

RESEARCH ARTICLE

Analisis Deteksi Masker Wajah menggunakan YOLOv8 dengan Dataset Facemask

Arya Beta Putra Pratama, Tjokorda Agung Budi Wirayuda* and Febriyanti Sthevanie

Fakultas Informatika, Universitas Telkom, Bandung, 40257, Jawa Barat, Indonesia

*Corresponding author: cokagung@telkomuniversity.ac.id

Abstrak

Penelitian ini menyajikan analisis mendalam tentang deteksi dan klasifikasi masker wajah menggunakan YOLOv8 dan akan diuji coba pada *dataset Facemask* yang didalamnya terdiri dari gambar-gambar yang dibagi dalam tahap pelatihan, pengujian, dan validasi dan melalui dua pendekatan, yaitu augmentasi dan non augmentasi. Penelitian ini menganalisis penilaian kinerja YOLOv8 dan menyoroti kemampuannya mengenali individu yang memakai masker wajah dan yang tidak memakai masker wajah. Tujuan utama dari penelitian ini adalah untuk menganalisis performa YOLOv8 dalam mendeteksi dan mengklasifikasikan penggunaan masker wajah. Hasil evaluasi berdasarkan tiga metrik utama yaitu *Mean Average Precision* (mAP), *Precision*, dan *Recall*. Hasil pada pendekatan non augmentasi model menunjukkan *Mean Average Precision* (mAP) 93,1%, *Precision* 79,7% , dan *Recall* 95,9%. Hasil pada pendekatan augmentasi menunjukkan *Mean Average Precision* (mAP) 91,9%, *Precision* 76,6% , dan *Recall* 94,7%.

Key words: Deteksi masker wajah, klasifikasi, YOLOv8, *dataset Facemask*, analisis.

Pendahuluan

Penggunaan masker saat pandemi COVID-19 menjadi salah satu protokol kesehatan yang wajib dilakukan oleh seluruh masyarakat dunia, penggunaan masker sangat penting guna mencegah penyebaran virus corona, sehingga *World Health Organization* (WHO) memberi himbauan pada pemerintah setiap negara untuk melakukan pencegahan penyebaran virus corona, salah satu dari himbauan yang diberikan adalah penggunaan masker yang baik dan benar. Di Indonesia sendiri keputusan penggunaan masker diterbitkan oleh Kementerian Kesehatan Republik Indonesia (Kemenkes RI). Berdasarkan keputusan Kementerian Kesehatan Republik Indonesia (Kemenkes RI) Nomor HK.01.07/MENKES/382/2020 masyarakat diimbau menggunakan alat pelindung diri berupa masker dengan baik dan benar jika hendak beraktivitas di luar rumah atau berinteraksi dengan orang lain yang tidak diketahui status kesehatannya (yang mungkin dapat menularkan COVID-19) pada saat di ruang publik [1]. Penggunaan masker sendiri dapat digunakan baik untuk melindungi diri sendiri (tidak terinfeksi) maupun untuk mencegah penyebarluasan virus (dipakai oleh orang yang sudah terinfeksi) [2]. Penggunaan masker efektif dalam membatasi penularan oleh pembawa virus yang tidak bergejala atau tidak terdeteksi secara medis dan sudah terbukti manfaatnya dalam hal mencegah penyakit menular melalui sistem pernafasan [3]. Berdasarkan hasil observasi yang dilakukan di lapangan masih banyak masyarakat yang tidak mematuhi protokol yang sudah ditetapkan, masih banyak masyarakat yang tidak patuh atau hanya menggunakan masker sebagai syarat untuk masuk ke ruang publik, penggunaan masker

harus digunakan dengan benar dan baik sesuai aturan dan sebaiknya menutupi area hidung dan mulut (sebagai sistem pernafasan).

Beberapa ruang publik maupun kesehatan sudah menegakan protokol kesehatan yang diantaranya memeriksa masker oleh petugas-petugas yang berwenang, namun karena cara ini masih melibatkan manusia atau manual, menyebabkan terjadinya celah karena bisa terjadi kelalaian saat memeriksa, berbeda jika petugas dan mesin bisa bekerja sama saat melakukan pemeriksaan yang membuat lebih efisien. Maka sebab itu dibuatlah sebuah sistem menggunakan cabang ilmu pengetahuan *Computer Vision* yang khususnya berada di bidang *object detection* untuk mendeteksi penggunaan masker. Tersedia beberapa metode untuk mengidentifikasi dan mengklasifikasikan objek pada sebuah citra, dan salah satu yang digunakan dalam penelitian ini adalah YOLOv8. Penelitian ini dilakukan untuk menganalisis performa YOLOv8 dalam mendeteksi dan mengklasifikasi Penggunaan masker wajah, melalui dua pengujian yaitu menggunakan augmentasi dan non augmentasi, dengan harapan hasil dari penelitian ini bisa dijadikan sebagai referensi untuk melakukan pendeteksian dan klasifikasi menggunakan YOLOv8. YOLOv8 sendiri adalah tambahan ke-8 untuk keluarga model detector YOLO. YOLOv8 dipilih dalam upaya menemukan *trade-off* terbaik antara inferensi kecepatan dan *Mean Average Precision* (mAP) dan YOLOv8 dianggap sebagai yang terancang [4]. Sehingga menjadi alasan utama mengapa YOLOv8 digunakan dalam penelitian deteksi dan klasifikasi penggunaan masker wajah.

Rumusan Masalah

1. Bagaimana hasil deteksi masker wajah menggunakan YOLOv8 dengan dan tanpa teknik augmentasi?
2. Bagaimana perbandingan hasil deteksi dan akurasi antara model dengan dan tanpa augmentasi?

