

Segmentasi Citra Rambu Lalu Lintas Menggunakan Arsitektur DeepLabV3+ Berbasis Convolutional Neural Network

Intan Aulia^{1*}, Mutaqin Akbar²

^{1,2}Program Studi Informatika, Fakultas Teknologi Informasi, Universitas Mercu Buana Yogyakarta, Indonesia

Email: ¹*221110108@student.mercubuana-yogya.ac.id, ²mutaqin@mercubuana-yogya.ac.id

Email Penulis Korespondensi: ¹221110108@student.mercubuana-yogya.ac.id

Abstrak—Keamanan di jalan raya sangat ditentukan oleh kemampuan pengemudi dalam memahami serta mengikuti petunjuk yang ada. Namun, pengenalan rambu secara manual rentan terhadap kelalaian manusia, sehingga diperlukan sistem yang dapat secara otomatis mendeteksi dan memisahkan rambu lalu lintas secara akurat. Tujuan dari studi ini adalah menciptakan model segmentasi citra rambu lalu lintas dengan memanfaatkan arsitektur DeepLabV3+ berbasis Convolution Neural Network (CNN) dengan backbone MobileNetV2. Kumpulan data yang dipakai terdiri atas 2.050 foto rambu lalu lintas yang mencakup 10 kategori rambu, yang diperoleh dari penelitian terdahulu. Karena dataset awal belum memiliki anotasi mask, dilakukan proses pelabelan manual menggunakan perangkat lunak LabelMe untuk menghasilkan mask segmentasi. Citra tersebut kemudian diubah ukurannya menjadi 128×128 piksel. Model DeepLabV3+ dikonfigurasi dengan optimizer Adam (learning rate 0,0002), ditraining selama 40 epoch dengan batch size 4 pada platform Google Colab menggunakan GPU T4. Evaluasi terhadap data pengujian mengindikasikan bahwa model ini memperoleh tingkat akurasi sebesar 93,59%, *Intersection over Union* (IoU) sebesar 79,12%, serta koefisien Dice sebesar 79,96%. Hasil ini membuktikan bahwa arsitektur DeepLabV3+ dengan backbone MobileNetV2 mampu melakukan segmentasi citra rambu lalu lintas secara efektif pada dataset 10 kelas.

Kata Kunci: Segmentasi Citra, Rambu Lalu Lintas, DeepLabV3+, Convolution Neural Network, MobileNetV2.

Abstract—Road safety is largely determined by drivers' ability to understand and follow traffic signs. However, manual recognition of traffic signs is prone to human error, necessitating a sistem capable of automatically detecting and segmenting traffic signs accurately. This study aims to develop a traffic sign image segmentation model utilizing the DeepLabV3+ architecture based on a Convolutional Neural Network (CNN) with MobileNetV2 as the backbone. The dataset consists of 2,050 traffic sign images covering 10 sign categories, obtained from previous research. Since the original dataset lacked mask annotations, manual labeling was performed using LabelMe software to generate segmentation masks. The images were resized to 128×128 pixels. The DeepLabV3+ model was configured with the Adam optimizer (learning rate 0.0002) and trained for 40 epochs with a batch size of 4 on Google Colab using a T4 GPU. Evaluation on the test data indicates that the model achieved an accuracy of 93,59%, an *Intersection over Union* (IoU) score of 79,12%, and a Dice coefficient of 79,96%. These results demonstrate that the DeepLabV3+ architecture with MobileNetV2 backbone is capable of effectively segmenting traffic sign images across a 10-class traffic sign dataset.

Keywords: Image Segmentation, Traffic Signs, DeepLabV3+, Convolutional Neural Network, MobileNetV2.

1. PENDAHULUAN

Pengembangan teknologi di sector transportasi memerlukan sistem yang mampu meningkatkan keselamatan berkendara melalui pemanfaatan kecerdasan buatan. Salah satu elemen krusial dalam sistem pengatur lalu lintas adalah rambu lalu lintas, yang berperan sebagai media informasi, peringatan, dan pengatur bagi pengguna jalan [1]. Namun, dalam praktiknya masih sering terjadi pelanggaran atau ketidaksadaran pengguna jalan terhadap rambu lalu lintas yang berpotensi menyebabkan kecelakaan [2].

Seiring perkembangan teknologi pengolahan citra digital, berbagai penelitian telah dikembangkan untuk mengidentifikasi rambu lalu lintas secara otomatis menggunakan pendekatan computer vision, yang dikenal sebagai Traffic Sign Recognition (TSR). Teknologi ini banyak dimanfaatkan dalam Advanced Driver Assistance System (ADAS) maupun kendaraan otonom [3], [4] membuktikan bahwa penerapan Convolutional Neural Network (CNN) pada dataset rambu lalu lintas Indonesia mencapai akurasi pengujian sebesar 99,33%.

