

PROTOTIPE ALAT PENDETEKSI PENGGUNA MASKER SEBAGAI UPAYA PENCEGAHAN COVID-19

Arif Johar Taufiq¹, Latiful hayat², Susiyadi³

^{1,2}Program Studi S1 Teknik Elektro Fakultas Teknik dan Sains, Universitas Muhammadiyah Purwokerto

³Fakultas Kedokteran, Universitas Muhammadiyah Purwokerto

Informasi Makalah

Dikirim, 5 Agustus 2021
Direvisi, 25 Juli 2022
Diterima, 23 Maret 2023

Kata Kunci:

Detektor masker,
Google Teachable machine,
Covid-19

Keyword:

Mask detector,
Google Teachable machine,
Covid-19

INTISARI

Upaya pencegahan penyebaran virus Covid-19 salah satunya dapat dilakukan dengan cara memakai masker, terutama diruang publik. Pengawasan terhadap seseorang memakai masker atau tidak memakai masker sangat membosankan terutama jika orang yang diawasi sangat banyak misalnya di terminal, bandara dan sebagainya. Oleh karena itu diperlukan terobosan alat untuk membantu pengawasan terhadap seseorang memakai masker atau tidak memakai masker untuk mengurangi beban kerja pengawas, dan alat dapat bekerja 24 jam tanpa lelah. Implementasi detektor masker pada penelitian ini menggunakan cara cepat dengan memanfaatkan *google teachable machine*. Hasil pengujian alat dapat mendeteksi seseorang memakai masker atau tidak memakai masker dengan tingkat kebenaran 100% baik pada obyek riil maupun obyek berupa foto atau video di komputer.

ABSTRACT

One of the ways to prevent the spread of the Covid-19 virus is by wearing masks, especially in public spaces. Supervision of someone wearing a mask or not wearing a mask is very dull, especially if many people are being monitored, for example, in terminals, airports, etc. For this reason, a breakthrough tool is needed to help supervise someone wearing or not wearing a mask to reduce the supervisor's workload. The tool can work 24 hours without getting tired. The implementation of the mask detector in this study uses a fast method by utilizing the *google teachable machine*. From the testing results, the tool can detect someone wearing a mask or not wearing a mask with a 100% truth level both on real objects and objects in the form of photos or videos on a computer.

Korespondensi Penulis:

Arif Johar Taufiq
Program Studi Teknik Elektro
Universitas Muhammadiyah Purwokerto
JL. Raya Dukuwaluh Purwokerto, 53182
Email: arifjt@ump.ac.id

1. PENDAHULUAN

Penyebaran virus COVID-19 tergolong sangat cepat, penularan virus dari satu orang ke orang lain terjadi karena kontak fisik, terkena percikan cairan yang keluar dari hidung atau mulut penderita. Data WHO Oktober 2020 menyebutkan kasus terkonfirmasi penderita COVID-19 diseluruh dunia berjumlah 39.944.882 orang termasuk yang meninggal sejumlah 1.111.998 orang. Penyebaran yang cepat dan masif ini harus diupayakan segera mungkin dihentikan dengan cara *lock down*, *social distancing*, pola hidup bersih, cuci tangan, pemakaian *face shield*, pemakaian masker, dan sebagainya.

Salah satu upaya pencegahan virus ini adalah dengan menggunakan APD (Alat Pelindung Diri) sederhana berupa masker. Beberapa kantor layanan masyarakat: sekolah, perguruan tinggi, rumah ibadah, kantor bank, kantor layanan masyarakat lainnya mewajibkan untuk memakai masker disamping juga pengecekan suhu tubuh.

Studi kesehatan menyatakan penggunaan masker dapat menyaring penyebaran virus sampai batas tertentu saat di uji dengan partikel berukuran $\leq 0,072\text{m}$ dan hasilnya penyaringan akan lebih baik lagi jika menggunakan masker bedah [1]. Dalam *Journal of Breath* juga menyatakan alasan ilmiah mengapa pengguna masker sederhana dan penutup wajah dapat mengurangi penyebaran COVID-19 yang disebarkan melalui udara [2]. Juga di sebutkan dalam [3] penggunaan masker dapat mengurangi jumlah mikroba yang dikeluarkan oleh beberapa relawan kemanusiaan dan kemampuan masker bedah lebih baik tiga kali lipat dari pada masker biasa.

