

Klasifikasi Kendaraan pada Jalan Raya menggunakan Algoritma Convolutional Neural Network (CNN)

Radikto¹, Dadang Iskandar Mulyana², M Ainur Rofik³,
M Ohan Zoharuddin Zakaria⁴

STIKOM CKI Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika

Email: radikto.stikom@gmail.com¹, mahvin2012@gmail.com²

rofik.coboy@gmail.com³, ohanz15@gmail.com⁴

Abstrak

Kendaraan adalah alat angkut di jalan, termasuk kendaraan bermotor dan kendaraan tidak bermotor, sebagaimana dimaksud dalam angka 7 Pasal 1 Undang-Undang Nomor 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan. Melihat perkembangan kendaraan dimana masyarakat Indonesia mengandalkan kendaraan, tidak menutup kemungkinan jika kendaraan yang ada mulai memenuhi jalan di Indonesia, kebiasaan hidup masyarakat modern yang serba mengandalkan kendaraan untuk aktivitas sehari-hari. Oleh karena itu, peneliti membuat program pengenalan citra kendaraan menggunakan algoritma *Convolutional Neural Network*, yang merupakan kegiatan konvolusi dengan menggabungkan beberapa lapisan-lapisan persiapan, memanfaatkan beberapa komponen yang bergerak sama dan dimotivasi oleh sistem sensorik biologis. Gambar kendaraan yang digunakan adalah Sepedah, Sepeda Motor, Becak, Bajaj Mobil, Mobil Pickup, Mobil Molen, Bus, dan Truk. Implementasi pengenalan citra kendaraan dilakukan dengan menggunakan 2 model uji, model Sequential dan model VGG16 tingkat atas yang dijalankan dengan aplikasi Google Collaboratory, dan Keras. Data uji dalam penelitian ini adalah 1406 citra data latih dan 274 citra data uji, menghasilkan nilai evaluasi 98,18%, nilai loss 0,103 pada model Sequential, dan tingkat akurasi 99,64%. Implementasi pengenalan citra kendaraan dilakukan dengan menggunakan 2 model uji, model Sequential dan model VGG16 tingkat atas yang dijalankan dengan aplikasi Google Collaboratory, dan Keras. Data uji dalam penelitian ini adalah 1406 citra data latih dan 274 citra data uji, menghasilkan nilai evaluasi akurasi 98,18%, nilai loss 0,103 pada model Sequential, dan tingkat akurasi 99,64%, tingkat *loss* 0,014 pada model *on top* VGG16.

Kata kunci: *Image Classification*, Kendaraan, Keras, *Convolutional Neural Network* (CNN), VGG16

Abstract

Vehicle is a means of transportation on the road, including motorized vehicles and non-motorized vehicles, as referred to in number 7 Article 1 of Law Number 22 of 2009 concerning Road Traffic and Transportation. Seeing the development of vehicles where the Indonesian people rely on vehicles, it is possible that existing vehicles will start to fill the roads in Indonesia, the living habits of modern people who rely entirely on vehicles for daily activities. Therefore, the researcher created a vehicle image recognition program using the Convolutional Neural Network algorithm, which is a convolutional activity by combining several preparatory layers, utilizing several components that move together and are motivated by a biological sensory system. The images of the vehicles used are Bicycles, Motorcycles, Becaks, Bajaj Cars, Pickup Cars, Molen Cars, Buses, and Trucks. The implementation of vehicle image recognition is carried out using 2 test models, the Sequential model and the top-level VGG16 model that is run with the Google Collaboratory application, and Keras. The test data in this study were 1406 training data images and 274 test data images, resulting in an evaluation value of 98.18%, a loss value of 0.103 in the Sequential model, and an accuracy rate of 99.64%. The implementation of vehicle image

recognition was carried out using 2 test models, Sequential models and top-level VGG16 models running with Google Collaboratory applications, and Keras. The test data in this study were 1406 training data images and 274 test data images, resulting in an evaluation value of 98.18% accuracy, a loss value of 0.103 in the Sequential model, and an accuracy rate of 99.64%, a loss rate of 0.014 in the on top VGG16 model.

