

Deteksi Pelanggaran Sepeda Motor Menggunakan Algoritma Yolo Dan Mean Average Precision

Fauzi Taufik Hidayat¹, Adithya Kusuma Whardana²

Teknik Informatika
Fakultas Teknik dan
Teknologi

Universitas Tanri Abeng

fauzi.taufik@student.tau.ac.id¹, adithya@tau.ac.id²

Diterima : 30 Agustus 2024

Disetujui : 30 September 2024

Abstract— Karena keterbatasan Polisi Lalu Lintas dalam melakukan sosialisasi di jalan raya untuk meminimalkan jumlah pelanggaran, pelanggaran kendaraan sepeda motor kerap terjadi di jalan raya di antara mereka yang tidak memakai helm. Teknologi baru di bidang kecerdasan buatan mengajarkan komputer untuk menginterpretasikan dan menguasai gambar. Untuk melakukan deteksi, penelitian ini menggunakan tiga objek: helm, sepeda motor, dan tanpa helm. Algoritma You Only Look Once (YOLO) versi 5s dan 8s juga digunakan sebagai perbandingan, menggunakan dataset internet dan realtime yang berbeda. Selain itu, framework streamlit digunakan untuk melakukan implementasi berbasis website dan framework deepsort digunakan untuk menghitung jumlah pelanggaran tanpa helm. Hasil penelitian menunjukkan bahwa dataset internet YOLOv5 memiliki akurasi sebesar 94%, dengan dataset internet YOLOv5, YOLOv8, dan realtime masing-masing memiliki akurasi sebesar 93%. Dataset internet YOLOv5 unggul dibandingkan dengan dataset internet lainnya.

Kata kunci — Pelanggaran Sepeda Motor, YOLO V5s & V8s, mean Average Precision (mAP).

I. PENDAHULUAN

Sebanyak 4.729 kecelakaan lalu lintas terjadi di Provinsi DKI Jakarta selama tahun 2020, menurut data dari Badan Pusat Statistik DKI Jakarta dan Direktorat Lalu Lintas Polda Metro Jaya. Jumlah korban tewas yang tinggi disebabkan oleh kelalaian pengendara yang melakukan pelanggaran, salah satunya dari mereka tidak menggunakan helm atau pelindung kepala [8].

Dengan kemajuan teknologi dalam bidang kecerdasan buatan atau kecerdasan artistik, computer vision digunakan untuk menginterpretasikan dan menguasai objek secara visual, menggunakan gambar digital sebagai input dan jaringan neural untuk proses mengidentifikasi

dan mengklasifikasikan objek, kemudian mendeteksi apa yang dilihat komputer [2]. Penelitian yang dilakukan pada tahun 2018 oleh Yogameena, Menaka, Saravana. Kerangka kerja pada penelitian ini terdiri dari segmentasi latar depan melalui *Generalized Method of Moments* (GMM), deteksi sepeda motor dengan dan tanpa helm menggunakan Fast R-CNN, hasil penelitian menyatakan bahwa Fast R-CNN memberikan akurasi yang lebih tinggi [12].

Monish, Khan Umar, Manoj, Hemanth, Girish, dan Gurunshankar melakukan penelitian tambahan pada tahun 2021. Dalam penelitian ini, model atau algoritma Fast RCNN dapat menemukan helm dalam gambar dengan akurasi hingga 90% [10]. Penelitian berikutnya, yang

dilakukan oleh Muhammad Danang Adhiwijaya pada tahun 2021, menggunakan YOLO v3 sebagai algoritma untuk menemukan objek dalam gambar [10]. Hasil studi menunjukkan akurasi sebesar 94% [2].

Penelitian ini menggunakan algoritma YOLO Versi 5s dan 8s untuk deteksi objek, yang membandingkan data antara data real-time dan data internet. Diharapkan penelitian ini akan membantu mendeteksi pelanggaran pengendara sepeda motor.

