



Received 00th January 20xx
Accepted 00th February 20xx
Published 00th March 20xx

Open Access

DOI: 10.35472/x0xx0000

Analisis Perbandingan Kinerja Metode VGG16 dan MobileNetV2 Pada Klasifikasi Kepadatan Lalu Lintas

Aisyah Tiara Pratiwi¹, Lulu Christin Sihombing²,
Rizka Yustiana Zahra³, Muhamma Fahrul Aditya⁴,
Meinisa⁵, Erwan Arief⁶

aisyah.121450074@student.itera.ac.id¹

lulu.121450152@student.itera.ac.id²

rizka.121450058@student.itera.ac.id³

muhammad.121450156@student.itera.ac.id⁴

meinisa.121450076@student.itera.ac.id⁵

erwan.121450062@student.itera.ac.id⁶

Abstract : *Traffic congestion is a significant challenge that affects travel efficiency, economic costs, and environmental quality. Addressing this issue requires accurate identification and classification of traffic density, which can assist authorities in optimizing traffic flow and designing effective solutions. This study compares the performance of VGG16 and MobileNetV2 in classifying traffic density images into five classes: Empty, Traffic Jam, Low, High, and Medium. The class weights method was applied to address data imbalance. Results show that VGG16 achieved a test accuracy of 84%, while MobileNetV2 achieved 81%. Both models struggled to distinguish between the Medium and Low classes. MobileNetV2 excels in efficiency for real-time applications, whereas VGG16 provides higher accuracy for precision-oriented tasks.*

Keywords: VGG16, MobileNetV2, Traffic Density, Class Weights, Deep Learning

Abstrak: Kemacetan lalu lintas merupakan tantangan signifikan yang mempengaruhi efisiensi perjalanan, biaya ekonomi, dan kualitas lingkungan. Penanganan masalah ini memerlukan identifikasi dan klasifikasi kepadatan lalu lintas yang akurat, yang dapat membantu otoritas dalam mengoptimalkan aliran lalu lintas serta merancang solusi yang efektif. Penelitian ini membandingkan performa VGG16 dan MobileNetV2 dalam mengklasifikasikan gambar kepadatan lalu lintas menjadi lima kelas: Empty, Traffic Jam, Low, High, dan Medium. Metode class weights digunakan untuk mengatasi ketidakseimbangan data. Hasil menunjukkan VGG16 mencapai akurasi uji 84%, sementara MobileNetV2 mencapai 81%. Kedua model kesulitan membedakan kelas Medium dan Low. MobileNetV2 unggul dalam efisiensi untuk aplikasi real-time, sedangkan VGG16 lebih akurat untuk tugas presisi tinggi.

Kata kunci: VGG16, MobileNetV2, Kepadatan Lalu Lintas, Class Weights, Deep Learning



Pendahuluan

Lalu lintas merupakan elemen penting dalam kehidupan sehari-hari masyarakat modern. Sistem transportasi yang baik mendukung aktivitas seperti bekerja, belajar, hingga distribusi barang dan jasa [1]. Namun, seiring dengan pertumbuhan populasi dan urbanisasi, masalah lalu lintas seperti kemacetan menjadi tantangan utama di banyak kota besar. Masalah ini dapat mengganggu efisiensi waktu, meningkatkan biaya transportasi, dan memengaruhi kualitas hidup masyarakat [2]. Untuk mengatasi masalah kemacetan, identifikasi dan klasifikasi kepadatan lalu lintas menjadi sangat penting. Informasi ini membantu otoritas lalu lintas dalam merancang solusi yang tepat, seperti pengaturan waktu lampu lalu lintas, perencanaan rute alternatif, hingga pemberian informasi kepada pengguna jalan [3]. Dengan sistem yang mampu mengklasifikasikan kondisi lalu lintas seperti lancar, padat, atau macet, pengelolaan transportasi dapat dilakukan lebih efektif.

