

Sistem Deteksi Masker Berbasis Jetson Nano dengan Deep Learning Framework TensorFlow

Muhammad Luqman Bukhori¹, Erwan Eko Prasetyo²

¹ Teknik Dirgantara, Sekolah Tinggi Teknologi Kedirgantaraan, Jl. Parangtritis KM 4,5 Druwo, Bangunharjo, Sewon, Bantul, 55143 (tel.: (0274) 418 248, 439 6163; email: m.luqman@sttkd.ac.id)

² Aeronautika, Sekolah Tinggi Teknologi Kedirgantaraan, Jl. Parangtritis KM 4,5 Druwo, Bangunharjo, Sewon, Bantul, 55143 (tel.: (0274) 418 248, 439 6163; email: erwan.eko@sttkd.ac.id)

[Diterima: 8 September 2022, Revisi: 20 Oktober 2022]

Corresponding Author: Muhammad Luqman Bukhori

INTISARI — Indonesia merupakan salah satu negara yang mengalami dampak virus COVID-19. Segala upaya dilakukan untuk menanggulangi penyebaran virus ini. Salah satu langkah yang efisien guna menanggulangi dampak ini adalah dengan menerapkan protokol kesehatan yang ketat dan pemakaian masker dengan benar. Pemantauan penggunaan masker terus dilakukan di gedung perkantoran, supermarket, maupun ruang publik lainnya. Peran pengawas sangat diperlukan untuk mengawasi pemakaian masker dengan benar. Namun, seorang pengawas mempunyai keterbatasan dalam melakukan pengawasan, sehingga menimbulkan celah bagi orang-orang untuk tidak mematuhi aturan pemakaian masker dengan benar. Oleh karena itu, perlu adanya sistem yang bekerja secara otomatis untuk membantu pengawas dalam memantau pemakaian masker dengan benar. Tujuan makalah ini adalah merancang sebuah *computer vision* yang dapat mendeteksi seseorang bermasker atau tidak menggunakan *deep learning framework* TensorFlow. TensorFlow digunakan karena efisiensinya dalam mengolah data citra digital. Klasifikasi data citra digital pada TensorFlow menggunakan struktur *deep learning* Keras, sehingga ringan dan mampu digunakan pada perangkat tertanam seperti Jetson Nano untuk mendeteksi penggunaan masker secara *real-time*. Tahapan sistem pendeteksian masker terdiri atas pengumpulan *dataset* citra, ekstraksi ciri, pemisahan data, pembuatan model, pelatihan model, dan penerapan model. *Deep learning framework* TensorFlow memproses data citra secara langsung melalui kamera web (*webcam*). Ketika kamera menangkap objek orang yang sedang tidak menggunakan masker dengan benar, layar monitor menampilkan kotak merah pada wajah. Tanda ini dapat membantu pengawas saat melakukan pengawasan. Hasil pengujian menunjukkan bahwa sistem berhasil mendeteksi dengan benar orang yang tidak menggunakan masker, dengan akurasi sebesar 99,48%. Selain itu, sistem juga berhasil mendeteksi orang yang memakai masker dengan benar dan mendapatkan rata-rata akurasi sebesar 99,12%. Monitor akan menampilkan kotak hijau pada wajah jika orang yang dideteksi memakai masker dengan benar.

KATA KUNCI — *Computer Vision*, Jetson Nano, Deteksi Masker, *Deep Learning*, Keras, *Framework*, TensorFlow.

I. PENDAHULUAN

COVID-19 adalah penyakit menular yang disebabkan oleh virus SARS-CoV-2. Orang yang tertular COVID-19 akan mengalami gejala ringan hingga sedang dan dapat pulih tanpa perawatan khusus. Namun, beberapa orang dapat menjadi sakit parah dan memerlukan bantuan medis, bahkan sampai meninggal [1]. Sementara itu, jumlah kasus infeksi COVID-19 semakin meningkat. Data per akhir Juli 2022 menunjukkan bahwa kasus COVID-19 di Indonesia telah mencapai lebih dari 6 juta kasus, sedangkan secara global telah terjadi lebih dari 550 juta kasus [2]. Virus COVID-19 dapat menyebar dari mulut atau hidung orang yang terinfeksi melalui partikel kecil ketika orang tersebut batuk, bersin, berbicara, bernyanyi, atau bernapas.

Penyebaran virus COVID-19 dapat dikendalikan dengan menerapkan protokol kesehatan, antara lain menjaga jarak, memakai masker, sering mencuci tangan, menghindari kerumunan, melakukan vaksinasi, dan menjaga kebersihan. Penggunaan masker medis sangat efektif dalam mencegah penularan COVID-19 karena dapat menyaring partikel hingga 95% [3]. Namun, sebuah penelitian menunjukkan bahwa banyak orang mengabaikan atau enggan menggunakan masker di area publik hingga di dalam ruangan [4]. Sikap masyarakat yang tidak mematuhi penerapan protokol kesehatan ini perlu mendapatkan perhatian. Situasi tersebut dapat membahayakan orang lain karena meningkatkan risiko penularan virus COVID-19 [5]. Upaya pencegahan telah dilakukan dengan menyediakan tempat cuci tangan, memasang kamera termal untuk memeriksa suhu tubuh, serta menempatkan petugas

khusus untuk mengawasi penggunaan masker dan penegakan protokol kesehatan [6]. Keterbatasan petugas pengawas yang memantau penggunaan masker tiap orang menyebabkan masih adanya kemungkinan masyarakat mengabaikan aturan pemakaian masker di area publik, khususnya di dalam ruangan.

Dengan mempertimbangkan beberapa hal terkait pengendalian protokol kesehatan di area publik, diperlukan sebuah sistem kecerdasan buatan untuk mendeteksi orang bermasker secara *real-time*. *Computer vision* merupakan jawaban untuk sistem tersebut karena dapat digunakan untuk tugas-tugas seperti pengenalan objek, pelacakan visual, segmentasi semantik, hingga restorasi gambar [7]. Pada penelitian sebelumnya, telah diterapkan *computer vision* pada sistem deteksi objek manusia untuk membangun interaksi robot dengan manusia [8]. Data dideteksi berdasarkan bagian tubuh, seperti kepala, bahu, lengan, dan kaki, dengan beberapa kondisi gerak tubuh. Hasil eksperimen memperlihatkan bahwa dua *dataset* (INRIA dan Caltech) menunjukkan efektivitas yang baik sesuai yang ditentukan sebelumnya. Selain itu, telah dilakukan juga deteksi objek pejalan kaki berganda secara *real-time* [9]. Metode yang diusulkan memutuskan bahwa objek yang terdeteksi dalam bingkai saat ini sama dengan yang ada di bingkai sebelumnya atau tidak, serta memperbarui koordinat beberapa objek dan histogram yang sesuai selama proses pelacakan. Hasil eksperimen menunjukkan bahwa metode pelacakan objek ganda yang diusulkan mengungguli metode yang ada dalam situasi yang menantang dengan oklusi parsial.