Keywords : *Image Classification, Vehicle, Keras, Convolutional Neural Network (CNN), VGG16*

PENDAHULUAN

Transportasi adalah pemindahan barang dan orang dari asal ke tujuan, atau transportasi adalah perpindahan manusia, hewan (kuda, sapi, kerbau) atau kendaraan yang digerakkan mesin dari satu tempat ke tempat lain. perjalanan antara asal dan tujuan. Dalam transportasi, terdapat unsur-unsur yang erat kaitannya dengan pengoperasian konsep transportasi itu sendiri. Unsur-unsur tersebut adalah sebagai berikut: orang yang membutuhkan; barang yang dibutuhkan; kendaraan sebagai alat/sarana; jalan dan dermaga sebagai prasarana transportasi; dan organisasi (pengelola transportasi). Beberapa jenis kendaraan telah muncul di seluruh dunia, salah satunya adalah kendaraan bermotor [5]. Beragamnya jenis kendaraan di Indonesia khususnya di pulau Jawa membuat peneliti tertarik untuk membuat program pengenalan jenis kendaraan berdasarkan dataset citra kendaraan. Melihat perkembangan teknologi masyarakat Mengandalkan perkembangan ilmu pengetahuan dan teknologi di Indonesia, kendaraan non-bermotor yang ada saat ini sangat mungkin untuk dilupakan. kendaraan tidak bermotor untuk dilupakan.

Penelitian yang dilakukan oleh [3] dengan judul "Rekognisi Wayang Kulit Menggunakan Jaringan Syaraf Tiruan", untuk mengklasifikasikan wayang Srikandi, Hanoman, Gatotkaca, dan Arjuna. Dengan data latih yaitu 100 citra, data testing 30 citra dan memiliki akurasi rekognisi 96% [3].

Penelitian yang dilakukan oleh [6] dengan judul "Metode Convolutional Neural Network Dalam Klasifikasi Citra Tiga Tokoh Wayang Pandawa", untuk klasifikasi Tiga wayang pandawa. Dengan data sebanyak 600 data yang terdiri dari 480 data latihan dan 120 data pengujian dengan menggunakan ukuran 100x100. Penelitian tersebut memiliki nilai akurasi 96,67% [6].

Penelitian yang dilakukan oleh [4] berjudul "Analisa Pengaruh Tingkat Akurasi Klasifikasi Citra Wayang Dengan Algoritma Convolutional Neural Network", menggunakan algoritma Convolutional Neural Network untuk menghasilkan klasifikasi empat jenis wayang punakawan yaitu Bagong, Petruk, Gareng dan Semar. Dengan data sebanyak 400 data dengan perbandingan dataset 80:20, ukuran citra 180x180 dan epoch 100. Penelitian tersebut menggunakan proses augmentation dan perubahan Citra ke *grayscale* pada *preprocessing* data latih yang memiliki akurasi 97%, *loss* 2%, presisi 93% dan *recall* 87% [4].

Penelitian yang dilakukan oleh [7] dengan judul "Klasifikasi Topeng Cirebon Menggunakan Metode *Convolutional Neural Network*", untuk menghasilkan klasifikasi lima topeng daerah cirebon yaitu Tumenggung, Kelana, Samba, Ruyang, dan Panji. Menggunakan data untuk pendekatan awal 150 dan pendekatan kedua 860 yang memiliki akurasi 99,48% [7].

Penelitian yang dilakukan oleh [8] dengan judul "Klasifikasi Batik Riau Dengan Menggunakan *Convolutional Neural Networks (Cnn)*", untuk mengklasifikasikan batik Riau. Total data yang digunakan sebanyak 160 gambar terbagi menjadi 68 gambar batik Riau dan 100 gambar bukan batik Riau yang menghasilkan nilai akurasi sebesar 65% [8].

Penelitian oleh [9] dengan judul “Klasifikasi Motif Batik Menggunakan *Convolutional Neural Network*”, untuk menghasilkan klasifikasi 3 kelas batik yaitu Mega-mendung, Parang, dan Kawung. Dengan data pelatihan sebanyak 90 dan data pengujian sebanyak 30 dengan menggunakan perubahan *channel* citra ke *grayscale* yang menghasilkan nilai akurasi 70% [9].

Berdasarkan persoalan tersebut dan penelitian sebelumnya yang terkait, maka lahirlah ide penggunaan jaringan syaraf tiruan untuk klasifikasi citra 9 jenis kendaraan antara lain sepeda, sepeda motor, Becak, Mobil Bajaj, Mobil Pickup, Mobil Molen, Bus dan Truk dengan algoritma CNN [10]. Melalui penelitian ini, peneliti berharap dapat memberikan kontribusi bagi perkembangan mobil di Indonesia khususnya di DKI Jakarta.

Klasifikasi

Klasifikasi adalah proses menemukan model yang dapat membagi data menurut kategorinya dengan membagi data menjadi dua tahap, tahap pelatihan (*learning*), tahap mencari tahu bagaimana kategori data diketahui dan pengujian (*testing*), evaluasi dari tahap pelatihan Tahap pemaparan hasil model, dengan data baru sebagai data uji, hasil dari tahap ini adalah tingkat akurasi/ketercapaian model dalam memprediksi data kategori yang tidak diketahui, terutama data uji [4], [6].

