

Sistem Deteksi Jenis Kendaraan Metode YOLOv4 Untuk Mendukung Transportasi Cerdas Kota Medan

Vehicle Type Detection System Yolov4 Method to Support Smart Transportation in Medan City

M. Rizky Pramana^{1*}, Haida Dafitri², Sumi Khairani³

^{1,2,3}Program Studi Teknik Informatika, Universitas Harapan Medan

E-mail: ^{1*}mrpp894@gmail.com

Abstrak

Penelitian ini membahas evaluasi dan penerapan Model YOLOv4 dalam mendeteksi dan melacak jenis kendaraan dalam konteks lalu lintas jalan raya. Dalam rangka menjawab rumusan masalah, penelitian ini dilakukan dengan menguji performa model dalam beberapa aspek. Hasil penelitian menunjukkan bahwa Model YOLOv4 berhasil mencapai nilai Mean Average Precision (mAP) sebesar 77.88% pada dataset pelatihan setelah 7000 iterasi. Model ini memiliki kemampuan yang baik dalam mendeteksi berbagai objek jenis kendaraan dalam gambar, dengan akurasi yang bervariasi pada masing-masing kelas. Aplikasi yang dikembangkan dalam penelitian ini dapat merekam data hasil deteksi pada setiap frame dalam urutan video, memberikan informasi penting untuk analisis tingkat kepadatan kendaraan pada jalan. Meskipun memiliki tingkat akurasi yang cukup tinggi, masih terdapat kesalahan dalam deteksi dan pelabelan kelas objek. Kesimpulannya, penelitian ini memberikan wawasan tentang kemampuan dan potensi Model YOLOv4 dalam mengatasi tantangan deteksi kendaraan dalam lalu lintas jalan raya, sekaligus mengidentifikasi area yang perlu peningkatan lebih lanjut.

Kata kunci : YOLOv4, deteksi kendaraan, pelacakan, lalu lintas, akurasi

Abstract

This research discusses the evaluation and implementation of the YOLOv4 model in detecting and tracking vehicle types in the context of road traffic. To address the research questions, the study examined the model's performance across various aspects. The results indicate that the YOLOv4 model achieved a Mean Average Precision (mAP) of 77.88% on the training dataset after 7000 iterations. The model exhibits a commendable ability to detect different vehicle types within images, with varying accuracy rates across distinct classes. The developed application within this study can record detection data for every frame within a video sequence, providing crucial information for analyzing vehicle density on roads. Despite its relatively high accuracy level, errors persist in object detection and labeling. In conclusion, this research offers insights into the capabilities and potential of the YOLOv4 model in addressing challenges related to vehicle detection in road traffic, while also identifying areas that warrant further improvement.

Keyword : YOLOv4, vehicle detection, tracking, traffic, accuracy.

1. PENDAHULUAN

Intelligent Transportation System (ITS) merupakan salah satu solusi yang efektif untuk mengatasi masalah lalu lintas perkotaan, karena jumlah kendaraan yang terus meningkat pesat tetapi luas jalan tetap sama akan menyebabkan kemacetan.

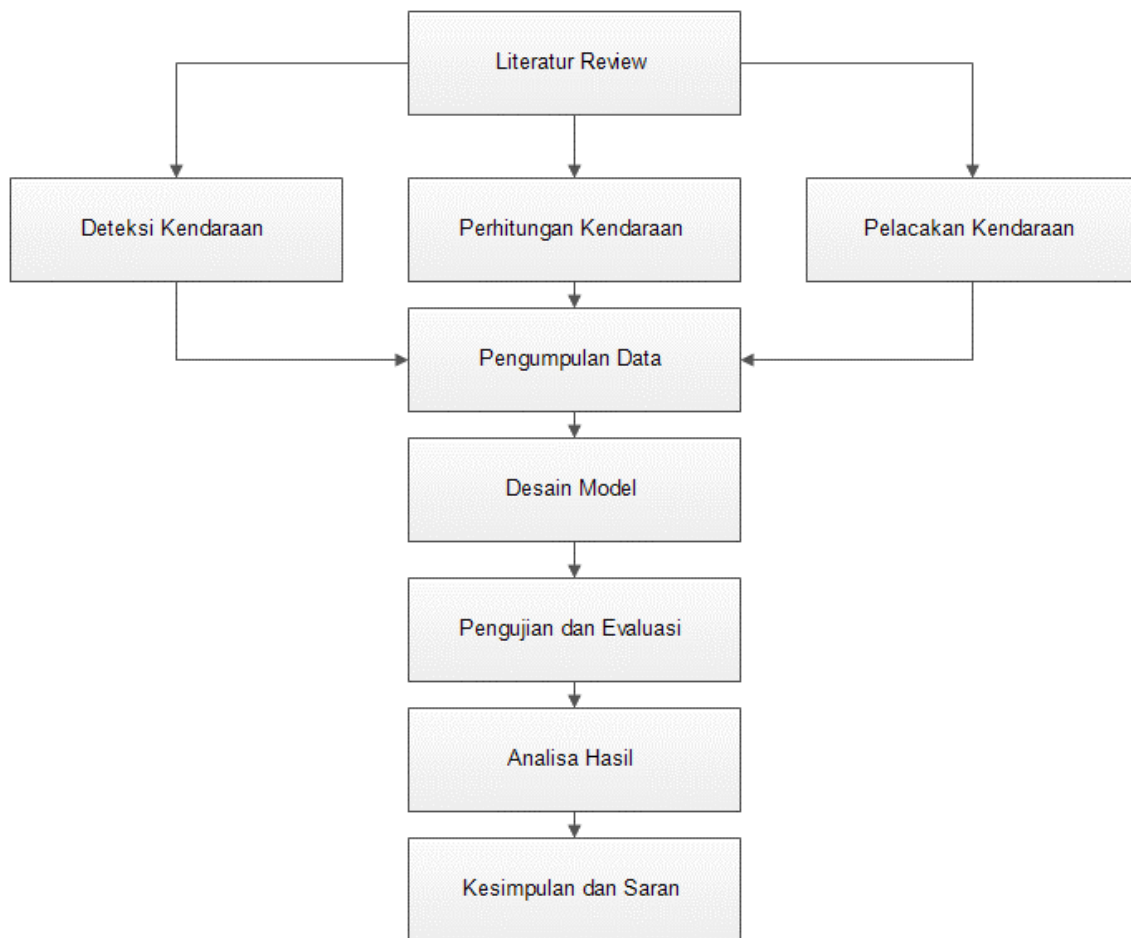
Menentukan kepadatan kendaraan di jalan dan analisis pengawasan video banyak membantu dalam sistem manajemen lalu lintas [1].. Namun, pemantauan lalu lintas masih mengandalkan operator manusia terhadap berbagai insiden lalu lintas di jaringan jalan raya, belum secara otomatis mendeteksi berbagai entitas yang berguna dalam pengambilan keputusan seperti deteksi kendaraan maupun terjadinya pelanggaran lalu lintas.