Di era pandemi COVID-19, telah dilakukan penelitian yang mengembangkan sistem deteksi objek untuk mendeteksi orang

yang mengenakan masker wajah [10]. Deteksi masker wajah yang dibangun menggunakan teknik *deep learning*. Teknik ini telah terbukti mendapatkan akurasi hingga 98%. Referensi [11] menyatakan bahwa dalam melakukan identifikasi orang bermasker perlu dikumpulkan data citra hingga ribuan dan perlu dilakukan pemisahan kelas data citra berwajah. Sebuah sistem lain dikembangkan menggunakan *convolutional neural network* (CNN) sederhana dengan modul Raspberry Pi 4 [12]. Hasil percobaan menunjukkan bahwa dengan menggunakan CNN, dapat dilakukan pengenalan wajah bermasker dengan akurasi 97,67%. CNN merupakan salah satu jenis arsitektur *deep neural network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra [13]. Beberapa jenis arsitektur *deep neural network* lainnya adalah *artificial neural network* (ANN) [14], *recurrent neural network* (RNN) [15], *long short term memory network* (LSTM), serta *self-organizing maps* (SOM). Arsitektur ini memiliki beberapa tantangan, termasuk ketidakjelasan jaringan saraf, jaminan kualitas data, keamanan data, dan kelas produksi kecerdasan buatan [16]. Masing-masing memiliki kekurangan dan kelebihan tersendiri, tetapi arsitektur yang cocok untuk *computer vision* dalam mengolah citra (gambar atau video) adalah CNN [17] karena mampu mengklasifikasikan bagian terkecil *node* yang saling terhubung [18], dan telah dipakai oleh beberapa *deep learning framework*, salah satunya adalah TensorFlow [19]. TensorFlow telah dikemas menjadi *library* Python yang dapat difungsikan dalam *machine learning* [20], karena mempunyai sumber daya yang kompleks dan waktu yang relatif singkat untuk mengidentifikasi orang bermasker atau tidak bermasker. Referensi [21] menyimpulkan bahwa Tensorflow-Keras dapat mendeteksi wajah bermasker dengan akurasi sebesar 93% dengan menggunakan perangkat modul Raspberry Pi 4. Berdasarkan analisis yang dilakukan, terdapat kekurangan kinerja dari perangkat yang digunakan. Hasil *benchmark* menunjukkan bahwa Raspberry Pi hanya dapat mencapai 8.1 *frame per second* (FPS), sedangkan perangkat lainnya, yaitu NVIDIA Jetson Nano, dapat mencapai 11,4 FPS [22].

Terdapat berbagai macam metode dalam penelitian sistem deteksi masker. Pengujian deteksi masker pada wajah berhasil telah dilakukan dengan baik menggunakan algoritme YOLO [23]. YOLO sendiri merupakan algoritme yang menggunakan metode *single-stage detection* (SSD), berbeda dengan *multi-stage detection* (MSD) yang dimiliki oleh CNN [24], R-CNN, Faster R-CNN [25], dan Mask R-CNN. Adapun penerapannya pada *machine learning* menggunakan PC mini seperti modul Raspberry Pi [26], tetapi hasilnya kurang efektif dan kecepatan deteksinya sangat lambat karena hanya mengandalkan *central processing unit* (CPU) saat pemrosesannya. Penelitian ini bertujuan merancang sistem deteksi penggunaan masker menggunakan Jetson Nano, yang merupakan alternatif PC mini yang berbasis *graphical processor unit* (GPU) internal.

Makalah ini menyajikan teknik pembelajaran *computer vision* yang dibantu dengan *library* OpenCV, Keras, dan TensorFlow. *Computer vision* yang dikembangkan menggunakan perangkat tertanam NVIDIA Jetson Nano Developer Kit. Perangkat ini mempunyai sebuah GPU yang tertanam di dalamnya untuk mengolah dan memproses segala data citra digital. Perangkat ini mempunyai ukuran yang kecil sehingga dapat diletakkan di berbagai lokasi untuk merekam video secara *real-time*. Implementasi ini dibantu dengan kamera Logitech C920 sebagai sumber data citra yang berkualitas tinggi. Data citra yang ditangkap oleh kamera akan diproses menggunakan *library* Python yaitu TensorFlow dan

Keras. TensorFlow dan Keras ini akan mengolah semua data citra video yang masuk dan mengidentifikasi setiap orang yang tidak memakai masker dengan benar.

Tujuan utama penelitian ini adalah: 1) mengembangkan sebuah *computer vision* yang dapat mendeteksi orang bermasker atau tidak; 2) menerapkan metode identifikasi yang menghasilkan informasi visual berupa kotak merah bagi orang yang tidak bermasker dan kotak hijau bagi yang menggunakan masker dengan benar; 3) dapat mengidentifikasi jumlah orang yang terdeteksi memakai masker atau tidak; dan 4) melakukan pengujian deteksi masker menggunakan jenis masker yang berbeda warna.

Penelitian ini menyajikan empat bagian utama sebagai berikut. Bagian pendahuluan menjelaskan latar belakang masalah, kesenjangan penelitian, dan tujuan penelitian. Kemudian, pada bagian metodologi ditunjukkan solusi permasalahan penelitian terkait sistem deteksi pada perangkat tertanam dan tahapan *deep learning framework*. Bagian hasil dan analisis menjelaskan pengujian deteksi menggunakan model dan melalui kamera web secara langsung, sedangkan kesimpulan menyampaikan hasil temuan yang didapatkan berdasarkan analisis yang dilakukan.