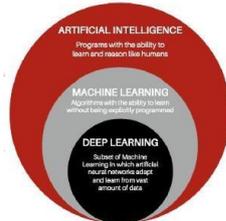
II. DASAR TEORI PENUNJANG

A. Pengolahan Citra

Pengelolaan citra merupakan proses untuk meningkatkan kualitas gambar sehingga lebih mudah dipahami oleh manusia atau komputer serta mengubahnya menjadi format yang berbeda. Salah satu contohnya adalah proses kompresi gambar, yang bertujuan untuk mengurangi jumlah bit yang digunakan untuk menampilkan gambar, sehingga ukuran data gambar dapat diperkecil. Proses ini merupakan tahap awal, atau preprocessing, dari pengolahan gambar untuk tujuan penggunaan komputer visi [6].

B. Artificial Intelligence (AI)

Kemampuan mesin atau robot yang dioperasikan oleh komputer untuk melakukan tugas-tugas yang biasanya dilakukan oleh makhluk cerdas disebut kecerdasan buatan (AI). Tujuan pengembangan sistem AI adalah untuk membuat mesin yang dapat mempelajari dan meniru sifat proses intelektual manusia, seperti kemampuan untuk menalar, menafsirkan makna, membuat generalisasi, dan belajar dari pengalaman sebelumnya [4].



Gambar 2 1 Korelasi antara AI, Machine Learning, Deep Learning

C. Machine Learning

Machine learning merupakan salah satu fungsi AI yang bertujuan untuk meningkatkan kinerja sistem dengan menganalisis data yang telah disimpan atau terekam sebelumnya dan membuat prediksi berdasarkan data tersebut. Ini

memungkinkan manusia untuk tidak perlu mengidentifikasi proses secara lengkap karena machine learning memungkinkan komputer untuk membuat pola dan membuat keputusan berdasarkan data yang telah dianalisis.

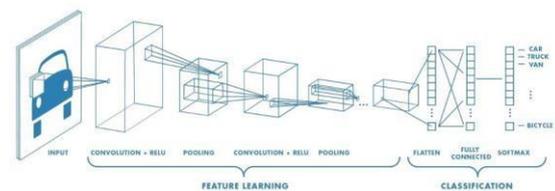
D. Deep Learning

Jenis model jaringan saraf tiruan yang sangat disukai adalah deep learning, yang berkembang pesat dalam pengenalan citra berkat dukungan teknologi Graphics Processing Unit (GPU). Pembelajaran mendalam terdiri dari dua tahap: pelatihan dan pengujian. Pada tahap pelatihan, model akan belajar mengekstrak fitur dari data dan membedakan label yang berbeda. Pada tahap pengujian, hasil pembelajaran sebelumnya akan digunakan untuk menganalisis data yang diuji. Convolutional Neural Network (CNN) adalah algoritma deep learning yang paling umum digunakan. [3] [12].

E. Convolutional Neural Network

Convolutional Neural Network (CNN), yang merupakan pengembangan dari Multi-Layer Perceptron (MLP), diciptakan untuk memproses data citra yang berdimensi dua. CNN termasuk ke dalam kategori Deep Neural Network karena memiliki banyak lapisan yang memungkinkan pemrosesan data citra yang lebih efisien.

Arsitektur CNN secara umum dapat digambarkan dalam diagram yang menggambarkan jaringan dan hubungan antara setiap layer. Arsitektur CNN terdiri dari beberapa tahap yang dapat dilatih dan menghasilkan output berupa peta fitur [1].



