

# Sistem Rekognisi Jenis Kendaraan Dengan Analisis Video Menggunakan Metode Yolov4

**Ichsan Surya Dharma<sup>1</sup>, Fauziah<sup>2\*</sup>, Winarsih<sup>3</sup>**

Program Studi Sistem Informasi, Universitas Nasional, Jakarta  
Jalan Sawo Manila, Pejaten, Pasar Minggu, Jakarta, 12520

ichsansuryad@gmail.com, fauziah@civitas.unas.ac.id\*, winarsih@civitas.unas.ac.id  
Korespondensi : fauziah@civitas.unas.ac.id

## **Abstract**

*Traffic density on the highway, especially at intersections where there are traffic lights, is a number that can be used to estimate the number of vehicles. The increase in the density of highways is directly proportional to the growth in Indonesia's population which is increasing. The problem was found that the survey to record the types of vehicles in Indonesia was still manual by taking notes and coming directly to the location. Based on this problem, a vehicle object detection system was created, especially cars based on their brands and variants to facilitate the police survey process to detect violations that have occurred. This study uses the You Only Look Once (YOLOv4) algorithm to detect cars, and classify and determine the level of accuracy. This research uses 7034 image dataset with 8 classes, namely Motorcycle, Suzuki Ertiga, Honda Jazz, Honda Brio, Toyota Avanza, Toyota Innova, Mitsubishi Xpander, Mitsubishi Pajero Sport. The results of this study indicate that the YOLOv4 algorithm can be used on Jogja CCTV to detect vehicle types with an accuracy of 82%.*

**Keywords:** Object Detection, Classification, You only look once v4 Algoritih, CCTV

## **Abstrak**

*Kepadatan lalu lintas di jalan raya, khususnya pada persimpangan yang terdapat lampu lalu lintas, merupakan angka yang dapat digunakan untuk memperkirakan jumlah kendaraan. Peningkatan kepadatan jalan raya berbanding lurus dengan pertumbuhan jumlah penduduk Indonesia yang kian meningkat. Ditemukan permasalahan bahwa survei untuk mendata jenis kendaraan di Indonesia masih manual dengan mencatat dan datang langsung ke lokasi. Berdasarkan masalah tersebut, dibuatlah sistem deteksi objek kendaraan khususnya mobil berdasarkan merk dan variannya untuk mempermudah proses survei yang dilakukan kepolisian untuk mendeteksi pelanggaran yang terjadi. Penelitian ini menggunakan algoritma You Only Look Once (YOLOv4) untuk mendeteksi mobil, dan klasifikasi serta mengetahui tingkat akurasi. Penelitian ini menggunakan 7034 Dataset gambar dengan 8 kelas yaitu Sepeda Motor, Suzuki Ertiga, Honda Jazz, Honda Brio, Toyota Avanza, Toyota Innova, Mitsubishi Xpander, Mitsubishi Pajero Sport. Hasil dari penelitian ini menunjukkan bahwa algoritma YOLOv4 dapat digunakan pada CCTV Jogja untuk mendeteksi jenis kendaraan dengan akurasi sebesar 82%.*

**Kata kunci:** Deteksi Objek, Klasifikasi, Algoritma You Only Look Once v4, CCTV

## 1. PENDAHULUAN

Ketika berpindah dari satu lokasi ke lokasi lain di Pulau Jawa, sebagian besar penduduk mengandalkan berbagai bentuk transportasi kendaraan[1]. Kepadatan lalu lintas di jalan raya, khususnya pada persimpangan yang terdapat lampu lalu lintas, merupakan angka yang dapat digunakan untuk memperkirakan jumlah kendaraan. Dari angka tersebut dapat ditarik kesimpulan bahwa mayoritas masyarakat yang tinggal di Indonesia menggunakan kendaraan bermotor khususnya kendaraan pribadi.

Selama ini survei yang dilakukan oleh kepolisian masih dilakukan dengan cara manual yaitu dengan menggunakan tenaga manusia dan menghitung jenis mobil dengan mencatat[2]. Hal tersebut tidak efektif karena sering menimbulkan berbagai macam kesalahan seperti petugas yang kelelahan, tidak tercatatnya mobil yang terhalang mobil lainnya, petugas yang tidak fokus karena diajak mengobrol.

