

Perbandingan Tingkat Akurasi Arsitektur Convolutional Neural Network untuk Model Deteksi Penggunaan Masker Secara Otomatis

(Comparison of Convolutional Neural Network Architecture Accuracy Levels for Automatic Mask Use Detection Models)

Dino Hariatma Putra^{*}, Sulistyowati, Veronica Yudisthiana

Program Studi Informatika, Institut Teknologi Indonesia
Jl. Raya Puspipetek, Serpong, Kota Tangerang Selatan, Provinsi Banten 15320

Abstrak

Pada saat ini Indonesia sedang menghadapi pandemi COVID19, anjuran pemerintah guna untuk menekan penyebaran virus adalah dengan cara menggunakan masker. Pada penelitian sebelumnya, pendeteksian masker hanya ditujukan untuk mendeteksi menggunakan masker dan tidak menggunakan masker. Namun, pada kenyataannya saat penggunaan masker harus dengan cara yang benar, sehingga manfaat yang didapat dalam menggunakan masker bisa maksimal. Oleh karena itu, pada penelitian ini ditambahkan satu kelas baru sehingga menjadi 3 kelas yaitu menggunakan masker dengan benar, tidak menggunakan masker, dan menggunakan masker tidak benar. Tujuan dari penelitian ini adalah membandingkan model pretrained keras Convolutional Neural network (CNN), yaitu VGG16, Mobilenetv2 dan Inceptionv3 untuk mendapatkan hasil akurasi terbaik dan mengimplementasikannya untuk sistem deteksi masker secara otomatis. Selain itu, penelitian ini juga menjelaskan bagaimana karakteristik arsitektur ketiga model tersebut. Setelah dilakukan pengujian, diketahui bahwa hasil akurasi yang didapatkan dengan dataset yang sama namun model CNN berbeda mendapatkan hasil yang berbeda juga. Mobilenetv2 mendapatkan hasil akurasi tertinggi dengan persentase 100%.

Kata Kunci : *Convolutional Neural Network, Image Recognition, Keras, COVID19*

Abstract

Currently, Indonesia is facing the COVID-19 pandemic, one of the government's recommendation to suppress the spread of the virus is to use masks. In previous studies, mask detection was only intended to detect using masks and not using masks. However, in reality, when using a mask, it must be done in the right way, so that the benefits obtained in using a mask can be maximized. Therefore, in this study, one new class was added so that it became 3 classes, namely using masks correctly, not using masks, and using masks incorrectly. The purpose of this study is to compare the hard pretrained model Convolutional Neural network (CNN), namely VGG16, Mobilenetv2 and Inceptionv3 to get the best accuracy and carry out for an automatic mask detection system. In addition, this study also explains the architectural characteristics of the three models. After testing, it is known that the results obtained with the same dataset but different CNN models get different accuracy as well. Mobilenetv2 gets the highest accuracy results with a percentage of 100%.

Keyword : *Convolutional Neural Network, Image Recognition, Keras, COVID19*

*Penulis Korespondensi.

Alamat E-mail : dino.hp@iti.ac.id; sulistyowati.if@iti.ac.id; veronicayudhisthiana@gmail.com

1. Pendahuluan

Dewasa ini, *Deep learning* telah menjadi salah satu topik hangat dalam dunia *Machine Learning* karena kapabilitasnya yang signifikan dalam memodelkan berbagai data kompleks seperti citra dan suara. Metode *Deep Learning* yang saat ini memiliki hasil paling signifikan dalam pengenalan citra adalah *Convolutional Neural network* (CNN) [1].

Banyak penelitian yang telah dilakukan dengan mengimplementasikan CNN, salah satunya adalah deteksi penggunaan masker. Penggunaan masker pada saat ini merupakan kegiatan wajib saat bepergian keluar rumah. [2] Oleh karena itu, penggunaan masker bertujuan untuk mencegah penyebaran virus corona yang menular melalui hidung dan mulut.

