

Rancang Bangun Aplikasi Android Deteksi Penggunaan Masker Wajah Menggunakan Tensorflow Mobilenet

Yusuf Arief Wicaksono, Sam Farisa Chaerul Haviana, Andi Riansyah

Fakultas Teknologi Industri, Prodi Teknik Informatika Universitas Islam Sultan Agung

Correspondence Author: alice@std.unissula.ac.id

Abstract

CoronaVirus Disease 2019 (COVID-19) merupakan sebuah penyakit menular yang disebabkan oleh virus SARS-CoV-2 yang berpotensi menimbulkan kematian. Menanggapi penyebaran COVID-19 banyak kantor memilih untuk melaksanakan kegiatan bekerja secara online atau lebih dikenal dengan istilah *working from home* (WFH). Setelah melaksanakan pembelajaran berbasis daring selama 21 bulan, Fakultas Teknologi Industri Universitas Islam Sultan Agung (FTI UNISSULA) kini tengah mempersiapkan pelaksanaan kegiatan kependidikan secara luring namun terbatas. Berkaitan dengan ini, Dekan FTI UNISSULA mengeluarkan Keputusan Dekan Fakultas Teknologi Industri UNISSULA Nomor 852/A/SA-TI/X/2021 tentang Panduan Pembelajaran Luar Jaringan (Luring) Terbatas Di Fakultas Teknologi Industri Universitas Islam Sultan Agung (UNISSULA) Tahun Akademik 2021-2022. Panduan ini berisi penjelasan mengenai Prosedur Protokol Kesehatan yang wajib ditaati oleh seluruh warga FTI. Salah satu langkah pencegahan COVID yang disarankan dalam Panduan tersebut yaitu penggunaan masker wajah pada saat melakukan kegiatan. Namun prosedur ini cenderung dilupakan dan seringkali tidak diindahkan. Untuk mengatasi persoalan tersebut melalui penelitian ini akan dibuat sebuah program aplikasi yang dapat secara otomatis mendeteksi penggunaan masker wajah pada civitas FTI UNISSULA dan memberikan peringatan apabila belum mengenakan masker.

Keyword: Deteksi Masker, Aplikasi Mobile, MobileNet, TensorFlow, Data Mining

1. PENDAHULUAN

CoronaVirus Disease 2019 (COVID-19) merupakan sebuah penyakit menular yang disebabkan oleh virus SARS-CoV-2. Gejala penyakit ini bermacam-macam, tetapi seringkali berupa demam, batuk, sakit kepala, kecapekan, kesulitan bernapas, dan hilangnya kemampuan mencium bau dan merasakan makanan. Penyakit ini berpotensi menimbulkan kematian. Menanggapi penyebaran COVID-19 banyak kantor dan fasilitas umum pada awal pandemi memilih untuk menghentikan sementara kegiatan aktifitas fisik di tempat, dengan sebagian kantor memilih untuk melaksanakan kegiatannya secara digital dengan memanfaatkan perangkat lunak komputer untuk bekerja secara *online* sehingga memungkinkan proses bisnis untuk dapat berjalan seperti biasa atau lebih dikenal dengan istilah *working from home* (WFH) [1].

Setelah melaksanakan pembelajaran berbasis daring selama 21 bulan, Fakultas Teknologi Industri Universitas Islam Sultan Agung (FTI UNISSULA) berdasarkan Surat Edaran Rektor UNISSULA Nomor 9518/J/SA/X/2021 tentang Aktifitas Kerja Tenaga Kependidikan Secara *Work from Office* kini tengah mempersiapkan pelaksanaan kegiatan kependidikan secara luring namun terbatas. Berkaitan dengan ini, Dekan FTI UNISSULA mengeluarkan Keputusan Dekan Fakultas Teknologi Industri UNISSULA Nomor 852/A/SA-TI/X/2021 tentang Panduan Pembelajaran Luar Jaringan (Luring) Terbatas Di Fakultas Teknologi Industri Universitas Islam Sultan Agung (UNISSULA) Tahun Akademik 2021-2022. Salah satu langkah pencegahan COVID yang disarankan dalam Panduan tersebut yaitu penggunaan masker wajah pada saat melakukan kegiatan. Langkah ini merupakan cara mudah untuk mengurangi resiko penyebaran virus melalui droplet air pada saat berbicara. Namun prosedur ini cenderung dilupakan dan seringkali tidak diindahkan sehingga berpotensi mengakibatkan hal yang tidak diinginkan. Untuk mengatasi persoalan tersebut melalui penelitian ini akan dibuat sebuah program aplikasi yang dapat mendeteksi penggunaan masker wajah pada civitas FTI UNISSULA dan memberikan peringatan apabila ada yang belum mengenakan masker.

