

Analisis Deteksi Kerusakan pada Jalan Aspal menggunakan Deep Learning untuk Mendukung Efisiensi Biaya dan Waktu dalam Pemantauan BerkelaJutan

Lino Garda Denaro¹, Resmana Lim²

¹Prodi Pendidikan Profesi Insinyur Universitas Kristen Petra dan Department of Geography, National Taiwan University
dgarda.lino@gmail.com

²Prodi Teknik Elektro dan Prodi Pendidikan Profesi Insinyur, Universitas Kristen Petra
resmana@petra.ac.id

Abstract— Road damage, including cracks and other deformities, must be detected early because they are a sign of wear, deterioration, or underlying structural problems that could jeopardize the long-term stability and safety of road infrastructure. The goal of this research is to create an automatic inspection system that can efficiently identify and categorize road damage using deep learning and image processing. In order to automatically classify photos into damaged/undamaged classes or surface deformation, we utilize deep learning models that leverage Convolutional Neural Networks (CNN) and the ResNet architecture to discover inherent properties of road surface damage. This YOLO (You Only Look Once) model makes use of transfer learning techniques for both Indonesian national road data and global data. In contrast to traditional methods, the YOLO model is used in this work to identify and categorize road damage in real-time, enabling immediate detection on a variety of road surfaces. This technology can be utilized for regular road infrastructure inspections and ongoing road condition monitoring due to the YOLO's quick and effective object recognition capabilities. The evaluation results demonstrate that the YOLO model not only detects road damage with high accuracy but also exhibits superior detection speed compared to conventional methods. This approach was tested using available road damage datasets obtained in the national road in Palembang, South Sumatera, Indonesia. The findings of this study show how YOLO can be used for automated road inspections, supporting road maintenance and safety initiatives while assisting in the early detection and repair of damage.

Keywords: Road damage detection, Yolo artificial intelligence, time and scale efficiency

Abstrak— Deteksi dini kerusakan jalan, seperti retakan dan deformasi lainnya, sangat penting karena berfungsi sebagai indikator awal keausan, kerusakan, atau masalah struktural yang mendasari yang dapat membahayakan keselamatan dan daya tahan infrastruktur jalan. Penelitian ini bertujuan untuk mengembangkan sistem inspeksi otomatis berbasis pembelajaran mendalam (deep learning) dan pemrosesan citra untuk mendeteksi dan mengklasifikasikan kerusakan jalan secara efektif. Dalam proyek ini, kami menggunakan model pembelajaran mendalam dengan CNN (Convolutional Neural Networks) dan arsitektur ResNet untuk mempelajari fitur intrinsik kerusakan permukaan jalan melalui gambar, yang kemudian digunakan untuk klasifikasi otomatis ke dalam kelas rusak/tidak rusak atau deformasi permukaan. Model YOLO ini menggunakan metode pembelajaran transfer dari data global dan untuk data jalan nasional yang ada di Indonesia. Tidak seperti pendekatan konvensional, penelitian ini menerapkan model YOLO (You Only Look Once) untuk mendeteksi dan mengklasifikasikan kerusakan jalan secara real-time, memungkinkan deteksi langsung pada berbagai permukaan jalan. Dengan kemampuan YOLO dalam deteksi objek yang cepat dan efisien, sistem ini dapat digunakan dalam inspeksi infrastruktur jalan secara rutin dan pemantauan kondisi jalan secara berkelaJutan. Pendekatan ini telah dievaluasi menggunakan dataset kerusakan jalan yang tersedia di jalan nasional di Palembang, Sumatera Selatan, Indonesia, dan hasil evaluasi menunjukkan bahwa model YOLO tidak hanya mendeteksi kerusakan jalan dengan akurasi tinggi, tetapi juga menunjukkan kecepatan deteksi yang unggul dibandingkan dengan metode tradisional. Hasil penelitian ini menunjukkan potensi signifikan penggunaan YOLO untuk inspeksi jalan otomatis, membantu mengidentifikasi dan menangani kerusakan dini serta mendukung upaya pemeliharaan dan keselamatan jalan.

Kata Kunci : Deteksi kerusakan jalan, kecerdasan buatan Yolo, efisiensi waktu dan skala.

I. PENDAHULUAN

Pembangunan infrastruktur yang semakin cepat di seluruh dunia, seperti jalan raya, jembatan, dan fasilitas umum lainnya, memerlukan sistem pemeliharaan yang efisien dan reguler agar infrastruktur tersebut tetap aman dan berfungsi dengan layak. Sementara infrastruktur ini semakin meluas dan semakin lama, mengalami kejemuhan dan risiko kerusakan atau kegagalan struktural yang seiring

bertambahnya waktu meningkat. Sehingga dapat menyebabkan efek berbahaya, termasuk kecelakaan dan kematian oleh pengendara [1]. Indikator dominan masalah tersebut adalah kerusakan pada permukaan jalan, yang meliputi retakan, lubang, deformasi, dan lain sebagainya [2].

Kerusakan ini sebagian besar disebabkan oleh berbagai faktor, seperti *tensile* material, pergantian cuaca dan musim, curah hujan, dan mobil dengan membawa muatan berat [3],

4]. Oleh karena itu, deteksi dini dan pemantauan teratur permukaan jalan diperlukan untuk mencegah kerusakan yang parah, mendeteksi kerusakan saat dini, dan menjaga keselamatan dan keberlanjutan infrastruktur [5].

Dalam praktik konvensional, aplikasi jalan yang rusak dievaluasi dengan melakukan pengawasan secara visual oleh inspektor jalan. Kendati metode ini dapat menghasilkan deteksi yang akurat, metode ini hanya praktis dalam kasus-kasus tertentu karena membutuhkan banyak waktu, biaya dan perlu keahlian yang signifikan untuk dikerjakan oleh seorang yang professional. Sehingga perlu ribuan professional untuk mengatasi masalah ini di seluruh jalan di Indonesia. Dan juga, ini tidak memungkinkan di jalan yang luas dan terpencil [6]. Sementara itu, kebutuhan akan solusi otomatis yang lebih efisien dan praktis dalam mendeteksi dan memantau kerusakannya secara *real-time* berdampak pada infrastruktur jalan kritis [7-9].

Dengan perkembangan pesat teknologi kecerdasan buatan (AI) dan pengolahan citra digital, kini muncul peluang untuk mengotomatisasi proses deteksi kerusakan jalan menggunakan metode *deep learning* [10, 11]. Model jaringan saraf konvolusi CNN (*Convolutional Neural Network*) telah terbukti sangat efektif dalam mengenali pola dan fitur pada citra atau gambar, sehingga menjadi salah satu pendekatan yang menjanjikan untuk identifikasi dan klasifikasi kerusakan jalan [12, 13]. Salah satu model yang telah mendapatkan perhatian luas dalam konteks deteksi objek adalah arsitektur yang dikembangkan oleh Ultralytics (<https://github.com/ultralytics>), dengan nama YOLO (*You Only Look Once*) [14, 15]. YOLO mampu mendeteksi objek secara cepat dan efisien dengan satu tahap prediksi, sehingga sangat cocok untuk aplikasi deteksi secara *real-time*. Penggunaan YOLO dalam deteksi kerusakan jalan memberikan solusi yang lebih cepat dan akurat dibandingkan metode konvensional, karena model ini dapat memproses citra ataupun video dengan kecepatan tinggi dan mengenali berbagai jenis kerusakan permukaan jalan secara akurat [16].