Tujuan

1. Menilai Dampak Augmentasi Data
2. Membandingkan Hasil Model dengan dan Tanpa Augmentasi

Tinjauan Pustaka

Penelitian ini dimulai dengan menggaris bawahi bahwa pendeteksian masker yang dilakukan secara manual oleh manusia akan menyimpulkan jika Penggunaan *computer vision* dapat meningkatkan akurasi dan efisiensi dalam pendeteksian masker wajah. Sehingga penelitian-penelitian sebelumnya yang berada dalam bidang pendeteksi objek, khususnya pada bidang deteksi dan klasifikasi masker wajah. Selanjutnya, akan diberi penjelasan tentang model YOLOv8 yang digunakan dalam penelitian ini, semua dilakukan untuk memberikan dan memperkuat dasar penting untuk pemahaman tentang penelitian ini.

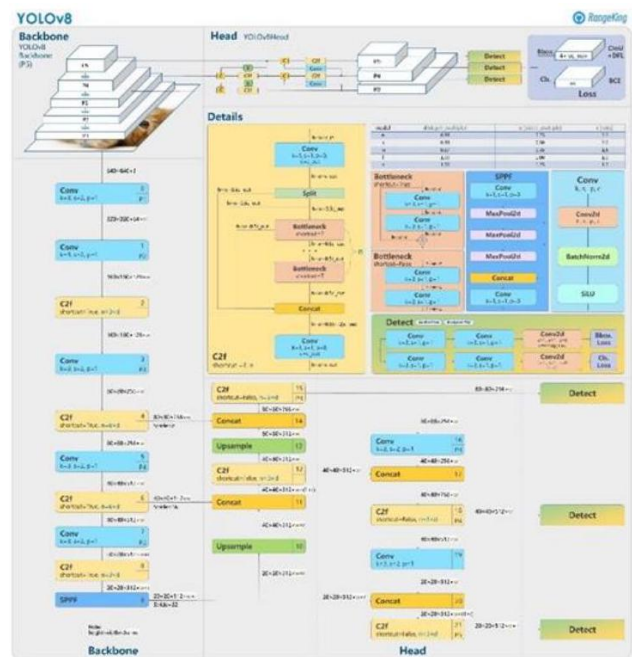
Deteksi Masker

Penelitian terkait deteksi masker wajah mulai ramai semenjak terjadinya pandemi COVID-19 pada tahun 2019. Pandemi COVID-19 menyebabkan krisis kesehatan secara global, dan salah satu metode efektif yang dilakukan untuk melindungi dari virus adalah memakai masker wajah di tempat umum menurut *World Health Organization* (WHO), sebuah model hibrida menggunakan *deep machine learning* untuk mendeteksi masker wajah [5]. Kebijakan dalam menggunakan masker terutama di ruang publik, menyebabkan beberapa protokol keamanan di ruang publik dipantau oleh petugas keamanan, namun tindakan ini tidak hanya membuat para penjaga terpapar udara yang memungkinkan penyebarannya virus, tetapi juga menyebabkan kepadatan pada saat di pintu karena tidak efisien. Oleh karena itu dibutuhkan metode yang cepat dan efektif dalam mengatasi situasi tersebut [6]. Penelitian sebelumnya yang dilakukan oleh bagus hardiansyah pendeteksian masker menggunakan algoritma *Deep Learning* YOLOv4 sebagai metode deteksi wajah, pada *pre-trained* menggunakan VGGFace model sebagai model ekstraksi fitur dan *artificial neural network* sebagai metode klasifikasi. Ketika menggunakan metode teknik *training* dilakukan maka hasil *testing* didapat 99.55%. menunjukkan bahwa hasil performa cukup baik ketika menyesuaikan dengan teknik *training* dalam sistem deteksi penggunaan masker diwajah.

Adapun dalam penelitian [7] Sebuah solusi diusulkan dalam penelitian ini: menggunakan AI untuk deteksi masker di perusahaan guna memastikan bahwa setiap pekerja mengenakan masker dengan benar selama bekerja. Mengaplikasikan algoritma YOLOv8 untuk mendeteksi penggunaan masker yang tepat. Tujuan utama dari penelitian ini adalah untuk memfasilitasi deteksi masker dan memastikan bahwa masker digunakan dengan benar, sehingga memastikan keselamatan dan kesehatan semua orang di lingkungan dengan pendekatan AI menggunakan metode YOLO. kelas dibagi menjadi tiga yaitu *Badmask*, *Mask*, dan *Nomask*, penelitian ini memiliki hasil akurasi yang tinggi dengan kelas *Badmask* sebesar 94%, *Mask* sebesar 97% dan *Nomask* sebesar 95%. Nilai *F1-Confidence*, *Precision*, dan *Recall* semua kelasnya juga tinggi, yaitu masing-masing sebesar 0.94, 0.96, dan 0.978. Waktu komputasi rata-rata yang diperlukan oleh YOLOv8 hanya 17 ms. Selama beberapa tahun terakhir, metode penggunaan *deep machine learning* telah mengungguli teknik *machine learning* di beberapa bidang, dengan *computer vision* sebagai salah satu kasus yang

Table 1. YOLOv8

No	Keunggulan
1	Performa YOLOv8 lebih baik dari versi sebelumnya dengan deteksi lebih cepat dan akurat
2	Model YOLOv8 memiliki ukuran lebih kecil dari versi sebelumnya, sehingga lebih mudah digunakan
3	Ekstraksi fitur YOLOv8 lebih baik, sehingga menghasilkan deteksi objek yang lebih akurat
4	YOLOv8 memiliki kemampuan multiskala, sehingga dapat menangani deteksi objek dengan ukuran berbeda pada gambar yang sama
5	YOLOv8 dapat mendeteksi banyak objek pada gambar yang sama
6	Kemampuan mendeteksi objek pada gambar berukuran besar dapat dilakukan oleh YOLOv8 dengan akurasi yang baik, dan mempermudah proses deteksi objek.