Tingkat pengetahuan masyarakat mempengaruhi kesadaran penggunaan masker secara mandiri, salah satu studi menyatakan kurangnya pengetahuan masyarakat tentang kesehatan berakibat dampak pada tingkat kepatuhannya orang memakai masker untuk mencegah penyebaran COVID-19 [4].

Penelitian tentang detektor masker antara lain dilakukan oleh [5] menggunakan modul Raspberry Pi, *webcame* dan sensor PIR. Metode yang dilakukan adalah mengintegrasikan semua modul dengan proses berdasarkan algoritma MobileNet V2, hasil yang didapat modul dapat mendeteksi wajah bermasker dan tidak bermasker tetapi ada kelemahan waktu pelatihan yang lama apalagi jika datanya berasal streaming video maka semakin lambat. Dalam penelitian ini akan dikaji pencegahan penyebaran COVID-19 dengan membuat alat detektor masker yang lebih praktis dan tingkat deteksi tinggi dengan proses pelatihan yang cepat dan langsung dapat diaplikasikan di *smartphone*. Harapannya alat dapat dipakai sebagai detektor masker dalam suatu ruang dengan satu pintu masuk. Tujuan yang ingin dicapai adalah: membuat prototipe alat pendeteksi masker untuk memudahkan mendeteksi seseorang memakai masker atau tidak secara otomatis.

Manfaat yang diharapkan adanya penelitian ini adalah membantu mengawasi orang yang tidak patuh pada aturan wajib memakai masker untuk pencegahan penyebaran Covid-19, alat yang dapat bekerja terus menerus tanpa banyak melibatkan pengawasan manusia yang cenderung jenuh dan lelah.

Computers Vision adalah bidang ilmiah interdisipliner yang membahas bagaimana komputer dapat memperoleh pemahaman tingkat tinggi dari citra atau video digital. Dari perspektif teknik, berusaha memahami dan mengotomatiskan tugas yang dapat dilakukan oleh sistem visual manusia [6]. Beberapa teori yang mendasari pada penelitian ini yaitu: *OpenCV*, *Keras/TensorFlow*, dan *Deep Learning*. *OpenCV* adalah pustaka fungsi pemrograman terutama ditujukan untuk *real time computer vision* [7]. *OpenCV* dikembangkan oleh Intel yang pada awalnya dikembangkan oleh Willow Garage diteruskan Itseez yang kemudian diakuisisi oleh Intel sampai sekarang. Pustaka ini bersifat lintas *platform* dan gratis dibawah Lisensi *open source* Apache 2. Mulai tahun 2011 *OpenCV* dilengkapi dengan fitur akselerasi GPU (*Graphics Processing Unit*) untuk operasi waktu nyata.

Keras adalah *open source library* yang menyediakan antarmuka bahasa pemrograman Python untuk Jaringan Syaraf Tiruan (JST). *Keras* bertindak sebagai antarmuka untuk pustaka *TensorFlow*. *Keras* berisi implementasi blok penyusun jaringan syaraf yang umum digunakan seperti lapisan, tujuan, fungsi aktivasi, pengoptimal, dan sejumlah *tools* agar dapat bekerja dengan data citra, teks serta lebih mudah untuk menyederhanakan pengkodean pada *deep neural network*.

Deep Learning adalah bagian dari keluarga metode *machine learning* yang lebih luas berdasarkan JST dengan pembelajaran representasi. Pembelajaran yang diawasi, semi diawasi atau tanpa pengawasan [8]. *Deep learning* telah diterapkan diberbagai bidang termasuk *computer vision*, *machine vision*, pengenalan suara, pemrosesan bahasa alami, pengenalan audio, pemfilteran media sosial, mesin penerjemah, bioinformatika, analisis citra medis, inspeksi material, yang telah memberikan hasil baik, dalam beberapa kasus melebihi kinerja dari seorang pakar [9].

