

Model Deep Learning Presisi Tinggi untuk Klasifikasi Kerusakan Jalan menggunakan Transfer Learning

High Precision Deep Learning Model for Road Damage Classification using Transfer Learning

¹Muhammad Abdul Ghofur*, ²Murdifin, ³Awan Gumilang Hardandrito, ⁴Shofwatul 'Uyun

^{1,2,3,4}Magister Informatika, UIN Sunan Kalijaga Yogyakarta

^{1,2,3,4}Jl. Marsda Adisucipto, Catur Tunggal, Yogyakarta, Daerah Istimewa Yogyakarta 55281, Indonesia

*e-mail: 23206052012@student.uin-suka.ac.id

(received: 16 August 2025 2025, revised: 15 September 2025, accepted: 16 September 2025)

Abstrak

Jalan merupakan infrastruktur penting yang sering mengalami kerusakan, yang secara langsung memengaruhi keselamatan dan efisiensi transportasi. Identifikasi kerusakan jalan secara manual memerlukan waktu dan sumber daya yang besar, yang menyoroti perlunya pendekatan berbasis citra digital otomatis. Studi ini membandingkan dua arsitektur Convolutional Neural Network (CNN), MobileNetV2 dengan transfer learning dan Custom CNN, untuk mengklasifikasikan tingkat keparahan kerusakan jalan. Dataset terdiri dari 1.800 citra permukaan jalan yang didistribusikan secara merata ke dalam tiga kategori: baik, kerusakan ringan, dan kerusakan parah. Semua citra menjalani normalisasi, augmentasi, dan pengubahan ukuran, diikuti dengan evaluasi menggunakan 5-Fold Cross Validation untuk memastikan kinerja yang andal. Hasil eksperimen menunjukkan bahwa MobileNetV2 mencapai akurasi 98%, mengungguli Custom CNN dengan 89%. Temuan ini menunjukkan efektivitas transfer learning dalam meningkatkan akurasi klasifikasi dengan data terbatas dan menyoroti potensi MobileNetV2 untuk sistem deteksi kerusakan jalan yang efisien dan real-time yang terintegrasi ke dalam teknologi pemantauan infrastruktur cerdas.

Kata kunci: CNN custom, K-Fold cross validation, MobileNetV2

Abstract

Roads are critical infrastructure that frequently experience damage, directly impacting transportation safety and efficiency. Manual road damage inspection is time-consuming and resource-intensive, highlighting the need for automated, image-based approaches. This study compares two Convolutional Neural Network (CNN) architectures—MobileNetV2 with transfer learning and a custom-built CNN—for classifying road surface damage severity. The dataset consists of 1,800 road surface images evenly distributed across three categories: good, minor damage, and severe damage. All images were normalized, augmented, and resized, followed by evaluation using 5-Fold Cross-Validation to ensure robust performance. Experimental results show that MobileNetV2 achieved an accuracy of 98%, outperforming the custom CNN, which achieved 89%. These findings demonstrate the effectiveness of transfer learning in improving classification accuracy with limited data and highlight the potential of MobileNetV2 for efficient, real-time road damage detection systems that can be integrated into intelligent infrastructure monitoring solutions.

Keywords: CNN custom, K-Fold cross validation, MobileNetV2,

1 Pendahuluan

Infrastruktur jalan merupakan elemen krusial dalam mendukung mobilitas dan pertumbuhan ekonomi di Indonesia. Kerusakan jalan, seperti retakan, lubang, atau permukaan yang tidak rata, mengganggu kelancaran transportasi, memperbesar potensi kecelakaan, dan menyebabkan kerugian ekonomi di kota besar dan daerah pedesaan [1]. Menurut Kementerian Pekerjaan Umum dan Perumahan Rakyat [2], sekitar 25% dari 47.000 kilometer jalan nasional dan 53% dari 50.000 kilometer jalan provinsi mengalami kerusakan ringan hingga berat, yang mengakibatkan kerugian

<http://sistemasi.ftik.unisi.ac.id>

ekonomi mencapai sekitar Rp15 triliun (~USD 1 miliar) per tahun akibat penurunan efisiensi logistik dan peningkatan biaya perawatan kendaraan. Faktor seperti beban lalu lintas berlebih, curah hujan tinggi, dan kurangnya pemeliharaan rutin memperburuk kondisi ini [3].

Oleh karena itu, metode yang akurat dan optimal secara komputasi diperlukan untuk mendeteksi serta mengklasifikasikan kerusakan jalan guna mendukung perawatan infrastruktur. Identifikasi manual seringkali dipengaruhi subjektivitas petugas dan keterbatasan jangkauan area inspeksi, baik secara spasial maupun temporal, pendekatan ini memiliki berbagai keterbatasan. Selain memerlukan biaya dan waktu yang besar, akurasi pun sering kali dipengaruhi oleh subjektivitas petugas serta keterbatasan cakupan wilayah inspeksi. Oleh karena itu, pendekatan otomatis berbasis teknologi citra digital dan kecerdasan buatan semakin dibutuhkan untuk meningkatkan efektivitas sistem deteksi kerusakan jalan.

