

Pengembangan Sistem Deteksi Hunian Parkir Menggunakan Metode Convolutional Neural Network

by john Doe

Submission date: 24-Nov-2020 12:40AM (UTC+0700)

Submission ID: 1453239064

File name: paper_publicasi_ilmiah_final_tanpa_penulis.docx (1.21M)

Word count: 3023

Character count: 18635

Pengembangan Sistem Deteksi Hunian Parkir Menggunakan Metode *Convolutional Neural Network*

Abstract—Kurangnya informasi mengenai area parkir merupakan masalah yang saat ini dihadapi oleh para pengendara. Seiring berjalannya waktu, jumlah kendaraan semakin meningkat. Peningkatan jumlah kendaraan dengan lahan parkir yang dibutuhkan seringkali tidak seimbang dimana hal ini dapat menimbulkan kemacetan di berbagai tempat, salah satunya yaitu di area parkir. Untuk mendapatkan tempat parkir, seringkali pengendara harus berkeliling ke seluruh area parkir, bahkan pengendara harus keluar dari area parkir tersebut karena tidak dapat menemukan tempat parkir untuknya. Pada penelitian ini, penulis mengembangkan sistem deteksi hunian parkir menggunakan metode *convolutional neural network* yang diimplementasikan menggunakan aplikasi android, sehingga pengendara dapat menggunakannya untuk mencari informasi dimana letak tempat parkir yang kosong.

Keywords—*Convolutional neural network, parkir, android.*

I. PENDAHULUAN

Berdasarkan data dari [1], jumlah pengguna kendaraan bermotor semakin meningkat dari tahun 2016 sampai tahun 2018. Hal ini dikarenakan alat transportasi memang sudah menjadi kebutuhan pokok bagi masyarakat dari berbagai kalangan [2]. Timbulnya kemacetan di berbagai tempat, salah satunya di area parkir baik itu di pusat perbelanjaan maupun tempat wisata terjadi karena tidak seimbangnya jumlah kendaraan yang ada dengan jumlah tempat parkir yang dibutuhkan [3].

Sulitnya mencari tempat parkir yang kosong merupakan salah satu masalah yang disebabkan oleh padatnya jumlah kendaraan yang terus meningkat. Akibatnya, terjadi antrian panjang yang bahkan sampai mengganggu arus kendaraan di jalan raya [4]. Untuk mendapatkan tempat parkir seringkali pengendara harus berkeliling ke seluruh tempat parkir yang ada, bahkan pengendara harus keluar dari area parkir karena tidak menemukan tempat parkir untuknya [5]. Kurangnya informasi mengenai area parkir menjadi sebuah masalah yang saat ini dihadapi yang mana dapat mengganggu kenyamanan pengendara baik wisatawan ataupun *customers* dari sebuah pusat perbelanjaan.

Deteksi hunian parkir merupakan salah satu solusi untuk mengetahui kondisi dari sebuah hunian parkir. Beberapa area parkir seperti mall atau area perkantoran menggunakan alat sensor untuk menentukan status suatu hunian parkir. Dalam proses instalasi dan pemeliharaan sensor dibutuhkan biaya yang mungkin tinggi, khususnya di area parkir yang memuat banyak hunian parkir, hal ini dikarenakan setiap hunian parkirnya dibutuhkan satu sensor [6] [7].

Pada penelitian sebelumnya [8] terdapat metode untuk mendeteksi hunian parkir dengan menggunakan pengolahan citra, yaitu *Histogram of Oriented Gradient* (HOG). Dalam HOG jarak kamera dengan objek sangatlah menentukan hasil deteksi hunian parkir. Semakin dekat jarak kamera dengan hunian parkir maka semakin tinggi akurasi yang didapat,

tetapi cakupan slot parkir semakin sedikit, begitu juga sebaliknya, semakin jauh jarak kamera dengan slot parkir maka akurasi semakin kecil bahkan bisa mencapai 0% tetapi cakupan slot parkir semakin banyak.

Salah satu metode yang dapat mengatasi kelemahan pada metode HOG adalah *Convolutional Neural Network* (CNN). CNN terdiri dari sejumlah lapisan tersembunyi dimana masing-masing melakukan perhitungan matematika dari *input* neuron dan menghasilkan *output* untuk setiap lapisannya berdasarkan nilai *weight*, bias, dan *activation function*. Untuk melakukan deteksi dibutuhkan proses pelatihan pada model CNN menggunakan data gambar berlabel [7]. CNN memiliki beberapa model arsitektur diantaranya *Alexnet*, *VGG16*, *GoogleNet*, dan *LetNet*. Berdasarkan pada penelitian sebelumnya [7], [9]–[11] yang telah membandingkan akurasi dari hasil deteksi hunian parkir pada setiap model arsitektur CNN. *AlexNet* dengan akurasi 91 – 99,77%, *VGG16* dengan akurasi 87 – 99,83%, *GoogleNet* dengan akurasi 88,29% - 98%, dan *LetNet* dengan akurasi 84,3 – 98,3%.

