

# Pengenalan Masker Wajah Menggunakan VGG-16 dan *Multilayer Perceptron*

Khairul Fadhli Margolang<sup>#1</sup>, Sugeng Riyadi<sup>#2</sup>, Rika Rosnelly<sup>#3</sup>, Wanayumini<sup>#4</sup>

<sup>#</sup>Program Magister Ilmu Komputer, Universitas Potensi Utama  
Jl. K. L. Yos Sudarso Km. 6,5 No. 3-A, Medan, Indonesia

<sup>1</sup>[khairulfadhl@gmail.com](mailto:khairulfadhl@gmail.com)

<sup>2</sup>[adhie.ogenk@gmail.com](mailto:adhie.ogenk@gmail.com)

<sup>3</sup>[rikarosnelly@gmail.com](mailto:rikarosnelly@gmail.com)

<sup>4</sup>[wanayumini@gmail.com](mailto:wanayumini@gmail.com)

**Abstract**— *The use of face masks during the Covid-19 pandemic can be identified based on images taken of a person's face and then classified based on the results of their feature extraction. VGG 16 is a pre-trained CNN model that can extract 4,096 features from an image and transfer learning to the multilayer perceptron algorithm in classifying someone using a face mask. The results of this study indicate that the combination of ReLu activation with adaptive moment optimization (Adam) and stochastic gradient descent (SGD), the combination of ReLu and Adam, produces the best classification performance with accuracy, precision, and recall values of 98.1%.*

**Keywords**— *ReLu, Adam, VGG 16, multilayer perceptron, transfer learning*

**Abstrak**— *Penggunaan masker wajah pada masa pandemi Covid-19 dapat diidentifikasi berdasarkan citra yang diambil dari wajah seseorang kemudian diklasifikasi berdasarkan hasil ekstraksi fiturnya. VGG 16 merupakan sebuah pre-trained CNN model yang dapat mengekstrak 4.096 fitur dari sebuah citra dan melakukan transfer learning kepada algoritme multilayer perceptron dalam mengklasifikasikan seseorang menggunakan masker wajah atau tidak. Hasil dari penelitian ini menunjukkan bahwa kombinasi aktivasi ReLu dengan optimasi adaptive moment (Adam) dan stochastic gradient descent (SGD), kombinasi ReLu dan Adam, menghasilkan performa klasifikasi terbaik dengan nilai accuracy, precision, dan recall sebesar 98,1%.*

**Kata Kunci**— *ReLu, Adam, VGG 16, multilayer perceptron, transfer learning*

## I. PENDAHULUAN

Covid-19 merupakan virus yang menyebabkan pandemi di Indonesia yang dimulai pada akhir tahun 2020 [1]. Sebagai tindakan pencegahan terhadap penyebaran virus ini, *face mask* (masker wajah) merupakan salah satu alat proteksi yang dianjurkan, bahkan diwajibkan, untuk digunakan sebagai *personal protective equipment* (PPE) [2]. Dengan menggunakan citra sebagai masukan, dapat dibentuk model untuk mengenali apakah seseorang mengenakan masker wajah atau tidak, khususnya dengan bantuan metode *machine learning* [3]. Beberapa penelitian sudah pernah melakukan hal ini, seperti pada penelitian yang menggunakan *pre-trained*

*CNN model* MobileNet V2 dan kombinasi tiga fungsi optimasi Adam, ADAGRAD, dan SGD dengan rata-rata akurasi 99,033% [4]; pada penelitian yang menggunakan *pre-trained CNN model* MobileNet V2, VGG-16, DenseNet201, dan Xception dengan akurasi tertinggi sebesar 98,1% [5]; serta pada penelitian yang menggunakan kombinasi SqueezeNet sebagai metode ekstraksi fitur dengan algoritme Naïve Bayes, *support vector machine* untuk proses klasifikasi, yang menghasilkan rata-rata akurasi 97,5% [6].

VGG-16 merupakan sebuah *pre-trained CNN model* yang dapat digunakan dalam proses *transfer learning* dengan cara mengekstrak fitur pada citra masukan dan mengirimkannya sebagai masukan bagi algoritme lain [7]. Model yang dapat diakses melalui ImageNet ini memiliki kemampuan untuk mengekstrak fitur-fitur, seperti spasial, garis tepi, rotasi, pencahayaan, dan bentuk [8]. VGG-16 dapat dimanfaatkan untuk mengekstraksi fitur dari sebuah citra sehingga dapat diklasifikasikan seseorang menggunakan masker wajah atau tidak.

*Perceptron* merupakan algoritme *neural network* sederhana yang hanya memiliki satu lapisan *hidden layer* [9]. Algoritme ini kemudian dikembangkan menjadi *multilayer perceptron* (MLP) yang memiliki lebih dari satu *hidden layer* pada arsitekturnya sehingga mampu menangani masalah yang lebih kompleks [10]. Dengan penambahan *hidden layer* pada arsitekturnya, MLP mampu beradaptasi terhadap *noise* pada data, dapat digunakan untuk mengklasifikasikan data *non-linearly seperable*, dan proses pelatihannya menjadi lebih dalam sehingga dapat dikategorikan sebagai algoritme *deep learning* [11]. MLP sudah banyak diteliti untuk masalah klasifikasi, khususnya yang berhubungan dengan bidang *computer vision*, seperti pada penelitian pengenalan citra ikan berformalin [12], prediksi beban sedimen tersuspensi [13], dan pengenalan angka tulisan tangan [14].

Beberapa penelitian sudah mencoba mengkombinasikan VGG-16 dengan MLP, dengan hasil yang cukup baik, seperti pada penelitian deteksi kematangan buah dengan rata-rata akurasi 83,6% [15], pengenalan gaya berjalan dengan rata-rata akurasi 99,77% [16], dan klasifikasi batik dengan rata-rata akurasi 89,01% [17].

