

Pendekatan Deep Learning untuk Deteksi Kantuk dengan YOLOv12

Diesti Hidayani¹, Mustofa Romadhani², Ardiansyah²

¹Universitas Muhammadiyah Klaten, Kabupaten Klaten

²Program Studi Teknologi Informasi, Fakultas Kesehatan dan Teknologi, Universitas Muhammadiyah Klaten, Klaten

Email: ¹Diestihidayani@gmail.com, ²Mustofaramadhani415@gmail.com, ³ardiansyah@umkla.ac.id

^{*}Penulis Korespondensi

ABSTRACT — Drowsiness while driving is a significant contributor to traffic accidents. To mitigate such occurrences, a precise and real-time drowsiness detection system is essential. This research aims to create a computer vision-based drowsiness detection system utilizing the YOLOv12 algorithm. The dataset was sourced from Kaggle and manually annotated with the help of Roboflow. It was categorized into two groups: drowsy and non-drowsy, with the original 5,000 images augmented to a total of 6,976 images. The model training utilized the AdamW optimizer (learning rate=0.001667, momentum=0.9) over 100 epochs and a batch size of 4. Performance assessment indicates that the model attained an mAP@50 of 0.732 and an mAP@50-95 of 0.62, alongside a precision of 0.648 and a recall of 0.928. These findings illustrate that YOLOv12 can successfully identify drowsiness in real-time. Nevertheless, the performance of the model is significantly influenced by the quality and balance of the dataset. Consequently, enhancing the structure and distribution of the dataset is vital for improving detection accuracy.

KEYWORDS — Real-Time, Deep Learning, Drowsiness Detection, YOLOv12.

INTISARI — Kantuk saat mengemudi adalah salah satu penyebab utama kecelakaan lalu lintas. Untuk mencegah kecelakaan yang disebabkan oleh kantuk, diperlukan sistem deteksi yang akurat dan real-time. Penelitian ini bertujuan untuk mengembangkan sistem pendeteksi kantuk yang berbasis computer vision dengan menggunakan algoritma YOLOv12. Dataset yang digunakan berasal dari Kaggle dan telah melalui proses anotasi manual menggunakan Roboflow. Data citra diklasifikasikan menjadi dua kategori, yaitu kantuk dan tidak kantuk, dengan total 5.000 gambar yang kemudian diperluas menjadi 6.976 gambar melalui teknik augmentasi. Proses pelatihan model dilakukan dengan menggunakan optimizer AdamW (lr=0.001667, momentum=0.9) dan konfigurasi 100 epoch, dengan batch size 4. Evaluasi performa menunjukkan mAP@50 sebesar 0,732 dan mAP@50-95 sebesar 0,62, dengan precision 0,648 dan recall 0,928. Hasil ini menunjukkan bahwa YOLOv12 dapat mendeteksi kantuk dengan akurasi tinggi secara real-time. Namun, performa model sangat tergantung pada kualitas dan keseimbangan data. Oleh karena itu, diperlukan peningkatan pada struktur dan distribusi dataset untuk mencapai hasil yang lebih optimal.

KATA KUNCI — Real-Time, Deep-Learning, Deteksi Kantuk, YOLOv12.

I. PENDAHULUAN

Mengantuk saat mengemudi sangat berbahaya dan sering menyebabkan kecelakaan lalu lintas. Banyak kecelakaan terjadi karena pengemudi kehilangan fokus akibat kantuk, sehingga perlu ada alat yang bisa mendeteksi tanda-tanda kantuk sejak dini untuk mencegah hal buruk terjadi [1].

Salah satu cara yang mudah digunakan adalah dengan memantau wajah pengemudi menggunakan kamera. Dengan melihat gerakan mata dan kepala, kita bisa tahu apakah seseorang mulai mengantuk tanpa harus memasang alat di tubuh pengemudi [2].

Teknologi deep learning, yaitu komputer yang belajar mengenali pola dari gambar, sangat membantu dalam mengenali tanda kantuk ini [3]. Ada algoritma bernama YOLO yang sangat cepat dan akurat dalam mendeteksi objek dan fitur pada gambar. Versi terbarunya, YOLOv12, lebih canggih dan cocok untuk mendeteksi perubahan kecil di wajah pengemudi [4].