Proses pengenalan rambu lalu lintas tidak hanya bergantung pada kategori, tetapi juga memerlukan segmentasi citra agar rambu terpisah dengan jelas dari lingkungan sekitarnya. Segmentasi citra merupakan proses fundamental yang mengklasifikasi piksel dalam suatu citra ke dalam beberapa kelompok region berdasarkan atribut visual tertentu [5]. Metode konvensional seperti thresholding, K-Means clustering, dan watershed memiliki keterbatasan signifikan dalam menghadapi variasi pencahayaan dan kompleksitas latar belakang [6], [7].

Pendekatan berbasis deep learning, khususnya arsitektur DeepLabV3+, hadir sebagai solusi efektif. Arsitektur ini mengintegrasikan modul encoder-decoder dengan Atrous Spatial Pyramid Pooling (ASPP) yang mampu menangkap informasi multi-skala [8]. [9] membuktikan bahwa DeepLabV3+ menghasilkan *Dice Similarity Coefficient* 96,91% pada segmentasi citra medis, sementara [10] mencapai F1-score 0,922 dan mIoU 0,733 pada segmentasi citra satelit. [11] turut mengonfirmasi bahwa arsitektur berbasis CNN mampu memberikan performa kompetitif dalam klasifikasi visual dengan efisiensi komputasi tinggi.

Berdasarkan kajian tersebut, penelitian ini mengusulkan penerapan arsitektur DeepLabV3+ berbasis CNN dengan backbone *MobileNetV2* untuk dataset yang terdiri dari 10 jenis rambu lalu lintas sesuai regulasi Indonesia. Model ini diharapkan dapat menghasilkan segmentasi yang akurat sebagai fondasi sistem pengenalan rambu lalu lintas otomatis, serta dapat dijadikan referensi bagi penelitian selanjutnya dalam pengembangan metode segmentasi berbasis deep learning [12]. Hasil model yang dihasilkan berpotensi diintegrasikan ke dalam sistem ADAS untuk mendukung keselamatan berkendara pada transportasi cerdas di Indonesia [13].

2. METODOLOGI PENELITIAN

Penelitian ini dilakukan melalui beberapa tahap, yaitu studi literatur, pengumpulan dan praproses data, perancangan serta implementasi model DeepLabV3+, hingga evaluasi model, sebagaimana ditunjukkan pada Gambar 1.

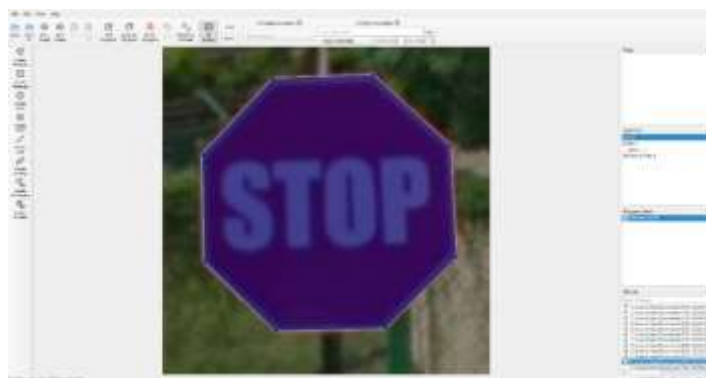


Gambar 1. Alur Penelitian

Gambar 1 menunjukkan alur penelitian yang dimulai dari tahap studi literatur untuk memahami konsep segmentasi citra dan metode DeepLabV3+. Selanjutnya dilakukan pengumpulan data dan praproses data sebelum masuk ke tahap perancangan arsitektur serta implementasi model DeepLabV3+. Tahap terakhir adalah evaluasi kinerja model untuk mengetahui hasil segmentasi yang diperoleh.

2.1 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini berdasar dari penelitian terdahulu oleh [14], terdiri dari 2.050 citra berwarna (RGB) rambu lalu lintas yang mewakili 10 kelas tanda lalu lintas Indonesia. Dimensi citra diseragamkan menjadi 128×128 piksel. Dataset dibagi menjadi data pelatihan (*Training*) sebanyak 1.750 citra (85%) dan data pengujian (*Testing*) sebanyak 300 citra (15%). Karena dataset awal tidak memiliki anotasi mask, maka pelabelan manual dilakukan menggunakan perangkat lunak LabelMe untuk menghasilkan ground truth mask segmentasi pada tingkat piksel.