Hasil *transfer learning* dari jaringan VGG-16 ke algoritme MLP pada penelitian ini dianalisis performa klasifikasinya

dengan menggunakan kombinasi fungsi aktivasi *rectified linear unit* (ReLU) dan dengan fungsi optimasi *stochastic gradient descent* (SGD) dan *adaptive moment*. Nilai *accuracy*, *precision*, dan *recall* yang diperoleh dari evaluasi *10-fold cross validation* digunakan untuk melihat performa model MLP yang terbaik.

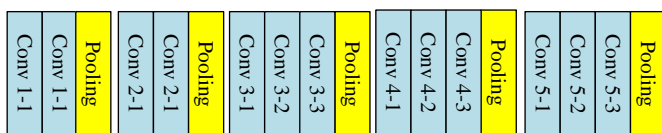
II. METODOLOGI

Pada penelitian ini digunakan 320 citra data latih yang terdiri atas 160 citra orang yang memakai *face mask*, dan 160 citra orang yang tidak memakai *face mask*. Untuk citra data uji digunakan citra sebanyak 80 yang terdiri dari 40 citra orang yang memakai *face mask* dan 40 citra orang yang tidak memakai *face mask*. Seluruh citra yang digunakan pada penelitian ini diperoleh dari situs *Kaggle.com* [18].

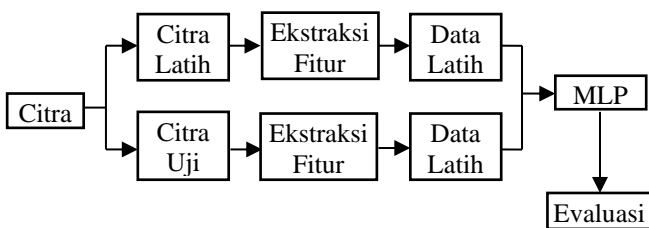
Masing-masing citra yang digunakan, baik dalam proses pelatihan maupun pengujian, diekstrak fitur-fitur di dalamnya menggunakan *pre-trained CNN model*, yaitu VGG-16. Dalam proses ekstraksi fitur, seperti yang ditunjukkan pada Gambar 1, VGG-16 menggunakan 13 *convolutional layer* dan 5 *pooling layer*, dan menghasilkan keluaran ekstraksi berupa 4096 fitur dari masing-masing citra yang diproses (lihat Gambar 2) yang kemudian digunakan sebagai masukan pada algoritme MLP.

Algoritme MLP dengan jumlah tiga *hidden layer* digunakan untuk mengklasifikasikan citra berdasarkan hasil ekstraksi fitur yang diterima melalui *input layer* (lihat Gambar 3). Pada penelitian ini digunakan enam model MLP dengan variasi kombinasi fungsi aktivasi ReLU dan fungsi optimasi Adam serta SGD. Kombinasi fungsi aktivasi RBF dan fungsi optimasi Adam terdiri dari tiga model yang memiliki variasi jumlah *neuron* yang berbeda, yaitu 50, 100, dan 150 untuk setiap *hidden layer*. Kombinasi fungsi aktivasi ReLU dan fungsi optimasi SGD juga terdiri dari tiga model yang memiliki variasi jumlah *neuron* yang berbeda, yaitu 50, 100, dan 150 untuk setiap *hidden layer*. Kombinasi tersebut diperlihatkan pada Tabel I.

Fungsi aktivasi ReLU yang digunakan pada proses klasifikasi berfungsi untuk memfilter keluaran hasil perhitungan dari lapisan sebelumnya, sehingga keluaran pada



Gambar 1 Arsitektur VGG-16



Gambar 2 Tahapan klasifikasi

masing-masing lapisan yang diperoleh berada pada rentang nilai positif. Keluaran dari fungsi aktivasi ReLU dapat dihitung menggunakan persamaan (1) [19].

$$f_{ReLU} = \max(0, x), x = \text{nilai input} \tag{1}$$

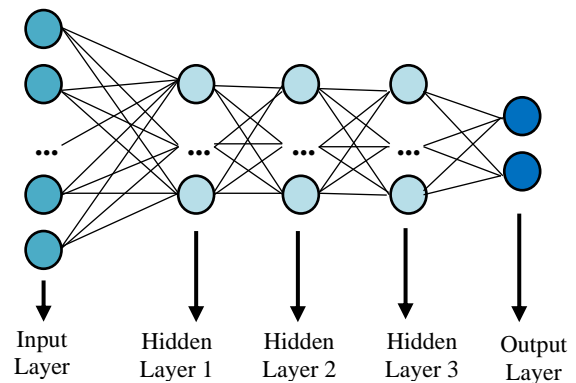
Fungsi optimasi Adam dan SGD pada proses klasifikasi berfungsi untuk mencari dan memperbaharui nilai bobot dan bias pada proses *update* bobot dan bias. Fungsi optimasi Adam menghitung nilai bobot dan bias baru dengan menggunakan persamaan (2), sedangkan fungsi optimasi SGD menghitung nilai bobot dan bias baru dengan menggunakan persamaan (3) [20].

$$q_{t+1} = q_t - \frac{h}{\sqrt{\hat{u}_t + \epsilon}} \hat{a}_{t,ac} \tag{2}$$

$$q_{t+1} = q_t - hdt \tag{3}$$

Hasil klasifikasi masing-masing model dievaluasi menggunakan *10-fold cross validation* untuk menghasilkan nilai *accuracy*, *precision*, dan *recall*. Nilai-nilai ini diperoleh dengan cara menghitung perbandingan nilai *true positive* (TP), *false positive* (FP), *false negative* (FN), dan *true negative* (TN) yang diperoleh dari tabel *confusion matrix*, menggunakan persamaan (4) sampai (6) [21].