Penelitian ini menggunakan Computer Vision dengan algoritma YOLOv12 untuk mendeteksi tingkat kantuk individu. Meskipun metode ini dapat menunjukkan kecepatan dan efisiensi dalam mengenali ekspresi kantuk, namun hasil yang diperoleh masih belum sepenuhnya sempurna. Hal ini disebabkan oleh perbedaan karakteristik dataset yang digunakan dalam proses pengujian, seperti adanya ketidakseimbangan dataset yang digunakan. Karenanya pada pengujian menunjukkan perbedaan yang signifikan, yaitu 0,9 untuk dataset yang menunjukkan

gejala kantuk, dan 0,3 untuk dataset yang menunjukkan kondisi tidak mengantuk. Hasil ini menunjukkan bahwa akurasi sistem sangat bergantung pada kesesuaian dataset dengan kondisi actual, oleh karena itu, diperlukan peningkatan kualitas dan keseimbangan dataset agar sistem pendeteksian kantuk berbasis YOLOv12 dapat bekerja dengan akurat.

Pada penelitian yang kami lakukan berfokus pada menguji coba model yolov12 untuk mendeteksi kantuk. Selain itu, penelitian ini melakukan uji coba data augmentasi dataset terhadap training model.

II. TINJAUAN PUSTAKA

Penelitian [5] ini menyajikan sistem deteksi kelelahan yang menggunakan sensor inframerah untuk mengidentifikasi tanda-tanda kelelahan, terutama untuk situasi yang terjadi pada malam hari, seperti ketika seseorang mengemudi di malam hari dan memicu peringatan pencegahan kecelakaan.

Penelitian [3] ini berfokus pada pengembangan sistem deteksi kantuk yang menggunakan CNN untuk memperingatkan pengemudi ketika mereka terlalu lama menutup mata, meningkatkan keselamatan di jalan. Namun pada penelitian yang kami lakukan kami menggunakan YOLOv12 karena dengan menggunakan YOLOv12 ini mampu mendeteksi objek secara real-time dengan akurasi tinggi, sehingga lebih efektif dalam mengenali tanda-tanda kantuk pada wajah pengemudi.

Deteksi kantuk sangat penting untuk keselamatan di tempat seperti transportasi [6]. Dengan menggunakan jaringan saraf konvolusional dan pembelajaran transfer, penelitian ini mencapai tingkat akurasi antara 90 dan 99,86% di beberapa kumpulan data, dan mencakup aplikasi seluler yang mudah digunakan untuk implementasi praktis. Pada penelitian yang kami lakukan menggunakan YOLOv12, Penelitian ini berhasil mendeteksi tanda-tanda kantuk secara real-time dengan akurasi tinggi pada berbagai dataset. Algoritma YOLOv12 dan teknologi visi komputer dirancang untuk dapat digunakan dalam aplikasi berbasis perangkat seluler.

Metode deteksi kantuk mempertimbangkan kondisi fisik dan perilaku kendaraan selain tanda-tanda seperti menguap, mata tertutup, dan gerakan kepala untuk menilai kewaspadaan pengemudi [7]. Dalam makalah ini, metode ini diklasifikasikan berdasarkan faktor subjektif, perilaku, kendaraan, dan fisiologis, dan kemudian memeriksa kekuatan dan kelemahan metode tersebut.

Untuk mendeteksi kelelahan, AI memantau gerakan mata dan frekuensi kedipan. Dengan memanfaatkan kumpulan data dan Python untuk implementasi dengan pustaka OpenCV, sistem ini meningkatkan keselamatan jalan dengan mencegah kecelakaan yang disebabkan oleh kelelahan pengemudi [8].

Sistem deteksi kantuk menggunakan AI, Python, OpenCV, dan Arduino memperingatkan pengemudi dengan menguap dan analisis postur kepala, yang bertujuan untuk mencegah kecelakaan yang disebabkan oleh mengemudi mengantuk [9].

Penelitian [10] menggunakan teknik pembelajaran mendalam dan pembelajaran mesin pada dataset MIT/BIH-PED, model BSRDNN mendeteksi kantuk dari data EEG saluran tunggal dengan akurasi hingga 100%. Namun pada penelitian yang kita lakukan menggunakan teknik pembelajaran mendalam dan pembelajaran mesin pada data citra wajah secara real-time, model YOLOv12 mendeteksi kantuk melalui analisis gerakan mata dan frekuensi kedipan dengan akurasi hingga 100% menggunakan pendekatan computer vision.

Penelitian[11] Untuk meningkatkan keselamatan di jalan, deteksi kantuk memantau perilaku pengemudi dengan melacak karakteristik wajah, gerakan mata, pola kedipan, dan menguap. Sistem menemukan tanda-tanda kelelahan, seperti penutupan kelopak mata yang lambat, dan durasi kedipan yang meningkat.