Gambar 2. Proses Anotasi Polygon Rambu Lalu Lintas Menggunakan LabelMe

Gambar 2 menunjukkan antarmuka perangkat lunak LabelMe yang digunakan untuk melakukan anotasi polygon pada citra rambu lalu lintas. Pada proses ini, setiap citra dianotasi secara manual dengan menggambar polygon yang mengikuti bentuk tepi objek rambu. Label kelas diberikan pada setiap polygon sesuai dengan kategori rambu yang bersangkutan, misalnya “Berhenti (STOP)” untuk rambu stop. Hasil anotasi kemudian disimpan dalam format JSON dan dikonversi menjadi citra mask PNG yang digunakan sebagai ground truth segmentasi pada proses pelatihan model.

2.2 Praproses Data

Tahap praproses meliputi pelabelan manual menggunakan LabelMe, di mana setiap citra dianotasi menggunakan fitur polygon annotation untuk menghasilkan area segmentasi sesuai bentuk objek rambu. Hasil anotasi dikonversi ke format citra PNG sebagai ground truth mask segmentasi biner. Seluruh citra kemudian dinormalisasi nilai piksel ke rentang 0–1 untuk menstabilkan proses pelatihan. Tabel 1 menampilkan 10 jenis rambu lalu lintas yang digunakan sebagai objek segmentasi dalam penelitian ini

Tabel 1. Kelas Rambu Lalu Lintas yang Digunakan

No	Kelas Rambu	Label ID
1.	Dilarang berhenti	1
2.	Rambu Stop	2
3.	Wajib Berhenti	3
4.	Kecepatan Max 40	4
5.	Lampu Lalu Lintas	5
6.	Persimpangan 4	6
7.	Zebra Cross	7
8.	Penyebrangan Anak	8
9.	Bahaya Umum	9
10.	Dilarang Parkir	10

2.3. Perancangan Arsitektur DeepLabV3+

Arsitektur yang diterapkan adalah DeepLabV3+ dengan backbone MobileNetV2 sebagai ekstraktor fitur utama. Citra digital merupakan representasi visual dalam matriks piksel dua dimensi yang dapat diproses komputer untuk analisis visual [15], umumnya dalam format RGB di mana setiap piksel memiliki tiga nilai intensitas yang memungkinkan model deep learning mengekstraksi fitur visual secara hierarkis [16]. Komponen encoder menggunakan atrous convolution yang memungkinkan perluasan receptive field tanpa mengurangi resolusi fitur, serta modul ASPP dengan beberapa dilation rate paralel untuk menangkap informasi multi-skala [17]. Decoder menggabungkan fitur encoder dengan informasi resolusi tinggi dari lapisan awal untuk menghasilkan peta segmentasi yang lebih detail. Deep learning memanfaatkan jaringan saraf berlapis untuk mengolah informasi dari fitur sederhana hingga abstrak [18], dengan kemampuan ekstraksi fitur otomatis yang melampaui metode konvensional [19]. [20] membuktikan bahwa metode berbasis CNN modern seperti YOLO V5 mampu mendeteksi beragam jenis rambu dengan akurasi tinggi, sedangkan [21] mengonfirmasi CNN memberikan hasil lebih unggul dibandingkan metode machine learning konvensional pada analisis citra [14]. [22] juga menunjukkan bahwa arsitektur CNN untuk segmentasi semantik secara konsisten menghasilkan akurasi lebih tinggi dibandingkan metode konvensional. Boly dan Akbar (2024) turut membuktikan bahwa arsitektur encoder-decoder seperti U-Net dan SegNet memiliki performa baik namun masih terbatas dalam menangani variasi skala objek yang tinggi [23].

Model dikonfigurasi dengan optimizer Adaptive Moment Estimation (Adam) dengan learning rate 0,0002, dilatih selama 40 epoch dengan batch size 4. Fungsi loss yang digunakan merupakan kombinasi antara Binary Cross-Entropy dan Dice Loss dan Dice Loss untuk mengatasi ketidakseimbangan distribusi kelas. Mekanisme ModelCheckpoint digunakan untuk menyimpan bobot model terbaik berdasarkan nilai validation loss terendah.