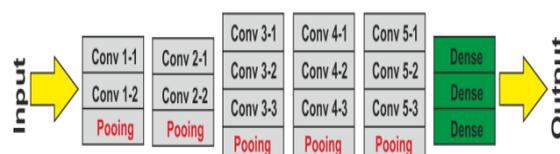
Deep Learning

JST (Jaringan Syaraf Tiruan) adalah suatu cabang algoritma *machine learning* [11]. *Machine learning* adalah suatu algoritma komputasi yang bekerja didasari pada data *history* untuk meningkatkan kualitas dalam membuat suatu prediksi [6]. *Deep Learning* adalah suatu teknik pada *machine learning* yang dapat mengolah informasi nonlinier dengan menggunakan banyak lapisan untuk menjalankan identifikasi pola, ekstraksi fitur dan klasifikasi yang merupakan metode studi representasi yang memungkinkan model perhitungan komputasi terdiri dari banyak *layer* pengolahan dengan mempelajari data dari banyak tingkat abstraksi [12], [13]. Berikut ini yang termasuk algoritma *Deep Learning* seperti *Stacked Autoencoders*, *Restricted Boltzmann Machine* (RBM), *Deep Belief Networks* (DBN), dan *Convolutional Networks* [14].

Convolutional Neural Network (CNN)

Convolutional Neural Networks adalah suatu jenis *neural network* pada *deep learning* yang dibuat menggunakan cara peringkat *Multi Layer Perceptron*, metode ini banyak dipakai pada aplikasi *computer vision* seperti klasifikasi citra, deteksian objek dan pengenalan wajah manusia [15], [16]. Dibandingkan ANN dan RNN, CNN merupakan *neural networks* yang digunakan dalam pengolahan data berbentuk citra yang dapat bekerja menggunakan kernel yang dimana kernel mengekstrak fitur pada data masukan melalui operasi konvolusi [17], [18].

Perbedaan antara CNN dengan *neural network*, *neuron* yang ada pada CNN memiliki bobot, bias, serta fungsi aktivasi. Dan terdapat *layer* penyusun pada CNN contohnya seperti *Fully Connected Layer*, *Pooling Layer*, *Activation ReLU Layer* serta *Convolution Layer* [19], [20]. VGG16 adalah model konvolusi yang dapat memanfaatkan pada lapisan konvolusional dengan menggunakan ukuran *convolutional filter* (3x3), dan mempunyai 16 (enam belas) *layer* yang terdiri atas 13 (tiga belas) *convolutional layer* dan 3 (tiga) *fully connected layer*[1].



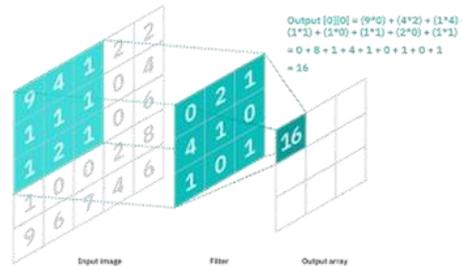
Gambar 1 Arsitektur VGG16

1. Convolutional Layer

Convolutional layer adalah proses manipulasi citra yang dicoba dengan eksternal mask ataupun subwindows buat menciptakan citra baru dengan merubah dimensi citra memakai pembedahan convolutional buat menciptakan fitur ekstraksi berarti dari citra semacam deteksi tepi/ karakteristik, warna, orientasi gradien, serta yang lain yang lewat proses encoding [4]. Persamaan matematika ini dari operasi konvolusi yang dapat dilihat pada persamaan dibawah ini [6]:

$$(t) = (x \times y)(t) \tag{1}$$

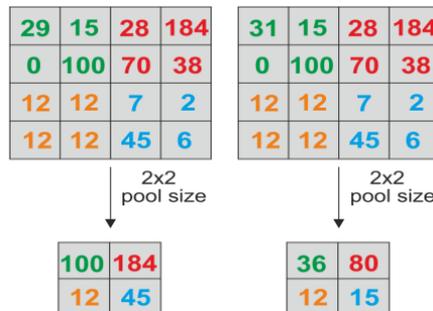
Dengan s(t) merupakan fungsi konvolusi, x = input, dan w = filter.



Gambar 2 Convolutional Layer

2. Pooling Layer

Pooling layer yang ada pada CNN memakai guna fitur map ataupun activation map selaku masukan yang diteruskan buat diolah dengan bermacam tipe pembedahan statistik bersumber pada nilai piksel terdekat. Keuntungan pooling layer ialah bisa secara bertahap kurangi dimensi volume output pada fitur map yang bisa bermanfaat buat mengatur overfitting [20]. Pada dasarnya terdapat 2 (dua) pooling yang digunakan yaitu average pooling dan max pooling [7].



