

Deteksi Jenis dan Kerusakan pada Penambat Rel Menggunakan *Night Vision* dan YOLOv8

Atniko Dwi Saputra¹, Muhammad Aditya Prasetyo², Muhammad Debi³, Muhammad Sadiqih Asbullah⁴,
Muhammad Azril Insanul Huda⁵, Vivi Aulia Husna Wahidah⁶

Jurusan Teknik Perkeretaapian Politeknik Negeri Madiun¹²³⁴⁵⁶

Jalan Ring Road Barat, Winongo, Manguharjo, Madiun, Jawa Timur

Atniko2181@gmail.com¹, madityap2004@gmail.com², ipa2muhammaddebi24@gmail.com³, sadiqihpenem@gmail.com⁴,
azrilhuda231@gmail.com⁵, auliavivi074@gmail.com⁶

Abstrak— Inspeksi penambat rel merupakan aspek krusial dalam menjaga keselamatan operasional perkeretaapian, karena adanya kecacatan seperti korosi dapat mengganggu integritas jalur dan berpotensi menyebabkan kecelakaan. Metode inspeksi manual memiliki keterbatasan dalam hal akurasi dan efisiensi, terutama dalam kondisi pencahayaan rendah. Penelitian ini bertujuan untuk menganalisis efektivitas sistem otomatisasi untuk mendeteksi cacat pada beberapa jenis penambat rel berbasis algoritma *deep learning* YOLOv8 yang diintegrasikan dengan teknik peningkatan pencitraan malam (*night vision*) untuk meningkatkan performa dalam lingkungan gelap. Dalam penelitian ini menggunakan 901 citra untuk pelatihan, validasi, dan pengujian, yang mencakup empat jenis penambat: DE Clip, KA Clip, E Clip, dan Wessel Clip, serta cacat korosi. Proses pengolahan citra data dilakukan menggunakan *platform* Roboflow yang juga digunakan untuk pelatihan dan evaluasi model. Kinerja sistem diukur dengan parameter seperti *Precision*, *Recall*, *mean Average Precision* (mAP), *Average Precision* (AP), dan *Frames Per Second* (FPS). Hasil menunjukkan bahwa sistem yang dikembangkan mampu melakukan deteksi dan klasifikasi secara andal dan *real-time*, bahkan dalam kondisi pencahayaan rendah. Pendekatan ini menawarkan solusi potensial untuk meningkatkan akurasi dan efisiensi inspeksi infrastruktur perkeretaapian.

Kata kunci—Penambat rel, klasifikasi kecacatan, YOLOv8, *night vision*, *deep learning*, pengolahan citra, dan deteksi *real-time*.

I. PENDAHULUAN

Kereta api merupakan salah satu moda transportasi yang penting dalam mendukung pertumbuhan ekonomi dan mobilitas masyarakat di berbagai negara, termasuk

Indonesia. Dalam menunjang keberlangsungan operasional transportasi kereta api dibutuhkan komponen-komponen infrastruktur rel, salah satunya adalah penambat rel. Penambat rel adalah komponen mekanis yang digunakan untuk mengikat rel ke bantalan, menjaga jarak dan posisi rel di bawah beban dinamis dari gerakan kereta [1]. Karena perannya yang sangat penting, kerusakan atau kegagalan pada penambat dapat menyebabkan pergeseran posisi rel, deformasi struktur lintasan, hingga risiko kecelakaan seperti anjlokkan. Oleh karena itu, inspeksi terhadap kondisi penambat secara rutin dan menyeluruh menjadi sangat penting untuk menjamin keselamatan perjalanan kereta api [1].

Pemeriksaan penambat rel konvensional umumnya dilakukan secara manual oleh petugas pemeliharaan. Meskipun efektif hingga batas tertentu, pemeriksaan manual memerlukan waktu yang lama, tenaga yang besar, dan rentan terhadap kesalahan manusia, terutama dalam kondisi pencahayaan yang kurang optimal seperti saat malam hari [2]. Di era otomatisasi seperti saat ini dibutuhkan sistem cerdas yang dapat membantu dalam proses inspeksi dan deteksi kerusakan secara *real-time*.