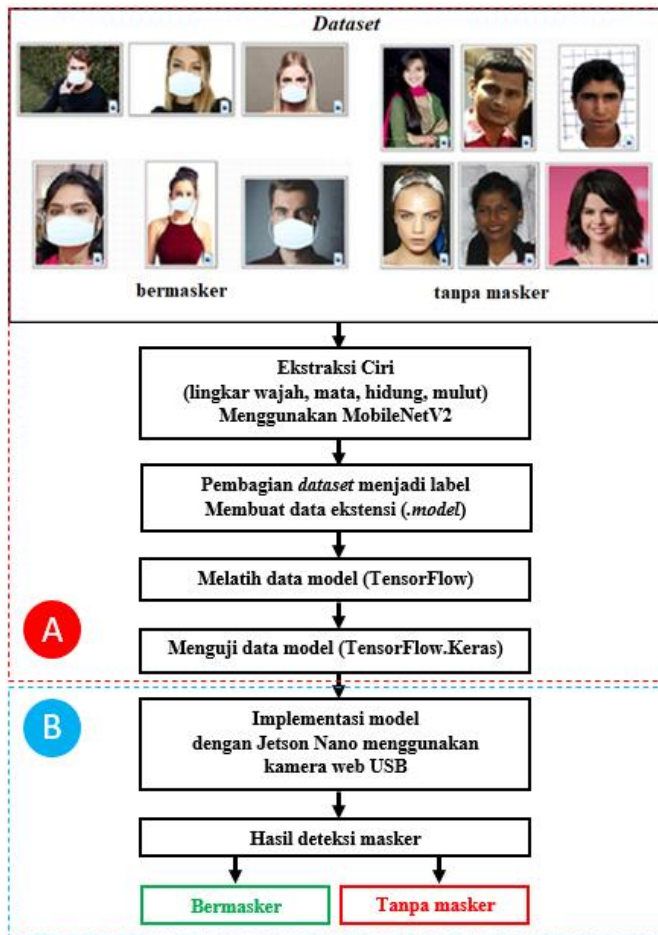
II. METODOLOGI

CNN adalah bagian dari suatu jaringan saraf tiruan yang umum digunakan dalam pengenalan dan pemrosesan data citra [27]. Algoritme CNN memiliki neuron yang didesain untuk bekerja layaknya lobus frontal, khususnya area *visual cortex* pada otak manusia dan hewan [28]. Pada penelitian sebelumnya, *computer vision* digunakan untuk mengekstraksi karakteristik secara manual agar dicapai klasifikasi yang lebih baik. Penggunaan CNN dapat mencapai tujuan klasifikasi dengan menghasilkan lapisan konvolusi, pengecilan dimensi, hingga ekstraksi ciri wajah.

A. MODEL YANG DIKEMBANGKAN

Algoritme pembelajaran yang diterapkan ini menggunakan model MobileNetV2. MobileNetV2 merupakan pengembangan model arsitektur CNN sebelumnya yang bernama MobileNetV1. MobileNet merupakan salah satu arsitektur CNN yang didesain untuk mengatasi kebutuhan akan komputasi yang tinggi [29]. Perbedaan yang mendasar antara arsitektur MobileNet dengan arsitektur CNN lainnya adalah penggunaan lapisan (*layer*) konvolusi dengan ketebalan filter hampir sama dengan citra masukannya. MobileNet membagi konvolusi menjadi *depthwise convolution* dan *pointwise convolution*.

Depthwise convolution dan *pointwise convolution* merupakan gagasan kedalaman dan dimensi spasial filter yang dapat dipisahkan. Sebagai contoh, filter Sobel digunakan dalam pemrosesan gambar untuk mendeteksi tepi. Filter yang digunakan memiliki sembilan parameter dengan konvolusi 1×1 terhadap segi salurannya. MobileNetV2 memiliki dua fitur tambahan, yaitu: 1) *linear bottleneck*, dan 2) *shortcut connections* antar *bottlenecks*. Referensi [30] menyebutkan bahwa pada bagian *bottleneck* terdapat masukan dan keluaran antar model, sedangkan lapisan bagian dalam melakukan enkapsulasi kemampuan model untuk mengubah masukan dari konsep tingkat yang lebih rendah ke *descriptor* tingkat yang lebih tinggi. Seperti halnya koneksi *residual* pada CNN tradisional, *shortcut bottleneck* memungkinkan pelatihan yang lebih cepat dan akurasi yang lebih baik. Model yang dikembangkan dimulai dengan memuat *dataset* untuk deteksi



Gambar 1. Diagram alir yang dikembangkan.

masker. Library deep learning Python yang digunakan untuk persiapan data adalah OpenCV, Keras, dan TensorFlow. Library ini difungsikan untuk melatih klasifikasi dengan model MobileNetV2, seperti yang diperlihatkan pada Gambar 1.

Terdapat dua tahapan dalam pengenalan model, yaitu pelatihan (*training*) dengan library TensorFlow dan pengujian (*testing*) dengan library TensorFlow.Keras [31]. Sebelum pelatihan dimulai, semua *dataset* perlu diekstrak terlebih dahulu berdasarkan fitur wajah, yaitu lingkaran wajah, mata, hidung, dan mulut. *Dataset* yang sudah dilatih langsung disimpan sebagai data model dengan ekstensi *.model*. Pada tahapan percobaan, model yang telah disimpan sebelumnya dipanggil dengan library TensorFlow.Keras. Implementasi model diterapkan pada perangkat tertanam Jetson Nano melalui kamera web untuk membaca citra video secara *real-time*. Jika seseorang tertangkap kamera memakai masker, hasil menunjukkan bermasker. Sebaliknya, jika tidak, hasil menunjukkan tanpa masker.