JST biasanya hanya disebut *neural network*, adalah sistem komputasi yang diilhami oleh cara kerja sistem jaringan syaraf biologis [10]. JST didasarkan pada kumpulan unit atau node yang terhubung yang disebut *neuron*, merupakan model cara kerja *neuron* di otak biologis. Setiap koneksi, seperti sinapsis di otak biologis, dapat mengirimkan sinyal ke *neuron* lain. *Neuron* menerima sinyal kemudian memprosesnya dan memberikan sinyal kepada neuron yang terhubung dengannya.

Lebih detail pada bagian pelatihan dan klasifikasi dikembangkan dari jaringan syaraf tiruan menjadi CNN (*Convolution Neural Network*). CNN adalah salah satu jenis JST yang biasa digunakan pada data citra. CNN dapat digunakan untuk mendeteksi dan mengenali obyek pada sebuah citra. Secara garis besar CNN tidak jauh berbeda dengan JST. CNN terdiri dari *neuron* yang memiliki *weight*, *bias* dan *activation function* yang membedakan antara CNN dengan JST ialah arsitektur dari CNN dibagi menjadi dua bagian besar yaitu *Feature Extraction Layer* dan *Fully Connected Layer* [11].

Penelitian tentang deteksi wajah bermasker diusulkan oleh [12] dibangun dengan *deep learning InceptionV3*. Model yang diusulkan dilatih dan diuji pada *Simulated Masked Face Dataset (SMFD)*. Teknik *augmentasi* citra diadopsi untuk mengatasi ketersediaan data yang terbatas untuk pelatihan dan pengujian model yang lebih baik. Model tersebut mengungguli pendekatan lain dengan mencapai akurasi 99,9% selama pelatihan dan 100% selama pengujian dalam kekurangan pada penelitian ini adalah data uji hanya SMFD saja, bukan data riil orang yang memakai masker atau tidak memakai masker. Penelitian yang dilakukan [12],[13] membutuhkan waktu proses yang lama dan sedikit rumit serta data berupa citra tidak riil foto wajah bermasker,

penelitian yang diusulkan akan lebih sederhana dan cepat karena dikerjakan terintegrasi memanfaatkan *cloud computing google* sebagaimana terlihat pada Gambar 2.

2. METODE

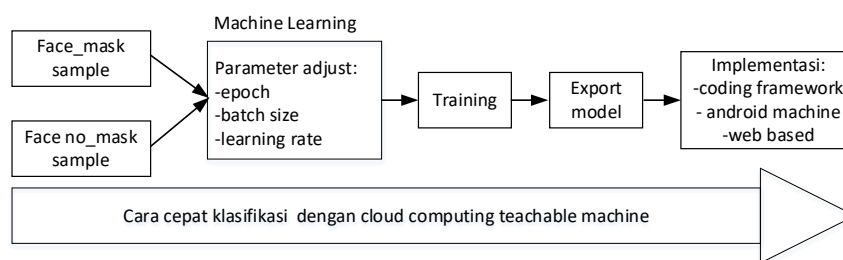
Langkah-langkah penelitian dilakukan sebagai berikut: (1) pengambilan sampel wajah bermasker dan tidak bermasker, (2) menyusun struktur JST, (3) pelatihan, (4) export model, (5) implementasi ke *smartphone* android atau program berbasis web, (6) serta pengujian. Mengatasi kendala dari penelitian sebelumnya maka pada penelitian ini menawarkan cara mendeteksi wajah masker dan tidak bermasker yang lebih sederhana dengan proses cepat, dapat mendeteksi obyek foto maupun video secara waktu nyata. Lebih sederhana dan waktu proses lebih cepat karena menggunakan *google cloud computing machine*. Hasil proses pelatihan dari *machine learning* berupa model yang dapat diimplementasikan ke platform yang kita inginkan sehingga proses kerja lebih sederhana dan efisien.