Kemajuan di dalam visi komputer berbasis pembelajaran mendalam dekade terakhir ini telah menjadi metode terdepan untuk deteksi objek umum berkualitas tinggi, termasuk deteksi kendaraan sebagai sistem pengawasan lalu lintas *real-time* dengan tujuan yang berbeda, seperti deteksi kendaraan [2]–[4], perhitungan kendaraan [5]–[7], pelacakan kendaraan [8], [9], klasifikasi kendaraan [10], [11] dan lainnya. Berbagai teknik telah diusulkan untuk sistem pengawasan lalu lintas cerdas, namun teknik pembelajaran mendalam lebih akurat dibandingkan dengan teknik tradisional [12] karena memiliki kemampuan dalam mempelajari fitur gambar atau video dan tugas klasifikasi maupun regresi [13]. Salah satu komponen yang sangat penting dari ITS adalah identifikasi dan deteksi kendaraan, yang bertujuan untuk mengumpulkan informasi tentang jenis kendaraan tertentu dari gambar atau video yang ada di dalam kendaraan. Pendekatan metode deteksi satu tahap seperti *Single Shot Multibox Detector* (SSD), *You Only Look Once* (YOLO) menerapkan jaringan konvolusi untuk secara langsung memprediksi kelas dan lokasi objek yang dilatih dari ujung ke ujung lebih cepat daripada metode dua tahap. Karya lain seperti RefinaDet [14], CenterNet [15], RetinaNet [16], EfficientNet [17], EfficientDet [18], FreeAnchor [19] mencapai trade-off yang lebih baik antara kecepatan dan akurasi.

Arsitektur jaringan YOLO merupakan salah model yang paling banyak menarik minat peneliti untuk membangun sistem pengawasan lalu lintas [20]–[23], beberapa peningkatan kinerja dan model baru diusulkan seperti LittleYOLO-SPP [12], Gaussian-YOLOv3 [24], MME-YOLO [25] dan SF-YOLO [26]. Lebih lanjut, dalam beberapa tahun terakhir, YOLO memiliki kinerja yang baik diterapkan dalam banyak bidang, seperti deteksi helm [27], deteksi harimau [28], deteksi opium [29], deteksi pejalan kaki [30], deteksi rambu lalu lintas [31], deteksi plat kendaraan [32] dan lainnya. Baru-baru ini, Alexey Bochkovskiy merilis arsitektur jaringan baru YOLOv4 [33] yang merupakan peningkatan dari YOLOv3 untuk mendeteksi objek di beberapa area deteksi dengan cepat dan akurat. YOLOv4 menggunakan *Cross-Stage-Partial-Connections* (CSP) sebagai tulang punggung baru yang dapat meningkatkan kemampuan pembelajaran CNN dan lebih cepat dibandingkan pendekatan satu tahap lainnya.

2. Metodologi Penelitian

Metodologi penelitian yang diusulkan dalam penelitian ini diilustrasikan seperti gambar 2.1 dibawah ini. Secara keseluruhan, penulis mengusulkan pendekatan visi komputer dengan teknik pembelajaran mendalam yang berbeda untuk memahami cara kerja serta menganalisis kinerja model yang paling optimal di terapkan.



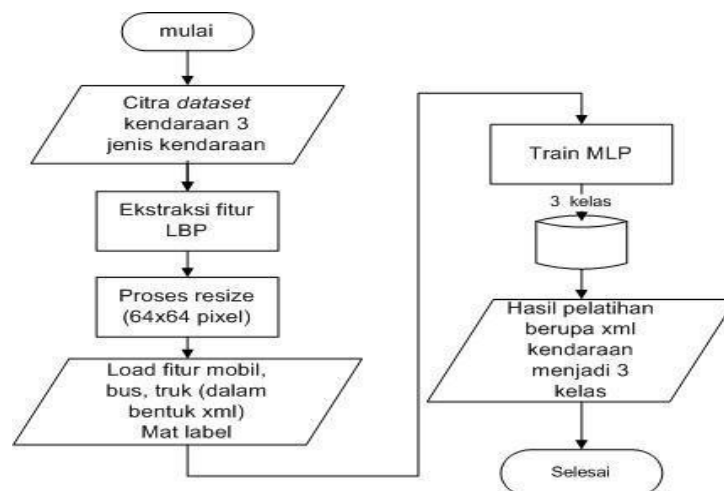
Gambar 2.1 Metodologi Penelitian

2.1 Metode yang diusulkan

Berdasarkan uraian literatur review pada bagian sebelumnya, ada banyak model dan teknik diusulkan dalam banyak literatur dengan hasil kinerja yang menjanjikan untuk diterapkan sebagai solusi sistem pengawasan lalu lintas cerdas. Tetapi sulit mengidentifikasi dan menerapkan metode yang optimal untuk diadaposi, hal disebabkan dalam banyak karya penelitian sebelumnya menggunakan tahapan yang berbeda, kumpulan data dan kondisi pengujian yang dilakukan berbeda. Oleh karena itu, dalam penelitian ini penulis melakukan identifikasi dan pengujian perbandingan teknik pembelajaran mendalam yang paling sering digunakan dan melaporkan hasil perbandingan hasil eksperimen pada dataset video CCTV jalan raya, kemudian seluruh model dikombinasikan dengan algoritma Deep Sort untuk pelacakan dan perhitungan kendaraan.