Identifikasi dan klasifikasi kepadatan lalu lintas memberikan banyak manfaat, seperti ,menyediakan informasi tentang kondisi jalan membantu pengguna memilih rute yang lebih efisien, membantu dalam manajemen lalu lintas yang lebih baik dengan didukungnya sistem pengelolaan lalu lintas berbasis data memungkinkan respons yang lebih cepat terhadap perubahan kondisi jalan, dan dengan mengurangi kemacetan emisi gas rumah kaca dari kendaraan juga dapat diminimalkan [4]. Dengan seiring berkembangnya ilmu pengetahuan dan teknologi, telah diciptakan berbagai macam metode yang dapat digunakan untuk pengolahan data citra. Dalam penelitian ini menggunakan dua metode deep learning populer, yaitu VGG16 dan MobileNetV2, untuk klasifikasi kepadatan lalu lintas.

VGG16 adalah arsitektur CNN yang andal untuk menghasilkan representasi fitur visual yang mendalam, meskipun memerlukan sumber daya komputasi yang cukup besar. VGG16 memiliki jaringan yang lebih mendalam dari AlexNet dan bisa lebih akurat dalam merepresentasikan karakteristik data gambar yang akan diidentifikasi dan klasifikasi [5]. MobileNetV2 dirancang untuk efisiensi komputasi, sehingga cocok digunakan dalam perangkat dengan sumber daya terbatas [6]. MobileNetV2 menerapkan teknik *Depthwise Separable Convolutions* (DSC) untuk portabilitas dan tidak hanya memperbaiki masalah penghancuran informasi dalam lapisan non-linier dalam blok konvolusi dengan menggunakan *Linear Bottlenecks*, tetapi juga memperkenalkan struktur baru bernama *Inverted residuals* untuk mempertahankan informasi [7].

Tujuan penelitian ini adalah untuk membandingkan hasil evaluasi kedua metode tersebut berdasarkan akurasi, efisiensi, dan kemampuan generalisasi dalam mengklasifikasikan tingkat kepadatan lalu lintas. Hasil dari penelitian ini diharapkan dapat memberikan wawasan tentang model yang lebih optimal untuk diterapkan dalam sistem manajemen lalu lintas berbasis kecerdasan buatan.

Metode

Deskripsi Dataset

Penelitian ini menggunakan dataset yang berasal dari website kaggle.com dengan total gambar 4038 kepadatan lalu lintas di Singapura. Dataset ini memiliki 5 kelas yaitu, *Empty*, *High*, *Low*, *Medium*, dan *Traffic Jam*. Dari masing-masing kelas tersebut akan dibagi menjadi data pelatihan, data uji, dan data validasi. Pembagian dataset untuk setiap kelas yaitu 80% untuk data pelatihan (training set), 10% untuk data validasi (*validation set*), dan 10% untuk data pengujian (*testing set*). Pembagian ini bertujuan untuk memastikan model dapat dilatih secara optimal, divalidasi untuk menghindari *overfitting*, dan diuji untuk mengevaluasi performa pada data yang belum pernah dilihat sebelumnya.

Preprocessing Data

1. Rescale Nilai Piksel

Semua gambar diskalakan ke rentang nilai [0,1] menggunakan *ImageDataGenerator* dengan argumen *rescale=1.0/255*. Hal ini memastikan piksel gambar sesuai untuk digunakan dalam model deep learning.

2. Augmentasi Data

Augmentasi data meningkatkan keragaman data pelatihan agar model menjadi lebih mampu mengenali pola yang lebih beragam. Pada penelitian ini diterapkan tahap augmentasi sebagai berikut.

Tabel 1. Penerapan Augmentasi Data

Method	Setting
Rotation	15°
Image height range	0.1
Image width range	0.1
Zoom	10%
Flipping horizontal	True
Shearing	0.1

Augmentasi diterapkan hanya pada data pelatihan, sedangkan data validasi dan pengujian hanya diskalakan.

3. Target Size Penyesuaian

Semua gambar diubah ukurannya menjadi (224,224) piksel agar sesuai dengan arsitektur model VGG16 dan MobileNetV2.

4. Pemisahan Dataset

Dataset dibagi menjadi tiga bagian, yaitu sebagai berikut.