B. DATASET CITRA

Sistem deteksi yang dikembangkan mempunyai beberapa tahapan. Tahap awal sistem adalah mengumpulkan *dataset* berupa citra orang yang bermasker dan orang tidak bermasker. *Dataset* yang dikumpulkan merupakan beberapa data yang sudah ada di internet dan data citra tambahan dari seorang model. Tidak ada standar jumlah yang ditentukan dalam penentuan *dataset* ini. Hal terpenting dari *dataset* adalah mengategorikan jenis data citra yang dipakai, memakai masker menutupi hidung, atau tidak menutupi hidung, atau tidak memakai masker sama sekali. Penelitian ini menggunakan total 1.360 data citra yang terdiri atas 680 data citra menggunakan

masker, 120 data citra menggunakan masker tanpa menutupi hidung, dan 560 data citra tanpa menggunakan masker. Setelah semua data terkumpul, data dibagi menjadi dua kelompok, yaitu bermasker dan tanpa masker. Kategori kelompok bermasker adalah data citra yang benar-benar menggunakan masker dengan menutupi mulut dan hidung. Kategori tanpa masker adalah data citra yang tanpa menggunakan masker serta citra yang menggunakan masker tetapi tidak menutupi hidung. Semua data citra yang dikumpulkan dapat dikenali oleh sistem.

C. EKSTRAKSI CIRI

Ekstraksi ciri adalah tahapan kedua dari model yang dikembangkan. Data citra diubah ukurannya menjadi 224×224 piksel, kemudian dikonversi ke data larik (*array*) menggunakan *encoding* MobileNetV2. Hasil yang didapatkan adalah titik koordinat *region of interest* (ROI) wajah pada data citra yang diperoleh dengan kalkulasi ciri lingkaran wajah, mulut, hidung, dan mata. Data yang telah diketahui koordinat ROI-nya kemudian disimpan ke dalam larik untuk digunakan pada fungsi *loop*. Data disimpan ke dalam label sesuai kategori yang telah ditetapkan sebelumnya. Berikut merupakan cuplikan *script* untuk membuat label data.

```

# perform encoding on the labels
lb = LabelBinarizer()
labels = lb.fit_transform(labels)
labels = to_categorical(labels)
    
```

Karena data larik hanya dapat membaca data angka, kategori yang telah ditentukan dikonversi ke dalam bentuk angka. Angka 0 merupakan kategori label tanpa masker, sedangkan angka 1 merupakan kategori label untuk data citra bermasker.

D. PEMISAHAN DATA DAN PEMBUATAN MODEL

Tahap ini merupakan satu kesatuan tahapan yang kedua. Total data yang diujikan pada tahap ini adalah 20%, sedangkan 80% sisanya digunakan untuk pelatihan data. Pembuatan model mempunyai beberapa tahapan, yaitu melatih generator citra, penambahan parameter model, kompilasi, pelatihan model, dan penyimpanan model untuk prediksi lebih lanjut dengan perangkat Jetson Nano. Berikut ini adalah cuplikan program pembuatan label dari *dataset* untuk melatih generator citra.

```

# partition the data into splits using 80%
# training and 20% for testing
(trainX, testX, trainY, testY) =
    train_test_split(data, labels,
                    test_size=0.20, stratify=labels,
                    random_state=42)

# construct the training image generator
# for augmentation
aug = ImageDataGenerator(rotation_range=20,
                          zoom_range=0.15, width_shift_range=0.2,
                          shear_range=0.15, height_shift_range=0.2,
                          horizontal_flip=True,
                          fill_mode="nearest")
    
```

E. PELATIHAN MODEL

Pelatihan model dilakukan untuk mendapatkan satu berkas berbentuk *namafilename.model*. Berkas ini berfungsi sebagai model pendeteksi wajah untuk menentukan bahwa citra yang diambil dari kamera merupakan citra yang bermasker atau bukan. Pelatihan dilakukan melalui perangkat Jetson Nano Developer Kit 4GB, langsung dengan beberapa pengaturan. Jumlah iterasi ditentukan sebanyak 50 iterasi dan *batch sample* (BS) sebesar 32 data. Hasil evaluasi model disajikan pada Tabel I dan



Gambar 2. Penampilan grafik training loss dan akurasi model.

TABEL I
EVALUASI JARINGAN MODEL

	Precision	Recall	F1-score	Support
Bermasker	93%	100%	99%	413
Tidak bermasker	100%	93%	97%	498
Rata-rata	97%	97%	98%	
Total				911

ditunjukkan pula dengan tampilan grafik pada Gambar 2. Untuk menghitung evaluasi hasil pelatihan model, dapat digunakan (1)-(3) [32].

$$Precision = \frac{[TP]}{[TP + FP]} \quad (1)$$

dengan TP adalah *true positive*, TN adalah *true negative*, dan FP adalah *false positive*.

Presisi merupakan metrik yang menunjukkan jumlah nilai positif yang diharapkan atau nilai yang sebenarnya adalah benar. Sebagai contoh, jika TP bernilai 1 dan FP bernilai 1, hasil yang didapatkan untuk nilai sebenarnya adalah 0,5. Statistik *recall* digunakan sebagai skala kuantitas sebuah kemampuan algoritme dalam mengklasifikasikan semua kasus positif, untuk mempermudah mengingat proporsi positif yang diidentifikasi dengan benar. Secara matematis, *recall* dituliskan pada (2).

$$Recall = \frac{[TP]}{[TP + FN]} \quad (2)$$

dengan FN adalah *false negative*.

$$F1\ score = 2 \times \frac{[Precision \times Recall]}{[Precision + Recall]} \quad (3)$$

Akurasi tes $F1\ score$ lalu dikuantifikasikan. Langkah evaluasi ini memberikan temuan yang paling akurat terhadap *dataset* yang seimbang. Tahapan ini merupakan tahap verifikasi terhadap prediksi yang akurat. Nilai positif pada persamaan menandakan citra yang masuk pada label benar-benar sesuai dengan prediksi label yang telah ditentukan. Sementara itu, citra negatif sebenarnya adalah citra sesuai dengan kategori benar, tetapi mendapatkan prediksi yang salah.

Gambar 2 memperlihatkan hasil evaluasi pelatihan model terhadap akurasi dengan data yang tidak dapat terbaca. Grafik menunjukkan bahwa data akurasi memperlihatkan nilai antara 0,8 hingga 1. Akurasi yang didapatkan sangat bagus karena nilainya lebih dari 0,8. *Data loss* merupakan data yang tidak boleh terlihat di atas angka 0,2. Gambar 2 menunjukkan bahwa *data loss* hampir menuju nilai 0. Hasil ini membuktikan bahwa pelatihan dapat mengidentifikasi citra model dengan benar.



Gambar 3. Hasil pengujian deteksi masker terhadap model menggunakan masker, (a) tanpa menutup hidung, (b) menutup hidung.