Berdasarkan masalah tersebut, Peneliti berusaha untuk membuat fitur baru dari deteksi objek kendaraan yaitu sebuah sistem rekognisi kendaraan roda dua dan roda empat yang dapat memudahkan kepolisian untuk dapat mendeteksi kendaraan berdasarkan merk dan jenisnya.

Penelitian yang berkaitan dengan membuat sistem deteksi untuk menentukan jenis kendaraan dan jumlah lahan parkir yang tersedia. Objek yang dideteksi adalah berdasarkan ukurannya seperti motor, mobil, bus, dan truk. Algoritma yang digunakan adalah algoritma YOLOv1. Pendeteksian jenis kendaraan dipengaruhi dengan kualitas foto yang dipakai[3].

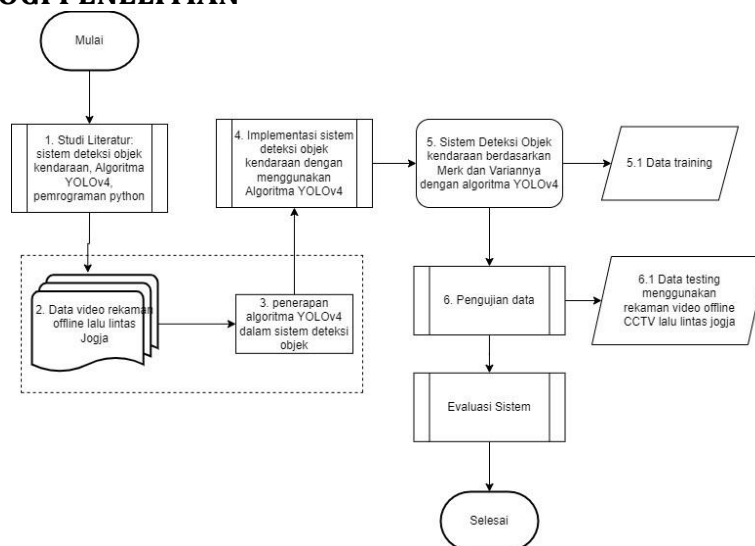
Penelitian selanjutnya memiliki masalah utama yaitu meningkatnya jumlah kendaraan bermotor dan semakin padatnya lalu lintas sehingga dibuatlah sistem deteksi plat nomor kendaraan dengan menggunakan metode Convolutional Neural Network (CNN) untuk dapat mengenali huruf karakter[4].

Adapun penelitian selanjutnya yang membahas tentang deteksi jumlah kendaraan. Masalah utama yang dihadapi adalah kemacetan yang terjadi di sekitar tugu kundang yang disebabkan oleh tingginya volume kendaraan. Oleh karena itu, dibuatlah sistem aplikasi yang diperuntukkan untuk mendeteksi kemacetan. Dengan menggunakan algoritma YOLOv3, sistem dapat mengenali, mendeteksi objek dan menghitung jumlah kendaraan yang melintas sesuai dengan jenis dan ukurannya[5].

Kemudian penelitian selanjutnya mengangkat masalah tentang bahu jalan pada jalan bebas hambatan atau jalan tol sering sekali disalahgunakan oleh pengendara. Pengendara sering kali menyalip kendaraan di depannya menggunakan bahu jalan. Selain digunakan untuk menyalip, bahu jalan juga sering digunakan untuk kepentingan lain yang tidak darurat. Hal ini dapat merugikan pengendara lain. Untuk itu, dibuatlah sebuah sistem deteksi kendaraan menggunakan algoritma YOLOv4. Hasil yang didapat adalah akurasi dari sistem deteksi objek mencapai angka rata-rata 80%[6].

Penelitian selanjutnya membahas tentang deteksi jenis kendaraan. Masalah utama yang dihadapi adalah seringnya terjadi kepadatan sehingga menyebabkan kemacetan di gerbang tol otomatis (GTO). Kemacetan yang terjadi dikarenakan proses penggolongan kendaraan masih dilakukan dengan cara manual oleh petugas jalan tol. Maka, dibuatlah deteksi golongan jenis kendaraan berdasarkan golongannya di gerbang tol dengan menggunakan algoritma YOLOv4[7].