Tabel 2. Pembagian Dataset

Data	Rasio
Data pelatihan	80%
Data validasi	10%
Data pengujian	10%

Pembagian ini dilakukan dengan mempertahankan struktur folder untuk memastikan setiap kelas diwakili secara proporsional.

5. Penanganan Ketidakseimbangan Kelas

Bobot kelas dihitung berdasarkan distribusi data pelatihan untuk mengatasi ketidakseimbangan antara kelas-kelas seperti *Empty*, *High*, *Low*, *Medium*, dan *Traffic Jam*. Bobot ini kemudian digunakan selama pelatihan model.

6. Visualisasi Data

Dilakukan visualisasi distribusi kelas dan gambar dari setiap kelas untuk memastikan distribusi data yang merata serta keberagaman dalam setiap kategori.

VGG16

VGG16 adalah arsitektur *Convolutional Neural Network* (CNN) yang diperkenalkan oleh Simonyan dan Zisserman dari Universitas Oxford dalam makalah berjudul "*Very Deep Convolutional Networks for Large-Scale Image Recognition*" pada tahun 2014 [8]. Arsitektur ini dirancang untuk tugas pengenalan dan klasifikasi gambar. VGG16 memiliki total 16 lapisan yang dapat dilatih (13 lapisan konvolusi dan 3 lapisan fully connected) [9]. VGG16 menerima input gambar dengan ukuran tetap, yaitu 224×224×3224 (tinggi, lebar, saluran warna). VGG16 memiliki performa yang luar biasa dalam klasifikasi gambar [10]. Namun, VGG16 memiliki parameter besar sehingga membutuhkan daya komputasi yang tinggi sehingga tidak efisien jika dibandingkan dengan arsitektur yang lebih modern [11].

MobileNetV2

MobileNetV2 adalah arsitektur *Convolutional Neural Network* (CNN) yang dikembangkan oleh Google pada tahun 2018, sebagai peningkatan dari versi sebelumnya, *MobileNetV1* [12]. *MobileNetV2* dirancang untuk memberikan efisiensi komputasi yang tinggi, terutama untuk aplikasi mobile dan perangkat dengan sumber daya

terbatas seperti CPU dan GPU dengan kekuatan rendah [13]. Model ini banyak digunakan dalam perangkat edge dan aplikasi *real-time*, seperti pengenalan gambar pada ponsel. *MobileNet* menawarkan keseimbangan antara kecepatan, efisiensi, dan akurat, membuatnya ideal untuk aplikasi di perangkat dengan keterbatasan sumber daya seperti ponsel, *drone*, dan perangkat IoT yang menjadikannya pilihan yang menarik untuk analisis klasifikasi citra [14].

Evaluasi Model

1. Nilai Loss dan Akurasi dari Data Pelatihan dan Validasi

Dengan melihat grafik nilai loss dan akurasi dari data pelatihan dan akurasi dapat memberikan wawasan tentang bagaimana model belajar selama pelatihan, dan membantu dalam diagnosis masalah untuk memperbaiki kinerja model [15]. Selain itu, dapat mendeteksi *overfitting* dan *underfitting*. *Overfitting* terjadi ketika model bekerja sangat baik pada data *training* tetapi buruk pada data *validation* [16]. Pada grafik, ini ditunjukkan dengan akurasi *training* yang tinggi tetapi akurasi *validation* yang stagnan atau menurun [17]. Sedangkan *underfitting* terjadi ketika model tidak mampu menangkap pola dalam data, yang terlihat dari akurasi *training* dan *validation* yang sama-sama rendah [18].

2. Confusion Matrix

Confusion Matrix adalah tabel yang digunakan untuk mengevaluasi kinerja model klasifikasi, terutama dalam kasus klasifikasi biner atau multikelas [19]. Tabel ini memberikan gambaran rinci tentang bagaimana prediksi model sesuai (atau tidak sesuai) dengan label sebenarnya, dengan membagi hasil prediksi ke dalam beberapa kategori. Dengan mengidentifikasi kesalahan dapat membantu memahami jenis kesalahan apa yang lebih sering terjadi (FP atau FN). Berikut adalah elemen yang ada di dalam confusion matrix.

