

# Deep Transfer Learning dan Augmentasi Gambar Pada Klasifikasi Masker

Erwin Erdiyanto<sup>1</sup>, Achmad Andrian Maulana<sup>2</sup>, Muhammad Ahsanur Rafi<sup>3</sup>, Ahmad Anwar Saifurridzal<sup>4</sup>, Mochammad Thoriq Wafa<sup>5</sup> Andika Wira Yumna<sup>6</sup>, Dr. Basuki Rahmat, S.Si. MT.<sup>7</sup>

<sup>1,2,3,4,5,6</sup> Informatika (Universitas Pembangunan Nasional "Veteran" Jawa Timur)

[21081010288@student.upnjatim.ac.id](mailto:21081010288@student.upnjatim.ac.id)<sup>1</sup>

[21081010267@student.upnjatim.ac.id](mailto:21081010267@student.upnjatim.ac.id)<sup>2</sup>

[21081010305@student.upnjatim.ac.id](mailto:21081010305@student.upnjatim.ac.id)<sup>3</sup>

[21081010011@student.upnjatim.ac.id](mailto:21081010011@student.upnjatim.ac.id)<sup>4</sup>

[21081010309@student.upnjatim.ac.id](mailto:21081010309@student.upnjatim.ac.id)<sup>5</sup>

[21081010002@student.upnjatim.ac.id](mailto:21081010002@student.upnjatim.ac.id)<sup>6</sup>

<sup>7</sup> Informatika, Universitas Pembangunan Nasional Veteran Jawa Timur

\*Corresponding author email: [basukirahmat.if@upnjatim.ac.id](mailto:basukirahmat.if@upnjatim.ac.id)

**Abstrak**— Covid 19 atau Coronavirus Disease 2019 adalah penyakit baru yang dapat menyebabkan gangguan pernapasan dan radang paru. Penyakit ini disebabkan oleh infeksi Severe Acute Respiratory Syndrome Coronavirus. Oleh sebab itu, terdapat dampak yang merubah kebiasaan masyarakat. Masyarakat dianjurkan untuk memakai masker saat berada di ruangan umum maupun diluar ruangan. Maka dari itu mengatasi masalah tersebut dibutuhkan sebuah sistem yang dapat membedakan antara wajah yang memakai masker dan wajah yang tidak memakai masker dengan menghasilkan tingkat akurasi yang baik, yaitu menggunakan Model arsitektur deep learning yang menggunakan arsitektur berbasis Convolution Neural Network. Meskipun begitu, terdapat teknik transfer learning dan augmentasi gambar yang diterapkan untuk meningkatkan kinerja model Deep learning yang dapat diusulkan secara keseluruhan. Pada penelitian ini menggunakan data set Celeb A untuk wajah yang tidak memakai masker adapun dataset masked face net untuk wajah yang bermasker dengan benar dan yang memakai masker. Setelah melakukan percobaan augmentasi gambar. Selanjutnya terdapat hasil pengujian, pendekatan augmentasi gambar terbukti dapat meningkatkan akurasi kinerja model dengan melakukan verifikasi data. Selanjutnya, jika percobaan telah selesai lalu melakukan transfer learning yang telah terbukti meningkatkan akurasi model secara keseluruhan

**Kata Kunci**— Covid 19, Klasifikasi, Deteksi masker wajah, deep learning, transfer learning, augmentasi gambar

COVID- 19 yaitu masker. Masker merupakan suatu alat yang berfungsi sebagai pelindung diri dari penularan virus yang berbahaya. Pada tanggal 29 januari 2020 Organisasi Kesehatan Dunia (WHO) mengeluarkan anjuran penggunaan masker dalam konteks COVID-19 (WHO, 2020)[1]. Terdapat masalah yang terjadi untuk proses pengenalan wajah, yaitu menutupi sebagian besar wajah, sehingga mengurangi akurasi pengenalan. Laporan dari US National Institute of Standard and Technology (NIST) pada akhir Juli menyatakan bahwa penggunaan masker dapat menurunkan akurasi pengenalan wajah antara 5% hingga 50%, berdasarkan uji coba terhadap 89 algoritma pengenalan wajah komersial.