Pada penelitian ini akan menggunakan metode *Convolutional Neural Network* dikarenakan metode ini mampu mengatasi kekurangan dari metode sensor yang memerlukan biaya yang banyak untuk instalasi dan *maintenance* sensor. Metode CNN dapat mendeteksi berbagai hunian parkir secara bersamaan tanpa bergantung pada jarak kamera, dan CNN mampu mengatasi batasan pada metode HOG untuk mempelajari objek pada gambar secara optimal. Dengan mengacu pada penelitian sebelumnya. Model arsitektur yang digunakan adalah *VGG16* dimana akurasi tertinggi dimiliki oleh model arsitektur *VGG16*.

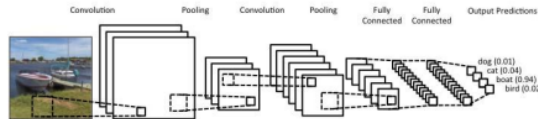
Tujuan dari penelitian ini adalah mengembangkan sistem deteksi hunian parkir yang dapat dipantau langsung oleh banyak orang. Untuk memantau hasil deteksi hunian parkir dibutuhkan suatu perangkat yang dapat menampilkan informasi mengenai kondisi suatu area parkir. Android merupakan salah satu sistem operasi yang dirancang oleh Google. Android biasa digunakan pada perangkat dengan layar sentuh seperti *smartphone* dan tablet. Pada tahun 2017, *Market share Android* di seluruh dunia menyentuh angka 73%, dengan *iOS* sebagai pesaing terdekatnya 19,99%. Di Indonesia sendiri 83,99% *market share* dikuasai oleh *android* pada bulan Juli 2017 [12]. Dari 265 juta total penduduk di Indonesia, 130 jutanya merupakan pengguna aktif media sosial. 177 juta pengguna *handphone* di Indonesia dengan 120 juta aktif di media sosial [13]. Penggunaan android yang sangat masif dalam kehidupan sehari – hari dapat dimanfaatkan untuk mengembangkan sistem deteksi hunian parkir pada penelitian ini.

II. LANDASAN TEORI

8

A. Convolution Neural Network (CNN)

Convolutional Neural network merupakan salah satu algoritma *deep learning* perkembangan dari *artificial neural network* dimana biasanya digunakan pada data gambar, teks, potongan suara, dan sebagainya. CNN memiliki dua metode dalam proses mengenali sebuah gambar yaitu *feedforward* sebagai tahap untuk klasifikasi/prediksi dan *backpropagation* sebagai tahap pembelajaran. CNN terdiri dari beberapa lapisan, dengan *convolutional layer*, *pooling layer* dan *fully connected* sebagai lapisan utama seperti pada Gambar 1. Setiap lapisan terbuat dari node berdasarkan data *input* dan menghasilkan *output* deteksi/klasifikasi.



Gambar 1. Arsitektur CNN sederhana [14]

B. Android

Android merupakan sebuah *operating system* yang dikembangkan oleh Google berbasis *linux*. Android memiliki *operating system* yang *open-source* sehingga setiap orang dapat mengunggah dan mendownload *source code android* dan menggunakannya untuk hardware yang dimiliki masing-masing.

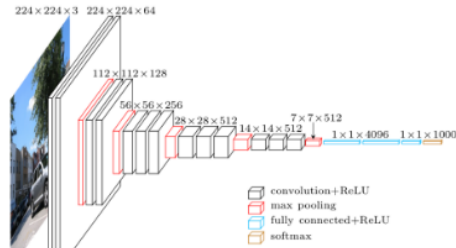
Pada penelitian ini, untuk memantau hasil deteksi hunian parkir *operating system* yang akan digunakan adalah android. *Android studio* merupakan sebuah *software* untuk membuat aplikasi android dengan *IDE Java*. Salah satu keunggulan menggunakan *android studio* yaitu memberikan akses ke *Android Software Development Kit (SDK)*. SDK merupakan sebuah ekstensi dari kode java yang memperbolehkannya untuk berjalan mulus di *device android*.

C. Firebase

Firebase merupakan sebuah *platform database* dari Google yang sekarang ini banyak digunakan bagi para *developer*. Firebase memiliki banyak *SDK* yang memungkinkan untuk mengintegrasikan layanan ini dengan *android*, *iOS*, *javascript*, *C++*, *Python*, hingga *unity*. Salah satu keunggulan dari firebase adalah *real-time database*, sebuah *cloud-hosted database* yang dapat menyimpan dan melakukan sinkronisasi data secara *real-time* untuk setiap *client* yang terhubung.

D. Model arsitektur VGG16

VGG16 merupakan salah satu model arsitektur CNN yang digunakan untuk memenangkan kompetisi ILSVR pada tahun 2014. Model arsitektur CNN merupakan suatu susunan lapisan utama pada CNN hingga membentuk sebuah jaringan yang dapat menghasilkan sebuah *output*. VGG16 dianggap sebagai salah satu model arsitektur yang sangat baik hingga saat ini. Model Arsitektur ini memiliki banyak *hyper-parameter* dengan 3×3 *convolution layer* dengan *stride* 1 dan selalu menggunakan *padding* dan *maxpool layer* yang sama yaitu 2×2 dengan *stride* 2. Pada tahap akhir VGG16 memiliki 2 *fully connected layer* yang diikuti *softmax* untuk outputnya. Model arsitektur ini memiliki jaringan yang cukup besar dan memiliki sekitar 138 juta parameter.

