

Pengembangan Sistem Pendeteksi Masker Sesuai Protokol Kesehatan dengan Algoritma Mobilenetv2 dan Raspberry Pi

Imam M. Shofi¹, Luh Kesuma Wardhani¹, Nenny Anggraini¹, Nashrul Hakiem¹, Denny Saputra¹, Ariq Cahya Wardhana^{2,*}

¹Fakultas Sains dan Teknologi, Program Studi Teknik Informatika, Universitas Islam Negeri Syarif Hidayatullah Jakarta, Tangerang Selatan, Indonesia

² Fakultas Informatika, Program Studi Rekayasa Perangkat Lunak, Institut Teknologi Telkom Purwokerto, Banyumas, Indonesia

Email: ¹imam@uinjkt.ac.id, ²luhkesuma@uinjkt.ac.id, ³nenny.anggraini@uinjkt.ac.id, ⁴deny.saputra@uinjkt.ac.id,

⁵deny.saputra@uinjkt.ac.id, ^{6,*}ariq@ittelkom-pwt.ac.id

Email penulis Korespondensi: ariq@ittelkom-pwt.ac.id

Submitted:07/09/2022; Accepted:24/09/2022; Published: 30/09/2022

Abstrak—Corona virus jenis baru ditemukan manusia sejak Desember 2019 di Wuhan Cina. Pada manusia, biasanya corona virus menyebabkan penyakit infeksi saluran pernapasan, mulai flu biasa hingga penyakit serius seperti Middle East Respiratory (MERS) dan Sindrom Pernafasan Akut Berat atau Severe Acute Respiratory Syndrome (SARS). Coronavirus jenis baru ini kemudian diberi nama Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-COV2) dan menyebabkan penyakit Coronavirus Disease-2019 (COVID-19). COVID-19 dapat menyebabkan gejala ringan hingga berat. Jadi, pemakaian masker dan jaga jarak sangat penting untuk menghentikan penyebaran COVID-19. Pada penelitian sebelumnya, sudah dikembangkan model deep learning untuk mengenali apakah orang tersebut mengenakan masker atau tidak. Dalam penelitian sebelumnya, klasifikasi terbatas pada apakah manusia mengenakan masker atau tidak. Belum terdapat klasifikasi apakah penggunaan masker tersebut benar atau salah dan apakah masker yang dikenakan merupakan masker yang sesuai dengan rekomendasi Kementerian Kesehatan. Sehingga pada penelitian kali ini, sistem deteksi penggunaan masker mampu mendeteksi penggunaan masker yang sesuai dengan anjuran Kementerian Kesehatan RI yang merujuk pada Panduan WHO interim 5 Juni 2020 tentang ajuran mengenai penggunaan masker dalam konteks COVID-19, yaitu penggunaan masker kain, masker medis, dan dapat memastikan masker menutupi mulut dan hidung, sesuaikan dengan batang hidung. Hasilnya sistem dengan model SSDLite Mobilenet V2 memiliki FPS yang paling tinggi dibandingkan dengan sistem yang menggunakan sistem dengan SSDMNv2. Yaitu FPS maksimal yang didapatkan adalah 3.57 FPS dan minimal FPS adalah 3.45 FPS.

Kata Kunci: Klasifikasi; Masker; MobileNetV2; Raspberry pi; Prototyping; Deep Learning

Abstract—A new type of human coronavirus was discovered in December 2019 in Wuhan, China. In humans, coronaviruses usually cause respiratory tract infections, ranging from the common cold to serious diseases such as Middle East Respiratory (MERS) and Severe Acute Respiratory Syndrome (SARS). This new type of coronavirus was later named Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-COV2) and caused Coronavirus Disease-2019 (COVID-19). COVID-19 can cause mild to severe symptoms. So, wearing a mask and keeping a distance is very important to stop the spread of COVID-19. In previous research, a deep learning model has been developed to identify whether the person is wearing a mask or not. In previous studies, the classification was limited to whether humans wore masks or not. There is no classification as to whether the use of masks is right or wrong and whether the masks worn are masks that are in accordance with the recommendations of the Ministry of Health. So that in this study, the detection system for the use of masks is able to detect the use of masks in accordance with the recommendations of the Indonesian Ministry of Health which refers to the interim WHO Guidelines June 5, 2020, regarding recommendations regarding the use of masks in the context of COVID-19, namely the use of cloth masks, medical masks, and masks. can ensure the cover of the mouth and nose, and adjust to the bridge of the nose. The result is a system with the SSDLite Mobilenet V2 model has the highest FPS compared to a system using a system with SSDMNv2. That is, the maximum FPS obtained is 3.57 FPS and the minimum FPS is 3.45 FPS.

Keywords: Classification; Mask; MobileNetV2; Raspberry pi; Prototyping; Deep Learning

1. PENDAHULUAN

Corona virus jenis baru ditemukan manusia sejak Desember 2019 di Wuhan Cina. Pada manusia, biasanya corona virus menyebabkan penyakit infeksi saluran pernapasan, mulai flu biasa hingga penyakit serius seperti Middle East Respiratory (MERS) dan Sindrom Pernafasan Akut Berat atau Severe Acute Respiratory Syndrome (SARS). Coronavirus jenis baru ini kemudian diberi nama Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-COV2) dan menyebabkan penyakit Coronavirus Disease-2019 (COVID-19) [1]. COVID-19 dapat menyebabkan gejala ringan hingga berat. Penyakit ini dapat menyebar melalui tetesan kecil (droplet) dari hidung atau mulut pada saat batuk, bersin, atau berbicara. Menurut Michal Tal, PhD, ahli imunologi di Universitas Stanford mengatakan orang yang sudah divaksinasi belum tentu kebal dari virus COVID-19. Serta Anne Rimoin, PhD, MPH, Profesor Epidemiologi di UCLA Fielding School of Public Health mengungkapkan bahwa perlu waktu untuk mendistribusikan vaksin secara luas dan mencapai kekebalan tubuh pada sekelompok orang. Jadi, pemakaian masker dan jaga jarak tetap penting untuk menghentikan penyebaran COVID-19.