Gambar 3 Operasi Max Pooling dan Average Pooling

3. Rectified Linear Unit (ReLU)

Rectified Linear Unit (ReLU) adalah fungsi aktivasi linier yang banyak digunakan di CNN. Tugas fungsi aktivasi adalah menentukan apakah neuron harus diaktifkan berdasarkan nilai bobot input [12]. Persamaan matematika ini dari fungsi aktivasi ReLU yang dapat dilihat pada persamaan dibawah ini [6].

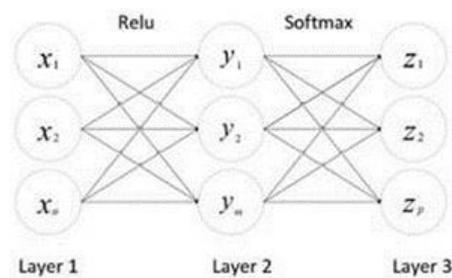
$$re(x) = \max(0, x) \tag{2}$$

relu(x) merupakan fungsi ReLU dari nilai x, x = input, dan max(0, x) merupakan fungsi nilai max dari 0 dan x.

4. Fully Connected Layer

Tiap neuron yang berada didalam susunan convolutional wajib dikonversi jadi suatu informasi berukuran tunggal, saat sebelum memasukkan kedalam proses fully connected layer. Kala proses tersebut dijalankan hingga hendak menjadikan data informasi spasialnya hendak lenyap serta tidak bisa dikembalikan, fully connected layer bisa

diimplementasikan cuma di akhir *neuron* [21].



Gambar 4 Fully Connected Layer

Pada gambar 4 tersebut, pada layer 1 akan melakukan proses feed forwarding dengan menuju layer 2 yang menggunakan aktivasi ReLU. Kemudian di layer 2 akan melakukan proses klasifikasi yang dilakukan menggunakan fungsi *softmax* [22]. Rumus persamaan *fully connected layer* dapat dituliskan dengan berikut [23]:

$$Z_r = \sum_{c=1}^j X_c W_{cr} + b_r \quad (3)$$

Pada rumus diatas r merupakan $1, 2, 3, \dots, R$, R merupakan jumlah *neuron* keluar, J merupakan jumlah *neuron* masuk, z_r merupakan *output neuron* ke- r , x_c merupakan *input neuron* ke- c , W_{cr} merupakan bobot antara *neuron input* ke- c dengan menuju *neuron output* ke- r , sedangkan b_r merupakan bias untuk *neuron output* ke- r .

5. Softmax Activation

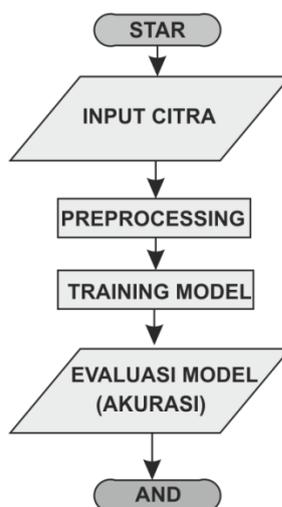
Softmax Activation adalah bentuk yang menyerupai dari *Logistic Regression*, berfungsi dalam proses klasifikasi lebih dari dua kelas. *Softmax* digunakan untuk merubah *output* dari lapisan terakhir menjadi distribusi probabilitas dasarnya. Berikut merupakan persamaan aktivasi *softmax* [20]:

$$f_{i(x)} = \frac{e^{x_i}}{\sum_{j=1}^k e^{x_j}} \quad (4)$$

Pada persamaan diatas terdapat notasi f_i yang merupakan hasil fungsi dari setiap bagian ke- i pada keluaran kelas vektor. Sedangkan x merupakan vektor yang memiliki nilai hasil *fully connected layer* yang terakhir. *Softmax* dapat menghitung probabilitas dari semua kelas kemudian akan diambil vektor yang bernilai riil dan mengubahnya kedalam nilai pada rentang 0 (nol) sampai dengan 1 (satu) yang dapat dijumlah semuanya akan bernilai satu [20].

METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini menggunakan Convolutional Neural Network (CNN) serta framework Keras dengan Google Collaboratory dan back-end Tensor Flow. Gambaran umum yang dilakukan dalam proses penelitian ini dijelaskan pada Gambar 5 berikut :



Gambar 5 Kerangka Uji

Dari gambar 5 diatas tahap awal yang dilakukan dengan memasukan citra berupa dataset yang sudah dipersiapkan sebelumnya, dataset ini terbagi jadi dua yaitu data latih dan test. Sebagian besar data gambar dalam penelitian ini diunduh dari web, yang tentu saja memiliki berbagai ukuran. Sejalan dengan itu, preprocessing gambar dilakukan untuk mendapatkan gambar yang siap pakai kemudian ditangani lebih lanjut lagi, untuk kebutuhan ekstraksi data ataupun kebutuhan klasifikasi data. Proses ini dilakukan menggunakan beberapa langkah preprocessing data yang dapat dilakukan dalam klasifikasi menggunakan metode CNN.