2.2 Gambaran Flowchart Sistem

Secara umum, gambaran system yang akan dibuat adalah seperti Gambar 2.2 berikut ini:



Gambar 2.2. Flowchart Klasifikasi Kendaraan

Penjelasan pada flowchart diatas adalah sebagai berikut:

1. Pengambilan Gambar

Adalah proses pengambilan gambar/video yang pada penelitian ini akan menggunakan stock video yang telah tersedia pada platfrom youtube.com pengambilan gambar/video juga dapat dilakukan menggunakan cctv lalulintas pada jalan raya.

2. Preprocessing

Kemudian pada tahapan ini, merupakan proses konvolusi, yaitu perkalian citra dengan sebuah mask atau kernel, kemudian akan menentukan jenis fitur yang dinilai sesuai dengan kualifikasi citra.

3. Ekstrasi Fitur

Merupakan penentuan fitur citra yang telah dilakukan konvolusi. Lalu dilakukan normalisasi citra kemudian akan ditentukan jenis fitur mana yang sesuai dengan citra input.

4. Pengenalan Citra

Pada tahap ini akan menampilkan hasil citra yang telah selesai diekstraksi.

2.3 Labelling

Dalam tahap pertama, data citra yang telah diunduh kemudian akan dilakukan tahap *labelling* yang nantinya berguna sebagai penanda pada system program yang dijalankan / *training*. Dalam prosesnya, *labelling* menggunakan *librabry Labelling* untuk memproses data. *Labelling* dilakukan secara manual atau satu persatu untuk menandakan setiap *label* citra terhadap kelasnya masing-masing. Tahap pertama saat melakukan *labelling*, yaitu dengan mengunduh *tools "labelling"* pada *anaconda prompt*.

2.4 Split Dataset

Tahap *split dataset* berguna untuk membagi jumlah *dataset* yang diperlukan sesuai dengan jumlah citra yang tersedia. Pada kasus ini *dataset* yang tersedia akan dikelompokkan menjadi tiga, yaitu sebagai *train*, *val*, dan *test*.

2.5 Train

Train/test split adalah salah satu metode yang dapat digunakan untuk mengevaluasi performa model machine learning. Metode evaluasi model ini membagi dataset menjadi dua bagian yakni bagian yang digunakan untuk training data dan untuk testing data dengan proporsi tertentu.

2.6 Pembagian Class

Pembagian *class* dilakukan agar program yang dijalankan nantinya dapat mengetahui jumlah *class* yang akan dieksekusi. Berikut merupakan kode program yang diperlukan.

```
train : data/dataset/images/train
```

```
val : data/dataset/images/val
```

```
test : data/dataset/images/test
```

```
# number of classes
```

```
nc : 5
```

```
# class name
```

```
names : [ 'sedan', 'sepeda motor', 'bus', 'minibus', 'truck' ]
```

Dimana, nc: 5 merupakan jumlah class names yang tersedia.

2.7 Himpunan Dataset

Pada bagian ini dijelaskan himpunan data yang digunakan pada penelitian. Sumber dataset dari Dinas Perhubungan Kota Medan, yaitu data video hasil rekaman CCTV dari 5 (lima) persimpangan di jalan Nasional Kota Medan. Secara keseluruhan video berjumlah 3.114 video yang dibagi menjadi 5 (lima) kelompok yaitu Pagi, Siang, Sore, Malam dan Dini Hari, kemudian video dibagi kembali sesuai kondisi cuaca hujan, gerimis. Pembagian himpunan dataset yang digunakan dilaporkan pada tabel 3.1.

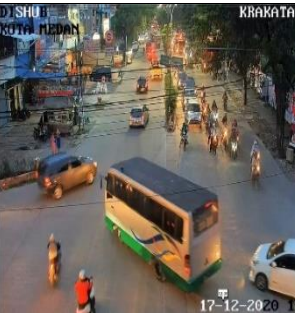
Tabel 2.1. Himpunan Dataset

No.	Nama Jalan	Jumlah Video					Total
		Pagi	Siang	Sore	Malam	Dini hari	
1	Simpang Krakatau-Bilal	138	76	63	107	100	484
2	Simpang Lapangan Merdeka	188	127	95	175	148	733

3	Simpang Glugur	155	91	121	133	109	609
4	Simpang Istana Plaza	185	92	98	148	133	656
5	Simpang Lonsum	173	100	81	147	131	632
TOTAL VIDEO							3114

Pada tabel 1 merupakan pembagian himpunan dataset yang digunakan. Hasil video CCTV merupakan kumpulan video selama seminggu untuk kelima simpang. Contoh sebahagian dataset ditunjukkan pada tabel 2.

Tabel 2.2 Contoh Sebahagian data rekaman video CCTV

No	Nama Jalan	Waktu		
		Pagi	Siang	Sore
1	Simpang Krakatau -Bilal			
2	Simpang Lapangan Merdeka			
3	Simpang Glugur			

2.8 Arsitektur Model yang diusulkan

Arsitektur model yang diusulkan pada penelitian didasarkan pada teknik pembelajaran mendalam dengan menerapkan 4 (empat) model YOLOv4 untuk klasifikasi dan deteksi kendaraan. Semua model akan dianalisa kinerja untuk

deteksi kendaraan dengan tujuan mendapatkan model yang paling optimal diterapkan saat implementasi sistem pengawasan lalu lintas di Kota Medan. Lebih jelas model yang diusulkan dilihat pada gambar 3.