Pemakaian masker diwajibkan untuk semua orang, baik orang sehat maupun sakit. Orang yang sehat disarankan untuk menggunakan masker kain. Bagi orang yang memiliki gejala infeksi pernapasan (batuk atau bersin), dicurigai infeksi COVID-19 dengan gejala ringan, merawat orang yang bergejala seperti demam dan batuk, dan para petugas kesehatan menggunakan masker bedah [2]. Direktur Jenderal Pencegahan dan Pengendalian Penyakit dr.

Achmad Yurianto menjelaskan, ada 3 jenis masker yang direkomendasikan, yaitu masker N95, masker bedah, dan masker kain. Untuk masker kain tidak boleh menggunakan kain yang tipis seperti masker scuba dan buff [3]. Pada update terbarunya, menurut [4], ditemukan penggunaan masker dengan mengkombinasikan masker kain di atas masker bedah dapat meningkatkan tingkat perlindungan terhadap virus. Pada simulasi menggunakan Fugaku. Fugaku adalah superkomputer tercepat di dunia. Pelindung elastik atau yang lebih dikenal dengan face shield tidak efektif untuk menahan droplets. Dalam pengujiannya dengan tetesan aerosol, tetesan tersebut dapat bergerak di sekitar pelindung dengan mudah [5].

Penggunaan masker yang salah juga ditemukan dengan tidak menutupi bagian hidung atau mulut secara sempurna [6]. Himbauan dr. Achmad Yurianto menyerukan agar masyarakat tidak menurunkan maskernya ke area dagu, baik saat bersantai, menyetir, maupun makan, dikarenakan hal tersebut dapat mencemari bagian dalam masker dengan berbagai virus yang mungkin menempel di dagu [7]. Menurut [4], penggunaan masker yang memiliki lubang ventilasi tidak disarankan, karena dapat menjadi media masuknya virus. Masker yang dapat memblokir cahaya saat diarahkan ke sumber cahayanya, dalam arti lain, masker tidak boleh menggunakan kain yang tipis.

Menurut [8] metode deteksi objek terbagi menjadi 2 jenis, yaitu dua tahap dan satu tahap. Deteksi objek dua tahap memiliki tingkat akurasi yang tinggi namun kecepatan deteksinya lebih lambat dari objek deteksi satu tahap. Sedangkan deteksi objek satu tahap memiliki tingkat akurasi yang lebih rendah, namun memiliki kecepatan deteksi yang lebih cepat dibandingkan dengan metode objek deteksi dua tahap.

Dari studi literatur yang peneliti lakukan, terdapat 2 metode yang dilakukan untuk klasifikasi penggunaan masker, yaitu, klasifikasi objek dan deteksi objek. Metode klasifikasi objek terdapat pada penelitian [9] menggunakan Xception dan penelitian [10] menggunakan MobileNetV2. Untuk metode deteksi objek terdapat pada penelitian [11] menggunakan YOLOV3-Tiny, penelitian [12] menggunakan YOLOV2, penelitian [13] menggunakan basis YOLOV3, penelitian [14] menggunakan VGG-16, dan penelitian [15] serta [16] menggunakan Haar Cascade serta penelitian [17] dan [18] menggunakan deteksi wajah manusia dari OpenCV dan diklasifikasikan penggunaan maskernya menggunakan MobileNetV2. Pada penelitian [18] algoritma ini disebut dengan SSDMN2.

Dalam penelitian sebelumnya, klasifikasi terbatas pada apakah manusia mengenakan masker atau tidak. Belum terdapat klasifikasi apakah penggunaan masker tersebut benar atau salah dan apakah masker yang dikenakan merupakan masker yang sesuai dengan rekomendasi Kementerian Kesehatan. Sehingga pada penelitian kali ini diharapkan bahwa sistem deteksi penggunaan masker mampu mendeteksi penggunaan masker yang sesuai dengan anjuran Kementerian Kesehatan RI yang merujuk pada Panduan WHO interim 5 Juni 2020 tentang anjuran mengenai penggunaan masker dalam konteks COVID-19, yaitu penggunaan masker kain, masker medis, dan dapat memastikan masker menutupi mulut dan hidung, sesuaikan dengan batang hidung.

2. METODOLOGI PENELITIAN

2.1 Komunikasi

Paradigma prototyping dimulai dengan adanya komunikasi antara aktor yang akan menggunakan sistem tersebut untuk menentukan sasaran hasil keseluruhan dari alat/sistem, mengidentifikasi kebutuhan dan lingkungan dimana sistem akan digunakan. Pada tahapan ini komunikasi yang dilakukan adalah dengan mencari informasi terkait, melalui referensi berita online, jurnal dan skripsi tentang sistem klasifikasi masker dan penggunaan masker di masyarakat.

2.2 Pemanasan Global

Pada tahap ini penulis membuat rencana sistem bekerja dengan menggunakan beberapa analisa sebagaimana di bawah ini.

a. Analisis Klasifikasi Penggunaan Masker

Klasifikasi penggunaan masker yang penulis akan buat pada sistem adalah 3 klasifikasi, menggunakan masker, tidak menggunakan masker, menggunakan masker dengan cara yang salah.

b. Analisis Dataset

Pada algoritma SSDMN2, model untuk pendeteksi wajah sudah tersedia dari OpenCV, sehingga pada algoritma SSDMN2, penulis hanya melakukan latihan pada model untuk klasifikasi penggunaan masker menggunakan algoritma MobilenetV2. Penulis menggunakan model pra latih dari Keras untuk dilatih dengan dataset dari penulis. Tujuan menggunakan model pra latih ini agar proses pelatihan lebih cepat. Proses pelatihan untuk model klasifikasi penggunaan masker, membutuhkan dataset berupa gambar yang diberi label dengan folder. Tiap folder merupakan satu label dari klasifikasi tersebut.

Pada algoritma SSDLite MobilenetV2, SSDLite MobilenetV3 Large. Pelatihan model untuk algoritma ini berbeda dari yang sebelumnya. Jika sebelumnya melakukan pelatihan untuk model klasifikasi objek, untuk algoritma ini melakukan pelatihan untuk model deteksi objek. Label tidak dibuat dengan folder, tetapi perlu dilakukan notasi bounding box untuk setiap target objek pada contoh gambar. Notasi ini nantinya diberi label, sesuai dengan klasifikasinya.