F. IMPLEMENTASI MODEL

Implementasi model diterapkan pada perangkat Jetson Nano. Perangkat ini merupakan komputer mini portabel dengan ukuran yang relatif kecil. Jetson Nano dibuat oleh NVIDIA dengan tujuan untuk pengembangan sistem tertanam yang kecil dan tangguh dalam menjalankan beberapa aplikasi jaringan saraf, seperti klasifikasi citra, deteksi objek, segmentasi, dan pemrosesan ucapan [33].

Spesifikasi Jetson Nano yang digunakan adalah Jetson Nano Developer Kit 4GB. Jetson Nano merupakan kit dengan harga yang terjangkau, tetapi memiliki kualitas yang mumpuni dalam pengolahan citra, berbekal inti GPU 128 *core* Maxwell, CPU *quad-core* ARM A57 @1,43GHz, dengan kemampuan *encode* dan *decode* video maksimum 1080p (FHD).

Implementasi model pada perangkat sistem deteksi masker akan bekerja setelah kamera web aktif mendeteksi seseorang. Jika wajah terdeteksi, pemrosesan ulang dilakukan dengan melakukan *resize* ukuran piksel citra, konversi ke matriks larik, kemudian pemrosesan menggunakan MobileNetV2. Setelah itu, model masukan disimpan untuk memprediksi data banding dengan model yang telah dikenai proses pelatihan sebelumnya. Jika prediksi yang didapatkan lebih dominan memakai masker, hasil yang ditampilkan adalah kotak hijau dengan tulisan "Masker." Jika prediksi yang didapatkan lebih dominan tanpa memakai masker, hasil yang ditampilkan adalah kotak merah yang bertuliskan "No Masker".

III. HASIL DAN PEMBAHASAN

A. PENGUJIAN DETEKSI MASKER TERHADAP MODEL

Pengujian model dari *dataset* perlu dilakukan untuk membuktikan bahwa hasil prediksi yang diproses benar-benar menghasilkan data uji yang valid. Berdasarkan Gambar 3, hasil pengujian terhadap data model sebelumnya menunjukkan bahwa model yang menggunakan masker tanpa menutup hidung dikenali oleh sistem sebagai tidak bermasker. Hal ini sudah sesuai dengan metode yang diterapkan. Model yang menggunakan masker dengan menutupi hidung juga dikenali dengan benar, yaitu bermasker dengan benar.

Gambar 3(a) memperlihatkan bahwa citra model yang memakai masker tanpa menutupi hidung berhasil dideteksi oleh sistem, yaitu model tidak mengenakan masker, yang ditandai dengan kotak merah berlabel "No Masker." Pengujian dilakukan sebanyak lima kali dan mendapatkan akurasi dan presisi sebesar 99,18%. Algoritme yang dilatih berhasil mengisyaratkan bahwa setiap model yang tidak memakai masker ataupun memakai masker tetapi tidak menutupi hidung dianggap tidak memakai masker.

Gambar 3(b) memperlihatkan bahwa citra model yang memakai masker dengan menutupi hidung berhasil dideteksi oleh sistem dengan prediksi bahwa model menggunakan masker dengan benar, yang ditandai dengan kotak hijau



(a)



(b)

Gambar 4. Hasil pengujian deteksi masker langsung dengan model, (a) tanpa masker dan (b) bermasker tanpa menutupi hidung.

berlabel “Masker.” Pengujian dilakukan sebanyak lima kali dan mendapatkan akurasi dan presisi sebesar 99,85%.

B. PENGUJIAN DETEKSI MASKER MELALUI KAMERA LANGSUNG

Pada pengujian ini, model yang diuji merupakan model yang sebelumnya tidak digunakan sebagai *dataset* pelatihan. Dengan kata lain, model yang digunakan adalah orang lain. Pengujian deteksi masker pada kamera langsung dikategorikan dengan dua kriteria. Kriteria pertama adalah tanpa menggunakan masker dan kriteria kedua adalah menggunakan masker. Pada kriteria pertama, terdapat dua kondisi yang menyatakan orang tidak bermasker, yaitu ketika orang benar-benar tidak bermasker seperti pada Gambar 4(a) dan orang yang mengenakan masker tanpa menutupi hidung, seperti yang ditunjukkan Gambar 4(b).

Gambar 4(a) memperlihatkan hasil pengujian deteksi tanpa mengenakan masker dengan tiga model, yaitu satu laki-laki dan dua perempuan. Didapatkan hasil pengujian berupa dua model dinyatakan tidak bermasker dengan akurasi 100% dan satu model lainnya dinyatakan tidak bermasker dengan akurasi 99,70%. Pada kondisi ini, semua model yang diuji dinyatakan tidak bermasker, sesuai dengan keadaan yang sebenarnya. Data pengujian disajikan pada Tabel II.

Gambar 4(b) memperlihatkan hasil pengujian deteksi orang yang mengenakan masker tanpa menutupi hidung. Terdapat empat model yang dideteksi dan dinyatakan tidak bermasker dengan ditandai kotak warna merah. Dua model sebelah kiri (laki-laki dan perempuan) memberikan hasil akurasi sebesar 100% tidak bermasker, sedangkan dua model lainnya memberikan hasil akurasi yang berbeda, yaitu model ketiga memberikan hasil akurasi yang berbeda, yaitu model ketiga memberikan hasil akurasi sebesar 90,70% dan model keempat sebesar 62%. Hasil akurasi pada model keempat ini menandakan bahwa antara model dan objek yang dideteksi mempunyai kemiripan di bawah rata-rata saat proses dikalkulasi oleh sistem. Kedua kondisi pada Gambar 4(a) dan Gambar 4(b) yang telah diujikan berhasil menunjukkan bahwa semua model tidak mengenakan masker, walaupun kondisi sebenarnya mengenakan masker tetapi tanpa menutupi hidung, seperti yang diperlihatkan pada Tabel III.