Gambar 1. Arsitektur YOLOv8 dari repositori GitHub RangeKing, yang menampilkan desain inovatifnya untuk deteksi objek dan segmentasi (<https://github.com/RangeKing>)

sering digunakan, beberapa kemampuan yang bisa dilakukan *computer vision*, seperti deteksi objek, pengenalan wajah, pengenalan aksi dan aktivitas, dan beberapa kemampuan lainnya [8]. Dengan adanya *computer vision* dapat membantu pendeteksian masker wajah di ruang publik dengan cepat dan aman.

YOLOv8

YOLOv8 [9] sendiri dirilis pada tahun 2023 oleh Ultralytics, sebuah perusahaan yang mengembangkan YOLOv5. YOLOv8 dapat dijalankan dari *Command Line Interface* (CLI), dan bisa diunduh sebagai paket PIP. YOLOv8 juga dilengkapi dengan beberapa integrasi untuk *labeling*, *training*, dan *deploying*. Selain itu YOLOv8 memiliki 5 versi skala: YOLOv8n (*nano*), YOLOv8s (*small*), YOLOv8m (*medium*), YOLOv8l (*large*), dan YOLOv8x (*extra large*) [10] Ultralytics telah melakukan banyak peningkatan pada YOLOv8 dan menjadikannya lebih baik dan

ramah bagi para pengguna dibandingkan dengan YOLOv5, YOLOv8 adalah model canggih yang meningkatkan keberhasilan YOLOv5 dengan modifikasi yang meningkatkan kekuatan dan keakuratan untuk berbagai tugas *computer vision* [11]. Berikut beberapa keunggulan metode YOLOv8 berdasarkan penelitian [12].

Pada gambar 1 terlihat YOLOv8 memiliki jaringan *backbone* baru yang merupakan modifikasi dari arsitektur CSPDarknet53 yang terdiri dari 53 lapisan-lapisan dan menggunakan teknik *Cross Stage Partial Network* (CSPNet) untuk meningkatkan transmisi informasi pada berbagai tingkat jaringan. Head pada YOLOv8 mengambil peta fitur yang telah dihasilkan oleh *backbone* dan diproses lebih lanjut untuk memberikan keluaran akhir model dalam bentuk *bouding box* dan kelas objek. Dan terakhir modul deteksi dengan menggunakan satu set lapisan konolusi dan linier untuk menempatkan fitur dimensi yang tinggi ke *bouding box* dan objek kelas [13].

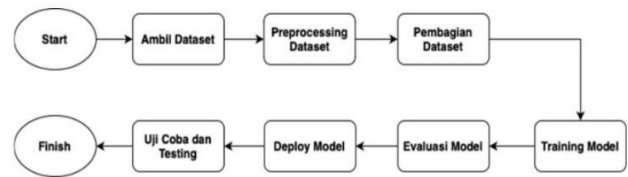
Pendeteksian dan klasifikasi

Pada penelitian [14] Pada sistem yang dirancang untuk mendeteksi dan mengklasifikasikan masker yang digunakan oleh orang-orang pada suatu kerumunan dengan menggunakan implementasi model *You Only Look Once* (YOLO), rancangan ini mengambil input berupa rekaman video dari kamera dan menghasilkan output berupa kotak pembatas yang berlabel tiga kelas yaitu masker medis, masker non medis, dan tidak menggunakan masker. Hasil dari penelitian ini yaitu aplikasi Pendeteksian Penggunaan Masker dan Klasifikasi Secara *Real Time* Melalui Video Webcam Dengan Metode YOLO ini mampu melakukan pendeteksian penggunaan masker dan klasifikasi dikerumunan cukup baik dengan tingkat akurasi sebesar 93.33%.

Penelitian Terkait

Penelitian terkait *real-time deep learning* yang efisien untuk mengotomatiskan proses pendeteksian masker wajah, dimana setiap masker wajah diidentifikasi secara *real-time* dengan bantuan kotak pembatas. Uji coba ekstensif dilakukan dengan model yang populer, yaitu, Faster RCNN dan YOLO v3. FRCNN memiliki presisi yang lebih baik, tetapi untuk menerapkannya di kamera pengintai dunia nyata, akan lebih baik menggunakan model dengan algoritma YOLO karena melakukan deteksi satu bidikan dan memiliki kecepatan bingkai yang jauh lebih tinggi daripada FasterRCNN atau algoritma pendeteksian objek mutakhir lainnya[15]. Pada evaluasi MS COCO *dataset test-dev* 2017, YOLOv8x mencapai AP 53,9% dengan ukuran gambar 640 pixels (dibandingkan dengan YOLOv5 hanya 50,7% dengan ukuran input yang sama) dengan kecepatan 280 FPS pada NVIDIA A100 dan pada TensorRT[16]. Dilakukan perbandingan antara YOLOv8 dengan Mask R-CNN terhadap segmentasi instance dalam konteks pertanian khususnya pada pohon apel, untuk segmentasi kelas tunggal pada buah hijau yang belum matang, kurva *presicion-confidence* mengungkapkan bahwa YOLOv8 mengungguli Mask R-CNN dalam hal *Mean Average Precision* (mAP), mencapai angka sebesar 0,939 pada ambang batas 0,5 *Intersection over Union* (IoU) untuk buah-buahan hijau dan semua kategori, sedangkan Mask R-CNN hanya mendapat mAP sebesar 0,902[17].

