Penerapan *Convolutional Neural Network* untuk Segmentasi Semantik Citra Perkotaan

Daniel Alexander[#1], Hery Heryanto[#2], Hans Christian Kurniawan[#3]
Departemen Informatika, Institut Teknologi Harapan Bangsa
Jalan Dipatiukur No. 80-84, Bandung, Indonesia 40132
[1]realdanielalexander@gmail.com
[2]hery_heryanto@ithb.ac.id
[3]hans_christian@ithb.ac.id

Abstract*—Semantic segmentation is the process of classifying each pixel in an image into one of the predesignated classes. In its application for self-driving systems, semantic segmentation is used to identify objects and road conditions to make the right control decisions. This work uses the convolutional neural network method to perform semantic segmentation with the DeepLabV3+ architecture. Object of study is the urban scene dataset called Bandung Cityscapes collected around the city of Bandung, West Java, Indonesia and annotated by hand. Architecture testing is performed by combining the values of epoch, learning rate, and number of convolution filters to find the best mean intersection over union accuracy. The most optimal model achieved the highest average accuracy of 77.431% and an average prediction time of 74.890 ms on the validation set, attained by the model with 128 convolutional filters, trained with 300 epochs and 0.0001 (10-4) learning rate. This model maintained an average training accuracy of 87.194%, average validation accuracy of 75.274%, and average difference between the two of 11.919% throughout the training process.

***Index Terms*—computer vision, semantic segmentation, urban scenes, Convolutional Neural Network (CNN), DeepLabV3+**

Abstrak—Segmentasi semantik adalah proses mengklasifikasikan setiap *pixel* pada citra ke dalam salah satu *class*. Dalam aplikasinya pada sistem swakemudi, segmentasi semantik citra perkotaan dilakukan untuk mengenali objek dan kondisi jalan guna mengambil keputusan kendali yang tepat. Pada penelitian ini, digunakan metode *convolutional neural network* dengan arsitektur DeepLabV3+ untuk melakukan segmentasi semantik. Objek penelitian adalah *dataset* citra perkotaan Bandung Cityscapes yang dikumpulkan di daerah Kota Bandung, Jawa Barat, Indonesia dan diberi anotasi secara manual. Pengujian arsitektur dilakukan dengan mengkombinasikan nilai *epoch*, *learning rate*, dan jumlah *filter* konvolusi untuk mencari nilai akurasi *mean intersection over union* yang terbaik. Model paling optimal mencapai akurasi rata-rata tertinggi sebesar 77.431% dan rata-rata waktu prediksi 74.890 ms pada dataset validasi, dicapai oleh model dengan 128 *filter* konvolusi, dilatih dengan 300 *epoch* dan *learning rate* 0.0001 (10^{-4}). Model tersebut mempertahankan rata-rata akurasi pelatihan sebesar 87.194%, rata-rata akurasi validasi sebesar 75.274%, dan rata-rata perbedaan antara keduanya sebesar 11.919% selama proses pelatihan.

***Kata kunci*—computer vision, segmentasi semantik, citra perkotaan, *Convolutional Neural Network* (CNN), DeepLabV3+**

I. Pendahuluan

Pemahaman akan citra digital menjadi fokus dalam banyak penelitian terkini di bidang *computer vision*, salah satunya adalah segmentasi semantik citra [3]. Segmentasi semantik adalah proses mengklasifikasikan setiap *pixel* pada citra ke dalam salah satu *class*. Secara umum, proses segmentasi semantik citra terbagi ke dalam dua tahapan, yaitu tahapan ekstraksi fitur dan tahapan klasifikasi ke dalam salah satu *class*. Sebuah fitur adalah sifat unik pada citra yang membedakan antara satu citra dengan citra lainnya untuk menyelesaikan suatu proses komputasi tertentu [1]. Dahulu, ekstraksi fitur dilakukan dengan mengaplikasikan suatu algoritme tertentu, yang disebut juga dengan *hand-engineered features*. Beberapa contoh algoritme yang digunakan untuk mengekstraksi fitur dari citra adalah *Histogram of Oriented Gradients* (HOG), *Scale-Invariant Feature Transform* (SIFT), dan *Speeded-Up Robust Features* (SURF). Tahap selanjutnya adalah menentukan *class* dari masing masing *pixel*. Algoritma *machine learning* umum digunakan pada tahap ini, sebagai contoh *Support Vector Machine* (SVM) dan *Random Decision Forest* (RDF). Akan tetapi, pendekatan dalam dua tahapan ini membutuhkan *preprocessing* citra dan *domain knowledge* yang menyeluruh. Permasalahan ini diselesaikan dengan pendekatan berbasis *deep learning*, khususnya *Convolutional Neural Network* (CNN). Pendekatan tradisional berbasis *hand-engineered features* dan *machine learning* sudah tergantikan dengan pendekatan deep learning, yang sudah terbukti lebih efisien dan efektif dengan ketersediaan sumber daya komputasi dan data yang kian meningkat.