Berdasarkan rata-rata akurasi prediksi deteksi model yang tidak menggunakan masker dan model yang menggunakan

TABEL II
 PENGUJIAN PREDIKSI DETEKSI TANPA MASKER

No.	Jumlah Model	Pengujian Deteksi Langsung Kamera		
		Jumlah Wajah	Prediksi Deteksi	Akurasi
1	1 orang	1 kotak merah	No Masker	100,00%
2	2 orang	2 kotak merah	No Masker	100,00%
3	3 orang	3 kotak merah	No Masker	99,93%
4	4 orang	4 kotak merah	No Masker	99,90%
5	5 orang	5 kotak merah	No Masker	100,00%
Rata-rata akurasi				99,97%

masker tanpa menutupi hidung, diperoleh total rata-rata akurasi sebesar 99,48%. Semua pengujian menunjukkan bahwa model yang dideteksi secara langsung menggunakan kamera dan model yang telah melalui pelatihan sebelumnya sama-sama menunjukkan prediksi yang tepat, yaitu hasil deteksi “No Masker.”

Pengujian selanjutnya terlihat pada Tabel IV, yaitu semua model mengenakan masker dengan benar hingga menutupi hidung. Pengujian ini dilakukan sebanyak lima kali pada masing-masing jumlah model. Pengujian pertama dilakukan oleh satu model dengan memakai masker warna putih. Hasil eksperimen menunjukkan bahwa sistem dapat mendeteksi orang yang memakai masker dengan benar dan rata-rata akurasi deteksi yang diperoleh sebesar 99,80%.

Pengujian kedua dilakukan oleh dua model dengan memakai masker warna putih dan abu-abu. Hasil pengujian menunjukkan bahwa sistem dapat mendeteksi orang yang memakai masker, yang ditandai dengan kotak warna hijau, dengan rata-rata akurasi sebesar 99,84%.

Pengujian ketiga dilakukan oleh tiga model dengan tiga warna masker yang berbeda. Hasil yang didapatkan menunjukkan bahwa sistem dapat mendeteksi masker dengan benar, dengan rata-rata akurasi sebesar 98,87%. Pengujian keempat dilakukan oleh empat model, tetapi dua model di antaranya menggunakan masker dengan warna yang sama, yaitu warna putih. Pengujian yang dilakukan berhasil mendeteksi dengan rata-rata akurasi sebesar 98,87%.

Pengujian kelima dilakukan oleh lima orang model dengan menerapkan empat warna masker yang berbeda. Warna masker yang digunakan adalah warna putih digunakan oleh dua orang, warna abu-abu satu orang, warna hitam satu orang, dan warna biru muda satu orang. Hasil pengujian menunjukkan bahwa sistem berhasil mendeteksi masker, dengan ditandai munculnya lima buah kotak hijau, dengan rata-rata akurasi deteksi sebesar 98,54%. Persentase keberhasilan ini didapatkan dari model yang terdiri atas dua model laki-laki dan tiga model perempuan. Masing-masing model mengenakan masker yang berbeda warna. Model pertama dinyatakan bermasker dengan akurasi sebesar 68,56%, sedangkan pada model lainnya akurasi yang didapatkan lebih dari 90%. Berdasarkan hasil pengujian yang dilakukan, semua model dapat terdeteksi mengenakan masker, walaupun masker yang digunakan mempunyai warna yang berbeda-beda. Berdasarkan kalkulasi dari lima kali pengujian yang dilakukan, rata-rata akurasi yang didapatkan pada sistem adalah 99,12%.

Hasil pengujian deteksi dari semua kategori yang telah ditetapkan sebelumnya berhasil dilakukan. Pengujian selanjutnya adalah mendeteksi model yang bermasker dan tidak bermasker yang berada dalam satu kondisi, yaitu bersamaan di depan kamera. Model yang bermasker dan tidak bermasker dapat diidentifikasi dengan benar. Terdapat lima

TABEL III
PENGUJIAN PREDIKSI DETEKSI MENGGUNAKAN MASKER TANPA MENUTUPI HIDUNG

No	Kondisi Wajah	Jumlah Model	Jumlah Warna Masker	Penguji Deteksi Langsung Kamera		
				Jumlah Wajah	Prediksi Deteksi	Akurasi
1	Bermasker tanpa menutup hidung	1 orang	1 warna	1 kotak merah	No Masker	100,00%
2		2 orang	2 warna	2 kotak merah	No Masker	100,00%
3		3 orang	3 warna	3 kotak merah	No Masker	98,80%
4		4 orang	3 warna	4 kotak merah	No Masker	98,18%
5		5 orang	4 warna	5 kotak merah	No Masker	98,00%
Rata-rata akurasi						98,99%

TABEL IV
PENGUJIAN PREDIKSI DETEKSI MENGGUNAKAN MASKER MENUTUPI HIDUNG

No	Kondisi Wajah	Jumlah Model	Jumlah Warna Masker	Penguji Deteksi Langsung Kamera		
				Jumlah Wajah	Prediksi Deteksi	Akurasi
1	Bermasker menutup hidung	1 orang	1 warna	1 kotak hijau	Masker	99,80%
2		2 orang	2 warna	2 kotak hijau	Masker	99,84%
3		3 orang	3 warna	3 kotak hijau	Masker	98,87%
4		4 orang	3 warna	4 kotak hijau	Masker	98,55%
5		5 orang	4 warna	5 kotak hijau	Masker	98,54%
Rata-rata akurasi						99,12%

model pada pengujian ini, yaitu dua model laki-laki dan tiga model perempuan. Pengujian berhasil mendeteksi salah satu model perempuan tidak mengenakan masker, dengan akurasi sebesar 100%. Sementara itu, empat model lainnya terdeteksi menggunakan masker dengan benar.

Sistem yang dikembangkan menggunakan Jetson Nano ini dapat mengolah data citra langsung dengan waktu proses mencapai 0,114 detik. Data ini didapatkan dari rata-rata setiap FPS citra dari kamera web. Kecepatan proses ini jauh lebih tinggi dibandingkan dengan proses pada perangkat komputer mini lainnya. Sebagai contoh, penelitian sebelumnya yang menggunakan komputer mini Raspberry Pi 4 versi 4GB membutuhkan waktu proses kurang lebih 7 detik. Oleh karena itu, dapat dikatakan bahwa perangkat Jetson Nano ini lebih bagus kinerjanya untuk mengolah dan mengidentifikasi objek secara langsung.