Penelitian ini bertujuan untuk mengembangkan sistem inspeksi otomatis berbasis *deep learning* dengan model arsitektur YOLO untuk mendeteksi dan mengklasifikasikan kerusakan jalan pada berbagai permukaan infrastruktur jalan [17]. Dalam proyek ini, penulis mengintegrasikan model CNN dengan arsitektur ResNet atau disebut YOLO untuk mempelajari fitur intrinsik dari kerusakan jalan melalui citra permukaan, yang memungkinkan klasifikasi secara otomatis menjadi dua kelas, rusak dan tidak rusak, termasuk retakan, lubang, dan deformasi lainnya. Pendekatan ini berfokus pada pemanfaatan *dataset* kerusakan jalan yang tersedia secara global di berbagai negara yang dapat diakses secara *open source*. Dengan tambahan data kerusakan jalan yang dikoleksi di Indonesia, maka penulis mengembangkan *system transfer learning* untuk melatih dan menguji model arsiteknya, serta mengoptimalkan kinerja desain YOLO dalam hal akurasi dan kecepatan deteksi. Selain itu, penelitian ini akan mengkaji keunggulan YOLO dibandingkan dengan model deteksi kerusakan lainnya, serta mengevaluasi sejauh mana sistem ini dapat diimplementasikan dalam skenario nyata, seperti pemantauan kondisi jalan secara berkala.

Dengan kemampuan model arsitek AI YOLO untuk melakukan deteksi objek secara cepat dan efisien, sistem

yang diusulkan dapat digunakan sebagai alat bantu dalam inspeksi rutin infrastruktur jalan, mengurangi beban kerja inspeksi manual yang dilakukan oleh para profesional, dan memberikan informasi yang lebih mendalam mengenai kondisi jalan secara *real-time*. Hasil dari penelitian ini diharapkan dapat menunjukkan potensi besar dari penggunaan YOLO dalam inspeksi infrastruktur jalan otomatis dan memberikan kontribusi signifikan terhadap pengembangan metode pemeliharaan infrastruktur jalan yang lebih aman, efisien, dan secara finansial sangat ekonomis atau terjangkau. Dengan sistem deteksi kerusakan yang akurat dan cepat, pemangku kepentingan di bidang konstruksi dan infrastruktur jalan dapat mengambil langkah pencegahan yang lebih tepat waktu untuk meminimalkan risiko kerusakan dan perbaikan, sehingga mendukung keberlanjutan dan keselamatan infrastruktur jangka panjang.

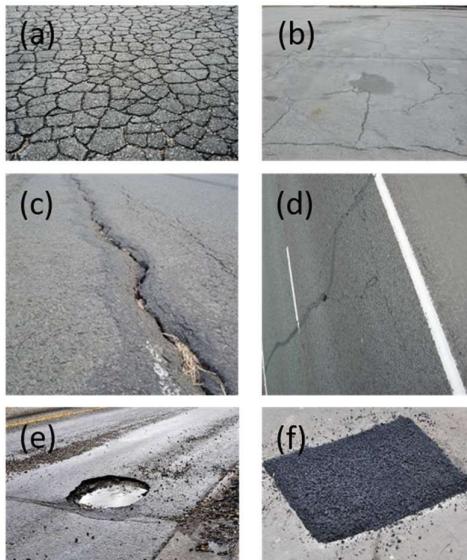
II. LANDASAN TEORI

A. Keretakan Jalan

Keretakan jalan dibagi menjadi beberapa jenis yang meliputi keretakan *alligator*, *block*, longitudinal, *transverse*, *tambalan* (*patching*), dan lubang (*pothole*) [2]. Contoh-contoh keretakan tersebut disajikan pada Gambar 1. Retakan *alligator*, yang disajikan pada Gambar 1(a), juga dikenal sebagai retak lelah. Retakan *alligator* merupakan formasi yang sangat umum terhadap permukaan perkerasan aspal dan tampak seperti sisik aligator. Jenis retak ini dihasilkan oleh serangkaian faktor termasuk beban lalu lintas yang berat yang menekan struktur perkerasan, fluktuasi suhu dalam pemuaian dan penyusutan, serta kelembapan infiltrasi yang merusak struktur perkerasan. Retakan yang setelahnya dihasilkan oleh faktor-faktor ini menghasilkan pola poligonal kecil yang mencakup permukaan dan merusak struktur [1]. Retakan *block* yang digambarkan pada Gambar 1(b) adalah jenis retakan permukaan yang muncul di perkerasan aspal sebagai pola berbentuk, seperti persegi panjang atau persegi [2]. Retakan ini biasanya terjadi karena penuaan aspal, yang menyebabkan hilangnya fleksibilitas. Retakan *block* sering diperburuk oleh fluktuasi suhu yang ekstrim, sehingga menyebabkan perkerasan memuai dan menyusut. Faktor-faktor seperti kadar aspal yang tidak mencukupi, drainase yang buruk, dan lalu lintas yang padat juga dapat menyebabkan perkembangan retakan blok. Tampilan retakan ini dapat bervariasi dalam ukuran, biasanya membentuk *block* yang lebih besar dan saling berhubungan. Meskipun retakan ini mungkin tidak langsung membahayakan, retakan ini dapat memungkinkan infiltrasi air, yang menyebabkan kerusakan lebih lanjut seiring waktu. Retakan longitudinal adalah retakan yang sejajar dengan garis tengah jalan atau perkerasan, biasanya tampak seperti garis lurus dan sempit, Gambar 1(c). Retakan ini dapat terjadi pada permukaan aspal dan beton dan sering kali disebabkan oleh faktor-faktor seperti fluktuasi suhu yang lama, menyebabkan pemuaian dan penyusutan, serta pergerakan tanah dasar karena pergeseran tanah di bawahnya. Praktik konstruksi yang buruk, seperti pemasatan yang tidak memadai atau penempatan material yang tidak tepat, juga dapat menyebabkan pembentukannya. Sama seperti kerusakan *block*, infiltrasi air dapat memperparah kerusakan ini.

Retakan *transverse* mengacu pada retakan yang berjalan tegak lurus dengan garis tengah jalan atau perkerasan, lihat pada Gambar 1(d). Retakan ini muncul seperti garis lurus di permukaan dan umumnya ditemukan di perkerasan aspal dan beton, sama seperti *block*, dan *longitudinal*. Retakan melintang sering terjadi karena fluktuasi suhu ekstrim yang mempengaruhi perkerasan, pemuaian dan penyusutan pada jalan. Tekanan yang diterima dari arah-arah melintang menyebabkan tekanan yang mengakibatkan retakan. Faktor penyebab lainnya termasuk infiltrasi kelembaban, drainase yang buruk, dan proses penuaan alami bahan perkerasan.

Pothole adalah cekungan atau lubang di permukaan jalan yang disebabkan oleh keausan lalu lintas, serta faktor lingkungan seperti kelembapan dan fluktuasi suhu, terlihat pada Gambar 1(e). Saat air merembes ke dalam retakan di trotoar, air dapat membeku dan mengembang selama cuaca dingin, menyebabkan material di sekitarnya pecah. Seiring waktu, tekanan lalu lintas yang berulang dan efek cuaca dapat memperbesar retakan ini, yang menyebabkan terbentuknya lubang jalan, yang dapat menimbulkan bahaya keselamatan dan merusak kendaraan. *Patchig* adalah metode umum yang digunakan untuk memperbaiki lubang jalan, seperti digambarkan pada Gambar 1(f). Penambalan bisa dikategorikan sebagai kerusakan karena terlihat tidak selevel dengan rerata jalan di sekitarnya sehingga dapat menimbulkan tumpukan aspal yang dapat mengganggu keselamatan pengendara lalu lintas.