## 2. METODOLOGI PENELITIAN



Gambar 2.1 Tahapan Penelitian

Gambar 2.1 Menjelaskan tentang alur tahapan penelitian. Penelitian dilakukan dengan melewati beberapa proses, yaitu studi literatur, pengumpulan data, pelatihan data, dan pengujian data. Tahapan penelitian tidak dapat dilakukan dengan cara melewati tahap yang lainnya karena untuk melewati satu tahapan harus melewati tahapan sebelumnya.

### 2.1 Pengumpulan Data

Pada tahap pengumpulan data, data yang dikumpulkan akan dilatih dan dilakukan testing pada rekaman video. Pengumpulan data dilakukan secara online dengan menggunakan rekaman video CCTV Jogjakota (<https://cctv.jogjakota.go.id/>). CCTV ini dapat diakses secara online dan memiliki kualitas kamera yang cukup bagus. Lokasi dari penelitian ini adalah di CCTV simpang demangan Jogja dimana pada persimpangan tersebut intensitas volume kendaraan sangat ramai dan terdapat jalan satu arah. Digunakan Aplikasi OBS Studio untuk melakukan rekaman dari situs CCTV Jogjakota dan selanjutnya disimpan secara lokal dengan format MP4. Data foto diambil melalui:

- a. Website Google Images,
- b. File rekaman video cctv dari website [www.cctv.jogjakota.go.id](http://www.cctv.jogjakota.go.id)

Data yang direkam akan digunakan sebagai bahan pelatihan YOLO v4 dalam format ".jpg". Konversi akan dilakukan dengan menggunakan skrip perangkat lunak yang ditulis dengan Python dan lingkungan pengembangan terintegrasi Jupyter Notebook. Penulis juga mengambil objek data dari internet yaitu google images yang terdiri dari tampilan depan mobil untuk melakukan proses pelatihan. Jumlah dataset yang berhasil dikumpulkan berjumlah 7034 data. Data tersebut terdiri dari delapan kelas yang dijelaskan pada Tabel 2.1

Tabel 2.1 Kelas Dataset

No.	Kelas
1	Sepeda Motor
2	Honda Jazz
3	Honda Brio
4	Suzuki Ertiga
5	Mitsubishi Pajero Sport
6	Mitsubishi Xpander
7	Toyota Avanza
8	Toyota Innova

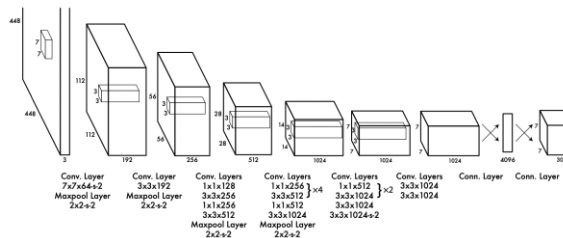
## 2.2 Metode Penelitian

Penelitian ini menggunakan Algoritma *You Only Look Once V4* (YOLOv4) untuk mendeteksi objek sesuai kelas yang dideteksi.

### 2.2.1 Algoritma YOLOv4

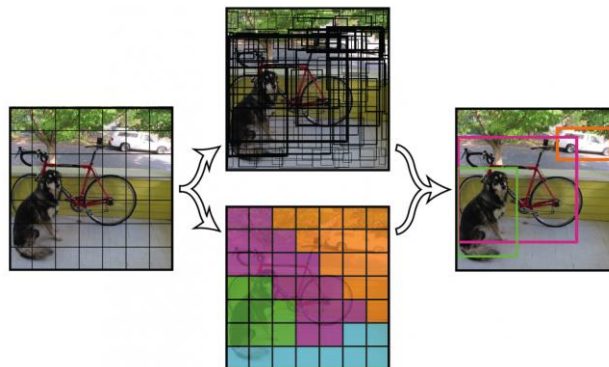
YOLO adalah sebuah singkatan dari "You Only Look Once" yang merupakan algoritma berbasis regresi. Jaringan pengenalan arsitektur YOLO terdiri dari 24 lapisan konvolusi, diikuti oleh 2 lapisan terkait. Joseph Redmon menemukan YOLO pada tahun 2015[8]. Metode ini berkembang dari jaringan YOLO dan YOLOV2 dan memiliki tingkat akurasi yang tinggi sekaligus menjaga kecepatan pemrosesan yang cepat. Setiap gambar dalam set pelatihan dibagi menjadi  $S \times S$  ( $S = 14$ ) oleh jaringan YOLO. Jika pusat kebenaran dasar target berada dalam delapan grid, grid bertugas menemukan target. Setiap kisi memperkirakan skor kepercayaan, kotak pembatas B, dan probabilitas bersyarat kelas C. memilih kotak pembatas yang ideal menggunakan teknik Non-Maximum Suppression (NMS). Faster-RCNN menawarkan kecepatan lebih dari YOLO, tetapi YOLO memiliki lebih banyak deteksi kesalahan. Untuk mengatasi masalah ini, YOLOv2 mengusulkan konsep "kotak jangkar" di Faster R-CNN dan membuat kotak pemisah sebelumnya yang sesuai menggunakan algoritma pengelompokan k-means. Akibatnya, jumlah kotak jangkar yang dibutuhkan untuk memberikan