Penelitian [12] Menggunakan sensor seperti monitor detak jantung, kamera, dan EEG, penelitian ini mengembangkan sistem yang dapat memantau dan mengingatkan pengemudi tentang tanda-tanda kantuk secara *Real-time*.

Sistem deteksi kantuk menggunakan teknologi untuk menemukan dan memperingatkan orang yang mungkin tertidur atau tidak waspada [13].

Tujuan dari penelitian ini adalah untuk mengukur kinerja algoritma YOLOv12 dalam mendeteksi tanda-tanda kantuk secara real-time dengan menggunakan metode visi komputer. Studi ini memberikan analisis kuantitatif terhadap efektivitas YOLOv12 dalam membedakan kondisi kantuk dan tidak kantuk pada wajah pengemudi dengan menggunakan metrik seperti mean average precision (mAP), precision, recall, dan skor F1. Pengembangan sistem deteksi kantuk berbasis visual yang dapat digunakan secara real-time, pengujian pengaruh kualitas dan keseimbangan dataset terhadap kinerja model, dan penggunaan teknik augmentasi untuk memperkaya data pelatihan adalah beberapa kontribusi utama dari penelitian ini. Studi ini juga menunjukkan bahwa YOLOv12 mendeteksi kantuk dengan akurat dan dapat digunakan untuk mencegah kecelakaan akibat

mengantuk pada sistem berbasis perangkat seluler atau kendaraan.

A. Dataset

Penggunaan dataset pada penelitian menggunakan dataset sekunder dari opensource kaggle [14] dan kemudian dianotasi secara manual menggunakan Roboflow. Tujuan pelabelan proses ini adalah untuk menghasilkan dataset yang terstruktur dan sesuai dengan klasifikasi objek yang diperlukan untuk mendukung akurasi dan efektivitas pelatihan model. Dari dataset tersebut didapatkan 2 jenis kondisi raut wajah atau klas yaitu: kantuk dan tidak mengantuk. Gambar pada dataset tersebut memiliki ukuran dimensi 204 x 1024pixel dengan jenis RGB dan sebanyak 5000 gambar. Dari dataset yang digunakan akan distribusikan ke tiap kelas nya masing-masing sebelum dan sesudah proses preprocessing seperti yang terlihat pada Table 1.

B. Preprocessing

Preprocessing merupakan [15] kumpulan data deteksi kantuk termasuk ekstraksi fitur seperti rasio aspek mulut, rasio aspek mata, sirkularitas pupil, dan rasio mulut di atas mata. Fitur-fitur ini sangat penting untuk mengklasifikasikan dan mendeteksi kantuk pengemudi dengan menggunakan berbagai algoritma pembelajaran mesin. Dalam [16] restorasi digital, pra pemrosesan melibatkan penggunaan filter gambar untuk meningkatkan kualitas gambar dengan menghilangkan noise dan artefak, mempertajam atau menghaluskan gambar, dan mengoreksi kontras dan kecerahan. Meskipun proses ini dapat meningkatkan gambar, ia juga dapat mengurangi detail kecil. Selain itu, penghapusan gambar ganda pada dataset hasil dari preprocessing dataset terlihat pada Tabel 1.

TABEL 1. PREPROCESSING DATASET DETEKSI KANTUK

Dataset	Preprosesing Dataset		
	Ngantuk	Tidak Ngantuk	Total
5000	3547	1453	5000

C. Data augmentasi alumentasi

Teknik augmentasi data (DA) meningkatkan model pembelajaran mendalam (DL) untuk mendeteksi kantuk pengemudi [17]. Ini dilakukan dengan mensimulasikan situasi dunia nyata yang mengganggu kinerja model, seperti gerakan pengemudi dan kurangnya pencahayaan. Studi ini menunjukkan bahwa DA meningkatkan akurat.

Pada penelitian yang dilakukan menggunakan augmentasi sebagai berikut: *Gaussian Blur* dan *Median Blur* dengan batas keaburan (blur limit) antara 3 hingga 7 piksel dan probabilitas penerapan sebesar 0,01. Selain itu, dilakukan konversi citra ke skala keabuan menggunakan metode *weighted average* dengan tetap mempertahankan tiga saluran warna (RGB). Untuk meningkatkan kontras lokal pada citra, juga diterapkan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE) dengan clip limit dalam rentang 1,0 hingga 4,0 serta ukuran grid 8x8, dan probabilitas penerapan sebesar 0,01.