Seiring dengan kemajuan teknologi visi komputer dan pembelajaran mendalam (*deep learning*), pendekatan berbasis deteksi objek telah menjadi solusi efektif dalam otomatisasi inspeksi infrastruktur. Salah satu algoritma yang digunakan adalah YOLO (*You Only Look Once*), yang dikenal mampu melakukan deteksi secara *real-time* dengan kecepatan tinggi dan akurasi yang baik [3]. Dalam penelitian ini, digunakan versi terbaru dari algoritma tersebut, yaitu YOLOv8, yang

dirilis oleh *Ultralytics* dan telah mengalami penyempurnaan signifikan dibandingkan versi sebelumnya [4]. Keunggulan YOLOv8 salah satunya terletak pada kemampuan *plug-and-play* untuk tugas deteksi, segmentasi, dan pelacakan objek, menjadikannya sangat cocok digunakan dalam inspeksi infrastruktur transportasi yang dinamis seperti kereta api [3].

Namun, penggunaan model deteksi di malam hari atau ketika kurang pencahayaan menimbulkan tantangan baru. Citra dengan pencahayaan rendah biasanya memiliki kontras rendah dan noise tinggi, yang menyebabkan akurasi deteksi menurun. Oleh karena itu, sistem ini juga dilengkapi dengan modul *night vision enhancement*, yaitu proses pra-pemrosesan gambar untuk meningkatkan kualitas citra sebelum dilakukan deteksi [5]. Dengan menggabungkan deteksi objek berbasis YOLOv8 dan pemrosesan gambar malam hari melalui model *deep learning*, sistem ini dikembangkan untuk melakukan klasifikasi jenis penambat sekaligus mendeteksi kecacatannya secara otomatis. Tujuan dari sistem ini adalah mampu bekerja optimal pada kondisi malam hari atau pencahayaan rendah, untuk menguji performa *real-time* seperti *Precision*, *Recall*, mAP, *Average Precision* (AP), dan *Frame Per Second* (FPS) sebagai indikator keberhasilannya. Inovasi ini diharapkan dapat meningkatkan efisiensi, akurasi, dan kesinambungan dalam proses inspeksi infrastruktur perkeretaapian.

II. TINJAUAN PUSTAKA

A. Penambat Rel

Penambat rel (*rail fastening*) merupakan komponen penting dalam sistem jalan rel yang berfungsi untuk mengikat rel ke bantalan serta menjaga posisi dan kestabilan rel terhadap beban dinamis dari perjalanan kereta [1]. Kualitas dan kondisi penambat sangat berpengaruh terhadap keselamatan dan kenyamanan perjalanan. Kegagalan fungsi penambat dapat menyebabkan terjadinya pergeseran rel, deformasi lintasan, hingga kecelakaan seperti anjlokkan [1]. Oleh karena itu, kondisi penambat harus dipantau secara berkala dan menyeluruh.

B. Inspeksi Penambat Rel

Metode konvensional inspeksi penambat umumnya dilakukan secara manual oleh petugas di lapangan

dengan menggunakan alat bantu visual atau sensor dasar. Namun, metode ini membutuhkan waktu lama, melibatkan risiko *human error*, dan tidak efektif untuk kondisi lingkungan dengan pencahayaan rendah seperti pada malam hari atau area sulit dijangkau [6]. Seiring berkembangnya teknologi, pendekatan berbasis *computer vision* dan otomatisasi berbantuan kecerdasan buatan mulai dikembangkan untuk meningkatkan efisiensi dan akurasi inspeksi penambat [6].