Untuk mengatasi masalah ini, penelitian ini mengusulkan penggunaan Convolutional Neural Network (CNN), sebuah metode deep learning, guna meningkatkan akurasi dalam mendeteksi wajah bermasker. CNN dianggap sebagai model yang paling tepat karena kemampuannya dalam mendeteksi dan mengenali objek tanpa memerlukan komputasi besar. Menurut Indra (Alam et al., 2019), CNN dapat menjadi model terbaik untuk deteksi dan pengenalan objek. Yann LeCun (LeCun et al., 2015) menyatakan bahwa CNN adalah model deep learning yang dapat mengurangi jumlah parameter bebas serta menangani deformasi gambar seperti translasi, rotasi, dan skala dengan baik[2].

CNN yang efektif dalam mendeteksi wajah bermasker dan penelitian ini bertujuan untuk mengimplementasikan deep learning guna klasifikasi wajah bermasker menggunakan metode CNN. Tujuan utama penelitian ini adalah mengembangkan algoriengimplementasikannya dalam aplikasi praktis. Penelitian ini diharapkan dapat berkontribusi dalam pengembangan ilmu matematika dan teknologi informasi, serta memberikan wawasan tentang klasifikasi wajah bermasker dengan metode CNN.

## II. METODE

## I. PENDAHULUAN

Dalam beberapa tahun yang lalu, terdapat masalah kesehatan akibat pandemi COVID-19 yang sangat membahayakan. Virus Covid-19 terbilang cepat dalam menyebar ke seluruh wilayah dan menyebabkan efek jangka panjang pada berbagai negara, diantaranya india, brasil, dan indonesia. Virus corona memiliki beberapa tipe varian, tetapi varian delta dari virus corona adalah yang paling menular dan bahaya. Benda yang wajib digunakan pada saat pandemi

[3]Metode yang digunakan untuk mengidentifikasi wajah yang menggunakan masker meliputi studi literatur, pengumpulan data, perancangan sistem, pelatihan sistem, dan pengujian sistem.

### 2.1. Studi Literatur

Studi literatur dalam penelitian ini digunakan untuk menganalisis dan menemukan metode yang paling sesuai untuk mencapai tujuan penelitian dan menyelesaikan masalah, termasuk *deep learning*, *convolutional neural network*, *tensorflow*. Berikut ini adalah uraian dari masing-masing studi literatur tersebut.

#### 2.1.1. Deep Learning

Deep learning adalah algoritma permodelan yang tersusun secara mendalam dan berlapis-lapis pada sekumpulan fungsi transformasi non-linier. Algoritma ini memanfaatkan berbagai pengolahan informasi non-linier untuk ekstraksi fitur, pengenalan pola, dan klasifikasi (Lambacing et al., 2020).

#### 2.1.2. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah salah satu algoritma dalam deep learning yang digunakan untuk mengklasifikasi gambar dan video, serta mendeteksi objek di dalam gambar atau area tertentu dalam gambar (Moolayil, 2019). CNN terdiri dari lapisan-lapisan yang memiliki susunan neuron dalam bentuk 3D, yaitu lebar, tinggi, dan kedalaman. Lebar dan tinggi mewakili ukuran lapisan, sedangkan kedalaman menunjukkan jumlah lapisan (Zufar et al., 2016).

#### 2.1.3. Tensorflow

TensorFlow adalah library open source yang dikembangkan oleh Tim Google Brain. Dalam deep learning, TensorFlow digunakan untuk secara otomatis mengelola memori yang digunakan oleh data. TensorFlow dapat digunakan untuk menjalankan dan melatih jaringan neural untuk keperluan klasifikasi.

#### 2.1.4. Klasifikasi

Klasifikasi adalah proses untuk menemukan fitur-fitur yang dapat membedakan atau mendeskripsikan data ke dalam kategori kelas tertentu. Tujuan klasifikasi adalah untuk memperkirakan kelas suatu objek yang labelnya belum diketahui.