TABEL II. HASIL DATA AUGMENTASI

Hasil Data Augmentasi		
Ngantuk	Tidak Ngantuk	Total
5580	1396	6976

D. Annotasi

Anotasi merupakan proses penting sebelum melaksanakan pelatihan model [18] dengan tujuan untuk merepresentasikan *Ground Truth* dan label kelas dari objek pada citra. Perilaku Yolo lebih mudah diterapkan bagi peneliti karena hanya membutuhkan anotasi manual sebagai data pelatihan. Kerangka kerja ini dibuat untuk mengidentifikasi perilaku visual yang berbeda dari rekaman video, yang memungkinkan pelatihan model yang efektif. Meskipun menunjukkan deteksi perilaku yang andal sebanding dengan anotasi manual, ada perbedaan, terutama dalam metrik deteksi berdasarkan peristiwa. Untuk mengatasi hal ini, pendekatan hibrida, di mana deteksi awal dikonfirmasi secara manual, dapat digunakan dengan kerangka kerja. Ini menghemat waktu selama proses anotasi [19].

TABEL III. LABEL KELAS PADA DATASET

Id kelas	Nama kelas
0	Ngantuk
1	Tidak Ngantuk

E. YOLO

YOLO adalah kerangka kerja deteksi objek real time yang menggunakan jaringan autoencoder tunggal untuk memprediksi selubung objek dan probabilitas kategori dalam satu lintasan maju, menawarkan efisiensi dan kecepatan, meskipun memiliki keterbatasan dalam akurasi dan deteksi objek kecil. YOLOv12 merupakan versi terbaru dari arsitektur YOLO yang memiliki karakteristik pada kecepatan deteksi, presisi tinggi, dan mudah dalam pelatihan data serta di implementasikan [20].

Penelitian ini menggunakan jaringan saraf konvolusional dan pembelajaran transfer, mencapai tingkat akurasi antara 90 dan 99,86% di beberapa kumpulan data, dan mencakup aplikasi seluler yang mudah digunakan untuk implementasi praktis. Penelitian ini menggunakan YOLOv12 untuk mendeteksi tanda kantuk secara real-time pada berbagai dataset. Teknologi visi komputer dan algoritma YOLOv12 dimaksudkan untuk digunakan dalam aplikasi berbasis perangkat seluler [21].

TABEL IV. PERBANDINGAN OBJEK DETEKTOR SIZE 640

Model	mAP val	M	B	Map/speed
YOLO12n	40.6	2.6	6.5	+2.1%/-9%
YOLO12s	48.0	9.3	21.4	+0.1%/+42%
YOLO12m	52.5	20.2	67.5	+1.0%/-3%
YOLO12l	53.7	26.4	88.9	+0.4%/-8%
YOLO12x	55.2	59.1	199.0	+0.6%/-4%

F. Matrix Evaluasi Performa

Matrik ini menilai kinerja deteksi objek, terutama untuk arsitektur YOLO, dengan menggunakan *Intersection over Union* (IoU) untuk mengukur tumpang tindih antara kotak pembatas

kebenaran yang diprediksi dan tanah. Ini mengevaluasi akurasi deteksi dan efektivitas melokalisasi objek [22].

Untuk model pengenalan objek, precision, recall, F1 score, dan mAP adalah indikator teknis yang populer untuk mengevaluasi kinerja secara keseluruhan. Ini dilakukan dengan menggunakan output IoU. Persamaan (1) hingga (4) mendefinisikan precision, recall, dan mAP [23], [24]. Nilai TP, TN, FP, dan FN telah diperoleh. Untuk mAP@IoU = 0,5, ambang batas adalah 0,5, dan untuk mAP: IoU = 0,5:0,95, ambang batas mengambil 10 nilai yang berbeda antara 0,5 dan 0,95 dalam langkah 0,05 [25].

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$mAP = \frac{1}{N} \sum_{i=1}^N AP \quad (3)$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

N : jumlah kueri

AP : rata-rata presisi

III. HASIL DAN PEMBAHASAN

Pelatihan model dilakukan menggunakan arsitektur YOLOv12 dengan konfigurasi parameter sebagai berikut: jumlah *epochs* sebanyak 100, *batch size* 4, dan *optimizer* menggunakan pelatihan model dilakukan menggunakan optimizer AdamW dengan learning rate sebesar 0.001667 dan momentum sebesar 0.9. Pengaturan parameter dibagi menjadi tiga grup, yaitu 113 parameter bobot dengan weight decay sebesar 0.0, 120 parameter bobot dengan weight decay sebesar 0.0005, dan 119 parameter bias dengan weight decay sebesar 0.0. Proses pelatihan dilanjutkan dari checkpoint terakhir di epoch ke-99 menuju total 100 epoch, dengan memuat model dari direktori.

