0.5 sebesar 91,7%, precision maksimum 93,6%, dan F1-score 87%. Uji coba terhadap 15 gambar menunjukkan akurasi rata-rata 80,37% dan error 19,63%. Performa model menurun pada gambar luar-dataset akibat variasi pencahayaan dan kepadatan ekstrem. Sistem diuji melalui black-box testing dan menunjukkan bahwa semua fitur utama—unggah gambar, deteksi objek, visualisasi, dan unduhan hasil—berfungsi dengan baik. Sistem telah berhasil dideploy di Streamlit Cloud. Hasil ini menunjukkan bahwa sistem mampu memberikan solusi praktis, ringan, dan responsif untuk mendukung pengawasan kerumunan di area publik. Pada tahap selanjutnya, sistem dapat ditingkatkan untuk mendukung pemrosesan input berupa stream video secara langsung serta diintegrasikan dengan modul klasifikasi guna mengidentifikasi dan membedakan arus pergerakan individu yang masuk dan keluar dari suatu area pengamatan.

**Kata Kunci:** Crowd Counting, Deteksi Objek, Streamlit, Transportasi Publik, YOLOv8.

### PENDAHULUAN

Dalam beberapa dekade terakhir, lonjakan populasi urban dan percepatan urbanisasi telah membawa tantangan besar dalam pengelolaan

kerumunan, terutama di fasilitas umum seperti stasiun kereta api (Hassan et al., 2023; Khan et al., 2022). Kemampuan untuk memantau dan memperkirakan jumlah orang secara akurat di area-area padat menjadi

sangat penting, baik untuk menjaga keselamatan, meningkatkan kenyamanan pengguna, maupun mendukung efisiensi operasional (Lee & Marinov, 2022; Zhang et al., 2021).

Teknologi *computer vision* telah berkembang pesat, dengan algoritma deteksi objek seperti YOLO (You Only Look Once) yang memungkinkan sistem untuk melakukan pendeteksian objek secara cepat dan akurat dalam waktu nyata (Darmadi & Doni, 2024; Sohan et al., 2024). Versi terbarunya, YOLOv8, menawarkan peningkatan signifikan dalam hal akurasi deteksi, kecepatan inferensi, dan efisiensi penggunaan sumber daya komputasi (Gündüz & Işık, 2023; Shyaa & Hashim, 2024). Bersamaan dengan itu, kemunculan framework seperti Streamlit membuat proses pembuatan aplikasi berbasis machine learning menjadi lebih cepat, ringan, dan dapat diakses dengan mudah melalui web (Mhadhbi, 2021).

Namun demikian, sebagian besar studi sebelumnya masih berfokus pada skenario umum dan belum sepenuhnya mengadaptasi teknologi ini untuk kebutuhan spesifik area transportasi publik, yang diwarnai dengan tantangan seperti keragaman tingkat pencahayaan, kepadatan dinamis, dan kebutuhan real-time monitoring (Kusuma et al., 2021; Zhu et al., 2024). Pada area seperti Stasiun Bekasi, yang melayani ribuan penumpang setiap harinya, diperlukan pendekatan yang lebih terfokus dan adaptif.

Penelitian ini bertujuan mengembangkan sistem *crowd counting* berbasis web yang mampu mendeteksi dan memprediksi jumlah orang secara real-time di area pintu masuk Stasiun Bekasi. Sistem ini mengintegrasikan model YOLOv8 dengan *framework* Streamlit, diharapkan dapat mendukung pengelolaan kerumunan yang lebih efektif dan meningkatkan keselamatan serta kenyamanan pengguna transportasi publik. Metode AI Project Life Cycle digunakan untuk membangun model deteksi, sedangkan pengembangan sistem mengikuti pendekatan Waterfall agar pengembangan berjalan sistematis dan terstruktur.

## KAJIAN LITERATUR

### *Crowd Counting* dan Tantangan Manajemennya

*Crowd counting*, atau penghitungan jumlah orang dalam sebuah area, menjadi salah satu kebutuhan mendesak di era urbanisasi pesat. Pengelolaan kerumunan secara efektif dibutuhkan untuk meningkatkan keselamatan publik, mengoptimalkan operasional, serta memitigasi risiko pada area dengan mobilitas tinggi seperti stasiun kereta (Hassan et al., 2023; Khan et al., 2022). Studi dari Lee & Marinov (2022) menunjukkan bahwa lonjakan volume

penumpang yang tidak terkelola dapat meningkatkan risiko kecelakaan dan memperburuk pengalaman pengguna transportasi.

Berbagai pendekatan telah dikembangkan untuk mengatasi masalah ini, mulai dari metode manual hingga solusi berbasis teknologi, seperti sensor thermal, kamera CCTV, hingga penerapan algoritma berbasis kecerdasan buatan.