Seiring kemajuan kecerdasan buatan perkembangan teknologi deep learning, khususnya pada bidang image classification dengan *Convolutional Neural Networks* (CNN), membuka peluang besar untuk menyelesaikan permasalahan tersebut. CNN mampu mengenali dan membedakan pola visual dari citra kerusakan jalan dengan *accuracy* tinggi, bahkan dalam kondisi pencahayaan dan sudut pandang yang bervariasi. Beberapa arsitektur CNN yang telah terbukti efektif dalam tugas klasifikasi visual adalah CNN Custom dan *MobileNetV2*.

Model *MobileNetV2* dirancang khusus untuk perangkat dengan sumber daya terbatas, seperti sistem pemantauan bergerak, dengan tetap mempertahankan performa klasifikasi yang baik, dan resolusi input untuk mencapai keseimbangan antara *accuracy* tinggi dan efisiensi komputasi [4]. Efisiensinya menjadikannya cocok untuk penerapan dalam sistem inspeksi real-time.

Beberapa penelitian sebelumnya telah menunjukkan hasil yang menjanjikan. Misalnya, [5] membahas klasifikasi kerusakan permukaan jalan menggunakan *MobileNetV3-Small*, sebuah arsitektur CNN ringan yang dirancang untuk perangkat dengan sumber daya terbatas. Dataset berisi citra berbagai kerusakan jalan (retak, lubang, permukaan bergelombang) diproses dengan teknik augmentasi data untuk meningkatkan kemampuan generalisasi. Namun, hingga saat ini belum banyak penelitian yang secara komprehensif membandingkan.

Sebagian besar penelitian terdahulu juga hanya mengklasifikasikan kondisi jalan dalam dua kategori sederhana (rusak atau tidak rusak), tanpa mempertimbangkan tingkat kerusakan secara detail. Selain itu, banyak yang belum mengintegrasikan teknik validasi silang seperti *K-Fold Cross Validation* atau menyamakan *preprocessing* antar model, sehingga evaluasi performa model kurang komprehensif. Penelitian ini memberikan kontribusi dalam mengisi gap tersebut dengan memanfaatkan dataset berisi 1.800 gambar yang dibagi secara seimbang ke dalam tiga kelas kerusakan jalan: baik, rusak ringan, dan rusak berat. Masing-masing model akan dievaluasi menggunakan validasi silang *5-Fold*, dengan *preprocessing* yang disesuaikan dengan arsitektur masing-masing, serta augmentasi data untuk memperkuat generalisasi.

Dengan pendekatan ini, penelitian diharapkan dapat mengidentifikasi arsitektur model terbaik yang bukan hanya akurat, tetapi juga efisien secara komputasi dan dapat diandalkan dalam implementasi nyata di lapangan. Hal ini penting dalam mendukung sistem monitoring infrastruktur berbasis AI, terutama dalam konteks pengembangan smart city dan pemeliharaan jalan berkelanjutan.

2 Tinjauan Literatur

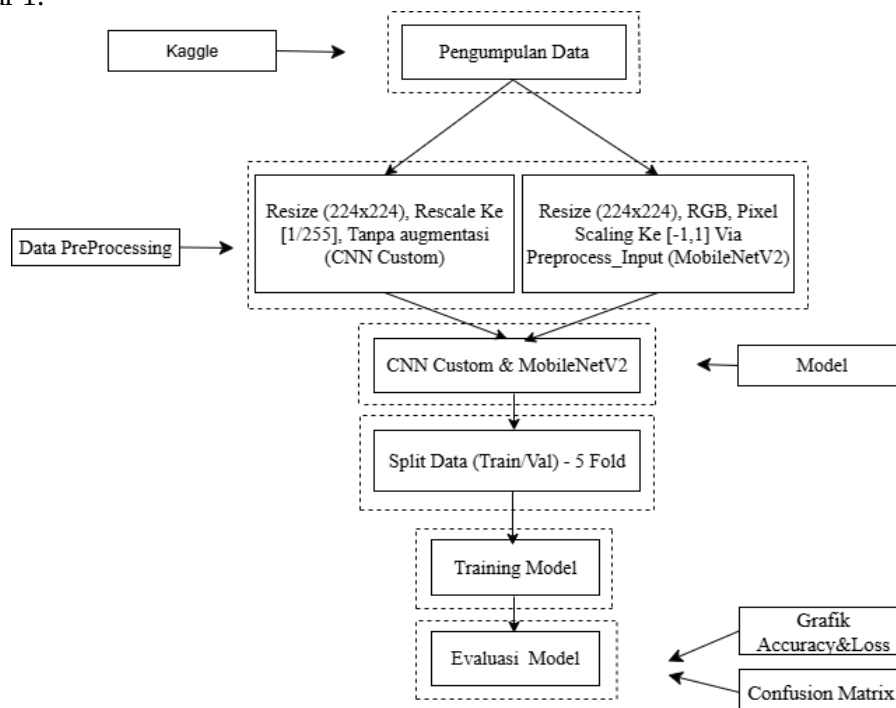
Pembelajaran transfer (*transfer learning*) merupakan pendekatan yang memanfaatkan model *Convolutional Neural Network* (CNN) yang telah dilatih sebelumnya pada dataset berskala besar sebagai ekstraktor fitur, kemudian fitur tersebut digunakan untuk melatih lapisan baru yang disesuaikan dengan permasalahan spesifik [6]. Berbagai penelitian telah mengkaji penerapan *Convolutional Neural Network* (CNN) dalam deteksi dan klasifikasi kerusakan jalan berbasis citra digital. [7] mengembangkan sistem deteksi keretakan jalan aspal menggunakan CNN dengan dataset 2.074 citra yang diperoleh dari Kaggle, melalui tahapan *preprocessing* seperti konversi citra RGB ke *grayscale*, *resize*, dan *edge detection*. Hasil pengujian menunjukkan *accuracy* sistem sebesar 92,9%, menandakan metode ini efektif dalam mengidentifikasi kerusakan jalan. [8] melakukan klasifikasi kerusakan jalan di wilayah Pontianak menggunakan CNN arsitektur LeNet-5 dengan 140 citra terbagi dalam tiga kelas (berlubang, retak, tidak rusak). Model mencapai *accuracy* terbaik 80% pada *epoch* ke-6, dengan nilai *precision* dan *recall* masing-masing 79%. [9] mengusulkan klasifikasi kualitas permukaan jalan berbasis CNN dengan *transfer learning* arsitektur Xception, menggunakan dataset