IV. KESIMPULAN

Telah berhasil dirancang dan dibuat sebuah perangkat, menggunakan Jetson Nano, untuk mendeteksi objek atau orang yang bermasker dan tidak bermasker. Pengujian menunjukkan bahwa model yang diuji, yang terdiri atas laki-laki dan perempuan, berhasil dideteksi dengan *deep learning framework* TensorFlow. Pengujian model tidak bermasker berhasil dideteksi dengan rata-rata akurasi sebesar 99,48%, dengan jumlah model yang diuji sebanyak lima orang. Pengujian model bermasker juga berhasil dideteksi dengan rata-rata akurasi sebesar 99,12% dari lima orang yang dideteksi. Masker yang dipakai oleh model terdiri atas warna hitam, abu-abu, putih, dan biru muda. Semua masker dapat dideteksi dengan benar oleh TensorFlow.

Penelitian ini baru melakukan pengujian untuk mendeteksi orang bermasker atau tidak. Perlu adanya penelitian lanjutan terkait akurasi jarak jangkauan yang dapat dideteksi oleh sistem berdasarkan kondisi di luar ruangan maupun di dalam ruangan. Pada penelitian selanjutnya, perlu dikembangkan pengujian dengan jarak jangkauan perangkat terhadap model yang dideteksi, supaya mendapatkan hasil prediksi yang optimal.

KONFLIK KEPENTINGAN

Selama pelaksanaan penelitian dan penulisan makalah dengan judul “Sistem Deteksi Masker Berbasis Jetson Nano

dengan *Deep Learning Framework TensorFlow*” ini, tim penulis tidak memiliki konflik kepentingan dengan pihak mana pun.

KONTRIBUSI PENULIS

Berikut merupakan pembagian kontribusi masing-masing penulis yang berperan dalam pelaksanaan penelitian ini. Konseptualisasi dan implementasi program perangkat keras, Muhammad Luqman Bukhori; perancangan alat, Erwan Eko Prasetyo; penulisan–penyusunan draf asli, Muhammad Luqman Bukhori dan Erwan Eko Prasetyo; pengumpulan data, Muhammad Luqman Bukhori.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada Lembaga Pengolah Dana Pendidikan Kementerian Keuangan dan Direktorat Jenderal Pendidikan Tinggi, Kementerian Pendidikan, Kebudayaan, Riset dan Teknologi Nomor: PRJ-79/LPDP/2021 dan 6/E1/HK.02.06/2021 atas bantuan dan dukungan dana dalam pelaksanaan Program Riset Keilmuan. Peneliti juga mengucapkan terima kasih kepada Sekolah Tinggi Teknologi Kedirgantaraan (STTKD) Yogyakarta yang telah memberi dukungan pendanaan penelitian sesuai dengan kontrak nomor Skep/63/XII/2021/STTKD. Terima kasih juga disampaikan kepada rekan-rekan taruna/i serta karyawan yang terlibat sebagai model pengujian dan semua pihak yang turut menyiapkan dan mendukung ketersediaan tempat pengujian penelitian.

REFERENSI

- [1] (2021) “Coronavirus disease (COVID-19): How is it transmitted?” [Online], <https://www.who.int/news-room/questions-and-answers/item/coronavirus-disease-covid-19-how-is-it-transmitted>, tanggal akses: 6-Jul-2022.
- [2] (2022) “Virus corona (COVID-19),” [Online], <https://news.google.com/covid19/map?hl=id&mid=%2Fm%2F03ryn&g1=ID&ceid=ID%3Aid>, tanggal akses: 6-Jul-2022.
- [3] (2020) “Anjuran mengenai penggunaan masker dalam konteks COVID-19,” [Online], <https://www.who.int/docs/default-source/searo/indonesia/covid19/anjuran-mengenai-penggunaan-masker-dalam-konteks-covid-19-june-20.pdf>, tanggal akses: 6-Jul-2020.
- [4] H.E. Siahaineinia dan T.L. Bakara, “Persepsi Masyarakat tentang Penggunaan Masker dan Cuci Tangan selama Pandemi Covid-19 di Pasar Sukaramai Medan,” *Wahana Inov.: J. Penelit. dan Pengabd. Masy.*