hasil yang sama dengan persimpangan pada penyatuan (IOU). Dengan menggunakan lapisan konvolusi untuk mengaktifkan lapisan sepenuhnya pada akhir produksi yolo, YOLO-V2 meningkatkan struktur jaringan. Seiring dengan normalisasi batch, klasifikasi resolusi tinggi, pengelompokan dimensi, prediksi lokasi langsung, fitur halus, multiskala, dan teknik lainnya, YOLO-V2 secara signifikan meningkatkan akurasi deteksi dibandingkan YOLO. Versi Yolo V2 yang lebih baik adalah Yolo V3. Struktur jaringannya lebih rumit daripada YOLO-V2 dan menggunakan prediksi multi-skala untuk menemukan target akhir. Dengan kemampuannya untuk meramalkan kotak pembatas pada berbagai skala, YOLO V3 lebih unggul dari YOLO V2 dalam menemukan target kecil[3].



Gambar 2.2 Arsitektur YOLO

Pada Gambar 2.2 YOLO memiliki 24 Jaringan Konvolusi dan 2 jaringan layar penuh.



Gambar 2.3 Proses Deteksi YOLO

Gambar 2.3 menunjukkan YOLO memproses seluruh gambar menggunakan pendekatan jaringan saraf tunggal, yang sangat berbeda dari teknik sebelumnya. Jaringan ini akan membagi gambar menjadi wilayah dan kemudian memperkirakan kotak pembatas dan probabilitas untuk setiap kotak area pembatas, dengan opsi untuk mengkategorikan setiap kotak area pembatas sebagai item atau tidak berbobot[9].

### 2.3 Mean Average Precision

Mean Average Precision adalah nilai presisi rata-rata (AP). Mean average precision digunakan sebagai metrik penilaian untuk menilai kinerja sistem deteksi objek. Nilai AP dihitung menggunakan perhitungan presisi,

dan perhitungan recall. yang selanjutnya dihitung seperti yang ditunjukkan pada persamaan 1[10].

$$AP = \sum (recall_{n+1} - recall_n) \cdot Precision_{interp.(recall_{n+1})} \quad (1)$$

## 2.4 Precision

Nilai precision dihitung dengan membagi jumlah total sampel positif yang diklasifikasikan dengan benar dengan total sampel positif yang diantisipasi, seperti yang ditunjukkan pada persamaan 2. Presisi tinggi berarti contoh berlabel positif memang positif (FP kecil)[7].

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

## 2.5 Recall

Recall didefinisikan sebagai rasio sampel positif yang dikategorikan dengan benar terhadap total sampel positif. Penarikan kembali yang tinggi menunjukkan bahwa kelas tersebut diidentifikasi dengan benar (sedikit FN). Persamaan 3 dapat digunakan untuk menghitung perolehan recall[4].

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

## 2.6 Confusion Matrix

		Actual Values	
		1 (Positive)	0 (Negative)
Predicted Values	1 (Positive)	<b>TP</b> (True Positive)	<b>FP</b> (False Positive) <i>Type I Error</i>
	0 (Negative)	<b>FN</b> (False Negative) <i>Type II Error</i>	<b>TN</b> (True Negative)

Gambar 2.4 Confusion Matrix

Berdasarkan gambar 2.4, matriks konfusi memiliki keterangan sebagai berikut:

- a. Positive (P): aktual positif.
- b. Negative (N): aktual negatif.