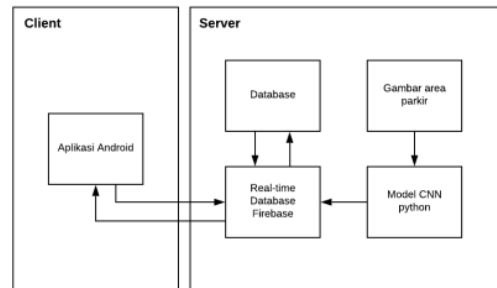


Gambar 2. Model arsitektur VGG16 [15]

III. METODOLOGI

A. Gambaran Sistem

Pada penelitian ini sistem, sistem dibagi menjadi 2 bagian yaitu *server* dan *client* seperti pada Gambar 3. Server merupakan tempat untuk proses deteksi dan menyimpan hasil deteksi pada database. Pada bagian *server* terdapat gambar seluruh area parkir yang digunakan sebagai *input* model CNN yang telah dilatih sebelumnya. Hasil dideteksi akan disimpan di database melalui *real-time database* yang dimiliki firebase. Firebase yang terhubung dengan *client* (dalam hal ini aplikasi android) dapat mengakses database secara langsung. Ketika *user* membuka aplikasi android maka sistem akan mengambil data hasil deteksi pada firebase dan menampilkan hasil deteksi tersebut pada aplikasi android.



Gambar 3. Sistem Overview

B. Dataset

Data yang digunakan bersumber dari *cnrpark.it* dimana data tersebut memuat beberapa area parkir. Pada penelitian ini, terdapat dua data yang diambil yaitu gambar yang memuat seluruh area parkir berukuran 1000×750 *pixel* (Gambar 4) beserta letak lokasi hunian parkirnya (Tabel 1) dengan format file csv dan gambar per hunian parkir (**Error! Reference source not found.**) beserta labelnya (Tabel 2) dengan format file csv. Data *CNRPark* sudah pernah digunakan oleh para peneliti sebelumnya [7][9].



Gambar 4. Contoh data gambar area parkir

Slot id	X	Y	H	W
1	2088	1570	380	373
2	1204	1560	380	380
3	264	1556	380	380
4	2050	1032	340	340
5	1723	1029	330	330

Tabel 1. Contoh data yang memuat lokasi hunian parkir



Gambar 5. Contoh data potongan gambar yang digunakan untuk training, validation, dan test

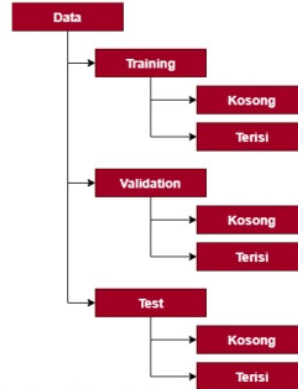
No	Path file	Label
1	camera8/R_2016-02-12_C08_328.jpg	0
2	camera8/R_2015-12-22_C08_288.jpg	1
3	camera8/S_2016-01-16_C08_244.jpg	1
4	camera8/R_2015-12-22_C08_289.jpg	0

Tabel 2. Contoh label data

C. Perancangan

- Pre-processing data

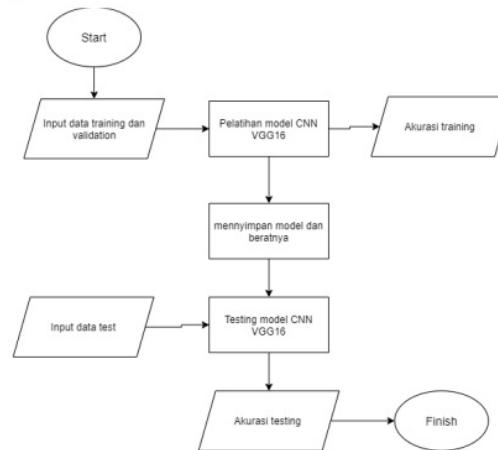
Data gambar per hunian parkir (Gambar 5) yang didapatkan dari CNRpark yang sudah terlabeli (Tabel 2) dengan format file CSV berjumlah 24.011 data. Pada tahap *pre-processing* data akan dibagi menjadi data *training set*, data *validation set*, dan data *test set* dimana hasil pembagian data tersebut sudah dikelompokkan sesuai status huniannya menggunakan Bahasa pemrograman Python seperti pada Gambar 6. Pembagian data ini bertujuan untuk mempermudah pada proses *training* dan *testing* model.