2.4. Implementasi Model

Implementasi dilakukan menggunakan bahasa pemrograman Python dengan framework TensorFlow dan Keras, dijalankan pada platform Google Colab dengan GPU T4. Dataset dimuat menggunakan mekanisme generator untuk mengoptimalkan penggunaan memori. Library utama yang digunakan meliputi TensorFlow, Keras, OpenCV, NumPy, dan Albumentations untuk augmentasi data. Kode implementasi menggunakan MobileNetV2 pre-trained sebagai backbone yang diintegrasikan ke dalam arsitektur DeepLabV3+ secara end-to-end.

2.5. Metrik Evaluasi

Evaluasi kinerja model menggunakan tiga metrik utama. Pertama, *Accuracy* yang mengukur proporsi piksel yang diklasifikasikan dengan benar. Kedua, *Dice Coefficient* yang mengukur tingkat kesamaan antara hasil segmentasi dengan ground truth, diformulasikan pada persamaan (1). Ketiga, *Intersection over Union* (IoU) yang mengukur tumpang tindih antara area prediksi dan ground truth, diformulasikan pada persamaan (2). Ketiga metrik dihitung berdasarkan confusion matrix yang terdiri dari True Positive (TP), False Positive (FP), dan False Negative (FN) [24].

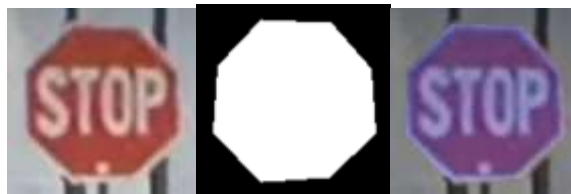
$$\text{Dice} = \frac{2TP}{2TP+FP+FN} \quad (1)$$

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \quad (2)$$

3. HASIL DAN PEMBAHASAN

3.1 Hasil Praproses Data

Tahap praproses menghasilkan pasangan data berupa citra RGB dan mask segmentasi biner untuk seluruh 2.050 citra. Pelabelan menggunakan LabelMe menghasilkan mask segmentasi yang mampu memisahkan area objek rambu dari latar belakang dengan baik, di mana piksel bernilai 1 merepresentasikan area objek rambu dan piksel bernilai 0 merepresentasikan background. Citra telah dinormalisasi ke rentang 0-1 dan dibagi menjadi data training (1.750 citra, 85%) dan testing (300 citra, 15%).



Gambar 3. Contoh Hasil Praproses Data: Citra Asli, Ground Truth Mask, dan Overlay Segmentasi

Gambar 3 memperlihatkan tiga tahapan hasil praproses data pada satu contoh citra rambu “Stop”. Bagian kiri merupakan citra asli berukuran 128×128 piksel yang diambil dari dataset. Bagian tengah adalah ground truth mask hasil konversi anotasi LabelMe, di mana piksel putih merepresentasikan area objek rambu dan piksel hitam merepresentasikan latar belakang. Bagian kanan menunjukkan hasil overlay antara citra asli dan mask segmentasi, yang memvisualisasikan akurasi anotasi terhadap bentuk dan batas tepi objek rambu. Ketiga elemen ini menjadi pasangan data (input-label) yang digunakan untuk melatih model DeepLabV3+ secara end-to-end.

3.2 Arsitektur Model DeepLabV3+

Model DeepLabV3+ yang diimplementasikan menggunakan MobileNetV2 sebagai backbone encoder. Encoder MobileNetV2 secara bertahap menurunkan resolusi spasial dari $128 \times 128 \times 3$ piksel menjadi $8 \times 8 \times 576$ piksel sambil meningkatkan channel fitur. Modul ASPP menggunakan konvolusi paralel dengan atrous rate berbeda untuk menangkap informasi multi-skala dan menghasilkan fitur berukuran $8 \times 8 \times 1280$ piksel. Selanjutnya decoder melakukan proses upsampling dan refinement fitur hingga menghasilkan output segmentasi biner berukuran $128 \times 128 \times 1$ piksel. Ringkasan arsitektur model ditampilkan pada Tabel 2.

Tabel 2. Ringkuman Arsitektur Model DeepLabV3+

Komponen	Output Shape	Keterangan
Input Layer	(None, 128, 128, 3)	Citra RGB ukuran 128×128
Encoder (MobileNetV2)	(None, 8, 8, 576)	Backbone ekstraksi fitur
ASPP Module	(None, 8, 8, 1280)	Atrous Spatial Pyramid Pooling
Decoder	(None, 128, 128, 256)	Upsampling dan refinement fitur
Output Layer	(None, 128, 128, 1)	Segmentasi biner
Total Parameter	6.525.953	Total parameter model DeepLabV3+