$$\text{Akurasi} = \frac{TP + TN}{TP + FP + FN + TN} \cdot 100\% \tag{4}$$



Gambar 3 Arsitektur MLP

TABEL I  
KONFIGURASI MODEL MLP

Model	Neuron Hidden Layer	Fungsi Aktivasi	Fungsi Optimasi
ReLU - Adam	50-50-50	ReLU	Adam
ReLU - Adam2	100-100-100	ReLU	Adam
ReLU - Adam3	150-150-150	ReLU	Adam
ReLU - SGD	50-50-50	ReLU	SGD
ReLU - SGD2	100-100-100	ReLU	SGD
ReLU - SGD3	150-150-150	ReLU	SGD

$$Presisi = \frac{TP}{TP + FP} \cdot 100\% \quad (5)$$

$$Sensitivitas = \frac{TP}{TP + FN} \cdot 100\% \quad (6)$$

Aplikasi *Orange 3 Data Mining* digunakan untuk membangun model klasifikasi dengan memanfaatkan *widget* yang disediakan pada aplikasi tersebut, seperti:

1) *Widget Import Images*

*Widget* ini digunakan dengan dua nama yang berbeda, yaitu Gambar Latih dan Gambar Uji. Gambar Latih berfungsi untuk membaca 320 citra latih yang disimpan di dalam sebuah *folder*, sedangkan Gambar Uji berfungsi untuk membaca 80 citra uji yang disimpan di dalam *folder* yang lain. Gambar 4 menunjukkan sampel citra wajah yang menggunakan *face mask* dan Gambar 5 menunjukkan sampel citra wajah tanpa menggunakan *face mask* yang digunakan dalam penelitian ini.

2) *Widget Image Embedding*

*Widget* ini digunakan dengan nama Ekstrak Fitur. Ekstrak Fitur berfungsi untuk mengekstrak fitur-fitur dari citra latih melalui Gambar Latih dan citra uji melalui Gambar Uji dengan menggunakan *pre-trained CNN model VGG-16*.



Gambar 4 Sampel citra wajah menggunakan *face mask*



Gambar 5 Sampel citra wajah tanpa *face mask*

Gambar 6 menunjukkan konfigurasi *widget Image Embedding* yang digunakan dalam proses ekstraksi fitur.

3) *Widget Feature Statistics*

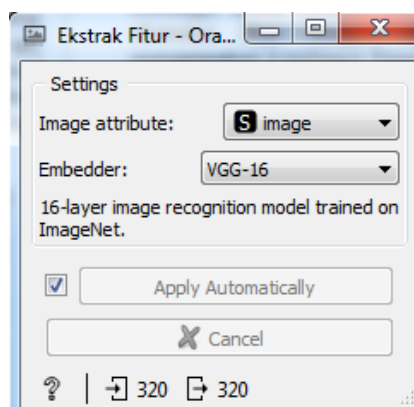
*Widget* ini digunakan dengan nama Statistik Fitur. Statistik Fitur berfungsi untuk mengolah dan menghitung statistik hasil ekstraksi fitur dari Ekstrak Fitur untuk menghasilkan nilai *mean*, *median*, dan *dispersion*.

4) *Widget Data Table*

*Widget* ini digunakan dengan dua nama yang berbeda, yaitu Data Fitur dan Data Statistik. Data Fitur digunakan untuk menampung data hasil ekstraksi fitur dari *widget Image Embedding*, sedangkan Data Statistik digunakan untuk menampung data statistik hasil ekstraksi fitur dari *widget Feature Statistic*.

5) *Widget Neural Network*

*Widget* ini digunakan dengan enam nama yang berbeda, yaitu ReLu-Adam, ReLu-Adam2, ReLu-Adam3, ReLu-SGD, ReLu-SGD2, dan ReLu-SGD3. *Widget* ini diperlihatkan pada Gambar 7 sampai Gambar 12, sedangkan model klasifikasi yang digunakan dalam penelitian ini diperlihatkan pada Gambar 13. ReLu-Adam berfungsi untuk melatih dan menguji *dataset* hasil ekstraksi fitur dengan algoritme MLP yang menggunakan kombinasi fungsi aktivasi ReLu dengan fungsi optimasi Adam, masing-masing *neuron* di *hidden layer* berjumlah 50. ReLu-Adam2 berfungsi untuk melatih dan menguji *dataset* hasil ekstraksi fitur dengan algoritme MLP yang menggunakan kombinasi fungsi aktivasi ReLu dengan fungsi optimasi Adam, masing-masing *neuron* di *hidden layer* berjumlah 100. ReLu-Adam3 berfungsi untuk melatih dan menguji *dataset* hasil ekstraksi fitur dengan algoritme MLP yang menggunakan kombinasi fungsi aktivasi ReLu dengan fungsi optimasi Adam, masing-masing *neuron* di *hidden layer* berjumlah 150. ReLu-SGD berfungsi untuk melatih dan menguji *dataset* hasil ekstraksi fitur dengan algoritme MLP yang menggunakan kombinasi fungsi aktivasi ReLu dengan fungsi optimasi SGD, masing-masing *neuron* di *hidden layer* berjumlah 50. ReLu-SGD2 berfungsi untuk melatih dan menguji *dataset* hasil ekstraksi fitur dengan algoritme MLP yang menggunakan kombinasi fungsi aktivasi ReLu dengan fungsi optimasi SGD, masing-masing *neuron* di *hidden layer*

