



## SISTEM DETEKSI DAN ESTIMASI DIMENSI MENGGUNAKAN METODE *DEEP LEARNING* DAN *ADAPTIVE ROI*

Arfian Iqbal Fatkhulhuda<sup>1\*</sup>, Sulfan Bagus Setyawan<sup>2\*</sup>, Hanum Arrosida<sup>3</sup>, Denny  
Hardiyanto<sup>4</sup>, Aulia El Hakim<sup>5</sup>, Nadia Ilmiya Maulinda<sup>6</sup>

<sup>1,2,3,4,5,6</sup> Politeknik Negeri Madiun, Indonesia

\*Email Responden: [sulfan@pnm.ac.id](mailto:sulfan@pnm.ac.id)

(Artikel diterima: bulan dan tahun pengumpulan jurnal, direvisi: bulan dan tahun jurnal terbit)

### ABSTRAK

Kerusakan jalan merupakan salah satu faktor penyebab kecelakaan lalu lintas. Berdasarkan data Dit Lantas Polda Metro Jaya, tercatat 40 kasus kecelakaan, dengan rincian 12 kasus disebabkan oleh jalan rusak, 15 kasus akibat jalan berlubang, 11 kasus karena jalan licin, dan 2 kasus terkait tidak adanya rambu serta tikungan tajam. Selain menyebabkan kecelakaan, kondisi jalan yang rusak juga berdampak negatif terhadap perekonomian masyarakat dan memperlambat akses transportasi, sehingga mengurangi kenyamanan pengguna jalan. Untuk mengatasi permasalahan ini dan menekan angka kecelakaan akibat kerusakan jalan, diperlukan upaya inovatif. Salah satu solusinya adalah pemanfaatan *Tensorflow* untuk mendeteksi kerusakan jalan secara *realtime*, yang memungkinkan pengiriman data kerusakan beserta lokasinya. Namun, teknologi ini belum mampu memperkirakan dimensi kerusakan. Oleh karena itu, penulis mengusulkan "Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive RoI*". Sistem ini menggunakan kamera untuk menangkap gambar kerusakan jalan, yang kemudian diproses oleh *Mini PC*. Data diproses dengan *Framework PyTorch Object Detection* dengan *YOLOv5s*. Setelah mengidentifikasi jenis kerusakan jalan, *Area Estimation System* akan digunakan untuk melakukan proses estimasi area pada kerusakan jalan. Selanjutnya, data akan dikirim ke *cloud server*. Dari hasil pengujian seluruh sistem yang telah dilakukan, sistem deteksi dapat melakukan pengujian seluruh sistem secara *realtime* dan mengidentifikasi jenis kerusakan seperti retak memanjang, retak melintang, retak kulit buaya, alur bekas roda, dan lubang presentasi dengan nilai *Accuracy* sebesar 83,07%, *Precision* sebesar 0,92, *Recall* sebesar 0,89, dan *F1-Score* sebesar 90,4%. Sistem juga dapat menghitung estimasi area dengan kesalahan 24,6 cm pada lebar, 27,6 cm pada Panjang dan 4,82 mm pada kedalaman.

**Kata kunci:** Jalan Rusak, *Deep Learning*, *Computer Vision*, *Training*, *Realtime*.

### I. PENDAHULUAN

Kecelakaan lalu lintas merupakan salah satu penyebab utama tingginya angka kematian di Indonesia. Tingkat fatalitas akibat kecelakaan lalu lintas tergolong tinggi, di antara korban kecelakaan, terdapat sejumlah anak sekolah yang turut menjadi korban [1]. Terdapat sekitar 1,19 juta kasus kematian serta 20 sampai 50 juta korban cedera bahkan ada yang sampai menderita kecacatan yang diakibatkan oleh kecelakaan lalu lintas setiap tahunnya [2]. Data dari Pusiknas Bareskrim Polri pada periode Januari-November 2023 terdapat 134.867 kasus kecelakaan dengan korban jiwa mencapai 19.825 orang dan kerugian diperkirakan mencapai 258,18 miliar rupiah [3]. Data dari Dit Lantas Polda Metro Jaya juga membuktikan terjadi sebanyak 40 kasus dengan rincian 12 kasus akibat jalanan rusak, 15 kasus akibat jalanan berlubang, 11 kasus akibat jalanan licin, 2 kasus kecelakaan akibat tidak ada rambu dan tikungan tajam [4]. Selain itu, pada 30 Agustus 2023, terjadi kecelakaan jalan rusak di Serpong, Tangerang Selatan, Banten.