Gambar 1. Macam-macam kerusakan pada jalan, (a) *alligator*, (b) *block*, (c) *longitudinal*, (d) *transverse*, (e) *pothole*, dan (f) *patching*.

B. Pendekripsi Kerusakan Jalan

Deteksi kerusakan jalan konvensional melibatkan berbagai metode, biasanya dilakukan melalui inspeksi visual di lapangan [4]. Tim inspeksi memeriksa permukaan jalan untuk mengidentifikasi masalah seperti keretakan *alligator*, *block*, *longitudinal*, *transverse*, tambalan, dan lubang [1].

Setelah kerusakan terdeteksi, dimensi dan kedalaman kerusakan diukur untuk menilai tingkat keparahannya. Kerusakan yang teridentifikasi kemudian diklasifikasikan berdasarkan jenis dan tingkat keparahannya, yang kemudian akan dilakukan perbaikan dan pemeliharaan. Semua temuan

dicatat dalam laporan, didokumentasikan termasuk koordinatnya untuk analisis lebih lanjut, dan data historis mengenai upaya pemeliharaan sebelumnya dievaluasi untuk memahami pola kerusakan. Meskipun metode konvensional efektif, metode tersebut memiliki keterbatasan dalam hal akurasi dan kecepatan dibandingkan dengan teknologi modern, seperti pemantauan sensor atau perangkat lunak analisis data [3]. Namun demikian, metode tersebut tetap menjadi bagian penting dari proses pemeliharaan jalan, terutama di area yang kurang dapat diakses oleh teknologi canggih atau daerah terpencil.

Selain inspeksi visual, metode konvensional juga dapat menggabungkan teknik manual, seperti pemeriksaan dan pengujian, untuk menilai kondisi perkerasan jalan. Personal professional dipilih untuk mengenali berbagai jenis kerusakan secara akurat, yang mana sangat penting dalam hal pendekripsi yang efektif. Sehingga, menggabungkan metode konvensional dengan teknologi modern dapat meningkatkan keseluruhan proses penilaian, memastikan perbaikan tepat waktu, dan perbaikan hasil dan analisa dengan tujuan untuk memperpanjang umur jalan raya.

Pemantauan jalan dan analisis, akhir-akhir ini dapat dilakukan dengan menggunakan perangkat lunak seperti pengembangan kecerdasan buatan. Selain dengan metode sensor, metode kecerdasan buatan tersebut menggabungkan teknik inspeksi kerusakan dengan cara pengambilan gambar secara otomatis yang diletakkan pada sebuah kendaraan [15].

Ketika membandingkan metode deteksi kerusakan jalan konvensional dengan teknik inspeksi visual seperti kecerdasan buatan *deep learning*, YOLO dapat digunakan sebagai salah satu alternatifnya [17]. Dengan menggunakan metode ini, perbedaan signifikan dalam efisiensi, akurasi, dan skalabilitas muncul.

Metode konvensional sangat bergantung pada inspektur manusia yang melakukan penilaian visual, yang dapat memakan waktu dan subjektif, yang mengarah pada potensi kelalaian, terutama di jaringan jalan yang luas. Pengumpulan data dilakukan secara manual, yang mengakibatkan ketidakstabilan dan keterlambatan dalam menangani kerusakan. Sebaliknya, teknik kecerdasan buatan model YOLO memanfaatkan deteksi otomatis melalui gambar yang diambil oleh *drone* atau kamera genggam, yang secara signifikan mempercepat proses inspeksi [16, 18].

Model pembelajaran mendalam seperti YOLO dapat secara akurat mengidentifikasi dan mengklasifikasikan berbagai jenis kerusakan, mengurangi risiko kesalahan manusia. Otomatisasi ini memungkinkan area yang luas untuk survei dengan cepat, dengan data yang diproses secara *real time*. Di sisi lain, algoritma canggih dapat menganalisis data historis untuk memprediksi kebutuhan pemeliharaan di masa mendatang, yang memungkinkan pendekatan secara objektif. Meskipun biaya pengaturan awal untuk teknologi YOLO mungkin lebih tinggi, penghematan jangka panjang dalam tenaga kerja dan peningkatan efisiensi yang didapat akan lebih besar. Dalam penggunaan YOLO, inspeksi metode konvensional tidak semerta-merta dapat diabaikan, melainkan tenaga profesional dapat menganalisa hasil dari inspeksi modern dan dapat dilakukan secara lebih efektif.