Gambar 2.3. Arsitektur Model yang diusulkan

3. HASIL DAN PEMBAHASAN

3.1 Performa Klasifikasi Jenis Kendaraan menggunakan YOLOV4

Berikut merupakan hasil dari perbuatan program pendeteksian objek kendaraan menggunakan YOLOV4. Pada bagian ini, peneliti akan melakukan evaluasi terhadap kinerja model klasifikasi menggunakan metrik-metrik evaluasi standar seperti akurasi, presisi, recall, dan *FI-score*.

3.2 Pengujian Sistem

Pada bagian ini kami akan uraikan hasil pelatihan dan pengujian deteksi jenis kendaraan di Jalan Raya Kota Medan berdasarkan video dengan menggunakan YOLOv4. Semua eksperimen pelatihan menggunakan *Google Colab* yang merupakan salah satu *notebook* Python Jupyter yang berjalan pada, sedangkan pengujian deteksi objek langsung pada video menggunakan spesifikasi *Intel Core i5-7300HQ quad-core 2,5GHz TurboBoost* hingga 3,5GHz, RAM 8GB, GPU *Nvidia GeForce GTX 1050 4 GB* berjalan di *Windows 10*. Proses pengujian dilakukan dengan tahapan persiapan data, pelabelan *class* jenis kendaraan, pelatihan dan pengujian.

3.3 Persiapan Data

Proses pengumpulan dataset bersumber dari Dinas Perhubungan Kota Medan yang secara khusus dikumpulkan rekaman aktifitas di jalan Juanda Katamso, Gatot Subroto dan Uniland dari tanggal 21 April 2020 sampai tanggal 27 April 2020 kapasitas 247 GB dengan total video 1.357 untuk ketiga simpang. Kemudian di lakukan beberapa pemotongan pada masing-masing video dengan mengabaikan kondisi rekaman lampu merah dan hanya mengambil data dengan kondisi kendaraan ramai. Hasil perubahan data video dibagi tiap hari untuk tiap simpang sehingga di masing-masing simpang terdapat 7 video dengan total keseluruhan sebanyak 21 video. Rincian dataset ditunjukkan pada tabel 3.1 dan seluruh video memiliki dimensi 1280x720 dengan 25 *frame* per detik (fps).

Tabel 3.1. Dataset

Tanggal	Jumlah Video		
	Juanda Katamso	Gatot Subroto	Uniland
21/04/2020	45	38	94

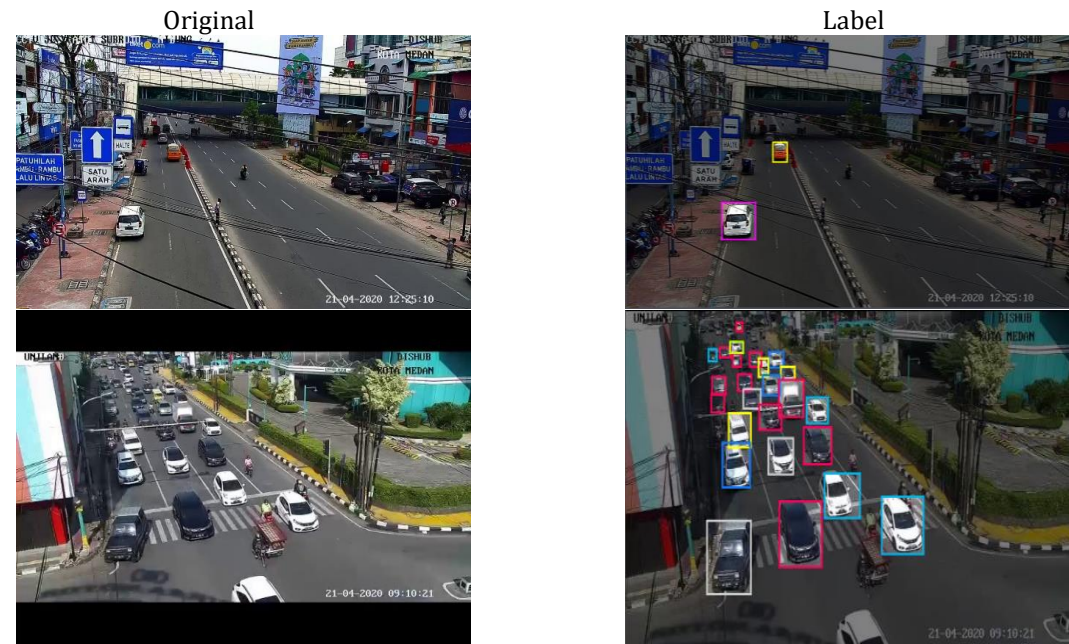
22/04/2020	58	10	85
23/04/2020	58	45	86
24/04/2020	79	81	83
25/04/2020	79	79	79
26/04/2020	79	85	85
27/04/2020	43	38	38
Total	441	376	550

3.4 Pelabelan Class

Dalam proses pembuatan label dilakukan ekstraksi gambar dari video dengan maksimum *frame* gambar 250 dari 3 simpang dimana 1 simpang 2 video diekstrak dengan total gambar yang digunakan untuk *labelling* ketiga simpang sebanyak 800 gambar dengan rincian dilihat pada tabel 3.2.

Tabel 3.2. Dataset *Labelling*

Nama Simpang	Source Image		Train/Test Split		
	Images	Classes	Train	Valid	Test
Juanda Katamso	335	13	231 images	69 images	35 images
Gatot Subroto	237	13	165 images	49 images	23 images
Uniland	302	13	213 images	30 images	33 images





Gambar 3.1. Proses *Labelling*

Dari hasil label yang telah dilakukan, maka secara keseluruhan jumlah *class* yang digunakan untuk deteksi jenis kendaraan ini berjumlah 13 *class* dengan rincian dilihat pada tabel 3.3.