Sampel yang digunakan dalam penelitian ini adalah wajah bermasker dan tanpa masker. Citra diambil secara langsung melalui *webcame* pada laptop, sedang citra sample tidak langsung didapat dari file citra RWMF (*Real World Masked Face*) di <https://github.com/X-zhangyang/Real-World-Masked-Face-Dataset>. Contoh sample wajah bermasker RWMF seperti pada Gambar 1 ditambah sampel data riil perekaman citraA wajah lewat *webcame*.



Gambar 1. Contoh sampel wajah bermasker RWMF (a) dan sampel wajah perekaman riil (b)

Pada penelitian ini digunakan sampel data wajah bermasker sejumlah 5454, sedangkan wajah tidak bermasker sejumlah 4675. Proses implementasi cepat dipersingkat dengan adanya adanya *google teachable machine* lihat Gambar 2.



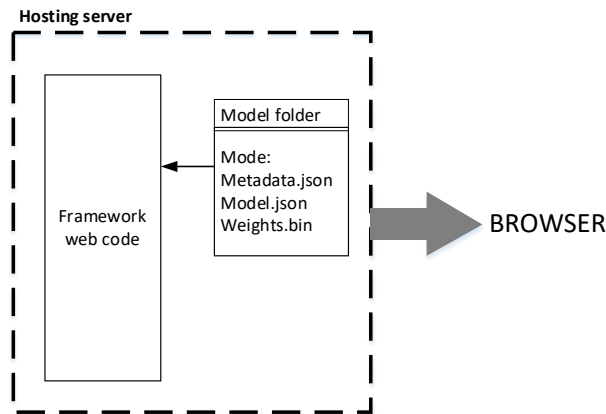
Gambar 2. Bagan cara cepat klasifikasi dengan goole teachable machine

Pada bagian *teachable machine* data yang harus dimasukkan nilai *Epoch*, *batch size* dan *learning rate*. Kombinasi nilai tersebut akan mempengaruhi hasil dari proses mendeteksi wajah bermasker atau tidak. Setelah proses pelatihan selesai maka didapat model *tensorflow* yang dapat diimplementasikan dengan program berbasis web misalnya *javascript*, program pada perangkat bergerak (misalnya android *smartphone*, *Iphone*, *dsb*) serta program *native tensorflow* (*python*). Pada penelitian ini model akan diimplementasikan pada program berbasis web dan perangkat bergerak *smartphone* android merk Samsung A20.

Langkah Implementasi Pada Web

Berdasarkan data dan pelatihan yang telah dilakukan maka model kita pilih berjenis *tensorflow.js*. File hasil *export model* jenis *tensorflow.js* yaitu: *metadata.json*, *model.json* dan *weights.bin*. File *weights.bin* file

biner berisi data bobot optimal hasil pelatihan, bobot ini berperan dalam mengatur klasifikasi citra atau video wajah memakai masker atau tidak. Skema implementasi detektor masker berbasis web seperti pada Gambar 3.



Gambar 3. Skema implementasi pada Web Hosting

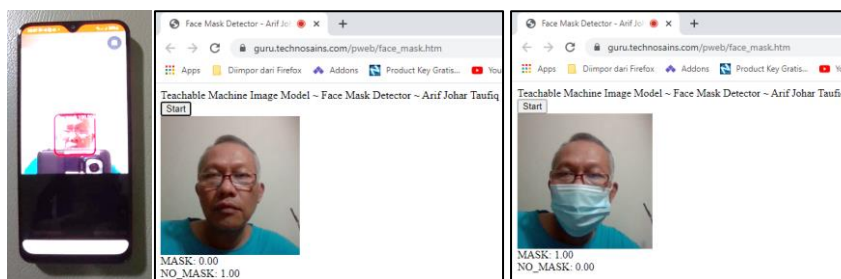
Langkah Implementasi Pada Mesin Android

Model yang dapat digunakan pada perangkat bergerak smartphone android yaitu *Tensorflow Lite*. Model *Tensorflow Lite* berisi file: *label.txt* dan *model.tflite* (dapat berjenis unquantized atau quantized). File *model.tflite* berisi data bobot optimal hasil pelatihan, sedangkan file *label.txt* berisi label kelas dari sampel.