C. Model YOLOv8

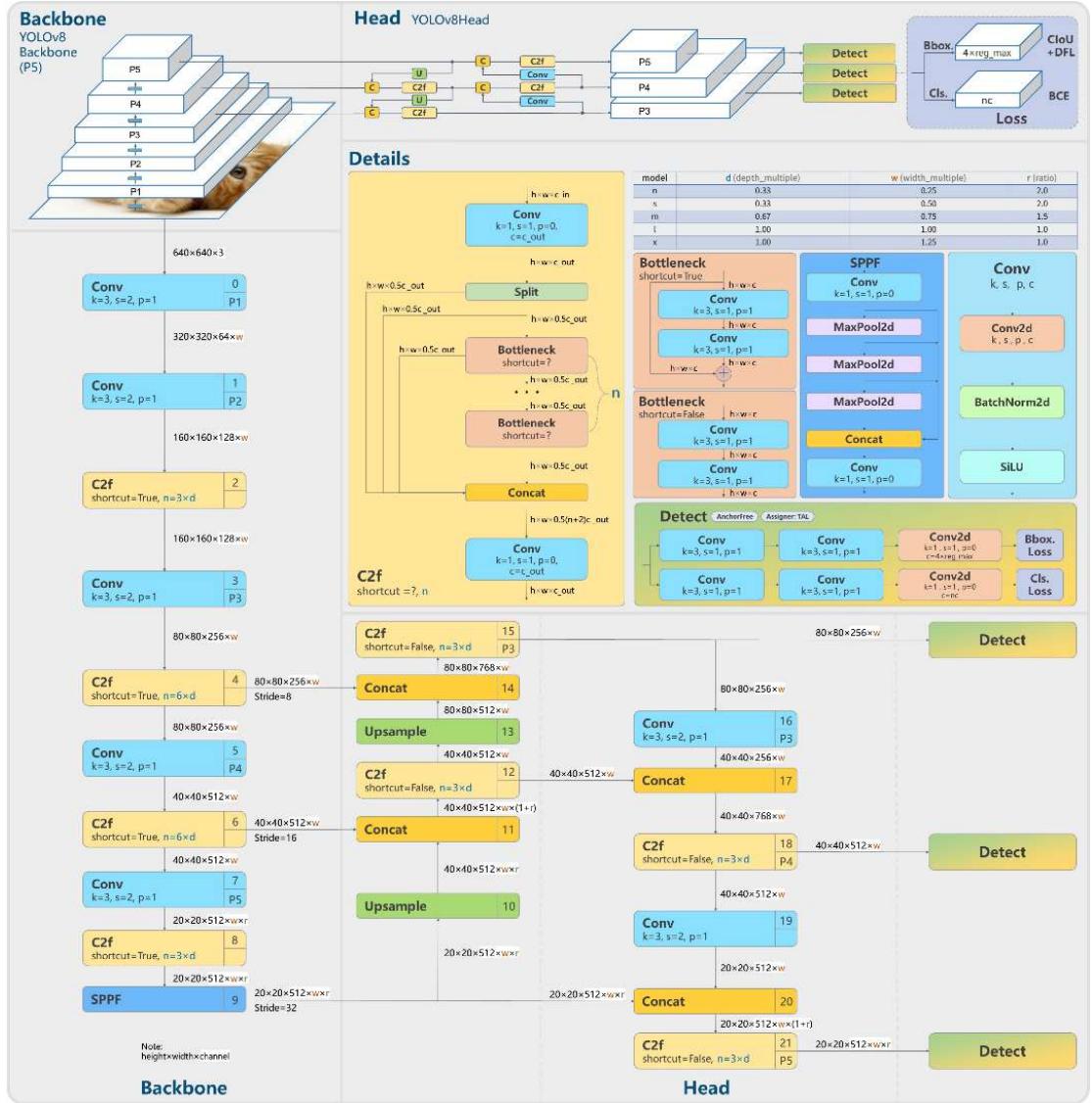
Model yang digunakan dalam pendekripsi kerusakan jalan adalah dengan YOLO versi v8 (YOLOv8). Untuk menjalankan model ini, penelitian ini menggunakan komputer Desktop Windows 11 Intel Core i7 dengan RAM 64GB dan dilengkapi dengan kecepatan komputasi grafik NVIDIA GeForce RTX 3060 Ti. Model ini dilengkapi dengan jaringan *backbone* yang telah didefinisikan untuk mengekstraksi representasi fitur multi-skala secara mendetail. Jaringan *backbone* ini adalah jaringan dasar dalam model ini karena terdapat proses *up-scale* atau dalam kata lain adalah *pyramid pooling* yang mana digambarkan pada Gambar 2. Jaringan *backbone* ini digambarkan pada simbol P1, P2, P3, P4, dan P5 pada sisi kiri gambar yang membentang vertikal ke bawah dengan kotak bertuliskan *Conv* warna biru. Pada proses ini adalah proses konvolusi dengan input 640x640x3, yang mana secara berurutan adalah panjang, lebar, dan layer; layer yang digunakan adalah berjumlah tiga yaitu *red*, *green*, *blue* (RGB). Kemudian dilanjutkan dengan *backbone* P2 dengan dimensi 320x320x64xw, dimana dimensi Panjang dan lebarnya merupakan setengah dari sebelumnya dan layer nya menjadi 64 dengan perkalian terhadap pembobotan (w). Diantara P2 dan P3, P3 dan P4, P4 dan P5, dan P5 dan SPPF, terlihat ada kotak yang bernama C3f, dimana kotak ini merupakan sebuah jaringan yang kompleks, yang digambarkan pada *table* detail C2f. Diperlihatkan bahwa terdapat variasi convolusi dan split yang digunakan untuk membuat jaringan *Residual Network* (ResNet). Blok C2f ini berbeda dengan blok *Conv* pada input dan outputnya, dimana pada blok C2f tidak mengalami perubahan dimensi; panjang, lebar, dan layer nya tetap. Di dalam blok C2f, ada sebuah blok *bottleneck* yang digambarkan pada secara detil pada bagian kanannya yang mempunyai warna yang sama. Ada dua macam detail blok *bottleneck*, yaitu dengan *shortcut=false* dan *shortcut=true*, yang menjelaskan fungsi jaringan tambahan fitur ResNet.

Backbone ini mengintegrasikan teknik konvolusi (*Convolutional Neural Network*) dan blok CBS (lapisan *Convolutional*, *BatchNorm*, dan aktivasi *SiLU*) guna meningkatkan kapasitas pembelajaran dan generalisasi model. Selain jaringan *backbone* yang merupakan pondasi dasar dalam jaringan YOLOv8, terdapat juga jaringan *Head* yang mana akan menghubungkan jaringan *backbone* terhadap parameter deteksi atau pemberian label. Pada proses jaringan ini, terdapat *up-sample* (*Upsample*) data dengan menggandakan dimensinya. Proses ini menggunakan *concat* sebagai penggabungan dua jaringan konvolusi dan ada dua level *up-sample* yang dilakukan. Pada masing-masing level, akan dihubungkan dengan output *Detect* sebagai output dari hasil labelisasi sebuah objek. Arsitektur ini memastikan bahwa baik fitur tingkat rendah (cocok untuk objek kecil) maupun informasi semantik tingkat tinggi (lebih baik untuk objek besar) dapat digabungkan dengan efektif, sehingga meningkatkan kemampuan model dalam mendekripsi objek dengan berbagai ukuran di lingkungan yang kompleks. Salah satu perubahan signifikan dalam YOLOv8 adalah adopsi kepala deteksi tanpa anchor (*anchor-free*), yang menyederhanakan arsitektur dengan menghilangkan kebutuhan akan kotak *anchor* yang telah ditentukan sebelumnya. Metode berbasis *anchor* tradisional sering

membutuhkan penyesuaian hyperparameter yang intensif, yang dapat mempersulit pelatihan dan penerapan. YOLOv8 menghindari hal ini dengan langsung memprediksi koordinat kotak pembatas dan probabilitas kelas untuk setiap objek yang terdeteksi. Desain tanpa *anchor* ini tidak hanya mengurangi beban komputasi tetapi juga meningkatkan kecepatan dan presisi deteksi, terutama untuk objek kecil dan padat, di mana metode berbasis *anchor* sering kali mengalami kesulitan.

Dalam jaringan saraf CNN, YOLOv8, lapisan dan hyperparameter memainkan peran penting dalam menentukan arsitektur dan kinerja model. Lapisan-lapisan tersebut, termasuk lapisan konvolusi, normalisasi *batch*, aktivasi, dan *pooling*, disusun secara berurutan untuk mengekstraksi fitur dari data input pada berbagai tingkat abstraksi. Lapisan konvolusi menerapkan filter (atau kernel) yang bergerak melintasi gambar input, menangkap pola spasial seperti tepi, tekstur, dan bentuk. Lapisan ini menghasilkan peta fitur yang kemudian disempurnakan melalui operasi lanjutan. Lapisan normalisasi *batch* menormalkan output dari lapisan konvolusi, menstabilkan proses pembelajaran, dan mempercepat pelatihan. Fungsi aktivasi, seperti ReLU atau SiLU, memperkenalkan non-linearitas ke dalam jaringan, memungkinkan model mempelajari pola yang lebih kompleks. Lapisan *pooling*, seperti *max pooling*, mengurangi dimensi peta fitur, membuat model lebih efisien secara komputasi sambil tetap mempertahankan informasi penting. Hyperparameter dalam lapisan konvolusi mengontrol perilaku model dan berdampak signifikan pada kinerjanya. Hyperparameter utama mencakup ukuran kernel, yang menentukan bidang reseptif dari filter; *stride*, yang mengontrol seberapa jauh filter bergerak melintasi input; dan *padding*, yang menentukan apakah batas input dipertahankan atau berkurang selama proses konvolusi. Selain itu, jumlah filter (atau layer) per lapisan menentukan berapa banyak fitur yang dapat dipelajari jaringan pada tahap tertentu. Semakin banyak filter memungkinkan jaringan menangkap fitur yang lebih kompleks, tetapi juga meningkatkan biaya komputasi. Hyperparameter pelatihan seperti *learning rate*, *batch size*, dan jumlah *epoch* mempengaruhi seberapa cepat model beradaptasi dan dapat melakukan generalisasi. Penyetelan hyperparameter yang tepat sangat penting untuk mencapai akurasi optimal, karena pengaturan yang terlalu agresif dapat menyebabkan *overfitting* atau gradient menghilang, sementara pengaturan yang terlalu konservatif dapat memperlambat proses pembelajaran. Oleh karena itu, kombinasi lapisan dan hyperparameter yang efektif sangat penting dalam membangun model CNN yang efisien, akurat, dan skalabel seperti YOLOv8.