Berdasarkan Tabel 2, arsitektur DeepLabV3+ menggunakan MobileNetV2 sebagai backbone encoder yang mampu mengekstraksi fitur secara efisien dengan jumlah parameter yang relatif ringan. Encoder menghasilkan fitur berukuran $8 \times 8 \times 576$ piksel, kemudian diproses oleh modul ASPP untuk menangkap informasi multi-skala hingga menghasilkan fitur berukuran $8 \times 8 \times 1280$ piksel. Selanjutnya decoder melakukan proses upsampling dan refinement fitur sehingga model mampu menghasilkan output segmentasi biner berukuran $128 \times 128 \times 1$ piksel. Penggunaan MobileNetV2 juga membantu proses pelatihan model berjalan optimal pada platform Google Colab dengan kebutuhan komputasi yang lebih efisien.

Arsitektur DeepLabV3+ yang digunakan mampu mempertahankan detail objek melalui proses penggabungan fitur resolusi rendah dan resolusi tinggi pada bagian decoder. Hal ini membantu model dalam menghasilkan segmentasi area rambu lalu lintas yang lebih jelas meskipun objek memiliki ukuran yang bervariasi. Selain itu, jumlah parameter model sebesar 6.525.953 masih tergolong efisien sehingga proses pelatihan dapat dilakukan dengan baik pada lingkungan komputasi terbatas.

3.3 Hasil Pelatihan Model

Proses pelatihan model DeepLabV3+ dilakukan selama 40 epoch dengan total 437 langkah per epoch menggunakan batch size 4. Tabel 3 menampilkan perkembangan metrik pada data pelatihan pada beberapa epoch

representatif. Selama proses pelatihan, nilai loss model mengalami penurunan secara bertahap seiring bertambahnya epoch, sedangkan nilai akurasi dan IoU menunjukkan peningkatan yang stabil. Hal ini menunjukkan bahwa model mampu mempelajari pola segmentasi objek rambu lalu lintas dengan baik pada data pelatihan maupun data validasi.

Tabel 3. Hasil Pelatihan Model pada Epoch Representatif

Epoch	Train Loss	Accuracy	IoU	Dice Coefficient
1	0,1705	0,9590	0,9089	0,9513
10	0,0897	0,9771	0,9476	0,9730
20	0,0730	0,9813	0,9567	0,9778
30	0,0625	0,9841	0,9629	0,9811
40	0,0543	0,9861	0,9676	0,9835

Berdasarkan Tabel 3, nilai train loss mengalami penurunan konsisten dari 0,1705 pada epoch pertama menjadi 0,0543 pada epoch ke-40. Nilai accuracy, IoU, dan Dice Coefficient meningkat secara bertahap hingga mencapai nilai terbaik 0,9861; 0,9676; dan 0,9835 pada akhir pelatihan, menunjukkan model semakin mampu mempelajari pola segmentasi. Pada data validasi, nilai validation loss terbaik diperoleh pada epoch pertama sebesar 0,1670 dan nilai val_accuracy menunjukkan performa stabil di kisaran 0,95-0,96. Pola ini menunjukkan bahwa model memiliki kemampuan generalisasi yang baik, sehingga bobot model terbaik pada epoch pertama dengan validation loss minimum sebesar 0,1670 disimpan sebagai model final.

3.4 Hasil Evaluasi Model

Evaluasi dilakukan menggunakan data pengujian yang tidak dipakai selama proses pelatihan. Model yang dievaluasi merupakan model dengan parameter terbaik (validation loss minimum = 0,1670 pada epoch 1). Tabel 4 menyajikan perbandingan hasil evaluasi pada ketiga subset data.

Tabel 4. Hasil Evaluasi Model DeepLabV3+ pada Seluruh Data

Metrik Evaluasi	Data Pelatihan	Data Validasi	Data Pengujian
Loss	0,0543	0,1670	0,2543
Accuracy	0,9861	0,9606	0,9359 (93,59%)
IoU	0,9676	0,9054	0,7912 (79,12%)
Dice Coefficient	0,9835	0,9499	0,7996 (79,96%)