Gambar 2 2 Arsitektur Convolutional Neural Network (CNN)

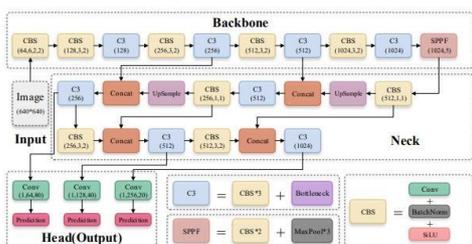
F. Computer Vision

Salah satu bagian dari kecerdasan buatan adalah visi komputer, yang dimaksudkan untuk membantu komputer dalam proses pengambilan keputusan dengan mengubah data visual menjadi informasi yang dapat diinterpretasikan. Pada dasarnya, komputer vision mengubah data digital visual menjadi deskripsi data multidimensi

yang dapat diinterpretasikan oleh mesin untuk mengumpulkan data. Oleh karena itu, tujuan utama dari visi komputer adalah untuk mengajarkan mesin cara mengolah data dari piksel-piksel visual [3].

G. YOLOv5

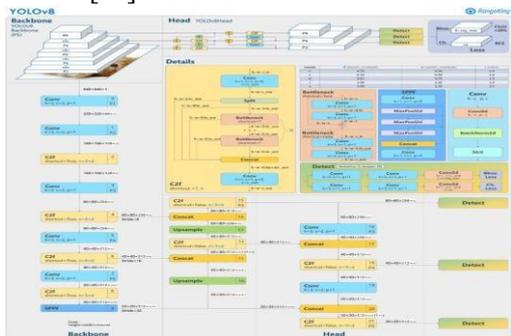
You Only Look Once (YOLO) mudah digunakan dan telah berkembang secara bertahap. Glenn Jocher, seorang peneliti dan CEO Ultralytics LLC, merilis YOLO versi 5 pada tahun 2020. YOLO tidak menggunakan jaringan terpisah untuk membagi area. YOLOv5 secara garis besar terdiri dari Backbone, Neck (Panet), dan Head [7].



Gambar 2 3 Struktur YOLOv5

H. YOLOv8

You Only Look Once (YOLO) adalah versi terbaru dari model YOLO, dan dapat digunakan untuk menemukan objek, mengklasifikasikan gambar, dan melakukan tugas segmentasi. Ultralytics mengembangkan YOLO versi 8 pada 10 Januari 2023, bersama dengan peluncuran YOLOv5 yang signifikan dan mendefinisikan industri. Dibandingkan dengan YOLOv5, YOLOv8 melakukan banyak perbaikan dan peningkatan dalam pengalaman pengembang dan arsitektur [11].



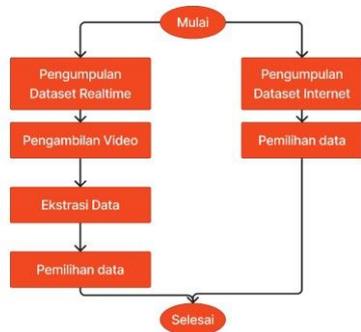
Gambar 2 4 Struktur YOLOv8

III. METODOLOGI PENELITIAN

Pada tahap ini menjelaskan alur penelitian yang dilakukan. Tahap pertama yaitu pengumpulan dataset yang dilakukan secara realtime.

Flowchart pengumpulan dataset ditunjukkan pada gambar 3.2.

A. Flowchart pengupulan dataset
Pada tahap ini menjelaskan alur riset dalam pengumpulan dataset yang dilakukan dalam riset ini, yang dimana alur data pengumpulan data riset yang dilakukan ada pada gambar 3.2.



Gambar 3 2 Flowchart Pengumpulan Dataset

Pengumpulan dataset dilakukan dengan 2 cara yaitu dataset realtime yang diambil secara langsung menggunakan camera dan dataset internet yang diambil melalui situs google.

B. Pelabelan Objek

Pelabelan objek bertujuan dalam memberikan data yang berisikan informasi berupa nama kelas dan posisi gambar yang berupa bounding box (xmin, ymin,xmax, ymax) yang akan dilakukan deteksi atau prediksi.



Gambar 3 3 Flowchart Pelabelan Objek

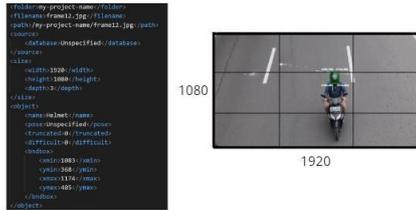
Penelitian ini menggunakan 3 objek yang terdiri dari *Helmet*, *NoHelmet* dan *Motorcycle*.