Berdasarkan penelitian yang sudah ada [5]— [9], penelitian ini menggunakan metode CNN dengan arsitektur DeepLabV3+ [9] untuk melakukan segmentasi semantik citra perkotaan. Arsitektur tersebut dipilih karena menunjukkan performa *state-of-the-art* dataset test PASCAL VOC 2012 sebesar 85.7%. Objek penelitian adalah dataset citra perkotaan Bandung Cityscapes yang dikumpulkan di sekitar daerah Kota Bandung, Jawa Barat, Indonesia. Pengujian dilakukan dengan mengombinasikan nilai *epoch*, *learning rate*, dan jumlah *filter* konvolusi untuk mencari nilai akurasi mean intersection over union yang terbaik.

II. Metodologi

A. Citra Digital

Sebuah citra digital didefinisikan sebagai fungsi dua dimensi $f(x, y)$, di mana x dan y adalah koordinat bidang, dan amplitudo f pada pasangan koordinat (x, y) adalah intensitas

atau nilai keabuan pada titik tersebut. Sebuah citra digital terdiri atas sejumlah elemen terkecil yang disebut pixel. Pixel diperoleh dari proses digitasi sinyal citra kontinu yang terdiri *sampling*, yaitu proses digitasi dengan cara mengambil beberapa titik pada citra kontinu, dan *quantization*, yaitu pembulatan titik-titik tersebut ke dalam nilai keabuan terhingga [2].

B. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu kategori neural network yang paling banyak digunakan dalam pemrosesan data berdimensi tinggi seperti gambar dan video. Pada CNN, setiap unit pada *network* digantikan dengan sebuah *filter* yang kemudian dilakukan operasi konvolusi dengan masukan unit tersebut. Komponen pembangun yang paling umum pada CNN adalah *convolutional layer*, *activation layer*, dan *pooling layer* [1].

C. Overfitting dan Underfitting

Pada bidang *machine learning*, salah satu masalah utama yang perlu diatasi adalah bahwa model yang dibuat harus berhasil memprediksi data baru yang belum pernah dipelajari sebelumnya. Kemampuan ini disebut *generalization*. Pengukuran kesalahan yang dihitung pada *training set* disebut juga dengan *training error*, sedangkan pengukuran kesalahan yang dihitung pada *testing set* disebut dengan *generalization error* atau *test error*. Model yang optimal diukur dari dua faktor, yaitu *training error* yang rendah dan selisih antara *training error* dan *generalization error* yang rendah. Kedua faktor tersebut berhubungan dengan dua masalah utama pada bidang *machine learning*, yaitu *underfitting*, yaitu ketika model memiliki nilai *error* yang rendah pada *training set*, dan *overfitting*, yaitu ketika nilai selisih antara *training error generalization error* terlalu besar. Untuk mencapai pembangunan model yang optimal, digunakan strategi *regularization*. Teknik *regularization* yang digunakan dalam penelitian ini adalah pendekatan level data dengan teknik *image data augmentation* dan normalisasi statistik dengan melibatkan *batch normalization*.

D. Tinjauan Objek

Objek dalam penelitian ini adalah dataset Bandung Cityscapes yang dikumpulkan di daerah Kota Bandung, Jawa Barat, Indonesia dan diberikan anotasi secara manual. Citra ditangkap dengan menggunakan ponsel pintar yang disematkan pada dasbor mobil. Proses pengambilan dataset dilakukan dengan mode video pada pukul 14:00 dalam waktu kurang lebih 30 menit menyusuri rute sepanjang 7.67 kilometer. Video dengan format 1920 × 1080 berdurasi 30 menit yang dihasilkan kemudian diekstraksi setiap 30 frame menggunakan fitur *scene video filter* pada kakas VLC, dengan konfigurasi citra keluaran yang diubah menjadi ukuran 960 × 720. Dari keseluruhan citra yang diekstrak dari keluaran video, dipilih 300 buah citra berdasarkan beberapa skenario yang merepresentasikan kondisi berkendara sehari-hari. 300 buah citra yang dipilih kemudian diberikan anotasi secara hati-hati menggunakan fitur *pen tool* pada kakas Adobe Photoshop dengan lama anotasi per setiap citra sekitar 15 menit dan total waktu anotasi untuk keseluruhan dataset sekitar 150 jam. Seluruh data citra dan labelnya kemudian diambil setiap lima citra sebagai data masukan proses validasi. Proses pembagian ini dilakukan untuk membagi *dataset* berdasarkan komposisi 80% (240 citra) sebagai data masukan proses pelatihan dan 20% (60 citra) sebagai data masukan proses validasi. Proses validasi penting dilakukan untuk melihat kecenderungan *overfitting* dan *underfitting*. Tabel I menunjukkan bahwa berdasarkan perincian proporsi untuk setiap *class* pada data pelatihan dan validasi, kedua kategori data memiliki proporsi yang relatif sama.