### Teknologi *Computer Vision* dalam *Crowd Counting*

*Computer vision* menjadi pendekatan dominan dalam penghitungan kerumunan dalam beberapa tahun terakhir. Teknologi ini memungkinkan sistem untuk secara otomatis mendeteksi, mengenali, dan menghitung objek manusia dari citra maupun video. Salah satu teknik yang paling banyak digunakan adalah deteksi objek berbasis *Convolutional Neural Networks* (CNN), di mana model dilatih untuk mengenali fitur-fitur visual spesifik dari manusia (Zhang et al., 2021).

Metode deteksi objek berbasis CNN terbukti mampu meningkatkan akurasi dibandingkan pendekatan tradisional. Namun, tantangan tetap ada, terutama dalam menghadapi perubahan pencahayaan, tumpang tindih antarindividu, dan kepadatan tinggi yang sering terjadi di area public (Zhu et al., 2024).

### Algoritma YOLO dan Perkembangannya

YOLO (*You Only Look Once*) adalah algoritma deteksi objek berbasis CNN yang dirancang untuk memproses gambar secara *end-to-end* dalam satu tahap prediksi, sehingga menghasilkan kecepatan dan efisiensi yang tinggi (Redmon et al., 2016). Sejak pertama kali diperkenalkan, YOLO terus dikembangkan dengan berbagai versi perbaikan.

YOLOv8, sebagai versi terbaru yang dikembangkan oleh Ultralytics, membawa beberapa inovasi signifikan, antara lain deteksi anchor-free, arsitektur model yang lebih ringan, serta peningkatan dalam akurasi dan kecepatan inferensi (Shyaa & Hashim, 2024; Sohan et al., 2024). Penelitian oleh Gündüz & Işık (2023) membuktikan bahwa YOLOv8 memiliki performa lebih baik dibandingkan YOLOv5 dalam skenario *real-time crowd detection*.

Studi lain oleh Shyaa & Hashim (2024) juga menegaskan bahwa YOLOv8 efektif dalam menangani tantangan deteksi objek pada situasi keramaian padat, berkat mekanisme *feature extraction* yang lebih adaptif terhadap berbagai kondisi lingkungan.

### Framework Streamlit untuk Implementasi Web-Based System

Dalam implementasi sistem *crowd counting* berbasis web, *framework* Streamlit menjadi pilihan yang populer karena kemudahannya dalam membangun aplikasi machine learning secara cepat, ringan, dan interaktif (Mhadhbi, 2021). Dengan menggunakan Python sebagai bahasa utama, Streamlit mendukung integrasi berbagai model *machine learning* tanpa memerlukan keahlian *front-end development* yang kompleks.

Dalam konteks *crowd counting*, penggunaan Streamlit memungkinkan visualisasi hasil deteksi secara *real-time* di *web browser*, mempercepat proses *deployment*, dan memperluas aksesibilitas sistem untuk pengguna di lapangan.

### Kesenjangan Penelitian Terdahulu

Beberapa penelitian sebelumnya telah memanfaatkan algoritma YOLO untuk *crowd counting*, namun masih terdapat keterbatasan dalam penerapannya pada lingkungan nyata seperti stasiun kereta api. Kusuma et al. (2021) menggunakan YOLOv4 untuk menghitung jumlah orang di transportasi publik dengan akurasi 69%, namun tidak mendukung visualisasi berbasis web dan tidak mempertimbangkan dinamika kepadatan tinggi. Penelitian lain oleh Zhu et al. (2024) menggunakan kombinasi YOLO dan Deep SORT untuk pelacakan kerumunan, tetapi fokusnya lebih pada pelacakan individu dan belum spesifik terhadap kebutuhan transportasi publik. Oleh karena itu, penelitian ini mencoba mengisi *gap* tersebut dengan mengembangkan sistem *crowd counting* berbasis YOLOv8 yang diintegrasikan ke dalam platform web menggunakan Streamlit, dengan fokus utama pada skenario nyata di area pintu masuk Stasiun Bekasi.

### METODE PENELITIAN

Penelitian ini memadukan dua pendekatan metodologis utama: AI Project Life Cycle untuk pengembangan model kecerdasan buatan dan metode Waterfall untuk pengembangan sistem berbasis *web*. Pendekatan ini dipilih untuk memastikan keterpaduan antara model deteksi yang akurat dan sistem yang praktis serta aplikatif.