Oleh karena itu, dataset yang digunakan pada pelatihan MobilenetV2 dengan algoritma objek deteksi satu tahap berbeda. Untuk pelatihan objek deteksi satu tahap, memerlukan proses lebih banyak dibandingkan dengan pelatihan klasifikasi objek dalam menyiapkan datasetnya.

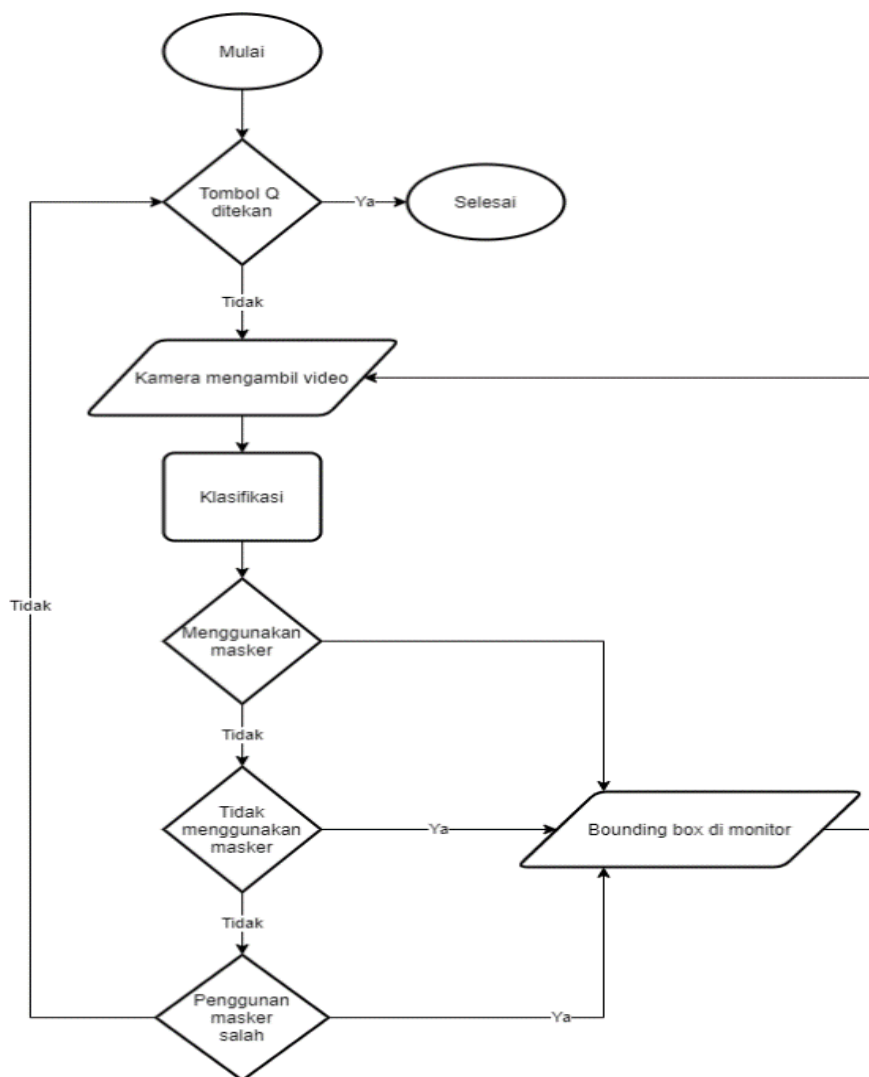
Penulis menggunakan dataset [12] yang sudah terbagi menjadi dataset latih dan dataset uji. Dataset latih memiliki 7385 gambar yang terdiri atas 6702 anotasi menggunakan masker (withmask), 9680 anotasi tanpa masker (without_mask) dan 320 anotasi menggunakan masker yang salah (incorrect_mask). Sementara itu, dataset uji memiliki 1820 gambar yang terdiri atas 993 anotasi menggunakan masker (withmask), 791 anotasi tanpa masker (without_mask) dan 46 anotasi menggunakan masker yang salah (incorrect_mask).

Selain dataset di atas, penulis juga menggunakan dataset [17] untuk melatih model klasifikasi masker dengan MobileNetV2, yang terdiri dari 5521 gambar menggunakan masker dengan label with_mask dan 5521 gambar tanpa menggunakan masker dengan label without_mask.

Selanjutnya dari dataset tersebut, penulis menambahkan 5521 gambar penggunaan masker yang salah, yang didapat dari [18]. Dataset ini diberi nama dengan MaskedFace-Net. Terdapat 70.000 gambar pada dataset tersebut. Pada dataset ini, dari foto wajah manusia, masker diberikan secara otomatis melalui program. Untuk menyesuaikan dengan jumlah dataset yang digunakan oleh [6]. Penulis mengambil 1841 gambar dengan label Mask_Mouth_Chin di mana masker hanya menutup bagian mulut dan dagu. 1840 gambar dengan label Mask_Nose_Mouth di mana masker hanya menutup bagian hidung dan mulut. Dan 1840 gambar dengan label Mask_Chin di mana masker hanya menutup bagian dagu.

c. Analisis Kebutuhan Fungsional Sistem

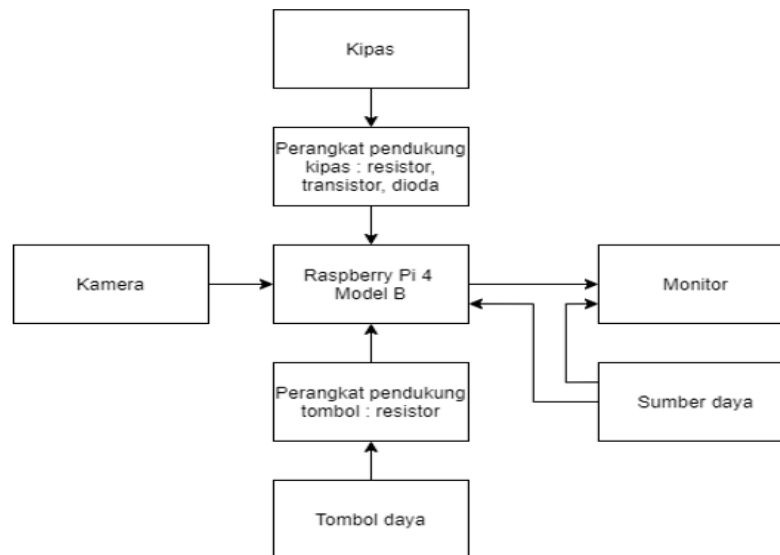
Kebutuhan sistem adalah mampu mengklasifikasikan penggunaan masker dan memberikan output berupa suara. Seperti yang digambarkan pada flowchart di bawah ini:



Gambar 1. Flowchart Program

2.3 Pemodelan Cepat

Pada tahap pemodelan cepat ini, penulis membuat blok diagram dari sistem klasifikasi penggunaan masker :



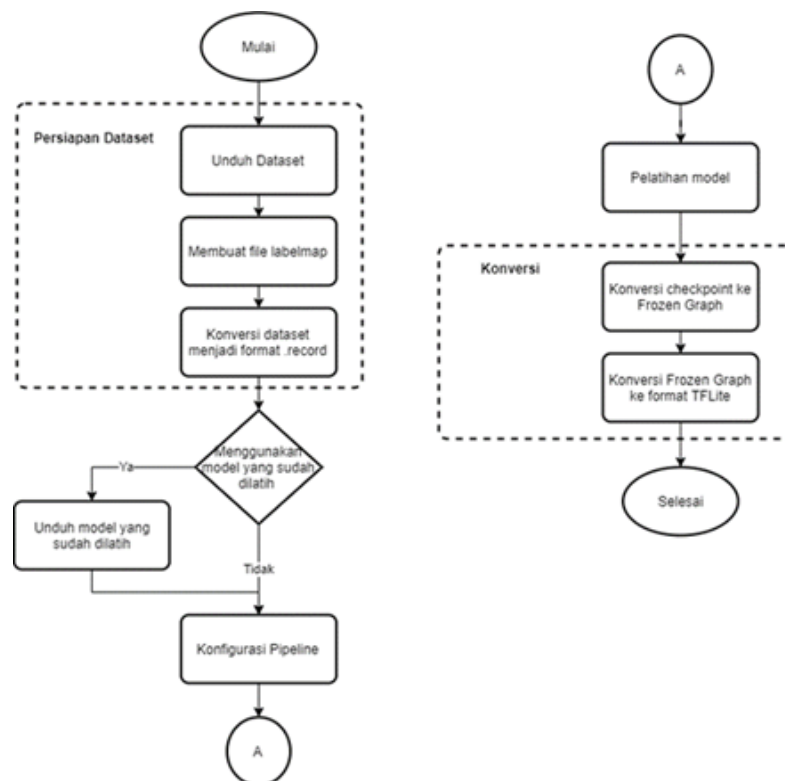
Gambar 2. Blok diagram alat

2.4 Konstruksi

Tahapan selanjutnya adalah konstruksi, pada tahap ini penulis bagi menjadi 2, yaitu pembuatan model deteksi di mana model ini nanti akan digunakan sistem untuk mendeteksi penggunaan masker.

a. Model Deteksi

Dalam pembuatan model deteksi, pada penelitian ini dibagi menjadi 2, yaitu pelatihan untuk model deteksi satu tahap dan pelatihan untuk model deteksi dua tahap. Pada gambar 7 ditunjukkan proses pembuatan model deteksi satu tahap.

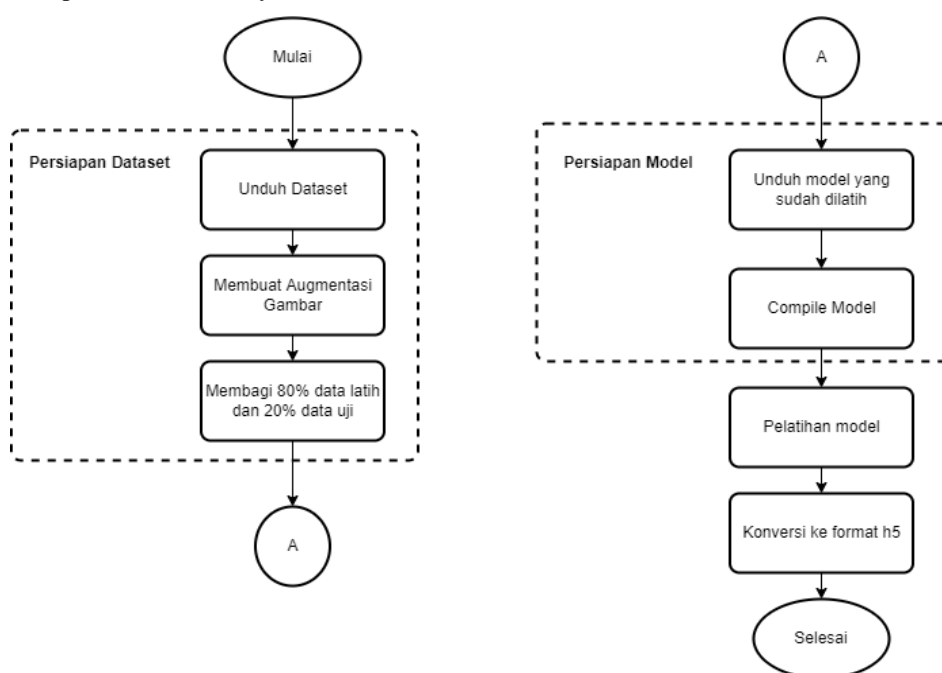


Gambar 7. Tahapan pembuatan model satu tahap

Proses dimulai dengan mengunduh dataset. Untuk model deteksi satu tahap, dataset yang digunakan sudah dibagi menjadi 7385 gambar untuk proses pelatihan dan 1820 gambar untuk proses validasi. Setelah dataset diunduh, dilakukan pembuatan file yang bernama labelmap, labelmap ini berisi kategori klasifikasi untuk proses pelatihan nanti. Setelah dataset dan labelmap sudah ada, dilanjutkan dengan proses konversi keduanya menjadi file .record. Sebuah file yang nantinya akan digunakan dalam proses pelatihan dengan Tensorflow Object Detection API.