Gambar 3 4 Proses Pelabelan Objek

Pada dataset realtime objek yang diberi label hanya tampak depan sedangkan dataset internet

objek yang diberi label tampak depan, belakang dan samping.



Gambar 3.5 File XML dan Gambar Objek

Setelah dilakukan pelabelan objek pada **Gambar 3.4** kemudian di export dengan format *Extensible Markup Language* (XML) yang berisikan beberapa objek seperti pada **Gambar 3.5** diantaranya *filename*, *size (width, height)*, *object(name,xmin,xmax,ymin,ymax)*.



Gambar 3.6 Flowchart Label YOLO.

Penggunaan dalam melakukan proses *training* model YOLO membutuhkan label dengan format txt, dimana file XML dilakukan ekstrak pada objek *xmin,xmax,ymin,ymax* menjadi *center_x, center_y,w,h*.

Adapun rumus dalam melakukan perhitungan *center_x,center_y,w,h* :

- $Center_x = \frac{x_{min} + x_{max}}{2}$
width of the image
- $Center_y = \frac{y_{min} + y_{max}}{2}$
height of the image
- $w = \frac{x_{max} - x_{min}}{}$
- $h = \frac{y_{max} - y_{min}}{}$
height of the image



Gambar 3.7 Pelabelan Objek Model

D. Training



Gambar 3.8 Flowchart Training Dataset

Training dataset dilakukan agar komputer dapat mengenali atau mempelajari sebuah objek sehingga dapat melakukan deteksi objek dengan baik.

Dalam melakukan proses training dataset menggunakan Google Collaboratory yang memberikan server *Graphics Processing Unit* (GPU) secara gratis akan tetapi terdapat pembatasan ketika dalam melakukan proses *running training* model hanya mengizinkan hingga 12 jam.

E. Pengembangan Sistem

Tahapan pengembangan sistem pada penelitian ini melakukan pembangunan sistem berbasis web dengan memanfaatkan framework streamlit dan perhitungan deteksi pelanggaran menggunakan framework deepsort.

F. Pengujian

Pada penelitian ini, tahapan pengujian dilakukan untuk memeriksa apakah model yang telah dilatih menggunakan algoritma YOLO versi 5s dan 8s dapat digunakan untuk mendeteksi objek dengan efektif. Tahap ini bertujuan untuk memeriksa bahwa setiap bagian atau unsur dari sistem berjalan dengan baik dan optimal, sehingga memastikan bahwa hasil input dan output sesuai dengan tahapan yang telah ditentukan.

G. Evaluasi

Evaluasi merupakan tahapan penting untuk menilai kinerja dari model yang sudah dilatih dan memastikan bahwa hasilnya cukup akurat. Penelitian ini menggunakan *mean Average Precision* (mAP) sebagai evaluasi model.

$$mAP = \frac{1}{n} \sum_{k=1}^K AP_k$$

mAP : *mean Average Precision*
N : Jumlah data AP
AP : *Average Precision*

Confusion matrix atau matriks kebingungan digunakan untuk mengevaluasi kinerja model pada masalah klasifikasi atau prediksi.

	<i>Actual Positive</i>	<i>Actual Negative</i>
<i>Predicted Positive</i>	<i>True Positive (TP)</i>	<i>False Positive (FP)</i>
<i>Predicted Negative</i>	<i>False Negative (FN)</i>	<i>True Negative (TN)</i>

Tabel 3 1 *Confusion Matrix*

Confusion matrix memiliki 4 sel diantaranya :

- *True Positive (TP)* : Jumlah data yang berhasil diprediksi dengan akurasi sebagai kelas positif.
- *False Positive (FP)* : Jumlah data yang diprediksi secara keliru sebagai kelas positif, tetapi sebenarnya kelasnya negatif.
- *False Negative (FN)* : Jumlah data yang secara keliru diprediksi sebagai kelas negatif oleh model, meskipun sebenarnya kelasnya adalah positif.
- *True Negative (TN)* : Jumlah data yang berhasil diprediksi dengan benar sebagai kelas negatif adalah hasil true negative atau benar negatif, yaitu ketika model berhasil memprediksi data sebagai negatif ketika memang data tersebut benar-benar negatif.