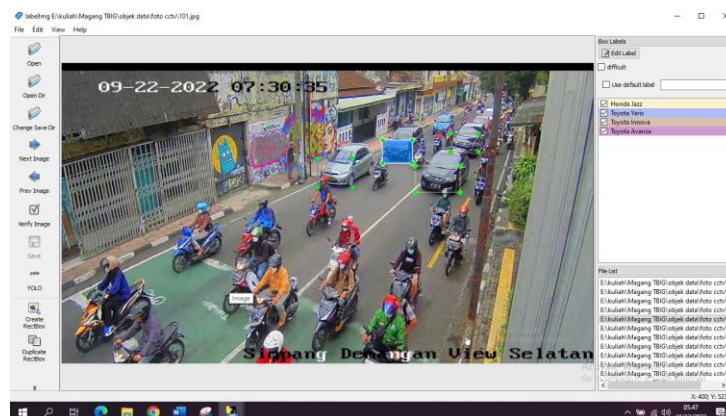
- c. True Positive (TP): aktual positif, prediksi positif. Contohnya ada mobil dengan kelas ertiga, keadaan aslinya mobil tersebut adalah suzuki ertiga dan sistem mendeteksi sebagai ertiga maka dikatakan sebagai True Positive.
  - d. True Negative (TN): aktual negatif, prediksi negatif. Contohnya mobil yang melewati ruas jalan adalah daihatsu grand max, grand max tersebut tidak dimiliki dalam kelas, serta sistem tidak mendeteksi sebagai kelas mobil yang ada.
  - e. False Positive (FP): aktual negatif, prediksi positif. Contohnya adalah jika mobil yang lewat tidak dimiliki dalam dataset, tetapi sistem memprediksi sebagai salah satu kelas mobil yang dimiliki, maka disebut seagai false positive.
  - f. False Negative (FN): aktual positif, prediksi negatif. Contohnya adalah jika mobil yang melintas terdapat dalam dataset tetapi sistem salah memprediksi maka disebut dengan false negative.
- Lalu, untuk menghitung akurasi, digunakan persamaan:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (4)$$

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pelabelan Objek

Setelah dataset disiapkan, selanjutnya adalah membuat anotasi pada dataset. Praktek pembuatan label pada foto dengan cara membubuhkan nama kelas kotak pembatas (bounding box) pada barang tersebut dikenal dengan istilah anotasi atau pelabelan. Objek yang diberi label adalah pada tampak depan. Proses pelabelan untuk objek data mobil dimulai dari bagian lampu lalu bagian grill dan bumper mobil. Sedangkan untuk sepeda motor, proses pelabelan dimulai dari helm pengendara motor hingga ban depan sepeda motor. Proses pemberian label pada gambar menggunakan alat Labeling.



Gambar 3.1 Proses Labelling

Pada Gambar 3.1 menunjukkan Proses anotasi citra dataset dengan menggunakan perangkat lunak Labelimg (label image). Proses anotasi dilakukan secara manual dengan memberikan sebuah kotak pembatas pada objek yang dimiliki classnya. Proses anotasi dilakukan dengan format YOLO.

```
24 0.467187 0.413194 0.067187 0.106944  
26 0.692578 0.423611 0.072656 0.105556  
16 0.603516 0.313194 0.057031 0.079167  
20 0.751563 0.290972 0.057813 0.070833
```

Gambar 3.2 Hasil *labelling*

Pada Gambar 3.2 menjelaskan hasil dari proses pelabelan objek dengan keterangan sebagai berikut:

- Nomor kelas objek terletak paling kiri. 0 untuk Sepeda Motor, 5 untuk honda brio, 11 untuk mitsubishi xpander, 16 untuk toyota avanza, 20 untuk toyota innova, 25 untuk suzuki ertiga, 26 untuk honda jazz, 27 untuk mitsubishi pajero sport.
- Koordinat x dan y objek kelas pada urutan kedua dan ketiga.
- Lebar dan Tinggi objek terdapat pada urutan ke empat dan lima (dari kiri ke kanan).