C. YOLOv8 untuk Deteksi Objek

YOLO (*You Only Look Once*) merupakan salah satu algoritma deteksi objek *real-time* paling populer karena kemampuannya mendeteksi secara cepat dan akurat dalam satu tahap (*single-stage detector*). YOLOv8 adalah versi terbaru yang dikembangkan oleh *Ultralytics* dengan berbagai peningkatan signifikan, termasuk arsitektur *backbone CSPDarknet-P6*, *anchor-free detection*, dan dukungan multitask (*detection, segmentation, tracking*) [4]. YOLOv8 terbukti memiliki performa mAP dan FPS lebih tinggi dibanding YOLOv5 dan YOLOv6 pada dataset benchmark seperti COCO, serta lebih efisien dalam mendeteksi objek kecil pada latar kompleks [4].

D. Night Vision dalam Deteksi Gambar

Deteksi objek pada citra malam hari atau kondisi *low-light* menjadi tantangan besar karena noise tinggi, kontras rendah, dan detail visual yang hilang. Untuk mengatasi hal tersebut, digunakan pendekatan *night vision enhancement* sebagai langkah pra-pemrosesan, yaitu peningkatan kualitas gambar sebelum masuk ke model deteksi [5]. Beberapa metode yang digunakan termasuk Zero-DCE (*Zero-Reference Deep Curve Estimation*) [7], LLNet (*Low-Light Net*) [8], dan EnlightenGAN [9], yang mampu memperbaiki pencahayaan gambar malam tanpa kehilangan detail penting.

E. Integrasi Night Vision dan YOLOv8

Integrasi *preprocessing night vision* dan deteksi YOLOv8 membentuk sistem cerdas yang mampu melakukan deteksi objek secara optimal di kondisi minim cahaya. Sistem ini cocok digunakan dalam aplikasi inspeksi infrastruktur, termasuk deteksi cacat penambat rel di malam hari. Peningkatan akurasi hingga 17% terjadi ketika YOLO diintegrasikan dengan *preprocessing* citra malam [10]. Kombinasi ini

memberikan solusi praktis untuk inspeksi infrastruktur transportasi yang menuntut akurasi tinggi dan kecepatan *real-time*, sekaligus adaptif terhadap lingkungan dengan pencahayaan tidak ideal [11].

F. Precision dan Recall

Precision dan *recall* merupakan dua metrik utama yang digunakan dalam evaluasi kinerja sistem klasifikasi, termasuk dalam sistem deteksi objek. *Precision* mengukur ketepatan prediksi model, yaitu perbandingan antara jumlah prediksi positif yang benar dengan total prediksi positif. Sebaliknya, *recall* mengukur sejauh mana model mampu menemukan semua *instance* positif yang benar dari seluruh data [12]. Dalam konteks deteksi objek, *precision* tinggi berarti sedikit prediksi salah (*false positives*), sedangkan *recall* tinggi berarti sedikit objek yang terlewat (*false negatives*).

G. Average Precision (AP)

Average Precision (AP) adalah metrik yang mengukur area di bawah kurva *precision-recall* untuk satu kelas objek tertentu. AP digunakan untuk mengevaluasi performa model deteksi objek dengan menghitung rata-rata *precision* pada berbagai nilai [13]. AP memberikan ringkasan yang komprehensif atas performa deteksi untuk setiap kelas objek dalam dataset.

H. Mean Average Precision (mAP)

Mean Average Precision (mAP) merupakan rata-rata dari nilai AP untuk seluruh kelas objek yang diuji. Dalam praktik umum seperti pada COCO dan Pascal VOC dataset, mAP digunakan untuk memberikan gambaran keseluruhan terhadap performa deteksi objek dari model. mAP @0.5 adalah nilai mAP dengan IoU *threshold* 50%, sedangkan mAP@0.5:0.95 menunjukkan rata-rata dari berbagai nilai *threshold* antara 0.5 hingga 0.95, nilai AP yang tinggi menunjukkan bahwa model memiliki kemampuan yang baik dalam mendeteksi objek dengan presisi yang tinggi dan tanpa melewatkan banyak objek [14].