Pada tahap *problem scoping*, masalah utama yang diidentifikasi adalah kesulitan mendeteksi dan menghitung jumlah orang secara akurat di area padat seperti Stasiun Bekasi, khususnya pada jam sibuk. Tujuan penelitian ditetapkan untuk mengembangkan sistem otomatis berbasis gambar yang mampu mendeteksi jumlah orang secara *real-time* guna

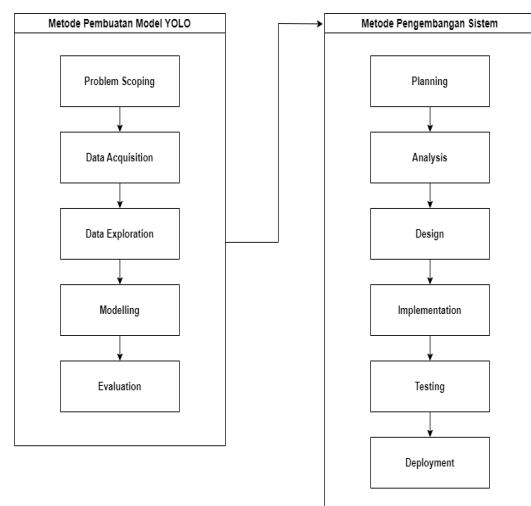
mendukung manajemen kerumunan dan peningkatan keamanan.

Tahap data *acquisition* melibatkan pengumpulan data melalui rekaman video menggunakan kamera *smartphone*, yang kemudian diekstrak menjadi *dataset* gambar. Data di anotasi menggunakan platform *Roboflow* dan dibagi menjadi tiga *subset*: data latih, validasi, dan uji. Proses data *exploration* memastikan kualitas, distribusi, dan keragaman data sebelum tahap *modelling*.

Pada tahap *modelling*, model YOLOv8 dilatih menggunakan Google Colab dengan pemanfaatan sumber daya komputasi awan. Evaluasi model dilakukan secara berkala menggunakan metrik kinerja seperti akurasi, *precision*, *recall*, dan F1-score. Jika performa belum optimal, dilakukan *tuning* parameter dan pelatihan ulang.

Tahap *deployment* mencakup integrasi model ke dalam sistem berbasis web yang dikembangkan menggunakan Visual Studio Code dan Streamlit. Sistem memungkinkan pengguna mengunggah gambar untuk deteksi jumlah orang secara otomatis, dengan hasil yang di visualisasikan secara *real-time*.

Sementara itu, pengembangan sistem mengikuti pendekatan Waterfall, dimulai dari analisis kebutuhan, perancangan, implementasi, pengujian, hingga pemeliharaan. Rangkaian metodologi ini secara keseluruhan dirancang untuk menghasilkan solusi yang tidak hanya kuat secara teknis, tetapi juga relevan dan bermanfaat secara praktis di lingkungan nyata, khususnya dalam mendukung pengelolaan kepadatan kerumunan di area transportasi publik. Visualisasi alur metodologi penelitian ini dapat dilihat pada Gambar 1. Diagram Kerangka Metodologi Penelitian yang disajikan berikut.



Gambar 1. Diagram Kerangka Metodologi Penelitian

## HASIL DAN PEMBAHASAN

### Hasil Penelitian

Penelitian ini dilakukan dengan mengimplemen tasikan algoritma YOLOv8 pada sistem berbasis web menggunakan framework Streamlit untuk mendeteksi dan menghitung jumlah orang di area pintu Stasiun Bekasi. Penelitian ini mencakup tahap pengumpulan data (*data acquisition*), eksplorasi data (*data exploration*), pemodelan (*modelling*), evaluasi model, hingga pengembangan sistem.

### Hasil Data Acquisition

Data yang digunakan dalam penelitian ini diperoleh melalui rekaman video dengan durasi sekitar 4 menit 50 detik, menggunakan kamera *smartphone*. Rekaman video dilakukan pada jam-jam sibuk (06.00-08.00 WIB dan 17.00-18.00 WIB) serta jam non-sibuk (12.00-13.00 WIB) untuk mendapatkan variabilitas kondisi yang tinggi. Hasil ekstraksi dari rekaman video menghasilkan total 528 *frame* gambar yang digunakan pada tahap selanjutnya (dapat dilihat pada Gambar 2 Hasil Data Acquisition Dalam Bentuk *Frame*).

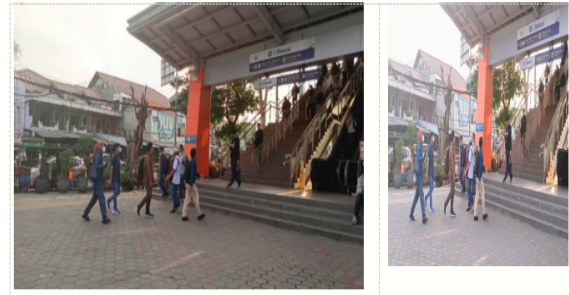


Gambar 2. Hasil Data Acquisition Dalam Bentuk *Frame*

### Hasil Data Exploration

Proses eksplorasi data melibatkan tahap *preprocessing* yang terdiri dari auto-orient, static crop, dan resize ke ukuran standar 640x640 piksel. Tahap ini menghasilkan 505 *frame* valid. Selanjutnya, augmentasi data diterapkan melalui teknik flip horizontal, *brightness adjustment*, dan *exposure adjustment*. Jumlah gambar untuk training set meningkat menjadi 1060 gambar, validation set sebanyak 101 gambar, dan testing set sebanyak 50 gambar, memperkaya dataset pelatihan secara signifikan (dapat dilihat pada Gambar 3 Sampel Hasil Preprocessing Data: Resize dan Gambar 4 Sampel Hasil Augmentasi Data: Flip Horizontal). Gambar 4 secara khusus menggambarkan contoh hasil

