

PENERAPAN ALGORITMA YOLO v7 SEBAGAI DETEKSI KECELAKAAN KENDARAAN PADA LALU LINTAS

Moch. Ghulam Abrari Binuri¹⁾, Tining Haryanti²⁾, Muhamad Amirul Haq³⁾

^{1), 2), 3)} Program Studi Informatika, Fakultas Teknik, Universitas Muhammadiyah Surabaya
 Jl Sutorejo No. 59, Surabaya
 Email : ghulamabrari@gmail.com¹⁾

Abstrak

Penerapan algoritma YOLOv7 dalam deteksi kecelakaan lalu lintas menggunakan google colab pada data training dari dataset robolow "accident 1" yang terdiri dari total 1522 gambar, dengan parameter batch size 1, epoch 40 dan menggunakan optimizer SGD menghasilkan model dengan tingkat precision sebesar 65.1%, recall 45.3%, mAP@.5 52.1%, dan mAP@.5:.95 26.4%, dengan waktu pelatihan 2.319 hours berdasarkan hasil tersebut untuk deteksi kurang memuaskan. Analisis pada grafik training data menunjukkan bahwa untuk mencapai hasil yang lebih memuaskan, diperlukan jumlah epoch yang lebih besar saat melatih data. Dari hasil grafik training data, terlihat bahwa semakin banyak epoch pelatihan, semakin akurat data trainingnya. Oleh karena itu, disarankan untuk melatih data lebih dari 40 epoch guna meningkatkan kualitas model dan deteksi kecelakaan lalu lintas.

Kata kunci: YOLOv7, deteksi, kecelakaan, lalu lintas.

Abstract

The application of the YOLOv7 algorithm in traffic accident detection using google colab on training data from the robolow dataset "accident 1" consisting of a total of 1522 images, with batch size parameters of 1, epoch 40 and using the SGD optimizer resulted in a model with a precision level of 65.1%, recall 45.3%, mAP@.5 52.1%, and mAP@.5:.95 26.4%, with a training time of 2,319 hours based on these results for unsatisfactory detection. Analysis on the training data chart shows that in order to achieve more satisfactory results, a larger number of epochs are required when training the data. From the results of the training data graph, it can be seen that the more training epochs, the more accurate the training data will be. Therefore, it is recommended to train data on more than 40 epochs to improve model quality and traffic accident detection.

Keywords : YOLOv7, detection, accident, traffic.

1. Pendahuluan

Peningkatan jumlah penduduk dalam negara berkembang memicu mobilitas masyarakat yang lebih tinggi di daerah dan kota. Namun, jika sarana dan prasarana lalu lintas tidak memadai, maka kecelakaan lalu lintas sering terjadi, menimbulkan kerugian yang signifikan. Kecelakaan ini menimbulkan kerugian material dan dapat mengakibatkan korban yang meninggal dunia [1].

Berdasarkan [2], pada tahun 2021 terdapat 103.645 kasus kecelakaan lalu lintas di Indonesia, yang mengalami peningkatan dibandingkan dengan 100.028 kasus pada tahun 2020. Kecelakaan tersebut mengakibatkan 25.266 korban jiwa. Selain itu, terdapat 10.553 korban luka berat dan 117.913 korban luka ringan. Data ini menunjukkan bahwa kecelakaan lalu lintas merupakan masalah yang mengancam nyawa. Untuk meningkatkan keselamatan lalu lintas, pengembangan sistem deteksi kecelakaan menjadi sangat penting. Dengan sistem ini, penyebab kecelakaan dapat di dokumentasikan dan dianalisis, sehingga solusi yang tepat dapat segera diterapkan untuk mencegah kecelakaan serupa di masa mendatang.

Di tengah beragamnya algoritma yang dikembangkan untuk mendeteksi objek, YOLO (You Only Look Once) framework telah meraih ketenaran karena keseimbangannya yang sangat baik antara kecepatan dan akurasi. Fitur ini memungkinkan pengenalan objek yang efisien dan handal dalam gambar [3]. YOLO (You Only Look Once) adalah algoritma deteksi objek yang berbasis Convolutional Neural Network (CNN). Dalam konteks YOLO telah terbukti unggul dibandingkan dengan model Mask R-CNN dan SSD menurut [4] [5]. Karena keunggulan ini, peneliti memilih YOLO untuk deteksi kecelakaan.