Adapun penelitian Pemanfaatan Google Colab Untuk Aplikasi Pendeteksian Masker Wajah Menggunakan Algoritma *Deep Learning* YOLOv7. Hasil *training* pada penelitian menggunakan YOLOv7 menunjukkan nilai *Precision* konsisten berada pada angka 0,4–0,8. Sedangkan nilai *Recall* maksimum pada angka 0,6. Pada penelitian lain [18] dengan kerangka kerja *Roboflow* di Google Colab menghasilkan hasil yang luar biasa untuk *dataset Smart Factory* kami, mencapai rata-rata *Average Precision* (mAP) sebesar 91,2%, presisi sebesar 82,1%, dan *recall* sebesar 89,7%. Menunjukkan konvergensi yang stabil dalam hal presisi, *recall*, dan mAP. Setelah melalui banyak pertimbangan dan literatur dibuat keputusan jika penelitian ini akan menggunakan metode



Gambar 2. Flowchart



Gambar 3. Dataset FaceMask

YOLOv8 untuk mendeteksi dan klasifikasi masker wajah menggunakan dataset yang sudah ditentukan.

Metodologi Penelitian

Desain Penelitian

Flowchart pada Gambar 2 menggambarkan langkah-langkah rinci dalam pengembangan sistem deteksi masker wajah menggunakan YOLOv8. Pada tahap pelatihan, model YOLOv8 dilatih menggunakan *training set* di Google Colab. Proses ini menghasilkan model yang dapat mendeteksi masker wajah dengan akurasi yang diukur melalui metrik seperti *Mean Average Precision* (mAP), *Precision*, dan *Recall*. Setelah pelatihan selesai, model di-deploy ke *Roboflow*, yang memungkinkan visualisasi hasil dalam bentuk grafik dan evaluasi kinerja. Model yang sudah di-deploy kemudian diuji coba melalui dua metode: scan QR code untuk penggunaan di perangkat seluler, dan integrasi dengan webcam (baik eksternal maupun internal) untuk deteksi masker secara *real-time*. Proses ini diakhiri dengan evaluasi hasil pengujian untuk memastikan model berfungsi sesuai dengan yang diharapkan. Alur yang tersusun rapi ini memastikan setiap langkah dalam pengembangan sistem terstruktur dengan baik dan menghasilkan model yang optimal untuk deteksi masker wajah.

Facemask Dataset

Gambar 3 menampilkan beberapa contoh gambar-gambar yang ada pada *Dataset FaceMask*. Pada penelitian ini, dataset yang digunakan untuk algoritma YOLOv8, dataset yang digunakan diambil dari *kaggle*. Proses dimulai dengan pengambilan dataset "*Facemask Detection*" dari *Kaggle*, yang terdiri dari 6.240 gambar yang dikategorikan ke dalam dua kelas: "*WithMask*" dan "*WithoutMask*". Dataset ini kemudian melalui tahap *preprocessing*, dimana setiap gambar di-resize menjadi ukuran 640x640 piksel untuk memastikan konsistensi dalam input model. Selanjutnya, dataset dibagi menjadi tiga set: *training set* (80%), *validation set* (13%), dan *testing set* (7%).

Table 2. Split Dataset

TRAIN SET	80%	4984 Gambar
VALID SET	13%	799 Gambar
TEST SET	7%	457 Gambar

Preprocessing Data

Tahap *preprocessing* dilakukan dengan menerapkan *resizing* pada gambar untuk memastikan ukuran konsisten yaitu 640×640 piksel. Proses augmentasi dilakukan pada pengujian yang berbeda untuk mendapatkan hasil perbandingan yang berbeda pada penelitian ini, pada pengujian yang menggunakan augmentasi *preprocessing* dilakukan dengan ukuran konsisten yang sama yaitu 640×640 dengan menambahkan *grayscale*, untuk augmentasi ditambahkan *Flip*, *Exposure*, dan *Noise*.

Augmentasi

Pada tahap ini dilakukan augmentasi pada data. Augmentasi yang digunakan pada model yaitu, *horizontal flip*, perubahan *exposure*, dan penambahan *noise*. *Horizontal flip* memungkinkan model untuk belajar dari variasi orientasi objek, meningkatkan *robustness* dalam mendeteksi masker wajah dari berbagai sudut. Perubahan *exposure*, dengan rentang antara -10% hingga $+10\%$, membantu model menghadapi variasi pencahayaan dalam gambar, sedangkan penambahan *noise* hingga $0,1\%$ dari piksel menambah ketahanan model terhadap gangguan visual. Kombinasi teknik augmentasi ini berkontribusi pada peningkatan akurasi deteksi objek, sehingga model dapat bekerja dengan lebih baik dalam berbagai kondisi nyata.

Pembagian Dataset

Pada tahap ini dataset dibagi menjadi tiga set, yaitu:

1. *Training Set*: Sebanyak 4.984 gambar digunakan untuk melatih model YOLOv8.
2. *Validation Set*: Sebanyak 799 gambar digunakan untuk mengevaluasi kinerja model selama pelatihan.
3. *Testing Set*: Sebanyak 457 gambar digunakan untuk menguji kinerja model setelah pelatihan selesai.

Proses Training

Proses pelatihan dilakukan menggunakan GoogleColab dengan menggunakan YOLOv8 yang diintegrasikan dengan *Roboflow*. Pada tahap ini, model dilatih untuk mendeteksi objek dengan menggunakan dataset yang telah disiapkan. Setelah proses pelatihan selesai, model di-deploy ke *Roboflow* untuk menghasilkan visualisasi hasil berupa *Mean Average Precision* (mAP), *Precision*, dan *Recall*.