Pertama yang dilakukan adalah siapkan *framework code example lite* dari alamat github <https://github.com/tensorflow/examples> buka dengan android studio, kemudian masukan file model hasil pelatihan ke direktori main/assets. Kemudian hubungkan smartphone ke laptop dengan kabel data, build gradle program detektor masker yang ada di android studio maka program aplikasi detektor masker akan terinstal di smartphone.

3. HASIL DAN PEMBAHASAN

Hasil implementasi program detektor masker pada perangkat bergerak smartphone android dan program berbasis web telah berhasil dibuat seperti pada Gambar 4.



Gambar 4. Tampilan program pada browser Chrome dan Hp Samsung A20

Detektor maskter yang berjalan berbasis web telah dapat bekerja dengan baik obyek, wajah orang dibaca melalui *webcame*, program detektor masker memprosesnya dan memberikan tanggapan berupa hasil klasifikasi MASK=0 dan NO_MASK=1 berarti orang tersebut tidak memakai masker. Jika hasil detektor adalah MASK=1 dan NO_MASK=0 maka wajah orang tersebut memakai masker. Data angka 0 dan 1 adalah derajat deteksi terhadap kelasnya, tingkatan klasifikasi antara 0 sampai 1 atau setara dengan 0% sampai 100%.

Pengujian Detektor Masker pada Web Browser

Pengujian dilakukan dengan cara menjalankan laptop yang sudah terkoneksi dengan internet kemudian membuka *browser* dan mengetikkan alamat URL dimana kode detektor masker telah dibuat dalam hal ini alamat URL nya adalah https://guru.technosains.com/pweb/face_mask.htm atau dapat juga diakses di alamat https://guru.technosains.com/pweb/face_mask.htm. Contoh hasil detektor masker seperti pada Gambar 5 model yang digunakan sampel wajah bermasker 5.454 dan sampel tidak bermasker 4.675.



Gambar 5. Contoh pengujian detektor masker berbasis web
 Hasil pengujian seseorang bermasker dan tidak bermasker dapat dilihat pada Tabel 1.

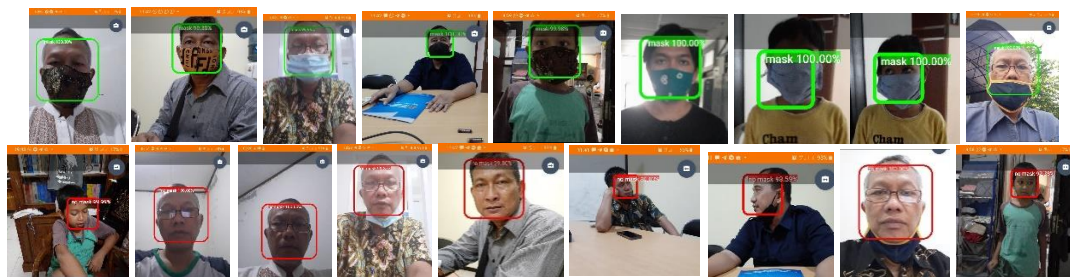
Tabel 1. Data pengujian Detektor masker berbasis web