Hasil penelitian ini menunjukkan data dengan menggunakan *epoch* 50 dan rasio dataset 90% dan data latih 10%, data uji mendapatkan nilai akurasi terbaik mencapai 96%. [3] Namun pada penelitian ini, tidak dijelaskan menggunakan arsitektur CNN yang mana.

Penelitian lainnya mengenai deteksi masker juga dilakukan Bunardi Budiman, Chairisni Lubis dan Novario Jaya Perdana. Pada penelitian ini dilakukan pembuatan sistem Pendeteksian Penggunaan Masker Wajah dengan Metode *Convolutional Neural network*. [4] Pada Penelitian ini, arsitektur CNN yang digunakan adalah *Mobilenetv2*. Total data set yang diuji adalah 1400 gambar yang dibagi menjadi 2 kelas, yaitu kelas menggunakan masker dan kelas tidak menggunakan masker. Dari hasil penelitian ini, didapatkan akurasi terbaik pada model CNN dengan menggunakan arsitektur *Mobilenetv2* sebesar 88.53%.

MobilenetV2 merupakan sebuah *pre-trained* CNN, yaitu model arsitektur CNN yang sudah dilatih sebelumnya. *Pre-trained* model biasanya sudah dilatih pada dataset yang besar, sehingga memungkinkan mendapatkan tingkat akurasi yang lebih baik. Terdapat banyak *pre-trained* model yang terdapat saat ini, diantaranya adalah model arsitektur *Mobilenetv2*, *InceptionV3*, dan juga *VGG16*.

Pada penelitian sebelumnya, pendeteksian masker hanya ditujukan untuk mendeteksi menggunakan masker dan tidak menggunakan masker. Namun, pada kenyataannya saat penggunaan masker harus dengan cara yang benar, sehingga manfaat yang didapat dalam menggunakan masker bisa maksimal. Oleh karena itu, pada penelitian ini ditambahkan satu kelas baru sehingga menjadi 3 kelas yaitu menggunakan masker dengan benar, tidak menggunakan masker, dan menggunakan masker

tidak benar. Tujuan dari penelitian ini adalah membandingkan tingkat akurasi dari ketiga model *Convolutional Neural network* (CNN) dalam kasus pendeteksian menggunakan masker. Adapun data yang dilatih untuk model *Convolutional Neural network* berasal dari data gambar dengan *weight ImageNet*. Hasil akurasi terbaik dari proses *training* arsitektur CNN, akan diimplementasikan untuk pendeteksian masker secara otomatis.

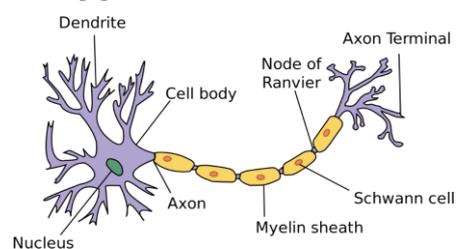
2. Teori Dasar

A. Citra Digital

Citra digital merupakan citra yang dapat diolah komputer. Sebuah citra dapat diwakili oleh sebuah matriks yang terdiri dari M kolom dan N baris, dimana perpotongan antara kolom dan baris tersebut disebut dengan pixel (*picture element*) yang merupakan elemen terkecil dari sebuah citra. Pixel mempunyai dua parameter, yaitu koordinat dan intensitas atau warna. Nilai yang terdapat pada koordinat (x,y) adalah f(x,y) yang merupakan besar intensitas atau warna dari pixel titik tersebut Yang disimpan dalam komputer hanyalah angka-angka yang menunjukkan besar intensitas pada masing-masing piksel. Karena berbentuk data numerik, maka citra digital dapat diolah dengan komputer. [5].