Gambar 6. Contoh pengelompokan dataset

- Perancangan model CNN

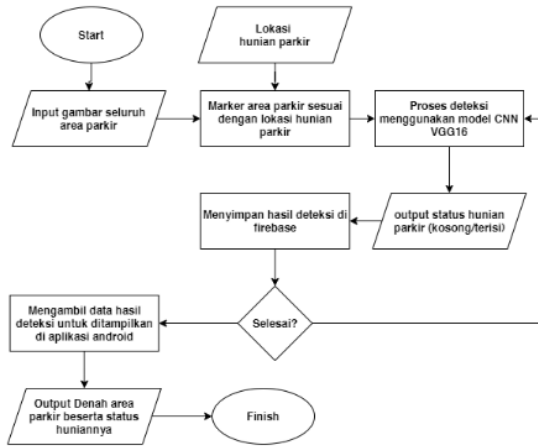
Setelah data dikelompokkan menjadi data *training set*, *validation set*, dan *test set* maka selanjutnya data dapat digunakan untuk proses pelatihan model. Berikut tahapan untuk pelatihan dan *testing* model CNN seperti pada Gambar 7.



Gambar 7. Flowchart training dan testing model CNN VGG16

- Perancangan sistem deteksi hunian parkir

Setelah model selesai melakukan proses pelatihan model dan mendapatkan bobot yang mampu untuk mendeteksi status hunian parkir. Maka selanjutnya menampilkan denah area parkir beserta status huniannya. Tahapan untuk sistem deteksi hunian parkir dapat dilihat pada Gambar 8.



Gambar 8. Flowchart sistem deteksi hunian parkir

D. Evaluasi

Pada penelitian ini, terdapat dua tahap evaluasi yaitu evaluasi model arsitektur VGG16 dan evaluasi sistem deteksi hunian parkir.

- Evaluasi model arsitektur VGG16
Tahap evaluasi pada model arsitektur VGG16 untuk mendeteksi status hunian parkir yaitu dengan melihat nilai akurasi dan *loss* dari hasil proses *training* dan *testing* model menggunakan *training set*, *validation set*, dan *test set*.
- Evaluasi sistem deteksi hunian parkir
Pada tahap evaluasi sistem deteksi hunian parkir akan menggunakan metode *black box testing*. Evaluasi sistem ini bertujuan untuk mengetahui apakah hasil deteksi pada suatu area parkir dapat divisualisasikan dengan baik pada aplikasi android.

IV. HASIL DAN PEMBAHASAN

A. Pre-processing data

Sebelum dilakukan *training* model, akan dilakukan tahap *pre-processing data*. Data gambar yang didapatkan dari CNRPark sudah terlabeli dengan format file CSV. Data akan dibagi menjadi data *training set*, data *validation set*, dan data *test set* dimana setiap kelompok data tersebut sudah dikelompokkan sesuai status huniannya seperti yang terlihat pada Gambar 6. *Training set* berfungsi sebagai partisi data yang digunakan untuk memperoleh bobot dari CNN, untuk menentukan bobot dari model CNN yang diperoleh dari *training set* sudah cukup, maka dilakukan proses verifikasi menggunakan *validation set*. Sedangkan *test set* berfungsi untuk memprediksi output berdasarkan model yang telah dibangun pada *training* dan *validasi*. Dengan memanfaatkan label data dari CNRPark serta menggunakan python dapat dilakukan pengelompokan data secara otomatis. Bentuk data gambar yang dikelompokkan merupakan gambar area parkir yang sudah dipotong berdasarkan tempat huniannya seperti pada Gambar 5.

B. Membangun model arsitektur VGG16

Setelah data dikelompokkan, selanjutnya melakukan pelatihan pada model CNN menggunakan *training set* dan *validation set*. Model arsitektur yang digunakan yaitu VGG16

diimplementasikan menggunakan Bahasa pemrograman Python dengan library Tensorflow. Model dibangun secara berurutan atau *sequential* seperti pada Gambar 9.

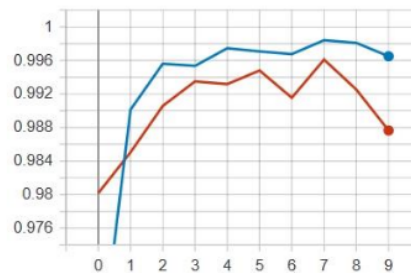
Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 224, 224, 64)	1792
conv2d_1 (Conv2D)	(None, 224, 224, 64)	36928
max_pooling2d (MaxPooling2D)	(None, 112, 112, 64)	0
conv2d_2 (Conv2D)	(None, 112, 112, 128)	73856
conv2d_3 (Conv2D)	(None, 112, 112, 128)	147584
max_pooling2d_1 (MaxPooling2D)	(None, 56, 56, 128)	0
conv2d_4 (Conv2D)	(None, 56, 56, 256)	295168
conv2d_5 (Conv2D)	(None, 56, 56, 256)	590080
conv2d_6 (Conv2D)	(None, 56, 56, 256)	590080
max_pooling2d_2 (MaxPooling2D)	(None, 28, 28, 256)	0
conv2d_7 (Conv2D)	(None, 28, 28, 512)	1180160
conv2d_8 (Conv2D)	(None, 28, 28, 512)	2359808
conv2d_9 (Conv2D)	(None, 28, 28, 512)	2359808
max_pooling2d_3 (MaxPooling2D)	(None, 14, 14, 512)	0
conv2d_10 (Conv2D)	(None, 14, 14, 512)	2359808
conv2d_11 (Conv2D)	(None, 14, 14, 512)	2359808
conv2d_12 (Conv2D)	(None, 14, 14, 512)	2359808
max_pooling2d_4 (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 4096)	102764544
dense_1 (Dense)	(None, 4096)	16781312
dense_2 (Dense)	(None, 2)	8194