Berdasarkan Tabel 4, model DeepLabV3+ menghasilkan akurasi 93,59% pada data uji, yang menunjukkan bahwa sebagian besar piksel dapat dikategorikan dengan benar. Nilai Dice Coefficient sebesar 79,96% menunjukkan tingkat kesesuaian yang baik antara hasil prediksi dengan ground truth mask. Selain itu, nilai IoU sebesar 79,12% mengindikasikan kemampuan model dalam menghasilkan area tumpang tindih yang baik antara hasil prediksi dan ground truth. Perbedaan nilai metrik antara data pelatihan dan pengujian menunjukkan adanya gap performa yang wajar. Namun nilai IoU dan Dice Coefficient pada data uji yang mencapai 79,12% dan 79,96% melampaui nilai mIoU 73,3% yang dicapai Wijaya dkk. (2025) menggunakan arsitektur DeepLabV3+ dengan backbone ResNet18 [9], membuktikan keunggulan model yang diusulkan. Visualisasi hasil segmentasi menunjukkan bahwa area objek rambu lalu lintas berhasil dipisahkan dari latar belakang dengan cukup baik. Pada hasil segmentasi, area objek direpresentasikan dengan area objek rambu ditampilkan berwarna putih, sedangkan latar belakang ditampilkan dengan warna hitam. Hasil ini membuktikan model mampu mengenali dan melakukan segmentasi area rambu lalu lintas secara efektif meskipun terdapat variasi latar belakang pada citra input.

Meskipun hasil segmentasi telah menunjukkan performa yang baik, masih terdapat beberapa prediksi yang kurang optimal pada citra dengan pencahayaan rendah atau latar belakang yang kompleks. Kondisi tersebut menyebabkan batas objek rambu dan background menjadi kurang jelas sehingga mempengaruhi nilai IoU dan Dice Coefficient. Namun secara keseluruhan, model tetap mampu mempertahankan performa segmentasi yang stabil pada sebagian besar data pengujian.

3.5 Perbandingan dengan Penelitian Terdahulu

Tabel 5 menampilkan perbandingan performa model yang diusulkan dengan penelitian-penelitian terdahulu yang relevan.

Tabel 5. Perbandingan Kinerja Model dengan Penelitian Terdahulu

Peneliti	Metode	Accuracy	IoU/mIoU	Dice/F1
Wahyuningrum dkk. (2023)	DeepLabV3+ (modifikasi)	-	-	0,9691
Wijaya dkk. (2025)	DeepLabV3+ + ResNet18	0,923	0,733	0,922
Penelitian ini	DeepLabV3+ + MobileNetV2	0,9359	0,7912	0,7996

Berdasarkan Tabel 5, model yang diusulkan mampu melampaui nilai mIoU penelitian Wijaya dkk. (2025) yang menggunakan backbone ResNet18. Hasil ini mengonfirmasi bahwa DeepLabV3+ dengan backbone

MobileNetV2 merupakan pilihan yang efisien untuk segmentasi citra rambu lalu lintas dengan keseimbangan antara akurasi dan efisiensi komputasi.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan dan mengimplementasikan model segmentasi citra rambu lalu lintas menggunakan arsitektur DeepLabV3+ berbasis Convolutional Neural Network dengan backbone MobileNetV2. Model dilatih selama 40 epoch menggunakan optimizer Adam dengan learning rate 0,0002 dan fungsi loss kombinasi Weighted Sparse Categorical Cross-Entropy dan Dice Loss pada dataset 2.050 citra rambu lalu lintas Indonesia yang terdiri dari 10 jenis rambu. Hasil evaluasi pada data pengujian menunjukkan bahwa model berhasil meraih akurasi sebesar 93,59%, Dice Coefficient sebesar 79,96%, dan IoU sebesar 79,12%. Nilai IoU yang dicapai melampaui nilai mIoU 73,3% dari penelitian sebelumnya menggunakan arsitektur yang sama namun dengan backbone berbeda, membuktikan bahwa DeepLabV3+ dengan MobileNetV2 mampu melakukan segmentasi citra rambu lalu lintas secara efektif. Penggunaan backbone MobileNetV2 juga memberikan keuntungan dari sisi efisiensi komputasi karena jumlah parameter model yang relatif ringan, sehingga proses pelatihan dapat dilakukan secara optimal pada platform Google Colab dengan sumber daya GPU terbatas. Selain itu, modul ASPP pada DeepLabV3+ membantu model dalam menangkap informasi objek pada berbagai skala sehingga hasil segmentasi mampu mempertahankan bentuk area rambu lalu lintas dengan baik. Untuk penelitian selanjutnya, disarankan agar dataset diperbanyak dan proses anotasi dilakukan lebih beragam untuk meningkatkan kemampuan generalisasi model. Perbandingan langsung dengan arsitektur lain seperti U-Net atau SegNet, serta penggunaan backbone yang lebih kompleks dan teknik augmentasi data yang lebih variatif, diharapkan dapat meningkatkan nilai IoU dan Dice Coefficient secara signifikan. Model yang dihasilkan berpotensi diintegrasikan ke dalam sistem ADAS untuk mendukung keselamatan berkendara di Indonesia.

UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada Allah SWT atas rahmat dan karunia-Nya sehingga penelitian ini dapat diselesaikan dengan baik. Penulis juga menyampaikan rasa terima kasih yang sebesar-besarnya kepada kedua orang tua tercinta, terutama kepada ibu yang selalu merawat, membiayai, mendoakan, serta memberikan kasih sayang, dukungan, dan semangat tanpa henti selama penulis menempuh pendidikan hingga penyusunan jurnal ini dapat terselesaikan. Selain itu, penulis juga mengucapkan terima kasih kepada teman-teman seperjuangan yang telah memberikan bantuan, motivasi, dan semangat selama proses penelitian dan penulisan jurnal ini.

REFERENCES

- [1] R. Silaban and I. M. Pase, "TINJAUAN YURIDIS SANKSI PIDANA TERHADAP PELAKU PELANGGARAN LALU LINTAS MENURUT UNDANG-UNDANG NOMOR 22 TAHUN 2009 TENTANG LALU LINTAS DAN ANGKUTAN JALAN," *jurnalrectum*, vol. 3, no. 1, p. 107, Jan. 2021, doi: 10.46930/jurnalrectum.v3i1.823.
- [2] M. E. Prasetyo, M. R. Faza, R. Pratama, S. N. H. Alhabsy, H. Purwanti, and A. P. A. Masa, "Klasifikasi Ragam Kendaraan Menggunakan Metode Convolutional Neural Network (Cnn)," *ATASI*, vol. 2, no. 2, pp. 142–148, Dec. 2023, doi: 10.30872/atasi.v2i2.1156.
- [3] M. Akbar, A. S. Purnomo, and S. Supatman, "Multi-Scale Convolutional Networks untuk Pengenalan Rambu Lalu Lintas di Indonesia," *SISFOKOM*, vol. 11, no. 3, pp. 310–315, Dec. 2022, doi: 10.32736/sisfokom.v11i3.1452.
- [4] M. Akbar, "Pengenalan Rambu Lalu-lintas menggunakan Convolutional Neural Network (Studi Kasus: Rambu Lalu-lintas Indonesia)," *InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan*, vol. 6, no. 2, pp. 272–276, Mar. 2022, doi: 10.30743/infotekjar.v6i2.4564.
- [5] A. J. Rindengan and M. Mananohas, "PERANCANGAN SISTEM PENENTUAN TINGKAT KESEGERAN IKAN CAKALANG MENGGUNAKAN METODE CURVE FITTING BERBASIS CITRA DIGITAL MATA IKAN," *JIS*, vol. 17, no. 2, p. 161, Dec. 2017, doi: 10.35799/jis.17.2.2017.18128.
- [6] O. D. Nurhayati, D. N. Afifah, . N., and N. Rustanti, "Segmentasi Thresholding untuk Pemilihan Kualitas Telur Asin," *J. Sistem Info. Bisnis*, vol. 8, no. 1, p. 42, Apr. 2018, doi: 10.21456/vol8iss1pp42-48.
- [7] M. U. Habibah and M. Kurniawan, "Segmentasi Citra Wajah dengan Implementasi Adaptif Threshold-Integral Image," *JTIK*, vol. 8, no. 5, pp. 919–928, Oct. 2021, doi: 10.25126/jtiik.2021853840.
- [8] R. T. Wahyuningrum, M. Jannah, B. D. Satoto, A. K. Sari, and A. D. Sensusiaty, "Segmentasi Citra X-Ray Dada Menggunakan Metode Modifikasi Deeplabv3+," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 10, no. 3, pp. 687–698, Jul. 2023, doi: 10.25126/jtiik.2023106754.