No	Pengujian	Obyek Wajah Asli	Hasil Deteksi
1	Percobaan ke-1	Bermasker	No_Mask = 0% Mask = 100%
2	Percobaan ke-2	Bermasker	No_Mask = 0% Mask = 100%
3	Percobaan ke-3	Bermasker	No_Mask = 0% Mask = 100%
4	Percobaan ke-4	Bermasker	No_Mask = 0% Mask = 100%
5	Percobaan ke-5	Bermasker	No_Mask = 0% Mask = 100%
6	Percobaan ke-6	Bermasker	No_Mask = 0% Mask = 100%
7	Percobaan ke-7	Bermasker	No_Mask = 0% Mask = 100%
8	Percobaan ke-8	Bermasker	No_Mask = 0% Mask = 100%
9	Percobaan ke-9	Bermasker	No_Mask = 0% Mask = 100%
10	Percobaan ke-10	Bermasker	No_Mask = 0% Mask = 100%
11	Percobaan ke-11	Tidak Bermasker	No_Mask = 100% Mask = 0%
12	Percobaan ke-12	Tidak Bermasker	No_Mask = 100% Mask = 0%
13	Percobaan ke-13	Tidak Bermasker	No_Mask = 100% Mask = 0%
14	Percobaan ke-14	Tidak Bermasker	No_Mask = 100% Mask = 0%
15	Percobaan ke-15	Tidak Bermasker	No_Mask = 100% Mask = 0%
16	Percobaan ke-16	Tidak Bermasker	No_Mask = 100% Mask = 0%
17	Percobaan ke-17	Tidak Bermasker	No_Mask = 100% Mask = 0%
18	Percobaan ke-18	Tidak Bermasker	No_Mask = 100% Mask = 0%
19	Percobaan ke-19	Tidak Bermasker	No_Mask = 100% Mask = 0%
20	Percobaan ke-20	Tidak Bermasker	No_Mask = 100% Mask = 0%

Hasil pengujian mendapatkan hasil deteksi sangat baik dan telah berhasil mendeteksi dengan tingkat kebenaran 100% hal ini karena model hasil pelatihan mendapatkan bobot JST yang optimal sehingga proses deteksi klasifikasi obyek sangat akurat.

Pengujian Detektor Masker pada Smartphone Samsung A20

Pada pengujian ini detektor masker dijalankan di smartphone android Samsung A20 dengan cara menjalankan aplikasi mask detektor yang sudah diinstall dari hasil *build gradle* program yang telah dibuat dengan *framework example lite* dan mengisi model dari hasil pelatihan sejumlah 5454 wajah bermasker dan 4675 wajah tanpa masker. Contoh hasil pengujian untuk obyek tunggal seperti pada Gambar 6, detektor telah berhasil mendeteksi dengan tingkat kebenaran deteksi 100%.



Gambar 6. Hasil detektor untuk obyek tunggal

Untuk hasil uji deteksi obyek orang lebih dari dua seperti pada Gambar 7. Obyek diambil dari orang yang berkegiatan dalam suatu ruangan. Detektor telah berhasil mendeteksi obyek bermasker dan tidak bermasker tetapi jika obyek citra wajah terlalu kecil detektor tidak berhasil mendeteksi masker.



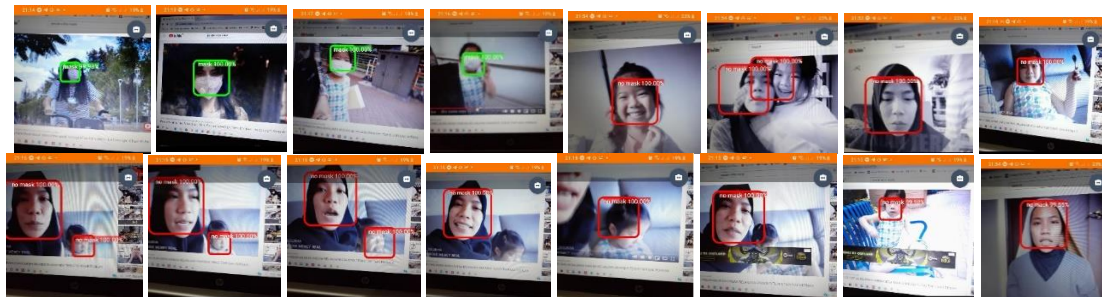
Gambar 7. Hasil detektor masker untuk obyek lebih dari 1 orang

Hasil pengujian waktu pada obyek berupa orang tunggal terdata pada Tabel 2. sebagai berikut.