Tabel I
PERINCIAN PROPORSI JUMLAH *Pixel* UNTUK MASING-MASING *Class*

Class	Total Proporsi	Total Proporsi	
		Pelatihan	Validasi
Tree	37.6%	37.7%	37.1%
Building	17.5%	17.5%	17.6%
Sky	16.3%	16.3%	16.5%
Road	15.7%	15.7%	15.6%
Car	5.9%	5.8%	6.3%
MotorcycleScooter	3.2%	3.2%	3.2%
Sidewalk	3.1%	3.2%	3.0%
Pedestrian	0.5%	0.5%	0.5%
SignSymbol	0.2%	0.2%	0.2%

E. Perancangan Sistem

Kerangka pemikiran dan *flowchart global* dari konfigurasi metode yang diusulkan ditunjukkan pada Gambar 1 dan 2. Pertama, citra dimuat dengan masing-masing label *ground truth* yang sesuai. Selanjutnya, dilakukan *preprocessing* dengan teknik *image data augmentation*, yaitu *horizontal flip* dan *gaussian blur*, dan konversi label menjadi bentuk *one-hot encoded*. Citra dan label kemudian dilalui ke dalam CNN untuk fase pelatihan, di mana bobot dari masing-masing *kernel* akan diperbarui. Terakhir, dilakukan pengujian CNN yang menghasilkan nilai prediksi setiap pixel pada citra dan dibandingkan dengan label yang menghasilkan nilai akurasi *mean intersection over union*. Variabel yang diuji untuk menentukan akurasi tertinggi adalah sebagai berikut:

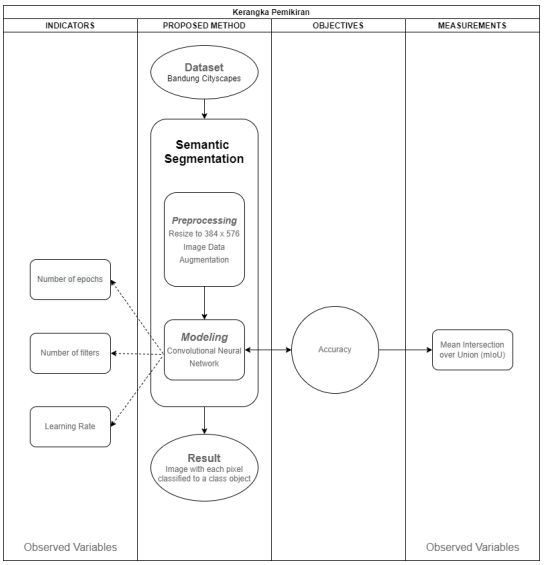
1) *Learning rate* adalah seberapa besar langkah koreksi yang diambil oleh CNN pada setiap iterasi.
2) Jumlah *filter* konvolusi menentukan banyaknya fitur yang dapat dipelajari oleh CNN.
3) *Epoch* menentukan berapa kali CNN melakukan pembaruan bobot terhadap data pelatihan.