### 3.2 Pelatihan Data

Langkah selanjutnya yang terjadi setelah proses anotasi selesai adalah proses pelatihan atau disebut juga *training* terhadap dataset yang telah terkumpul. Tujuan dari proses ini adalah untuk melatih dataset, yang dilakukan dengan memproses file gambar dan anotasi yang telah dibuat. Hal ini dilakukan agar dapat terbentuk karakteristik dari masing-masing kelas yang selanjutnya akan dijadikan sebagai bahan pertimbangan komputer dalam melakukan prediksi.

Karena proses training sangat berat, maka penulis menggunakan aplikasi Google Collab yang disediakan secara gratis oleh Google. Ketika proses pelatihan berjalan, dibutuhkan koneksi internet yang stabil agar koneksi tidak terputus dan runtime dapat berjalan secara terus-menerus. Proses pelatihan menggunakan framework Darknet dengan konstruksi sebagai berikut.

Konfigurasi	Keterangan
Model	Darknet53
Weight	Yolov4
CUDNN	1
GPU	1
OPENCN	1



Framework darknet digunakan sebagai model beban dan YOLOv4 sebagai bobot beban dalam proses pelatihan jaringan. Batch dan subdivisi adalah terminologi yang digunakan dalam YOLOv4. Pengaturan batch membatasi jumlah foto yang dapat diproses sebelum bobot diubah. Sedangkan Subdivision digunakan pada GPU untuk memproses batch. GPU digunakan untuk mempercepat proses pelatihan. Teknik pemasangan Google Drive diperlukan untuk melakukan proses pelatihan.

Pada proses pelatihan ini, akan digunakan 7034 citra *train set* yang terdiri dari foto tampilan depan grill mobil dan rekaman cctv lalu lintas beserta bobot *pretrained* darknet53. Proses pelatihan YOLOv4 akan dilakukan menggunakan aplikasi Google Collab.

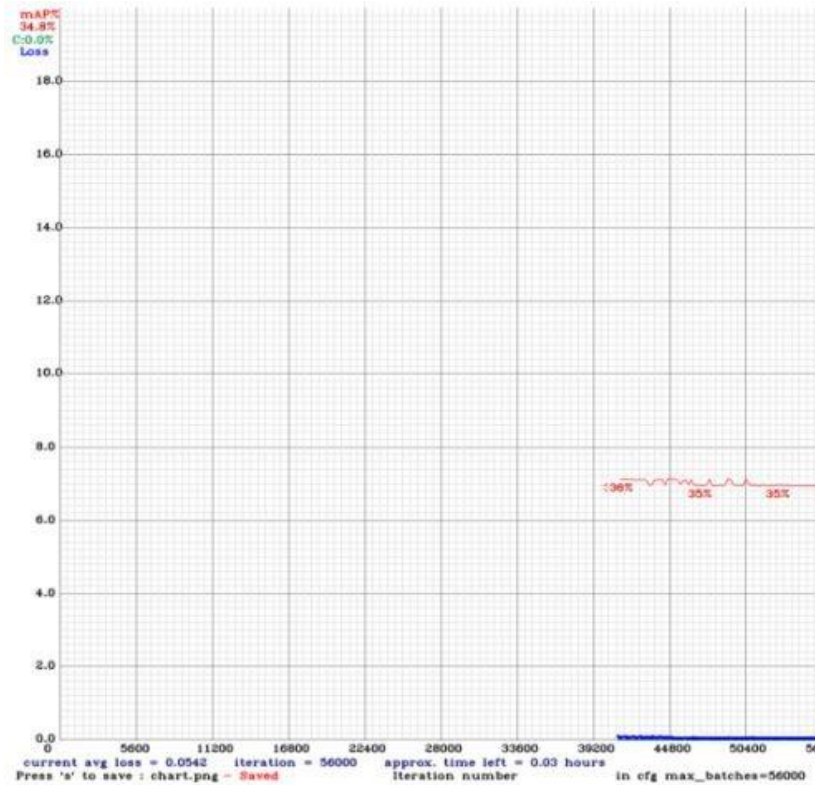
### 3.3 Pengujian Performa

Matriks konfusi, perolehan, akurasi, dan skor F1 jaringan deteksi kendaraan yang dilatih akan dihitung sebagai bagian dari penilaian kinerja jaringan YOLO v4.

```
def imshow(path):  
    import cv2  
    import matplotlib.pyplot as plt  
    %matplotlib inline  
  
    image = cv2.imread(path)  
    height, width = image.shape[:2]  
    resized_image = cv2.resize(image, (3*width, 3*height), interpolation = cv2.INTER_CUBIC)  
  
    fig = plt.gcf()  
    fig.set_size_inches(18, 10)  
    plt.axis("off")  
    plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
```

Gambar 3.2 Kode uji performa

Gambar 3.2 menunjukkan kode program untuk menguji kinerja, yang menggunakan import cv2. Library cv2 digunakan untuk menyederhanakan pemrosesan gambar, sedangkan library numpy digunakan untuk melakukan perhitungan matematis.