Implementasi dan Uji Coba

Hasil dari model YOLOv8 yang telah dilatih kemudian diuji coba dengan berbagai cara:

1. Scan QR Code: Model dapat diakses melalui ponsel dengan memindai QR code yang dihasilkan.
2. Webcam: Model diintegrasikan dengan webcam (baik eksternal maupun internal laptop) untuk mendeteksi masker wajah secara langsung.

Evaluasi

Evaluasi dilakukan dengan mengukur performa model menggunakan beberapa metrik berikut:

Table 3. Hasil Training YOLOv8

Non augmentasi	mAP = 93,1%	Precision = 79,7%	Recall = 95,9%
Augmentasi	mAP = 91,9%	Precision = 76,6%	Recall = 94,7%

1. *Mean Average Precision* (mAP): Pada tahap *Mean Average Precision* (mAP) mengukur seberapa baik model dalam mendeteksi objek yang benar. Dengan menghitung *Average Precision* (AP) pada semua kelas dalam dataset. mAP memberikan gambaran keseluruhan terhadap performa model, dengan menggabungkan aspek *Precision* dan *Recall* diberbagai *threshold*, sehingga mAP menjadi metrik yang komperhensif untuk menilai kinerja terhadap model deteksi objek. Berikut rumus dari mAP.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (1)$$

2. *Precision*

Pada tahap ini *Precision* (P) mengukur sejauh mana metrik dapat mendeteksi kelas positif yang dilakukan oleh model dengan hasil yang tepat. Berikut rumus dari *precision*:

$$Precision (P) = \frac{True Positive (TP)}{True Positive (TP) + False Positive (FP)} \quad (2)$$

3. *Recall*

Recall adalah metrik yang, mengukur sebuah model untuk menemukan Kembali atau mengidentifikasi kembali semua kelas positif yang berada dalam dataset. Berikut rumus dari *Recall*:

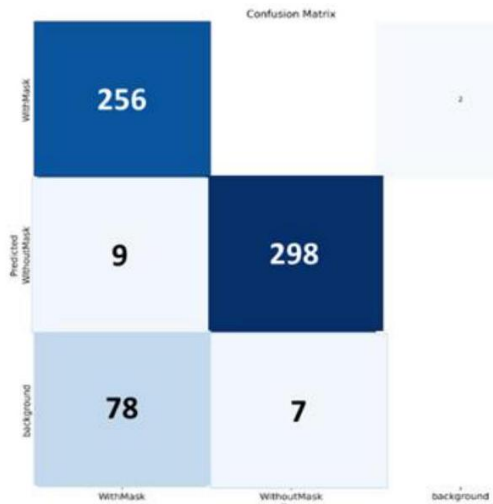
$$Recall (P) = \frac{True Positive (TP)}{True Positive (TP) + False Negative (FN)} \quad (3)$$

Hasil dan Pembahasan

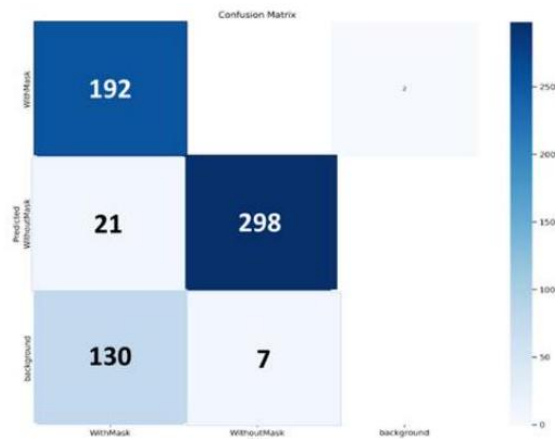
Pada tahap hasil dan pembahasan, berisikan hasil dan membahas setiap hasil yang telah didapatkan oleh model YOLOv8 dalam identifikasi dataset *FaceMask* dengan menggunakan *Roboflow*.

Hasil Pelatihan Model YOLOv8

Pada pengujian model YOLOv8, dilakukan pelatihan dengan masing-masing proyek sebanyak 25 epoch, baik dengan augmentasi maupun non augmentasi, menunjukkan hasil yang memuaskan. Pada model non augmentasi, *Mean Average Precision* (mAP) mencapai 93,1%, yang patut dipuji karena menunjukkan akurasi tinggi dalam mengidentifikasi dan mengingat objek. *Precision*, yang mengukur seberapa sering prediksi model benar, mencapai 79,7%, menandakan kemampuan model untuk secara tepat dan relevan mengidentifikasi objek. Sementara itu, *recall* yang mengukur persentase label relevan yang berhasil diidentifikasi, mencapai 95,9%, menunjukkan efektivitas model dalam menangkap kasus positif aktual. Pada model dengan augmentasi, mAP sedikit menurun menjadi 91,9%, yang tetap merupakan angka tinggi dan mencerminkan kemampuan model yang baik dalam mengenali objek. *Precision* juga sedikit turun menjadi 76,6%, namun *recall* masih berada diangka tinggi, yaitu 94,7%, yang menunjukkan efektivitas serupa dalam mendeteksi kasus positif. Meskipun terdapat penambahan *preprocessing* seperti *Grayscale* dan augmentasi, perbedaan kinerja antara kedua model tidak terlalu signifikan. Hal ini menunjukkan bahwa YOLOv8 tetap mampu mendeteksi objek dengan baik, bahkan dengan perubahan pada dataset yang digunakan. Secara keseluruhan,



Gambar 4. Confusion Matrix non Augmentasi



Gambar 5. Confusion Matrix Augmentasi

kedua pendekatan ini berhasil mencapai hasil yang selaras dengan tujuan awal, yaitu mengidentifikasi individu yang memakai masker dan yang tidak.