Seorang pengendara motor kehilangan kendali dan terjatuh saat mencoba menghindari jalan berlubang di sisi kiri jalan, hingga akhirnya terlindas oleh sebuah *dump truk* [5]. Kerusakan jalan juga memberikan dampak negatif pada perekonomian masyarakat dan memperlambat akses transportasi, yang pada akhirnya mengurangi kenyamanan pengguna jalan [6]. Kerusakan jalan juga berdampak pada kenyamanan, ekonomi, serta meningkatkan emisi gas kendaraan [7]. Berdasarkan kasus-kasus kecelakaan lalu lintas yang disebabkan kerusakan jalan, perlu ada tindakan untuk mengurangi angka kecelakaan yang disebabkan permasalahan tersebut. Oleh karena itu telah diciptakan beberapa inovasi penelitian oleh [8] Pada penelitian tersebut belum memakai kamera dan *image processing* untuk mendeteksi kerusakan jalan. Kemudian penelitian yang dilakukan oleh [9], pada sistem yang dibuat belum dapat melakukan estimasi area kerusakan jalan. Kemudian inovasi dari [10], masih memiliki kelemahan yaitu FPS yang rendah dan membutuhkan *hardware* dengan spesifikasi tinggi.

Kemudian inovasi dari [11], namun terkendala daya tahan baterai dan jarak kendali. Penelitian yang dilakukan oleh [12] hanya memiliki dua *class* kerusakan jalan. Sementara itu, Setyawan dkk. [13] mengusulkan deteksi kerusakan jalan secara realtime dengan memanfaatkan transfer learning pada perangkat Nvidia Jetson Nano.

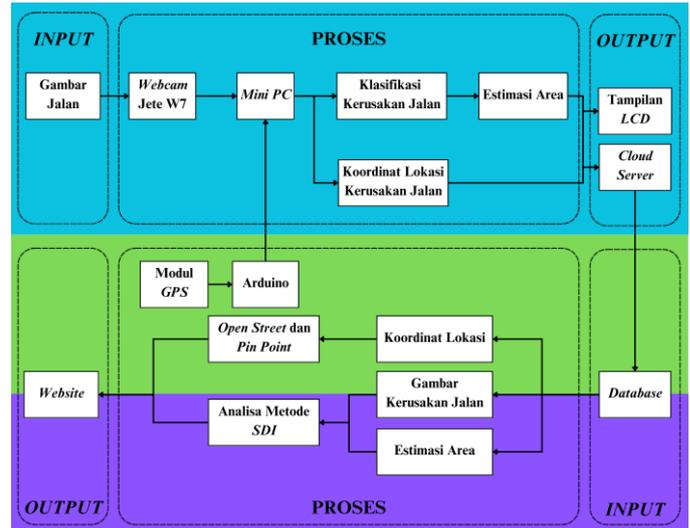
Berdasarkan latar belakang dan permasalahan yang terdapat pada penelitian sebelumnya, penulis mengajukan judul “Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive Roi*”. *Dataset* ini berisi kumpulan foto atau gambar yang mencakup jalan berlubang, jalan retak dan alur bekas roda. *Training dataset* dilakukan guna menghasilkan model yang terlatih sebagai objek deteksi. Pada proses *training dataset* tersebut dilakukan menggunakan *Google Colab* berdasarkan kumpulan *dataset* yang telah *diaugmentasi*. Kemudian sistem ini menggunakan kamera untuk menangkap gambar kerusakan jalan sebagai data masukan, yang kemudian diproses oleh *Mini PC*. Pemrosesan data dilakukan menggunakan *Framework PyTorch Object Detection* dengan memanfaatkan *YOLOv5s* sebagai model terlatih, sistem ini juga dilengkapi fitur untuk menangkap gambar (*screenshot*) kerusakan jalan secara otomatis. Setelah mendeteksi jenis kerusakan jalan, sistem akan melakukan estimasi luas area kerusakan, seperti lubang pada jalan, menggunakan *Area Estimation System*, hasil estimasi tersebut kemudian dikirimkan ke *cloud server* untuk penyimpanan dan analisis lebih lanjut. Penelitian ini diharapkan dapat menghasilkan alat yang mampu membantu petugas dinas terkait dalam mendeteksi dan mengestimasi area kerusakan jalan, sekaligus menganalisis indeks kerusakan yang ditampilkan melalui *website* lengkap dengan pemetaan lokasinya. Dengan demikian, kerusakan jalan yang berpotensi membahayakan pengguna dapat segera ditangani, sehingga membantu menekan angka kecelakaan lalu lintas yang disebabkan oleh masalah tersebut.

## II. METODOLOGI

Pada bagian ini akan menjelaskan tentang diagram blok sistem, prinsip kerja alat, serta *flowchart* sistem deteksi dan estimasi dimensi kerusakan jalan, serta teknik untuk menganalisa data.