Gambar 3.3 Grafik Performa MAP

Pada Gambar 4.18 menunjukkan grafik hasil pengujian performa menunjukkan rata-rata presisi di angka 34.8%. hal ini disebabkan oleh beragamnya objek yang dideteksi, objek yang bergerak juga mempengaruhi tingkat angka *mean average precision* sehingga memberikan efek blur terhadap objek yang dideteksi, rendahnya resolusi kamera juga berpengaruh terhadap *dataset* yang diuji.

Selanjutnya dilakukan pengujian terhadap bobot terbaik yang sudah dilatih. Dari pelatihan, didapat hasil pengujian sebagai berikut:

Tabel 3.2 Hasil *training dataset*

Kelas	Average Precision	True Positive	Waktu Pemrosesan
Motor	96.16%	72	6 Detik
Jazz	100.00%	71	6 Detik
Brio	100.00%	44	6 Detik
Ertiga	100.00%	66	6 Detik
Pajero Sport	100.00%	65	6 Detik
Xpander	100.00%	53	6 Detik
Avanza	99.83%	69	6 Detik
Innova	100.00%	46	6 Detik

Berdasarkan Tabel 4.1 Dapat diketahui bahwa tingkat rata-rata presisi menunjukkan angka sebesar 99,49% dimana terdapat total 608 kendaraan terdeteksi sesuai dengan kelasnya masing-masing dengan waktu pemrosesan rata-rata 6 detik. Setelah pengujian performa, selanjutnya peneliti melakukan testing menggunakan video rekaman cctv selama 1 menit. Hasil yang didapat adalah sebagai berikut:



Gambar 3.4 Hasil Pengujian video 1 menit

Berdasarkan gambar 3.4 dapat dilihat bahwa tingkat akurasi terhadap mobil dengan kelas Motor 97 – 100 %, Toyota Yaris sebesar 96%, Toyota Avanza 100%, Toyota Innova 100% dan Honda Jazz 100%. Hasil dari test cukup baik. Tingkat akurasi dipengaruhi oleh kecepatan laju kendaraan serta objek tidak terhalang oleh objek lainnya. Dari pengujian satu menit video rekaman cctv, diperoleh hasil sebagai berikut:

Tabel 3.3 Hasil tes video 1 menit

Kelas	True Positive	False Positive
Sepeda Motor	5	0
Honda Brio	1	0
Suzuki Ertiga	1	0
Mitsubishi Xpander	0	1
Toyota Avanza	2	1
Toyota Innova	3	0

dapat dicari nilai TP, FP, TN, FN dari pengujian satu menit video rekaman cctv. Berikut persamaan dan hasil perhitungan untuk nilai-nilai tersebut.

$$\text{True Positive} = 5 + 1 + 3 + 2 + 1 = 12 \quad (5)$$

$$\text{False Positive} = 1 + 1 + 1 = 3 \quad (6)$$

$$\text{False Negative} = 0 \quad (7)$$

$$\text{precision} = \frac{TP}{TP + FP} = \frac{12}{12 + 3} = 0,80 \quad (8)$$

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{12}{12 + 0} = 1,00 \quad (9)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} = \frac{12 + 10}{12 + 5 + 0 + 10} = 0,82 \quad (10)$$

Dari hasil pengujian 1 menit, didapat nilai *True Positive* sebanyak 12, *False Positive* 3, *False Negative* 0, *True Negative* 10, *Precision* 0,80, *Recall* 1,00, dan akurasi 0,82 atau 82%.