Penelitian deteksi masker berbasis Android sebelumnya sudah pernah dilakukan oleh Nyoman Purnama, et al dari STMIK PRIMAKARA yang menggunakan MobileNet dan TensorFlow untuk melakukan identifikasi jenis masker. Model dilatih menggunakan *dataset* kumpulan gambar masker yang diperoleh melalui pengambilan gambar sampel secara langsung menggunakan smartphone maupun citra yang diambil dari internet yang dikelompokkan menjadi 3 kategori yaitu masker bedah/medis, masker N95, dan masker kain/scuba. Sistem yang dihasilkan mampu

mengidentifikasi secara realtime jenis masker yang digunakan dengan akurasi mencapai 90% pada jarak ideal 40 cm dengan kamera dan pencahayaan yang baik [2].

Penelitian ini dibuat menggunakan MobileNet yaitu sebuah model *Convolutional Neural Network* (CNN) yang didesain untuk dijalankan pada *platform mobile* dan *embedded applications*. CNN adalah sebuah algoritma yang dapat memberikan *importance* (berupa *weight* dan *bias*) ke *features* dalam gambar sehingga dapat dibedakan satu objek dari yang lain [3]. MobileNet merupakan model berbasis *Depthwise Separable Convolution* yang merupakan kombinasi dari 2 *convolutions*: *depthwise convolution* dan *pointwise convolution*. Penggunaan *depthwise separable convolutions* mengurangi secara signifikan jumlah *parameter* jika dibandingkan dengan *regular convolutions* pada *depth* yang sama [4]. MobileNet V1 terdiri dari 30 *layers*. *Layer* pertama merupakan sebuah *regular convolution*, kemudian diikuti dengan *depthwise* dan *pointwise layers* secara bergantian. Semua *convolutional layers* diikuti oleh *ReLU activation function* [5] dan sebuah *softmax function* [6]. MobileNet V2 merupakan peningkatan dari MobileNet V1 yang memiliki arsitektur yang hampir sama dengan tambahan dua fitur baru: *Inverted Residuals* dan *Linear Bottlenecks* [7]. *Residual blocks* menghubungkan bagian awal dan akhir dari *convolution* melalui sebuah jalur terkoneksi (disebut '*skip*') yang memungkinkan *network* untuk mengakses data yang terdapat pada tingkat atas *layer* yang belum tersentuh *convolution*. MobileNet V2 menggunakan struktur yang disebut *inverted residual blocks* karena merupakan kebalikan dari struktur *residual blocks* normal [8]. *Bottleneck Residual Block* terdiri dari tiga *convolution layers*: sebuah *expansion layer*, sebuah *depthwise convolution layer* yang sama seperti terdapat pada MobileNet V1, dan sebuah *1x1 pointwise convolution* yang disebut *projection layer* [9].

Melalui penelitian ini diharapkan dapat diciptakan sebuah prototipe aplikasi Android yang memanfaatkan model MobileNet untuk mendeteksi masker wajah yang dapat memberikan peringatan kepada orang yang belum mengenakan masker dengan tujuan dapat membantu mengawal pelaksanaan protokol kesehatan untuk mengurangi resiko penyebaran COVID-19 di lingkungan FTI UNISSULA.

2. METODE PENELITIAN

2.1 Pengembangan Model Deteksi Masker Berbasis MobileNet

Pemilihan model MobileNet untuk penggunaan pada aplikasi *face mask recognition* didasarkan pada *paper* MobileNet yang membandingkan kemampuan *object detection* MobileNet dengan beberapa model lain yaitu VGG dan Inception [10]. Percobaan *object detection* dilakukan menggunakan *dataset COCO Challenge* dan ditampilkan sebagai skor *mean Average Precision* (mAP) yang merupakan sebuah metrik populer untuk menguji performa model dalam melakukan operasi *information retrieval* dan *object detection*. Pembuatan model dilakukan dengan bantuan Google Colab dan *library* Keras. Model dibuat menggunakan metode *transfer learning* yaitu melatih ulang model yang sudah di-training sebelumnya dengan *dataset* yang diinginkan, dalam kasus ini *dataset* masker, dan melakukan *fine tuning* lebih lanjut pada model untuk meningkatkan akurasi.