### A. Diagram Blok Sistem

Pada Gambar 1 akan ditunjukkan blok diagram dari Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive Roi*.



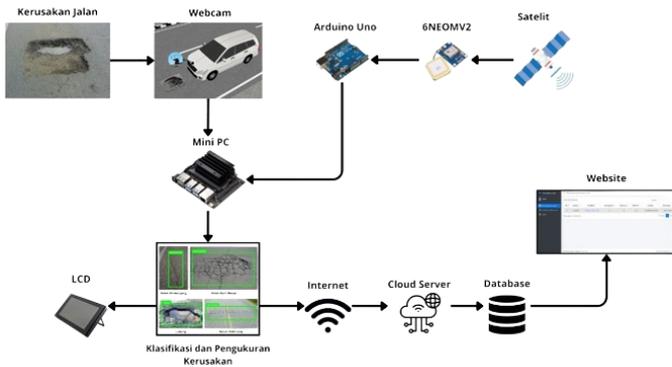
Gambar 1. Diagram Blok Sistem

Gambar 1 merupakan diagram blok sistem dari penelitian ini, pada Gambar 1 yang merupakan diagram blok sistem terdiri dari deteksi kerusakan jalan, sistem pemetaan kerusakan jalan, dan sistem analisa kerusakan jalan. Penulis mengerjakan bagian sistem deteksi kerusakan jalan, penjelasan dari Gambar 1 dibagi menjadi tiga bagian yaitu:

1. *Input*, gambar kerusakan jalan ditangkap oleh *webcam*. Kemudian hasil tangkapan *webcam* tersebut dikirimkan ke *Mini PC* dengan menggunakan kabel *USB*. Modul *GPS* menjadi masukan untuk *Arduino Uno* untuk mengetahui titik koordinat lokasi kerusakan jalan.
2. *Process*. Bagian proses adalah dimana semua data masukan diolah oleh *Mini PC* dan mikrokontroler. Data masukan berupa tangkapan *webcam* diolah oleh *Mini PC* dan data masukan berupa modul *GPS* diolah oleh *Arduino Uno*. Hasil pengolahan *Arduino* dikirimkan ke *Mini PC* dengan komunikasi serial.
3. *Output*. Setelah data masukan diproses oleh masing-masing pengolah data. Data keluaran yang dihasilkan oleh *Jetson Nano* adalah klasifikasi kerusakan permukaan jalan aspal yaitu retak memanjang, retak melintang, retak kulit buaya, alur bekas roda, dan lubang. Setelah itu akan dilakukan estimasi area berdasarkan *bounding box*. Kemudian data keluaran *GPS* yang dihasilkan oleh pemrosesan *Arduino Uno* berupa koordinat lokasi *latitude* dan *longitude*. Ketiga data keluaran tersebut diambil setiap segmen berdasarkan *start* dan *stop* deteksi *realtime*, kemudian data tiap segmen *upload* dalam database

### B. Prinsip Kerja Alat

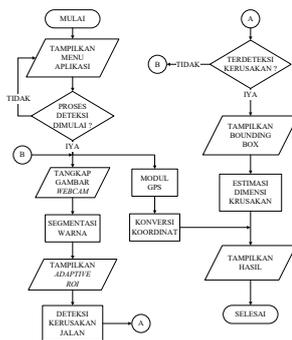
Pada Gambar 2 akan ditunjukkan prinsip kerja dari alat dari Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive Roi*.



Gambar 2. Prinsip Kerja Alat

Pada Gambar 2 memperlihatkan prinsip kerja alat dari tugas akhir berjudul "Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive Roi*" beroperasi. Proses dimulai dengan *webcam* mengambil gambar jalan. Pada *Mini PC*, deteksi kerusakan permukaan jalan dilakukan menggunakan model *deep learning YOLOv5s* untuk mengidentifikasi keberadaan kerusakan permukaan jalan. Hasil deteksi dan klasifikasi ditandai dengan *bounding box*. Berdasarkan *bounding box* tersebut dilakukan estimasi area kerusakan jalan. Selanjutnya, data hasil deteksi dan data lokasi dikirimkan ke *cloud server* dan diteruskan ke *database* melalui jaringan internet. Setiap *start* dan *stop* sistem deteksi *realtime* dilakukan pada tiap segmen jalan, sehingga data yang terkirim berupa data persegmen jalan yang berisi lebar jalan, panjang segmen jalan dan lokasi segmen jalan. Hasil deteksi tersebut akan ditampilkan pada *website*. Proses ini memungkinkan pemantauan dan analisis kondisi kerusakan permukaan jalan yang menghasilkan keluaran tingkat kerusakan jalan.