Tabel 3.3 *Class* Jenis Kendaraan

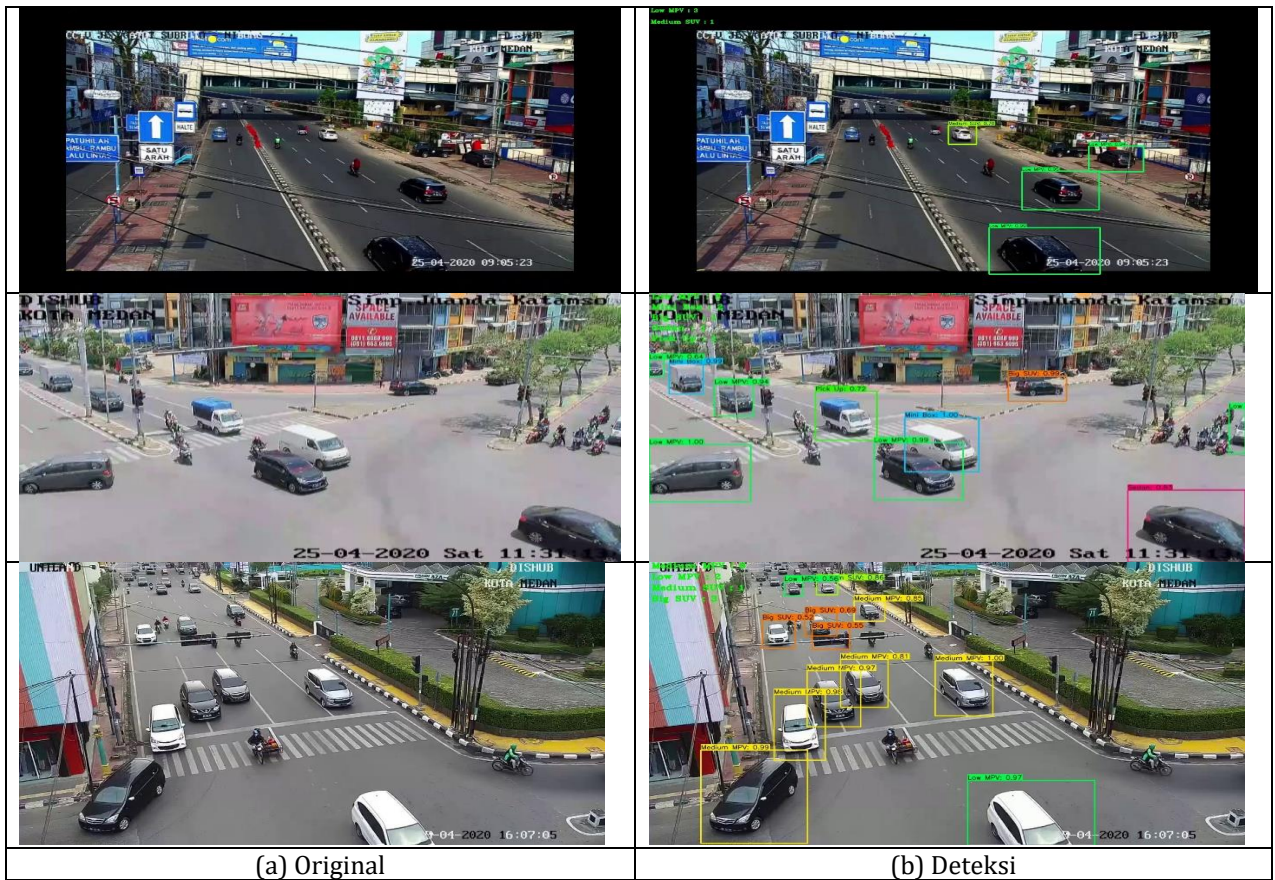
Class ID	Class Name
0	Big SUV
1	Bus
2	City Car
3	Low MPV
4	Medium Box
5	Medium MPV
6	Medium SUV
7	Mini Box
8	Minibus
9	Pick Up
10	Premium MPV
11	Sedan
12	Truk

3.5 Pengujian Video CCTV

Berdasarkan hasil pelatihan nilai mAP tertinggi berada pada iterasi ke 7000, maka dalam pengujian akan diterapkan langsung untuk mendeteksi video. Proses pengujian dilakukan dengan membagi semua video menjadi satu video untuk tiap hari. Semua video merupakan gabungan waktu pagi, siang dan sore hal ini untuk mengoptimalkan waktu pengujian, sehingga total keseluruhan video yang diuji berjumlah 23 video, 21 video dari masing-masing simpang tiap hari, 1 video baru dari rekaman aktifitas kendaraan di jalan manhantan, terakhir 1 video malam dengan kondisi hujan di Jalan Juanda Katamso. Hasil deteksi model YOLOv4 akan disimpan pada *file tracking.csv* dan *file detec.csv* dari tiap simpang.

File tracking.csv akan menyimpan data *tracking* semua kendaraan tiap *frame* dan *file detec.csv* menyimpan informasi jumlah *class* kendaraan yang terdeteksi pada tiap *frame*. Secara keseluruhan video yang di uji adalah gabung waktu pagi, siang

dan sore. Hasil deteksi jenis kendaraan pada masing-masing simpang dapat dilihat pada gambar 4.3 dan gambar 4.4 merupakan video baru dari kondisi yang berbeda.