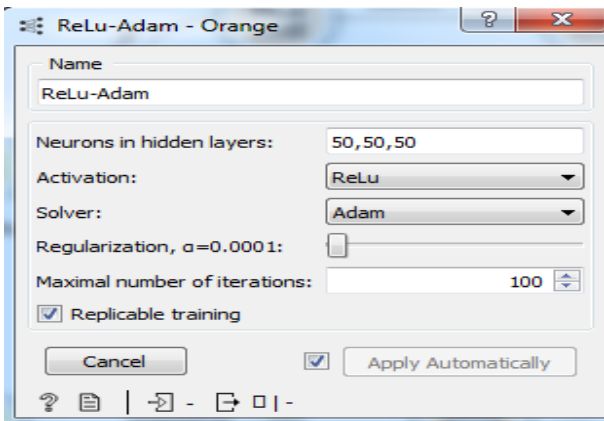


Gambar 6 *Widget Image Embedding*

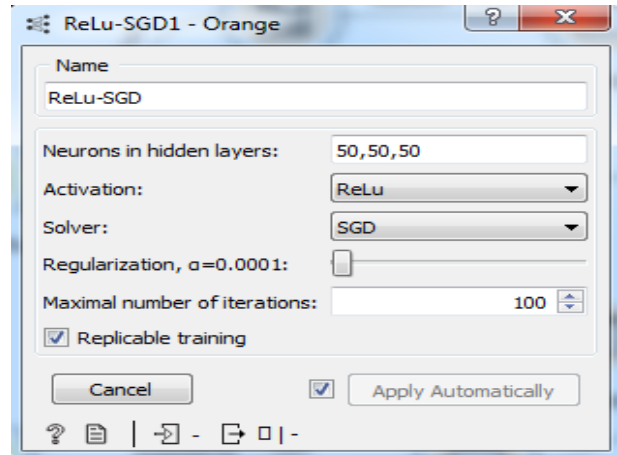
berjumlah 100. ReLu-SGD3 berfungsi untuk melatih dan menguji *dataset* hasil ekstraksi fitur dengan algoritme MLP yang menggunakan kombinasi fungsi aktivasi ReLu dengan fungsi optimasi SGD, masing-masing *neuron* di *hidden layer* berjumlah 150.

6) *Widget Test and Score*

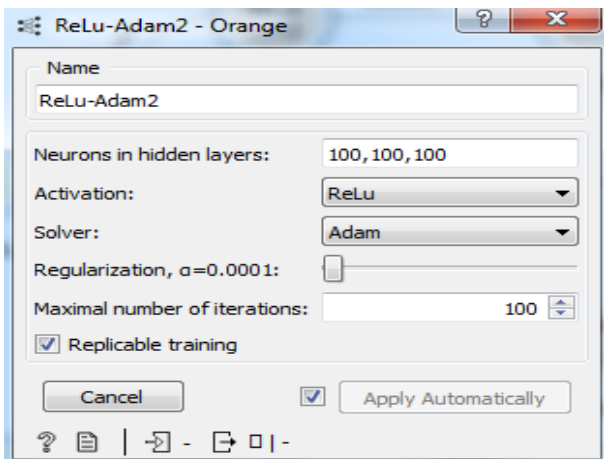
*Widget* ini digunakan untuk mengevaluasi hasil klasifikasi masing-masing model dengan menggunakan *10-fold cross validation*.