augmentasi data menggunakan teknik flip horizontal, di mana gambar asli dibalik secara horizontal untuk menciptakan variasi orientasi objek. Teknik ini berguna untuk meningkatkan robustnes model terhadap variasi orientasi objek yang mungkin ditemui di lingkungan nyata.



Gambar 3. Sampel Hasil *Preprocessing* Data: *Resize*



Gambar 4. Sampel Hasil Augmentasi Data: *Flip Horizontal*

### Hasil Pemodelan

Pemodelan menggunakan algoritma YOLOv8s dengan 50 *epoch* pelatihan yang dilakukan secara intensif. Dari pelatihan model tersebut, model terbaik yang diperoleh memiliki ukuran file 22.5MB setelah proses *stripping optimizer*. Hasil ini menunjukkan bahwa model memiliki kapasitas komputasi yang optimal dengan akurasi prediksi yang tinggi (dapat dilihat pada Gambar 5 Hasil Train Model).

```
50 epochs completed in 0.436 hours.
Optimizer stripped from runs/detect/train/weights/last.pt, 22.5MB
Optimizer stripped from runs/detect/train/weights/best.pt, 22.5MB

Validating runs/detect/train/weights/best.pt...
Ultralytics YOLOv8 8.1.96 Python-3.10.12 torch-2.3.1+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 11125971 parameters, 0 gradients, 28.4 GFLOPs

Class Images Instances Box(P R mAP50 mAP50-95) 100% 4/4 [00:05:00:00, 1.39s/it]
all 101 1457 0.9 0.844 0.917 0.707

Speed: 0.4ms preprocess, 6.6ms inference, 0.0ms loss, 7.7ms postprocess per image
Results saved to runs/detect/train
Learn more at https://docs.ultralytics.com/modes/train
```

Gambar 5. Hasil Train Model

### Evaluasi Kinerja Model Deteksi Objek Person

Evaluasi kinerja model dilakukan guna menilai efektivitas model dalam mendeteksi objek dari kelas

person. Evaluasi ini bertujuan untuk memastikan bahwa model yang telah dilatih tidak hanya bekerja optimal pada data pelatihan, tetapi juga memiliki kemampuan generalisasi yang baik terhadap data baru. Beberapa metrik evaluasi yang digunakan meliputi Confusion Matrix, Precision, Recall, F1-Score, serta Mean Average Precision (mAP). Penggunaan metrik-metrik ini memungkinkan analisis performa model secara komprehensif, sehingga potensi kekuatan dan keterbatasan model dapat diidentifikasi secara objektif.

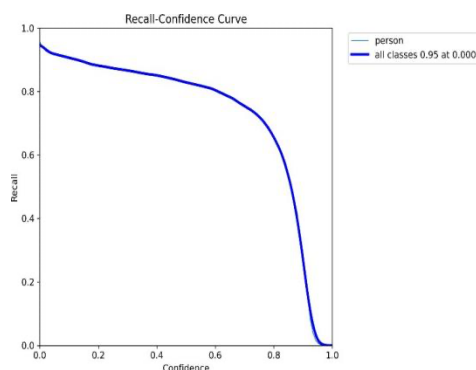
### Confusion Matrix

Hasil *confusion matrix* menggambarkan kemampuan model dalam mengklasifikasikan kelas person secara benar dan salah. Berdasarkan hasil evaluasi, model mencatat:

- *True Positive* (TP) sebanyak 1.282, menunjukkan jumlah objek person yang berhasil terdeteksi secara akurat.
- *False Positive* (FP) sebesar 237, yaitu jumlah prediksi salah di mana objek background diklasifikasikan sebagai person.
- *False Negative* (FN) sebanyak 175, yaitu jumlah objek person yang tidak berhasil terdeteksi dan diklasifikasikan sebagai background.
- *True Negative* (TN) tidak tersedia secara eksplisit mengingat model hanya mengevaluasi satu kelas utama.