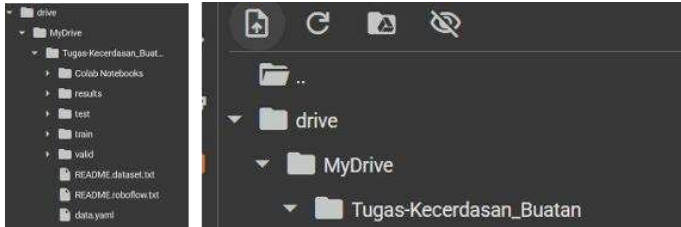
A. Pelatihan dataset

Dalam penelitian yang dilaksanakan pada pelatihan sebelumnya, dataset asli telah melalui tahap preprocessing seperti yang dijelaskan pada bagian preprocessing, dengan penjelasan lebih rinci tercantum pada gambar 1, 2, dan 3. Setelah dataset menjalani proses preprocessing, dilakukan augmentasi data dengan tujuan meningkatkan jumlah data dari 5002 menjadi 6976 gambar latih. Selanjutnya, dilakukan proses penandaan yang merupakan tahap krusial sebelum melaksanakan pelatihan model, di mana penandaan ini berfungsi untuk merepresentasikan ground truth dan label kategori dari objek yang terdapat dalam citra. Metode anotasi manual ini memfasilitasi penerapan algoritma YOLO, karena algoritma ini hanya membutuhkan file anotasi itu sebagai data pelatihan. Kerangka kerja ini dibuat untuk mengenali perilaku visual yang berbeda dari video yang direkam, yang memungkinkan pelatihan model dengan lebih efisien. Walaupun menunjukkan deteksi perilaku yang konsisten dan sebanding dengan anotasi manual, ada beberapa perbedaan, terutama dalam metrik deteksi yang berbasis peristiwa. Untuk menangani masalah tersebut, dapat digunakan pendekatan hibrida, di mana hasil deteksi awal dikonfirmasi secara manual, sehingga waktu dalam proses anotasi bisa lebih efisien.

File label dan gambar yang telah melalui proses anotasi dibagi menjadi data latih dan validasi dengan presentasi 80:20

yang diletakan pada directory data seperti yang terlihat pada gambar 4. Pembagian data train dan val yaitu Ngantuk 2486 data latih dan 704 data validasi, dan Tidak Ngantuk 1445 data latih dan 7 data validasi.

Tahapan selanjutnya adalah pembuatan file *custom.yaml* dan *training yolov12-custom.yaml* yang berisi informasi mengenai lokasi data pelatihan (*train*) dan validasi (*val*), jumlah kelas (*number of classes*), serta penamaan masing-masing kelas, sebagaimana ditunjukkan pada Gambar 1.



Gambar. 1. *Custom.yaml* dan *training yolov12-custom.yaml* beserta lokasi folder

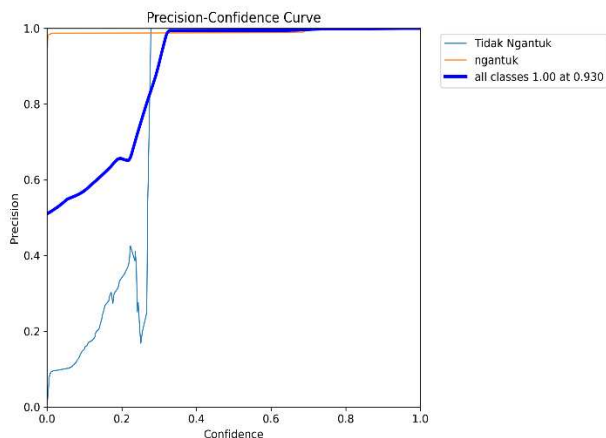
A. Evaluasi performa

Evaluasi performa yang digunakan pada penelitian untuk deteksi kantung menggunakan parameter, $mAP@IoU = 0,5$, ambang batas adalah 0,5, dan untuk $mAP: IoU = 0,5:0,95$, ambang batas mengambil 10 nilai yang berbeda antara 0,5 dan 0,95 dalam langkah 0,05. Secara keseluruhan dari kelas yang dilatih 0.732 ,0.62 seperti yang terlihat pada tabel 5.