Dalam pelatihan ini, pengujian model deteksi objek satu tahap memiliki 2 kategori, yaitu dengan menggunakan model yang sudah dilatih sebelumnya atau disebut dengan fine tuning dan melakukan proses pelatihan dari awal, tanpa bantuan model yang sudah dilatih sebelumnya. Jika menggunakan fine tuning maka diharuskan mengunduh model yang sudah dilatih sebelumnya terlebih dahulu. Setelah itu, dilakukan konfigurasi pipeline. Pipeline di sini berisi konfigurasi mengenai pelatihan yang akan dilakukan. Konfigurasi mencakup, model apa yang digunakan, menggunakan fine tuning atau tidak, berapa jumlah stepsnya, lokasi penyimpanan dataset, dan lokasi penyimpanan labelmap. Setelah selesai dengan persiapan dan konfigurasi. Maka dilakukan pelatihan.

Setelah pelatihan selesai, model yang sudah dilatih akan dikonversi menjadi frozen graph. Setelah menjadi frozen graph, dilakukan proses konversi lagi menjadi .tflite. Di mana file .tflite ini nantinya akan digunakan pada Raspberry Pi. Pada gambar 4.10 dijelaskan mengenai tahapan pembuatan model deteksi dua tahap. Pada pembuatan model deteksi dua tahap, dalam penelitian ini hanya membuat bagian untuk mengklasifikasikan penggunaan masker. Jadi pada model deteksi dua tahap, tahap pertama adalah mengenali wajah manusia terlebih dahulu. Setelah wajah manusia dikenali, nantinya akan diambil daerah dari wajah tersebut yang selanjutnya akan dilakukan proses klasifikasi pada daerah tersebut. Proses dua tahap ini terjadi pada Raspberry Pi. Pada penelitian ini menggunakan SSD sebagai pendeteksi wajah yang sudah ada modelnya. Dan MobilenetV2 untuk mengklasifikasikan penggunaan masker yang pada penelitian ini penulis membuatnya sendiri.



Gambar 8. Tahapan pembuatan model klasifikasi penggunaan masker

Proses pembuatan klasifikasi penggunaan masker untuk model deteksi dua tahap diawali dengan persiapan dataset. Dataset ini merupakan gabungan dari beberapa dataset sebagaimana yang telah dijelaskan sebelumnya. Dataset masih berupa gambar biasa, oleh karena itu ditambahkan proses augmentasi untuk menambah keanekaragaman bentuk. Setelah itu, dataset dibagi menjadi 80% data latih dan 20% data uji.







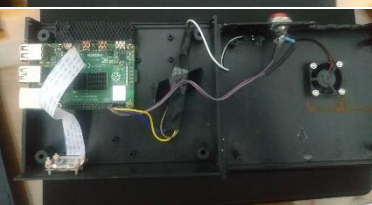
Selanjutnya, siapkan model MobilenetV2 yang sudah dilatih sebelumnya. Gunanya menggunakan model yang sudah dilatih sebelumnya adalah untuk mempercepat proses pengenalan objek sehingga tidak membutuhkan waktu yang lama untuk mendapatkan hasil yang baik. Setelah model disiapkan, maka model akan di-compile untuk nantinya digunakan pada tahap pelatihan. Setelah semuanya siap, proses pelatihan dapat dilakukan. Terakhir, setelah proses pelatihan selesai, konversi model yang telah dilatih menjadi format .tflite. Format ini nantinya akan digunakan pada Raspberry Pi.

b. Alat

Alat di susun berdasarkan blok diagram pada pemodelan. Pada bagian atas, terdapat kipas yang dipasang menggunakan baut, kipas berguna untuk mendinginkan Raspberry Pi ketika suhunya mulai panas. Pada bagian depan, terdapat kamera yang dipasang menggunakan baut, kamera menghadap ke depan agar dapat menangkap objek dengan baik. Pada bagian bawah, tidak terdapat komponen, hanya nampak 4 baut yang digunakan untuk mengunci Raspberry Pi ke casing. Pada bagian belakang, terdapat tombol daya, port power, port micro-hdmi, dan port jack audio. Pada bagian kanan terdapat port usb yang dapat dimanfaatkan sebagai penghubung ke keyboard dan mouse serta port ethernet agar dapat terhubung ke jaringan melalui kabel LAN.

Pada bagian kiri tidak terdapat komponen. Jika dilihat bagian dalamnya, terdapatnya raspberry pi 4 model b, kabel jumper, dan resistor. Untuk komponen pendukung kipas, tidak nampak, karena sudah dilapisi dengan flame tube. Berikut adalah penampakannya, ditunjukkan pada gambar dalam tabel :

Tabel 3. Penampakan alat

Posisi	Tampak
Atas	
Depan	
Bawah	
Belakang	
Kanan	
Kiri	
Dalam	

2.5 Evaluasi

Evaluasi ini dilakukan pada lingkungan dengan tingkat cahaya sebesar 250-257 lux. Pengujian ini dilakukan sampai mendapatkan output yang diinginkan. Ketika sudah beberapa kali percobaan, klasifikasi tetap tidak berhasil, maka hasil itu yang dimasukkan ke dalam tabel.

3. HASIL DAN PEMBAHASAN

3.1 Hasil

Dalam satu kemasan terdapat kamera, Raspberry Pi 4 Model B, led, dan tombol daya, tombol program, kipas, beserta dengan perangkat pendukungnya. Kemasan ini sebagai inti dari sistem. Sementara di bagian terpisah terdapat speaker dan monitor sebagai perangkat keluaran. Serta sumber daya yang diambil dari listrik AC (listrik PLN) menggunakan adaptor. Baik adaptor untuk monitor dan adaptor untuk Raspberry Pi 4 Model B. Pemasangan perangkat lain dengan Raspberry Pi 4 Model B akan dijelaskan lebih rinci di bawah ini :

a. Kamera

Pemasangan kamera dilakukan dengan menghubungkan kabel kamera dengan port CSI camera pada Raspberry Pi 4 Model B. utama, persegi panjang hitam dan lingkaran adalah lubang untuk port dan sensor.