### Recall

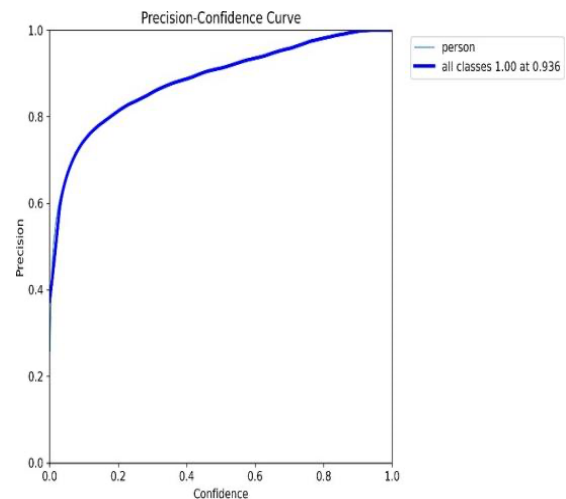
Recall mengukur kemampuan model dalam mendeteksi semua instansi dari kelas person. Berdasarkan Gambar 6, nilai *recall* mencapai puncaknya sebesar 0.95 pada saat *confidence threshold* berada pada nilai rendah. Namun, seiring peningkatan nilai *confidence*, terjadi penurunan nilai *recall*, yang mengindikasikan bahwa model menjadi lebih selektif tetapi juga cenderung melewatkan beberapa objek yang relevan.



Gambar 6. Kurva nilai *Recall* terhadap nilai *Confidence*

### Precision

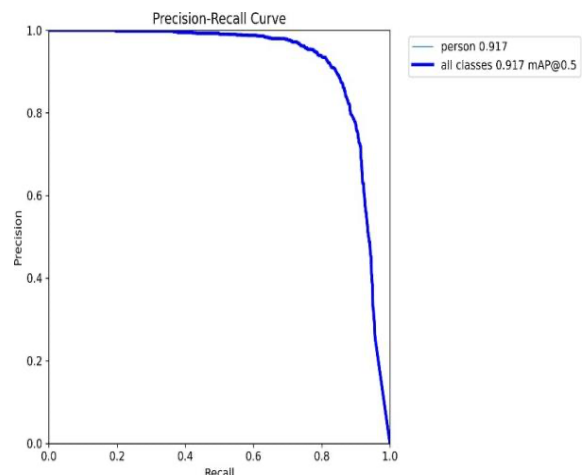
*Precision* menunjukkan proporsi prediksi *person* yang benar terhadap keseluruhan prediksi *person* yang dilakukan model. Dari Gambar 7, diperoleh nilai *precision* tertinggi saat *confidence score* mencapai 0.936. Hal ini menunjukkan bahwa pada ambang kepercayaan tersebut, model memiliki tingkat ketelitian yang optimal dalam mendeteksi objek dari kelas person.



Gambar 7. Kurva nilai *Precision* terhadap nilai *Confidenc*

### Precision-Recall Curve

Kurva *Precision-Recall* (Gambar 8) memberikan informasi tentang keseimbangan antara kedua metrik tersebut dalam berbagai tingkat ambang batas. Berdasarkan grafik, diketahui bahwa model mencapai nilai maksimum *mAP@0.5* sebesar 0.917. Hal ini menunjukkan bahwa model memiliki akurasi tinggi dalam mendeteksi objek dengan tingkat kesesuaian spasial (IoU) minimal 50%.

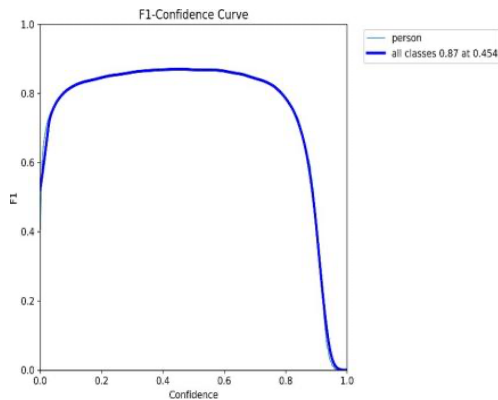


Gambar 8. Kurva nilai *precision* terhadap nilai *recall*



### F1-Score

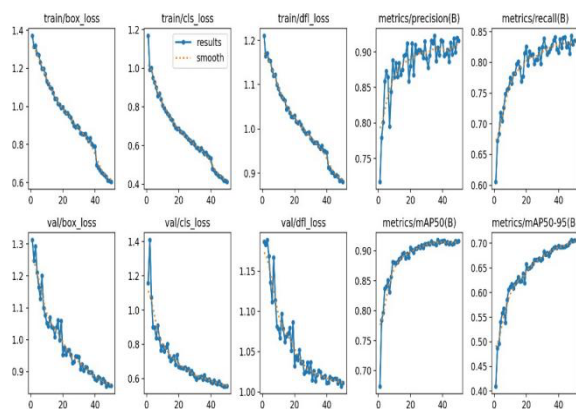
F1-Score merupakan metrik harmonisasi antara *precision* dan *recall*. Gambar 9 menunjukkan bahwa model mencapai nilai F1 tertinggi sebesar 0.87 pada *confidence score* 0.454. Nilai ini mencerminkan kinerja yang seimbang antara sensitivitas dan ketelitian model dalam melakukan deteksi objek.