Gambar 3.2. Video Kondisi Berbeda



Gambar 3.3. Video Kondisi Berbeda Diluar Dataset

Tabel 3.4. Hasil Deteksi di Simpang Juanda Katamso

TGL.	Class Kendaraan												
	0	1	2	3	4	5	6	7	8	9	10	11	12
21/0 4/20 20	1.64 7	-	4.35 4	13. 65 9	67 8	3.8 94	4.2 57	233	832	1.02 3	35	102	5
22/0 4/20 20	2.06 4	-	3.42 9	17. 05 7	37 8	7.4 80	3.0 55	587	974	2.51 5	212	216	221
23/0 4/20 20	4.09 9	76	5.93 2	20. 03 3	2.2 66	10. 54 8	6.2 30	2.16 7	2.29 4	3.21 8	104	258	446
24/0 4/20 20	1.57 2	37	3.62 6	16. 35 4	44 6	3.1 00	3.4 85	1.08 6	701	2.19 8	34	166	327
25/0 4/20 20	3.88 8	-	7.45 6	27. 86 3	80 4	7.2 77	5.9 94	2.37 7	2.33 6	2.85 3	176	435	482
26/0 4/20 20	3.04 1	-	4.21 9	14. 94 9	14 4	5.4 51	2.6 48	255	585	1.00 3	92	114	-
27/0 4/20 20	1.03 1	-	1.82 1	10. 37 8	92 6	1.7 66	1.1 66	229	1.03 1	986	18	185	401
22 & 26/0 4/20 20	4.42 2	-	2.08 1	8.7 84	1.5 79	8.9 59	2.3 86	1.96 9	1.01 3	2.70 1	33	11	911

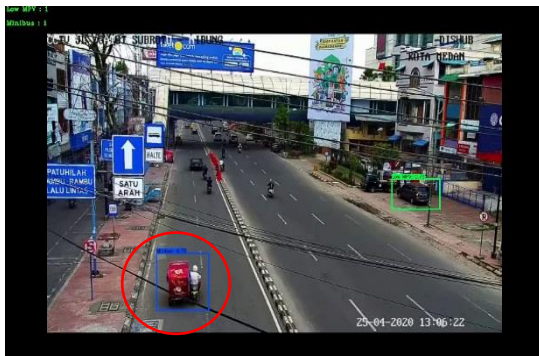
Berdasarkan hasil pengujian deteksi jenis kendaraan pada 23 video, model YOLOv4 yang diusulkan dapat mendeteksi secara akurat jenis kendaraan walaupun ada beberapa proses dimana model ini menghasilkan deteksi yang tinggi dengan nama *class* yang salah dan objek terdeteksi dengan kotak yang salah, lebih jelasnya dapat dilihat pada gambar 4.



Mendeteksi motor box menjadi minibus



Mendeteksi becak menjadi medium box



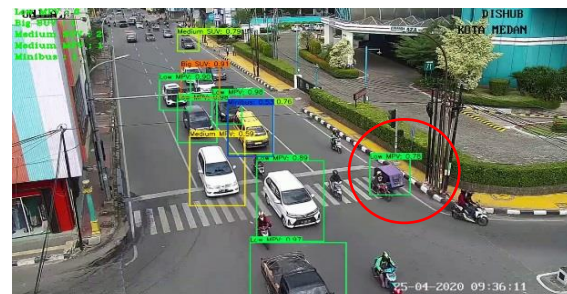
Mendeteksi becak menjadi minibus



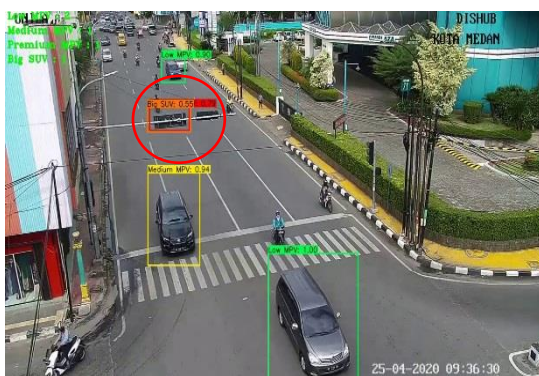
Mendeteksi becak menjadi medium MPV



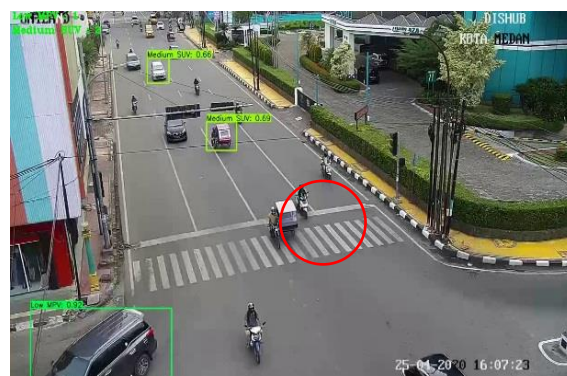
Mendeteksi area kosong menjadi medium Box



Mendeteksi becak menjadi low MPV



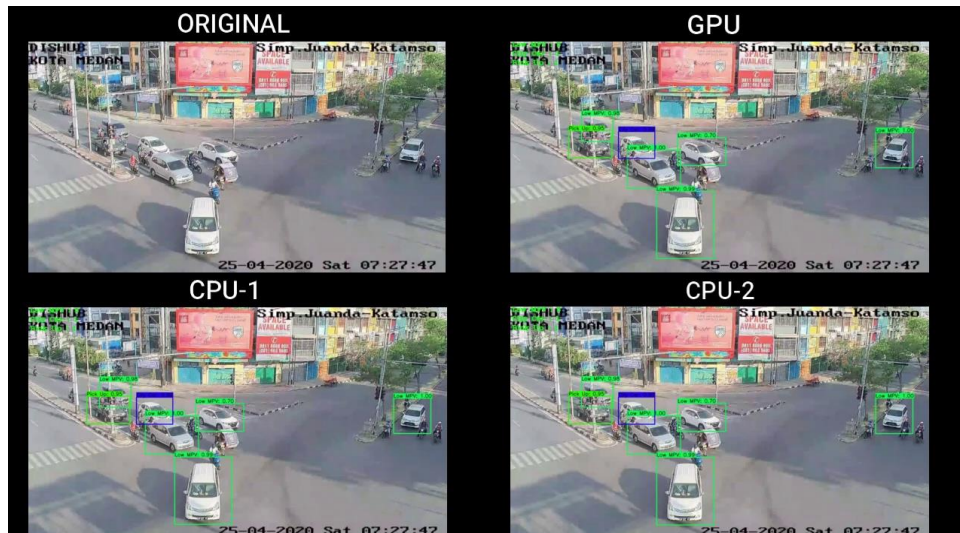
Mendeteksi lampu lalu lintas menjadi Big SUV