I. Frame Per Second (FPS)

FPS atau *Frame Per Second* adalah metrik yang mengukur efisiensi waktu atau kecepatan sistem deteksi objek dalam memproses gambar atau video secara *real-time*. FPS sangat penting dalam aplikasi waktu nyata seperti sistem pemantauan, kendaraan otonom, dan robotika. FPS dihitung dengan membagi jumlah total

frame yang diproses dengan total waktu pemrosesan. Model seperti YOLOv8n (nano), yang berukuran sangat ringan, mampu mencapai lebih dari 150 FPS pada GPU *high-end* (seperti NVIDIA RTX 3090) dan bahkan lebih dari 500 FPS pada CPU dengan dukungan engine optimasi seperti *DeepSparse* [15].

III. METODOLOGI

Penelitian ini bertujuan untuk mengembangkan sistem deteksi otomatis untuk klasifikasi jenis penambat rel dan mendeteksi kecacatannya, dengan menggunakan model *deep learning* berbasis YOLOv8. Metodologi penelitian ini terdiri dari beberapa tahapan utama, yaitu: pembuatan dataset, klasifikasi penambat rel dan jenis kecacatan, pelatihan model di *platform* Roboflow, serta evaluasi akurasi model menggunakan pembagian dataset.

A. Pembuatan Dataset

Dataset yang digunakan dalam penelitian ini dikumpulkan melalui pengambilan gambar penambat rel yang berada di jalur kereta api. Penambat rel yang dianalisis meliputi beberapa jenis, yaitu: de clip, ka clip, e clip, dan wessel clip yang merupakan jenis penambat umum digunakan dalam sistem rel kereta api.

Setiap gambar penambat rel dianotasi dengan menentukan *bounding box* dan label yang sesuai, serta jenis kecacatan yang ada pada penambat, seperti korosi. Gambar ini kemudian digunakan untuk pelatihan model YOLOv8 untuk melakukan klasifikasi jenis penambat dan deteksi kecacatannya secara otomatis.

B. Pembagian Dataset

Untuk memastikan model dapat belajar dengan efektif untuk menghasilkan keakurasian, dataset yang telah dikumpulkan dibagi menjadi tiga bagian utama yaitu: *training*, *validation*, dan *testing*. Pembagian dataset ini dilakukan dengan perbandingan sebagai berikut:




- 70% (604) gambar digunakan untuk pelatihan (*training*),
- 15% (162) gambar digunakan untuk validasi (*validation*), untuk memonitor kinerja model selama proses pelatihan,
- 15% (135) gambar digunakan untuk pengujian (*testing*), yang digunakan untuk evaluasi akhir kinerja model.

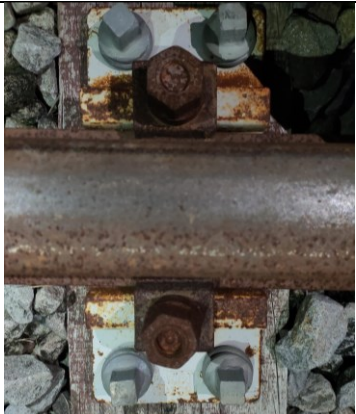
Dengan pembagian ini, dataset memberikan ruang untuk model belajar, sehingga data yang dihasilkan tidak terlalu spesifik dan memunculkan keakurasian yang sesuai.

C. Klasifikasi Penambat Rel dan Kecacatan

Pada tahap ini, setiap gambar penambat rel yang telah dianotasi dikelompokkan menjadi kategori yang sesuai. Jenis penambat yang akan diklasifikasikan dapat terlihat pada Tabel 1.


Tabel 1. Jenis-Jenis Penambat Rel

No	Jenis Penambat	Gambar
1	DE Clip	
2	KA Clip	
3	E Clip	

No	Jenis Penambat	Gambar
4	Wessel Clip	

Selain itu, kecacatan yang dapat terdeteksi pada penambat rel juga diklasifikasikan, seperti yang terlihat pada Tabel 2.