2.2 Pengembangan Aplikasi Deteksi Masker

Aplikasi dibuat menggunakan *framework* Flutter dan berjalan pada perangkat berbasis Android. Aplikasi menggunakan kamera smartphone untuk mengambil tampak depan wajah kemudian menjalankan model MobileNet yang sudah dibuat untuk menganalisa gambar dan menampilkan peringatan apabila belum menggunakan masker. Tampilan rancangan UI aplikasi dapat dilihat pada gambar 1 dan berisi gambar wajah yang tertangkap kamera disertai hasil dari proses deteksi berupa *result*, *score*, *pre processing time*, *inference time*, dan *total prediction time*. *Result* menampilkan kesimpulan hasil deteksi model. *Score* berisi hasil prediksi model kemungkinan pengguna sedang menggunakan masker. *Pre processing time* menampilkan waktu yang dibutuhkan untuk menyesuaikan gambar agar bisa diterima model. *Inference time* merupakan waktu yang dibutuhkan model untuk menghitung probabilitas penggunaan masker. *Total prediction time* merupakan keseluruhan waktu proses untuk melakukan deteksi penggunaan masker.



Gambar 1. Rancangan UI Aplikasi

3. HASIL DAN ANALISA

3.1 Hasil

Pada gambar 2 merupakan tampilan dari aplikasi sudah yang dibuat sesuai dengan rancangan UI. Aplikasi menampilkan gambar wajah yang tertangkap kamera beserta hasil dari proses deteksi berupa *result*, *score*, *inference time*, *pre processing time*, dan *total prediction time*. Setelah model bekerja dan menghasilkan prediksi akan ditampilkan layar berisi hasil prediksi dan peringatan apabila pengguna belum mengenakan masker.



Gambar 2. Implementasi Rancangan Aplikasi

3.2 Pengujian

Pengujian sistem dilakukan menggunakan metode *black box testing* yaitu pengujian tanpa melihat cara kerja internal aplikasi dengan hanya memperhatikan *input* dan *output* sehingga dapat memberikan gambaran kinerja aplikasi di lapangan. Pengujian dilakukan sesuai cara kerja sistem yaitu dengan menganalisa hasil tangkapan tampak depan wajah dan prediksi yang dihasilkan model. Hasil prediksi sistem memiliki 4 kemungkinan keluaran [11] yaitu sebagai berikut:

1. *True Positive*, model mendeteksi masker pada pengguna dengan masker
(*system: with_mask, actual: with_mask*)
2. *False Positive*, model mendeteksi masker pada pengguna tanpa masker
(*system: with_mask, actual: without_mask*)
3. *True Negative*, model tidak mendeteksi masker pada pengguna tanpa masker
(*system: without_mask, actual: without_mask*)
4. *False Negative*, model tidak mendeteksi masker pada pengguna dengan masker
(*system: without_mask, actual: with_mask*)

Telah dilakukan pengujian pada 14 orang dengan masing-masing orang diuji pada kondisi dengan dan tanpa menggunakan masker sehingga menghasilkan total 28 percobaan. Pengujian model dilakukan pada ruangan dengan pencahayaan yang cukup dan menggunakan beberapa *Confidence Threshold* yang menentukan apakah sebuah keluaran *Positive* dianggap sebagai *True Positive* (jika nilai *confidence* lebih tinggi dari *threshold*) atau *False Positive* (jika nilai *confidence* lebih rendah dari *threshold*). Hasil pengujian sistem dapat dilihat pada tabel 1 di bawah ini.