Hasil Confusion Matrix

Confusion matrix adalah alat yang digunakan untuk mengevaluasi hasil prediksi model dengan membandingkan prediksi model terhadap label sebenarnya dari data. Dengan adanya *confusion matrix* dapat mengukur ketepatan objek dalam mengukur True Positives (TP) jumlah prediksi yang benar-benar positif. *True Negatives* (TN) jumlah prediksi yang benar-benar negatif *False Positive* (FP) Jumlah prediksi yang positif tetapi sebenarnya negatif, dan *False Negative* (FN) Jumlah prediksi yang negatif tetapi sebenarnya positif. *Confusion matrix* memberikan dasar untuk menghitung *precision* dan *recall* yang merupakan bagian integral dari perhitungan AP dan, akhirnya, mAP. Keduanya membantu mengevaluasi seberapa baik model dalam mengidentifikasi objek dengan benar dan seberapa baik model tersebut menangkap semua objek relevan dalam dataset.

Pada Gambar 4, ditampilkan *confusion matrix* yang dihasilkan oleh model YOLOv8. *Confusion matrix* ini mengukur ketepatan model dalam

mendeteksi objek, serta mengidentifikasi kesalahan berupa *False Positive* (salah mendeteksi objek) dan *False Negative* (gagal mendeteksi objek). Penjelasan lebih mendetail sebagai berikut:

1. *True Labels* (Baris): Pada sumbu vertikal (yaxis), terdapat label asli atau sebenarnya dari data uji, yang menunjukkan kelas asli dari setiap sampel.
2. *Predicted Labels* (Kolom): Pada sumbu horizontal (x-axis), terdapat prediksi yang dibuat oleh model, yang menunjukkan kelas yang diprediksi oleh model.
3. Isi Matriks: Angka-angka didalam matriks menunjukkan jumlah sampel yang termasuk dalam kategori tertentu. Sebagai contoh, angka 256 di pojok kiri atas menunjukkan bahwa model memprediksi dengan benar sebanyak 256 sampel yang sebenarnya berlabel "WithMask" sebagai "WithMask".

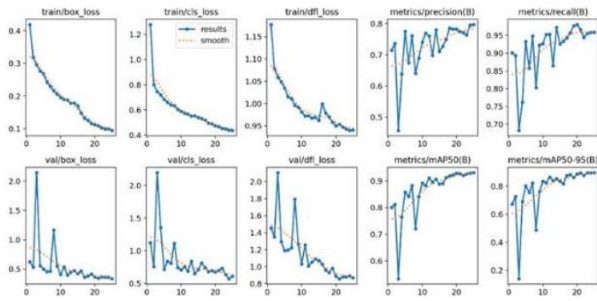
Detail:

1. *WithMask* (Baris pertama):
 - a. 256 sampel diklasifikasikan dengan benar sebagai "WithMask".
 - b. 9 sampel yang sebenarnya "WithMask" diklasifikasikan salah sebagai "WithoutMask".
 - c. 78 sampel yang sebenarnya "WithMask" diklasifikasikan salah sebagai "Background".
2. *WithoutMask* (Baris kedua):
 - a. 298 sampel diklasifikasikan dengan benar sebagai "WithoutMask".
 - b. 2 sampel yang sebenarnya "WithoutMask" diklasifikasikan salah sebagai "WithMask".
 - c. 7 sampel yang sebenarnya "WithoutMask" diklasifikasikan salah sebagai "Background".
3. *Background* (Baris ketiga):
 - a. Tidak ada sampel yang diklasifikasikan sebagai "Background" untuk kelas "WithMask" atau "WithoutMask".

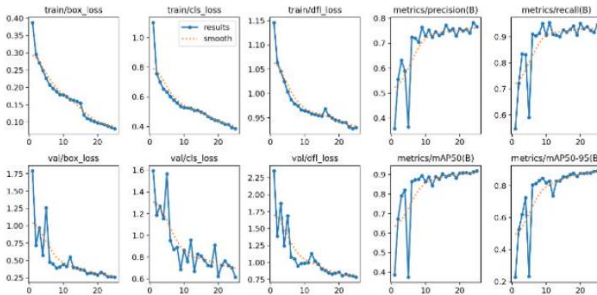
Pada Gambar 5 menjelaskan *confusion matrix* yang dihasilkan oleh model YOLOv8 dengan ditambahkan *preprocessing Grayscale* dan Augmentasi. Disini akan dijelaskan lebih mendetail sebagai berikut: 1 Isi Matriks:

1. *WithMask* (Baris pertama):
 - a. 192 sampel diklasifikasikan dengan benar sebagai *WithMask*.
 - b. 21 sampel yang sebenarnya *WithMask* diklasifikasikan salah sebagai *WithoutMask*.
 - c. 130 sampel yang sebenarnya *WithMask* diklasifikasikan salah sebagai Background.
2. *WithoutMask* (Baris kedua):
 - a. 298 sampel diklasifikasikan dengan benar sebagai *WithoutMask*.
 - b. sampel yang sebenarnya *WithoutMask* diklasifikasikan salah sebagai Background.
3. *Background* (Baris ketiga):
 - a. sampel yang diklasifikasikan sebagai Background untuk kelas *WithoutMask*.
4. Kesimpulan:

Secara umum, model cukup baik dalam mengklasifikasikan *WithMask* dan *WithoutMask*. Pengaruh Augmentasi dan penambahan *preprocessing Grayscale* memberikan dampak positif pada performa model dengan meningkatkan akurasi klasifikasi *WithMask* dan *WithoutMask*. Pengaruh Augmentasi dan penambahan *preprocessing Grayscale* memberikan dampak positif pada performa model dengan meningkatkan akurasi klasifikasi *WithMask* dan *WithoutMask*. Kemunculan Kelas "Background" Pada setiap hasil pengujian meskipun penelitian hanya memiliki dua kelas yaitu "WithMask" dan "WithoutMask", kelas "Background" muncul dalam *confusion matrix*. Ini dapat disebabkan oleh beberapa faktor: Labeling dan Dataset: Ada