Preprocessing adalah tahap yang dilakukan sebelum training atau testing model dengan melakukan proses resize, konversi citra RGB dan feature extraction VGG16. Resize yaitu mengubah ukuran citra yang digunakan untuk menyesuaikan citra agar dapat melakukan pelatihan . Konversi citra RGB adalah proses merubah gambar berwarna menjadi grayscale. Feature extraction adalah suatu teknik pengenalan pada objek yang melihat ciri khusus pada objek tersebut yang bertujuan melakukan perhitungan dan perbandingan untuk mengklasifikasi suatu citra.

Proses klasifikasi kendaraan dicoba memakai dua model yang tidak sama. Pengujian awal dilakukan dengan menggunakan model sekuensial, yaitu model pengujian simple yang dibuat dengan menentukan parameter secara acak yang berguna untuk mendapatkan nilai akurasi yang maksimal. Pengujian kedua dilakukan pada model VGG16 atas, dan ekstraksi fitur VGG16 dilakukan sebelum pengujian. Model ini sering digunakan pada penelitian sebelumnya karena akurasinya yang baik. Dengan mengevaluasi setiap model pengujian sebelumnya, diperoleh nilai akurasi dan loss yang diperoleh dari masing-masing model pengujian. Dari sini kita dapat melihat kemungkinan bahwa objek gambar yang dibaca selama proses klasifikasi akan gagal. Selama proses evaluasi, nilai akurasi akhir juga diperoleh dari keseluruhan model pengujian.

a. Dataset

Dataset uji yang digunakan adalah 1406 citra pada data latih kendaraan dan 274 citra pada data uji, masing-masing kendaraan diinterpretasikan menggunakan metode scraping dari berbagai sumber di internet yaitu Google Images, dengan mendownload sama seperti yang ada pada penelitian ini. gambar-gambar. Berikut 9 kendaraan yang dipakai.

Tabel 1. Variabel Penelitian

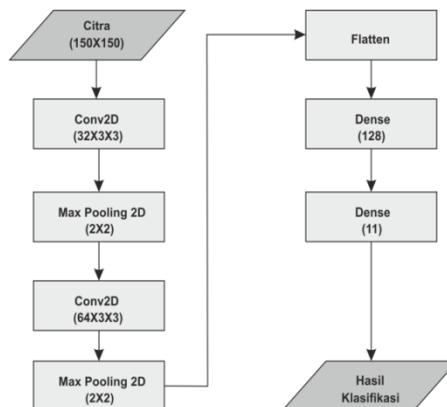
Variabel	Latih	Uji	Definisi Variabel
Bajaj	114	30	Citra berupa kendaraan bajaj
Becak	108	30	Citra berupa kendaraan becak
Bus	160	30	Citra berupa kendaraan bus
Mobil	160	30	Citra berupa kendaraan mobil
Mobil Pickup	160	31	Citra berupa kendaraan mobil pickup
Mobil Molen	160	30	Citra berupa kendaraan mobil molen
Sepeda	192	32	Citra berupa kendaraan sepeda
Sepeda Motor	192	32	Citra berupa kendaraan sepeda motor
Truk	160	30	Citra berupa kendaraan truk



Gambar 6 Dataset

Perancangan Nilai Uji

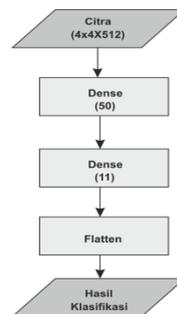
Pada riset ini menggunakan dua model arsitektur CNN berikut.



Gambar 7 Model *Sequential*

Gambar 7 di atas merupakan model aliran uji *sequensial*. Citra masukan berukuran 150x150x3 piksel dan diubah menjadi *grayscale*, kemudian dikonvolusi menggunakan dua konpolasi *layer* dengan 32 dan 64 filter yang masing-masing berukuran 3x3. Di lapisan pooling, operasi pooling maks digunakan dengan ukuran pool maks 2x2, yang membagi setiap ukuran gambar dengan 2 saat melewatinya. Proses perataan digunakan untuk mengubah format gambar 2d menjadi format gambar 1d dengan nilai yang telah ditentukan sebesar 150x150x3 piksel. Kemudian kita menggunakan dua lapisan padat, lapisan pertama

digunakan sebagai aktivasi ReLu , berukuran 128 *neuron*, dan lapisan *Softmax* kedua memiliki 11 *neuron*, tergantung pada jumlah kelas data yang diambil dari *dataset*.