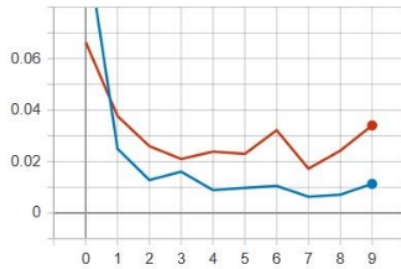
Total params: 134,268,738
Trainable params: 134,268,738
Non-trainable params: 0

Gambar 9. Summary model VGG16

Dalam penelitian ini, proses *training* model menggunakan *learning rate* 0.00001 dan 10 epoch. Untuk mencatat hasil *accuracy* dan *loss* digunakan tensorboard untuk memvisualisasikannya. Hasil *training model* dapat dilihat pada Gambar 10 dimana akurasi pada *training set* mencapai 0.9965 dan akurasi pada *validation set* mencapai 0.9877. Model hasil pelatihan disimpan untuk diuji kembali menggunakan *test set*.



Gambar 10. Grafik akurasi dan epoch pada training model



Gambar 11. Grafik loss dan epoch pada training model

Hasil test model menggunakan test set memperoleh akurasi sebesar 0,9901 dan *loss* sebesar 0,0461.

C. Database

Untuk menyimpan hasil deteksi pada area parkir diperlukan database yang terintegrasi dengan Bahasa pemrograman Python dan Java. Langkah awal yaitu membuat *project* pada firebase, *feature* pada firebase yang digunakan pada penelitian ini adalah *realtime database*. Struktur data pada database hanya memuat atribut “area-beserta nomor tempat parkir” untuk menyimpan hasil deteksi berupa status hunian parkir yaitu kosong (1) atau penuh (0). Firebase tersebut diintegrasikan dengan Python dengan menuliskan file config firebase pada project Python.

D. Aplikasi android

Pembuatan aplikasi android ini menggunakan *software* Android studio dengan Bahasa pemrograman Java. Project android yang telah dibuat diintegrasikan dengan firebase sehingga aplikasi android tersebut dapat mengambil data hasil deteksi hunian parkir. Pada project android ini menggunakan konsep MVC dimana *Model* berfungsi untuk mengatur data hasil deteksi hunian parkir, *View* untuk menampilkan *layout* denah area parkir beserta status huniannya, dan *controller* untuk mengatur jalanya data hasil deteksi untuk ditampilkan pada *view*. *Layout* denah area parkir pada aplikasi android dapat dilihat pada Gambar 12.

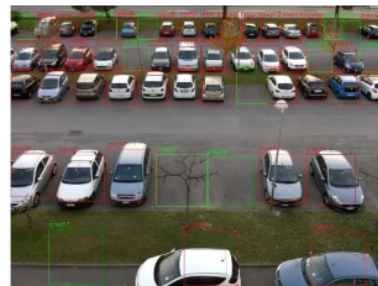


Gambar 12. Layout denah area parkir pada aplikas android

E. Implementasi sistem deteksi hunian parkir

Sistem deteksi parkir ini diimplementasikan sesuai *flowchart* pada Gambar 8 menggunakan Bahasa pemrograman python. Data gambar area parkir yang

digunakan seperti pada Gambar 4 dimana lokasi per-hunian parkirnya sudah ditentukan seperti pada Tabel 1. Lokasi per-hunian parkir tersebut memuat nilai X, Y, H, dan W. X dan Y sebagai titik koordinat, H dan W sebagai panjang dan lebar tempat parkir. Nilai – nilai tersebut digunakan untuk membuat *marker*. Gambar seluruh area parkir dipotong sesuai dengan *marker* yang telah dibuat, dimana potongan gambar tersebut digunakan sebagai input pada model CNN untuk ditentukan status huniannya. Setelah mendapatkan status hunian parkir, lalu membuat *bounding box* dengan warna berdasarkan status hunian parkir (jika kosong akan berwarna hijau dan jika terisi akan berwarna merah) dengan menggunakan *marker* yang telah dibuat seperti pada Gambar 13. Proses ini dilakukan sebanyak jumlah lokasi hunian parkir. Status hasil deteksi tersebut disimpan pada database. Pada aplikasi android menampilkan denah area parkir dimana setiap lokasi hunian parkir memiliki warna sesuai status huniannya (Gambar 14). Gambar 14). Gambar 14).