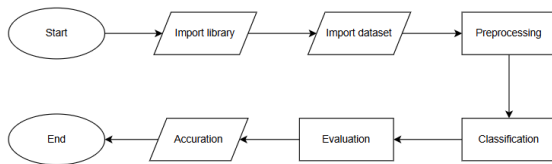
Tabel 3. Elemen Confusion Matrix

Actual/Predicted	Positive	Negative
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

- True Positive (TP): Kasus di mana model memprediksi positif dan sebenarnya memang positif.

- True Negative (TN): Kasus di mana model memprediksi negatif dan sebenarnya memang negatif.
- False Positive (FP): Kasus di mana model memprediksi positif tetapi sebenarnya negatif.
- False Negative (FN): Kasus di mana model memprediksi negatif tetapi sebenarnya positif.

Flowchart



Gambar 1. Flowchart

Hasil dan Pembahasan

Pada penelitian ini, model VGG16 dan MobileNetV2 dibandingkan berdasarkan performa pada dataset kepadatan lalu lintas yang terdiri dari lima kelas : *Empty*, *Traffic Jam*, *Low*, *High* dan *Medium*. Dataset memiliki ketidakseimbangan jumlah data pada setiap kelas, yang diatasi dengan menerapkan *class weights* selama pelatihan. Berikut merupakan hasil evaluasi model VGG16 dan MobileNetV2 :

Hyperparameter

Tabel 4. Hyperparameter

Hyperparameter	Keterangan
Epoch (VGG16)	10
Epoch (MobileNetV2)	20
Optimizer	Adam
Batch size	32
Fungsi aktivasi FS layer	ReLU
Fungsi aktivasi layer output	Softmax
Fungsi loss	Categorical cross-entropy

Model VGG16

1. Akurasi dan Loss Selama Pelatihan

- Pada awal pelatihan, akurasi VGG16 sebesar 34,5% dengan loss sebesar 3,04. Setelah 10 epoch, akurasi

pelatihan meningkat menjadi 62,3%, dengan loss menurun ke 0,78.

- Akurasi validasi menunjukkan hasil yang stabil, mencapai 78,8%, dengan loss validasi terakhir tercatat sebesar 0,57.

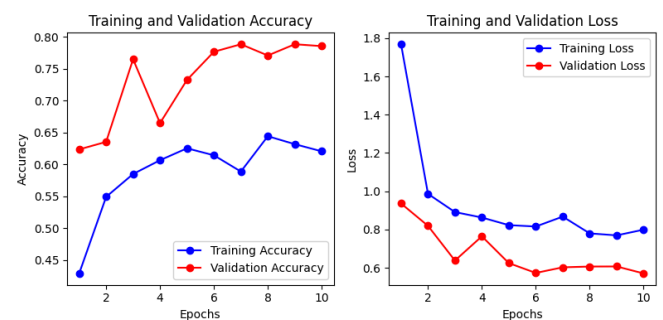
2. Evaluasi pada data Uji

- VGG16 mencapai akurasi keseluruhan 84% pada data uji.
- Pada data uji, VGG16 berhasil mencapai akurasi keseluruhan sebesar 84%, dengan precision terbaik pada kelas *Traffic Jam* (97%) dan recall tertinggi juga pada kelas yang sama (95%).

3. Analisis Kesalahan Klasifikasi

- Kesalahan terbesar terjadi pada kelas *Medium* (22 kesalahan) dan *Low* (20 kesalahan).
- Berdasarkan *heatmap confusion matrix*, kesalahan ini terjadi karena ketidaktepatan dalam membedakan kelas dengan karakteristik visual yang mirip.