Tabel 2. Jenis Cacat pada Penambat Rel

No	Jenis Cacat	Gambar
1	<i>Corrosion</i> (korosi)	

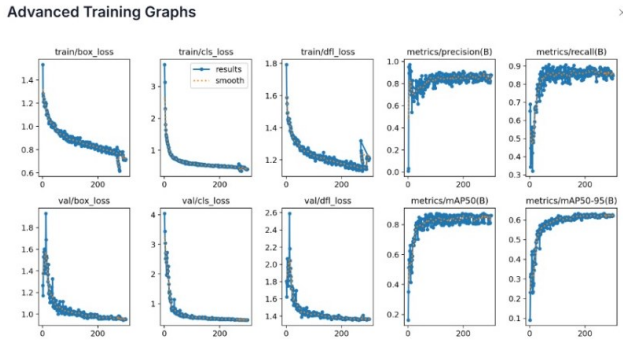
Setelah melalui proses klasifikasi, dilakukan proses *labeling* atau pemberian label yang jelas pada setiap gambar untuk memungkinkan model belajar mengidentifikasi perbedaan jenis penambat dan kerusakan yang terjadi.

D. Pelatihan Model di Platform Roboflow

Proses pelatihan dilakukan menggunakan *platform* Roboflow, yang merupakan *platform* penyedia alat dan infrastruktur untuk pelatihan model deteksi objek berbasis *deep learning*. Dataset yang telah *labeling* kemudian diunggah ke Roboflow untuk memulai pelatihan. *Platform* ini memungkinkan penggunaan model YOLOv8 yang telah disiapkan dengan parameter yang tepat untuk mendeteksi objek dengan akurasi tinggi.

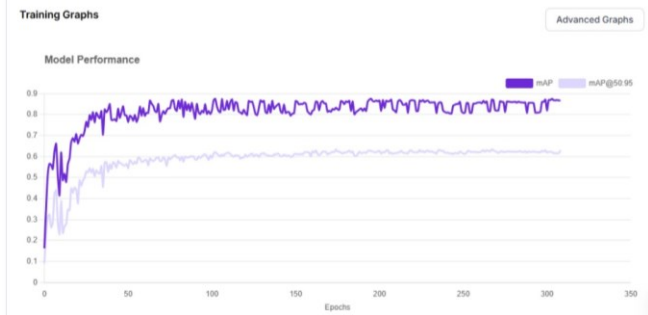
Pada tahap ini, proses pelatihan dilakukan dengan menggunakan dataset yang telah dibagi. Roboflow secara otomatis melakukan augmentasi gambar untuk meningkatkan kualitas dataset, termasuk rotasi, zoom,

dan perubahan pencahayaan untuk membuat model lebih akurat dalam menghadapi variasi gambar yang lebih luas. Dalam proses pelatihan model dihasilkan grafik pelatihan lanjutan (*Advanced Training Graphs*), yang berfungsi untuk menampilkan visualisasi performa model selama proses pelatihan secara detail. Berikut grafik pelatihan lanjutan terlihat pada Gambar 1.



Gambar 1. Grafik Pelatihan Lanjutan

Berdasarkan pelatihan model yang telah dilakukan menggunakan platform Roboflow, didapatkan grafik data pelatihan yang berfungsi untuk memantau dan mengevaluasi kinerja model selama proses pelatihan. Berikut grafik pelatihan (*Training Graphs*) terlihat pada Gambar 2.



Gambar 2. Grafik Pelatihan

E. Evaluasi Model

Setelah pelatihan selesai, model diuji menggunakan dataset pengujian yang terpisah. Evaluasi dilakukan dengan menghitung beberapa parameter untuk mengukur kinerja model, antara lain:

- *Precision*: Mengukur akurasi model dalam mengklasifikasikan objek yang terdeteksi.
- *Recall*: Mengukur seberapa baik model dapat menemukan semua objek yang relevan dalam gambar.
- *mAP (mean Average Precision)*: Menyediakan gambaran umum tentang akurasi model pada semua kelas yang terdeteksi.

- *AP (Average Precision)*: Mengukur ketepatan model dalam mendeteksi objek tertentu pada tiap kelas.
- *FPS (Frames Per Second)*: Mengukur kecepatan inferensi model dalam melakukan deteksi objek secara *real-time*.