Gambar 9. Kurva nilai F1 terhadap nilai *confidence*

### Mean Average Precision (mAP)

Mean Average Precision (mAP) digunakan sebagai metrik utama untuk menilai performa keseluruhan model dalam deteksi objek. Berdasarkan Gambar 10, diperoleh bahwa nilai mAP@0.5 melebihi 90%, sedangkan mAP@[.5:.95] juga menunjukkan tren peningkatan selama proses pelatihan. Selain itu, tren penurunan pada metrik train/box\_loss, train/cls\_loss, train/df\_l\_loss, serta val/box\_loss, val/cls\_loss, dan val/df\_l\_loss menunjukkan bahwa model mampu belajar secara efektif dan menjaga performanya pada data validasi.



Gambar 10. Hasil mAP (Mean Average Precision)  
Metric

### Validasi Model

Validasi dilakukan menggunakan *validation set* yang tidak dilibatkan dalam proses pelatihan. Proses validasi bertujuan untuk memastikan bahwa model

tidak mengalami *overfitting* dan mampu mengeneralisasi pada data yang belum pernah dilihat. Berdasarkan hasil pelaksanaan skrip validasi model *best.pt* diuji dengan ambang kepercayaan sebesar 25% pada citra uji.

Hasil validasi menunjukkan bahwa model dapat mempertahankan performa yang baik pada data uji, serta memberikan deteksi yang konsisten terhadap objek *person*.

### Hasil Uji Coba Model dan Sistem

#### Uji Coba Model

Model diuji menggunakan 15 gambar dari berbagai kondisi kepadatan pengunjung di area pintu Stasiun Bekasi. Hasil pengujian menunjukkan rata-rata persentase *error* sebesar 19,63%, dengan tingkat akurasi prediksi mencapai 80,37%. Pengujian pada gambar-gambar yang tidak termasuk dalam dataset pelatihan menunjukkan peningkatan persentase error, menegaskan perlunya dataset yang lebih beragam untuk meningkatkan generalisasi model.

### Hasil Pengembangan Sistem

Setelah diperoleh model deteksi terbaik melalui proses pelatihan dan evaluasi, tahap selanjutnya adalah pengembangan sistem berbasis model tersebut untuk mendeteksi dan menghitung jumlah orang secara otomatis. Pengembangan sistem dijelaskan dalam tahapan-tahapan berikut ini.

### Hasil Implementation

Pengembangan sistem terdiri dari dua tahap implementasi utama. Pertama, implementasi antarmuka sistem dilakukan menggunakan framework Streamlit untuk menghasilkan tampilan yang sederhana dan intuitif. Pengguna dapat mengunggah gambar, menjalankan proses deteksi, dan melihat hasil perhitungan secara langsung. Komponen antarmuka mencakup judul aplikasi, deskripsi singkat, serta sidebar untuk navigasi dan input gambar.

Kedua, implementasi penghitungan jumlah orang menggunakan model YOLOv8. Model ini dimuat dari file *best.pt* dan dijalankan pada gambar yang diunggah. Proses prediksi menghasilkan bounding box untuk setiap objek yang terdeteksi sebagai orang, lalu dihitung menggunakan fungsi *len(boxes)*. Hasil jumlah orang yang terdeteksi dianotasi ke dalam gambar menggunakan OpenCV dan ditampilkan di antarmuka.

### Hasil Testing

Pengujian sistem terbagi menjadi dua bagian utama. Pertama, uji coba model dilakukan terhadap 15 gambar uji, dengan membandingkan hasil prediksi model dan perhitungan manual (ground truth). Hasil menunjukkan rata-rata persentase error sebesar 19,63% dan tingkat akurasi sebesar 80,37%. Model menunjukkan kinerja yang baik untuk gambar yang berasal dari dataset pelatihan, namun cenderung menurun pada gambar dari luar dataset (out-of-sample) karena variasi sudut pandang, pencahayaan, dan resolusi.

Kedua, uji coba sistem menggunakan metode *Blackbox Testing* untuk mengevaluasi fungsionalitas sistem. Lima skenario diuji, termasuk pemuatan model, unggah gambar, deteksi objek, tampilan hasil, dan unduhan hasil deteksi. Semua fungsi sistem berjalan sesuai ekspektasi, menunjukkan stabilitas dan kesiapan sistem untuk digunakan dapat dilihat pada tabel dibawah ini.