<http://sistemasi.ftik.unisi.ac.id>

empat kelas sesuai klasifikasi Kementerian PUPR. Model ini unggul dalam efisiensi waktu pelatihan sekaligus *accuracy*, dengan hasil uji mencapai 90,11% dan validasi 90%. [10] mengembangkan aplikasi Android pendeteksi keretakan jalan berbasis *hybrid* CNN-LSTM, di mana CNN digunakan untuk ekstraksi fitur visual dan LSTM untuk analisis pola sekuensial. Dataset terdiri dari 734 citra dalam tiga kelas, dan skenario pengujian memilih model dengan *accuracy* tertinggi dari proporsi data uji 10%, 20%, dan 30%. Penelitian sebelumnya menunjukkan bahwa CNN unggul dalam ekstraksi fitur hierarkis dengan akurasi tinggi, sedangkan YOLO menonjol pada deteksi objek *real-time* meskipun kurang optimal pada objek kecil atau pencahayaan ekstrem. Hasil pengujian terkini mencatat CNN membutuhkan waktu pelatihan 56,23 menit dengan akurasi 96,9%, sementara YOLO hanya 12,73 menit dengan akurasi 95,6%. Uji-T berpasangan menunjukkan perbedaan signifikan pada waktu pelatihan (YOLO lebih cepat), namun tidak pada kecepatan deteksi maupun akurasi, sehingga pemilihan algoritma dapat disesuaikan dengan kebutuhan proyek dan sumber daya komputasi [11]. Penelitian [12], yaitu menggunakan YOLOv8 pada dataset RDD2020, memperlihatkan akurasi sebesar 43,98% dengan precision 68,13% dan recall 47,67%. Hasil ini masih lebih rendah dibandingkan penelitian terdahulu, sehingga menekankan perlunya optimasi lebih lanjut melalui augmentasi data, regularisasi, atau fine-tuning model. Dari berbagai penelitian tersebut, dapat disimpulkan bahwa CNN dan transfer learning terbukti efektif dalam klasifikasi citra kerusakan jalan dengan akurasi tinggi. Namun, sebagian besar penelitian masih terbatas pada klasifikasi biner (rusak/tidak rusak) atau dua tingkat keparahan, serta jarang memanfaatkan evaluasi validasi silang. Belum banyak pula yang membandingkan langsung performa CNN *Custom* dengan model pretrained seperti *MobileNetV2* pada kasus multi-kelas. Oleh karena itu, penelitian ini hadir untuk mengisi kesenjangan tersebut dengan pendekatan multi-kelas (baik, rusak ringan, rusak berat), menggunakan validasi silang 5-Fold, serta preprocessing yang konsisten antar model.

3 Metode Penelitian

Penelitian ini menggunakan metode kuantitatif dengan pendekatan eksperimental untuk mengevaluasi kinerja beberapa arsitektur Convolutional Neural Network (CNN), yaitu *MobileNetV2* dan *CNN Custom*, dalam mengklasifikasikan tiga tingkat kerusakan jalan: baik, rusak ringan, dan rusak berat. Tahapan penelitian mencakup: (1) identifikasi masalah, (2) pengumpulan dan pra-pemrosesan data, (3) pelatihan model, (4) validasi, serta (5) evaluasi performa klasifikasi, yang terlihat pada Gambar 1.



Gambar 1 Metode penelitian

Tahap awal dimulai dengan studi literatur untuk mengkaji keterbatasan pendekatan klasifikasi citra kerusakan jalan sebelumnya, yang umumnya hanya terbatas pada dua kelas dan kurang memanfaatkan validasi silang. Hal ini mendorong perumusan strategi klasifikasi multi-kelas menggunakan pendekatan *transfer learning* pada arsitektur CNN.

1. Data Penelitian

Data yang digunakan dalam penelitian ini bersumber dari dataset yang tersedia pada platform Kaggle. Dataset ini mengandung serangkaian gambar dengan 3 kelas jenis kerusakan jalan yaitu jalan baik, jalan rusak ringan dan jalan rusak berat dengan berbagai tingkat kerusakan. Dataset ini dipilih karena kekayaan fitur visual yang dapat dimanfaatkan untuk melatih model dengan kemampuan deteksi yang baik, sebagaimana ditunjukkan pada Tabel 1.