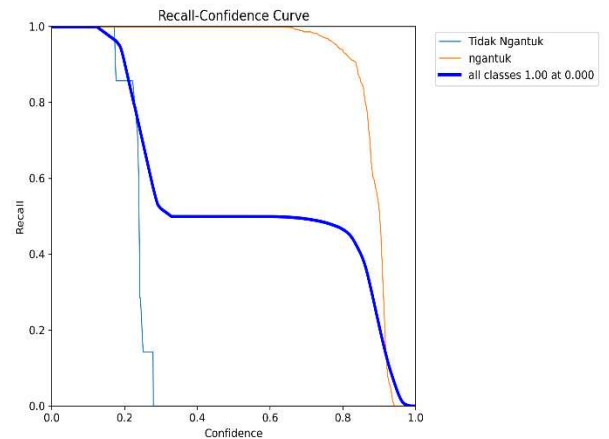
TABEL V. EVALUASI PERFORMA PADA DATASET

Class	Image	Instances	P	R	mAP50	mAP50-95
All	720	720	0.648	0.928	0.732	0.62
Tidak Ngantuk	7	7	0.324	0.857	0.482	0.426
Ngantuk	703	703	0.973	0.999	0.981	0.834

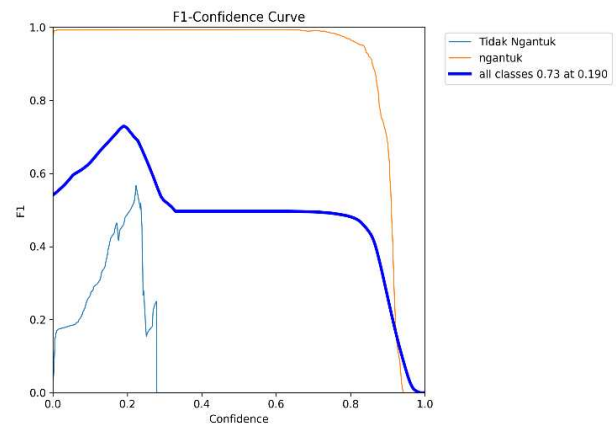
Grafik evaluasi performa lainnya F1-score, Precision, Recall, dan Confusion matrix terlihat pada gambar 2 dan 3.



(a)

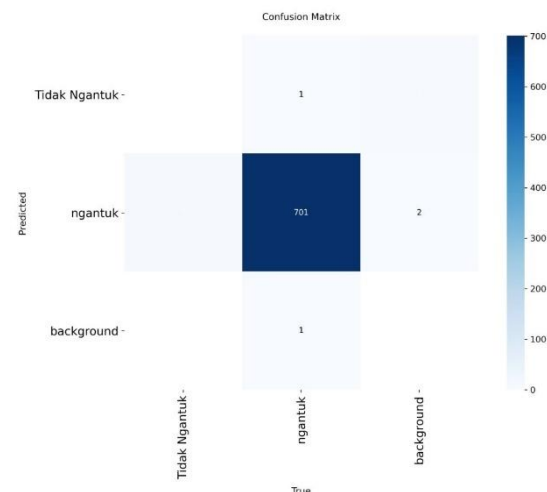


(b)



(c)

Gambar. 2. Grafik *Precision* (a), Grafik *Recall* (b), Grafik *F1-Score* (c)



Gambar. 3. *Confusion Matrix* latih dataset

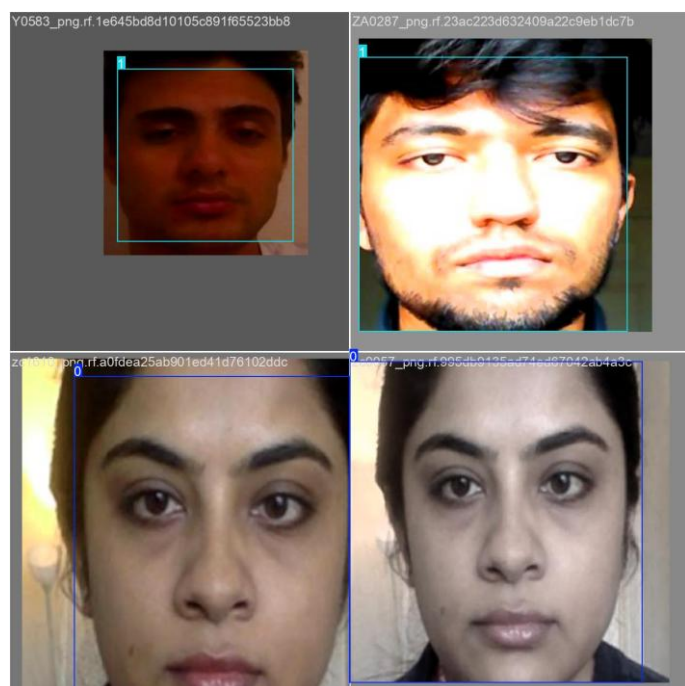
B. Pendeteksian Objek

Pada penelitian ini, proses pendeteksian objek dilakukan dengan memanfaatkan model terbaik (best weight) yang diperoleh dari hasil pelatihan sebelumnya. Model tersebut digunakan untuk menentukan posisi serta klasifikasi objek pada citra. Seluruh parameter yang digunakan dalam proses inferensi disajikan pada Tabel 6.