Tabel 1. Hasil Balck Box Testing

No	Skenario Pengujian	Langkah Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Memuat model	Sistem memuat model deteksi dari file best.pt	Model berhasil dimuat tanpa error	Sesuai
2	Mengunggah gambar	Pengguna mengunggah file gambar ke sistem	Gambar berhasil diunggah dan ditampilkan	Sesuai
3	Deteksi objek dan prediksi jumlah orang	Sistem mendeteksi objek dan memprediksi jumlah orang pada gambar yang diunggah	Jumlah orang terdeteksi sesuai dengan gambar yang diunggah	Sesuai
4	Menampilkan hasil deteksi	Sistem menampilkan hasil deteksi dalam bentuk gambar yang dianotasi	Gambar dengan anotasi ditampilkan di antarmuka	Sesuai
5	Mengunduh	Pengguna mengunduh	File gambar berhasil	Sesuai

### Hasil Deployment

Tahap akhir adalah deployment sistem menggunakan *platform* Streamlit *cloud*. Proses ini diawali dengan pembuatan repository publik di GitHub yang memuat seluruh kode sumber. Setelah itu, integrasi dengan Streamlit Cloud dilakukan melalui akun GitHub untuk mempermudah pemilihan *repository*. Formulir "Deploy an app" diisi dengan informasi terkait repository, branch, dan path file utama.

### Pembahasan

Dalam penelitian ini, implementasi algoritma YOLOv8 pada sistem *crowd counting* berbasis web terbukti efektif dalam mendeteksi dan menghitung jumlah orang di area publik, dengan akurasi yang tinggi pada kondisi gambar yang serupa dengan *dataset* pelatihan. Namun, tantangan utama yang muncul adalah pada kondisi gambar yang bervariasi seperti pencahayaan yang kurang optimal, resolusi rendah, dan objek tumpang tindih, yang menyebabkan penurunan akurasi. Oleh karena itu, penelitian lebih lanjut diperlukan untuk memperluas dan memperkaya *dataset* guna meningkatkan kemampuan generalisasi model.

Penggunaan framework Streamlit telah terbukti mampu menghasilkan antarmuka sistem yang responsif dan mudah digunakan oleh pengguna, sehingga mendukung penerapan sistem ini secara praktis dalam lingkungan nyata. Fitur yang interaktif dan responsif tersebut memungkinkan pengguna untuk melakukan pemantauan secara *real-time* dengan hasil yang dapat diandalkan.

### Rekomendasi Pengembangan

Penelitian ini mengusulkan beberapa rekomendasi penting untuk pengembangan sistem di masa depan, antara lain:

1. Menambah kemampuan sistem untuk mengklasifikasikan jumlah pengunjung masuk dan keluar secara terpisah di area pintu stasiun.
2. Mengembangkan sistem yang dapat memproses media *input* berupa video secara *real-time* untuk pemantauan langsung.
3. Menggunakan kamera dengan resolusi lebih tinggi dan penempatan strategis untuk meningkatkan akurasi deteksi.
4. Memperluas dataset pelatihan dengan variasi kondisi lingkungan yang lebih banyak untuk meningkatkan *robustnes* model.

Implementasi rekomendasi ini diharapkan akan meningkatkan kontribusi sistem dalam manajemen kerumunan secara signifikan, khususnya di area-area publik dengan intensitas pengunjung yang tinggi.

### KESIMPULAN

Penelitian ini berhasil mengembangkan sebuah sistem *crowd counting* berbasis web dengan menggunakan model YOLOv8 yang diintegrasikan ke dalam framework Streamlit. Sistem ini dirancang untuk mendeteksi dan menghitung jumlah orang secara *real-time* di area pintu masuk Stasiun Bekasi. Hasil implementasi menunjukkan bahwa sistem mampu menampilkan visualisasi deteksi secara langsung, yang

sangat bermanfaat untuk mendukung pengelolaan kerumunan di lokasi yang padat seperti stasiun kereta api. Dari hasil pengujian, model YOLOv8 menunjukkan performa yang cukup baik dengan tingkat akurasi sebesar 80,37%, dan rata-rata kesalahan sebesar 19,63% dalam mendeteksi serta menghitung jumlah orang pada gambar. Meski demikian, kinerja model masih dipengaruhi oleh beberapa kendala, seperti:

1. Variasi kondisi pencahayaan, sudut pandang kamera, dan tingkat kepadatan kerumunan yang tidak sepenuhnya tercakup dalam data pelatihan.
2. Jumlah dan kualitas data pelatihan yang terbatas, sehingga kemampuan model dalam mengenali kondisi baru masih belum optimal.
3. Kesulitan dalam mendeteksi objek manusia yang berada terlalu jauh atau saling bertumpukan, terutama jika resolusi gambar rendah atau bounding box terlalu kecil.

Melalui pengujian *black-box*, seluruh fungsi utama dalam sistem—mulai dari pemuatan model, pengunggahan gambar, hingga proses deteksi dan perhitungan objek—berhasil dijalankan dengan baik dan sesuai ekspektasi. Secara keseluruhan, sistem ini memiliki potensi sebagai alat bantu dalam pengawasan kerumunan, khususnya untuk meningkatkan keselamatan dan kenyamanan pengguna transportasi publik.