-
- [9] B. S. Wijaya, R. Munir, and N. P. Utama, "Segmentasi Awan pada Citra Satelit Multispektral Menggunakan Convolutional Neural Networks," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 12, no. 6, pp. 1251–1260, Dec. 2025, doi: 10.25126/jtiik.2025125.
- [10] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," in *Computer Vision – ECCV 2018*, vol. 11211, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., in Lecture Notes in Computer Science, vol. 11211., Cham: Springer International Publishing, 2018, pp. 833–851. doi: 10.1007/978-3-030-01234-2_49.
- [11] M. M. Subagio, M. S. Bhakti, A. Y. Yulestiono, and A. P. Sari, "Perbandingan Kinerja Metode Convolutional Neural Network (CNN) dan VGG-16 dalam Klasifikasi Rambu Lalu Lintas," *Jurnal Mahasiswa Teknik Informatika*, vol. 3, no. 2, pp. 79–87, Oct. 2024, doi: 10.35473/jamastika.v3i2.3361.
- [12] T. T. Zalukhu, A. R. Himamunanto, and H. Budiati, "Sistem Pengenalan Citra Dokumen Teks Terdistorsi menjadi Teks Menggunakan Metode Deep Learning," *Jurnal JTik (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 10, no. 1, pp. 1–14, Jan. 2026, doi: 10.35870/jtik.v10i1.4700.
- [13] M. F. Ikhsal, B. A. Dermawan, and R. I. Adam, "Peningkatan Deteksi Kecelakaan di Jalan Raya Menggunakan Real-ESRGAN pada Citra CCTV Persimpangan Jalan," *JAIC*, vol. 7, no. 1, pp. 51–56, Jul. 2023, doi: 10.30871/jaic.v7i1.5562.
- [14] M. Akbar, "Pengenalan Rambu Lalu-lintas menggunakan Convolutional Neural Network (Studi Kasus: Rambu Lalu-lintas Indonesia)," *InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan*, vol. 6, no. 2, pp. 272–276, Mar. 2022, doi: 10.30743/infotekjar.v6i2.4564.
- [15] D. Irfan, R. Rosnelly, M. Wahyuni, J. T. Samudra, and A. Rangga, "PERBANDINGAN OPTIMASI SGD, ADADELTA, DAN ADAM DALAM KLASIFIKASI HYDRANGEA MENGGUNAKAN CNN," *JSSR*, vol. 5, no. 2, p. 244, Jun. 2022, doi: 10.54314/jssr.v5i2.789.
- [16] A. A. G. Bagus Janapriya, "Pengenalan Jenis Rambu Lalu Lintas menggunakan Metode YOLO V5," *JLK*, vol. 11, no. 4, p. 1011, Apr. 2023, doi: 10.24843/JLK.2023.v11.i04.p32.
- [17] N. Yudistira, A. W. Widodo, and B. Rahayudi, "Deteksi Covid-19 pada Citra Sinar-X Dada Menggunakan Deep Learning yang Efisien," *JTIK*, vol. 7, no. 6, p. 1289, Dec. 2020, doi: 10.25126/jtiik.2020763651.
- [18] M. A. Hidayat, N. L. Husni, and F. Damsi, "Pendeteksi Banjir Dengan Image Processing Berbasis Convolutional Neural Network (CNN) pada Kamera Pengawas: Image Processing Based Flood Detector Using Convolutional Neural Network (CNN) Within Surveillance Camera," *MALCOM*, vol. 2, no. 2, pp. 10–18, Aug. 2022, doi: 10.57152/malcom.v2i2.382.
- [19] B. Nugroho and E. Y. Puspaningrum, "Kinerja Metode CNN untuk Klasifikasi Pneumonia dengan Variasi Ukuran Citra Input," *JTIK*, vol. 8, no. 3, pp. 533–538, Jun. 2021, doi: 10.25126/jtiik.2021834515.
- [20] P. A. Nugroho, I. Fenriana, R. Arijanto, and M. Kom, "IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN) PADA EKSPRESI MANUSIA," vol. 2, no. 1, 2020.
- [21] D. Irfan, R. Rosnelly, M. Wahyuni, J. T. Samudra, and A. Rangga, "PERBANDINGAN OPTIMASI SGD, ADADELTA, DAN ADAM DALAM KLASIFIKASI HYDRANGEA MENGGUNAKAN CNN," *JSSR*, vol. 5, no. 2, p. 244, Jun. 2022, doi: 10.54314/jssr.v5i2.789.
- [22] A. Peryanto, A. Yudhana, and R. Umar, "Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network," *Format : Jurnal Ilmiah Teknik Informatika*, vol. 8, no. 2, pp. 138–147, Feb. 2020, doi: 10.22441/format.2019.v8.i2.007.
- [23] Subhan Bole Boly and Mutaqin Akbar, "Segmentasi Citra Sel Darah Menggunakan Convolutional Neural Network," *Restikom*, vol. 6, no. 2, pp. 390–398, Aug. 2024, doi: 10.52005/restikom.v6i2.336.
- [24] A. W. Setiawan, "Image Segmentation Metrics in Skin Lesion: Accuracy, Sensitivity, Specificity, Dice Coefficient, Jaccard Index, and Matthews Correlation Coefficient," in *2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, Surabaya, Indonesia: IEEE, Nov. 2020, pp. 97–102. doi: 10.1109/CENIM51130.2020.9297970.