C. Flowchart Sistem



Pada Gambar 3 menunjukkan *flowchart* sistem untuk deteksi dan estimasi kerusakan jalan. Konfigurasi awal dimulai dengan menghubungkan sistem ke internet melalui jaringan *hotspot* atau *WiFi*. Proses ini dilakukan dengan menggunakan layar *LCD touchscreen* yang terhubung dengan *Mini PC*. Setelah terhubung ke internet, *webcam* Jete W7 akan menangkap gambar kerusakan jalan. Kemudian gambar tersebut dikirimkan ke *Mini PC* menggunakan kabel *USB*. Hasil gambar kerusakan jalan yang ditangkap oleh kamera akan dilakukan proses deteksi menggunakan model *YOLOv5s*. Disaat yang sama modul *GPS* akan menangkap koordinat dari kerusakan jalan.

Kemudian ketika terdeteksi adanya kerusakan jalan, selanjutnya akan melakukan proses klasifikasi kerusakan jalan, setelah proses klasifikasi akan ada tampilan *bounding box*, maka akan dilakukan proses estimasi area berdasarkan *bounding box*. Selanjutnya *Mini PC* akan mengambil foto kerusakan jalan tersebut serta lokasinya yang dikirimkan *Arduino Uno* dengan komunikasi serial. Setelah mencapai segmen yang diinginkan, maka dapat menekan tombol *stop* atau tombol "q" pada *keyboard* untuk menghentikan sistem deteksi, data yang tersimpan mulai dari sistem deteksi di mulai sampai sistem deteksi dihentikan merupakan data dari satu segmen jalan. Gambar dan keterangan lokasi tersebut disimpan sementara pada *Micro SD*, dan kemudian menyimpannya ke *database* ketika terdapat koneksi internet.

D. Teknik Analisa Data

Analisa data digunakan untuk mengevaluasi performa algoritma dari *Machine Learning* (ML) (khususnya *supervised learning*), menggunakan acuan *Confusion Matrix*. *Confusion Matrix* menunjukkan prediksi dan kondisi sebenarnya (aktual) yang dibuat oleh algoritma *Machine Learning*. Berdasarkan *Confusion Matrix*, kita bisa menentukan *Accuracy*, *Precision*, *Recall* dan *F1 Score*.

1. *Accuracy*, merupakan rasio prediksi benar (*positive* dan *negative*) dengan keseluruhan data yang ditunjukkan pada Persamaan (1).
2. *Precision*, merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif yang ditunjukkan pada Persamaan (2).
3. *Recall* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif yang ditunjukkan pada Persamaan (3).
4. *F1 Score* merupakan perbandingan rata-rata presisi dan *recall* yang dibobotkan yang ditunjukkan pada Persamaan (4).

III. HASIL DAN ANALISA

Pada bab ini akan dijelaskan mengenai hasil dan pembahasan mengenai Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive Roi*. Setelah melalui tahapan, perancangan hingga pengujian, pengambilan data serta menganalisa hasil yang diperoleh pada pengujian sistem.

A. Pengujian Threshold atau Confident Score

Beberapa *dataset* yang dijelaskan pada bagian sebelumnya digunakan untuk proses pengujian. Hasil dari proses pengujian adalah munculnya prediksi kotak pembatas (*bounding box*) dan kelas (*class*) pada objek dalam citra. Namun dalam proses pengujian perlu juga diperhatikan nilai *threshold* sebagai batas nilai kepercayaan kotak pembatas tentang termuatnya suatu objek dalam kotak pembatas. Nilai *threshold* juga mempengaruhi hasil kotak deteksi termasuk *True Positive (TP)*, *False Positive (FP)* atau *False Negative (FN)*.

Tabel 1. Data *Confusion Matrix* Threshold 0,2

Segmen Jalan (meter)	<i>Confusion Matrix</i>		
	<i>TP</i>	<i>FN</i>	<i>FP</i>
0-100	53	11	1

100-200	38	4	0
200-300	80	15	5

**Tabel 2.** Hasil *Confusion Matrix* Threshold 0,2

Segmen Jalan (meter)	Accuracy (%)	Precision	Recall	F1-Score (%)
0-100	81,5	0,98	0,82	89,2
100-200	90,4	1	0,904	94,9
200-300	80	0,94	0,84	88,7
Rata-rata	83,9	0,97	0,85	90,9

**Tabel 3.** Data *Confusion Matrix* Threshold 0,4

Segmen Jalan (meter)	Confusion Matrix		
	TP	FN	FP
0-100	38	22	4
100-200	32	10	0
200-300	52	43	0

**Tabel 4.** Hasil *Confusion Matrix* Threshold 0,4

Segmen Jalan (meter)	Accuracy (%)	Precision	Recall	F1-Score (%)
0-100	59,3	0,904	0,63	74,2
100-200	76,1	1	0,76	86,3
200-300	54,7	1	0,54	70,1
Rata-rata	63,3	0,96	0,64	76,8