Tabel 1 Rincian jumlah dataset

No	Label	Jumlah
1	Jalan baik	600
2	Jalan rusak ringan	600
3	Jalan rusak berat	600
Total		1800

2. Preprocessing

Terdapat dua jalur praproses sesuai tipe model:

A. Untuk CNN Custom

Pada model CNN Custom, citra masukan terlebih dahulu di-*resize* menjadi ukuran 224×224 piksel, kemudian dilakukan *rescale* terhadap intensitas piksel dengan faktor 1/255 sehingga berada pada rentang [0, 1]. Proses ini dilakukan tanpa penerapan teknik *augmentasi* sesuai dengan alur pada diagram. Pemilihan metode praproses ini didasarkan pada pertimbangan bahwa CNN buatan sendiri tidak terikat pada standar normalisasi tertentu, sehingga skala [0, 1] sudah dianggap cukup konsisten untuk mendukung proses pelatihan dasar model.

B. Untuk model pretrained MobileNetV2

Untuk arsitektur pretrained seperti MobileNetV2 citra masukan di-*resize* menjadi 224×224 piksel dan dikonversi ke format RGB. Selanjutnya, dilakukan *pixel scaling* ke rentang [-1, 1] menggunakan fungsi `preprocess_input` yang sesuai dengan masing-masing arsitektur. Langkah ini bertujuan untuk menyesuaikan skema normalisasi data dengan bobot pretrained yang sebelumnya telah dilatih pada dataset ImageNet, sehingga kompatibilitas dan efektivitas fitur yang telah dipelajari dapat terjaga. Selain itu, apabila dataset yang digunakan tidak seimbang, pada tahap ini dapat pula diterapkan teknik *class weighting* atau *resampling* untuk mengurangi bias prediksi, meskipun langkah tersebut tidak ditampilkan pada diagram.

3. Pemodelan

Penelitian ini membandingkan kinerja dua arsitektur jaringan, yaitu CNN Custom dan MobileNetV2. CNN Custom merupakan arsitektur yang dirancang sendiri dengan kombinasi lapisan konvolusi, *pooling*, dan *fully connected* sebagai model baseline. MobileNetV2 dirancang untuk efisiensi pada perangkat dengan sumber daya terbatas, menggunakan konsep *inverted residual* dan *depthwise separable convolution*. Strategi yang digunakan pada metode *transfer learning* umumnya melibatkan pemanfaatan *backbone pretrained* sebagai ekstraktor fitur, kemudian menambahkan *classification head* yang terdiri dari lapisan *GlobalAveragePooling* atau *Flatten*, dilanjutkan dengan *Dense* dan *Dropout*, serta diakhiri dengan lapisan *Dense Softmax* untuk klasifikasi tiga kelas. Pada tahap awal, sebagian lapisan awal dibekukan (*frozen*) untuk mempertahankan bobot pralatih, kemudian dilakukan *fine-tuning* pada lapisan atas setelah bagian *classification head* mencapai kestabilan (opsional).

4. Split Data 5-Fold Cross Validation

Pada tahap split data, digunakan teknik *K-Fold Cross Validation*, *K-Fold Cross Validation* merupakan salah satu metode statistik yang diimplementasikan untuk mengevaluasi performansi dari model atau algoritma yang telah dirancang. Pada tahap pelatihan dataset dibagi menjadi data latih dan data validasi. Model akan dilatih menggunakan data latih dan divalidasi menggunakan data validasi sebanyak *k-Fold* kali [13]. Penggunaan metode *5-Fold Cross Validation* bertujuan untuk memperoleh estimasi kinerja model yang lebih robust sekaligus mengurangi bias akibat pemilihan *split* data. Pada

proses ini, data terlebih dahulu di-*shuffle* kemudian dibagi menjadi lima lipatan (*Fold*) dengan ukuran yang kurang lebih sama, dan sebisa mungkin menggunakan teknik *stratified* agar proporsi kelas tetap seimbang di setiap lipatan. Selanjutnya, pada setiap iterasi ke- k ($k = 1 \dots 5$), satu *Fold* digunakan sebagai data validasi, sementara empat *Fold* lainnya digunakan untuk pelatihan. Proses pelatihan dan evaluasi dilakukan secara penuh di setiap iterasi. Setelah semua iterasi selesai, metrik evaluasi dari kelima *Fold* dirata-ratakan, lalu dihitung pula deviasi standarnya untuk memberikan gambaran mengenai stabilitas kinerja model. Dengan cara ini, setiap arsitektur model akan menghasilkan lima skor (satu per *Fold*), yang kemudian dapat digunakan untuk memilih model dengan performa terbaik atau melaporkan hasil rata-ratanya. Selain itu *k-Fold Cross Validation* adalah cara yang efektif dalam menggabungkan atribut dan setting parameter *machine learning* dalam melatih model *prediction* yang lebih baik [14].