TABEL VI. DETEKSI OBJEK PARAMETER

Argumen	Detail Argumen	Default	Parameter yang digunakan
Weights	model.pt path(s)	Yolov12.pt	Best.pt
Augment	Augmented inference	True/False	False
Conf thresh	Object confidence	0.25	0.5
Img-size	Inference size	640	640

Percobaan pendeteksian objek serta pengklasifikasi deteksi kantuk terlihat pada gambar 4 dengan intensitas ujicoba 2 kali untuk setiap kelas.



Gambar. 4. Hasil deteksi dan klasifikasi objek deteksi kantuk

IV. KESIMPULAN

Proses deteksi dan klasifikasi objek penyakit tanaman kopi menggunakan YOLOv12 yang meliputi tahapan *preprocessing*, augmentasi data, serta anotasi objek, menghasilkan performa yang baik sebagaimana ditunjukkan pada Tabel 4. Pelatihan dataset dilakukan dengan konfigurasi *epochs* sebanyak 100, *batch size* 4, dan *optimizer* menggunakan optimizer AdamW dengan learning rate sebesar 0.001667, momentum sebesar 0.9 dan model YOLOv12. Penelitian ini dijalankan pada platform Google Colab dengan GPU Tesla T4, menghasilkan nilai evaluasi F1-score sebesar 0.72, Precision 1.00, Recall 1.00, mAP@IoU = 0,5, ambang batas adalah 0,5, dan untuk mAP: IoU = 0,5:0,95, ambang batas mengambil 10 nilai yang berbeda antara 0,5 dan 0,95 dalam langkah 0,05.

REFERENSI

- [1] D. Salem and M. Waleed, "Drowsiness detection in real-time via convolutional neural networks and transfer learning," *Journal of Engineering and Applied Science*, vol. 71, no. 1, p. 122, Dec. 2024, doi: 10.1186/s44147-024-00457-z.
- [2] S. P. M. A. AAR Lubis, "Sistem Pendeteksi Kantuk Berbasis Metode Haar Cascade Untuk Aplikasi Computer Vision".
- [3] P. J. Reddy, "Driver Drowsiness Detection Using CNN," *Int J Res Appl Sci Eng Technol*, vol. 12, no. 5, pp. 5047–5050, May 2024, doi: 10.22214/ijraset.2024.61620.
- [4] Thrupthi C P, Dr. Chitra K, and Mrs Harilakshmi V M, "Object Detection for Autonomous Vehicles Using YOLO Algorithm,"