Pada Tabel 2, nilai *Accuracy* sebesar 83,9%, *Precision* 0,97, *Recall* 0,85, dan *F1-Score* 90,9%. Sedangkan pada pengujian dengan *confidence score* 0.4 didapatkan hasil pada Tabel 4, dimana *Accuracy* sebesar 63,3%, *Precision* 0,96, *Recall* 0,64, dan *F1-Score* 76,8%. Dari data tersebut dapat disimpulkan jika semakin kecil *confidence threshold*, maka sistem deteksi dapat mendeteksi lebih banyak data kerusakan jalan. Namun hal ini juga dapat menyebabkan *false positive* dimana objek yang seharusnya tidak terdeteksi menjadi terdeteksi.

**B. Pengujian Deteksi Terhadap Intensitas Cahaya**

Pengujian proses deteksi terhadap intensitas cahaya memiliki tujuan untuk mengetahui seberapa besar tingkat keakurasian pada model deteksi dalam mendeteksi suatu objek terhadap intensitas cahaya yang diterima kamera serta untuk mengetahui pencahayaan optimal yang dapat digunakan untuk pendeteksian. Hasil dari pengujian intensitas cahaya ditunjukkan pada Tabel 5 berikut.

**Tabel 5.** Pengujian Intensitas Cahaya

Intensitas Cahaya (Lux)	Jumlah Kerusakan Jalan	Jumlah		
		TP	FP	FN
1.000	26	14	1	12
10.000	33	23	3	10
50.000	23	15	1	8

**Tabel 6.** Hasil *Confusion Matrix*

No	Intensitas Cahaya (Lux)	Accuracy (%)	Precision	Recall	F1-Score (%)
1	1.000	51,8	0,93	0,53	67,5
2	10.000	63,8	0,88	0,69	77,3
3	50.000	62,5	0,93	0,65	76,5

1	1.000	51,8	0,93	0,53	67,5
2	10.000	63,8	0,88	0,69	77,3
3	50.000	62,5	0,93	0,65	76,5

Berdasarkan performa hasil deteksi berdasarkan intensitas Cahaya yang ditunjukkan pada Tabel 6. Pada saat intensitas cahaya bernilai 1.000 *Lux*, didapatkan nilai *Accuracy* sebesar 51,8%, nilai *Precision* sebesar 0,93, nilai *Recall* sebesar 0,53 dan *F1-Score* sebesar 67,5%. Pada saat intensitas cahaya bernilai 10.000 *Lux*, didapatkan nilai *Accuracy* sebesar 63,8%, nilai *Precision* sebesar 0,88, nilai *Recall* sebesar 0,69 dan *F1-Score* sebesar 77,3%. Pada saat intensitas cahaya bernilai 50.000 *Lux*, didapatkan nilai *Accuracy* sebesar 62,5%, nilai *Precision* sebesar 0,93, nilai *Recall* sebesar 0,65 dan *F1-Score* sebesar 76,5%.

Untuk nilai performa deteksi dapat dikatakan bahwa pada saat intensitas cahaya 10.000 *Lux* menghasilkan performa deteksi yang lebih baik daripada saat intensitas cahaya 1.000 *Lux* dan 50.000 *Lux*, untuk pengujian kali ini, performa deteksi terbaik yang dihasilkan adalah pada saat intensitas Cahaya 10.000 lux.

**C. Pengujian Deteksi Terhadap Kecepatan Kendaraan**

Pengujian pengaruh kecepatan kendaraan dengan proses deteksi ini sangat bergantung dengan hasil pengujian performa *FPS*. Pengujian bertujuan untuk mencari kecepatan ideal kendaraan dalam mendeteksi kerusakan jalan. Hasil dari pengujian ditunjukkan pada Tabel 7 dan Tabel 8 berikut.

**Tabel 7.** Hasil Pengujian Saat Kecepatan 30 km/jam

Jenis Kerusakan Jalan	Jumlah asli	Confusion Matrix		
		TP	FN	FP
Lubang	15	14	1	2
Retak Buaya	5	4	1	1
Retak Memanjang	7	6	1	0
Retak Melintang	1	1	0	0
Alur Bekas Roda	1	1	0	0

**Tabel 8.** Hasil Pengujian Saat Kecepatan 40 km/jam

Jenis Kerusakan Jalan	Jumlah asli	Confusion Matrix		
		TP	FN	FP
Lubang	15	12	3	2
Retak Buaya	5	4	1	0
Retak Memanjang	7	4	3	0
Retak Melintang	1	1	0	0
Alur Bekas Roda	1	1	0	0