Gambar 13. Gambar area parkir hasil deteksi



Gambar 14. Tampilan aplikasi android setelah mendeteksi gambar area parkir

F. Evaluasi

- Evaluasi model arsitektur VGG16
Setelah melakukan proses *training* dan *testing* model didapatkan hasil akurasi yang terbilang baik yaitu pada *training set* mencapai nilai 0,9965 dengan nilai *loss* 0,0113 dan akurasi pada *validation set* mencapai 0,9877 dengan nilai *loss* 0,03406. Sedangkan pada *test set* akurasi mencapai nilai 0,9901 dengan *loss* sebesar 0,0461. Walaupun dengan akurasi yang sangat tinggi,

terdapat hasil deteksi yang tidak sesuai dengan keadaan sebenarnya seperti pada Gambar 15. dimana kondisi cahaya sangat minim seperti ketika malam hari dan hanya ada sedikit penerangan lampu.



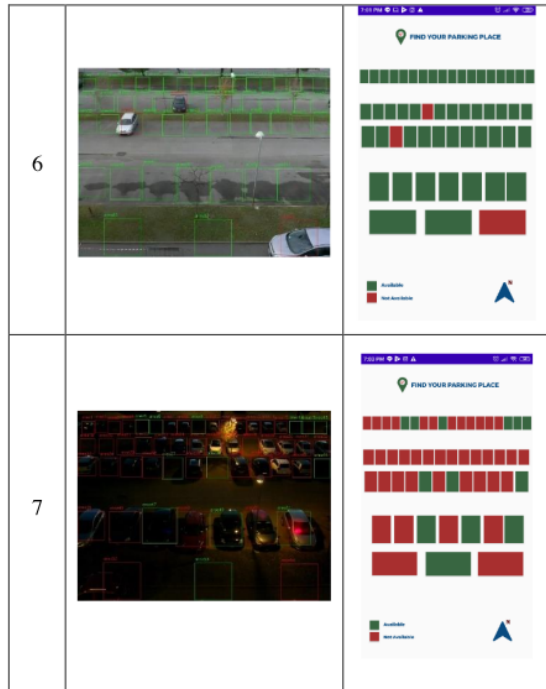
Gambar 15. Hasil deteksi gambar area parkir ketika malam hari

- Evaluasi sistem deteksi hunian parkir. Evaluasi sistem menggunakan *black box testing* yaitu dengan membandingkan gambar area parkir hasil deteksi dengan denah area parkir pada aplikasi android seperti pada Tabel 3. Berdasarkan evaluasi sistem tersebut menunjukkan bahwa setiap hasil deteksi pada gambar area parkir dapat digambarkan dengan baik oleh denah area parkir pada aplikasi android secara *realtime*. Jika dimasukkan suatu gambar area parkir baru untuk dideteksi, maka hasil deteksi pada gambar area parkir tersebut dapat dilihat secara langsung pada aplikasi android.