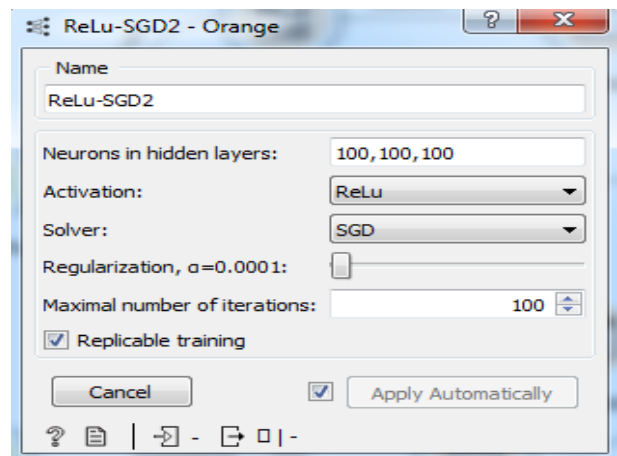
Gambar 7 ReLu-Adam



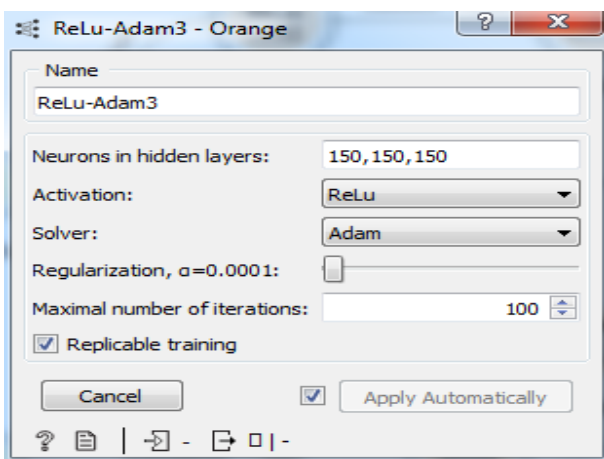
Gambar 10 ReLu-SGD



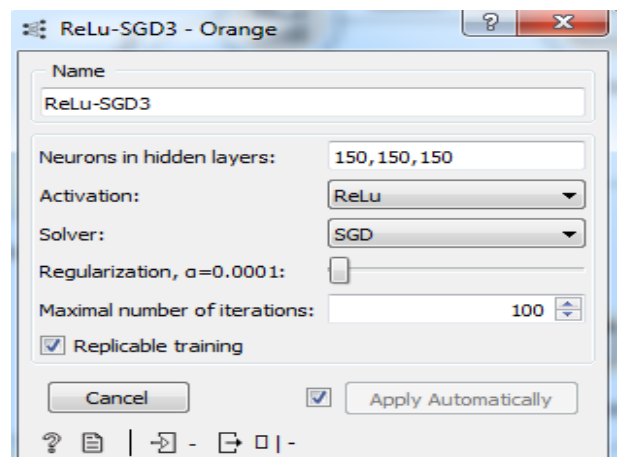
Gambar 8 ReLu-Adam2



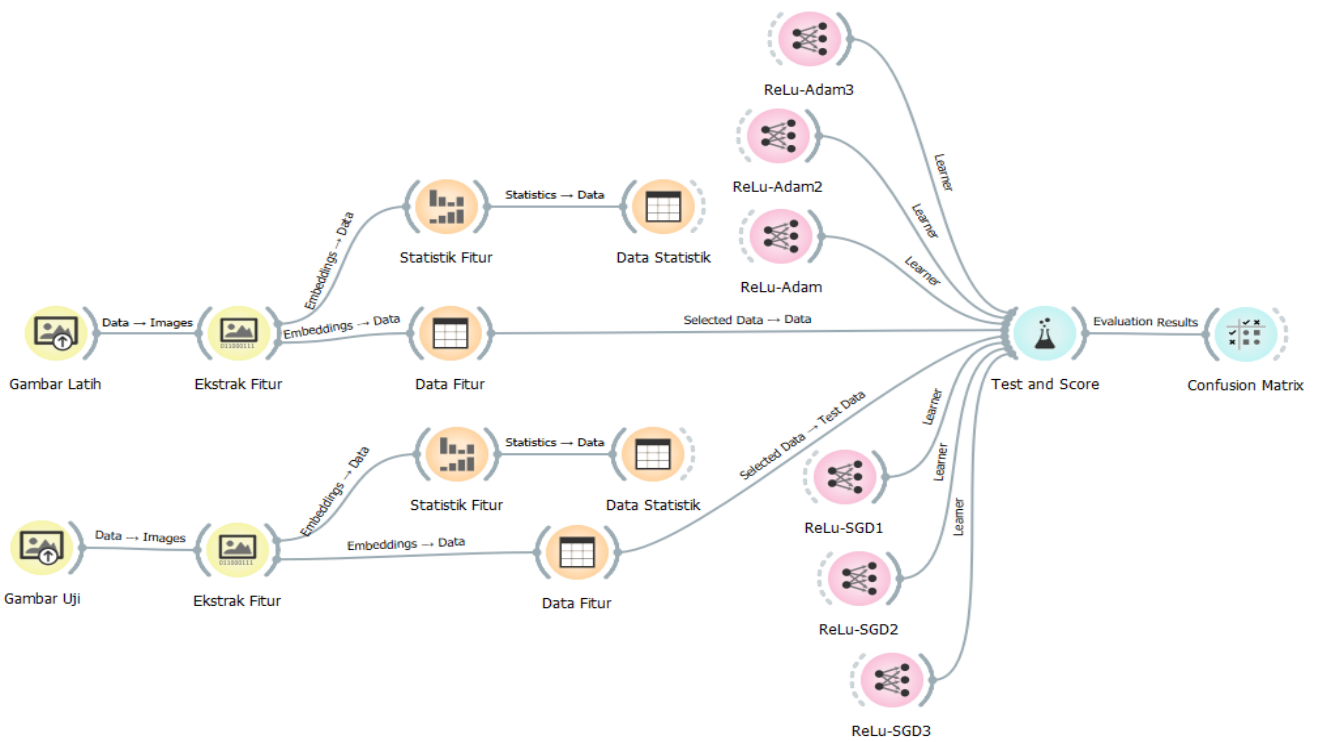
Gambar 11 ReLu-SGD2



Gambar 9 ReLu-Adam3



Gambar 12 ReLu-SGD3



Gambar 13 Model klasifikasi

7) *Widget Confusion Matrix*

*Widget* ini digunakan untuk menampilkan nilai-nilai *true positive* (TP), *false positive* (FP), *false negative* (FN), dan *true negative* (TN) dari hasil evaluasi *widget* Test and Score. Ditampilkan dalam bentuk tabel-tabel berdasarkan hasil dari masing-masing model.

III. HASIL DAN PEMBAHASAN

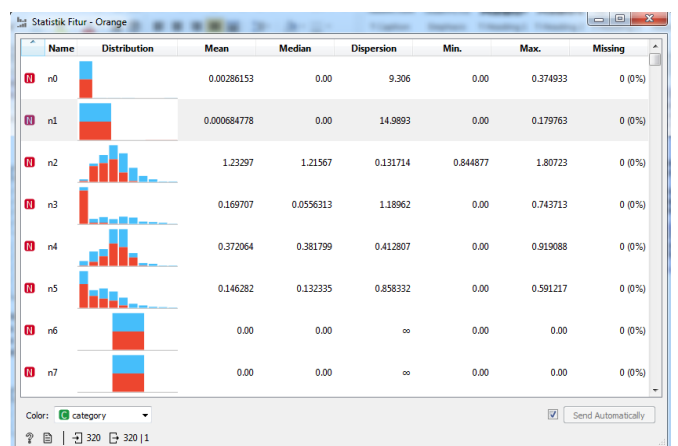
VGG-16 digunakan untuk mengekstrak fitur dari masing-masing *dataset*, data latih, data uji dan data validasi, sehingga menghasilkan 4096 fitur ( $n_0$  sampai  $n_{4095}$ ) dan dua kategori kelas (*mask* dan *no mask*). Tabel II menunjukkan hasil ekstraksi data latih dan Tabel III menunjukkan hasil ekstraksi fitur data uji.