III. HASIL DAN PEMBAHASAN

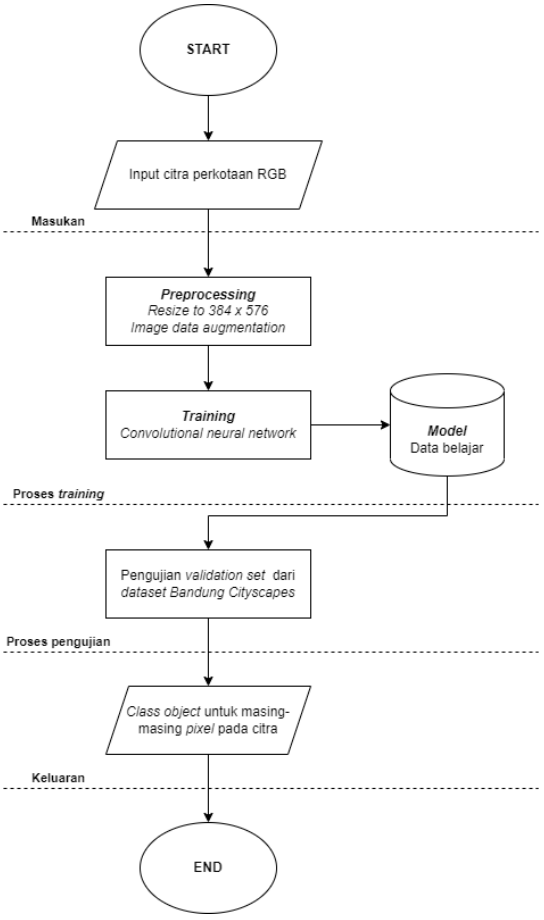
A. Hasil Pengujian

Tabel II
RANGKUMAN VARIABEL PENGUJIAN

Parameter Pengujian	Nilai yang Diujikan
Epoch	100; 200; 300
Learning rate	0,001 (10^{-3}); 0,0001 (10^{-4}); 0,00001 (10^{-5})
Jumlah filter konvolusi	64; 128; 256



Gambar 1. Kerangka pemikiran



Gambar 2. *Flowchart global* sistem segmentasi semantik

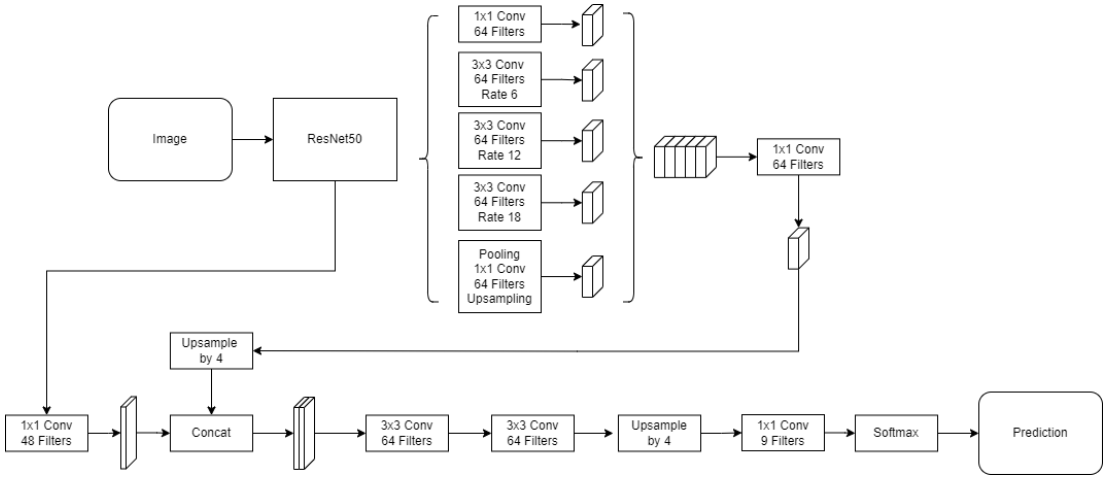
Pengujian dilakukan dengan mengelompokkan setiap kasus pengujian berdasarkan jumlah *filter* pada arsitektur, yaitu arsitektur dengan 64, 128, dan 256 *filter* konvolusi yang ditunjukkan pada Gambar 3 hingga Gambar 5. Untuk setiap arsitektur, diujikan kombinasi nilai *epoch* dan *learning rate* untuk menganalisis akurasi dan laju pergerakan akurasi seiring bertambahnya *epoch*. Pengujian kombinasi ketiga nilai variabel tersebut dilakukan untuk memilih nilai akurasi rata–rata yang terbaik. Rangkuman variabel pengujian ditunjukkan pada Tabel II Setelah pengujian dilakukan, dipilih model terbaik dari masing-masing arsitektur, yaitu model yang dilatih dengan 300 *epoch* dan *learning rate* 0,0001 (10^{-4}). Model ini dianggap terbaik karena memiliki rata-rata akurasi validasi tertinggi dan rata-rata selisih antara akurasi yang terendah.