Tabel 1. Hasil pengujian sistem

Percobaan ke	Confidence Score	System	Actual	Is Prediction Correct	Threshold				
					0.9	0.8	0.7	0.6	0.5
1	0.99	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
2	0.99	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
3	0.96	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
4	0.89	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	FP	TP	TP	TP	TP
5	0.64	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	FP	FP	FP	TP	TP
6	0.95	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
7	0.96	<i>without_mask</i>	<i>with_mask</i>	<i>False</i>	FN	FN	FN	FN	FN
8	0.99	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
9	0.96	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
10	0.99	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
11	0.97	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
12	0.86	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	FN	TN	TN	TN	TN
13	0.98	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
14	0.99	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
15	0.98	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
16	0.99	<i>with_mask</i>	<i>without_mask</i>	<i>False</i>	FP	FP	FP	FP	FP
17	0.99	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
18	0.99	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
19	0.86	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	FP	TP	TP	TP	TP
20	0.99	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
21	0.98	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
22	0.63	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	FN	FN	FN	TN	TN
23	0.94	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
24	0.94	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN
25	0.99	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
26	0.92	<i>with_mask</i>	<i>without_mask</i>	<i>False</i>	FP	FP	FP	FP	FP
27	0.98	<i>with_mask</i>	<i>with_mask</i>	<i>True</i>	TP	TP	TP	TP	TP
28	0.98	<i>without_mask</i>	<i>without_mask</i>	<i>True</i>	TN	TN	TN	TN	TN

3.3 Evaluasi

Dalam melakukan evaluasi model *machine learning* terdapat beberapa *metrics* yang dapat digunakan untuk menilai performa dan kebenaran model dalam melaksanakan fungsinya. Untuk model berupa *classifier* seperti model deteksi masker *metrics* yang digunakan untuk melakukan evaluasi [12] yaitu:

1. *Precision*, yaitu *metric* yang menggambarkan berapa persen prediksi *Positive* yang merupakan *True Positive*. *Precision* berguna jika *False Positive* merupakan hasil yang tidak diinginkan.
2. *Recall*, yaitu *metric* yang menggambarkan berapa persen kasus *Positive* yang berhasil diprediksi. *Recall* berguna jika *False Negative* merupakan hasil yang tidak diinginkan.

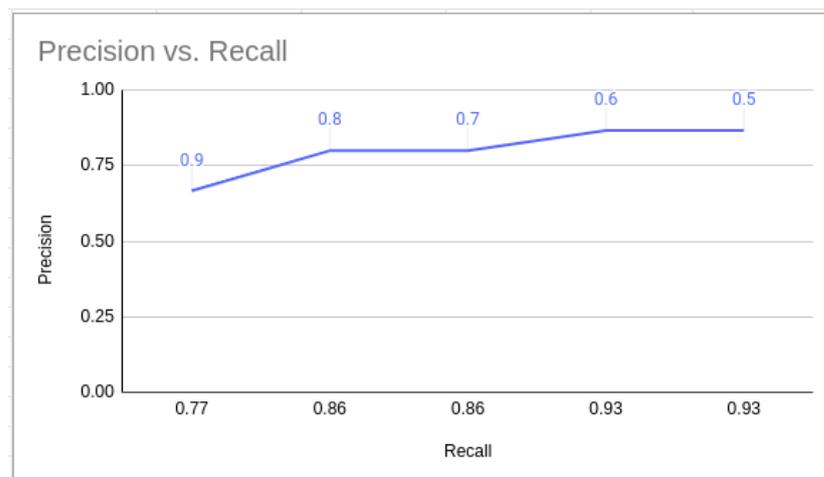
3. *Accuracy*, yaitu *metric* yang menggambarkan berapa persen kemungkinan model untuk menghasilkan prediksi benar.

Hasil percobaan di atas dapat direpresentasikan sebagai *metrics* evaluasi untuk setiap *threshold* yaitu seperti dapat dilihat pada tabel 2 di bawah ini.