### 2.2. Pengumpulan Data

Dataset yang digunakan dalam penelitian ini berisi gambar wajah dengan dan tanpa masker, terbagi menjadi tiga file: data training, data validasi, dan data testing. Untuk meningkatkan tingkat akurasi, program training dilakukan secara intensif dengan memanfaatkan jumlah data yang cukup besar.

Data yang digunakan merupakan data skunder yang diperoleh dari <https://www.kaggle.com/wobotintelligence/face-mask-detection-dataset>, terdiri dari 4326 gambar yang dibagi menjadi dua kelas. Setelah pemilihan data, beberapa data diubah dan ditambahkan. Kemudian, data dibagi

menjadi tiga file: training, validasi, dan testing, dengan masing-masing file terbagi menjadi dua kategori: dengan masker dan tanpa masker. Proses ini menghasilkan total 3847 gambar.

### 2.3. Perancangan Sistem

Perancangan sistem klasifikasi wajah dilakukan pada tahapan ini. Tahapan ini terdiri dari grayscale, thresholding, segmentasi, dan resize.

#### 2.3.1. Grayscale

Dalam tahap ini, gambar berwarna Red, Green, Blue (RGB) diubah menjadi gambar keabuan (greyscale). Menurut (Sinaga, 2019), setiap komponen RGB memiliki rentang nilai antara 0 hingga 255 atau setara dengan 8 bit, yang secara umum disebut sebagai true color atau intensitas 24bit. Namun, untuk mengurangi kompleksitas dan mempermudah proses, citra RGB diubah menjadi citra keabuan, yang hanya memerlukan satu saluran warna. Untuk mencapai ini, tiap komponen RGB dikalikan, sehingga jumlah byte yang diperlukan untuk file citra RGB lebih dari tiga kali ukuran file citra keabuan. Perhitungan ini dilakukan untuk menormalkan rentang warna citra menjadi 0 hingga 255, sehingga mempermudah proses pada tahapan thresholding selanjutnya. Perhitungan tersebut dapat dijelaskan dengan persamaan berikut:

$$y = (0.2989 * R) + (0.5870 * G) + (0.1141 * B)$$

Keterangan

Y : *pixel grayscale value*

R : *pixel red value*

G : *pixel green value*

B : *pixel blue value*

#### 2.3.2. Thresholding

Pada tahap ini, dilakukan proses untuk menghasilkan citra biner dari citra berwarna (citra grayscale). Proses ini dilakukan dengan menetapkan nilai piksel ke nilai 0 atau tergantung pada ambang batas yang ditetapkan. Jika nilai piksel di atas ambang batas, maka nilai piksel tersebut diatur menjadi 1, sedangkan jika nilainya di bawah ambang batas, maka nilai piksel tersebut diatur menjadi 0. Proses ini membantu memisahkan objek dari latar belakang dan menyederhanakan representasi citra untuk mempermudah proses analisis selanjutnya (Solomon, 2011).

#### 2.3.3. Segmentasi

Proses segmentasi merupakan tahap pembagian kelompok pada gambar, di mana dalam konteks penelitian ini, langkah ini melibatkan hasil dari proses thresholding yang menghasilkan citra biner. Proses ini melibatkan dua tahap utama: pertama, pemotongan pada setiap baris secara vertikal, dan kedua, pemotongan pada setiap kolom secara horizontal. Dengan demikian, gambar akan terbagi menjadi berbagai kelompok atau region yang mewakili bagian-bagian yang berbeda dari objek yang ingin diidentifikasi atau dianalisis. Proses ini membantu dalam memisahkan dan mengidentifikasi bagian-bagian penting

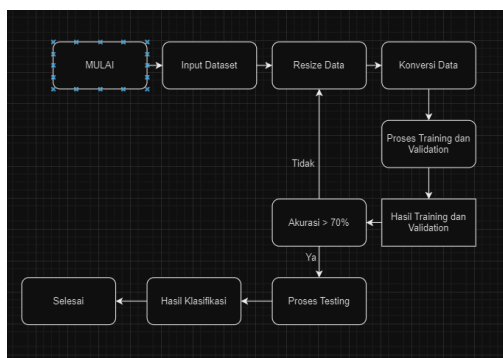
dari gambar, memungkinkan analisis lebih lanjut untuk dilakukan dengan lebih efektif.