Versi terbaru dari YOLO, yaitu YOLOv7, diterbitkan pada Juli 2022 dan menurut [6], berhasil mengungguli pendahulunya dalam hal kecepatan dan akurasi. Keberadaan YOLOv7 yang relatif baru, serta minimnya penelitian menggunakan CNN untuk deteksi kecelakaan, menjadi alasan utama untuk menerapkan YOLOv7 dalam deteksi kecelakaan. Ini diharapkan dapat memberikan kontribusi signifikan pada keselamatan lalu lintas dan pengurangan risiko kecelakaan.

2. Dasar teori

2.1 CNN

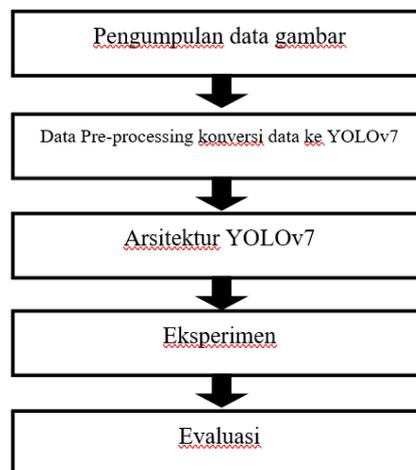
Convolution Neural Network (CNN) merupakan salah satu jenis jaringan syaraf tiruan yang digunakan untuk memproses data gambar. CNN melakukan proses transformasi gambar menjadi array dengan memperhitungkan nilai setiap piksel, kemudian melakukan perhitungan matriks yang dikenal sebagai konvolusi [7]. Dalam perkembangannya, penggunaan CNN semakin disederhanakan dengan teknik transfer learning, di mana kita dapat menggunakan algoritma yang sudah tersedia dan hanya perlu memasukkan data input sesuai dengan kebutuhan. Terdapat berbagai macam algoritma CNN yang ditujukan untuk berbagai kebutuhan, seperti deteksi objek, pengenalan objek, analisis dokumen, klasifikasi gambar, klasifikasi video, dan lain sebagainya [8].

2.2 Deteksi Objek

Metode deteksi objek merupakan bagian dari computer vision yang digunakan untuk mendeteksi objek dalam gambar dengan menggunakan bounding box dan mengklasifikasikan objek ke dalam kelas-kelas tertentu [9]. Salah satu metrik yang paling relevan dan umum digunakan untuk mengevaluasi kinerja algoritma pendeteksian objek adalah mean average precision (mAP) [10].

3. Metodologi Penelitian

Penelitian ini bertujuan menerapkan algoritma YOLOv7 dalam deteksi kecelakaan lalu lintas dengan tujuan meningkatkan keselamatan. Penelitian ini melibatkan serangkaian tahap pelaksanaan yang harus dijalani, yaitu:



Gambar 1. *Desain Penelitian*

3.1 Pengumpulan data gambar

Untuk melatih model YOLOv7 deteksi kecelakaan, data dikumpulkan dengan mendownload dataset dari situs web Roboflow dengan url <https://roboflow.com/> yang berjudul "Accident Detection 1". Dataset ini terdiri dari total 1522 gambar kecelakaan, yang kemudian dibagi menjadi tiga set utama untuk keperluan pelatihan, validasi, dan pengujian [11]. Diperoleh Data Train Set mencakup 70% dari total dataset, yaitu 1065 gambar, Data Valid Set menyertakan 20% dari dataset, yaitu sebanyak 435 gambar dan data Test Set memiliki 10% dari dataset, yang terdiri dari 219 gambar.