Tabel 2. Data hasil pengujian obyek tunggal

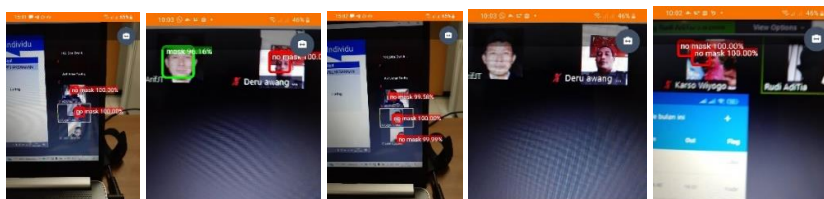
No	Obyek	Obyek Wajah Asli	Hasil Deteksi
1	Obyek-1	Bermasker	No_Mask = 0% Mask = 100%
2	Obyek-2	Bermasker	No_Mask = 0% Mask = 100%
3	Obyek-3	Bermasker	No_Mask = 0% Mask = 100%
4	Obyek-4	Bermasker	No_Mask = 0% Mask = 100%
5	Obyek-4	Bermasker	No_Mask = 0% Mask = 100%
6	Obyek-5	Bermasker	No_Mask = 0% Mask = 100%
7	Obyek-6	Bermasker	No_Mask = 0% Mask = 100%
8	Obyek-7	Bermasker	No_Mask = 0% Mask = 100%
9	Obyek-8	Bermasker	No_Mask = 0% Mask = 100%
10	Obyek-9	Bermasker	No_Mask = 0% Mask = 100%
11	Obyek-10	Tidak Bermasker	No_Mask = 100% Mask = 0%
12	Obyek-11	Tidak Bermasker	No_Mask = 100% Mask = 0%
13	Obyek-12	Tidak Bermasker	No_Mask = 100% Mask = 0%
14	Obyek-13	Tidak Bermasker	No_Mask = 100% Mask = 0%
15	Obyek-14	Tidak Bermasker	No_Mask = 100% Mask = 0%
16	Obyek-15	Tidak Bermasker	No_Mask = 100% Mask = 0%
17	Obyek-16	Tidak Bermasker	No_Mask = 100% Mask = 0%
18	Obyek-17	Tidak Bermasker	No_Mask = 100% Mask = 0%
19	Obyek-18	Tidak Bermasker	No_Mask = 100% Mask = 0%

Hasil pengujian pada obyek wajah orang yang tanyang di video komputer, contoh video tersebut adalah video yang tayang di youtube, hasil deteksi terlihat seperti pada Gambar 8 dengan tingkat keberhasilan deteksi 100%.



Gambar 8. Hasil deteksi masker pada obyek video yang tayang di komputer

Sedangkan pengujian mendeteksi wajah bermasker maupun tidak bermasker pada obyek berupa foto dilayar komputer contoh pada saat berlangsung zoom meeting dengan 5 kali pengujian acak didapat data deteksi benar = 8 dari 14 (57%), hasil deteksi salah = 1 dari 14 (0,07%) dan tidak dapat mendeteksi = 2 dari 14 (0,14%) sebagai mana dapat dilihat pada Gambar 9 hasil deteksi rendah 57% akibat dari pengambilan obyek bergerak tidak secara langsung tetapi melalui tampilan video hal ini karen citra obyke wajah yang tidka jelas dan terlalu kecil.



Gambar 9. Deteksi masker pada layar komputer saat zoom meeting

4. KESIMPULAN

Berdasarkan penelitian yang telah dilaksanakan maka dapat disimpulkan bahwa prototipe detektor masker wajah telah berhasil dibuat baik pada perangkat bergerak maupun pada web hosting. Tingkat keberhasilan deteksi obyek waktu nyata obyek tunggal maupun banyak dapat mencapai 100%. Alat juga dapat mendeteksi *still image* atau citra tidak bergerak di komputer atau citra foto dengan tingkat keberhasilan 100%.

Detektor wajah juga dapat bekerja untuk mendeteksi obyek yang tampil di video komputer dengan keberhasilan mencapai 100%, tetapi pada percobaan mendeteksi obyek orang bermasker pada zoom meeting tingkat keberhasilan hanya 57% disebabkan obyek terlalu kecil.

Saran yang dapat diberikan setelah melakukan penelitian ini yaitu perlu dikembangkan ke tingkat penelitian berikutnya yaitu ditambahkan sensor suhu, ataupun dihubungkan dengan perangkat lain sebagai kontrol proses yang diinginkan, misalnya sistem buka tutup pintu berdasarkan deteksi pemakaian masker dan aplikasi lainnya.