### 2.3.4. Resize

Setelah proses segmentasi, langkah selanjutnya adalah resize. Resize adalah proses untuk mengubah ukuran citra menjadi lebih kecil atau lebih besar dari ukuran sebelumnya. Dalam penelitian ini, citra wajah di-resize menjadi citra berukuran 150x150 piksel. Langkah ini penting untuk memastikan konsistensi ukuran citra dan mempersiapkan data secara optimal untuk proses klasifikasi atau analisis selanjutnya. Dengan ukuran yang telah ditetapkan, citra-citra wajah akan memiliki dimensi yang seragam, memudahkan dalam proses pengolahan dan analisis.

### 2.4. Pelatihan Sistem

Pelatihan sistem dimulai sebagai langkah awal dalam pengolahan data, di mana dataset citra input diproses untuk pelatihan dan validasi menggunakan arsitektur CNN. Tahapan dalam arsitektur CNN termasuk inisialisasi, feedforward, backpropagation, dan update bobot. Proses ini mengubah citra biner menjadi output yang merupakan klasifikasi apakah wajah tersebut bermasker atau tidak.



Gambar 1. Flowchart proses training sistem

#### 2.4.1. Inisialisasi

Pada tahapan ini, dilakukan inisialisasi parameter yang mencakup penentuan jumlah epoch, nilai learning rate, dan minimum error. Selain itu, dilakukan juga inisialisasi bobot yang melibatkan penentuan nilai awal untuk bobot dan bias pada berbagai tahap seperti convolution layer, hidden layer, dan fully connected layer.

#### 2.4.2. Feedforward

Tahapan ini merupakan proses arsitektur CNN, yang melibatkan beberapa langkah seperti convolution layer, activation layer, pooling layer, dan fully connected layer. Proses ini mengolah data melalui serangkaian layer tersebut sehingga menghasilkan keluaran berupa vektor yang merepresentasikan hasil dari proses tersebut.

##### a. Convolution Layer

Tahapan ini merupakan lapisan yang melakukan proses konvolusi antara matriks citra input dengan matriks filter sehingga menghasilkan keluaran berupa feature map.

Persamaan rumus dari operasi konvolusi dapat dituliskan sebagai berikut:

$$FM[i]_{j,k} = (\sum_m \sum_n N_{[j-m,k-n]} F_{[m,n]}) + bF$$

##### Keterangan

FM[i]	:	Matriks <i>feature map</i> ke-i
j,k	:	Posisi <i>pixel</i> pada matriks citra input
m,n	:	Posisi <i>pixel</i> pada matriks <i>filter</i> konvolusi
N	:	Matriks citra masukan
F	:	Matriks <i>filter</i> konvolusi
bF	:	Nilai bias pada <i>filter</i>

selanjutnya dengan memasukkan fungsi aktivasi Rectified Linear Unit (ReLU). Seluruh pixel yang ada di feature map dimasukkan ke dalam fungsi ReLU, dimana fungsi aktivasi ini bekerja memilah pixel yang memiliki nilai kurang dari 0, dengan rumus  $f(x) = \max(0, x)$ .

##### b. Pooling Layer

Pada lapisan ini dilakukan proses pooling. Jenis yang digunakan pada penelitian ini adalah max-pooling. Proses ini bekerja untuk mengurangi nilai dari feature map dengan cara memilah nilai maksimum di area tertentu. Hasil dari proses ini berupa matriks feature map yang berisi nilai maksimum yang telah dipilih.

29	15	28	184
0	100	70	34
12	12	7	2
12	12	45	6

➡

100	184
12	45

Gambar 2. Contoh proses max pooling.

##### c. Fully Connected Layer

Lapisan ini terdiri dari input layer, hidden layer dan output layer.

##### d. Input Layer

Pada tahap ini, seluruh matriks peta fitur yang diperoleh dari proses pooling digabungkan. Selanjutnya, semua piksel dari matriks tersebut diubah menjadi vektor dengan panjang yang sesuai dengan jumlah piksel pada matriks hasil pooling. Nilai-nilai yang diperoleh dari proses ini akan digunakan untuk perhitungan pada tahap berikutnya.