Gambar 6. Grafik Metrics non Augmentasi



Gambar 7. Grafik Metrics Augmentasi

kemungkinan kesalahan dalam proses labeling data, di mana beberapa sampel tidak memiliki label yang sesuai atau tidak termasuk dalam kategori "WithMask" atau "WithoutMask". Hal ini dapat menyebabkan model mengategorikan sampel sebagai "Background". Pengelompokan Kelas oleh Model: Model deteksi objek seperti YOLOv8 mungkin memiliki kategori tambahan untuk menangani area yang tidak sesuai dengan kelas target yang ada. Kelas "Background" ini digunakan untuk mendeteksi dan mengklasifikasikan area yang tidak memiliki objek target yang relevan. Masalah dengan *Preprocessing* atau Augmentasi: *Preprocessing* atau augmentasi data yang mengubah visual objek atau latar belakang dapat mempengaruhi klasifikasi model. Misalnya, jika *background* menjadi lebih menonjol, model mungkin salah mengklasifikasikannya sebagai "Background". Penerapan Model dan Pengaturan Parameter: Pengaturan *threshold* deteksi dalam model bisa mempengaruhi klasifikasi. Jika *threshold* terlalu rendah, model mungkin lebih cenderung mengklasifikasikan area yang tidak jelas sebagai "Background". Penyimpangan Data dan Evaluasi: Ketidaksesuaian antara data pelatihan dan data uji atau kategori tambahan dalam data uji dapat menyebabkan model mengklasifikasikan data dengan cara yang tidak diinginkan.

Hasil Metrics

Berdasarkan pada gambar 6 hasil pengujian non augmentasi, analisis statistik pada grafik train memvalidasi lebih lanjut terhadap proses optimasi model. Penurunan pada *metrics loss*, termasuk *train/box loss* dan *train/cls loss*, penurunan loss merupakan hal positif karena hal itu menunjukkan jika akurasi meningkat, dari angka 0,4 turun menjadi 0 yang menunjukkan keberhasilan mengurangi error yang terkait dengan *bounding box* dan kelas prediksi. Berdasarkan gambar 7 pada hasil pengujian menggunakan augmentasi, analisis statistik pada grafik train memvalidasi lebih lanjut terhadap proses optimasi model. Penurunan pada *metrics loss*, termasuk *train/box loss* dan *train/cls loss*, penurunan loss merupakan hal positif karena hal itu menunjukkan jika akurasi meningkat, dari angka 0,40 turun menjadi 0 yang menunjukkan



Gambar 8. Hasil Bounding Box



Gambar 9. Hasil Objek dengan Augmentasi

keberhasilan mengurangi error yang terkait dengan *bounding box* dan kelas prediksi. *Precision*, mAP, dan *recall* menunjukkan hasil peningkatan yang signifikan. Secara keseluruhan, hasil ini menunjukkan bahwa proses optimasi yang dilakukan, termasuk penambahan augmentasi, telah berhasil meningkatkan kinerja model YOLOv8 dalam mendeteksi dan mengklasifikasikan objek.

Selain itu, metrik lain seperti *precision*, *mean Average Precision* (mAP), dan *recall* menunjukkan peningkatan yang signifikan. *Precision* yang meningkat menunjukkan bahwa model lebih berhasil dalam mengidentifikasi objek secara akurat dan mengurangi jumlah prediksi positif yang salah. Peningkatan mAP menunjukkan bahwa model secara keseluruhan memberikan hasil deteksi yang lebih baik dan lebih konsisten untuk berbagai kelas objek. Sementara itu, peningkatan *recall* mengindikasikan bahwa model lebih efektif dalam menangkap semua kasus positif yang sebenarnya, mengurangi jumlah *False Negatives*. Secara keseluruhan, hasil ini mendukung efektivitas model YOLOv8 baik dalam pengujian menggunakan augmentasi dan non augmentasi untuk tugas deteksi objek, dengan menunjukkan kemajuan yang signifikan dalam pengurangan error dan peningkatan akurasi serta kemampuan deteksi model.

Hasil Train pada Dataset Facemask

Bounding Box pada gambar 8 telah dicapai hasil pendeteksian yang menjanjikan untuk pengujian non augmentasi, pada salah satu gambar, meskipun gambar tersebut memiliki kualitas gambar yang rendah atau *blur*, model tetap dapat mengidentifikasi objek tersebut, YOLOv8 mampu mengidentifikasi "*WithMask 0,7*" dengan score sebesar 0,7 yang menunjukkan angka yang cukup tinggi, mengingat kondisi gambar yang kurang ideal. YOLOv8 seperti pada gambar menghasilkan *bounding box*, yaitu sebuah kotak yang mengelilingi objek tersebut dalam gambar. *Bounding box* ini memberikan informasi tentang lokasi dan ukuran objek yang terdeteksi. Meskipun gambar *blur*, *bounding box* ditempatkan dengan cukup akurat di sekitar wajah, yang menunjukkan bahwa model dapat mengekstraksi dan mengenali fitur-fitur penting meskipun detail visualnya kabur. Selain itu ukuran *bounding box* yang dihasilkan juga sesuai dengan ukuran objek, Ini menunjukkan bahwa model memiliki kemampuan generalisasi yang kuat dan dapat bekerja dengan baik bahkan ketika data input tidak sempurna. Gambar 9 memperlihatkan salah satu contoh hasil pengujian menggunakan augmentasi, walau sudah ditambahkan *Grayscale* dan augmentasi model tetap dapat mengidentifikasi hasil tersebut dengan "*WithMask 0,82*" dengan score yang cukup tinggi menandakan jika model tidak mengalami kesulitan dalam mendeteksi objek yang sudah diubah. Dengan *bounding box* yang ditempatkan dengan cukup akurat disekitar wajah.