Gambar 8 Model on top VGG16

Pada Gambar 8 di atas adalah model kedua yang digunakan dalam proses klasifikasi, memanfaatkan salah satu ekstraksi fitur VGG16 dan pelatihan pada ImageNet. Setelah ekstraksi fitur dari VGG16, tahap pertama dari citra masukan adalah 4x4x512 piksel. Kemudian ratakan dari gambar yang sudah diinput. Kemudian terdapat dua lapisan padat, lapisan pertama berukuran 50 neuron sebagai aktivasi ReLu (rectified linear unit) dan lapisan *Softmax* kedua memiliki 11 neuron.

HASIL DAN PEMBAHASAN

Tahap pengujian penelitian ini dilakukan dengan melatih model yang dipasang pada data citra kendaraan menggunakan 20 *epoch*, ukuran *batch* = 128 dan validasi *split* = 0,2 yang berarti pelatihan 80% dan validasi 20%. *Epoch* adalah jumlah neuron yang dapat melihat semua data yang dikumpulkan, dan ukuran *batch* adalah jumlah sampel pelatihan dalam 1 (satu) *forward/backward pass*.

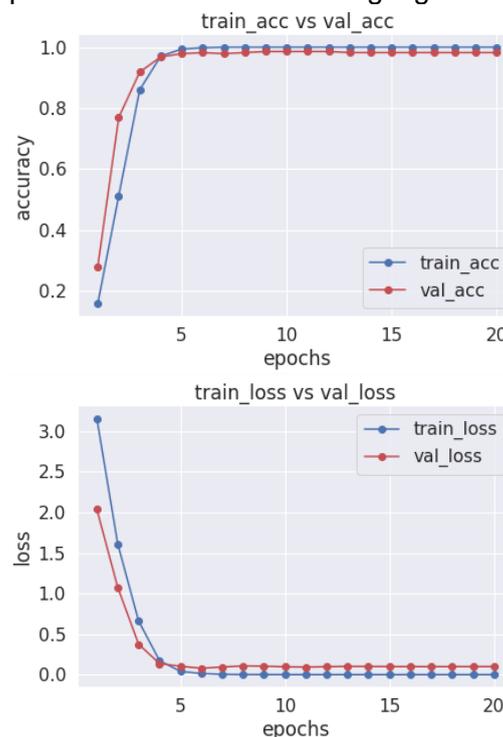
b. Model Sequential

Tabel 2. Hasil Model Sequential

Epoch	Data Latih		Data Validasi	
	Kegagalan	Akurasi	Kegagalan	Akurasi
1	3,1573	0,1584	2,0393	0,2801
2	1,6122	0,5125	1,0762	0,7695
3	0,6608	0,8603	0,3753	0,9184
4	0,1729	0,9706	0,1366	0,9681
5	0,0406	0,9938	0,1035	0,9787
6	0,0144	0,9982	0,0790	0,9823
7	0,0044	1,0000	0,0936	0,9787
8	0,0017	1,0000	0,1082	0,9823
9	0,0012	1,0000	0,1048	0,9858
10	7,4485	1,0000	0,0967	0,9858
11	5,2604	1,0000	0,0937	0,9858
12	4,1851	1,0000	0,0980	0,9858
13	3,4374	1,0000	0,1022	0,9823
14	3,0623	1,0000	0,1013	0,9823
15	2,7966	1,0000	0,1000	0,9823
16	2,5724	1,0000	0,0989	0,9823
17	2,3793	1,0000	0,0988	0,9823

18	2,2265	1,0000	0,0995	0,9823
19	2,0770	1,0000	0,0993	0,9823
20	1,8593	1,0000	0,1003	0,9823

Tabel 2 diatas merupakan hasil pelatihan menggunakan 20 *epoch* data training dan data validasi. Terlihat bahwa iterasi mendapatkan nilai akurasi dan loss dari data latih dan data validasi. Nilai akurasi adalah nilai yang digunakan sebagai acuan untuk menentukan tingkat keberhasilan/kelayakan model yang dibuat sebelumnya, sedangkan nilai kegagalan atau *loss* adalah nilai ukuran kegagalan/kesalahan yang disebabkan oleh jaringan yang bertujuan untuk meminimalkannya. Pada data latih nilai akurasi tertinggi yaitu 1,0000 pada *epoch* 7 sampai 20, sedangkan nilai *loss* terendah adalah 0,0012 pada *epoch* 9, dan data validasi mendapatkan akurasi tertinggi pada *epoch* 9, 10, 11 nilai 0.9858, dan 12, sedangkan kerugian terendah pada *epoch* 6 adalah 0,0790. Berdasarkan hasil data latih dan validasi pada data latih, dapat kita visualisasikan sebagai gambar/grafik berikut..