**Tabel 9.** Hasil *Confusion Matrix* Pengujian Kecepatan

No	Kecepatan kendaraan (km/jam)	Accuracy (%)	Precision	Recall	F1-Score (%)
1	30 km/jam	86,9	0,93	0,91	92,3
2	40 km/jam	81,5	0,97	0,83	88,7

Berdasarkan performa hasil deteksi berdasarkan kecepatan kendaraan yang ditunjukkan pada Tabel 9. Pada saat kecepatan kendaraan berada pada 30 km/jam didapatkan nilai *Accuracy* sebesar 86,9%, nilai *Precision* sebesar 0,93, nilai *Recall* sebesar 0,91 dan *F1-Score* sebesar 92,3%. Sedangkan

saat kecepatan kendaraan berada pada 40 km/jam didapatkan nilai *Accuracy* sebesar 81,5%, nilai *Precision* sebesar 0,97, nilai *Recall* sebesar 0,83 dan *F1 Score* sebesar 88,7%.

Untuk nilai performa deteksi dapat dikatakan bahwa kecepatan kendaraan berada pada 30 km/jam pada saat survei menghasilkan performa deteksi yang lebih baik daripada saat kecepatan kendaraan berada pada 40 km/jam. Untuk pengujian kali ini, performa deteksi terbaik yang dihasilkan adalah pada saat kecepatan kendaraan 30 km/jam.

**D. Pengujian Adaptive Region of Interest**

Pengujian bertujuan untuk mengetahui apakah *Adaptive Region of Interest* sudah membatasi area deteksi pada area badan jalan saja. Berikut merupakan hasil pengujian estimasi dimensi pada Tabel 10.

**Tabel 10.** Pengujian *Adaptive Region of Interest*

Frame	Total	Error
1	2	1
2	1	0
3	1	0
4	1	1
5	2	0
6	1	0
7	2	0
8	2	0
9	1	1
10	2	1
11	1	0
12	1	0
13	2	0
14	2	0
15	1	0
16	2	1
17	2	0
18	2	0
19	1	0
20	1	0
Total	30	5
Rata-rata error		0,25

Berdasarkan performa hasil deteksi *Adaptive Region of Interest* yang ditunjukkan pada Tabel 10. Dari data tersebut didapatkan total 30 kerusakan jalan yang terdeteksi dari 20 *frame* dan terdapat *error* deteksi berjumlah 5 dengan rata-rata *error* 0,25 atau 25%. Hal ini bisa terjadi dikarenakan kondisi lingkungan disekitar jalan menyerupai jalan sehingga sistem mengalami kesulitan untuk membedakan bagian luar jalan.

**E. Estimasi Dimensi Kerusakan Jalan**

Pengujian bertujuan untuk mengetahui keakuratan dari sistem estimasi area kerusakan jalan berdasarkan *bounding box*. Berikut merupakan hasil pengujian estimasi dimensi pada Tabel 11, Tabel 12, dan Tabel 13.

**Tabel 11.** Pengujian Estimasi Lebar

No	Estimasi sistem (cm)	Estimasi manual (cm)	Error (cm)
1	23,5	16	7,5

2	25,4	16	9,4
3	57,8	30	27,8
4	117,9	25	92,9
5	140,5	173	32,5
6	35,7	47	11,3
7	84,1	95	10,9
8	132	140	8
9	109,5	150	40,5
10	39,4	45	5,6
Rata-rata			24,64

**Tabel 12.** Pengujian Estimasi Panjang

No.	Estimasi sistem (cm)	Estimasi manual (cm)	Error (cm)
1	24	26	2
2	20,5	15	5,5
3	169,6	295	125,4
4	181,8	160	21,8
5	42,3	25	17,3
6	54,5	50	4,5
7	31,4	20	11,4
8	102	30	72
9	37,1	30	7,1
10	30,5	40	9,5
Rata-rata			27,65

**Tabel 13.** Pengujian Estimasi Kedalaman

No	Estimasi <i>OpenCV</i> (mm)	Pengukuran Manual (mm)	Error (mm)
1	16,4	19	2,6
2	18,6	24	5,4
3	17,9	12	5,9
4	16,1	11	5,1
5	17,6	13	4,6
6	15,8	22	6,2
7	13	15	2
8	16,6	18	1,4
9	20,6	9	11,6
10	17,6	21	3,4
Rata-rata			4,82