Confusion matrix memberikan informasi penting untuk mengevaluasi kinerja model diantaranya [9]:

- *Precision* : Rasio antara jumlah prediksi positif yang benar dan total jumlah prediksi positif dikenal sebagai recall atau tingkat keberhasilan dalam mengidentifikasi hasil positif.
 $Precision = TP / (TP + FP)$
- *Recall* : Perbandingan antara jumlah prediksi benar untuk kelas positif dan total jumlah data yang sebenarnya positif. $Recall = TP / (TP + FN)$
- *F1-Score* : Rata-rata harmonik antara *Precision* dan *Recall*, yang memberikan

pengukuran yang seimbang antara kedua metrik dalam mengevaluasi performa suatu model dalam memprediksi kelas positif dan negatif.

$$F1-Score = \frac{2 * (Precision * Recall)}{(Precision + Recall)}$$

IV. IMPLEMENTASI DAN EVALUASI

Implementasi dari hasil perancangan yang meliputi tahapan extract dataset, pelabelan objek, training dataset menggunakan algoritma YOLO version 5s dan 8s, pembuatan aplikasi berbasis website menggunakan framework streamlit dan melakukan perhitungan jumlah pelanggaran menggunakan framework deepsort. Pada implementasi ini menggunakan bahasa pemrograman python versi 3.9.12.

A. Pengumpulan Dataset

Dataset realtime memiliki total sebanyak 1.274 gambar.



Gambar 4 1 Dataset Realtime.

Sedangkan dataset internet memiliki total sebanyak 294 gambar.



Gambar 4.2 Dataset Internet

B. Pelabelan Objek

Seperti pada Gambar 3.4 dimana melakukan pelabelan objek dengan bentuk bounding box atau persegi empat yang melingkupi suatu objek. Adapun jumlah label pada objek dengan tiga kelas yaitu *Helmet*, *NoHelmet*, dan *Motorcycle*.

No	Data	<i>Helmet</i>	<i>NoHelmet</i>	<i>Motorcycle</i>	Jumlah

1	Real Time	2615	998	2667	6280
2	Internet	371	407	476	1254

Tabel 4 1 Data Label Objek

Kemudian masing-masing dataset realtime dan internet dilakukan split atau pemisahan data dengan persentase 80% data *training* dan 20% data *testing*.

No	Dataset	Data Training	Data Testing
1	Realtime	1019	255
2	Internet	235	59

Tabel 4 2 Hasil Dataset yang telah di Split

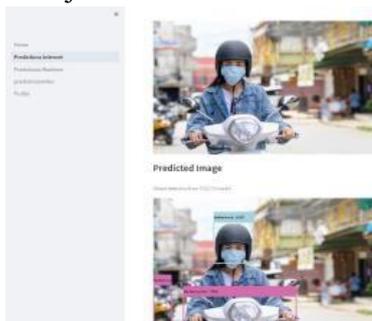
C. Training Dataset Algoritma YOLO

Proses training dataset dengan algoritma YOLOv5s dan YOLOv8s menggunakan Google Collaboratory karena terdapat fitur GPU (Graphics Processing Unit). Karena algoritma YOLO membutuhkan sistem untuk memiliki kecepatan dan kekuatan pemrosesan yang tinggi.