##### e. Hidden Layer

Lapisan ini melakukan perhitungan dengan cara mengalikan nilai dari lapisan input dengan bobot yang telah diinisialisasi, kemudian menambahkan nilai bias. Rumus perhitungan tersebut dapat ditulis sebagai berikut:

$$z\_in_i = \sum_{j=1}^n X_j * V_{j,i} + V_{o,i}$$

##### Keterangan

$z\_in_i$	:	Input untuk <i>node</i> hidden layer ke-i dengan jumlah <i>node</i> n
$X_j$	:	<i>Node</i> X ke-j
$V_{j,i}$	:	Bobot V untuk $X_j$ dengan <i>node</i> $Z_i$
$V_{o,i}$	:	Bias V untuk $z\_in_i$

Selanjutnya dengan memasukkan fungsi aktivasi ReLU pada seluruh hasil perhitungan sehingga didapatkan nilai

output Z. Kemudian digunakan pada proses perhitungan di output layer

#### f. Output Layer

Pada Output Layer dilakukan perhitungan hasil dari *hidden layer* dikalikan dengan bobot yang sudah diinisialisasi, kemudian nilai tersebut ditambahkan dengan nilai bias. Rumus dari perhitungan dapat ditulis sebagai berikut:

$$y_{in_i} = \sum_{j=1}^m Z_j * W_{j,i} + W_{o,i}$$

Keterangan

$y_{in_i}$	:	Masukan untuk <i>node hidden layer</i> Z ke-i dengan jumlah <i>node</i> m
$W_{j,i}$	:	Bobot W untuk $Z_j$ dengan <i>node</i> $Y_i$
$W_{o,i}$	:	Bias W untuk $y_{in_i}$
$Z_j$	:	<i>Node</i> Z ke-j

Selanjutnya dengan memasukkan fungsi aktivasi, pada penelitian ini digunakan *softmax* untuk semua hasil dari perhitungan, kemudian diperoleh nilai hasil perhitungan berupa nilai output Y. Rumus dari fungsi dapat ditulis sebagai berikut:

$$Y_i = \frac{e^{y_{in_i}}}{\sum_{m=1}^m e^m}$$

Keterangan

M	:	Semua masukan untuk <i>output layer</i> sejumlah m
$y_{in_i}$	:	Input untuk <i>node layer</i> ke-i
$Y_i$	:	Output untuk <i>output layer</i> ke-i

#### 2.4.3. Backpropagation

Tahapan ini dilakukan untuk menyesuaikan bias dan seluruh bobot berdasarkan kesalahan yang diperoleh selama proses feedforward. Langkah-langkah dalam proses backpropagation adalah sebagai berikut: (1) menghitung gradien kesalahan terhadap bobot  $W_{ij}$ , (2) menghitung gradien kesalahan terhadap bobot  $V_{kj}$  dan (3) menghitung gradien kesalahan terhadap filter.

Setelah gradien pada lapisan input didapatkan, nilai tersebut dikembalikan dalam bentuk peta fitur (feature map). Kemudian, berdasarkan nilai maksimum pada peta fitur hasil dari proses max-pooling selama feedforward, dilakukan perhitungan nilai gradien terhadap lapisan pooling. Hasil akhir dari tahap ini adalah gradien yang digunakan untuk memperbarui filter.

#### 2.4.4. Update Parameter

Pada tahap ini metode yang digunakan pada penelitian adalah metode adam optimizer. Langkah-langkah pada proses update parameter yaitu, (1) hitung nilai momentum pertama ( $m_t$ ), (2) hitung nilai momentum kedua ( $v_t$ ), (3) hitung nilai momentum pertama dengan pembaruan bias ( $\tilde{m}_t$ ), (4) update bobot dengan persamaan berikut:

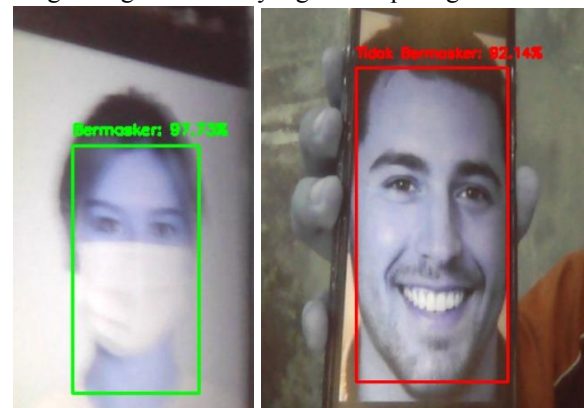
$$\theta_t = \theta_{t-1} - \alpha \left( \frac{\tilde{m}_t}{\sqrt{v_t + \epsilon}} \right)$$

Keterangan

$\theta_t$	:	Nilai dari bobot baru
$\theta_{t-1}$	:	Nilai dari bobot lama
$\alpha$	:	Learning rate
$\tilde{v}_t$	:	Nilai momentum kedua dengan bias baru
$\tilde{m}_t$	:	Nilai momentum pertama dengan bias baru

#### 2.5. Pengujian Sistem

Setelah melewati proses pelatihan sistem, selanjutnya akan dilakukan pengujian sistem yang mana data akan langsung terklasifikasi. Pengujian sistem memberikan hasil akhir berupa kecocokan antara data masukan dan database dengan tingkat akurasi yang baik seperti gambar.



Gambar 3. Pengujian sistem.

### III. KESIMPULAN

Kesimpulan yang diperoleh dari penelitian ini, penelitian berhasil mengimplementasi deep learning dengan menggunakan metode convolutional neural network, serta menggunakan library tensorflow untuk klasifikasi gambar wajah bermasker yang terbagi dalam kelas with mask dan without mask. Dari hasil pengujian dataset sebanyak 3847 gambar ukuran 150x150 px dan epoch sebanyak 20 memperoleh tingkat akurasi data sebesar 98,20% pada 5 layer. Pencocokan 17 data testing menghasilkan semua data memiliki kecocokan dengan database yang ada. Penggunaan jumlah layer mempengaruhi waktu dan tingkat akurasi pada pengujian data. Proses dengan jumlah layer yang sedikit tidak membutuhkan waktu yang lama, sementara proses dengan jumlah layer yang lebih banyak membutuhkan waktu yang cukup lama dan memperoleh hasil klasifikasi yang lebih baik.

#### UCAPAN TERIMA KASIH

Kami mengucapkan terima kasih yang sebesar-besarnya kepada dosen pengampu, Bapak Dr. Basuki Rahmat, S.Si. MT. yang telah memberikan bimbingan dan dukungan selama satu semester ini. Terima kasih atas semua nasihat, koreksi, dan pengetahuan yang diberikan kepada kami.

Kami juga ingin menyampaikan rasa terima kasih kepada seluruh tim yang telah bekerja sama dan berkontribusi dalam menyelesaikan tugas akhir ini. Kerja keras, dedikasi, dan kerjasama yang solid dari setiap anggota tim sangat berharga dan tak ternilai harganya. Terima kasih atas semua usaha dan

waktu yang telah dicurahkan untuk menyelesaikan proyek ini dengan baik. Semoga hasil dari tugas akhir ini dapat memberikan manfaat dan kontribusi positif.

#### REFERENSI

- [1] Alam, I. F., Sarita, M. I., & Sajiah, M. A. (2019). Implementasi Deep Learning dengan Metode Convolutional Neural Network (CNN) untuk Identifikasi Objek Secara Real Time Berbasis Sistem Android. *semanTIK*, 237-244.
- [2] Lambacing, M. M., & Ferdiansyah. (2020). Rancang Bangun New Normal COVID-19 Masker Detector Dengan Notifikasi Telegram Berbasis Internet of Things. *Jurnal Dinamik*, 25, 77-84.
- [3] NIST. (2020). Algorithms created before the pandemic generally perform less accurately with digitally masked faces.(Online). (<https://www.nist.gov/news-events/news/2020/07/nist-launches-studies-masks-effect-face-recognition-software>), diakses 20 September 2021