Gambar 9 Plot Accuracy vs Loss Model Sequential

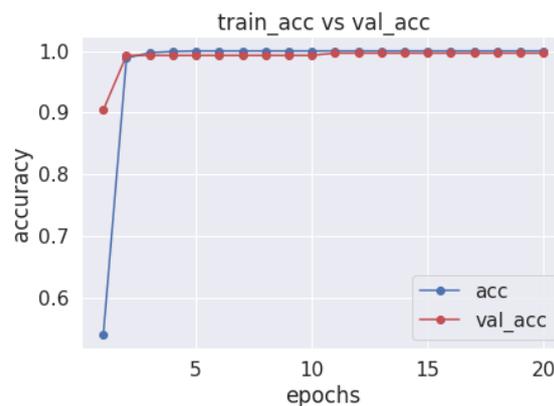
Pada Gambar 9 di atas diketahui bahwa hubungan antara nilai akurasi dan nilai *loss* pada data latih dan validasi berkaitan dengan jumlah *epoch*/iterasi. Korelasi/hubungan yang muncul pada nilai akurasi menunjukkan korelasi positif, yaitu memiliki hubungan satu arah, dengan ketentuan semakin tinggi jumlah *epoch* maka semakin tinggi akurasi data training dan validasi data. Maka sebaliknya

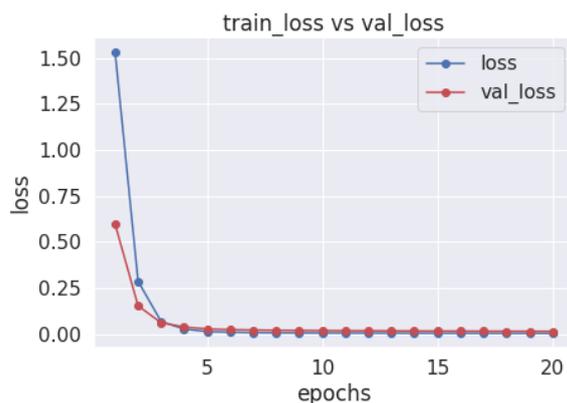
dibandingkan dengan nilai akurasi, jumlah *epoch* berkorelasi negatif dengan nilai *loss*, dan jumlah *epoch* yang digunakan akan mempengaruhi nilai *loss* yang dihasilkan oleh data latih. Dari hasil tersebut dapat disimpulkan bahwa mengurangi nilai kerugian yang diharapkan dapat dicapai dengan menambah atau menambah jumlah *epoch* selama pelatihan. Dari hasil pelatihan *sequence model* diatas menghasilkan nilai evaluasi kinerja yang dihasilkan dari data pengujian dengan nilai akurasi 98,48% dan nilai *loss* sebesar 0,077. Model on top VGG16.

Tabel 3. Hasil *Fit Model on top VGG16*

Epoch	Data Latih		Data Validasi	
	Kegagalan	Akurasi	Kegagalan	Akurasi
1	1.5344	0.5391	0.5974	0.9043
2	0.2857	0.9884	0.1517	0.9929
3	0.0668	0.9973	0.0600	0.9929
4	0.0257	0.9991	0.0367	0.9929
5	0.0121	1.0000	0.0265	0.9929
6	0.0083	1.0000	0.0222	0.9929
7	0.0062	1.0000	0.0203	0.9929
8	0.0051	1.0000	0.0187	0.9929
9	0.0044	1.0000	0.0180	0.9929
10	0.0039	1.0000	0.0171	0.9929
11	0.0035	1.0000	0.0164	0.9965
12	0.0032	1.0000	0.0160	0.9965
13	0.0029	1.0000	0.0154	0.9965
14	0.0027	1.0000	0.0148	0.9965
15	0.0025	1.0000	0.0146	0.9965
16	0.0023	1.0000	0.0142	0.9965
17	0.0021	1.0000	0.0138	0.9965
18	0.0020	1.0000	0.0136	0.9965
19	0.0019	1.0000	0.0134	0.9965
20	0.0017	1.0000	0.0131	0.9965

Dari Tabel 3 di atas, kita dapat melihat hasil pengujian model kedua, model VGG16 di atas, dengan total 20 *epoch*. Dari data tersebut pengujian data latih memiliki nilai akurasi tertinggi sebesar 1,0000 pada *epoch* 5-20 dan nilai *loss* terendah 0,0017 pada *epoch* 20, kemudian data validasi didapatkan nilai tertinggi pada *epoch* 11 sd 20 Nilai akurasi adalah 0.9965 untuk *epoch* ke-20, sedangkan nilai *loss* terendah untuk *epoch* ke-20 adalah 0,0131. Hasil data pengujian pada tabel di atas dapat divisualisasikan sebagai grafik/grafik berikut.