B. Neural Network

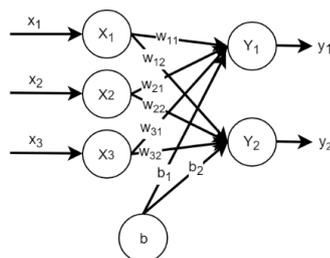
Jaringan Syaraf Tiruan (JST) atau biasa disebut *Neural network* (NN) atau *Artificial Neural network* (ANN) merupakan sebuah mesin pembelajaran yang dibangun dari sejumlah elemen pemrosesan sederhana yang disebut neuron atau node. Jaringan Syaraf Tiruan (JST) berasal dari penelitian pada otak manusia, dimana komponen dasarnya yaitu otak dan saraf, yang ditemukan pada tahun 1836. Sel syaraf memiliki *nucleus* dan dua pelengkap yang lain yaitu *dendrit* dan *akson*. Dendrit memiliki fungsi untuk menerima impuls dari syaraf lain dan diteruskan ke badan sel, sedangkan akson bertujuan untuk menghantarkan impuls yang meninggalkan badan sel dan diteruskan ke neuron lain. [6]



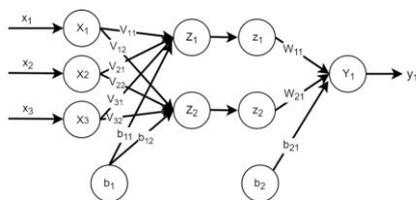
Gambar 1 Susunan Syaraf Biologis Manusia

C. Arsitektur Neural Network

Berdasarkan jumlah lapisan yang dimiliki oleh *neural network*, arsitektur dibedakan menjadi Jaringan Lapisan Tunggal (*Single Preceptron*) dan Jaringan Multi Lapis (*Multi Layer Preceptron*). Sebuah jaringan disebut jaringan lapis tunggal (*single perceptron*) jika jaringan tersebut tidak mempunyai lapisan tersembunyi atau hanya mempunyai satu lapisan bobot koneksi [6]. Pada jaringan ini, sekumpulan *input neuron* dihubungkan langsung dengan sekumpulan *output*. Secara literal, *multi layer perceptron* memiliki beberapa leyers, yaitu *input layer*, *hidden layer* dan *output layer*. *Input layer* menerima *input*, kemudian nilai *input* di berikan ke *hidden units*, *input* diproses lalu hasilnya diberikan ke *layer* berikutnya. *Output* dari *input layer* akan diterima sebagai *input* bagi *hidden layer*. Begitu pula seterusnya, *hidden layer* akan diterima sebagai *input* bagi *hidden layer* selanjutnya. Kegiatan ini dinamakan *feed forward* [7]



Gambar 2 Single Preceptron



Gambar 3 Multi Layer

D. Backpropagation Neural Network

Untuk melatih MLP, algoritma yang umumnya digunakan adalah *backpropagation* [8]. Dalam *backpropagation*, parameter (*weight*) diperbaharui secara bertahap (dari *output* ke *input layer*, karena disebut *backpropagation*) berdasarkan *error/Loss* (*output* dibandingkan dengan *desired output*). [7]

E. Deep Learning

Deep Neural network atau *Deep Learning* adalah *Neural network* yang memiliki banyak *layer*. Pada umumnya, *deep Neural network* memiliki 3 *layers* (*input layer*, $N \geq 2$ *hidden layers*, *output layers*), dengan kata lain adalah *Multi Layer Preceptron* dengan lebih banyak *layer*. Karena relatif banyak

layer, disebutlah *deep*. *Deep learning* dapat menyelesaikan permasalahan yang tidak dapat diselesaikan oleh *multilayer perceptron (3layers)*. [7] *Deep learning* terjadi banyak proses transformasi yaitu mengubah *input* menjadi suatu representasi berbentuk *hidden layer*.

F. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah pengembangan dari *MultilayerPerceptron* (MLP) yang didesain untuk mengolah data dua dimensi. CNN merupakan salah satu algoritma dari *Deep Learning*. Metode CNN memiliki hasil paling signifikan dalam pengenalan citra, hal tersebut dikarenakan CNN berusaha meniru sistem pengenalan citra pada visual cortex manusia, sehingga memiliki kemampuan mengolah informasi citra. [9]. Arsitektur *Convolutional Neural network* (CNN) dibagi menjadi 2 yaitu *feature learning* dan *fully connected layer* (MLP). Didalam *feature learning*, terdapat beberapa *layer* yang difungsikan untuk melakukan filter pada setiap prosesnya, proses tersebut diantaranya adalah *Convolutional layer* dan *Pooling layer*. Dalam *convolutional layer* terjadi proses konvolusi. Konvolusi adalah istilah matematis yang bearti mengaplikasikan sebuah fungsi pada *output* fungsi lain secara berulang [10] Seluruh data yang menyentuh *convolutional layer* akan mengalami proses konvolusi. *Layer* akan mengkonversi setiap filter ke seluruh bagian data masukan dan menghasilkan sebuah *activation map* atau *feature map*. Tujuan dilakukannya konvolusi pada data citra adalah untuk menekstrasi fitur dari citra *input*. [11]. Didalam arsitektur VGG16, *Inceptionv3* dan *Mobilenetv2*, terjadi banyak transformasi konvolusi. Arsaitektur VGG adalah model CNN yang memanfaatkan *convolutional layer* dengan spesifikasi *convolutional filter* yang kecil (3x3) dengan *stride* 1 dan selalu menggunakan padding yang sama dan *maxpooling 2x2 layers*. Susunan ini terus berulang hingga seluruh arsitektur, yang kemudian diakhiri oleh lapisan *fully connected layer* dan diikuti oleh fungsi aktivasi *softmax*. [12]. *Inceptionv3* memiliki modul *inception* sebagai penyusun arsitekturnya. *Inception* modul ini merupakan gabungan dari beberapa konvolusi seperti konvolusi 1x1, 3x3 dan 1x3 dan yang lainnya. Hasil dari konvolusi-konvolusi dari modul *inception* tersebut dijadiann satu lagi dengan menggunakan filter concat sebelum masuk ke iterasi berikutnya. Tujuan dari modul *inception* ini adalah untuk bertidak sebagai *multi-level feature extractor* dengan menghitung filter-filter *convolutions* dalam modul yang sama, dan kemudian hasil dari filter-filter tersebut dijadikan satu lagi oleh filter concat sebelum dimasukan ke lapisan berikutnya

C. Perancangan Arsitektur

Pada penelitian ini, arsitektur yang digunakan adalah arsitektur *pretrained Convolutional Neural network* atau arsitektur yang sudah dilatih sebelumnya, lalu disesuaikan dengan data pada penelitian ini. Sebagaimana yang sudah dijelaskan pada sebelumnya, arsitektur CNN dibagi menjadi 2 bagian besar yaitu *Feature Extractor Layer* dan *Fully Connected Layer*. Pada penelitian ini model *pretrained CNN* digunakan sebagai *Feature Extractor Layer* saja. Oleh karena itu, diperlukan merancang bagian bagian *fully connected layer* yang pada akhirnya digunakan untuk pengklasifikasian data sesuai dengan kebutuhan kelas pada penelitian ini.



Gambar 5. Desain MLP

D. Evaluasi Model

Setelah dilakukan *preprocessing*, merancang arsitektur CNN, dilakukan proses *training*, dan setelah itu dilakukan juga proses *testing*. Hasil dan akurasi *testing* diuji dengan menggunakan *confusion matriks*. *Confusion matriks* berbentuk tabel dengan informasi data *testing* dengan data hasil *testing* (prediksi).

E. Implementasi

Pada tahap implementasi dilakukan proses mengimplementasikan model yang sudah melalui proses *training* dan *testing*. Model yang akan digunakan adalah model salah satu arsitektur yang memiliki tingkat akurasi terbaik. Berikut adalah alur kerja implementasi model untuk sistem deteksi masker yang ditunjukkan oleh dibawah ini:



Gambar 6. Alur Kerja Implementasi

4. Hasil dan Pembahasan

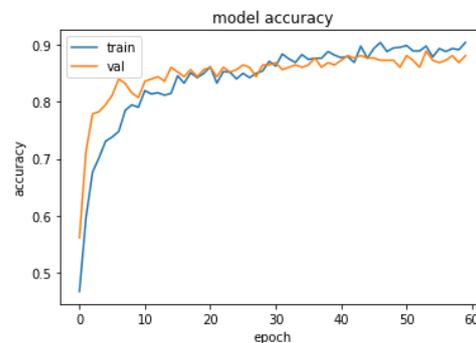
Dalam proses *training*, dilakukan 3 kali pengamatan *epoch* terhadap 3 arsitektur *pretrained CNN* yang digunakan. *Epoch* yang diamati adalah *epoch* 20, 40 dan juga 60. Berikut adalah hasil dari *training* dengan arsitektur *VGG16*, *Mobilenetv2* dan juga *Inceptionv3* yang dijelaskan pada table dibawah ini:

A. Hasil Training

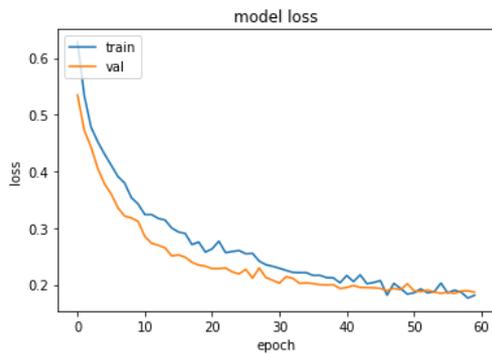
1. VGG16

Tabel 2. Hasil Training VGG16

<i>Epoch</i>	Waktu Training	Akurasi	Loss
20	4 menit 18 detik	0.8443	0.1816
40	10 menit 52 detik	0.8811	0.2027
60	15 menit 16 detik	0.8895	0.1868



Gambar 7 Grafik Accuracy VGG16

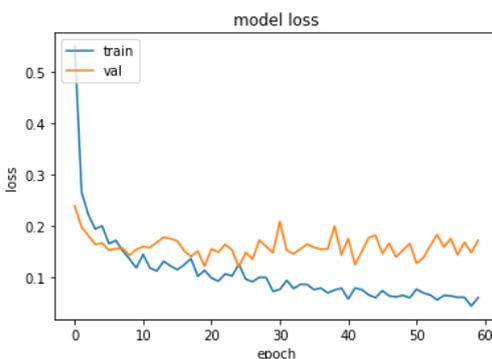
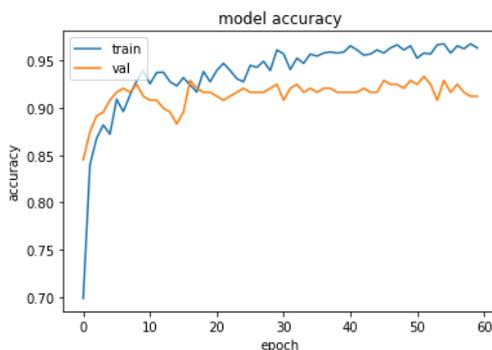


Gambar 8. Grafik Loss VGG16

2. Mobilenetv2

Tabel 3 Hasil Training *Mobilenetv2*

Epoch	Waktu Training	Akurasi	Loss
20	4 menit 9 detik	0.9219	0.1171
40	10 menit 17 detik	0.9416	0.0972
60	15 menit 55 detik	0.9544	0.0758