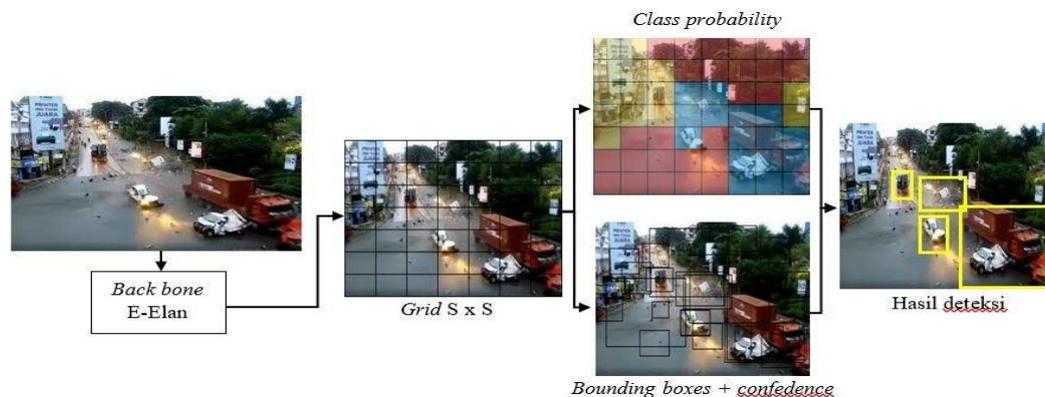
Gambar 2. Gambar kecelakaan lalulintas

3.2 Data Pre-processing konversi data ke YOLOv7

Proses konversi data ke YOLOv7 melibatkan penggunaan layanan bernama Roboflow. Proses ini melibatkan dua tahap utama, yaitu konversi data ke dalam format TXT annotations (anotasi teks) dan konfigurasi YAML yang nantinya akan digunakan dengan model YOLOv7.

3.3 Tahapan YOLOv7

YOLOv7 menggunakan satu model CNN untuk melakukan deteksi target *end-to-end*. Algoritma ini mengambil seluruh gambar sebagai input dan secara langsung menghitung lokasi kotak pembatas serta kategori objek dalam lapisan keluaran. Terdiri dari empat modul utama: *input*, *backbone*, *head*, dan prediksi.



Gambar 3. Tahapan YOLOv7

Pada tahap awal citra akan masuk ke *backbone* E-Elan untuk memastikan bahwa model tidak menjadi terlalu kompleks, sehingga mengurangi risiko *overfitting* pada data pelatihan. Lalu Citra dibagi menjadi grid dengan ukuran $s \times s$. Setiap grid akan diprediksi menggunakan *bounding box* dan menghasilkan nilai *confidence*. *Confidence* merupakan nilai yang mengindikasikan sejauh mana *bounding box* tersebut berisi objek yang sesuai dengan perencanaan dan akurasi prediksi.

3.4 Eksperimen

Proses pelatihan data dilakukan menggunakan google colab. Tahap awal pelatihan YOLOv7 membutuhkan format spesifik untuk dataset kustom dalam bentuk *file* yaml. Proses pelatihan data menggunakan parameter seperti *Epoch*, *Batch size*, dan *Optimizer*. Dengan menerapkan *callback* untuk menghentikan proses pelatihan jika nilai kerugian meningkat selama prosesnya.

3.5 Evaluasi

Eksperimen memerlukan indeks performa untuk mengevaluasi model algoritma. Menurut ke indeks evaluasi model jaringan saraf [12]. Evaluasi hasil dengan menggunakan metrik evaluasi

seperti akurasi, presisi,

recall, lalu analisis hasil tersebut untuk menarik kesimpulan tentang keberhasilan dan kekurangan metode yang diimplementasikan. Hal ini penting untuk menghasilkan laporan yang akurat dan obyektif.

$$Precision = \frac{tp}{tp + fp} \tag{1}$$

$$Recall = \frac{tp}{tp + fn} \tag{2}$$

Dalam evaluasi ini adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *True Negative* (FN). *True Positive* (TP) terjadi ketika model berhasil mengklasifikasikan data sebagai positif (ya) dan jawaban aktualnya juga adalah positif (ya). Sebaliknya, *True Negative* (TN) terjadi ketika model mengklasifikasikan data sebagai negatif (tidak) dan jawaban aktualnya juga negatif. Namun, ada juga situasi di mana model dapat salah mengklasifikasikan data. *False Positive* (FP) terjadi ketika model mengklasifikasikan data sebagai positif (ya), tetapi jawaban aktualnya adalah negatif (tidak). Sedangkan *False Negative* (FN) terjadi ketika model mengklasifikasikan data