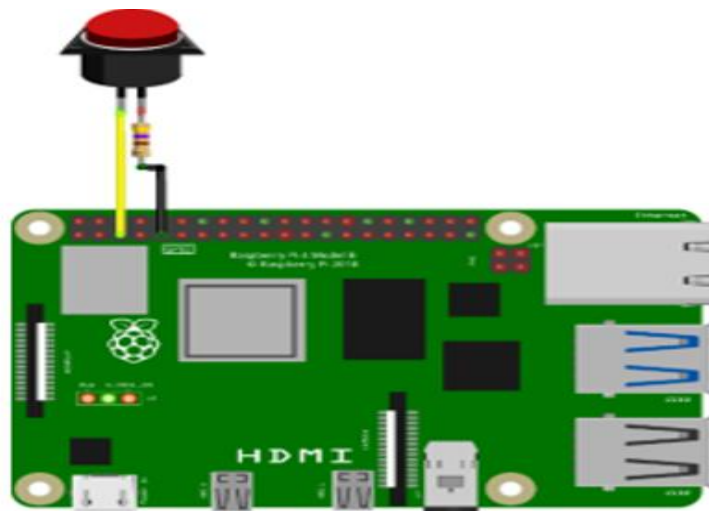


Gambar 3. Port CSI

b. Tombol Daya dan Tombol Program

Menghubungkan tombol dengan Raspberry Pi 4 Model B memerlukan perangkat pendukung, yaitu resistor. Masing – masing tombol yang digunakan memiliki jenis yang berbeda. Untuk tombol daya menggunakan tombol tekan tanpa momentary switch. Sementara tombol program menggunakan tombol tekan dengan momentary switch.

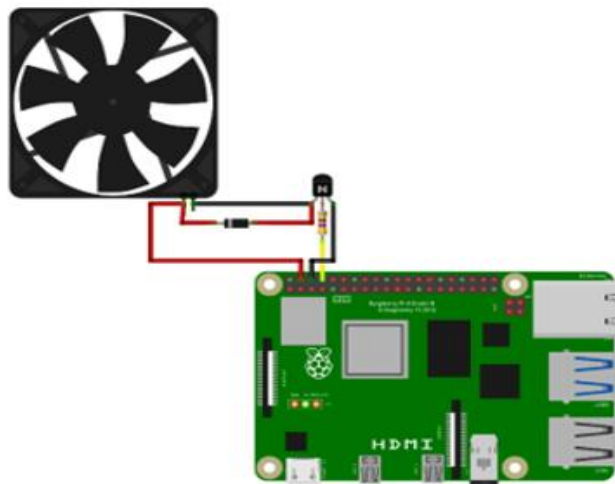
Untuk tombol daya, hubungkan pin GPIO 3 ke salah satu kaki tombol, dan untuk kaki yang satunya, hubungkan dengan resistor 470 Ohm, setelahnya hubungkan ke pin 9 pada papan Raspberry Pi 4 Model B.



Gambar 4. Rangkaian Tombol

c. Kipas

Menghubungkan kipas dengan Raspberry Pi 4 Model B memerlukan perangkat pendukung, yaitu, resistor, dioda, dan transistor. Hubungkan kabel positif kipas dengan pin 4 pada papan Raspberry Pi 4 Model B dan kutub katoda diode. Untuk kabel negative kipas, akan dihubungkan dengan collector transistor. Base dari transistor akan terhubung dengan resistor sebelum menyambung dengan pin GPIO 14. Untuk emitter transistor akan terhubung dengan pin 6 pada papan Raspberry Pi 4 Model B.



Gambar 5. Rangkaian Kipas

d. Monitor dan Sumber Daya

Sumber daya akan terhubung dengan port usb Type C (1) yang terhubung dengan adaptor. Monitor akan terhubung dengan micro-HDMI 1 (2).



Gambar 6. Port USB Type-C, Micro HDMI

Percobaan SSDLite Mobilenet V2 Menggunakan Fine Tuning dengan 20.000 Steps. Selama proses evaluasi, FPS maksimal yang didapatkan adalah 3.52 FPS dan minimal FPS 3.33 FPS. Serta konsumsi daya yang digunakan adalah 8.0 – 8.5 Watt. Pada tabel 10 di bawah ini adalah hasil dari evaluasi kemampuan deteksinya.

Table 4. Percobaan SSDLite MobileNetV2 Menggunakan *Fine Tuning* dengan 20.000 Steps

No.	Input	Keterangan
1	Menggunakan masker medis	Mudah terdeteksi
2	Penggunaan masker medis salah	Mudah terdeteksi
3	Tidak menggunakan masker	Mudah terdeteksi
4	Menggunakan masker kain	Mudah terdeteksi
5	Menggunakan masker kain salah	Mudah terdeteksi
6	Menutup dengan tangan	Mudah terdeteksi
7	Menutup dengan koran	Mudah terdeteksi

Percobaan SSDLite Mobilenet V2 Tanpa Menggunakan Fine Tuning dengan 20.000 Steps. Selama proses evaluasi, FPS maksimal yang didapatkan adalah 3.44 FPS dan minimal FPS 3.31 FPS. Serta konsumsi daya yang digunakan adalah 7.7 – 8.5 Watt. Pada tabel 11 di bawah ini adalah hasil dari evaluasi kemampuan deteksinya.

Tabel 5. Percobaan SSDLite MobileNet V2 Tanpa Menggunakan *Fine Tuning* dengan 20.000 Steps

No.	Input	Keterangan
1	Menggunakan masker medis	Mudah terdeteksi
2	Penggunaan masker medis salah	Terdeksi benar, seharusnya terdeteksi salah
3	Tidak menggunakan masker	Mudah terdeteksi
4	Menggunakan masker kain	Mudah terdeteksi

5	Menggunakan masker kain salah	Terkadang terdeteksi tidak menggunakan masker, terkadang terdeteksi benar, seharusnya terdeteksi salah
6	Menutup dengan tangan	Mudah terdeteksi
7	Menutup dengan koran	Mudah terdeteksi

Percobaan SSD Mobilenet V2 Menggunakan Fine Tuning dengan 20.000 Steps. Selama proses evaluasi, FPS maksimal yang didapatkan adalah 3.39 FPS dan minimal FPS 3.11 FPS. Serta konsumsi daya yang digunakan adalah 7.7 – 8.5 Watt. Pada tabel 12 di bawah ini adalah hasil dari evaluasi kemampuan deteksinya.