Estimasi dimensi dilakukan dengan mengubah ukuran piksel dari *bounding box* menjadi ukuran mm/piksel. Untuk mengubah nilai tersebut diperlukan lebar piksel kamera dan lebar area yang terdeteksi kamera. Dari Tabel 11 didapatkan hasil rata-rata *error* yaitu 24,64 cm pada lebar, Tabel 12 didapatkan hasil rata-rata *error* yaitu 27,65 cm pada panjang. Sedangkan untuk estimasi kedalaman dilakukan dengan mengubah nilai warna menjadi ukuran mm. Tabel 13 adalah hasil rata-rata *error* yang didapat yaitu 4,82 mm. *Error* tersebut bisa terjadi karena perbedaan lokasi pengukuran lebar area yang terdeteksi kamera dan posisi *bounding box*, jika posisi *bounding box* berada di atas area yang dilakukan pengukuran, maka *error*

akan menjadi besar. Hal tersebut dikarenakan semakin ke atas, lokasi jalan semakin jauh, sedangkan pengukuran lebar area kamera yang diukur berada di area bawah kamera atau dekat dengan *webcam*.

**F. Pengujian Keseluruhan Sistem**

Pengujian keseluruhan sistem memiliki tujuan untuk memastikan bahwa seluruh fitur pada alat dan aplikasi SIROGA dapat bekerja sesuai dengan perencanaan serta mencapai hasil yang diharapkan seperti dapat mendeteksi objek berupa kerusakan jalan, mengukur dimensi kerusakan jalan, mendeteksi lokasi keberadaan kerusakan jalan, dan mengirimkan data berupa gambar kerusakan jalan, dimensi dan lokasinya ke *database* dan ditampilkan pada *website* SIROGA.

**Tabel 14.** Pengujian Keseluruhan Sistem

Segmen	Jumlah Kerusakan Jalan	Jumlah		
		TP	FP	FN
0-100 meter	46	41	4	1
100-200 meter	8	6	0	2
200-300 meter	5	2	1	2
300-400 meter	3	3	0	0
400-500 meter	8	7	0	1
500-600 meter	24	20	1	3
600-700 meter	17	16	0	1
700-800 meter	17	15	1	1
800-900 meter	25	23	1	1
900-1000 meter	24	17	3	4
1000-1100 meter	18	12	3	3
Jumlah		162	14	19

Berdasarkan pengujian keseluruhan sistem deteksi, alat dan aplikasi SIROGA mampu melakukan pendeteksian, pengklasifikasian, dan mengestimasi dimensi kerusakan jalan. Dari data hasil pengujian yang menggunakan intensitas cahaya 10.000 *Lux* dan kecepatan kendaraan 30 km/jam didapatkan nilai *Accuracy* sebesar 83,07%, *Precision* sebesar 0,92, *Recall* sebesar 0,89, dan *F1-Score* sebesar 90,4%.

**Tabel 15.** Perbandingan Dengan Penelitian Terdahulu

Judul	<i>Accuracy</i> (%)	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i> (%)
Sistem Deteksi Kerusakan Permukaan Jalan Aspal Berbasis <i>Tensorflow</i> dan <i>Nvidia Jetson Nano</i>	51	0,51	1	73,3
Sistem Deteksi dan Estimasi Dimensi Kerusakan Jalan Berbasis <i>Computer Vision</i>	70,1	0,83	0,81	87,6

Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode <i>Deep Learning</i> dan <i>Adaptive Roi</i>	83,07	0,92	0,89	90,4
---	-------	------	------	------

Berdasarkan perbandingan dari Tabel 15, pada tugas akhir dengan judul “Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive Roi*” memiliki peningkatan pada *Accuracy* sebesar 12,97%, *Preceision* meningkat sebanyak 0,09, *Recall* meningkat sebanyak 0,08 dan *F1-Score* meningkat sebanyak 2,8%.

**IV. KESIMPULAN**

Berdasarkan hasil perancangan dan pengujian yang telah dilakukan pada tugas akhir yang berjudul “Sistem Deteksi dan Estimasi Dimensi Menggunakan Metode *Deep Learning* dan *Adaptive Roi*”, maka dapat diambil kesimpulan yaitu *training* model deteksi menggunakan *dataset* gambar kerusakan jalan dengan jenis retak memanjang, retak melintang, retak kulit buaya, alur bekas roda, dan lubang yang berjumlah 13.400 gambar, lalu *training* selama ± 6 jam dan menghasilkan nilai *Precision* sebesar 0.905, *Recall* sebesar 0.853, *mAP@50* sebesar 0,909. Sistem dapat mendeteksi objek berupa kerusakan jalan dengan jenis retak memanjang, retak melintang, retak kulit buaya, alur bekas roda, dan lubang dengan nilai akurasi tertinggi sebesar 77,3% pada saat intensitas Cahaya sebesar 10.000 *lux*. Sistem juga dapat melakukan estimasi dimensi kerusakan jalan berdasarkan *input* nilai lebar kamera, nilai resolusi kamera, serta panjang dan lebar *bounding box* dengan rata-rata *error* saat pengujian yaitu lebar sebesar 24,64 cm, panjang sebesar 27,65 cm, dan kedalaman sebesar 4,82 mm. Penerapan sistem deteksi, klasifikasi dan estimasi dimensi memerlukan beberapa parameter agar sistem dapat bekerja dengan maksimal. Proses pendeteksian ideal dilakukan dengan kecepatan 30 km/jam saat intensitas cahaya 10.000 *lux* hingga 50.000 *lux* dengan *FPS* yang didapatkan adalah 4 *FPS*.