- International Journal of Advanced Research in Science, Communication and Technology*, pp. 180–184, Nov. 2024, doi: 10.48175/IJARST-22539.
- [5] Manish Mate, Abhishek Sahu, Atharva Kadam, Rajat Tandulkar, and Arpita Agarwal, "Drowsiness Detection and Alert System," *International Research Journal on Advanced Engineering Hub (IRJAEH)*, vol. 2, no. 05, pp. 1361–1369, May 2024, doi: 10.47392/IRJAEH.2024.0188.
- [6] D. Salem and M. Waleed, "Drowsiness detection in real-time via convolutional neural networks and transfer learning," *Journal of Engineering and Applied Science*, vol. 71, no. 1, p. 122, Dec. 2024, doi: 10.1186/s44147-024-00457-z.
- [7] A. Chopra, N. Kumar, and R. K. Kaushal, "A comprehensive analysis of driver drowsiness detection techniques," in *Applied Data Science and Smart Systems*, London: CRC Press, 2024, pp. 134–139. doi: 10.1201/9781003471059-19.
- [8] V. Pandey, "Driver Drowsiness Detection (SLEEP GUARDIAN)," *Int J Res Appl Sci Eng Technol*, vol. 12, no. 11, pp. 1238–1244, Nov. 2024, doi: 10.22214/ijraset.2024.65023.
- [9] M. S. Sankar Reddy, S. Potturi, B. K. M. P. Harish, R. R. Malagi, and M. Yogeswar Reddy, "Artificial Intelligence Based Drowsiness Detection," in *2023 Fourth International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, IEEE, Dec. 2023, pp. 1–6. doi: 10.1109/ICSTCEE60504.2023.10585197.
- [10] H. O. D. Alkhalidi and S. Bilgen, "Drowsiness Detection Using Brain Signal Recognition Deep Neural Network (BSRDNN)," *International Journal of Safety and Security Engineering*, vol. 14, no. 3, pp. 753–764, Jun. 2024, doi: 10.18280/ijssse.140308.
- [11] M. Dey, M. Majhi, Y. Koda, B. Maji, and R. Chatterjee, "Drowsy Driver Detection System," *Int J Res Appl Sci Eng Technol*, vol. 12, no. 5, pp. 1488–1492, May 2024, doi: 10.22214/ijraset.2024.61832.
- [12] Mr. G. Rama Rao, V. Rahul SA, N. Sowjanya, M. Prashanth, M. Akshay Raj, and CH. Naresh, "Drowsiness Detection using Python," *International Journal of Advanced Research in Science, Communication and Technology*, pp. 431–436, May 2024, doi: 10.48175/IJARST-18164.
- [13] S. Venugopal, N. AJ, and S. K.S, "Drowsiness Detection System," *International Journal of Innovative Research in Engineering*, pp. 208–210, May 2023, doi: 10.59256/ijire.2023040374.
- [14] I. Nasri, M. Karrouchi, H. Snoussi, K. Kassmi, and A. Messaoudi, "Detection and Prediction of Driver Drowsiness for the Prevention of Road Accidents Using Deep Neural Networks Techniques," 2022, pp. 57–64. doi: 10.1007/978-981-33-6893-4_6.
- [15] S. Mittal, S. Gupta, Sagar, A. Shamma, I. Sahni, and N. Thakur, "Driver Drowsiness Detection Using Machine Learning and Image Processing," in *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, IEEE, Sep. 2021, pp. 1–8. doi: 10.1109/ICRITO51393.2021.9596358.
- [16] A. de Polo, "Digital picture restoration and enhancement for quality archiving," in *2002 14th International Conference on Digital Signal Processing Proceedings. DSP 2002 (Cat. No.02TH8628)*, IEEE, pp. 99–102. doi: 10.1109/ICDSP.2002.1027824.
- [17] G. M. Mohamed, S. S. Patel, and N. Naicker, "Data Augmentation for Deep Learning Algorithms that Perform Driver Drowsiness Detection," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 1, 2023, doi: 10.14569/IJACSA.2023.0140127.
- [18] Y. Jamtsho, P. Riyamongkol, and R. Waranusast, "Real-time Bhutanese license plate localization using YOLO," *ICT Express*, vol. 6, no. 2, pp. 121–124, Jun. 2020, doi: 10.1016/j.ict.2019.11.001.
- [19] A. H. Hang Chan *et al.*, "YOLO-Behaviour: A simple, flexible framework to automatically quantify animal behaviours from videos," Aug. 27, 2024, doi: 10.1101/2024.08.26.609387.
- [20] R. R. Ajith Babu, H. M. Dhushyanth, R. Hemanth, M. Naveen Kumar, B. A. Sushma, and B. Loganayagi, "Fast and Accurate YOLO Framework for Live Object Detection," 2023, pp. 555–567. doi: 10.1007/978-981-99-5166-6_38.
- [21] M. Fauzan Ridho, Fransiskus Panca, Welly Yandi, and Almeera Amsana Rachmani, "Drowsiness Detection in the Advanced Driver-Assistance System using YOLO V5 Detection Model," *ELECTRON Jurnal Ilmiah Teknik Elektro*, vol. 5, no. 1, May 2024, doi: 10.33019/electron.v5i1.136.
- [22] Y. Li, S. Li, H. Du, L. Chen, D. Zhang, and Y. Li, "YOLO-ACN: Focusing on Small Target and Occluded Object Detection," *IEEE Access*, vol. 8, pp. 227288–227303, 2020, doi: 10.1109/ACCESS.2020.3046515.

- [23] A. Ardiansyah, K. N. Qodri, D. Al Banna, and M. Z. Al-Baihaqi, "PEMANFAATAN SAM DAN YOLOV8 UNTUK DETEKSI DAN SEGMENTATION MRI TUMOR OTAK," *TEKNIMEDIA: Teknologi Informasi dan Multimedia*, vol. 5, no. 1, pp. 82–89, Jun. 2024, doi: 10.46764/teknimedia.v5i1.192.
- [24] A. Ardiansyah, *Buku Monograf Pengenalan Objek Dalam Bidang Kesehatan*. PT. Sonpedia Publishing Indonesia, 2024.
- [25] I. Ahmad *et al.*, "Deep Learning Based Detector YOLOv5 for Identifying Insect Pests," *Applied Sciences*, vol. 12, no. 19, p. 10167, Oct. 2022, doi: 10.3390/app121910167.