5. Training Model

Pada proses pelatihan model, fungsi *loss* yang digunakan adalah *sparse_categorical_crossentropy* karena permasalahan klasifikasi melibatkan tiga kelas. Optimizer yang umum digunakan yaitu Adam, yang dipilih berdasarkan kebutuhan konvergensi dan stabilitas pelatihan. Parameter seperti *batch size* = 32 dan jumlah *epoch* = 30 disesuaikan dengan kapasitas GPU serta tingkat konvergensi yang diinginkan. Untuk meningkatkan efisiensi dan mencegah *overfitting*, digunakan beberapa *callback* yang direkomendasikan, antara lain Early Stopping untuk menghentikan pelatihan jika *Validation loss* tidak menunjukkan perbaikan atau mengalami kenaikan, Model Checkpoint untuk menyimpan bobot model terbaik selama proses pelatihan, serta ReduceLROnPlateau untuk menurunkan *learning rate* secara otomatis ketika proses pelatihan berada pada kondisi *plateau*.

6. Evaluasi Model

Pada tahap ini, pengujian sistem dievaluasi menggunakan *Confusion Matrix* untuk menghitung nilai *Accuracy*, *Precision*, *Recall*, dan *F-1 score* [15]. Persamaan untuk menghitung nilai *accuracy*, *precision*, *recall*, dan *F-1 score* dapat dilihat pada persamaan (1), (2), (3), dan (4).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

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

True Positive (TP): Jumlah data positif yang diidentifikasi dengan benar oleh sistem. True Negative (TN): Jumlah data negatif yang diidentifikasi dengan benar oleh sistem. False Positive (FP): Jumlah data yang seharusnya negatif tetapi diklasifikasikan sebagai positif oleh sistem. False Negative (FN): Jumlah data yang seharusnya positif tetapi diklasifikasikan sebagai negatif oleh sistem.

4 Hasil dan Pembahasan

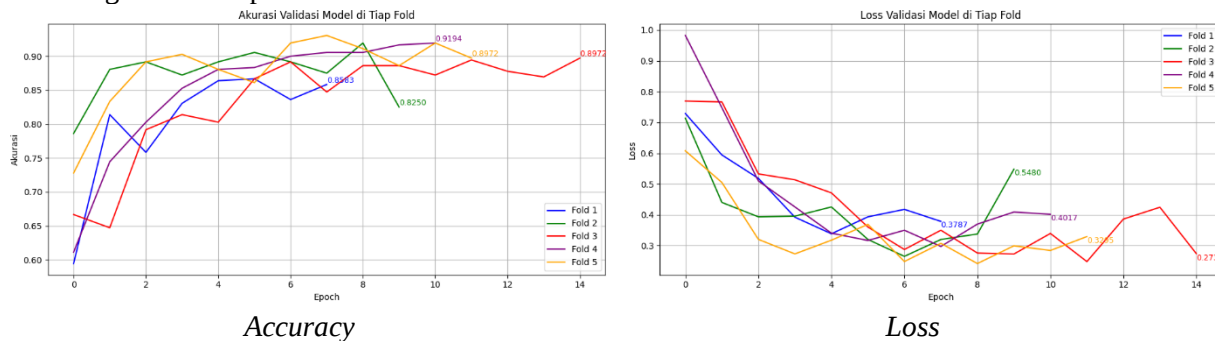
Hasil klasifikasi menunjukkan perbedaan performa yang cukup signifikan antara kedua arsitektur CNN. Model CNN Custom memperoleh nilai *accuracy*, *presisi*, *recall*, dan *f1-score* sebesar 89%. Hal ini menandakan bahwa model buatan sendiri mampu mengenali pola kerusakan jalan dengan cukup baik, meskipun masih tertinggal dibandingkan model pretrained. Keterbatasan ini wajar karena CNN Custom dilatih dari awal (*from scratch*), sehingga membutuhkan dataset yang lebih besar dan proses *fine-tuning* lebih mendalam agar mencapai hasil optimal. Meskipun demikian, konsistensi nilai pada seluruh metrik menunjukkan bahwa model ini relatif stabil.

Sebaliknya, MobileNetV2 menunjukkan performa yang jauh lebih unggul dengan seluruh metrik mencapai 98%. Capaian ini membuktikan bahwa MobileNetV2 tidak hanya unggul dalam akurasi, tetapi juga efisien, sehingga sangat sesuai untuk diimplementasikan pada perangkat dengan keterbatasan sumber daya, seperti sistem berbasis mobile. Secara keseluruhan, hasil ini menegaskan bahwa model pretrained, khususnya MobileNetV2, sangat efektif untuk klasifikasi kerusakan jalan dan berpotensi diadopsi dalam implementasi nyata di lapangan, yang terlihat pada Tabel 2.

Tabel 2 Hasil penelitian pada model

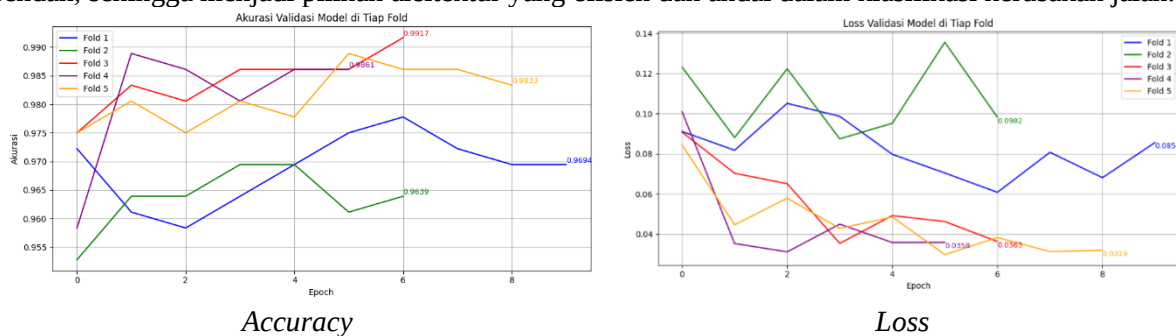
No	Arsitektur	Accuracy	Precision	Recall	F-1 Score
1.	CNN Custom	89%	89%	89%	89%
2.	MobileNetV2	98%	98%	98%	98%