- UISU, Vol. 9, No. 1, hal. 172–176, Jan.-Jun. 2020.
- [5] S.F. Rizqah, Haeruddin, dan A.R. Amelia, “Hubungan Perilaku Masyarakat dengan Kepatuhan Penggunaan Masker untuk Memutus Rantai Penularan Covid-19 di Kelurahan Bontoa Maros,” *J. Muslim Community Health*, Vol. 2, No. 3, hal. 165–175, Jul.-Sep. 2021, doi: 10.52103/jmch.v2i3.553.
- [6] E. Lubis, “Peran Protokoler dalam Menunjang Keberhasilan Kinerja Kepala Bagian Umum Pemerintahan Kabupaten Deli Serdang,” *Perspektif*, Vol. 7, No. 2, hal. 362–373, Jul. 2014, doi: 10.31289/perspektif.v4i2.165.
- [7] J. Chai, H. Zeng, A. Li, dan E.W.T. Ngai, “Deep Learning in Computer Vision: A Critical Review of Emerging Techniques and Application Scenarios,” *Mach. Learn. Appl.*, Vol. 6, hal. 1–13, Des. 2021, doi: 10.1016/j.mlwa.2021.100134.
- [8] S.C. Hsu, Y.W. Wang, dan C.L. Huang, “Human Object Identification for Human-Robot Interaction by Using Fast R-CNN,” *2018 Second IEEE Int. Conf. Robot. Comput. (IRC)*, 2018, hal. 201–204, doi: 10.1109/IRC.2018.00043.
- [9] D. Kim dkk., “Real-Time Multiple Pedestrian Tracking Based on Object Identification,” *2019 IEEE 9th Int. Conf. Consum. Electron. (ICCE-Berlin)*, 2019, hal. 363–365, doi: 10.1109/ICCE-Berlin47944.2019.8966205
- [10] S.I. Ali, S.S. Ebrahimi, M. Khurram, dan S.I. Qadri, “Real-Time Face Mask Detection in Deep Learning Using Convolution Neural Network,” *2021 10th IEEE Int. Conf. Commun. Syst., Netw. Technol. (CSNT)*, 2021, hal. 639–642, doi: 10.1109/CSNT51715.2021.9509704.
- [11] M.I. Amin, M.A. Hafeez, R. Touseef, dan Q. Awais, “Person Identification with Masked Face and Thumb Images under Pandemic of COVID-19,” *2021 7th Int. Conf. Control, Instrum., Automat. (ICCIA)*, 2021, hal. 1–4, doi: 10.1109/ICCIA52082.2021.9403577.
- [12] M.R. Alwanda, R.P.K. Ramadhan, dan D. Alamsyah, “Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle,” *Algoritma*, Vol. 1, No. 1, hal. 45–56, Okt. 2020, doi: 10.35957/algoritma.v1i1.434.
- [13] J. Pujoseno, “Implementasi Deep Learning Menggunakan Convolutional Neural Network untuk Klasifikasi Alat Tulis,” Tugas Akhir, Universitas Islam Indonesia, Indonesia, Mar. 2018.
- [14] M.N.H. Siregar, “Model Arsitektur Artificial Neural Network pada Pelanggan Listrik Negara (PLN),” *InfoTekJar (J. Nas. Inform., Teknol. Jar.)*, Vol. 3, No. 1, hal. 1–5, Sep. 2018, doi: 10.30743/infotekjar.v3i1.642.
- [15] M.B. Herlambang (2019) “Deep Learning: Recurrent Neural Networks homepage on website Epam,” [Online], <https://www.megabagus.id/deep-learning-recurrent-neural-networks/>, tanggal akses: 6-Jul-2022.
- [16] G. Kaur dkk., “Face Mask Recognition System Using CNN Model,” *Neurosci. Inform.*, Vol. 2, No. 3, hal. 1-9, Sep. 2022, doi: 10.1016/j.neuri.2021.100035.
- [17] O. Kembuan, G.C. Rorimpandey, dan S.M.T. Tengker, “Convolutional Neural Network (CNN) for Image Classification of Indonesia Sign Language Using Tensorflow,” *2020 2nd Int. Conf. Cybern., Intell. Syst. (ICORIS)*, 2020, hal. 1-5, doi: 10.1109/ICORIS50180.2020.9320810.
- [18] R. Tiniges (2021) “Algoritma Deep Learning : Kenalan dengan Bagian-Bagian Deep Learning, Yuk!” [Online], <https://www.dqlab.id/algoritma-deep-learning-machine-learning>, tanggal akses: 6-Jul-2022.
- [19] (2021) “Convolutional Neural Network With Tensorflow and Keras,” [Online], <https://medium.com/geekculture/introduction-to-convolutional-neural-network-with-tensorflow-and-keras-cb52cdc66eaf>, tanggal akses: 6-Jul-2022.
- [20] R.M. Pradistya (2021) “Mengenal Tensorflow, Library untuk Keperluan Machine Learning Python” [Online], <https://www.dqlab.id/mengenal-tensorflow-library-untuk-keperluan-machine-learning-python>, tanggal akses: 8-Jul-2022.
- [21] Friendly, Z. Sembiring, dan H.R. Safitri, “Deteksi Wajah Bermasker Berbasis Tensorflow-Keras untuk Pengendalian Gerbang Akses Masuk Menggunakan Raspberry Pi4,” *JIKSTRA*, Vol. 2, No. 2, hal. 45–55, Okt. 2020.
- [22] (2021) “TensorFlow_Lite_Face_Mask_Jetson-Nano,” [Online], https://github.com/Qengineering/TensorFlow_Lite_Face_Mask_Jetson-Nano, tanggal akses: 6-Jul-2022.
- [23] F.A.M. Ali, dan M.S.H. Al-Tamimi, “Face Mask Detection Methods and Techniques: A Review,” *Int. J. Nonlinear Anal., Appl.*, Vol. 13, No. 1, hal. 3811-3823, Jan. 2022, doi: 10.22075/ijnaa.2022.6166.
- [24] V.K. Pandey, V.K. Gupta, dan S. Kumar, “Face Mask Detection Using Convolutional Neural Network,” *2021 3rd Int. Conf. Adv. Comput., Commun. Control, Netw. (ICAC3N)*, 2021, hal. 951–954, doi: 10.1109/ICAC3N53548.2021.9725689.
- [25] S. Singh dkk., “Face Mask Detection Using YOLOv3 and Faster R-CNN Models: COVID-19 Environment,” *Multimed. Tools, Appl.*, Vol. 80, No. 13, hal. 19753–19768, Mar. 2021, doi: 10.1007/s11042-021-10711-8.
- [26] V. Saminathan dkk., “Face Mask Detection Using Raspberry Pi,” *Ann. Romanian Soc. Cell Biol.*, Vol. 25, No. 4, hal. 9982–9988, Apr. 2021.
- [27] E.N. Arrofiqoh dan Harintaka, “Implementasi Metode Convolutional Neural Network untuk Klasifikasi Tanaman pada Citra Resolusi Tinggi,” *Geomatika*, Vol. 24, No. 2, hal. 61-68, Nov. 2018, doi: 10.24895/jig.2018.24-2.810.
- [28] Trivusi (2022) “Pengertian dan Cara Kerja Algoritma Convolutional Neural Network (CNN),” [Online], <https://www.trivusi.web.id/2022/04/algoritma-cnn.html>, tanggal akses: 11-Agu-2022.
- [29] J. Feriawan dan D. Swanjaya, “Perbandingan Arsitektur Visual Geometry Group dan MobileNet pada Pengenalan Jenis Kayu,” *Sem. Nas. Inov. Teknol.*, 2020, hal. 185–190, doi: /10.29407/inotek.v4i3.84.
- [30] P. Nyoman dan P.K. Negara, “Deteksi Masker Pencegahan Covid19 Menggunakan Convolutional Neural Network Berbasis Android,” *J. RESTI (Rekayasa Sist., Teknol. Inf.)*, Vol. 5, No. 3, hal. 576–583, Jun. 2021, doi: 10.29207/resti.v5i3.3103.
- [31] A. Rosebrock (2020) “COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning,” [Online], <https://pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/>, tanggal akses: 6-Jul-2022.
- [32] F.R. Lumbanraja dkk., “An Evaluation of Deep Neural Network Performance on Limited Protein Phosphorylation Site Prediction Data,” *Procedia Comput. Sci.*, Vol. 157, hal. 25–30, 2019, doi: 10.1016/j.procs.2019.08.137.
- [33] (2021) “Jetson Nano Developer Kit,” [Online], <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>, tanggal akses: 7-Jul-2022.