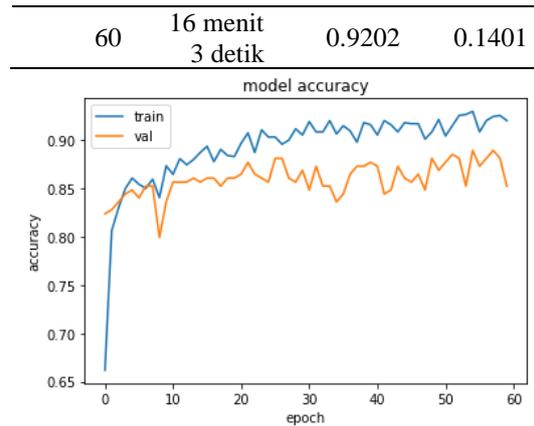


Gambar 9. Grafik Loss *Mobilenetv2*

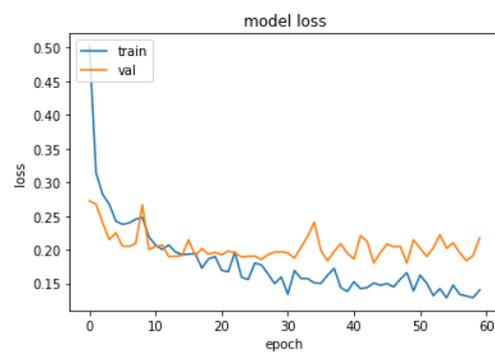
3. Inceptionv3

Tabel 4 Hasil Training *Inceptionv3*

Epoch	Waktu Training	Akurasi	Loss
20	5 menit 3 detik	0.8872	0.1722
40	11 menit 19 detik	0.9266	0.1270



Gambar 10. Grafik Accuracy *Inceptionv3*



Gambar 11. Grafik Loss *Inceptionv3*

Setelah dilakukan hasil *training* ke-3 pretrained arsitektur yaitu *VGG16*, *Mobilenetv2*, dan *Inceptionv3* dengan pengamatan 3 *epoch* untuk setiap arsitektur didapatkan tingkat akurasi yang berbeda-beda. Tingkat akurasi tertinggi didapatkan oleh *Mobilenetv2* dengan 60 *epoch* dengan akurasi 95,44% dan nilai akurasi terendah adalah 84,43% terdapat pada arsitektur *VGG16* dengan 20 *epoch*. Namun berdasarkan grafik yang dihasilkan, tingkat akurasi bisa ditingkatkan kembali dengan menyesuaikan nilai *epoch* sesuai dengan karakteristik arsitektur yang digunakan. Sebagai contoh, pada gambar 4.3 yang menunjukkan grafik akurasi model *Inceptionv3*, nilai akurasi lebih tinggi dengan *epoch* diantara 50-60 dibandingkan dengan tingkat akurasi tepat pada 60 *epoch*.

B. Hasil Testing

Seperti yang diketahui bahwa model *mobilenetv2* adalah model dengan tingkat akurasi *training* yang terbaik. Dengan begitu, evaluasi model akan dilakukan dengan model tersebut menggunakan data testing. Berikut adalah *confusion matriks* dari hasil data testing yang diperoleh pada penelitian ini:

Tabel 5. Hasil Confusion Matriks Mobilenetv2 epoch 60

Predict Class		Pakai Masker Benar	Tidak Pakai Masker	Pakai Masker Tidak Benar
Actual Class		0	1	2
Pakai Masker Benar	0	75	8	1
Tidak Pakai Masker	1	7	66	2
Pakai Masker Tidak Benar	2	1	3	78

$$Precision = \frac{TP}{TP+FP}$$

$$Precision = \frac{218}{218+21} = 0.9121$$

$$Recall = \frac{TP}{TP+FN}$$

$$Recall = \frac{218}{218+14} = 0.9396$$

$$F-1 = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

$$F-1 = \frac{2 \times 0.9396 \times 0.9121}{0.9396 + 0.9121} = 0,9256$$

C. Hasil Implementasi

Telah diketahui bahwa nilai akurasi tertinggi untuk *training* untuk deteksi masker dengan menggunakan model arsitektur pretrained CNN adalah *Mobilenetv2* dengan *epoch* 60, dengan begitu untuk mengimplementasikan sistem deteksi masker secara otomatis digunakan model yang telah disimpan dalam format H5. Berikut adalah hasil dari implementasi sistem deteksi masker secara otomatis yang diterangkan oleh tabel dibawah ini:

Tabel 6. Hasil Implementasi Sistem Deteksi Masker

No	Kelas	Hasil	Confidence
1	Menggunakan Masker dengan Benar		99,87%
2	Tidak Menggunakan Masker		100%
3	Menggunakan Masker Tidak Benar		99,40%

5. Kesimpulan

Setelah dilakukan pengujian untuk membandingkan ke-3 arsitektur pretrained Concolutional Neural network (CNN) (*VGG16*, *Mobilenetv2*, dan *Inceptionv3*), *Mobilenetv2* memiliki performa keakuratan yang lebih baik dibandingkan dua model arsitektur yang lainnya. Selain itu, model *Mobilenetv2* dapat diimplemntasikan kedalam sistem pendeteksian masker secara otomatis dengan baik, dengan tingkat *confidence* tertinggi sebesar 100%.

