

Deteksi Penggunaan Masker Wajah Menggunakan YOLOv5

1st Hasbi Dawami
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
hasbidawami@student.telkomuni-
versity.ac.id

2nd Ema Rachmawati
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
emarachmawati@telkomuni-
versity.ac.id

3rd Mahmud Dwi Sulistiyo
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
mahmuddwis@telkomuni-
versity.ac.id

Abstrak-Pandemi COVID-19 menyebabkan global krisis kesehatan. Mengenakan masker wajah menjadi salah satu protokol kesehatan yang penting dan diwajibkan oleh pemerintah. Namun, masih banyak masyarakat yang enggan mengenakan masker wajah ketika berada di ruang publik. Oleh karena itu, diperlukan sistem yang dapat mendeteksi penggunaan masker wajah pada manusia yang bertujuan untuk membantu petugas dalam menegakkan kedisiplinan masyarakat dalam rangka menerapkan salah satu protokol kesehatan tersebut. Sistem tersebut dirancang dengan model object detection yang akurat dan efisien untuk mendeteksi penggunaan masker wajah pada manusia. Tugas akhir ini membahas bagaimana membangun sistem untuk mendeteksi masker pada wajah menggunakan metode YOLOv5 menggunakan dataset face mask detection yang asli dan yang telah di augmentasi serta berbagai nilai IoU threshold mulai dari 0,1; 0,2; 0,3; 0,5 dan 0,7. YOLOv5 merupakan versi terbaru dari YOLO sehingga memiliki akurasi yang tinggi, kemampuan mendeteksi small object, serta running speed yang cepat. Hasil terbaik jika menggunakan dataset face mask detection original didapatkan dengan nilai IoU threshold sebesar 0,3 yang memiliki nilai mAP pada saat testing semua kelas sebesar 0,876. Jika menggunakan dataset face mask detection yang diaugmentasi hasil terbaik didapatkan dengan nilai IoU threshold sebesar 0,5 yang memiliki nilai mAP pada saat testing untuk semua kelas sebesar 0,849.

Kata kunci- *object detection, you only look once, akurasi, small object, running speed, IoU threshold*

Abstract- *The COVID-19 pandemic has caused a global health crisis. Wearing a face mask is one of the important health protocols and is required by the government. However, there are still many people who are reluctant to wear face masks when in public spaces. Therefore, we need a system that can detect the use of face masks in humans which aims to assist officers in enforcing community discipline in order to implement one of these health protocols. The system is designed with an accurate and efficient object detection model to detect the use of face masks in humans. This final project discusses how to build a system to detect masks on the face using the YOLOv5 method using the original and augmented face mask detection datasets and various IoU threshold values ranging from 0.1; 0.2; 0.3; 0.5 and 0.7. YOLOv5 is the latest version of YOLO so it has high accuracy, the*

ability to detect small objects, and fast running speed. The best results when using the original face mask detection dataset are obtained with an IoU threshold value of 0.3 which has an mAP value when testing all classes of 0.876. If using the augmented face mask detection dataset, the best results are obtained with an IoU threshold value of 0.5 which has an mAP value at the time of testing for all classes of 0.849.

Keywords- *object detection, you only look once, akurasi, small object, running speed, IoU threshold*

I. PENDAHULUAN

A. Latar Belakang

Pandemi Covid-19 sudah menyebar ke seluruh dunia. Salah satu cara dilakukan pemerintah untuk mengurangi dampak yang diakibatkan oleh pandemi ini adalah wajib menggunakan masker ketika melakukan aktivitas diluar rumah.

Namun, masih banyak masyarakat yang mengabaikan salah satu protokol kesehatan tersebut. Pengawasan yang dilakukan juga masih belum efektif yang diakibatkan oleh banyaknya masyarakat yang melakukan aktivitas di area publik serta kurangnya petugas yang mengawasi.