Mendeteksi becak menjadi Medium SUV

Gambar 3.4. Proses Labelling

Pada gambar 4 dapat dilihat terjadi kesalahan deteksi dengan jenis kesalahan yang berbeda, hal ini terjadi karena proses pelabelan data yang tidak konsisten khususnya dalam membuat kotak yang menjadi penanda *class* kendaraan, selain itu jarak kamera dengan objek yang di deteksi memiliki pengaruh terhadap hasil *class* kendaraan walaupun nilai akurasi deteksi cukup tinggi. Dari beberapa kesalahan deteksi akan menjadi bahan evaluasi untuk penelitian selanjutnya. Terakhir pengujian dilakukan perbandingan deteksi menggunakan GPU, CPU 1 dengan spesifikasi Core I3 gen8 HDD dan CPU 2 Core I3 gen7 SSD seperti ditunjukkan pada gambar 5.



Gambar 3.5. Perbandingan GPU, CPU 1 dan CPU 2

Pada tabel 5 dapat dilihat bahwa perbedaan waktu testing pada GPU adalah 9 menit 23 detik, sedangkan pada CPU 1 adalah 2 jam 4 menit 30 detik dan CPU 2 adalah 2 jam 31 menit 1 detik. Dari hasil perbandingan ini untuk pengolahan data *Computer Vision* sebaiknya menggunakan spesifikasi GPU.

Tabel 3.5. Perbandingan waktu GPU, CPU 1 dan CPU 2

GPU	CPU 1	CPU 2
00:09:23	02:04:30	02:31:01

3.6 Eksperimen dan Pengujian

Pada bagian ini penulis melakukan pengujian 4 model teknik pembelajaran yang diusulkan untuk memahami cara kerja dan evaluasi hasil model yang paling optimal diterapkan untuk deteksi, pelacakan dan perhitungan kendaraan.

4. KESIMPULAN

Berdasarkan hasil pengujian yang dilakukan pada 23 video CCTV aktifitas kendaraan di jalan raya Kota Medan, secara umum model YOLOv4 yang diusulkan dapat mendeteksi secara langsung dengan akurat jenis kendaraan dari 13 class yang

digunakan. Dalam hal ini, maka dapat ditarik beberapa kesimpulan, yaitu:

1. Model YOLOv4 menghasilkan nilai mAP 77.88 % pada dataset pelatihan pada iterasi ke 7000
2. Model YOLOv4 dapat mendeteksi secara langsung beberapa objek jenis kendaraan pada file gambar dengan tingkat akurasi terendah 21.23% pada class Bus, tertinggi 98.24% pada class Truk dan rata-rata akurasi adalah 70.75%
3. Aplikasi yang di kembangkan dapat mencatat semua data hasil deteksi dalam urutan video setiap frame nya kemudian disimpan ke file yang berguna untuk memprediksi tingkat kepadatan kendaraan
4. Masih terdapat beberapa kesalahan dalam deteksi dan nama class walaupun nilai akurasi deteksi cukup tinggi
5. Model YOLOv4 dapat dikombinasikan dengan algoritma pelacakan seperti SORT, Deep SORT, IOU untuk lebih akurat mendeteksi dan mengoptimal waktu deteksi sehingga tidak terjadi deteksi yang berulang-ulang

DAFTAR PUSTAKA

- [1] J. Sang *et al.*, "An improved YOLOv2 for vehicle detection," *Sensors (Switzerland)*, vol. 18, no. 12, 2018, doi: 10.3390/s18124272.
- [2] Q. Meng, H. Song, Y. Zhang, X. Zhang, G. Li, and Y. Yang, "Video-Based Vehicle Counting for Expressway: A Novel Approach Based on Vehicle Detection and Correlation-Matched Tracking Using Image Data from PTZ Cameras," *Math. Probl. Eng.*, vol. 2020, 2020, doi: 10.1155/2020/1969408.
- [3] J. B. Kim, "Efficient vehicle detection and distance estimation based on aggregated channel features and inverse perspective mapping from a single camera," *Symmetry (Basel)*, vol. 11, no. 10, pp. 8–10, 2019, doi: 10.3390/sym11101205.
- [4] S. C. Hsu, C. L. Huang, and C. H. Chuang, "Vehicle detection using simplified fast R-CNN," *2018 Int. Work. Adv. Image Technol. IWAIT 2018*, pp. 1–3, 2018, doi: 10.1109/IWAIT.2018.8369767.
- [5] C. S. Asha and A. V. Narasimhadhan, "Vehicle Counting for Traffic Management System using YOLO and Correlation Filter," in *2018 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, Mar. 2018, pp. 1–6. doi: 10.1109/CONECCT.2018.8482380.
- [6] G. Oltean, C. Florea, R. Orghidan, and V. Oltean, "Towards Real Time Vehicle Counting using YOLO-Tiny and Fast Motion Estimation," *SIITME 2019 - 2019 IEEE 25th Int. Symp. Des. Technol. Electron. Packag. Proc.*, pp. 240–243, 2019, doi: 10.1109/SIITME47687.2019.8990708.
- [7] K. H. Nam Bui, H. Yi, and J. Cho, "A multi-class multi-movement vehicle counting framework for traffic analysis in complex areas using CCTV systems," *Energies*, vol. 13, no. 8, 2020, doi: 10.3390/en13082036.
- [8] Z. He, Y. Lei, S. Bai, and W. Wu, "Multi-Camera Vehicle Tracking with Powerful Visual Features and Spatial-Temporal Cue," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, p. 10, 2019.