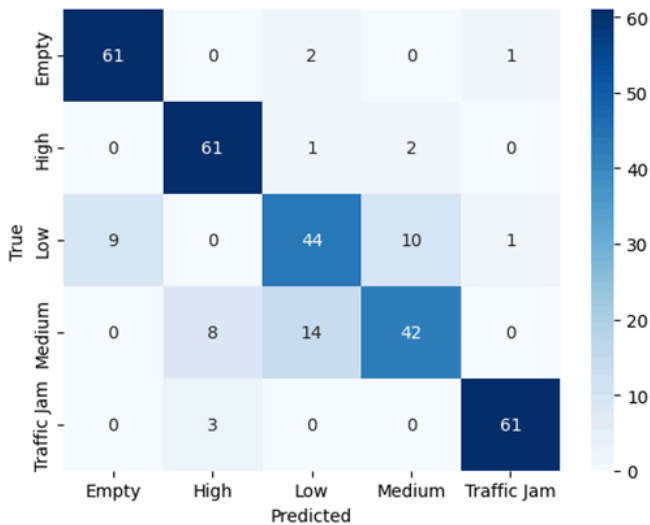
Pada kinerja model VGG16 hasil visualisasi grafik akurasi pada gambar 2 menunjukkan kinerja model VGG16 pada data pelatihan dan validasi selama 10 epoch. Pada grafik akurasi (kiri), akurasi validasi meningkat signifikan di awal dan mencapai stabilitas di sekitar 78-80%, sementara akurasi pelatihan cenderung meningkat secara bertahap namun lebih rendah dibanding validasi, sekitar 63-65%. Pada grafik loss (kanan), loss validasi menurun tajam dan stabil di sekitar 0.5-0.6, menunjukkan model dapat menggeneralisasi dengan cukup baik. Grafik dapat dilihat pada gambar (2) berikut :



Gambar 2. Grafik Akurasi VGG16

Hasil visualisasi *confusion matrix* pada model VGG16 memiliki performa yang baik dalam mengklasifikasikan data uji, terutama pada kelas *Empty*, *High*, dan *Traffic Jam*, yang memiliki prediksi benar tinggi (masing-masing 61). Namun, kelemahan terlihat pada kesalahan klasifikasi antara kelas *Low* dan *Medium*, dengan 14 prediksi salah dari *Low* ke *Medium* dan 10 sebaliknya, menunjukkan adanya kemiripan fitur yang sulit dibedakan oleh model. Meskipun secara

keseluruhan model cukup andal, peningkatan pada pembelajaran fitur untuk kelas yang saling mirip dapat dilakukan untuk meningkatkan akurasi lebih lanjut. Hasil visualisasi *confusion matrix* dapat dilihat pada gambar (3) berikut :



Gambar 3. Confusion Matrix (VGG16)

Selanjutnya, dilakukan prediksi menggunakan model VGG16, dari total 20 gambar yang ditampilkan, 16 prediksi berhasil diklasifikasikan dengan benar, sementara 4 prediksi masih keliru. Model menunjukkan performa yang baik pada kelas Empty, High, Medium, dan Traffic Jam, dimana sebagian besar gambar memiliki prediksi yang sesuai dengan label sebenarnya. Namun, kesalahan prediksi terjadi pada beberapa kelas seperti Low yang terkadang diprediksi sebagai Empty atau Medium, yang mengindikasikan tantangan model dalam membedakan kepadatan lalu lintas dengan sedikit perbedaan visual. Tingkat akurasi yang cukup tinggi menunjukkan bahwa VGG16 dapat diandalkan untuk tugas klasifikasi kepadatan lalu lintas. Hasil prediksi dapat dilihat pada gambar (4) berikut :



Gambar 4. Prediksi Model VGG16

MobileNetV2

1. Akurasi dan Loss Selama Pelatihan

- Pada fase awal pelatihan, akurasi MobileNetV2 mencapai 69,3%, dengan loss sebesar 1,84. Setelah fine-tuning, akurasi pelatihan meningkat hingga 75,5%, sedangkan loss menurun ke 0,68.
- Akurasi validasi mencapai 78,5%, dengan loss validasi akhir sebesar 0,66.