UCAPAN TERIMAKASIH

Penulis mengucapkan terimakasih kepada Lembaga Penelitian dan Pengabdian pada Masyarakat Universitas Muhammadiyah Purwokerto yang telah mendanai penelitian ini melalui surat perjanjian No. No. A.11-III/699-S.Pj./LPPM/XII/2020.

DAFTAR PUSTAKA

- [1] X. Rizki, Saraswati Anindita., Andree Kurniawan., 2020., *Efficacy of Cloth Mask in Reducing COVID-19 Transmission: A Literature Review*, Jurnal Kesehatan Masyarakat Nasional Kesmas, Vol 15 No 2.
- [2] Joachim D Pleil, Jonathan D Beauchamp., et al., 2020, *The scientific rationale for the use of simple masks or improvised facial coverings to trap exhaled aerosols and possibly reduce the breathborne spread of COVID-19*, Vol 4 No.3, Journal of Breath Research.
- [3] R. Davies, A., Thompson, K., Giri, K., Kafatos, G., Walker, J., & Bennett, A., 2013. *Testing the Efficacy of Homemade Masks: Would They Protect in an Influenza Pandemic?* Disaster Medicine and Public Health Preparedness, 7(4), 413-418
- [4] D. Sari, Devi Pramita., Nabila Sholihah 'Atiqoh., 2020., *Hubungan Antara Pengetahuan Masyarakat Dengan Kepatuhan Penggunaan Masker Sebagai Upaya Pencegahan Penyakit COVID-19 Di Ngronggah*, INFOKES, VOL 10 NO 1, FEBRUARI2020, ISSN : 2086 -2628.
- [5] Lambacing, M. and Ferdiansyah, F. 2020. *Rancang Bangun New Normal Covid-19 Masker Detektor Dengan Notifikasi Telegram Berbasis Internet Of Things*. Dinamik. 25, 2 (Jul. 2020), 77-84. DOI: <https://doi.org/10.35315/dinamik.v25i2.8070>
- [6] Dana H. Ballard; Christopher M. Brown., 1982. *Computer Vision*. Prentice Hall.
- [7] Pulli, Kari; Baksheev, Anatoly; et al., 2012., *Realtime Computer Vision with OpenCV: Mobile computer-vision technology will soon become as ubiquitous as touch interfaces*. ACM Queue Digital Library Journal.
- [8] Bengio, Y.; Courville, A.; Vincent, P. 2013. *Representation Learning: A Review and New Perspectives*. IEEE Transactions on Pattern Analysis and Machine Intelligence. 35 (8): 1798–1828.
- [9] Cireşan, D.; Meier, U.; Schmidhuber, J., 2012. *Multi-column deep neural networks for image classification*. 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 3642–3649.
- [10] Chen, Yung-Yao; Lin, Yu-Hsiu; Kung, Chia-Ching; Chung, Ming-Han; Yen, I-Hsuan. 2019. "Design and Implementation of Cloud Analytics-Assisted Smart Power Meters Considering Advanced Artificial Intelligence as Edge Analytics in Demand-Side Management for Smart Homes" ., Sensors 19, no. 9: 2047.
- [11] Matusugu, Masakazu; Katsuhiko Mori; Yusuke Mitari; Yuji Kaneda., 2003. "Subject independent facial expression recognition with robust face detection using a convolutional neural network". Jurnal Neural Networks, Volume 16, Issues 5–6, June–July 2003, Pages 555-559.
- [12] G. Jignesh Chowdary, et al., 2020., *Face Mask Detection using Transfer Learning of InceptionV3*., Cornell University, <https://arxiv.org/abs/2009.08369v2>
- [13] Loey, Mohamed Loey, et al., 2020., *A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic*, Science Direct Measurement, Volume 167, 2021, 108288, ISSN 0263-2241, <https://doi.org/10.1016/j.measurement.2020.108288>. Published online 2020 Jul 28. doi: 10.1016/j.measurement.2020.108288