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

sebagai negatif (tidak), tetapi jawaban aktualnya adalah positif (ya). (3)

Mean Average Precision (mAP) adalah dengan membagi nilai AP dari setiap jumlah kelas pada data dandihitung melalui nilai threshold [13].

4. Pengujian dan Pembahasan

Hasil dari eksperimen dengan parameter-parameter mendapat hasil *training* berikut : Tabel 3.1 Hasil data *training* menggunakan beberapa parameter

Parameter		Precision	Recall	mAP@.5	mAP@.5:95	Time
Epoch	Batch size					
10	1	77,8%	16,9%	21,4%	8%	0.635 Hours
20	1	76,1%	25,9%	26,8%	11%	1.132 Hours
30	1	50,1%	46,6%	44,3%	21.6 %	1.746 Hours
40	1	65,1%	57,9%	60,1%	28,4%	2.309 Hours
10	8	12%	28,9%	9%	2%	0.335 Hours
20	8	73,4%	23,9%	24,4%	10%	0.640 Hours
30	8	47,2%	35,2%	31,9%	13,7%	0.960 Hours

40 8 36,9% 56,9% 43,3% 19,4% 1.277 Hours

Dari hasil tabel diatas dapat diketahui bahwa nilai *epoch* 40 dengan *batch size* 1 lebih unggul dari padaparameter lainnya yang mendapatkan *precision* 65,1%, *Recall* 57,9%, *mAP@.5* 60,1%, *mAP.5:95* 28,4% dengan waktu 2.309 *hours*, sehingga untuk selanjutnya *mentraining* menggunakan *optimizer* SGD dan Adam menggunakan parameter *epoch* 40 dan *batch size* 1.

Tabel 3.2 Hasil *training epoch 40 bacth size 1* menggunakan beberapa optimizer

<i>Optimizer</i>	<i>Precision</i>	<i>Recall</i>	<i>mAP@.5</i>	<i>mAP@.5:95</i>	<i>Time</i>
SDG	65,1%	57,9%	60,1%	28,4%	2.309 Hours
ADAM	55,8%	12,8%	3%	0,07%	2.300 Hours

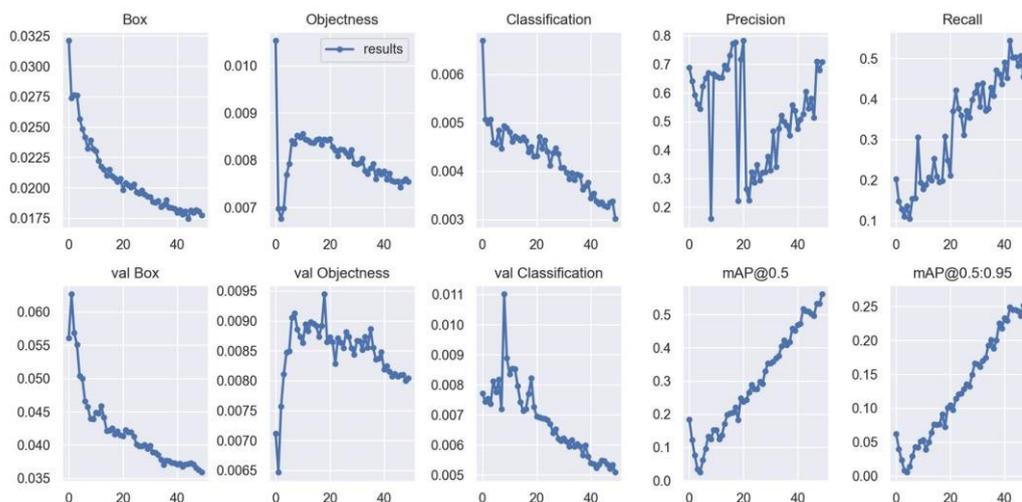
Pada hasil tabel diatas untuk digunakan dalam *training* data menggunakan dataset dan parameter yang sama *optimizer* dengan SGD lebih unggul jika dibandingkan *optimizer* ADAM dengan selisih *precision* 9,3%, *recall* 45,1%, *mAP@.5* 57,1%, *mAP@.5:95* 28,33% tetapi untuk waktu SGD membutuhkan waktu yang lebih lama sekitar 0.009 *hours*.