TABEL II  
EKSTRAKSI FITUR DATA LATIH

n0	n1	n2	...	n4095	Kategori
0	0	0,909504	...	0,999069	Mask
0	0	0,897519	...	0,61016	Mask
0	0	1,18554	...	0,966277	Mask
0	0	1,18234	...	0,912343	Mask
0	0	1,4022	...	1,09213	Mask
...	...	...	...	...	...
0	0	1,28811	...	1,15537	No Mask
0	0	1,18393	...	1,03433	No Mask
0	0	1,25714	...	0,953759	No Mask
0	0	1,045	...	1,13569	No Mask
0	0	1,10799	...	1,10935	No Mask

Hasil ekstraksi fitur pada Tabel II kemudian dianalisis statistiknya dalam bentuk nilai *mean*, *median*, *dispersion*, nilai terendah, dan nilai tertinggi, seperti ditunjukkan pada Gambar 14. Nilai fitur  $n_0$  sampai  $n_{4095}$  dari ekstraksi fitur data latih ini kemudian digunakan sebagai data latih dalam proses klasifikasi menggunakan MLP.

Dari statistik fitur data latih diperoleh hasil bahwa untuk nilai *mean*, nilai terendah yang dihasilkan adalah 0, nilai tertinggi adalah 3,43949, dan nilai rata-ratanya adalah 0,457847437. Untuk nilai *median*, nilai terendah yang dihasilkan adalah 0, nilai tertinggi adalah 3,42641, dan nilai rata-ratanya adalah 0,445705392. Untuk nilai *dispersion*, nilai



Gambar 14 Statistik fitur data latih

terendah yang dihasilkan adalah 0,0432234, nilai tertinggi adalah 0,0432234, dan nilai rata-ratanya adalah 0,445705392.

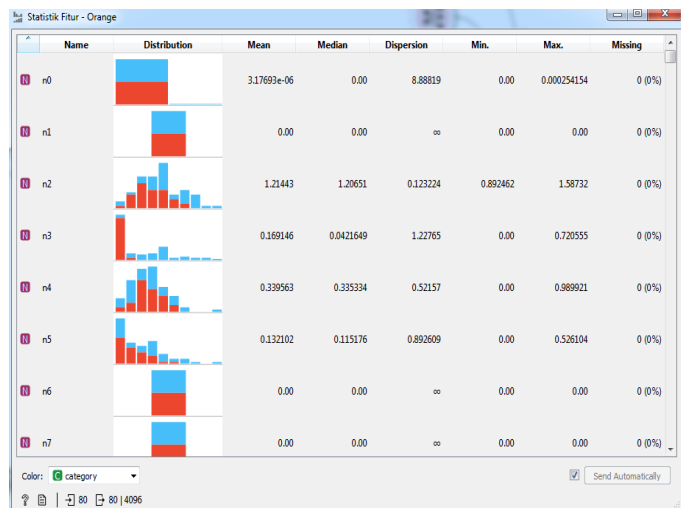
Hasil ekstraksi fitur pada Tabel III kemudian dianalisis statistiknya dalam bentuk nilai *mean*, *median*, *dispersion*, nilai terendah, dan nilai tertinggi, seperti ditunjukkan pada Gambar 15. Nilai fitur *n0* sampai *n4095* dari ekstraksi fitur data uji ini kemudian digunakan sebagai data uji dalam proses klasifikasi menggunakan MLP.

Dari statistik fitur data uji diperoleh hasil bahwa untuk nilai *mean*, nilai terendah yang dihasilkan adalah 0, nilai tertinggi adalah 3,46484, dan nilai rata-ratanya adalah 0,456251858. Untuk nilai *median*, nilai terendah yang dihasilkan adalah 0, nilai tertinggi adalah 3,48626, dan nilai rata-ratanya adalah 0,445736628. Untuk nilai *dispersion*, nilai terendah yang dihasilkan adalah 0,0436628, nilai tertinggi adalah 8,88819, dan nilai rata-ratanya adalah 1,381051598. Tabel IV memperlihatkan hasil perbandingan statistik fitur.

Setelah mengolah *dataset* menggunakan model klasifikasi yang dirancang, diperoleh hasil berupa hasil prediksi yang

TABEL III  
EKSTRAKSI FITUR DATA UJI

<i>n0</i>	<i>n1</i>	<i>n2</i>	...	<i>n4095</i>	Kategori
0	0	1,37636	...	0,943135	Mask
0	0	1,22626	...	0,729596	Mask
0	0	1,36415	...	0,531679	Mask
0	0	1,47814	...	1,1783	Mask
0	0	1,2277	...	0,774282	Mask
...	...	...	...	...	...
0	0	1,23839	...	0,944285	No Mask
0	0	0,962568	...	0,87716	No Mask
0	0	1,09875	...	1,04549	No Mask
0	0	1,10751	...	0,76216	No Mask
0	0	1,11193	...	0,929184	No Mask



Gambar 15 Statistik fitur data uji

tepat dan hasil prediksi yang tidak tepat untuk masing-masing model, dalam bentuk sebuah tabel *confusion matrix* (Tabel V). Dengan menggunakan persamaan (4) sampai (6), nilai-nilai pada Tabel VI diolah untuk menghasilkan nilai *accuracy*, *presisi*, dan *recall* masing-masing model.