Dengan menggunakan Roboflow, hasil akurasi model dapat dipantau dan diperbaiki lebih lanjut. Platform ini juga menyediakan visualisasi yang jelas tentang hasil deteksi objek, sehingga dapat memberikan keakuratan model dalam klasifikasi penambat rel dan mendeteksi kecacatan.

IV. HASIL DAN PEMBAHASAN

Dalam penelitian ini, sistem deteksi otomatis berbasis YOLOv8 dikembangkan untuk mendeteksi kecacatan serta mengklasifikasikan jenis penambat rel dalam kondisi pencahayaan rendah menggunakan modul *night vision enhancement*. Dataset yang digunakan diperoleh dari hasil pengambilan gambar penambat di lintasan rel, yang kemudian diberi anotasi manual menggunakan Roboflow.

A. Dataset dan Pelatihan Model

Dataset terdiri dari total 901 citra, dengan pembagian sebagai berikut:

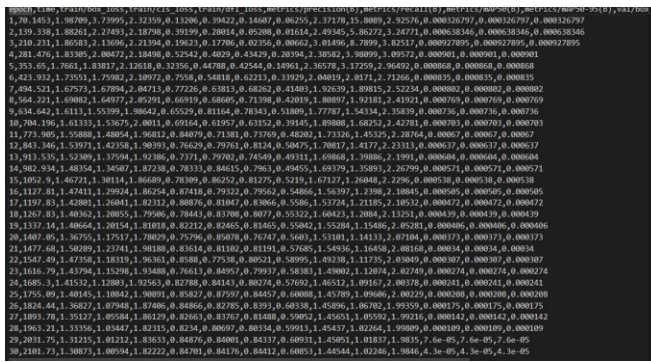
- 604 (70%) untuk pelatihan (*training*)
- 162 (15%) untuk validasi (*validation*)
- 135 (15%) untuk pengujian (*testing*)

Model dilatih menggunakan platform Roboflow yang mendukung format YOLOv8. Hasil dari pelatihan menunjukkan nilai parameter evaluasi, terlihat pada Tabel 3.

Tabel 3. Parameter Evaluasi

Parameter Evaluasi	Nilai
<i>Precision</i>	88,9%
<i>Recall</i>	87,8%
mAP@50	87,1%
AP	87,0%
FPS	60

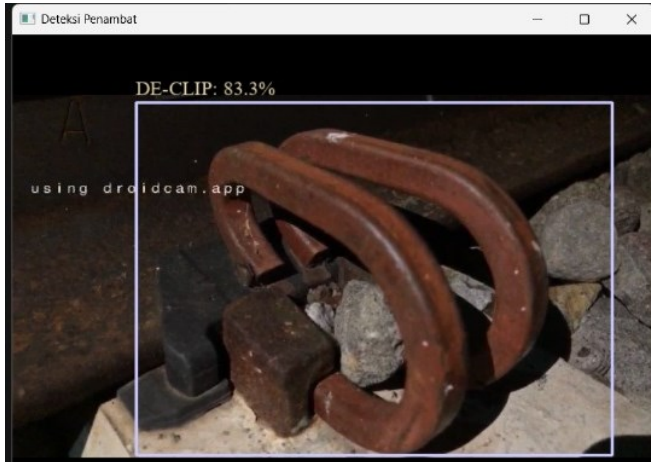
Setelah dilakukan pelatihan dataset diperoleh hasil data dengan format file *Comma Separated Values* (CSV), terlihat pada Gambar 3.