Setelah mendapatkan hasil data *training* dengan hasil paling unggul dari eksperimen, akan dievaluasi terhadap data lain dari dataset yang digunakan untuk mengetahui hasil deteksi menggunakan data *training* dengan parameter *epoch* 40, *batch size* 1 dan menggunakan *optimizer* SGD.

Tabel 3.3 Hasil evaluasi training 40 epoch, batch size 1 dan optimizer SGD

<i>Data</i>	<i>Precision</i>	<i>Recall</i>	<i>mAP@.5</i>	<i>mAP@.5:95</i>	<i>Time</i>
<i>Validation</i>	64,8%	57,90%	60%	28%	12 second
<i>Test</i>	63,4%	57,1%	59%	28,2%	7 second

Hasil dari data *training* diuji untuk deteksi kecelakaan pada citra dataset validasi yang berjumlah 435 gambar mendapatkan hasil *precision* 64,80%, *recall* 57,90%, *mAP@.5* 60%, *mAP@.5:95* 28% dibutuhkan waktu 12 second dan menguji dengan deteksi kecelakaan dengan citra dataset test yang berjumlah 219 mendapatkan hasil *precision* 63,4%, *Recall* 57,1%, *mAP@.5* 59%, *mAP@.5:95* 28,2% dengan waktu 7 second. Setelah pelatihan selesai, hasilnya dapat dilihat pada grafik *confusion matrix*. Grafik ini memberikan gambaran lebih lanjut tentang kinerja model terhadapsetiap kelas deteksi.



Gambar 4. Hasil *confusion matrix*

Berdasarkan gambar 4, untuk mencapai hasil yang memuaskan diperlukan jumlah *epoch* yang lebih besar pada saat melatih data. Hal ini disebabkan oleh kenaikan pada grafik di samping, yang dapat diartikan bahwa semakin banyak *epoch* pelatihan, semakin akurat hasil *training*-nya. Hasil deteksi pada data *training* menunjukkan bahwa model YOLOv7 berhasil mengidentifikasi dan mengklasifikasikan objek yang dimana waktu yang diperlukan untuk mendeteksi antara 0.2 ms - 6.9 ms :



Gambar 5. Deteksi hasil menggunakan yolo v7

Dalam hasil deteksi, terjadi variasi di mana beberapa hasil klasifikasi gambar sesuai, sementara yang lainnya tidak sesuai. Terkadang, dalam satu citra, kita bisa menemukan kombinasi dari hasil klasifikasi yang benar dan salah.

5. Kesimpulan

Dalam penelitian ini, model YOLOv7 diterapkan menggunakan citra berukuran 640x640 piksel sebagai input dengan dataset berisi 1522 gambar untuk data *training*, 219 gambar untuk data testing, dan 435 gambar untuk data validasi, yang mencakup variasi kondisi kecelakaan kendaraan dalam lalu lintas. Dengan menggunakan nilai *epoch* 40 dan *batch size* 1 dengan *optimizer* SGD, hasil evaluasi model menunjukkan *precision* sebesar 65.1%, *recall* 45.3%, *mAP@.5* 52.1%, dan *mAP@.5:95* 26.4%, dengan waktu pelatihan 2.319 jam. Penelitian ini menunjukkan bahwa *optimizer* SGD memberikan hasil yang lebih baik daripada ADAM, dan pentingnya nilai *batch size* dalam menentukan akurasi dan waktu pelatihan model. Rekomendasi penelitian berikutnya adalah untuk mengeksplorasi penggunaan *batchsize* yang lebih kecil dan lebih dari 40 *epoch* untuk meningkatkan performa model, khususnya untuk mencapai *precision* yang lebih tinggi daripada 65.1%.