TABEL IV  
KOMPARASI STATISTIK FITUR

Data	Terendah	Tertinggi	Rata-rata
Mean Data Latih	0	3,43949	0,457847437
Median Data Latih	0	3,42641	0,445705392
Dispersion Data Latih	0,0432234	1,20651	0,445705392
Mean Data Uji	0	3,46484	0,456251858
Median Data Uji	0	3,48626	0,445736628
Dispersion Data Uji	0,0436628	8,88819	1,381051598

TABEL V  
HASIL *CONFUSION MATRIX*

		Model	
		Predicted	
Actual	Mask	Mask	No Mask
		ReLu-Adam	157
Actual	No Mask	3	157
		ReLu-Adam2	156
Actual	Mask	3	157
		ReLu-Adam3	155
Actual	No Mask	4	156
		ReLu-SGD	156
Actual	Mask	6	154
		ReLu-SGD2	156
Actual	No Mask	3	157
		ReLu-SGD3	155
Actual	Mask	4	156
		ReLu-SGD3	155
Actual	No Mask	4	156

TABEL VI  
HASIL EVALUASI KLASIFIKASI

Data	Accuracy	Precision	Recall
Model RBF-Adam	98,1	98,1	98,1
Model RBF-Adam2	97,8	98,1	97,5
Model RBF-Adam3	97,2	97,5	96,9
Model RBF-SGD	96,9	96,3	97,5
Model RBF-SGD2	97,8	98,1	97,5
Model RBF-SGD3	97,2	97,5	96,9

Pada Tabel VI terlihat bahwa model RBF-Adam menghasilkan nilai *accuracy*, *precision*, dan *recall* sebesar 98,1%. Model RBF-Adam2 menghasilkan nilai *accuracy* sebesar 97,8%, nilai *precision* sebesar 98,1%, dan nilai *recall* sebesar 97,5%. Model RBF-Adam3 menghasilkan nilai *accuracy* sebesar 97,2%, nilai *precision* sebesar 97,5%, dan nilai *recall* sebesar 96,9%. Model RBF-SGD menghasilkan nilai *accuracy* sebesar 96,9%, nilai *precision* sebesar 96,3%, dan nilai *recall* sebesar 97,5%. Model RBF-SGD2 menghasilkan nilai *accuracy* sebesar 97,8%, nilai *precision* sebesar 98,1%, dan nilai *recall* sebesar 97,5%. Model RBF-SGD3 menghasilkan nilai *accuracy* sebesar 97,2%, nilai *precision* sebesar 97,5%, dan nilai *recall* sebesar 96,9%.

#### IV. SIMPULAN

Hasil komparasi statistik ekstraksi fitur data latih dan data uji menunjukkan rata-rata nilai yang tidak berbeda jauh. Namun, *dataset* uji lebih heterogen dibandingkan dengan *dataset* latih jika dilihat dari rata-rata *dispersion* data latih (0,445705392) dan data uji (1,381051598). Dari hasil evaluasi klasifikasi dapat disimpulkan bahwa kombinasi fungsi aktivasi ReLu dengan fungsi optimasi Adam yang dirancang menggunakan tiga *hidden layer* dengan konfigurasi 50-50-50 *neuron* menghasilkan model terbaik. Nilai *accuracy* sebesar 98,1%, *precision* sebesar 98,1%, dan *recall* sebesar 98,1% membuktikan bahwa model tersebut memiliki performa yang lebih unggul dibandingkan model lainnya. Dari hasil evaluasi klasifikasi juga dapat disimpulkan bahwa jumlah *neuron* yang semakin banyak pada masing-masing *hidden layer* tidak mempengaruhi peningkatan performa klasifikasi. Hal ini terlihat dari model dengan kombinasi fungsi aktivasi RBF dan fungsi optimasi Adam, performa terbaik diperoleh pada model dengan jumlah *neuron* 50. Sementara pada model dengan kombinasi fungsi aktivasi RBF dan fungsi optimasi SGD, performa terbaik diperoleh pada model dengan jumlah *neuron* 100.

Penelitian ini berhasil membuktikan kelayakan kombinasi *pre-trained CNN model* VGG-16 dengan MLP untuk mengklasifikasi citra wajah berdasarkan hasil ekstraksi fitur. Untuk pengembangan selanjutnya dapat dibandingkan dengan *pre-trained CNN model* lainnya, seperti *SquzeeNet* atau Inception V3, atau dikombinasikan dengan algoritme *machine learning* lain, seperti *support vector machine* (SVM).

#### DAFTAR REFERENSI

[1] C. K. Poetra, S. F. Pane, dan R. N. S. Fatonah, "Meningkatkan akurasi *long-short term memory* (LSTM) pada analisis sentimen vaksin Covid-19 di Twitter dengan Glove," *J. Telemat.*, vol. 16, no. 2, hlm. 85–90, 2021.

[2] L. Martinelli, dkk., "Face masks during the Covid-19 pandemic: a simple protection tool with many meanings," *Front. Public Heal.*, vol. 8, Januari, hlm. 1-12, 2021.

[3] F. Amer, M. Ali, dan M. S. H. Al-Tamimi, "Face mask detection methods and techniques: a review," *Int. J. Nonlinear Anal. Appl.*, vol. 13, Februari, hlm. 2008–6822, 2022.

[4] G. NaliniPriya, M. Shobana, C. Siva, B. Kanisha, J. K. Monica, dan V. Siva Vadivu Ragavi, "Dynamic face mask detection using machine learning," dalam *2022 International Conference on Smart*

*Technologies and Systems for Next Generation Computing (ICSTSN)*, Juni, 2022, hlm. 1-5.

[5] M. F. Naufal dan S. F. Kusuma, "Pendeteksi citra masker wajah menggunakan CNN dan *transfer learning*," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 6, hlm. 1293, 2021.