Selain peningkatan arsitekturnya, YOLOv8 menawarkan modularitas dan skalabilitas yang luar biasa, membuatnya dapat disesuaikan untuk berbagai tugas visi komputer di luar deteksi objek standar. Misalnya, model ini dapat dengan mudah diperluas untuk tugas-tugas seperti segmentasi gambar, segmentasi instance, dan klasifikasi gambar. Fleksibilitas ini memastikan bahwa pengembang dapat memanfaatkan YOLOv8 sebagai solusi serbaguna untuk berbagai tugas pengenalan visual, dengan penyesuaian minimal sesuai kebutuhan aplikasi masing-masing.



Gambar 2. Ilustrasi sistem jaringan kecerdasan buatan YOLO v8 [19].

III. METODOLOGI PENELITIAN

A. Rumusan Masalah dan Aspek Keinsinyuran

Pada penelitian ini terdapat tiga rumusan masalah yang berkaitan terhadap aspek-aspek keinsinyuran. Dengan mengacu pada permasalahan tersebut, maka projek ini diharapkan dapat menjadi dasar dalam menjawab rumusan

masalah tersebut sehingga dapat dijadikan rujukan dalam pendekstrian kerusakan jalan secara modern dengan memanfaatkan keilmuan kecerdasan buatan. Ketiga rumusan masalah tersebut disusun dalam Tabel I yang didasarkan pada aspek keinsinyuran.

Tabel I
Rumusan Masalah Dan Aspek Keinsinyuran

| No. | Rumusan Masalah | Aspek Keinsinyuran |
|-----|---|--|
| 1 | Apa yang perlu untuk dikembangkan dalam ranah analisa inspeksi kerusakan jalan? | Etika dan Profesionalisme: Perencanaan jalan harus didasarkan pada kajian akademis. Kajian ini akan memberikan dasar (kode etik) dalam pengembangan deteksi kerusakan jalan secara professional dan <i>autonomous</i> . |
| 2 | Dimana lokasi yang diperlukan untuk melakukan pengumpulan data sample? | |
| 3 | Bagaimana caranya untuk menghemat waktu dan biaya dalam rangka melakukan inspeksi jalan secara menyeluruh dan akurat? | K3: Pendekstrian kerusakan jalan dapat dilakukan dengan pengembangan deteksi objek dengan memanfaatkan kecerdasan buatan. Sehingga dapat dilakukan untuk monitoring jalan dengan skala yang luas, akurat dan cepat secara <i>autonomous</i> untuk kepentingan keselamatan pengguna jalan. |

B. Batasan Masalah

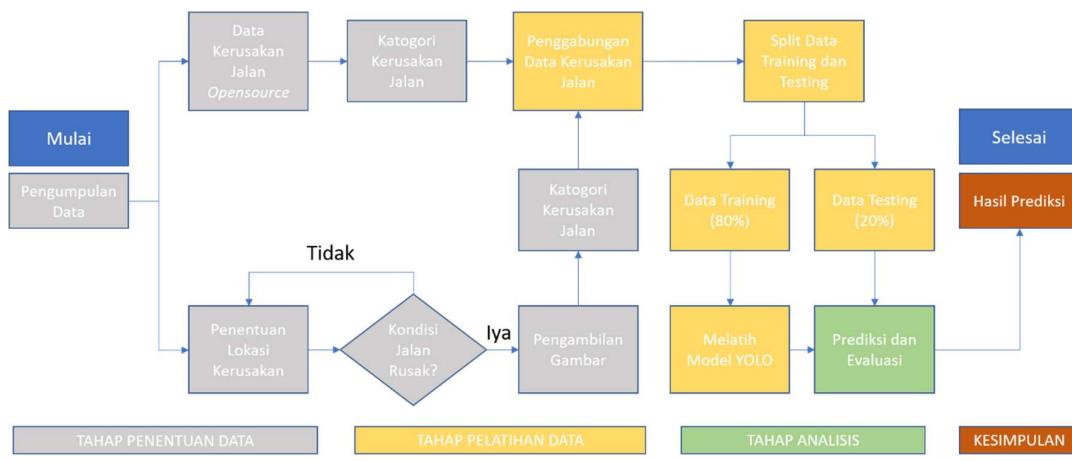
Proyek Batasan masalah pada penelitian ini, antara lain:

1. Penelitian ini dilakukan hanya untuk mendeteksi keretakan jalan pada jalan aspal yang diambil pada jalan nasional di kota Palembang dan data *opensource* yang ada di berbagai negara.
2. Penelitian ini dilakukan spesifik pada pendekatan enam tipe keretakan, yaitu *alligator*, *block*, *longitudinal*, *transverse*, *pothole*, dan *patching*.
3. Hasil dari macam-macam kerusakan tersebut akan diklasifikasikan menjadi dua kelas, yaitu rusak dan tidak rusak..

C. Diagram Alir Penelitian

Penelitian ini didasarkan pada rumusan masalah dan batasan masalah yang telah ditetapkan. Untuk menjawab penelitian ini dengan sistematis dan terstruktur, maka diperlukan diagram alir sebagai panduan alur proses penelitian. Diagram alir tersebut dibagi menjadi empat tahapan, yaitu tahapan penentuan data, pelatihan data, analisis data dan kesimpulan. Pada tahapan penentuan data mencakup tahapan pengumpulan data, yaitu pengumpulan mandiri dan pengumpulan data secara *opensource*. Pengumpulan data mandiri dilakukan di Indonesia dengan pengambilan gambar di tempat lokasi kerusakan. Proses

tahapan Pengumpulan data ini merupakan tahapan yang paling penting dalam penelitian ini. Tahapan ini mencakup proses *labeling* data yang mana akan dilakukan secara manual pada keseluruhan data. Proses *labeling* data dapat dilakukan dengan membagi menjadi dua kelas, rusak dan tidak rusak. Setelah keseluruhan data telah dilakukan labelisasi, maka selanjutnya adalah pada tahapan pelatihan data. Pada tahap pelatihan data, data *opensource* yang telah dilakukan *labeling* dijadikan sebagai data pelatihan secara keseluruhan. Kemudian pada pelatihan selanjutnya yang menggunakan data kerusakan jalan di Indonesia, dibagi menjadi dua menjadi data pelatihan dan data pengetesan atau validasi. Selanjutnya, hasil dari pelatihan model YOLOv8 tersebut dilakukan analisa berdasarkan nilai p-value dan optimisasi. Pada tahapan kesimpulan, berisi rangkuman dari keseluruhan penelitian yang mencakup tujuan, metode, hasil, dan analisis yang dilakukan. Bagian ini menekankan efektivitas penelitian dalam mencapai tujuan yang telah ditetapkan. Selain itu, kesimpulan menggarisbawahi temuan utama, signifikansi penelitian, dan implikasinya. Apabila terdapat keterbatasan atau rekomendasi untuk penelitian lanjutan, hal tersebut juga dapat diuraikan untuk memberikan wawasan dan arah bagi penelitian mendatang.