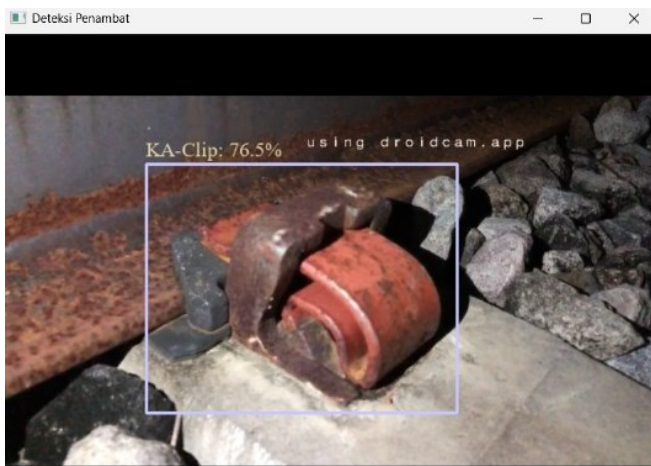
Gambar 3. Hasil Pelatihan Dataset

B. Visualisasi Hasil

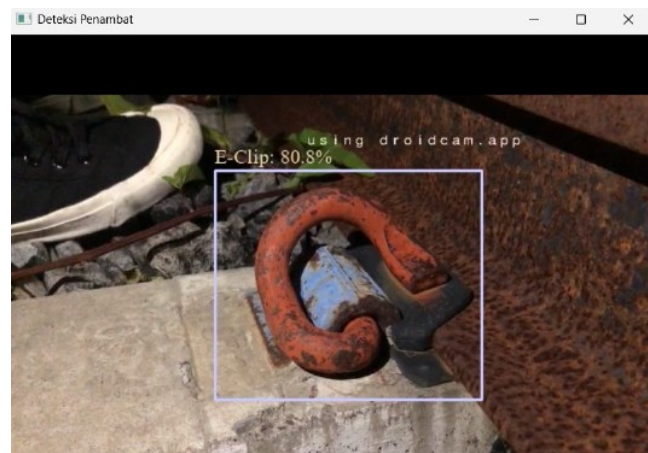
Sistem berhasil mendeteksi secara *real-time* jenis penambat dan keberadaan cacat seperti *corrosion* (korosi). Hasil ditampilkan melalui *bounding box* dengan label kelas dan tingkat kepercayaan (*confidence score*), berikut visualisasi hasil dari klasifikasi penambat dan cacatnya:



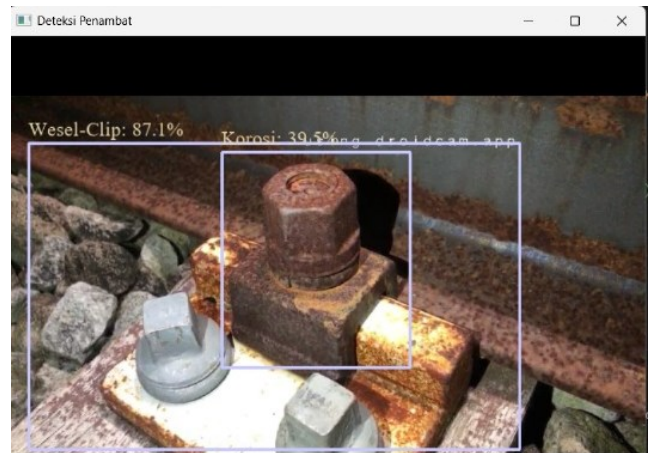
Gambar 4. DE Clip Tanpa Cacat



Gambar 5. KA Clip Tanpa Cacat



Gambar 6. E Clip Tanpa Cacat



Gambar 7. Wessel Clip Teridentifikasi Terjadi Korosi

V. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem berbasis YOLOv8 yang mampu melakukan klasifikasi jenis penambat rel serta mendeteksi kecacatan secara otomatis, khususnya dalam kondisi malam hari. Hasil pengujian menunjukkan bahwa sistem memiliki performa deteksi yang tinggi dengan nilai Map@50 sebesar 87,1% dan kecepatan pemrosesan mencapai 60 FPS.

Integrasi dengan modul *night vision enhancement* terbukti efektif dalam meningkatkan kualitas citra sebelum proses deteksi, sehingga memungkinkan akurasi tetap optimal dalam pencahayaan rendah. Oleh karena itu, sistem ini layak untuk diimplementasikan dalam inspeksi infrastruktur rel secara *real-time*, menggantikan metode manual yang selama ini digunakan.

Rekomendasi ke depan adalah pengembangan lebih lanjut pada deteksi jenis cacat secara spesifik dan

pengujian di lingkungan nyata dengan variasi kondisi cuaca dan pencahayaan.