Tabel 6. Percobaan SSD MobileNetV2 Menggunakan *Fine Tuning* dengan 20.000 Steps

No.	Input	Keterangan
1	Menggunakan masker medis	Mudah terdeteksi
2	Penggunaan masker medis salah	Terkadang terdeteksi salah, terkadang terdeteksi benar, terkadang tidak terdeteksi
3	Tidak menggunakan masker	Mudah terdeteksi
4	Menggunakan masker kain	Mudah terdeteksi
5	Menggunakan masker kain salah	Terdeteksi tanpa masker, seharusnya terdeteksi salah
6	Menutup dengan tangan	Tidak terdeteksi
7	Menutup dengan koran	Tidak terdeteksi

Percobaan SSD Mobilenet V2 Tanpa Menggunakan Fine Tuning dengan 20.000 Steps. Selama proses evaluasi, FPS maksimal yang didapatkan adalah 3.37 FPS dan minimal FPS 3.21 FPS. Serta konsumsi daya yang digunakan adalah 7.7 – 8.5 Watt. Pada tabel 13 di bawah ini adalah hasil dari evaluasi kemampuan deteksinya.

Tabel 7. Percobaan SSD MobileNetV2 Tanpa Menggunakan *Fine Tuning* dengan 20.000 Steps

No.	Input	Keterangan
1	Menggunakan masker medis	Mudah terdeteksi
2	Penggunaan masker medis salah	Terdeteksi benar, seharusnya terdeteksi salah
3	Tidak menggunakan masker	Mudah terdeteksi
4	Menggunakan masker kain	Mudah terdeteksi
5	Menggunakan masker kain salah	Mudah terdeteksi
6	Menutup dengan tangan	Mudah terdeteksi
7	Menutup dengan koran	Mudah terdeteksi

Percobaan SSDMN2. Selama proses evaluasi, FPS maksimal yang didapatkan adalah 1.42 FPS dan minimal FPS 0.78 FPS. Serta konsumsi daya yang digunakan adalah 8.1 – 9.2 Watt. Pada tabel 16 di bawah ini adalah hasil dari evaluasi kemampuan deteksinya.

Tabel 8. Percobaan SSDMN2

No.	Input	Keterangan
1	Menggunakan masker medis	Mudah terdeteksi
2	Penggunaan masker medis salah	Harus mendekatkan wajah ke kamera
3	Tidak menggunakan masker	Mudah terdeteksi
4	Menggunakan masker kain	Mudah terdeteksi
5	Menggunakan masker kain salah	Berhasil terdeteksi namun sulit
6	Menutup dengan tangan	Mudah terdeteksi
7	Menutup dengan koran	Terdeteksi benar, seharusnya terdeteksi tanpa masker

Berdasarkan tabel hasil evaluasi deteksi pada bab metode di evaluasi, untuk klasifikasi menggunakan masker medis, tidak menggunakan masker, menggunakan masker kain, menutup wajah dengan tangan, menutup wajah dengan koran mudah terdeteksi oleh hampir semua sistem. Kecuali menutup wajah dengan koran pada sistem SSDMN2 terdeteksi benar, seharusnya terdeteksi tanpa masker. Perlu menjadi catatan. Dataset yang digunakan pada sistem SSDMN2 berbeda dengan sistem yang menggunakan model deteksi satu tahap. Untuk klasifikasi menggunakan masker yang salah, baik dengan masker medis maupun masker kain sulit untuk dideteksi.

Terlihat juga perbedaan FPS yang didapatkan selama percobaan berlangsung. Dalam percobaan tersebut, peneliti mengambil nilai FPS maksimum dan minimum selama percobaan berlangsung.

Tabel 9. Hasil evaluasi FPS

No.	Percobaan	FPS
1	SSDLite Mobilenet V2 Fine Tuning	3.43 – 3.57 FPS
2	SSDLite Mobilenet V2 Tanpa Fine Tuning	3.33 – 3.49 FPS
3	SSD Mobilenet V2 Fine Tuning	3.40 – 3.83 FPS
4	SSD Mobilenet V2 Tanpa Fine Tuning	3.27 – 3.42 FPS

Serta, berikut adalah tabel pengujian daya yang dibutuhkan saat menjalankan sistem.

Tabel 10. Hasil evaluasi daya

No.	Percobaan	FPS
1	SSDLite Mobilenet V2 Fine Tuning	7.2 – 7.9 W
2	SSDLite Mobilenet V2 Tanpa Fine Tuning	7.3 – 8.0 W
3	SSD Mobilenet V2 Fine Tuning	7.3 – 8.0 W
4	SSD Mobilenet V2 Tanpa Fine Tuning	7.7 – 8.1 W
5	SSDMNV2	8.1 – 9.2 W

3.2 Diskusi

Hasil evaluasi deteksi bisa terdapat perbedaan karena adanya berbeda dataset, posisi masker salah pada sistem yang menggunakan deteksi objek satu tahap dan deteksi dua tahap berbeda. Pada sistem yang menggunakan deteksi objek satu tahap posisi masker tidak menutup hidung, namun masih menutup mulut. Selain itu, maka deteksinya bukan salah. Sementara pada sistem yang menggunakan deteksi objek dua tahap, posisi masker tidak menutupi hidung dan mulut. Selain itu, maka deteksinya bukan salah.

Berdasarkan tabel 9, hasil evaluasi FPS di atas terlihat bahwa *SSDLite Mobilenet V2* memiliki FPS yang lebih tinggi dibandingkan dengan model lainnya serta sistem deteksi penggunaan masker yang menggunakan dua tahap yang memiliki FPS paling kecil dibandingkan dengan yang lainnya. Hasil ini juga menunjukkan sistem deteksi satu tahap memang benar lebih cepat jika dibandingkan dengan sistem deteksi yang menggunakan dua tahap dalam mendeteksinya.

Berdasarkan tabel 10, daya yang dibutuhkan untuk menjalankan sistem relatif sama untuk setiap percobaan. Hanya saja terlihat perbedaan daya maksimal yang dibutuhkan pada saat percobaan dengan model SSDMNV2. Naik turunnya daya juga berkaitan dengan kipas yang ada pada sistem, kipas tersebut akan menyala, ketika sistem menyentuh suhu 60 derajat Celsius dan akan berhenti ketika suhu di bawah itu.