Berdasarkan pengujian yang dilakukan, disimpulkan beberapa hal berikut. Seiring dengan peningkatan jumlah *filter* konvolusi, hasil prediksi citra semakin detail terutama di bagian tepian objek. Semakin banyak jumlah *filter* konvolusi, maka semakin banyak juga fitur yang dapat diekstrak dari citra. Selain itu, objek-objek dengan skala kecil dan berada di kejauhan lebih mudah dideteksi semakin banyak jumlah *filter* konvolusi. Meskipun demikian, semakin banyak jumlah filter, maka semakin besar kemungkinan hasil prediksi yang terlalu detail sehingga model cenderung melakukan kekeliruan prediksi. Secara umum, citra kompleks yang memiliki banyak objek pada jarak dekat lebih baik diprediksi oleh model dengan 64 *filter* konvolusi. Sebaliknya, citra dengan skala objek yang lebih kecil dan berada di kejauhan lebih baik diprediksi oleh model dengan 256 *filter* konvolusi. Citra dengan kompleksitas dan skala objek menengah lebih baik diprediksi oleh model dengan 128 *filter* konvolusi. Hasil uji kualitatif ditunjukkan pada Gambar 6. Tabel III hingga V menunjukkan rata-rata akurasi pada pengujian masing-masing arsitektur.

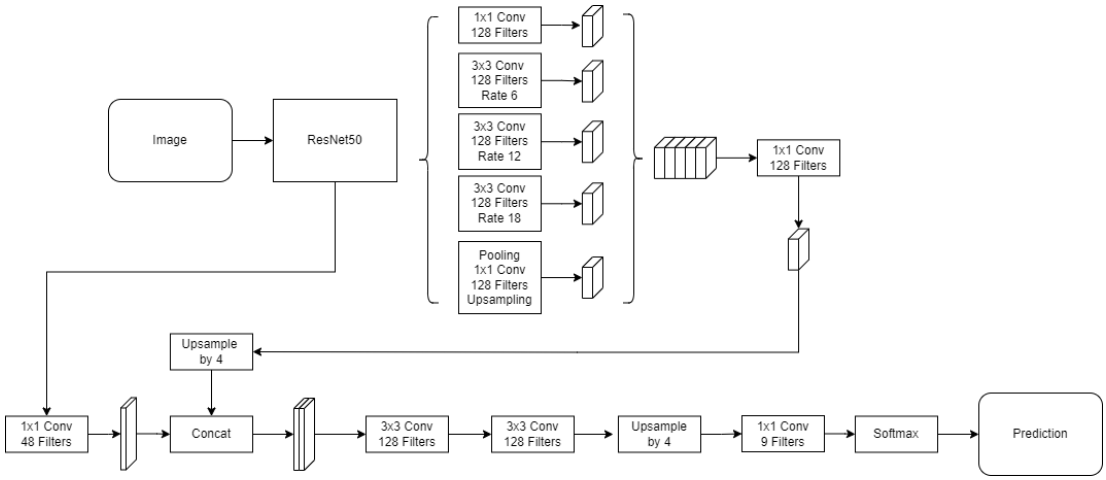
Nilai *learning rate* menentukan seberapa fluktuatif pergerakan akurasi terhadap waktu. Nilai *learning rate* terbesar yang diuji, yaitu *learning rate* 0,001 (10^{-3}) menyebabkan pergerakan akurasi yang tidak stabil dan menandakan adanya risiko *overfitting*. Nilai learning rate 10 kali lebih kecil, yaitu learning rate 0,0001 (10^{-4}), ketidakstabilan masih terjadi dalam skala yang lebih kecil. Terakhir, nilai *learning rate* yang 100 kali lebih kecil, yaitu learning rate 0,00001 (10^{-5}) menghasilkan pergerakan akurasi yang paling stabil. Dengan nilai *learning rate* yang bertambah kecil, model mencapai konvergensi dengan stabil tanpa pergerakan yang drastis pada setiap iterasinya, namun dengan konsekuensi bahwa untuk mencapai nilai akurasi yang baik dibutuhkan lebih banyak *epoch*. Model paling optimal mencapai akurasi rata-rata tertinggi sebesar 77,431% dan rata-rata waktu prediksi 74,890 ms pada dataset validasi, dicapai oleh model dengan 128 *filter* konvolusi, dilatih dengan 300 *epoch* dan *learning rate* 0,0001 (10^{-4}). Tabel VI dan VII masing-masing menunjukkan hasil pengujian waktu prediksi dan akurasi pada data validasi dengan arsitektur tersebut.