DAFTAR PUSTAKA

- [1] B. Sadeghi and S. Barati, "Effect of railway track components on track lateral resistance," *Journal of Mechanical Science and Technology*, vol. 25, no. 9, pp. 2307–2314, Sep. 2011. [Online]. Available: <https://doi.org/10.1007/s12206-011-0512-1>
- [2] Pramestya, R.H. (2018) 'Deteksi dan Klasifikasi Kerusakan Jalan Aspal Menggunakan Metode YOLO Berbasis Citra Digital', *Institut Teknologi Sepuluh Nopember*, p. 91. Available at: http://repository.its.ac.id/59044/1/06111650010019-Master_Thesis.pdf.
- [3] R. Fernandez et al., "Benchmarking YOLOv8 for Real-Time Object Detection Tasks," *arXiv preprint arXiv:2302.05709*, 2023.
- [4] G. Jocher, J. Chaurasia, A. Qutaishat, and L. Stoken, "YOLO by Ultralytics," *Ultralytics*, 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [5] C. Guo et al., "Zero-Reference Deep Curve Estimation for Low-Light Image Enhancement," in *Proc. CVPR*, 2020.
- [6] L. Wang, M. Jia, Y. Liu, and X. Wu, "Automatic rail fastener inspection based on machine vision and deep learning," *IEEE Access*, vol. 8, pp. 223904–223913, 2020. [Online]. Available: <https://doi.org/10.1109/ACCESS.2020.3043152>
- [7] C. Guo, C. Li, J. Guo, C. Loy, and D. Lin, "Zero-DCE: Zero-Reference Deep Curve Estimation for Low-Light Image Enhancement," in *Proc. CVPR*, 2020, pp. 1780–1789. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/papers/Guo_Zero-Reference_Deep_Curve_Estimation_for_Low-Light_Image_Enhancement_CVPR_2020_paper.pdf
- [8] L. Lore, A. Akintayo, and S. Sarkar, "LLNet: A Deep Autoencoder Approach to Natural Low-light Image Enhancement," *Pattern Recognition*, vol. 61, pp. 650–662, Jan. 2017. [Online]. Available: <https://doi.org/10.1016/j.patcog.2016.06.008>
- [9] Y. Jiang, X. Gong, D. Cheng, Y. Fang, and Z. Wang, "EnlightenGAN: Deep Light Enhancement Without Paired Supervision," *IEEE Transactions on Image Processing*, vol. 30, pp. 2340–2349, 2021. [Online]. Available: <https://doi.org/10.1109/TIP.2021.3054884>
- [10] Z. Deng, J. Zheng, and M. Zhang, "Low-Light Image Enhancement Using YOLO and GAN Fusion for Nighttime Object Detection," in *Proc. International Conf. on Computer Vision Applications (ICCV-A)*, 2021. [Online]. Available: <https://doi.org/10.1109/ICCV-A52855.2021.9455192>
- [11] M. Shamsuddin, R. Ishak, et al., "Performance Analysis of YOLO with Night-Vision Enhanced Images in Rail Component Detection," *Journal of Advanced Transportation*, vol. 2022, Article ID 7891036, 2022. [Online]. Available: <https://doi.org/10.1155/2022/7891036>
- [12] Saito, T., & Rehmsmeier, M. (2015). *The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets*. PLoS ONE, 10(3), e0118432. <https://doi.org/10.1371/journal.pone.0118432>
- [13] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). *The Pascal Visual Object Classes (VOC) Challenge*. International Journal of Computer Vision, 88(2), 303–338. <https://doi.org/10.1007/s11263-009-0275-4>
- [14] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, & Zitnick, C. L. (2014). *Microsoft COCO: Common Objects in Context*. In European Conference on Computer Vision (ECCV), 740–755. https://doi.org/10.1007/978-3-319-10602-1_48
- [15] Neural Magic. (2023). YOLOv8 on CPU: Real-Time Performance Benchmarks. Retrieved from <https://neuralmagic.com/blog/yolov8-cpu-benchmark>