Gambar 10 Plot Accuracy vs Loss Model on top VGG16

Pada Gambar 10 di atas merupakan ikatan antara nilai *accuracy* serta nilai *loss* pada informasi *train* serta informasi validasi, dari foto tersebut banyaknya iterasi pada model kedua. Model kedua ini mempunyai korelasi nilai *accuracy* yang searah baik pada informasi pelatihan/ *train* maupun informasi validasi serta berbanding dengan nilai *loss* ialah korelasi negatif. Dari pelatihan *model on top VGG16* diperoleh nilai penilaian kinerja yang dihasilkan pada informasi test dengan nilai *accuracy* sebesar 99,70% serta nilai *loss* sebesar 0,021.

Evaluasi

Penilaian dari hasil pengujian model Sequential serta model on top VGG16 yang dilatih memakai informasi train sebanyak 1406 citra diperoleh nilai penilaian yang dihasilkan dari informasi test sebanyak 274 citra dengan pengujian sebanyak 20 kali, batch size=128, validation split=0.2 didapat nilai *accuracy* serta nilai *loss* selaku berikut.

Tabel 4. Evaluasi Nilai Accuracy dan Nilai Loss

Model Sequential		Model on top VGG16	
Accuracy	Loss	Accuracy	Loss
98,48%	0,077	99,70%	0,021

SIMPULAN

Bersumber pada riset serta hasil implementasi Convolutional Neural Network(CNN) dalam proses klasifikasi Kendaraan di jalan raya, mempunyai kesimpulan buat bisa melewati proses preprocessing yang baik dimensi citra bisa diganti dimensinya jadi 150x150x3 piksel serta merubah warna citranya jadi grayscale pada model Sequential, setelah itu memakai feature extraction VGG16 yang dilatih di ImageNet dengan memakai dimensi ukuran 4x4x512 piksel pada on top VGG16. Akurasi yang bisa dihasilkan hendak terus menjadi baik apabila digunakan informasi train yang terus menjadi banyak. Perihal ini meyakinkan dengan pengujian pada informasi training dengan jumlah sebanyak 80% serta jumlah informasi validasi sebanyak 20% pada tiap model pengujian. Dengan melaksanakan penilaian pengujian model Sequential serta model on top VGG16 pada epoch sebanyak 20 kali, batch size=128, serta validation split=0, 2(80% training serta 20% validasi) diperoleh nilai *accuracy* serta nilai *loss* model bersumber pada informasi test sebesar 98,48% nilai *accuracy* serta 0,077 nilai *loss* pada model Sequential dan 99,70% nilai *accuracy* serta 0,021 nilai *loss* pada model on top VGG16.

DAFTAR PUSTAKA

- D. Sulaksono and K. Saddhono, "Ecological Concept of Wayang Stories and the Relation with Natural Conservation in Javanese Society," *KnE Soc. Sci.*, vol. 3, no. 9, p. 58, 2018, doi: 10.18502/kss.v3i9.2611.
- M. Boy Sandy, Johannes K Siahaan², Prayogi Permana, "Klasifikasi Citra Wayang Dengan Menggunakan Metode k- NN & GLCM," vol. 2, no. November, pp. 71–77, 2019.
- A. Susanto and I. U. W. Mulyono, "Rekognisi Wayang Kulit Menggunakan Jaringan Syaraf Tiruan," *Pros. SENDI_U*, no. October, pp. 978–979, 2019.
- M. Resa, A. Yudianto, and H. Al Fatta, "Wayang Dengan Algoritma Convolutional Neural Network," *J. Teknol. Inf.*, no. 2, pp. 182–190, 2020.
- G. O. F. Parikesit, "3D Wayang Kulit: Traditional shadow puppetry meets modern display technology," *Int. J. Arts Technol.*, vol. 9, no. 2, pp. 162–172, 2016, doi: 10.1504/IJART.2016.077238.
- K. Wisnudhanti and F. Candra, "Metode Convolutional Neural Network Dalam Klasifikasi Citra Tiga Tokoh Wayang Pandawa," vol. 7, no. 2018, pp. 1–5, 2020.
- F. I. Kurniadi, V. K. Putri, and Y. E. Wibawa, "Klasifikasi Topeng Cirebon Menggunakan Metode Convolutional Neural Network," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 1, pp. 163–169, 2021.
- H. Fonda, "Klasifikasi Batik Riau Dengan Menggunakan Convolutional Neural Networks (Cnn)," *J. Ilmu Komput.*, vol. 9, no. 1, pp. 7–10, 2020, doi: 10.33060/jik/2020/vol9.iss1.144.
- R. Mawan, "Klasifikasi motif batik menggunakan Convolutional Neural Network," *Jnanaloka*, pp. 45–50,