B. Pembangunan dan Penyebaran Aplikasi

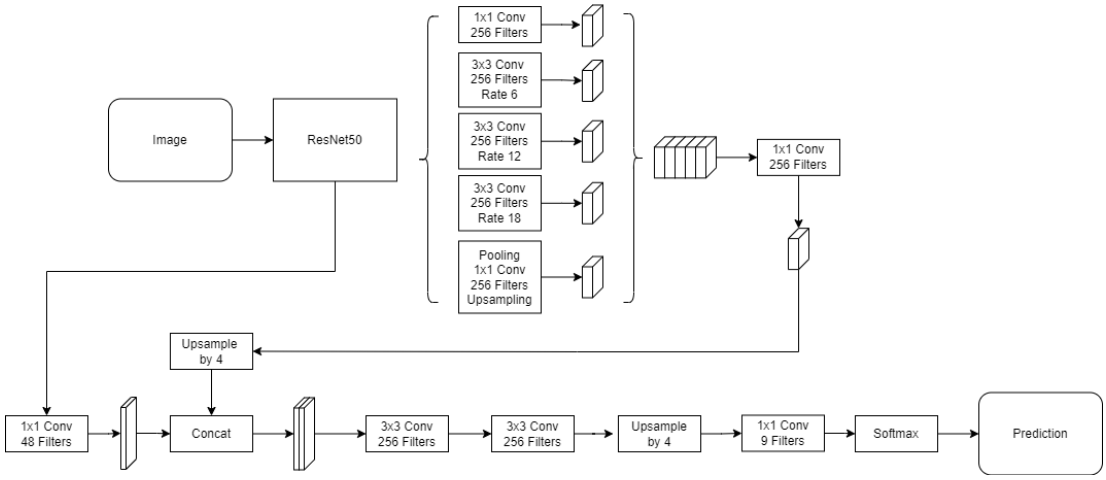
Antarmuka aplikasi dikembangkan menggunakan *framework* Flutter yang ditunjukkan pada Gambar 7, dapat diakses



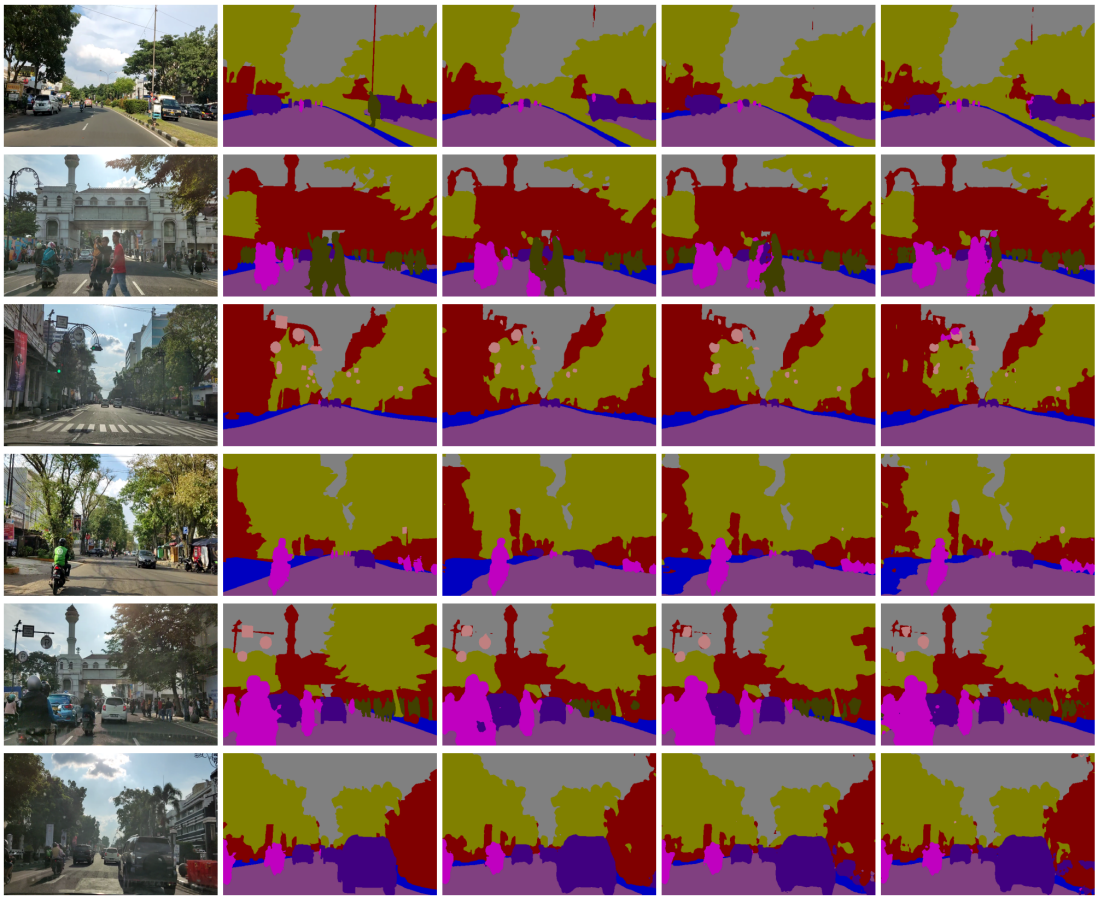
Gambar 3. Arsitektur dengan 64 *filter* konvolusi