2. Evaluasi pada data Uji

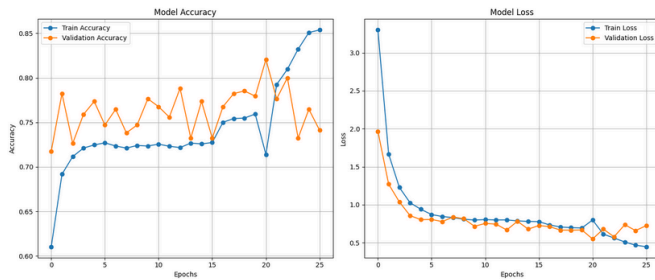
- MobileNetV2 mencapai akurasi keseluruhan 81% pada data uji.
- Pada data uji, MobileNetV2 berhasil mencapai akurasi keseluruhan sebesar 81%, precision tertinggi pada kelas *Traffic Jam* (98%) dan recall tertinggi juga pada kelas yang sama (94%).

3. Analisis Kesalahan Klasifikasi

- Kesalahan terbesar terjadi pada kelas *Low* (22 kesalahan) dan *Medium* (17 kesalahan).
- Heatmap confusion matrix menunjukkan pola kesalahan serupa dengan VGG16, terutama pada kelas yang sulit dibedakan.

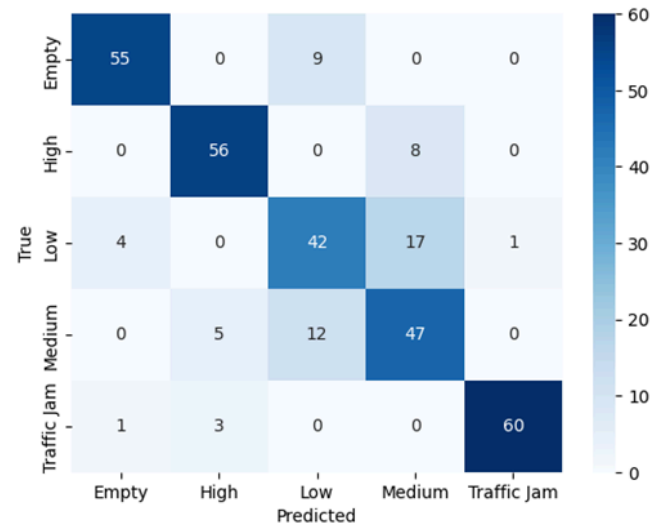
Pada kinerja model MobileNetV2 hasil visualisasi grafik akurasi pada gambar 4 menunjukkan bahwa selama proses pelatihan awal dan *fine-tuning*, model mengalami peningkatan performa secara bertahap. Pada grafik akurasi,

terlihat bahwa akurasi pelatihan terus meningkat hingga mendekati 85%, sedangkan akurasi validasi menunjukkan fluktuasi tetapi tetap mengikuti tren peningkatan, mencapai sekitar 80% pada akhir pelatihan. Pada grafik *loss*, baik *loss* pelatihan maupun validasi mengalami penurunan yang signifikan pada awal pelatihan dan stabil setelah beberapa *epoch*, dengan nilai akhir *loss* validasi sedikit lebih tinggi dibandingkan *loss* pelatihan, yang mengindikasikan potensi *overfitting* ringan. Secara keseluruhan, model menunjukkan performa yang baik, tetapi fluktuasi akurasi validasi dapat menunjukkan bahwa ada ruang untuk perbaikan dalam generalisasi model. Grafik dapat dilihat pada gambar (5) berikut :



Gambar 5. Grafik Akurasi Model MobileNetV2

Hasil visualisasi *confusion matrix* pada model MobileNetV2 menunjukkan bahwa model memiliki akurasi yang baik pada kelas *Empty*, *High*, dan *Traffic Jam*, dengan jumlah prediksi benar masing-masing 55, 56, dan 60. Namun, terdapat kelemahan dalam membedakan kelas *Low* dan *Medium*, yang terlihat dari 17 data *Low* salah diklasifikasikan sebagai *Medium* dan 12 data *Medium* salah diklasifikasikan sebagai *Low*. Selain itu, terdapat beberapa kesalahan minor seperti 9 data *Empty* diprediksi sebagai *Low* dan 8 data *High* diprediksi sebagai *Medium*. Secara keseluruhan, model cukup baik untuk kelas yang memiliki pola lebih jelas, namun perlu peningkatan dalam mengenali fitur yang membedakan kelas *Low* dan *Medium* untuk mengurangi tingkat kesalahan klasifikasi antar kedua kelas tersebut. Hasil visualisasi *confusion matrix* dapat dilihat pada gambar (6) berikut :