#### 4. SIMPULAN

Dari Penelitian yang dilakukan, terdapat beberapa kesimpulan yaitu: Hasil dari sistem deteksi objek dengan metode YOLO pada kendaraan dengan kelas Motor, Honda Jazz, Honda Brio, Suzuki Ertiga, Mitsubishi Pajero Sport, Mitsubishi Xpander, Toyota Avanza, Toyota Innova sudah berjalan dengan baik dan dapat membantu pihak kepolisian untuk mendapatkan data jenis kendaraan yang melintas dengan menunjukkan presentase presisi sebesar 99,49%. Serta berdasarkan testing terhadap video rekaman CCTV selama 1 menit menunjukkan akurasi sebesar 82%. Hasil pelatihan pada setiap kelas menunjukkan hasil True Positive yang lebih besar dibanding dengan False Positive.

Untuk Penelitian selanjutnya memperbanyak dataset serta variasinya untuk meningkatkan akurasi. Variasi artinya memperbanyak class serta memperbanyak pengambilan angle dari citra yang akan diolah. Menggunakan video dengan kualitas tinggi untuk mendapatkan resolusi yang maksimal demi keakuratan akurasi. Menggunakan algoritma deteksi objek yang lain seperti SSD, atau YOLO versi terbaru. Hal ini bermaksud agar dengan digunakannya algoritma lain, dapat mengetahui perbandingan yang cukup signifikan.

#### DAFTAR PUSTAKA

- [1] Samsir and J. H. P. Sitorus, "**Perancangan Sistem Monitoring Lokasi Kendaraan Menggunakan GPS U-Blox Berbasis Android,**" *J. Bisantara Inform.*, vol. 5, no. 1, pp. 1–10, 2021.
- [2] M. Rusmin, "**Analisis Kinerja Di Simpang Empat Tak Sebidang Kota Makassar Berbasis Mikrosimulasi,**" vol. 1, 2018, [Online]. Available: <http://dx.doi.org/10.31227/osf.io/sbv73>
- [3] A. A. B, A. Amin, and M. W. Kasrani, "**Penerapan Metode Yolo Object Detection V1 Terhadap Proses Pendeteksian Jenis Kendaraan Di Parkiran,**" *J. Tek. Elektro Uniba (JTE UNIBA)*, vol. 6, no. 1, pp. 194–199, 2021, doi: 10.36277/jteuniba.v6i1.130.
- [4] N. Hanum Harani, C. Prianto, and M. Hasanah, "**Deteksi Objek Dan Pengenalan Karakter Plat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Python,**" *J. Tek. Inform.*, vol. 11, no. 3, pp. 47–53, 2019.
- [5] F. Rachmawati and D. Widhyaestoeti, "**Deteksi Jumlah Kendaraan di Jalur SSA Kota Bogor Menggunakan Algoritma Deep Learning YOLO,**" pp. 360–370.
- [6] M. H. Yusfina, C. Setianingsih, and R. Astuti, "**Deteksi Pelanggaran Parkir Pada Bahu Jalan Tol Dengan Intelligent Transportation System Menggunakan Algoritma Yolo ( Parking Violation Detection On The Roadside Of Toll Roads With Intelligent Transportation System Using Yolo Algorithm ),**" *e-Proceeding Eng.*, vol. 9, no. 3, pp. 1064–1069, 2022.
- [7] R. M. Taufiq, Sunanto, Y. Rizki, and M. R. A. Pratama, "**Simulasi Deteksi Golongan Kendaraan pada Gerbang Tol Menggunakan YOLOv4,**" *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 3, no. 2, pp. 199–206, 2022, doi: 10.37859/coscitech.v3i2.3928.
- [8] J. Zhao *et al.*, "**Improved Vision-Based Vehicle Detection and Classification by Optimized YOLOv4,**" *IEEE Access*, vol. 10, no. i, pp. 8590–8603, 2022, doi: 10.1109/ACCESS.2022.3143365.
- [9] M. I. HermaTelkomwan *et al.*, "**Kendaraan Menggunakan Metode Yolo Traffic Light Control Based on Vehicle Density Using the Yolo,**" vol. 8, no. 1, pp. 198–205, 2021.
- [10] Y. Zhang, Z. Guo, J. Wu, Y. Tian, H. Tang, and X. Guo, "**Real-Time Vehicle Detection Based on Improved YOLO v5,**" *Sustain.*, vol. 14, no. 19, 2022, doi: 10.3390/su141912274.