Gambar 2 memperlihatkan grafik hasil validasi model CNN Custom, yang mampu belajar dengan baik meskipun performanya bervariasi antar *Fold*. Akurasi awal masih rendah di kisaran 0,60–0,75, namun meningkat seiring epoch dan mencapai nilai tertinggi 0,9194 pada *Fold* 5, sementara *Fold* lainnya berkisar antara 0,8250–0,8972 dengan rata-rata sekitar 0,89. Pola penurunan loss juga konsisten, dari kisaran 0,70–0,95 di awal hingga mencapai titik terendah 0,2736 pada *Fold* 5, sedangkan *Fold* lain berada di rentang 0,37–0,54. Hasil ini menunjukkan bahwa CNN Custom cukup efektif dalam mengklasifikasikan kerusakan jalan, meski stabilitas antar *Fold* masih kurang konsisten dibandingkan model pretrained.



Gambar 2 Grafik CNN custom

Gambar 3 menampilkan Grafik validasi model MobileNetV2 yang menunjukkan performa sangat baik dan stabil di setiap *Fold*. Akurasi validasi konsisten tinggi, berada pada rentang 0,9694 hingga 0,9917, dengan *Fold* 3 mencatat nilai tertinggi sebesar 0,9917, sedangkan *Fold* 1 dan *Fold* 2 relatif lebih rendah namun tetap kompetitif (0,9694 dan 0,9639). Sementara itu, grafik loss validasi memperlihatkan tren penurunan yang signifikan, dengan nilai akhir terendah dicapai pada *Fold* 5 (0,0319) dan *Fold* 4 (0,0358), sedangkan *Fold* 1 dan *Fold* 2 menunjukkan loss yang sedikit lebih tinggi (0,0856 dan 0,0982). Hasil ini mengindikasikan bahwa MobileNetV2 mampu mengklasifikasikan data dengan akurasi tinggi sekaligus menjaga tingkat kesalahan prediksi yang rendah, sehingga menjadi pilihan arsitektur yang efisien dan andal dalam klasifikasi kerusakan jalan.



Gambar 3 Grafik mobileNetV2

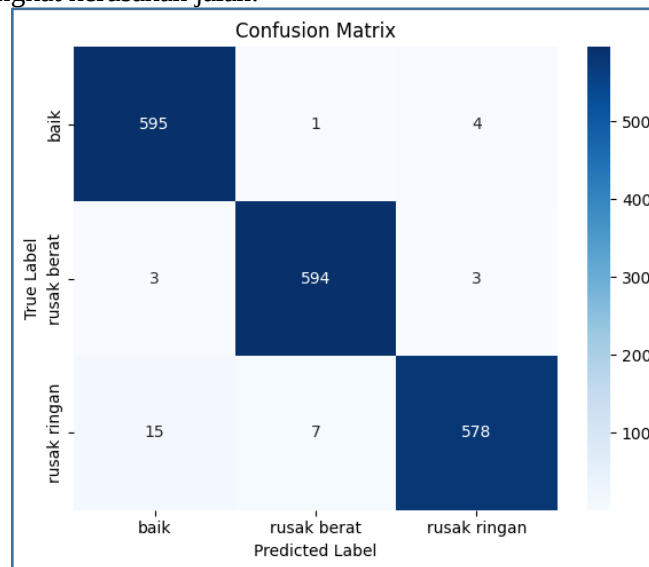
Hasil pengujian menunjukkan bahwa CNN Custom meraih akurasi rata-rata 87,94% dengan rentang cukup lebar (82,50–91,94%), sedangkan MobileNetV2 jauh lebih unggul dengan rata-rata 97,89% dan rentang sempit (96,39–99,17%). MobileNetV2 konsisten mengungguli CNN Custom di semua *Fold* dengan selisih +6,67 hingga +13,89 poin, membuktikan keunggulannya dalam akurasi sekaligus stabilitas performa, yang ditampilkan pada Tabel 3.

Tabel 3 Hasil accuracy validasi tiap fold

Model	Fold 1 (Biru)	Fold 2 (Hijau)	Fold 3 (Merah)	Fold 4 (Ungu)	Fold 5 (Oranye)
CNN Custom	0.8583	0.8250	0.8972	0.9194	0.8972
MobileNetV2	0.9694	0.9639	0.9917	0.9861	0.9833