Gambar 3. Diagram alir penelitian.

D. Pengambilan Data Penelitian

Untuk mendukung penelitian ini, koleksi data diperlukan dalam rangka melatih jaringan kecerdasan buatan untuk mencari titik optimasinya. Semakin banyak data yang akan dilatih, semakin bagus pula optimasi yang akan dihasilkan, sehingga akan mempengaruhi hasil prediksinya. Ada beberapa data yang digunakan untuk melatih model YOLOv8, yaitu dengan memanfaatkan data *opensource* yang disediakan di website <https://www.kaggle.com/> dan juga data pengambilan mandiri yang dilakukan di salah satu kota di Indonesia, yaitu jalan nasional kota Palembang. Pengambilan data kerusakan di Indonesia ini juga berkolaborasi dengan Politeknik Negeri Sriwijaya (POLSRI), Palembang. Data pada *opensource* digunakan sebagai pelatihan awal YOLOv8, sedangkan pada jalan di Indonesia dilakukan pelatihan lanjutkan atau disebut dengan pembelajaran transfer (*transfer learning*). Tujuan dari dua training ini

adalah untuk mencari optimasi secara global dan menjadikan landasan optimasi global tersebut terhadap jalan yang ada di Indonesia. Sehingga, pendekatan kerusakan jalan di Indonesia tersebut merupakan optimasi global minimum. Teknik ini sangat efektif untuk digunakan dalam pembelajaran transfer. Data koleksi tersebut dijelaskan pada Table II, dimana data koleksi gambar diambil dari berbagai negara seperti Cina, Jepang, Cehnya, India, Norwegia, Amerika dan lain-lain sebagai pelatihan awal, dan data yang diambil di Indonesia sebagai pelatihan tahap selanjutnya. Jumlah keseluruhan dari data yang diambil dari publik adalah 26,304 gambar, sedangkan data yang diambil di Indonesia adalah 338 gambar.

Tabel II
Koleksi Data Kerusakan Jalan

| Koleksi Data | Jumlah (gambar) | Tahap pelatihan | Keterangan |
|------------------|-----------------|----------------------|---|
| Cina | 912 | Pelatihan tahap awal | <i>Opensource</i> |
| Cehnya | 616 | | |
| India | 6.120 | | |
| Japan | 2.947 | | |
| Norwegia | 1.023 | | |
| Amerika | 464 | | |
| Lain-lain | 14.222 | | |
| Indonesia (Pal.) | 338 | Pelatihan lanjutan | Politeknik Negeri Sriwijaya (POLSRI), Palembang |
| Total | 26.642 | --- | --- |

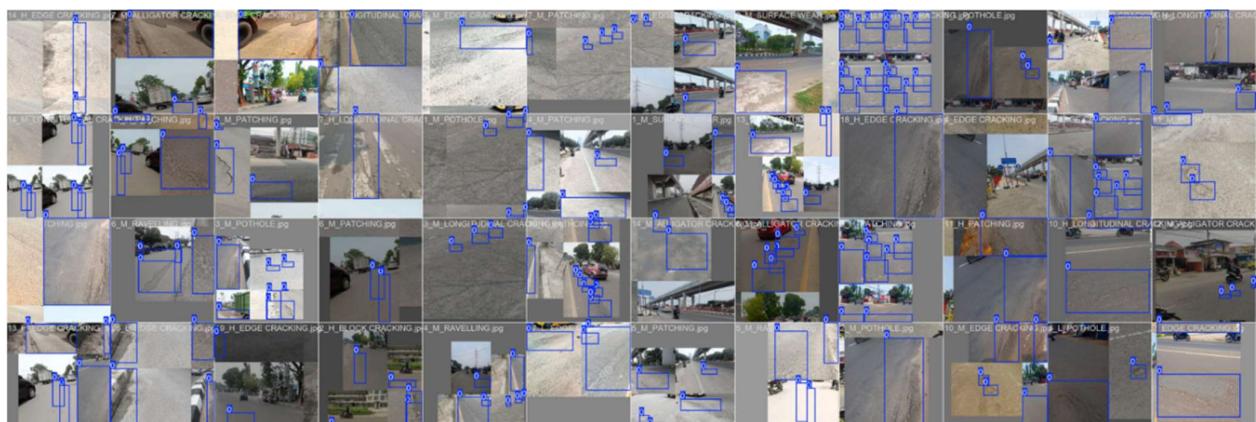
IV. HASIL DAN ANALISIS

A. Hasil Deteksi Kerusakan Jalan

Hasil dari deteksi kerusakan jalan adalah hasil dari model YOLO yang telah dilakukan pelatihan. Untuk memastikan akurasi dan keandalan dalam mendekteksi berbagai jenis kerusakan jalan, model YOLOv8 telah dilatih menggunakan dataset dengan label pada kerusakan seperti terlihat pada Gambar 4. Setiap jenis kerusakan, seperti *alligator*, *block*, *longitudinal*, *transverse*, *pothole*, dan

patching, telah diberi label dan diidentifikasi dengan kotak berwarna biru. Pendekatan ini tidak hanya meningkatkan kemampuan model dalam mengenali pola kerusakan secara presisi tetapi juga memastikan bahwa performa deteksi tetap optimal dalam berbagai kondisi lapangan. Gambar 4 menampilkan hasil *training* model YOLOv8 untuk mendekteksi berbagai jenis kerusakan permukaan jalan. Setiap objek yang terdeteksi telah diberi *bounding box* yang berwarna biru, menunjukkan kemampuan model dalam mengidentifikasi jenis kerusakan secara akurat. *Dataset* ini terdiri dari beberapa kondisi dan sudut pengambilan gambar yang bervariasi, hal ini untuk memastikan bahwa model dapat beradaptasi dengan situasi nyata di lapangan. *Labeling* dan anotasi yang konsisten dan tepat pada setiap kerusakan sangat penting untuk meningkatkan akurasi dan performa deteksi selama proses prediksi pada jalan pada saat masa mendatang.

Ketika proses pelatihan sudah selesai, baik pelatihan awal dan pelatihan lanjutan, maka proses prediksi ataupun validasi merupakan langkah untuk menganalisa hasil dari kedua pelatihan tersebut. Pelatihan yang optimal dalam arsitek YOLOv8 menghasilkan gambar dengan akurasi prediksi yang bervariasi. Hasil dari pelatihan tersebut, dapat dilihat pada Gambar 5 yang menggambarkan hasil prediksi pada kerusakan jalan yang diteksi sebagai kerusakan. Pada Gambar 5 bagian atas merupakan hasil dari pelatihan tahap awal yang menunjukkan nilai kepercayaan tersebut hingga menyentuh 0,4 ke 0,6. Ini membuktikan bahwa pada pelatihan awal pada arsitek YOLOv8 masih memerlukan optimasi untuk menentukan tingkat akurasi yang tinggi. Pada Gambar 5 bagian bawah merupakan hasil dari pelatihan tahap awal yang menunjukkan nilai kepercayaan yang lebih tinggi dan akurat, yaitu menunjukkan akurasi 0,8 hingga 0,9



Gambar 4. Pelatihan model Yolo dan proses pemberian label

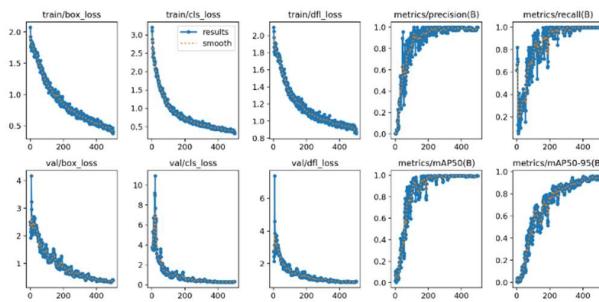


Gambar 5. Prediksi Kerusakan Jalan YOLOv8.