4. KESIMPULAN

Dari penelitian yang telah peneliti uraikan, maka dapat ditarik kesimpulan bahwa sistem dapat mendeteksi penggunaan masker yang benar sesuai rekomendasi kementerian kesehatan, yaitu dapat memastikan masker medis atau kain mampu menutupi mulut dan hidung. Setelah dilakukan percobaan, sistem dengan model *SSDLite Mobilenet V2* memiliki FPS yang paling tinggi dibandingkan dengan sistem yang menggunakan sistem dengan SSDMNV2. Yaitu FPS maksimal yang didapatkan adalah 3.57 FPS dan minimal FPS 3.43 FPS. Artinya semakin banyak gambar dan semakin cepat transisi antara satu gambar dengan gambar berikutnya, semakin nyata gerakan dari gambar-gambar tadi. Kemampuan deteksi yang diperoleh dengan model *SSDLite Mobilenet V2* dapat mendeteksi menutup wajah dengan koran terdeteksi tanpa masker, sementara pada penelitian [17] menutup wajah dengan koran masih terdeteksi menggunakan masker.

REFERENCES

- [1] P. D. H. Tandra, VIRUS CORONA BARU COVID-19: Kenali, Cegah, Lindungi Diri Sendiri & Orang Lain. Rapha Publishing, 2021. [Online]. Available: <https://books.google.co.id/books?id=-7AfEAAAQBAJ>
- [2] Kementerian Kesehatan, “QnA : Pertanyaan dan Jawaban Terkait COVID-19,” 2020. <https://infeksiemerging.kemkes.go.id/uncategorized/qna-pertanyaan-dan-jawaban-terkait-covid-19> (accessed Feb. 23, 2021).
- [3] Kementerian Kesehatan, “Kemenkes Sarankan 3 Jenis Masker untuk Dipakai,” 2020. <https://sehatnegeriku.kemkes.go.id/baca/umum/20200921/2434977/kemenkes-sarankan-3-jenis-masker-dipakai/> (accessed Feb. 05, 2021).
- [4] CDC, “Type of Masks,” 2021. <https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/types-of-masks.html#print> (accessed Feb. 24, 2021).
- [5] S. Verma, M. Dhanak, and J. Frankenfield, “Visualizing droplet dispersal for face shields and masks with exhalation valves,” *Physics of Fluids*, vol. 32, no. 9, 2020, doi: 10.1063/5.0022968.
- [6] HUFFPOST, “The Face Mask Mistakes People Make Without Realizing It,” 2021. https://www.huffpost.com/entry/face-mask-mistakes-covid-19_1_5ff35c27c5b65a92291089df (accessed Feb. 05, 2021).
- [7] klikdokter, “Benarkah Mengenakan Masker di Daggu Bisa Membahayakan Kesehatan?,” 2020. <https://www.klikdokter.com/info-sehat/read/3641494/benarkah-mengenakan-masker-di-daggu-bisa-membahayakan-kesehatan> (accessed Feb. 05, 2021).
- [8] M. Elgendy, *Deep Learning for Vision Systems*. Manning Publications Co, 2020.
- [9] Darmasita, “Deteksi Penggunaan Masker Menggunakan Xception Transfer Learning,” *JURNAL INSTEK (Informatika Sains dan Teknologi)*, vol. 5, pp. 279–288, 2020, doi: <https://doi.org/10.24252/instek.v5i2.20132>.
- [10] M. M. Lambacing and F. Ferdiansyah, “Rancang Bangun New Normal Covid-19 Masker Detektor Dengan Notifikasi Telegram Berbasis Internet of Things,” *Dinamik*, vol. 25, no. 2, pp. 77–84, 2020, doi: 10.35315/dinamik.v25i2.8070.
- [11] D. Giancini, E. Y. Puspaningrum, and Y. V. Via, “Identifikasi Penggunaan Masker Menggunakan Algoritma CNN YOLOv3-Tiny,” *Seminar Nasional Informatika Bela Negara (SANTIKA)*, vol. 1, pp. 153–159, 2020.



- [12] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection," *Sustainable Cities and Society*, vol. 65, no. June 2020, p. 102600, 2021, doi: 10.1016/j.scs.2020.102600.
- [13] X. Jiang, T. Gao, Z. Zhu, and Y. Zhao, "Real-time face mask detection method based on yolov3," *Electronics (Switzerland)*, vol. 10, no. 7, pp. 1–17, 2021, doi: 10.3390/electronics10070837.
- [14] S. v. Militante and N. v. Dionisio, "Real-Time Facemask Recognition with Alarm System using Deep Learning," 2020 11th IEEE Control and System Graduate Research Colloquium, ICSGRC 2020 - Proceedings, no. August, pp. 106–110, 2020, doi: 10.1109/ICSGRC49013.2020.9232610.
- [15] G. Aprilian Anarki, K. Auliasari, and M. Orisa, "Penerapan Metode Haar Cascade Pada Aplikasi Deteksi Masker," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 5, no. 1, pp. 179–186, 2021, doi: 10.36040/jati.v5i1.3214.
- [16] A. Thariq and R. Yusliana, "Sistem Deteksi Masker dengan Metode Haar Cascade pada Era New Normal COVID-19 Mask Detection System using Haar Cascade Method in New Normal Era of COVID-19," *Jurnal Sistem dan Teknologi Informasi*, vol. 9, no. 2, pp. 241–244, 2021, doi: 10.26418/jstin.v9i2.44309.
- [17] M. Abdul, R. Irham, and D. A. Prasetya, "PROTOTIPE PENDETEKSI MASKER PADA RUANGAN WAJIB MASKER UNTUK KENDALI PINTU OTOMATIS BERBASIS DEEP LEARNING SEBAGAI PENCEGAHAN PENULARAN COVID-19," *Symposium Nasional RAPI XIX Tahun 2020 FT UMS*, pp. 47–55, 2020, [Online]. Available: <http://hdl.handle.net/11617/12377>
- [18] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, and J. Hemanth, "SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2," *Sustainable Cities and Society*, vol. 66, no. December 2020, p. 102692, 2021, doi: 10.1016/j.scs.2020.102692.
- [19] A. Cabani, K. Hammoudi, H. Benhabiles, and M. Melkemi, "MaskedFace-Net – A dataset of correctly/incorrectly masked face images in the context of COVID-19," *Smart Health*, vol. 19, no. November 2020, p. 100144, 2021, doi: 10.1016/j.smhl.2020.100144.