Hasil Uji Coba dan Testing

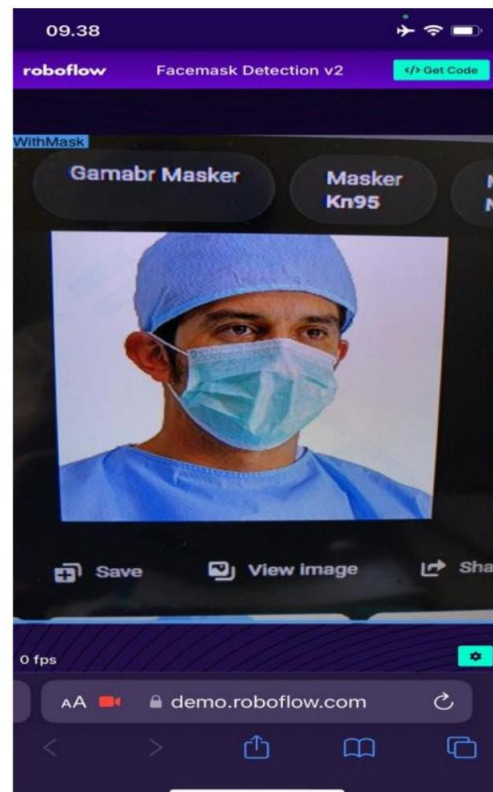
Eksperimen dilakukan dengan melakukan scan qr code pada web *roboflow* dan dicoba menggunakan *device* yang ada, seperti *smartphone* atau *webcam* dan dapat mendeteksi secara *real time*. Namun sangat penting untuk mempertimbangkan faktor-faktor yang berpotensi mempengaruhi. Variasi dalam pencahayaan kondisi pencahayaan dan skala objek yang beragam.

Kesimpulan

Dalam studi ini, penerapan model YOLOv8 dengan *framework Roboflow* pada Google Colab memberikan hasil yang memuaskan pada *FaceMask Dataset*. Dengan mAP mencapai 93,1%, *Precision* 79,7%, dan *Recall* 95,9%. Ditambah dengan penurunan *loss* yang menunjukkan jika akurasi meningkat, dan berhasil mengurangi error yang terkait dengan *bounding box* dan kelas prediksi. Secara keseluruhan YOLOv8 adalah Solusi yang efektif untuk mendeteksi penggunaan masker menggunakan *FaceMask Dataset*, dengan keunggulan dalam hal akurasi dan kecepatan. Diharapkan hasil penelitian ini memberikan dasar yang kuat untuk penerapan teknologi ini dalam konteks pengawasan kesehatan dan keamanan publik di masa depan.

Daftar Pustaka

1. Keputusan Menteri Kesehatan Republik Indonesia; 2020. Dokumen resmi pemerintah Indonesia. Peraturan Pemerintah.
2. Anjuran mengenai penggunaan masker dalam konteks COVID-19: Panduan interim; 2020. Pedoman interim WHO. World Health Organization.
3. Eikenberry SE, et al. To mask or not to mask: Modeling the potential for face mask use by the general public to curtail the COVID-19 pandemic. *Infectious Disease Modelling*. 2020 Jan;5:293-308.
4. Terven J, Cordova-Esparza D. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Machines*. 2023 Apr;5(4).
5. Loey M, Manogaran G, Taha MHN, Khalifa NEM. A hybrid deep transfer learning model with machine learning methods for face



Gambar 10. Hasil Uji coba Testing

- mask detection in the era of the COVID-19 pandemic. *Measurement*. 2021 Jan;167.
6. Bhambani K, Jain T, Sultanpure KA. Real-Time Face Mask and Social Distancing Violation Detection System using YOLO. In: *Proceedings of B-HTC 2020 - 1st IEEE Bangalore Humanitarian Technology Conference*. IEEE; 2020. .
7. YOLO-V8: Peningkatan Algoritma untuk Deteksi Pemakaian Masker Wajah; 2023. Diakses secara online. Artikel daring.
8. Voulodimos A, Doulamis N, Doulamis A, Protopapadakis E. Deep Learning for Computer Vision: A Brief Review. *Computational Intelligence and Neuroscience*. 2018.
9. Sohan M, Sai Ram T, Rami Reddy CV. A Review on YOLOv8 and Its Advancements. In: *Lecture Notes in Networks and Systems*; 2024. p. 529-45.
10. Reis D, Hong J, Kupec J, Daoudi A. Real-Time Flying Object Detection with YOLOv8; 2024. Preprint or conference work (detail publikasi tidak tersedia).
11. Lubis C. Deteksi Penggunaan Masker dan Klasifikasi Secara Real Time Melalui Video Webcam Dengan Metode YOLO; 2024. Tugas akhir/artikel lokal, detail publikasi tidak tersedia.
12. Singh S, Ahuja U, Kumar M, Kumar K, Sachdeva M. Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment. *Multimedia Tools and Applications*. 2021 May;80(13):19753-68.
13. Sapkota R, Ahmed D, Karkee M. Comparing YOLOv8 and Mask R-CNN for instance segmentation in complex orchard environments. *Artificial Intelligence in Agriculture*. 2024 Sep;13:84-99.
14. Syafira OA, Selviandro N, Wulandari GS. Safety Assurance Case for Autonomous Vehicle Object Detection: A Simulation for Smart Factory; 2024. Preprint atau prosiding, detail publikasi tidak tersedia.