Tabel 2. Evaluasi pengujian sistem

<i>Prediction</i>	<i>Threshold</i>				
	0.9	0.8	0.7	0.6	0.5
<i>True Positive</i>	10	12	12	13	13
<i>True Negative</i>	10	11	11	12	12
<i>False Positive</i>	5	3	3	2	2
<i>False Negative</i>	3	2	2	1	1
<i>Precision</i> $TP/(TP+FP)$	67%	80%	80%	87%	87%
<i>Recall</i> $TP/(TP+FN)$	77%	86%	86%	93%	93%
<i>Accuracy</i> $TP+TN/(TP+TN+FP+FN)$	71%	82%	82%	89%	89%

Evaluasi menunjukkan bahwa *threshold* lebih rendah menghasilkan prediksi yang lebih akurat. Karena model dikembangkan untuk membantu proses deteksi masker dan *False Positive* bukan merupakan resiko serius, maka aplikasi dapat secara aman menggunakan *threshold* rendah demi mendapatkan akurasi hasil prediksi yang lebih tinggi. Hubungan antara *Precision* dan *Recall* dapat dilihat pada gambar 3 di bawah ini.



Gambar 3. *Precision vs Recall*

Berdasarkan evaluasi hasil pengujian di atas dapat disimpulkan bahwa model berjalan dengan baik dan aplikasi dapat menjalankan modul MobileNet dan dapat melaksanakan fungsi deteksi penggunaan masker wajah berdasarkan hasil tangkapan foto tampak depan dengan cukup baik dengan akurasi mencapai 89%.

4. KESIMPULAN

Melalui penelitian ini dapat kita lihat sistem deteksi masker wajah menggunakan MobileNet pada perangkat Android dapat bekerja dengan baik dengan tingkat akurasi mencapai 89% pada ruangan dengan pencahayaan yang baik sehingga dapat disimpulkan bahwa sistem bekerja sesuai yang diharapkan.

ACKNOWLEDGEMENTS

Terima kasih kepada Fakultas Teknologi Industri Universitas Islam Sultan Agung atas dukungannya sehingga penelitian ini dapat berjalan dengan lancar. Paper ini merupakan tugas akhir mahasiswa teknik informatika dengan fokus menciptakan sebuah rancang bangun aplikasi deteksi masker wajah berbasis MobileNet untuk membantu proses penerapan prosedur pencegahan COVID-19 di lingkungan Fakultas Teknologi Industri Universitas Islam Sultan Agung.

DAFTAR PUSTAKA

- [1] O. Mungkasa, “Working from Home (WFH): Towards a New Order in the Era of the COVID-19 Pandemic,” *J. Perenc. Pembang. Indones. J. Dev. Plan.*, vol. 4, no. 2, pp. 126–150, 2020.
- [2] P. Nyoman and Putu Kusuma Negara, “Deteksi Masker Pencegahan Covid19 Menggunakan Convolutional Neural Network Berbasis Android,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 3, pp. 576–583, 2021, doi: 10.29207/resti.v5i3.3103.
- [3] Sumit Saha, “A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way,” 2018. <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53> (accessed Nov. 20, 2022).
- [4] Matthijs Hollemans, “Google’s MobileNets on the iPhone,” 2017. <https://machinethink.net/blog/googles-mobile-net-architecture-on-iphone/> (accessed Jun. 12, 2022).
- [5] SuperDataScience Team, “Convolutional Neural Networks (CNN): Step 1(b) - ReLU Layer,” 2018. <https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-1b-relu-layer> (accessed Nov. 20, 2022).
- [6] A. Bhardwaj, “What is the Softmax Function? — Teenager Explains,” 2020. <https://towardsdatascience.com/what-is-the-softmax-function-teenager-explains-65495eb64338> (accessed Nov. 20, 2022).
- [7] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.
- [8] Paul-Louis Pröve, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” 2018. <https://towardsdatascience.com/mobilenetv2-inverted-residuals-and-linear-bottlenecks-8a4362f4ffd5> (accessed Oct. 17, 2022).
- [9] Matthijs Hollemans, “MobileNet version 2,” 2018. <https://machinethink.net/blog/mobilenet-v2/> (accessed Oct. 17, 2022).
- [10] A. G. Howard *et al.*, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” Apr. 2017, [Online]. Available: <http://arxiv.org/abs/1704.04861>
- [11] Sumeet Kumar Agrawal, “Metrics to Evaluate your Classification Model to take the right decisions,” 2021. <https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/> (accessed Dec. 13, 2022).
- [12] Jonathan Grandperrin, “How to use confidence scores in machine learning models,” 2021. <https://mindee.com/blog/how-use-confidence-scores-ml-models/> (accessed Dec. 13, 2022).