Gambar 6. Confusion Matrix (MobileNetV2)

Selanjutnya, dilakukan prediksi menggunakan model MobileNetV2, dari total 20 gambar, 15 prediksi diklasifikasikan dengan benar, sementara 5 prediksi masih mengalami kesalahan. Model menunjukkan performa yang baik pada kelas *Empty* dan *Traffic Jam*, di mana seluruh gambar dalam kedua kategori tersebut diprediksi dengan benar. Namun, beberapa kesalahan terjadi pada kelas *High* dan *Low*, di mana model memprediksi kelas *Medium*. Hal ini menunjukkan bahwa model mengalami kesulitan dalam membedakan kepadatan lalu lintas yang memiliki perbedaan visual tipis antar kelas, seperti antara *Low* dan *Empty* atau *High* dan *Medium*. Hasil prediksi dapat dilihat pada gambar (7) berikut :



Gambar 7. Prediksi Model MobileNetV2

Kesimpulan

Penelitian ini berhasil membandingkan performa VGG16 dan MobileNetV2 pada dataset kepadatan lalu lintas yang tidak seimbang. Hasilnya menunjukkan bahwa VGG16 unggul dalam akurasi, dengan nilai tertinggi pada data uji sebesar 84%. Namun, MobileNetV2 menawarkan efisiensi yang lebih baik dalam hal waktu pelatihan dan penggunaan memori, meskipun akurasinya lebih rendah, yaitu 81%. Kedua model menghadapi kesulitan dalam membedakan kelas *Low* dan *Medium*, yang memiliki pola visual serupa. Confusion matrix menunjukkan bahwa sebagian besar kesalahan klasifikasi terjadi pada kedua kelas ini. Penelitian lanjutan dapat mengeksplorasi penggunaan data augmentation tambahan dan arsitektur model lain untuk meningkatkan performa pada kelas-kelas yang sulit dibedakan.