B. Analisis Hasil Model

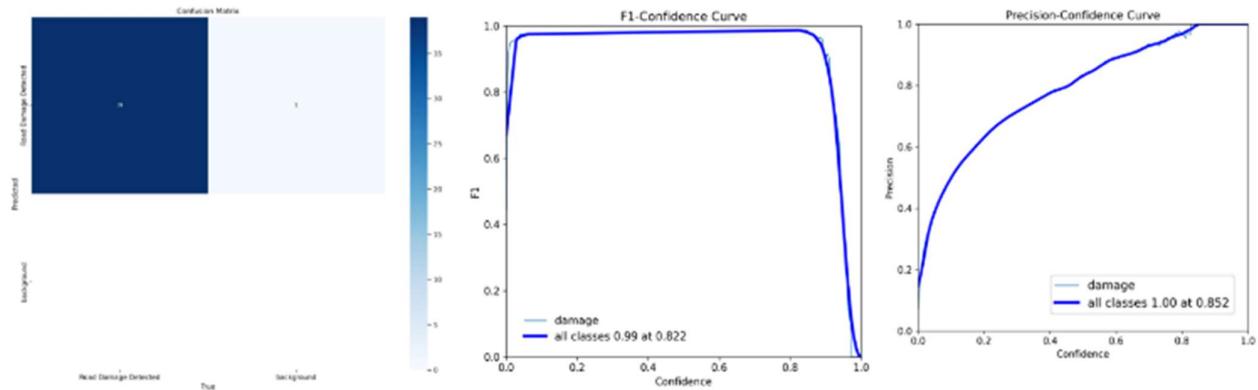
Optimasi dalam arsitek YOLOv8 adalah langkah yang penting untuk melakukan pelatihan model. Proses pelatihan ini melibatkan *updating kernel filter* pada setiap iterasi yang dilakukan selama pengerjaan. *Updating kernel filter* tersebut dilakukan secara numerik dan bertahap untuk mencari titik minimal dari sebuah optimasi yang tidak linear. Proses iterasi atau dikenal dengan *epoch* pada YOLOv8 ditetapkan hingga 450 *epoch*. Dengan melihat perkembangan dari setiap *epoch* tersebut, ada lima parameter yang dapat digunakan untuk mengetahui tingkat keakuratan yang secara bertahap mengalami kenaikan dalam hal akurasi. Dalam lima parameter tersebut dibagi menjadi dua kategori yaitu *training* (train) pada bagian atas dan *validation* (val) pada bagian bawah, pada Gambar 6. Kita bisa melihat bahwa *box loss* (*box_loss*) dan *class loss* (*cls_loss*) mengalami penurunan, dimana hal ini menunjukkan bahwa *prediction error* mengalami penurunan seiring bertambahnya *epoch*. Catatan bahwa *training data* adalah data yang digunakan untuk melakukan optimasi, sedangkan *validation data* adalah data yang tidak digunakan dalam optimasi. Sedangkan, *distribution focal loss* (*df1_loss*) adalah loss yang digunakan dalam model deteksi objek, khususnya untuk tugas regresi di dalam *bounding box*. *DFL loss* membantu model meningkatkan kemampuannya untuk secara akurat memprediksi koordinat *bounding box* yang kontinu dengan lebih memfokuskan pada sampel yang lebih sulit diprediksi. *DFL loss* dirancang untuk bekerja dengan baik pada label kontinu dengan memanfaatkan pendekatan berbasis

distribusi, yang berbeda dengan fungsi *loss* pada regresi tradisional seperti *Mean Squared Error (MSE)*. Sehingga, sama seperti *class loss* dan *box loss*, penurunan nilai pada *DFL* merupakan penurunan kesalahan atau *error* seiring pula dengan bertambahnya *epoch*. Pada teknik analisis akurasi *precision* dan *recall* adalah metrik evaluasi yang digunakan untuk mengukur kinerja model deteksi objek dalam mengidentifikasi dan mengklasifikasikan objek di dalam gambar. *Precision* mengukur seberapa akurat model dalam memprediksi objek yang terdeteksi sebagai positif (misalnya, "deteksi objek" yang sebenarnya benar). *Precision* dihitung sebagai rasio antara prediksi yang benar (*True Positives*) terhadap semua prediksi positif (*True Positives + False Positives*). *Precision* yang tinggi berarti model menghasilkan sedikit deteksi palsu (*False Positives*). Sedangkan *Recall* mengukur seberapa baik model dalam menemukan semua objek yang sebenarnya ada dalam gambar. Ini dihitung sebagai rasio antara prediksi yang benar (*True Positives*) terhadap semua kasus positif yang sebenarnya (*True Positives + False Negatives*). *Recall* yang tinggi menunjukkan bahwa model berhasil menemukan sebagian besar objek yang relevan, meskipun mungkin menghasilkan lebih banyak deteksi yang salah (*False Positives*).



Gambar 6. Pelatihan model hingga 450 epoch.

Pada Gambar 7 menampilkan tiga grafik evaluasi yang menunjukkan kinerja model dalam mendeteksi kerusakan, yaitu CM (*confusion matrix*), F1CC (*F1-confidence curve*), dan PCC (*precision-confidence curve*). CM terlihat bahwa model berhasil mendeteksi kerusakan dengan sangat baik, dengan 39 prediksi benar untuk kelas *damage* dan hanya satu kesalahan di mana latar belakang (*background*) terdeteksi



Gambar 7. Matrik konvolusi (kiri), kurva kepercayaan F1 (tengah), *p-value* (kanan)

V. KESIMPULAN DAN SARAN

Penelitian ini mengembangkan sistem deteksi otomatis berbasis model YOLOv8, yang merupakan salah satu model *deep learning* untuk mendeteksi kerusakan jalan pada permukaan aspal. Proses deteksi dan klasifikasi kerusakan jalan yang meliputi berbagai jenis, seperti retakan *alligator*, *block*, *longitudinal*, *transverse*, *pothole*, dan *patching*, telah berhasil dilakukan dengan menggunakan metode CNN (*Convolutional Neural Networks*) via arsitektur YOLOv8. Dengan sistem ini, deteksi kerusakan jalan dapat dilakukan secara *real-time*, dengan akurasi dan kecepatan yang optimal dibandingkan metode konvensional. Metode konvensional untuk inspeksi jalan yang bergantung pada tenaga ahli dalam pengecekan visual memang dapat memberikan hasil yang cukup akurat, tetapi sering kali terbatas pada cakupan area kecil dan tidak efisien dalam waktu serta biaya. Model arsitek YOLOv8 yang diterapkan dalam penelitian ini, mampu mengatasi kendala tersebut dengan mendeteksi kerusakan jalan dalam skala besar secara efisien, yang sangat bermanfaat bagi negara dengan jaringan jalan yang luas seperti Indonesia. Dengan pendekatan *transfer learning*, model YOLOv8 dilatih menggunakan data dari berbagai negara untuk menangkap pola kerusakan umum/global, kemudian disesuaikan dan dilatih menggunakan data lokal