Daftar Pustaka

- [1] R. Y. Sabilla dan D. Yendri, "Sistem Monitoring Kondisi dan Posisi Pengemudi Berbasis Internet of things," *Chipset*, vol. 2, no. 01, hal. 1–10, 2021, doi: 10.25077/chipset.2.01.1-10.2021.
- [2] V. A. Dihni, "Angka Kecelakaan Lalu Lintas di Indonesia Meningkat di 2021," *databoks.katadata.co.id*, 2022. <https://databoks.katadata.co.id/datapublish/2022/03/24/angka-kecelakaan-lalu-lintas-di-indonesia-meningkat-di-2021-tertinggi-dari-kecelakaan-motor> (diakses 23 November 2023).
- [3] J. Terven dan D. Cordova-Esparza, "A Comprehensive Review of YOLO: From YOLOv1 and Beyond," hal. 1–27, 2023, [Daring]. Tersedia pada: <http://arxiv.org/abs/2304.00501>
- [4] E. R. Setyaningsih dan M. S. Edy, "YOLOv4 dan Mask R-CNN Untuk Deteksi Kerusakan Pada Karung Komoditi," *Teknika*, vol. 11, no. 1, hal. 45–52, 2022, doi: 10.34148/teknika.v11i1.419.
- [5] F. R. Haikal, "ANALISIS PERBANDINGAN AKURASI DAN PERFORMA ALGORITMA DETEKSI OBJEK PADA YOLO v3 DENGAN SINGLE SHOT MULTIBOX DETECTOR (SSD)," *Front. Neurosci.*, vol.14, no. 1, hal. 1–13, 2021.
- [6] C.-Y. Wang, A. Bochkovskiy, dan H.-Y. M. Liao, "YOLOv7: Trainable bag of freebies sets new state of the art for real-time object detectors," hal. 1–17, 2022, [Daring]. Tersedia pada: <http://arxiv.org/abs/2207.02696>

- [7] Z. Li, F. Liu, W. Yang, S. Peng, dan J. Zhou, "CNN survey," *IEEE Trans. neural networks Learn. Syst.*, vol. 33, no. 12, hal. 6999–7019, 2022.
- [8] W. Wang dan Y. Yang, "Development of convolutional neural network and its application in image classification: A survey," *Opt. Eng.*, vol. 58, hal. 1, 2019, doi: 10.1117/1.OE.58.4.040901.
- [9] N. Dvornik, J. Mairal, dan C. Schmid, "Modeling visual context is key to augmenting object detection datasets," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol.

- 11216 LNCS, hal. 375–391, 2018, doi: 10.1007/978-3-030-01258-8_23.
- [10] T. Diwan, G. Anirudh, dan J. V. Tembhurne, “Object detection using YOLO: challenges, architectural successors, datasets and applications,” *Multimed. Tools Appl.*, vol. 82, no. 6, hal. 9243–9275, 2023, doi: 10.1007/s11042-022-13644-y.
- [11] College, “accident detection 1 Dataset,” *Roboflow universe*. Roboflow, 2023. Diakses: 16 November 2023. [Daring]. Tersedia pada: <https://universe.roboflow.com/college-ccvha/accident-detection-1>
- [12] R. Wang dan J. Li, “Bayes test of precision, recall, and F1 measure for comparison of two natural language processing models,” *ACL 2019 - 57th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf.*, hal. 4135–4145, 2020, doi: 10.18653/v1/p19-1405.
- [13] M. Dio, R. Pratama, B. Priyatna, S. Shofiah, dan A. Lia, “Deteksi Objek Kecelakaan Pada Kendaraan Roda Empat Menggunakan Algoritma YOLOv5 Car Vehicle Accident Object Detection Using YOLOv5 Algorithm,” *J. Pendidik. Tambusai*, vol. 12, no. 2, hal. 15–24, 2022.