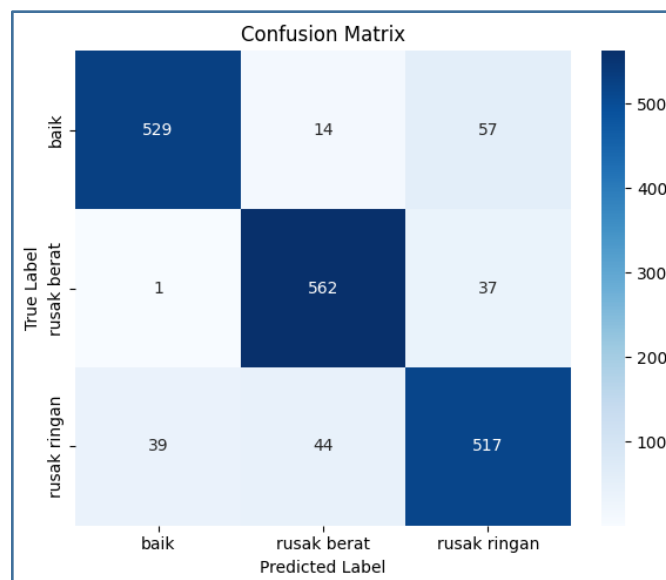
Berdasarkan hasil *Confusion Matrix* dari kedua arsitektur model, terlihat bahwa model CNN *Custom* masih mengalami cukup banyak kesalahan klasifikasi antar kelas, terutama antara kelas “baik” dan “rusak ringan”. Sementara itu, model *MobileNetV2* menunjukkan peningkatan *accuracy* dengan kesalahan prediksi yang jauh lebih sedikit, menjadikannya model paling andal untuk klasifikasi tingkat kerusakan jalan dari gambar.

Gambar 4 menunjukkan *Confusion Matrix* Model *MobileNetV2* dengan *accuracy* 98,3% dengan *precision*, *recall*, dan *F1-score* di atas 97% pada semua kelas. Mayoritas citra terklasifikasi dengan benar, meskipun terdapat sedikit kesalahan prediksi terutama pada kelas rusak ringan yang terkadang terdeteksi sebagai baik atau rusak berat. Hasil ini menandakan model sangat efektif dalam membedakan ketiga tingkat kerusakan jalan.



Gambar 4 *Confusion matrix mobileNetV2*

Gambar 5 menunjukkan *Confusion Matrix* model CNN *Custom* dengan model masih mengalami kesalahan prediksi yang relatif lebih banyak dibanding model sebelumnya. Kelas baik sering salah diklasifikasikan sebagai rusak ringan (57 kasus), sementara kelas rusak ringan cukup sering diprediksi sebagai baik atau rusak berat (39 dan 44 kasus). Meski sebagian besar prediksi tepat, distribusi kesalahan yang merata di beberapa kelas menunjukkan model memerlukan optimasi lebih lanjut, terutama dalam membedakan kondisi jalan baik dan rusak ringan. Berdasarkan *Confusion Matrix* tersebut, nilai evaluasi model adalah sebagai berikut: *accuracy* 89,3%, *precision* 89,4%, *recall* 89,3%, dan *F1-score* 89,3%.



Gambar 5 *Confusion matrix CNN custom*

Berdasarkan tabel perbandingan, terlihat bahwa setiap model memiliki tingkat akurasi yang berbeda dalam klasifikasi kerusakan jalan. Model Xception [9] yang menggunakan dataset privat mampu mencapai akurasi sebesar 90%, sedangkan LeNet-5 [8] pada dataset privat hanya memperoleh akurasi 80%, menunjukkan keterbatasan arsitektur klasik dalam menangani data visual yang lebih kompleks.

Pendekatan Hybrid CNN-LSTM [10] dengan dataset dari platform Kaggle menunjukkan performa yang jauh lebih tinggi, yakni akurasi 97,5% untuk model hybrid dan 92,76% untuk CNN saja. Hasil ini menegaskan bahwa integrasi CNN dan LSTM dapat meningkatkan kemampuan model dalam memahami fitur spasial sekaligus pola sekuensial.

Sementara itu, penelitian ini mengusulkan penggunaan model berbasis pretrained. *MobileNetV2* yang dilatih dengan dataset Kaggle berhasil mencapai akurasi 98%, menjadikannya model yang ringan sekaligus efektif untuk aplikasi pada perangkat dengan keterbatasan komputasi. Di sisi lain, CNN Custom hanya mencapai akurasi 89%, relatif lebih rendah dibandingkan model pretrained. Hal ini wajar karena CNN yang dibangun dari awal membutuhkan dataset yang lebih besar dan proses fine-tuning lebih dalam agar dapat menyaingi performa model pretrained.

Secara keseluruhan, hasil ini menunjukkan bahwa model pretrained seperti *MobileNetV2* unggul signifikan dibandingkan arsitektur lama (LeNet-5, Xception) maupun CNN yang dibangun sendiri, terutama dalam konteks klasifikasi kerusakan jalan multi-kelas pada dataset yang lebih bervariasi. Berikut adalah perbandingan penelitian sekarang dengan penelitian sebelumnya terlihat pada Tabel 4.