-
- [9] X. F. Gu, Z. W. Chen, T. S. Ma, F. Li, and L. Yan, "Real-Time vehicle detection and tracking using deep neural networks," *2016 13th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process. ICCWAMTIP 2017*, pp. 167–170, 2017, doi: 10.1109/ICCWAMTIP.2016.8079830.
 - [10] R. Watkins, N. Pears, and S. Manandhar, "Vehicle classification using ResNets, localisation and spatially-weighted pooling," *arXiv*, 2018.
 - [11] A. Schumann, L. Sommer, K. Valev, and J. Beyerer, "A systematic evaluation of recent deep learning architectures for fine-grained vehicle classification," in *Pattern Recognition and Tracking XXIX*, Apr. 2018, p. 1. doi: 10.1117/12.2305062.
 - [12] X. Li, Y. Liu, Z. Zhao, Y. Zhang, and L. He, "A deep learning approach of vehicle multitarget detection from traffic video," *J. Adv. Transp.*, vol. 2018, 2018, doi: 10.1155/2018/7075814.
 - [13] H. Song, H. Liang, H. Li, Z. Dai, and X. Yun, "Vision-based vehicle detection and counting system using deep learning in highway scenes," *Eur. Transp. Res. Rev.*, vol. 11, no. 1, 2019, doi: 10.1186/s12544-019-0390-4.
 - [14] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-Shot Refinement Neural Network for Object Detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4203–4212. doi: 10.1109/CVPR.2018.00442.
 - [15] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 6568–6577, 2019, doi: 10.1109/ICCV.2019.00667.
 - [16] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020, doi: 10.1109/TPAMI.2018.2858826.
 - [17] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.
 - [18] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Nov. 2020, pp. 10778–10787. doi: 10.1109/CVPR42600.2020.01079.
 - [19] X. Zhang, F. Wan, C. Liu, X. Ji, and Q. Ye, "Learning to Match Anchors for Visual Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, Sep. 2021, doi: 10.1109/TPAMI.2021.3050494.
 - [20] P. Mahto, P. Garg, P. Seth, and J. Panda, "Refining Yolov4 for vehicle detection," *Int. J. Adv. Res. Eng. Technol.*, vol. 11, no. 5, pp. 409–419, 2020, doi: 10.34218/IJARET.11.5.2020.043.
 - [21] Y. Q. Huang, J. C. Zheng, S. D. Sun, C. F. Yang, and J. Liu, "Optimized YOLOv3 algorithm and its application in traffic flow detections," *Appl. Sci.*, vol. 10, no. 9, p. 3079, Apr. 2020, doi: 10.3390/app10093079.
 - [22] L. Ouyang and H. Wang, "Vehicle target detection in complex scenes based on YOLOv3 algorithm," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 569, no. 5, 2019, doi:

- 10.1088/1757-899X/569/5/052018.
- [23] X. Wang, S. Wang, J. Cao, and Y. Wang, "Data-Driven Based Tiny-YOLOv3 Method for Front Vehicle Detection Inducing SPP-Net," *IEEE Access*, vol. 8, pp. 110227–110236, 2020, doi: 10.1109/ACCESS.2020.3001279.
- [24] J. Choi, D. Chun, H. Kim, and H. J. Lee, "Gaussian YOLOv3: An accurate and fast object detector using localization uncertainty for autonomous driving," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 502–511, 2019, doi: 10.1109/ICCV.2019.00059.
- [25] J. Zhu, X. Li, P. Jin, Q. Xu, Z. Sun, and X. Song, "MME-YOLO: Multi-sensor multi-level enhanced yolo for robust vehicle detection in traffic surveillance," *Sensors (Switzerland)*, vol. 21, no. 1, pp. 1–17, Dec. 2021, doi: 10.3390/s21010027.
- [26] B. G. Han, J. G. Lee, K. T. Lim, and D. H. Choi, "Design of a scalable and fast yolo for edge-computing devices," *Sensors (Switzerland)*, vol. 20, no. 23, pp. 1–15, Nov. 2020, doi: 10.3390/s20236779.
- [27] F. Wu, G. Jin, M. Gao, Z. He, and Y. Yang, "Helmet detection based on improved YOLO V3 deep model," *Proc. 2019 IEEE 16th Int. Conf. Networking, Sens. Control. ICNSC 2019*, pp. 363–368, 2019, doi: 10.1109/ICNSC.2019.8743246.
- [28] R. Wei, N. He, and K. Lu, "YOLO-mini-tiger: Amur tiger detection," in *ICMR 2020 - Proceedings of the 2020 International Conference on Multimedia Retrieval*, 2020, pp. 517–524. doi: 10.1145/3372278.3390710.
- [29] J. Zhou, Y. Tian, C. Yuan, K. Yin, G. Yang, and M. Wen, "Improved UAV opium poppy detection using an updated YOLOV3 model," *Sensors (Switzerland)*, vol. 19, no. 22, pp. 1–23, 2019, doi: 10.3390/s19224851.
- [30] Y. Lai, F. Sun, and H. Liu, "Small Object Detection Base on YOLOv3 for Pedestrian Recognition," in *2020 5th International Conference on Control and Robotics Engineering, ICCRE 2020*, 2020, pp. 235–241. doi: 10.1109/ICCRE49379.2020.9096492.
- [31] Á. Arcos-García, J. A. Álvarez-García, and L. M. Soria-Morillo, "Evaluation of deep neural networks for traffic sign detection systems," *Neurocomputing*, vol. 316, pp. 332–344, 2018, doi: 10.1016/j.neucom.2018.08.009.
- [32] R. Laroca, L. A. Zanlorensi, G. R. Gonçalves, E. Todt, W. R. Schwartz, and D. Menotti, "An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO detector," *arXiv*, Sep. 2019.
- [33] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv*. 2020.