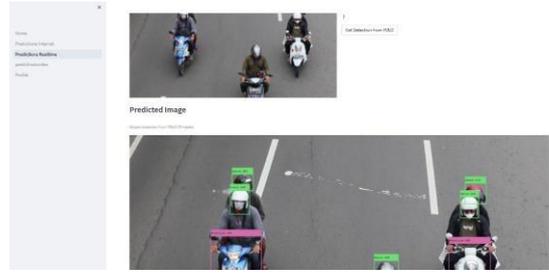
Pada tahap training model YOLOv5s pada Google Collaboratory membutuhkan waktu ±36 menit untuk dataset internet dan dataset realtime membutuhkan waktu ± 5 jam. Sedangkan training model YOLOv8s pada Google Collaboratory membutuhkan waktu ± 46 menit untuk dataset internet dan dataset realtime ± 5 jam.

D. Streamlit

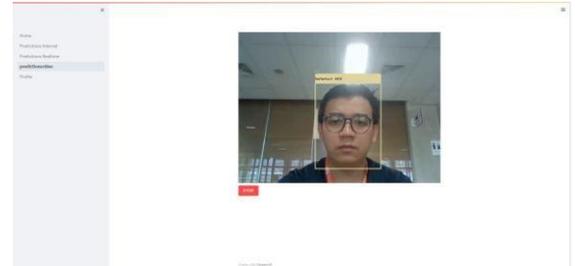
Penelitian ini menggunakan framework streamlit sebagai output dalam melakukan deteksi objek.



Gambar 4 4 Prediksi Dataset Internet



Gambar 4 3 Prediksi Dataset Realtime.



Gambar 4 5 Prediksi Video



Gambar 4 6 Halaman Profile

Pada implementasi streamlit hanya bisa menggunakan model YOLOv5s.

E. Deepsort

Framework deepsort yang digunakan untuk melakukan perhitungan jumlah pelanggaran salah satunya tidak menggunakan helm. Penelitian ini melakukan perhitungan pelanggaran pada 3 titik lokasi dengan video berdurasi sekitar ± 4 menit.

Pada lokasi JPO Lenteng Agung terdapat 103 pelanggaran.



Gambar 4 7 Lokasi JPO Lenteng Agung

Pada lokasi JPO UIN Syarif Hidayatullah terdapat 88 pelanggaran yang ditunjukkan pada gambar 4.8



Gambar 4 8 Lokasi JPO UIN Syarif Hidayatullah

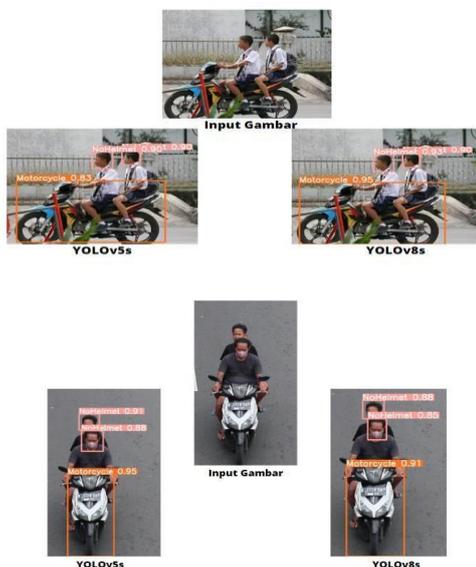
Pada lokasi JPO Andara terdapat 103 pelanggaran yang ditunjukkan pada gambar 4.9.



Gambar 4 9 Lokasi JPO Andara

F. Pengujian

Pengujian deteksi dilakukan dengan menggunakan berbagai posisi objek untuk mengukur keakuratan model dalam melakukan deteksi, termasuk pengendara dan penumpang yang tidak menggunakan helm.



Gambar 4 10 Deteksi Pengujian Dataset Internet & Realtime

Pada Gambar 4.10 dataset internet model dapat mendeteksi pengendara dan penumpang tidak menggunakan helm. Pada model YOLOv5s mendapatkan nilai confidence pada objek nohelm sebesar 90% pada pengendara serta penumpang dan objek motorcycle sebesar 83%, sedangkan model YOLOv8s mendapatkan nilai confidence pada objek nohelm 93% pada pengendara, 90% pada penumpang dan objek motorcycle 95%.