Tabel 4 Perbandingan dengan penelitian sebelumnya

Teknik/Model	Kumpulan Data	Accuracy
Xception [9]	Privat	90%
LeNet-5 [8]	Privat	80%
Hybrid CNN-LSTM [10]	Platform Kaggle.	Hybrid CNN-LSTM 97.5% dan CNN 92.76%.
MobileNetV2 (Usulan kami)	Kaggle.	98%
CNN Custom (Usulan kami)	Kaggle.	89%

5 Kesimpulan

Penelitian ini mengevaluasi kinerja CNN Custom dan *MobileNetV2* dalam klasifikasi kerusakan jalan. Hasil menunjukkan bahwa *MobileNetV2* mencapai akurasi tertinggi (98%) dengan efisiensi tinggi, sedangkan CNN Custom hanya sekitar 89%. Temuan ini menegaskan bahwa *transfer learning* dengan model pretrained mampu meningkatkan performa secara signifikan, terutama pada data terbatas. Penerapan augmentasi data dan validasi silang 5-Fold juga efektif dalam meningkatkan generalisasi dan mengurangi *overfitting*. Secara keseluruhan, *MobileNetV2* terbukti akurat, efisien, dan layak diimplementasikan dalam sistem pemantauan jalan berbasis smart city. Model ini dapat dimanfaatkan oleh pemerintah daerah melalui integrasi data dari CCTV, drone, atau sensor IoT yang diproses secara real-time dan ditampilkan dalam dashboard monitoring untuk mendukung pengambilan keputusan prioritas perbaikan jalan secara cepat, efisien, dan berbasis data. Penelitian selanjutnya disarankan mengeksplorasi arsitektur lebih mutakhir seperti EfficientNetV2 atau Vision Transformer (ViT) serta memanfaatkan data yang lebih bervariasi agar generalisasi model semakin optimal.

Referensi

- [1] R. Rustam, M., Susanto, A., & Hidayat, "Deteksi Kerusakan Jalan Menggunakan Deep Learning dan Citra UAV," *J. Teknol. dan Sist. Komput.*, Vol. 11(1), pp. 33–41, 2023.
- [2] K. P. U. dan P. Rakyat, "Statistik Infrastruktur Jalan 2021," Jakarta: Direktorat Jenderal Bina Marga.
- [3] N. Fitri, "Analisis Kerusakan Jalan Akibat Curah Hujan dan Beban Lalu Lintas di Indonesia," *J. Transp. dan Infrastruktur*, Vol. 8(2), pp. 77–85, 2022.
- [4] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th Int. Conf. Mach. Learn. ICML 2019*, Vol. 2019-June, pp. 10691–10700, 2019.
- [5] U. Brawijaya, M. A. Maulana, B. Darma Setiawan, and R. S. Perdana, "Fakultas Ilmu Komputer Klasifikasi Kerusakan Permukaan Jalan Menggunakan Model MobileNetV3-

- Small,” Vol. X, No. X, No. X, pp. 2548–964, 2024, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [6] A. Abubakar, M. Ajuji, and I. U. Yahya, “Comparison of deep transfer learning techniques in human skin burns discrimination,” *Appl. Syst. Innov.*, Vol. 3, No. 2, pp. 1–15, 2020, DOI: 10.3390/asi3020020.
- [7] Y. Yulianto and A. Wibowo, “Neural Network,” Vol. 4, No. 2, pp. 581–594, 2023.
- [8] A. W. Mulia, I. Ruslianto, D. M. Midyanti, J. Rekayasa, and S. Komputer, “Pontianak Dan Sekitarnya Dengan Menggunakan,” Vol. 11, No. 01, pp. 11–20, 2023.
- [9] R. L. Gaho, I. T. Ali, and E. Prakasa, “Klasifikasi Kualitas Permukaan Jalan Raya Menggunakan Metode CNN Berbasis Arsitektur Xception,” pp. 354–365, 2024.
- [10] I. A. Pradana, A. D. Rahajoe, and A. N. Sihananto, “Jalan Berbasis Android Dengan Implementasi Algoritma Hybrid CNN-LSTM,” Vol. 5, No. 2, pp. 1–10, 2024.
- [11] N. Khairunisa, . C., and A. Jamaludin, “Analisis Perbandingan Algoritma Cnn Dan Yolo Dalam Mengidentifikasi Kerusakan Jalan,” *J. Inform. dan Tek. Elektro Terap.*, Vol. 12, No. 3, 2024, DOI: 10.23960/jitet.v12i3.4434.
- [12] San Gabriel Vanness Kenrick Erwi and Hafiz Irsyad, “Implementasi Deteksi Objek Pada Jalan Rusak Menggunakan Metode YOLOv8,” *Bul. Ilm. Inform. Teknol.*, Vol. 3, No. 1, pp. 1–9, 2024, DOI: 10.58369/biit.v3i1.76.
- [13] Y. N. Fuadah, I. D. Ubaidullah, N. Ibrahim, F. F. Taliningsing, N. K. Sy, And M. A. Pramuditho, “Optimasi Convolutional Neural Network dan K-Fold Cross Validation pada Sistem Klasifikasi Glaukoma,” *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, Vol. 10, No. 3, p. 728, 2022, DOI: 10.26760/elkomika.v10i3.728.
- [14] G. Litjens *et al.*, “A survey on deep learning in medical image analysis,” *Med. Image Anal.*, Vol. 42, No. December 2012, pp. 60–88, 2017, DOI: 10.1016/j.media.2017.07.005.
- [15] U. Multi, D. Palembang, J. Rajawali, and N. Palembang, “Klasifikasi Kanker Kulit Pada Citra Dermatoskopi Menggunakan CNN 1,2,” Vol. 5, No. 1, 2024, DOI: 10.35957/algorithmexxxx.