Tabel 3. Evaluasi sistem deteksi hunian parkir

No	Gambar hasil deteksi	Denah area parkir pada aplikasi andorid
1		

2		
3		
4		
5		

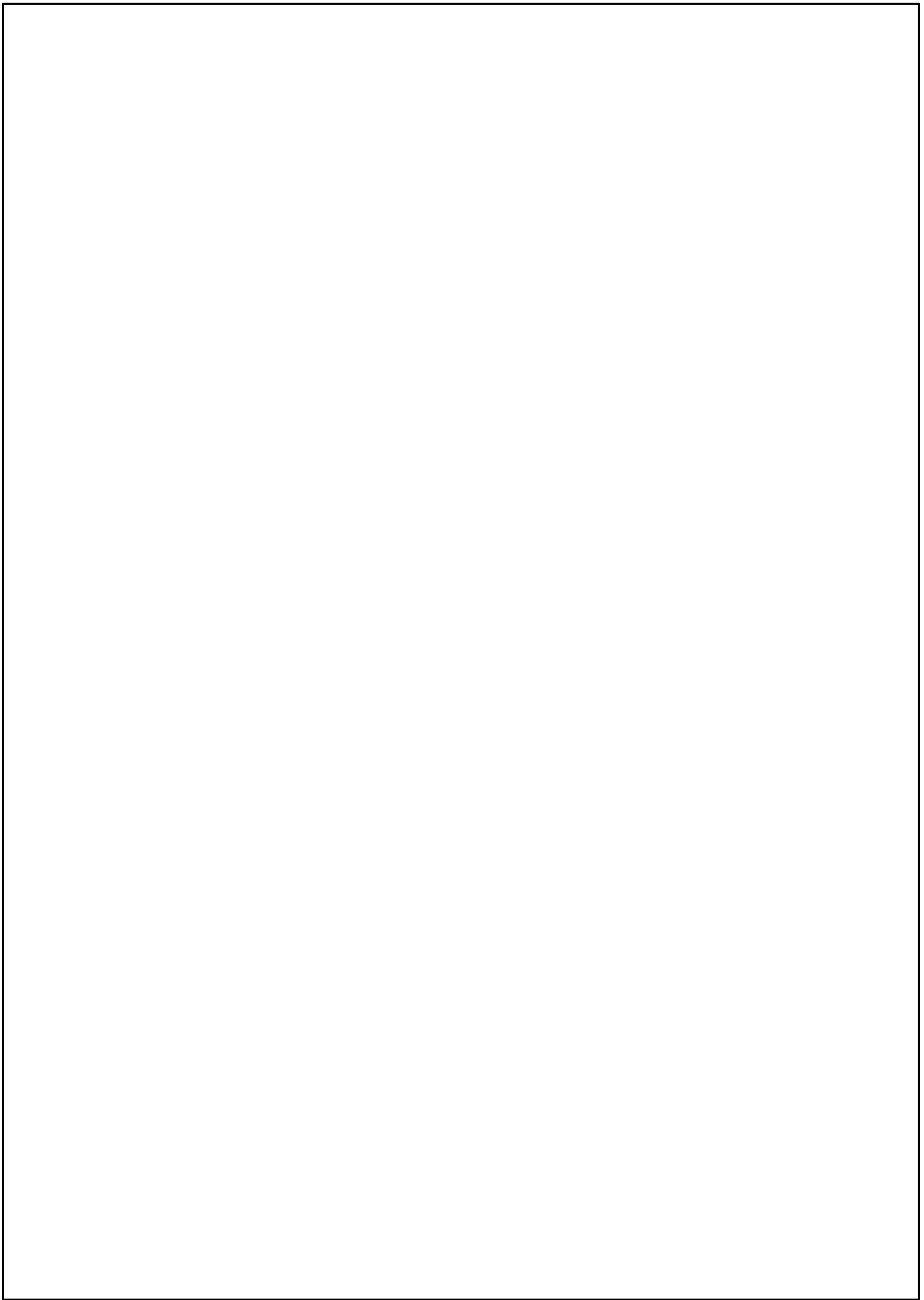


V. KESIMPULAN

Dalam penelitian pengembangan sistem deteksi hunian parkir ini, penerapan menggunakan metode *convolutional neural network* dengan model arsitektur VGG16 dapat berjalan dengan baik. Hasil akurasi dan *loss* yang didapatkan terbilang sangat baik, dimana akurasi pada *training set* mencapai nilai 0.9965 dengan nilai *loss* 0,0113 dan akurasi pada *validation set* mencapai 0,9877 dengan nilai *loss* 0,03406. Sedangkan pada *test set* akurasi mencapai nilai 0,9901 dengan *loss* sebesar 0,0461. Pada kondisi cahaya yang minim seperti ketika malam hari terdapat kesalahan deteksi status hunian parkir, walaupun dengan akurasi yang didapatkan hampir mendekati sempurna. Sistem deteksi gambar area parkir yang ditampilkan pada aplikasi android dapat divisualisasikan dengan baik secara *realtime*. Ketika sistem diberikan input gambar area parkir baru, hasil deteksi tersebut disimpan pada database dan pada aplikasi android dapat mengambil nilai hasil deteksi secara langsung lalu menampilkannya melalui denah area parkir pada aplikasi android.

REFERENSI

- [1] B. P. Statistik, "Badan Pusat Statistik," 2018. <https://www.bps.go.id/indicator/17/57/1/jumlah-kendaraan-bermotor.html> (accessed Nov. 15, 2020).
- [2] M. Akbar, S. Jura, N. Umar, and P. S. Komputer, "Sistem Informasi Realtime Web Untuk Slot Parkir Berbasis Embedded System," pp. 399–404, 2017.
- [3] J. C. Setiawan, R. Lim, M. Eng, and J. Andjarwirawan, "Implementasi Internet of Things Untuk Parkir Mobil Dengan Pembayaran Menggunakan QR Code," 2017.
- [4] D. Zulkarnain and E. S. Julian, "Perancangan Sistem Parkir Dengan Rekomendasi Lokasi Parkir," *JETri J. Ilm. Tek. Elektro*, vol. 14, no. 2, pp. 17–28, 2017. [Online]. Available: <http://trijurnal.lemlit.trisakti.ac.id/index.php/jetri/article/view/1611>.
- [5] S. A. M. A. K and S. Amini, "Sistem Monitoring Tempat Parkir dengan Sensor Ultrasonik Berbasis Arduino Uno pada Cibinong City Mall," *Seniati*, pp. 350–355, 2016.
- [6] E. Tanuwijaya and C. Faticah, "Penandaan Otomatis Tempat Parkir Menggunakan YOLO Untuk Mendeteksi Ketersediaan Tempat Parkir Mobil Pada Video CCTV," *Briliant J. Ris. dan Konseptual*, vol. 5, no. 1, p. 189, 2020, doi: 10.28926/briliant.v5i1.434.
- [7] G. Amato, F. Carrara, F. Falchi, C. Gennaro, and C. Vairo, "Car parking occupancy detection using smart camera networks and Deep Learning," *Proc. - IEEE Symp. Comput. Commun.*, vol. 2016-Augus, no. D1, pp. 1212–1217, 2016, doi: 10.1109/ISCC.2016.7543901.
- [8] R. Nugraha, A. N. Jati, and U. A. Ahmad, "Implementasi Histogram of Oriented Gradient (HOG) pada Embedded System untuk Identifikasi Slot Parkir sebagai Pendukung Smart Parking System," *e-Proceeding Eng.*, vol. 3, no. Universitas Telkom, pp. 771–777, 2016.
- [9] E. Tanuwijaya and C. Faticah, "Modification of Alexnet Architecture for Detection of Car Parking Availability in Video Cctv," *J. Ilmu Komput. dan Inf.*, vol. 13, no. 2, pp. 47–55, 2020, doi: 10.21609/jiki.v13i2.808.
- [10] L. Zhang, J. Huang, X. Li, and L. Xiong, "Vision-Based Parking-Slot Detection: A DCNN-Based Approach and a Large-Scale Benchmark Dataset," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5350–5364, 2018, doi: 10.1109/TIP.2018.2857407.
- [11] R. G. Paidi, V., Fleyeh, H., Nyberg, "Deep learning-based vehicle occupancy detection in an open parking lot using thermal camera," 2018.
- [12] M. Z. Zahid, "Aplikasi Berbasis Android untuk Pembelajaran : Potensi dan Metode Pengembangan," *Prism. Pros. Semin. Nas. Mat.*, vol. 1, pp. 910–918, 2018.
- [13] S. Samsiana *et al.*, "Optimasi Penggunaan Android Sebagai Peluang Usaha Di Masa Pandemi COVID '19," vol. 3, no. 2, pp. 137–148, 2020.
- [14] B. Hicham, A. Ahmed, and M. Mohammed, "Vehicle Type Classification Using Convolutional Neural Network," *Colloq. Inf. Sci. Technol. Cist*, vol. 2018-October, pp. 313–316, 2018, doi: 10.1109/CIST.2018.8596500.
- [15] D. Acharya, W. Yan, and K. Khoshelham, "Real-time image-based parking occupancy detection using deep learning," *CEUR Workshop Proc.*, vol. 2087, pp. 33–40, 2018.