sebagai *damage*. Hal ini menunjukkan tingkat akurasi yang tinggi dalam membedakan antara kerusakan dan latar belakang. Pada grafik F1CC, model mencapai nilai F1 sebesar 0,99 pada ambang keyakinan 0,822, yang menunjukkan keseimbangan yang baik antara *precision* dan *recall*. Nilai F1 yang tinggi ini mengindikasikan bahwa model mampu mempertahankan tingkat deteksi yang efektif dengan keseimbangan antara akurasi dan cakupan prediksi. Sementara itu, pada grafik PCC, *precision* model mencapai nilai baik (1,0) pada *confidence threshold* 0,852, menunjukkan bahwa pada ambang keyakinan ini, model tidak menghasilkan kesalahan positif palsu dalam mendeteksi kerusakan. Secara keseluruhan, ketiga grafik ini menunjukkan bahwa model bekerja dengan sangat baik dalam mendeteksi kerusakan, dengan performa tinggi pada nilai *precision*, *recall*, dan *F1 score* pada ambang keyakinan yang signifikan.

dari Palembang untuk mendapatkan pola instruksi pada kerusakan jalan di Indonesia. Pendekatan ini tidak hanya meningkatkan akurasi deteksi untuk kondisi jalan nasional tetapi juga mengurangi waktu yang dibutuhkan untuk pelatihan model. Hasil evaluasi menunjukkan bahwa model ini mampu mengenali jenis kerusakan, dengan nilai *precision* dan *recall* yang signifikan. Namun, mengingat keterbatasan data yang digunakan dalam tahap *training* pada tahap kedua, diperlukan penambahan data dengan jumlah yang seimbang seiring dengan data pada tahap *training* pertama. Penelitian ini membuka peluang untuk pengembangan lebih lanjut di masa depan, termasuk peningkatan performa model dengan dataset yang lebih beragam, klasifikasi jenis kerusakan, pengujian pada kondisi lingkungan yang berbeda, serta penerapan pada skala yang lebih luas untuk berbagai jenis kerusakan permukaan jalan.

UCAPAN TERIMA KASIH

Terimakasih kepada Dosen Pembimbing Program Profesi Insinyur Bapak Ir. Resmana, S.T., M.Eng., Ph.D. Kemudian kepada Politeknik Negeri Sriwijaya yang telah mendukung upaya kolaborasi terhadap data survey penelitian

untuk menjalankan pendidikan di Program Studi Pendidikan Profesi Insinyur Universitas Kristen Petra.

VI. DAFTAR PUSTAKA

- [1] D. Cebon, "Vehicle-Generated Road Damage - a Review," *Vehicle System Dynamics*, vol. 18, no. 1-3, pp. 107-150, 1989.
- [2] C. Koch, K. Doycheva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure (vol 29, pg 196, 2015)," *Advanced Engineering Informatics*, vol. 30, no. 2, pp. 208-210, Apr, 2016.
- [3] A. Shtayat, S. Moridpour, B. Best, A. Shroff, and D. Raol, "A review of monitoring systems of pavement condition in paved and unpaved roads," *Journal of Traffic and Transportation Engineering-English Edition*, vol. 7, no. 5, pp. 629-638, Oct, 2020.
- [4] V. M. Karbhari, J. W. Chin, D. Hunston, B. Benmokrane, T. Juska, R. Morgan, J. J. Lesko, U. Sorathia, and D. Reynaud, "Durability gap analysis for fiber-reinforced polymer composites in civil infrastructure," *Journal of Composites for Construction*, vol. 7, no. 3, pp. 238-247, Aug, 2003.
- [5] F. Outay, H. A. Mengash, and M. Adnan, "Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges," *Transportation Research Part a-Policy and Practice*, vol. 141, pp. 116-129, Nov, 2020.
- [6] S. Dorafshan, and M. Maguire, "Bridge inspection: human performance, unmanned aerial systems and automation," *Journal of Civil Structural Health Monitoring*, vol. 8, no. 3, pp. 443-476, Jul, 2018.
- [7] Z. Q. He, and L. J. Cao, "SOD-YOLO: Small Object Detection Network for UAV Aerial Images," *Ieej Transactions on Electrical and Electronic Engineering*, Oct 7, 2024.
- [8] Y. W. Jiang, "Road damage detection and classification using deep neural networks," *Discover Applied Sciences*, vol. 6, no. 8, Aug 1, 2024.
- [9] J. Y. Zeng, and H. Zhong, "YOLOv8-PD: an improved road damage detection algorithm based on YOLOv8n model," *Scientific Reports*, vol. 14, no. 1, May 27, 2024.
- [10] L. Y. Du, and Y. S. Wang, "Bi-YOLO: A novel object detection network and dataset for components of China heritage buildings," *Journal of Building Engineering*, vol. 97, Nov 15, 2024.
- [11] M. J. Choi, D. G. Ku, and S. J. Lee, "Integrated YOLO and CNN Algorithms for Evaluating Degree of Walkway Breakage," *Ksce Journal of Civil Engineering*, vol. 26, no. 8, pp. 3570-3577, Aug, 2022.
- [12] F. Elghaish, S. Talebi, E. Abdellatef, S. T. Matarneh, M. R. Hosseini, S. Wu, M. Mayouf, A. Hajirasouli, and T. Q. Nguyen, "Developing a new deep learning CNN model to detect and classify highway cracks," *Journal of Engineering Design and Technology*, vol. 20, no. 4, pp. 993-1014, Jun 29, 2022.
- [13] M. M. Naddaf-Sh, S. Hosseini, J. Zhang, N. A. Brake, and H. Zargarzadeh, "Real-Time Road Crack Mapping Using an Optimized Convolutional Neural Network," *Complexity*, vol. 2019, Sep 29, 2019.
- [14] F. Wan, C. Sun, H. Y. He, G. B. Lei, L. Xu, and T. Xiao, "YOLO-LRDD: a lightweight method for road damage detection based on improved YOLOv5s," *Eurasip Journal on Advances in Signal Processing*, vol. 2022, no. 1, Oct 18, 2022.
- [15] M. Ren, X. F. Zhang, X. Chen, B. Zhou, and Z. Y. Feng, "YOLOv5s-M: A deep learning network model for road pavement damage detection from urban street-view imagery," *International Journal of Applied Earth Observation and Geoinformation*, vol. 120, Jun, 2023.
- [16] S. V. H. Pham, and K. V. Nguyen, "Productivity Assessment of the Yolo V5 Model in Detecting Road Surface Damages," *Applied Sciences-Basel*, vol. 13, no. 22, Nov, 2023.
- [17] G. G. Guo, and Z. Y. Zhang, "Road damage detection algorithm for improved YOLOv5," *Scientific Reports*, vol. 12, no. 1, Sep 15, 2022.
- [18] R. S. Cheng, W. S. Chen, and H. Hao, "Performance and damage assessment of fault-crossing road tunnel subjected to internal boiling liquid expanding vapor explosion (BLEVE)," *Tunnelling and Underground Space Technology*, vol. 154, Dec, 2024.
- [19] Ultralytics. <https://github.com/ultralytics/ultralytics/issues/189> (accessed 2024-10-10)