[6] P. R. Togatorop dan A. Fauzi, "Classification of face mask usage using Squeezenet," *Jatiji (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 1, hlm. 397–406, 2022.

[7] R. Rismiyati dan A. Luthfiarta, "VGG-16 transfer learning architecture for salak fruit quality classification," *Telematika*, vol. 18, no. 1, hlm. 37, 2021.

[8] S. Tammina, "Transfer learning using VGG-16 with deep convolutional neural network for classifying images," *Int. J. Sci. Res. Publ.*, vol. 9, no. 10, hlm. 143-150, 2019.

[9] M. Kurniawan, M. Hakimah, dan S. Agustini, "Perbandingan SVM dan *perceptron* dengan optimasi heuristik," *J. Telemat.*, vol. 15, no. 2, hlm. 85–92, 2020.

[10] M. Handayani, M. Riandini, dan Z. Zakarias, "Comparison of neural network optimization functions in candidate husband eligibility classification," *J. Inform.*, vol. 9, no. 1, hlm. 78–84, 2022.

[11] D. Pardede, B. H. Hayadi, dan Iskandar, "Multilayer perceptron literature review how well this algorithm performs," *J. ICT Apl. Syst.*, vol. 1, no. 1, hlm. 23–35, 2022.

[12] E. Pirdia Wanti, "Pengidentifikasian citra ikan berformalin dengan menggunakan metode *multilayer perceptron*," *J. Sains Komput. Inform.*, vol. 5, no. 1, hlm. 491–502, 2021.

[13] S. Shadkani, A. Abbaspour, S. Samadianfar, S. Hashemi, A. Mosavi, dan S. S. Band, "Comparative study of multilayer perceptron-stochastic gradient descent and gradient boosted trees for predicting daily suspended sediment load: the case study of the Mississippi River, U.S.," *Int. J. Sediment Res.*, vol. 36, no. 4, hlm. 512–523, 2021.

[14] M. A. Firmansyah, K. N. Ramadhani, dan A. Arifianto, "Pengenalan angka tulisan tangan menggunakan *diagonal feature extraction* dan klasifikasi *artificial neural network multilayer perceptron*," *Indones. J. Comput.*, vol. 3, no. 1, hlm. 65–74, 2018.

[15] J. Pardede, B. Sitohang, S. Akbar, dan M. L. Khodra, "Implementation of transfer learning using VGG-16 on fruit ripeness detection," *Int. J. Intell. Syst. Appl.*, vol. 13, no. 2, hlm. 52–61, 2021.

[16] J. N. Mogan, C. P. Lee, K. M. Lim, dan K. S. Muthu, "VGG-16 - MLP: Gait recognition with fine-tuned VGG-16 and multilayer perceptron," *Appl. Sci.*, vol. 12, no. 15, hlm. 1–12, 2022.

[17] Y. Gultom, A. M. Arymurthy, dan R. J. Masikome, "Batik classification using deep convolutional network transfer learning," *J. Ilmu Komput. dan Inf.*, vol. 11, no. 2, hlm. 189-194, 2018.

[18] O. Gurav, "Face mask detection dataset," 2020. [Daring]. Tersedia: <https://www.kaggle.com/datasets/omkargurav/face-mask-dataset>.

[19] K. L. Kohsasih, M. Dipo, A. Rizky, T. Fahriyani, V. Wijaya, dan R. Rosnelly, "Analisis perbandingan algoritme *convolutional neural network* dan algoritme *multilayer perceptron neural* dalam klasifikasi citra sampah," *J. Times*, vol. 10, no. 2, hlm. 22–28, 2022.

[20] D. Irfan, R. Rosnelly, M. Wahyuni, J. T. Samudra, dan A. Rangga, "Comparison of SGD, Adadelta, and Adam optimization in hydrangea classification using CNN," *J. Sci. Soc. Res.*, vol. 5, no. 2, hlm. 244-253, 2022.

[21] J. Kusuma, B. H. Hayadi, dan R. Rosnelly, "Comparison of multilayer perceptron (MLP) and support vector machine (SVM) methods for breast cancer classification," *MIND (Multimedia Artif. Intell. Netw. Database) J.*, vol. 7, no. 1, hlm. 51–60, 2022.

**Khairul Fadhli Margolang**, lahir di kota Ajamu. Menempuh pendidikan S1 di STMIK Triguna Dharma. Saat artikel ini diterbitkan sedang menempuh pendidikan S2 Ilmu Komputer di Universitas Potensi Utama. Penulis tertarik pada bidang penelitian seputar *machine learning*, *deep learning*, dan *computer vision*.  
**Sugeng Riyadi**, lahir di Cilacap. Menempuh pendidikan S1 di STMIK Triguna Darma Medan. Saat artikel ini diterbitkan sedang menempuh pendidikan S2 Ilmu Komputer di Universitas Potensi Utama. Penulis tertarik pada bidang *machine learning*, *deep learning*, dan *computer vision*.

**Rika Rosnelly**, lahir di kota Medan, Menempuh pendidikan S3 di Universitas Gajah Mada. Aktif sebagai dosen pengajar Fakultas Teknik dan Ilmu Komputer di Universitas Potensi Utama. Penulis tertarik pada bidang *machine learning*, *deep learning*, dan *computer vision*.

**Wanayumini**, lahir di kota Medan, Menempuh pendidikan S3 di Universitas Sumatera Utara. Saat ini aktif sebagai dosen pengajar Fakultas Teknik dan Ilmu Komputer di Universitas Potensi Utama. Penulis tertarik pada bidang penelitian *machine learning*, *deep learning*, dan *computer vision*.