**DAFTAR PUSTAKA**

- [1] F. Lestari, L. F. Lina, N. D. Puspaningtyas, dan I. C. Pratama, “Peningkatan Pengetahuan Patuh Berlalu Lintas Dan Berkendara Aman Pada Siswa Sma 1 Natar,” *JSSTCS*, vol. 3, no. 2, hlm. 249, Sep 2022, doi: 10.33365/jsstcs.v3i2.2118.
- [2] WHO, “Road traffic injuries,” *World Health Organization*. [Daring]. Tersedia pada: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>
- [3] N. Muhamad, “Kecelakaan Lalu Lintas Terbanyak Terjadi di Jawa Timur hingga November 2023,” *Databoks*. [Daring]. Tersedia pada: <https://databoks.katadata.co.id/demografi/statistik/7d1430d9d138f1b/kecelakaan-lalu-lintas-terbanyak-terjadi-di-jawa-timur-hingga-november-2023>
- [4] M. L. Andika, “Duh! Jalan Rusak Jadi Penyebab Banyak Kecelakaan Lalu Lintas di Jakarta,” *detikoto*. Diakses: 8 Januari 2024. [Daring]. Tersedia pada: <https://oto.detik.com/berita/d-5624401/duh-jalan-rusak-jadi-penyebab-banyak-kecelakaan-lalu-lintas-di-jakarta>
- [5] M. C. Halim dan J. Carina, “Hindari Lubang Jalan, Pengendara Motor di Serpong Tewas Terlindas Truk,” *KOMPAS.com*. [Daring]. Tersedia pada: <https://megapolitan.kompas.com/read/2023/08/30/12032091/hindari-lubang-jalan-pengendara-motor-di-serpong-tewas-terlindas-truk>

- [6] A. N. Fajar, A. Khamid, W. Diantoro, D. D. Apriliano, dan M. Yunus, "Analisis Tingkat Kerusakan pada Jalan Pagerbarang – Margasari Kabupaten Tegal," 2021.
- [7] I. Nabawi, Y. Feriska, dan W. Diantoro, "Analisis Dampak Kerusakan Jalan terhadap Pengguna Jalan dan Lingkungan di Ruas Jalan Pebatan - Rengaspendawa Brebes," 2021.
- [8] I. Bosi, E. Ferrera, D. Brevi, dan C. Pastrone, "In-Vehicle IoT Platform Enabling the Virtual Sensor Concept: A Pothole Detection Use-case for Cooperative Safety.," dalam *Proceedings of the 4th International Conference on Internet of Things, Big Data and Security*, Heraklion, Crete, Greece: SCITEPRESS - Science and Technology Publications, 2019, hlm. 232–240. doi: 10.5220/0007690602320240.
- [9] Y. Liu, X. Zhang, B. Zhang, dan Z. Chen, "Deep Network For Road Damage Detection," IEEE. [Daring]. Tersedia pada: <https://ieeexplore.ieee.org/document/9377991/metrics#metrics>
- [10] G. Guo dan Z. Zhang, "Road damage detection algorithm for improved YOLOv5," *Sci Rep*, vol. 12, no. 1, Art. no. 1, Sep 2022, doi: 10.1038/s41598-022-19674-8.
- [11] L. A. Silva, V. R. Q. Leithardt, V. F. L. Batista, G. Villarrubia González, dan J. F. De Paz Santana, "Automated Road Damage Detection Using UAV Images and Deep Learning Techniques," *IEEE Access*, vol. 11, hlm. 62918–62931, 2023, doi: 10.1109/ACCESS.2023.3287770.
- [12] E. P. Nugroho, "Sistem Deteksi Kerusakan Permukaan Jalan Aspal Berbasis Tensorflow Dan Jetson Nano," 2022.
- [13] S. B. Setyawan, H. Arrosida, A. Elhakim, D. Shahab, dan E. P. Nugroho, "Realtime road damage detection using transfer learning with Nvidia Jetson Nano," *AIP Conference Proceedings*, vol. 3145, no. 1, p. 020006, Jul. 2024, doi: 10.1063/5.0212345