Sedangkan pada dataset realtime model YOLOv5s mendapatkan nilai confidence pada objek nohelm sebesar 88% pada pengendara, 91% pada penumpang dan objek motorcycle sebesar 95%, sedangkan model YOLOv8s mendapatkan nilai confidence pada objek nohelm 85% pada pengendara, 88% pada penumpang dan objek motorcycle 91%. • Kelebihan Penumpang.

Dataset internet model dapat mengidentifikasi lebih dari dua individu yang menggunakan satu sepeda motor, seperti yang ditunjukkan pada Gambar 4.11. Pada model YOLOv5s, deteksi benar-benar positif mendeteksi objek helm dengan nilai keyakinan sebesar 89% pada pengendara, 92% pada penumpang paling belakang, dan nilai keyakinan sebesar 88% pada motor, dan deteksi false negative mendeteksi objek nohelm yang sebenarnya adalah helm dengan nilai keyakinan sebesar 78% pada penumpang paling belakang.



Gambar 4 11 Deteksi Pengujian Kelebihan Penumpang Dataset

Internet & Realtime

Pada dataset real-time, model YOLOv5s memiliki keyakinan sebesar 89% pada penumpang depan, 85% pada pengendara, 89% pada penumpang belakang, dan 97% pada motor. Di sisi lain, model YOLOv8s memiliki keyakinan sebesar 73% pada penumpang depan, 84% pada pengendara, 84% pada penumpang belakang, dan 91% pada motor.

G. Evaluasi

Tahapan evaluasi dilakukan untuk mengetahui tingkat akurasi dari hasil pengujian menggunakan algoritma You Only Look Once versi 5s dan 8s yang menggunakan 2 sumber dataset diantaranya dataset internet dan dataset realtime. Dalam tahap evaluasi awal, *confusion matrix* digunakan untuk merepresentasikan performa model klasifikasi pada sejumlah data uji di mana nilai sebenarnya telah diketahui.

Pada algoritma You Only Look Once versi 5s dan 8s menggunakan dataset internet melakukan pengujian dengan gambar sebanyak 59, dimana pada gambar tersebut memiliki lebih dari 1 object. Sehingga untuk melakukan perhitungan confusion matrix menggunakan jumlah objek dengan total 228, terdiri dari 53 objek Helmet, 85 objek NoHelmet, 90 objek Motorcycle.

Pada algoritma You Only Look Once versi 5s menggunakan dataset realtime melakukan pengujian dengan gambar sebanyak 255, dimana pada gambar tersebut memiliki lebih dari 1 object. Sehingga dalam melakukan perhitungan confusion matrix menggunakan jumlah objek yang berjumlah 1.256, terdiri dari 540 objek Helmet, 193, objek NoHelmet, 532 objek Motorcycle. Sedangkan pada You Only Look Once versi 8s terdapat jumlah objek yang berjumlah 1344, terdiri dari 539 objek Helmet, 235 objek NoHelmet, 570 objek Motorcycle.

Tabel 4 3 Hasil Evaluasi .

Model	Dataset	Precision	Recall	mAP 50
-------	---------	-----------	--------	-----------

YOLOv5s	Internet	90%	89%	94%
	Realtime	92%	89%	93%
YOLOv8s	Internet	92%	88%	93%
	Realtime	92%	89%	93%

V. SIMPULAN

Berdasarkan pembahasan dari implemtasi, pengujian dan hasil evaluasi, maka dapat ditarik beberapa kesimpulan diantaranya sebagai berikut:

1. Dengan menggunakan model algoritma You Only Look Once (YOLO) versi 5s dan 8s berhasil melakukan deteksi penggunaan helm dan penumpang berlebihan berdasarkan penggunaan helm ataupun tidak menggunakan helm.
2. Menggunakan framework streamlit dalam membangun sistem deteksi berbasis websiste, berhasil melakukan deteksi gambar dengan baik akan tetapi hanya bisa digunakan pada model Algoritma YOLO versi 5s.
3. Penggunaan framework deepsort hanya bisa diimplementasikan menggunakan model YOLOV5s dengan menghitung jumlah pelanggaran yang tidak menggunakan helm bagi pengendara maupun penumpang.
4. Jumlah pelanggaran yang tidak menggunakan helm bagi pengendara maupun penumpang pada lokasi JPO Lenteng Agung dan JPO Andara sebanyak 103 pelanggaran sedangkan pada JPO UIN Syarif Hidayatullah sebanyak 88 pelanggaran.
5. Dengan menggunakan mean Average Precision sebagai evaluasi model, menunjukkan bahwa YOLOv5s dengan dataset internet memiliki tingkat akurasi yang tinggi sebesar 94%, YOLOv5s dataset realtime sebesar 93%, YOLOv8s dataset internet sebesar 93% dan YOLOv8s dataset realtime sebesar 93%.

VI. DAFTAR PUSTAKA

[1] Sofia, N., 2018. *Medium*. [Online] Available at: <https://medium.com/@nadhifasofia/1convolutional-neural->

- [network-convolutional-neuralnetwork-merupakan-salah-satu-metode-machine28189e17335b](#)
[Diakses 28 October 2022].
- [2] Adhiwijaya, M. D., 2021. *Deteksi Helm Sepeda Motor Berbasis WEB dengan Algoritma You Only Look Once (YOLO)*, Jakarta: Politeknik Negeri Jakarta.
- [3] Chatterjee, M., 2022. *Great Learning*. [Online] Available at: <https://www.mygreatlearning.com/blog/what-is-computer-vision-the-basics/> [Diakses 29 October 2022].
- [4] Copeland, B., 2022. *britannica*. [Online] Available at: <https://www.britannica.com/technology/artificialintelligence> [Diakses 1 November 2022].
- [5] Hanafi, Y. U., 2020. *Deteksi Penggunaan Helm Pada Pengendara Bermotor Berbasis Deep Learning*, Surabaya: Institut Teknologi Sepuluh Nopember.
- [6] Hermana, A. N., 2022. *PENGEMBANGAN SISTEM DETEKSI HELM PADA PENGENDARA SEPEDA MOTOR MENGGUNAKAN METODA HOUGH CIRCLE TRANSFORM (HCT)*, Bandung: INSTITUT TEKNOLOGI NASIONAL.
- [7] H.-K. J. & G.-S. C., 2022. Improved YOLOv5: Efficient Object Detection Using Drone Images under Various Conditions. *Applied Sciences*, II(12), pp. 116.
- [8] Kurnia, E., 2022. *Kompas*. [Online] Available at: <https://www.kompas.id/baca/metro/2022/03/01/polisi-klaim-kecelakaan-lalu-lintas-di-jakarta-dansekitarnya-menurun> [Diakses 18 Oktober 2022].
- [9] Narkhede, S., 2018. *Understanding Confusion Matrix*. [Online] Available at: <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62> [Diakses 28 Februari 2023].
- [10] R, M. et al., 2021. Neural Network Based Smart City Application for Traffic Violation Detection. *International Journal of Advanced Scientific Innovation*, II(04), pp. 1-5.
- [11] Solawetz, J. & F., 2023. *Roboflow*. [Online] Available at: <https://blog.roboflow.com/whats-new-in-yolov8/#:~:text=YOLOv8%20was%20launched%20on%20January%2010th%2C%202023> [Diakses 11 02 2023].
- [12] Yোগameena, B., Menaka, K. & Perumaal, S. S., 2019. Deep Learning-Based Helmet Wear Analysis of a Motorcycle Rider For Intelligent Surveillance System. *IET*, 13(7), pp. 1190-1198.
- [13] Zulkiflie, M. A., 2021. *Implementasi Algoritma Object Detection YOLOv4 dan Euclidean Distance dalam Mendeteksi Pelanggaran Sosial Distancing*, Makassar: Universitas Hasanuddin.