Lebih jauh, penelitian ini juga menjawab beberapa kekurangan dari studi sebelumnya, terutama pada aspek visualisasi berbasis web dan responsivitas sistem dalam kondisi kepadatan tinggi. Kontribusi ini diharapkan menjadi langkah awal menuju sistem pemantauan kerumunan yang lebih adaptif, efisien, dan mudah diakses.

## SARAN

Berdasarkan hasil penelitian dan keterbatasan yang telah diidentifikasi, terdapat beberapa arah pengembangan yang disarankan untuk penelitian selanjutnya. Pertama, sistem yang dikembangkan saat ini belum mampu membedakan jumlah orang yang masuk dan keluar di area pintu stasiun. Oleh karena itu, perlu dikembangkan mekanisme penghitungan dua arah untuk analisis arus pergerakan yang lebih akurat. Kedua, sistem masih terbatas pada input berupa citra (gambar), sehingga pengembangan sistem yang mendukung pemrosesan video atau *streaming* secara langsung diperlukan untuk memungkinkan pemantauan kerumunan secara real-time. Ketiga, hasil deteksi yang belum optimal dalam kondisi pencahayaan rendah, sudut pandang ekstrem, dan

kepadatan tinggi menunjukkan perlunya pelatihan model dengan data yang lebih beragam agar kemampuan generalisasi dapat ditingkatkan. Terakhir, optimalisasi parameter model serta integrasi teknik pelacakan objek (*object tracking*) direkomendasikan guna meningkatkan akurasi dan ketahanan sistem dalam menghadapi tumpang tindih objek dan kondisi kepadatan tinggi. Perbaikan-perbaikan ini diharapkan dapat mendukung pengembangan sistem *crowd counting* yang lebih andal, akurat, dan aplikatif di lingkungan nyata.

## DAFTAR PUSTAKA

- Darmadi, & Doni, H. N. (2024). Traffic Counting using YOLO Version-5 (A case study of Jakarta-Cikampek Toll Road). *IOP Conference Series: Earth and Environmental Science*, 1321(1), 012015. <https://doi.org/10.1088/1755-1315/1321/1/012015>
- Gündüz, M. Ş., & Işık, G. (2023). A new YOLO-based method for real-time crowd detection from video and performance analysis of YOLO models. *Journal of Real-Time Image Processing*, 20(1), 5. <https://doi.org/10.1007/s11554-023-01276-w>
- Hassan, M., Hussain, F., Khan, S. D., Ullah, M., Yamin, M., & Ullah, H. (2023). Crowd counting using deep learning-based head detection. *Electronic Imaging*, 35(9), 293--1-293--296. <https://doi.org/10.2352/EI.2023.35.9.IPAS-293>
- Khan, M. A., Menouar, H., & Hamila, R. (2022). *Revisiting Crowd Counting: State-of-the-art, Trends, and Future Perspectives*.
- Kusuma, T. A. A. H., Usman, K., & Saidah, S. (2021). People Counting for Public Transportations Using You Only Look Once Method. *Jurnal Teknik Informatika (Jutif)*, 2(1), 57--66. <https://doi.org/10.20884/1.jutif.2021.2.2.77>
- Lee, J., & Marinov, M. (2022). Analysis of Rail Passenger Flow in a Rail Station Concourse Prior to and During the COVID-19 Pandemic Using Event-Based Simulation Models and Scenarios. *Urban Rail Transit*, 8(2), 99--120. <https://doi.org/10.1007/s40864-022-00167-w>
- Mhadhbi, S. (2021). *Streamlit for machine learning and data science projects. Streamlit Documentation*.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779--788. <https://doi.org/10.1109/CVPR.2016.91>
- Shyaa, T. A. R., & Hashim, A. A. (2024). Enhancing real human detection and people counting using YOLOv8. *BIO Web of Conferences*, 97, 00061. <https://doi.org/10.1051/bioconf/20249700061>



- Sohan, M., Sai Ram, T., & Rami Reddy, Ch. V.  
(2024). *A Review on YOLOv8 and Its Advancements* (pp. 529–545).  
[https://doi.org/10.1007/978-981-99-7962-2\\_39](https://doi.org/10.1007/978-981-99-7962-2_39)
- Zhang, J., Liu, J., & Wang, Z. (2021). Convolutional Neural Network for Crowd Counting on Metro Platforms. *Symmetry*, 13(4), 703.  
<https://doi.org/10.3390/sym13040703>
- Zhu, Y., Ni, K., Li, X., Zaman, A., Liu, X., & Bai, Y.  
(2024). Artificial Intelligence Aided Crowd Analytics in Rail Transit Station. *Transportation Research Record: Journal of the Transportation Research Board*, 2678(2), 481–492.  
<https://doi.org/10.1177/03611981231175156>