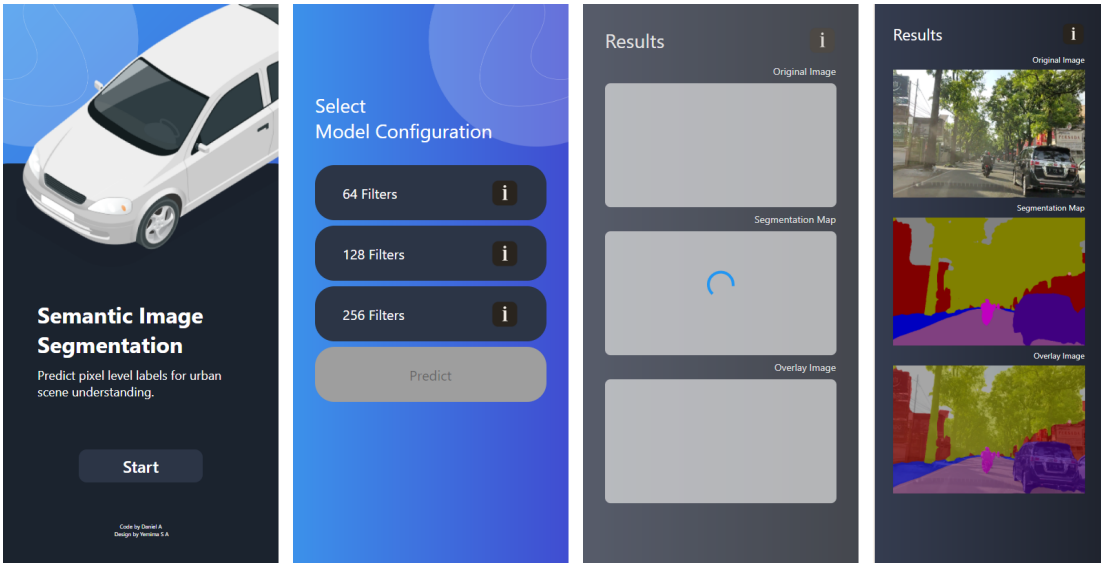
Gambar 4. Arsitektur dengan 128 *filter* konvolusi



Gambar 5. Arsitektur dengan 256 *filter* konvolusi



Gambar 6. Hasil uji kualitatif pada data validasi untuk setiap arsitektur. Dari kiri ke kanan: citra validasi, citra label, hasil prediksi arsitektur 64 *filter*, hasil prediksi arsitektur 128 *filter*, hasil prediksi arsitektur 256 *filter*



Gambar 7. Ilustrasi aplikasi

Tabel III
RATA-RATA HASIL PENGUJIAN - ARSITEKTUR DENGAN 64 FILTER KONVOLUSI

Learning Rate	Epoch	Akurasi Pelatihan	Akurasi Validasi	Selisih Akurasi
0,001	100	83,286%	71,701%	11,585%
	200	88,521%	73,909%	14,611%
	300	89,503%	74,556%	14,947%
0,0001	100	83,805%	72,943%	10,862%
	200	88,146%	74,770%	13,376%
	300	86,701%	75,423%	11,277%
0,00001	100	62,611%	59,362%	3,248%
	200	74,201%	66,466%	7,735%
	300	75.298%	66.858%	8.439%

Tabel IV
RATA-RATA HASIL PENGUJIAN - ARSITEKTUR DENGAN 128 FILTER KONVOLUSI

Learning Rate	Epoch	Akurasi Pelatihan	Akurasi Validasi	Selisih Akurasi
0,001	100	87,700%	73,359%	14,340%
	200	89,941%	74,289%	15,182%
	300	91,251%	75,904%	15,346%
0,0001	100	81,619%	72,113%	9,506%
	200	89,379%	75,152%	14,227%
	300	87,194%	75,274%	11,919%
0,00001	100	67,364%	62,666%	4,698%
	200	76,505%	67,227%	9,278%
	300	80,522%	68,887%	11,634%