Pengembangan Sistem Deteksi Hunian Parkir Menggunakan Metode Convolutional Neural Network

ORIGINALITY REPORT

10%

SIMILARITY INDEX

8%

INTERNET SOURCES

4%

PUBLICATIONS

0%

STUDENT PAPERS

PRIMARY SOURCES

1

idcloudhost.com

Internet Source

1%

2

blog.javan.co.id

Internet Source

1%

3

C. Attard, A. Naudi, S. Mallia, D. Gauci, R. Farrugia. "Assisting Motorists Using Parking Prediction through a Car App", 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), 2020

Publication

1%

4

Seta Samsiana, Rahmadya Trias Handayanto, Anita Setyowati Srie Gunarti, Irwan Raharja et al. "Optimasi Penggunaan Android Sebagai Peluang Usaha Di Masa Pandemi COVID'19", Jurnal Pengabdian kepada Masyarakat UBJ, 2020

Publication

1%

5

Wulandari, Nyayu Latifah Husni, Evelina,

Iskandar Lutfi, M. Nawawi, Ade Silvia, Dewi Permata Sari, Adella Rialita. "Safety Communicational System using Shifting Cryptography in Smart Parking", 2018 International Conference on Electrical Engineering and Computer Science (ICECOS), 2018

Publication

1%

6

r-libre.telug.ca

Internet Source

1%

7

www.scribd.com

Internet Source

1%

8

mc.ai

Internet Source

<1%

9

www.seminar.ilkom.unsri.ac.id

Internet Source

<1%

10

Riki Ruli A. Siregar, Hengki Sikumbang, Iriansyah BM. Sangadji, Indrianto. "KWh Meter Smart Card Model Token For Electrical Energy Monitoring", MATEC Web of Conferences, 2018

Publication

<1%

11

hdl.handle.net

Internet Source

<1%

12

www.coursehero.com

Internet Source

<1%

13

Oussama Sbai, Mohamed Elboukhari. "A simulation analysis of MANET's link-spoofing and replay attacks with ns-3", Proceedings of the 4th International Conference on Smart City Applications - SCA '19, 2019

Publication

<1%

14

ejournal.unitomo.ac.id

Internet Source

<1%

15

ipi.portalgaruda.org

Internet Source

<1%

16

eprints.uny.ac.id

Internet Source

<1%

17

Mohammad Asif Hossain, Rafidah Md. Noor, Kok-Lim Alvin Yau, Saaidal Razalli Azzuhri, Muhammad Reza Z'aba, Ismail Ahmedy. "Comprehensive Survey of Machine Learning Approaches in Cognitive Radio-Based Vehicular Ad Hoc Networks", IEEE Access, 2020

Publication

<1%

18

jurnal.unublitar.ac.id

Internet Source

<1%

19

id.scribd.com

Internet Source

<1%

20

repository.uinsu.ac.id

Internet Source

<1%

21 Zhang Yibo, Liu Qi, Hao Peifeng. "Vehicle Type Classification System for Expressway Based on Improved Convolutional Neural Network", 2020 3rd International Conference on Artificial Intelligence and Big Data (ICAIBD), 2020
Publication <1%

22 journal2.um.ac.id
Internet Source <1%

23 du.diva-portal.org
Internet Source <1%

Exclude quotes On

Exclude matches Off

Exclude bibliography On