References

- [1] N. K. Nur, P. R. Rangan, and Mahyuddin, *Sistem Transportasi*, vol. 1, no. 69. 2021.
- [2] Hadi Kusumah, "Klasifikasi Tingkat Kepadatan Lalu Lintas Oleh Pengguna Kendaraan Bermotor Dengan Menggunakan Random Forest," 2020, [Online]. Available: <https://library.likmi.ac.id/show/87/pdf>
- [3] A. F. Hidayah et al., "MENGUNAKAN ALGORITMA COUNTER DALAM," vol. 11, no. 2, 2024.
- [4] M. G. Boarnet, E. J. Kim, and E. Parkany, "Measuring traffic congestion," *Transp. Res. Rec.*, no. 1634, pp. 93–99, 1998, doi: 10.3141/1634-12.
- [5] H. Wang, "Garbage recognition and classification system based on convolutional neural network vgg16," *Proc. - 2020 3rd Int. Conf. Adv. Electron. Mater. Comput. Softw. Eng. AEMCSE 2020*, pp. 252–255, 2020, doi: 10.1109/AEMCSE50948.2020.00061.
- [6] S. Arnandito and T. B. Sasongko, "Comparison of EfficientNetB7 and MobileNetV2 in Herbal Plant Species Classification Using Convolutional Neural Networks," *J. Appl. Informatics Comput.*, vol. 8, no. 1, pp. 176–185, 2024, doi: 10.30871/jaic.v8i1.7927.
- [7] K. Dong, C. Zhou, Y. Ruan, and Y. Li, "MobileNetV2 Model for Image Classification," *Proc. - 2020 2nd Int. Conf. Inf. Technol. Comput. Appl. ITCA 2020*, pp. 476–480, 2020, doi: 10.1109/ITCA52113.2020.00106.
- [8] D. Hindarto and H. Santoso, "Plat Nomor Kendaraan dengan Convolution Neural Network," *J. Inov. Inform.*, vol. 6, no. 2, pp. 1–12, 2021, doi: 10.51170/jii.v6i2.202.
- [9] S. Tammina, "Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images," *Int. J. Sci. Res. Publ.*, vol. 9, no. 10, p. p9420, 2019, doi: 10.29322/ijrsrp.9.10.2019.p9420.
- [10] Weny Indah Kusumawati and Adisaputra Zidha Noorizki, "Perbandingan Performa Algoritma VGG16 Dan VGG19 Melalui Metode CNN Untuk Klasifikasi Varietas Beras," *J. Comput. Electron. Telecommun.*, vol. 4, no. 2, 2023, doi: 10.52435/complete.v4i2.387.
- [11] M. Khatama Insani and D. Budi Santoso, "Perbandingan Kinerja Model Pre-Trained CNN (VGG16, RESNET, dan INCEPTIONV3) untuk Aplikasi Pengenalan Wajah pada Sistem Absensi Karyawan," *J. Indones. Manaj. Inform. dan Komun.*, vol. 5, no. 3, pp. 2612–2622, 2024, [Online]. Available: <https://journal.stmiki.ac.id>
- [12] N. M. K. K. Handayani, E. Y. Hidayat, M. Naufal, and P. L. W. E. Putra, "Pengenalan Ekspresi Wajah Menggunakan Transfer Learning MobileNetV2 dan EfficientNet-B0 dalam Memprediksi Perkelahian," *J. Media Inform. Budidarma*, vol. 8, no. 1, p. 106, 2024, doi: 10.30865/mib.v8i1.7048.
- [13] P. Studi, M. Teknik, U. Pamulang, and T. Selatan, "yang Ditingkatkan Menggunakan Model Fine Tuning pada," vol. 2, pp. 1–24, 2024.
- [14] W. Bismi, D. Novianti, and M. Qomaruddin, "Analisis Perbandingan Klasifikasi Citra Genus Panthera dengan Pendekatan Deep learning Model MobileNet," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 6, no. 1, pp. 1–9, 2024.
- [15] K. H. Oktaviano and A. H. Nasution, "Pencarian Visual Berbasis Jaringan Convolutional Neural Network untuk Platform Pemasaran Digital Produk UMKM," vol. 6, no. 2, pp. 348–358, 2024, doi: 10.33650/jeecom.v4i2.
- [16] C. N. Syahputri and M. S. Hasibuan, "OPTIMASI KLASIFIKASI DECISION TREE DENGAN TEKNIK PRUNING UNTUK MENGURANGI OVERFITTING," vol. 11, no. 2, pp. 87–96, 2024, doi: 10.30656/jsii.v11i2.9161.
- [17] G. D. Nursyafitri, "Model Overfitting & Underfitting di Machine Learning," DQLab. [Online]. Available:

- <https://dqlab.id/model-overfitting-and-underfitting-di-machine-learning>
- [18] W. Nugraha, M. Syarif, U. Bina, S. Informatika, D. Tree, and R. Forest, "Evaluasi Performa Algoritma Klasifikasi dalam Prediksi Gagal Jantung : Studi Kasus Dataset Heart Failure Prediction," vol. 23, no. 4, pp. 897–908, 2024.
- [19] S. A. Utiahman and A. M. M. Pratama, "Analisis Perbandingan KNN, SVM, Decision Tree dan Regresi Logistik Untuk Klasifikasi Obesitas Multi Kelas," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 6, pp. 3137–3146, 2024, doi: 10.30865/klik.v4i6.1871.
- [20] K. Abdi, A. Warjaya, I. Muthmainnah, and P. H. Pahutar, "Penerapan Algoritma Random Forest dalam Prediksi Kelayakan Air Minum," *J. Ilmu Komput. dan Inform.*, vol. 3, no. 2, pp. 81–88, 2024, doi: 10.54082/jiki.81.