Tabel V
RATA-RATA HASIL PENGUJIAN - ARSITEKTUR DENGAN 256 FILTER KONVOLUSI

Learning Rate	Epoch	Akurasi Pelatihan	Akurasi Validasi	Selisih Akurasi
0,001	100	83,175%	70,506%	12,669%
	200	87,075%	71,239%	15,835%
	300	90,489%	74,846%	16,204%
0,0001	100	86,715%	73,156%	13,559%
	200	90,391%	76,080%	14,311%
	300	91,980%	76,478%	15,501%
0,00001	100	68,669%	62,541%	6,128%
	200	78,638%	68,227%	10,410%
	300	82,437%	70,321%	12,116%

Tabel VI
HASIL PENGUJIAN WAKTU PREDIKSI PADA DATA VALIDASI

Arsitektur	Hasil Rata-Rata Waktu Prediksi
64 *Filter*	68,538 ms
128 *Filter*	74,890 ms
256 *Filter*	78,433 ms

Tabel VII
HASIL PENGUJIAN RATA-RATA AKURASI *Mean IoU* PREDIKSI PADA DATA VALIDASI

Arsitektur	Hasil Rata-Rata Akurasi *Mean IoU*
64 *Filter*	75,893%
128 *Filter*	77,431%
256 *Filter*	73,942%

melalui tautan https://semantic-segmentation.vercel.app/. Penjelasan mengenai antarmuka adalah sebagai berikut:

1) *Welcome Screen* menjelaskan secara singkat aplikasi yang dirancang.
2) *Architecture Selection Screen* memberikan pilihan jenis arsitektur yang akan digunakan.
3) *Result Screen* mengirimkan citra yang dipilih untuk dilakukan prediksi.
4) *Result Screen* setelah prediksi dilakukan menampilkan citra asal, *segmentation map*, dan *overlay* dari kedua citra.

IV. SIMPULAN

Metode *convolutional neural network* berhasil melakukan segmentasi semantik citra perkotaan di Kota Bandung dengan model paling optimal mencapai akurasi rata-rata tertinggi sebesar 77.431% dan rata-rata waktu prediksi 74.890 ms pada dataset validasi, dicapai oleh model dengan 128 *filter* konvolusi, dilatih dengan 300 *epoch* dan *learning rate* 0,0001 (10^{-4}). Model tersebut mempertahankan rata-rata akurasi pelatihan sebesar 87.194%, rata-rata akurasi validasi sebesar 75.274%, dan rata-rata selisih antara keduanya sebesar 11.919%. Model dengan konfigurasi tersebut cocok dipilih dari sisi akurasi maupun waktu prediksi. Meskipun demikian, perlu diingat bahwa pada konfigurasi model tersebut masih ditemukan adanya selisih antara nilai akurasi pelatihan dan akurasi validasi yang menandakan adanya overfitting pada data latih. Untuk aplikasi di lapangan, perlu diselidiki seberapa jauh dan detail prediksi yang dibutuhkan untuk menentukan model yang paling tepat.

DAFTAR REFERENSI

[1] S. Khan, et al., *A Guide to Convolutional Neural Networks for Computer Vision*, 1st ed, Gérard Medioni and Sven Dickinson, Ed. California: Morgan and Claypool, 2018.

[2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2^{nd}ed. Prentice Hall, 1992.

[3] R. Szeliski, *Computer Vision Algorithms and Applications*, Springer, 2011.

[4] S. Haykin, *Neural Networks and Learning Machines*, 3^{rd} ed. New Jersey: Pearson Education Inc., 2009, pp.10-15, 123-125, 131-141.

[5] J. Long, et al. "Fully Convolutional Networks for Semantic Segmentation." 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, https://doi.org/10.1109/cvpr.2015.7298965.

[6] O. Ronneberger, et al. "U-Net: Convolutional Networks for Biomedical Image Segmentation." Lecture Notes in Computer Science, 2015, pp. 234–241., https://doi.org/10.1007/978-3-319-24574-4_28.

[7] V. Badrinarayanan, et al. "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 12, 2017, pp. 2481–2495., https://doi.org/10.1109/tpami.2016.2644615.

[8] H. Zhao, et al. "Pyramid Scene Parsing Network." 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, https://doi.org/10.1109/cvpr.2017.660.

[9] L.C. Chen, et al. "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation." Computer Vision ECCV 2018, 2018, pp. 833–851., https://doi.org/10.1007/978-3-030-01234-2_49.