Saran yang dapat dikembangkan lebih lanjut untuk penelitian selanjutnya adalah untuk menambah model pretrained CNN yang lain, menambah klasifikasi dataset seperti jenis masker sehingga sistem pendeteksian masker secara otomatis lebih baik. Selain itu, penelitian terhadap pengaruh intensitas cahaya dan juga sudut

pandang kamera akan menjadikan sistem ini menjadi lebih sempurna.

Daftar Pustaka

- [1] H. B. Muftah Afrizal Pangestu, "Analisis Performa dan Pengembangan Sistem Deteksi Ras Anjing pada Gambar Menggunakan Pre-Trained CNN Model," *Jurnal Teknik Informatika dan Sistem Informasi Volume 4 No 2 Agustus 2018*, 2018.
- [2] W. H. Organization, "Anjuran Mengenai Penggunaan Masker dalam Konteks Covid-19," *Panduan Sementara*, p. 1, 6 April 2020.
- [3] K. E. T. L. Arham Rahim, "Convolutional Neural Network untuk Klasifikasi

- Penggunaan Masker," *Jurnal Teknologi Informasi dan Komunikasi Volume 10 Nomer 2*, 2020.
- [4] C. L. N. J. P. Bunardi Budiman, "Pendeteksian Penggunaan Masker Wajah dengan Metode Convolutional Neural Network," *Jurnal Ilmu Komputer dan Informasi*, 2020.
- [5] H. S. M. Y. Nadzir Zaid Munantri, "Aplikasi Pengolahan Citra Digital untuk Identifikasi Umur Pohon," *TELEMATIKA*, 2019.
- [6] L. Fausset, *Fundamental of Neural Network*, Upper Sidle Rider, New Jersey: Prentice Hall, 1944.
- [7] J. W. G. Putra, *Pengenalan Konsep Pembelajaran Mesin dan Deep Learning*, Tokyo, Jepang, 2017.
- [8] H. G. E. a. W. R. Rumelgart D.E, *Parallel distributed processing; Explorations in the microstructure of cognition*, Cambridge, USA: MIT Press, 1986.
- [9] A. Y. W. R. S. Wayan Suartika E.P, "Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101," *Jurnal Teknik ITS*, 2016.
- [10] H. B. Muftah Afrizal Pangestu, "Analisis Performa dan Pengembangan Sistem Deteksi Ras Anjing pada Gambar Menggunakan Pre-Trained CNN Model," *Jurnal Teknik Informatika dan Sistem Informasi Volume 4 No 2 Agustus 2018*, 2018.
- [11] G. A. Aditya Santoso, "Implementasi Deep Learning Berbasis Keras untuk Pengenalan Wajah," *Jurnal Teknil Elektro*, 2018.
- [12] A. Z. Karen Simonyan, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *Conference paper at ICLR*, Enngland, 2015.
- [13] J. H. D. E. H. Nadia Ramadhani, "Pencarian Objek Wisata Bersejarah di Pulau Jawa Menggunakan Convolutional Neural Network," *Jurnal Ilmu Komputer dan Sistem Informasi*, 2018.
- [14] A. H. M. z. A. Z. L.-C. C. Mark sandler, "MobileNetV2 : Inverted Residual and Linear Bottlenecks," *IEEE/CVF Conforence on Computer Vision and Pattern Recognition*, 2018.
- [15] D. H. S. I. S. Samuel Febrian Tumewu, "Kalsifikasi Motif Batik menggunakan Metode Deep Convolutional Neural Network dengan Data Augmentation," *Jurnal Infra* , 2020.