Untuk mengatasi permasalahan tersebut, maka dibangun sebuah sistem untuk mendeteksi penggunaan masker yang diharapkan dapat membantu petugas untuk mengawasi penggunaan masker di area publik. Sebelumnya, sudah ada beberapa penelitian yang berhasil membangun sistem untuk mendeteksi masker seperti [2] menggunakan model MobileNetV2, dan [14] menggunakan model YOLOv4.

Selain model tersebut, ada beberapa model yang bisa digunakan untuk membangun sistem tersebut, antara lain: InceptionV3, Xception, MobileNet, MobileNetV2, VGG16, ResNet50, dan You Only Look Once (YOLO) [1].

Pada penelitian ini, penulis menggunakan metode YOLOv5. YOLOv5 merupakan versi terbaru dari YOLO sehingga memiliki tingkat akurasi yang lebih tinggi dan kemampuan untuk mendeteksi objek kecil yang di klaim lebih baik dari versi sebelumnya [3].

B. Identifikasi Masalah

Berdasarkan latar belakang sebelumnya, perumusan masalah yang ada pada tugas akhir ini adalah bagaimana merancang dan membuat sebuah sistem untuk mendeteksi penggunaan masker wajah menggunakan metode YOLOv5.

C. Tujuan

Tujuan tugas akhir ini adalah membangun dan menganalisis performa dari sistem untuk mendeteksi penggunaan masker wajah menggunakan metode YOLOv5.

D. Batasan Masalah

Adapun batasan masalah pada tugas akhir ini adalah sebagai berikut:

1. Dataset yang digunakan adalah *Face Mask Detection*.
2. Dataset terdiri dari tiga kelas, yaitu: *with_mask*, *without_mask*, dan *mask_wearred_incorrect*.
3. Format citra pada dataset berupa png.
4. Jumlah dataset yang digunakan terbatas.

E. Rencana Kegiatan

1. Kajian Pustaka
Proses kajian pustaka pada tugas akhir ini adalah kegiatan untuk mempelajari berbagai teori yang berkaitan dengan tugas akhir ini, seperti dataset yang digunakan untuk melatih model, metode yang digunakan untuk deteksi objek, dan implementasi metode YOLO pada berbagai sistem deteksi objek khususnya deteksi penggunaan masker.
2. Pengumpulan Data
Pada proses ini kegiatan yang dilakukan adalah mengumpulkan dataset yang akan digunakan dalam penelitian tugas akhir ini.
3. Perancangan sistem
Proses perancangan sistem pada tugas akhir ini adalah melakukan perancangan mengenai sistem yang akan dibangun selama tugas akhir.
4. Pengujian Tugas Akhir
Proses pengujian pada tugas akhir ini adalah melakukan pengujian berdasarkan model yang sebelumnya sudah dirancang dan dataset yang telah ditentukan.
5. Analisis Hasil dan Penulisan Laporan Tugas Akhir
Proses analisis hasil dan penulisan laporan tugas akhir adalah melakukan analisis terkait hasil pengujian tugas akhir

dan juga penulisan laporan analisis hasil pengujian berdasarkan pengujian yang telah dilakukan sehingga mendapatkan kesimpulan tugas akhir.

II. KAJIAN TEORI

A. Penelitian Terkait

Giancini, dkk. [4] menggunakan bobot pra-latih yang telah disediakan untuk dilatih ulang, dan *network* yang digunakan adalah YOLOv3-tiny yang memiliki kemiripan dengan *backbone* Darknet-19. Hasilnya memiliki akurasi sebesar 95 sampai 100%.

Cao, dkk. [5] menggunakan model berbasis YOLOv4. Hasilnya MaskHunter mencapai kinerja lebih baik dibanding yang lain.

Nithiyasree, dkk. [6] menggunakan arsitektur ResNet50 CNN yang telah dilatih sebelumnya menggunakan dataset ImageNet. Model tersebut menghasilkan nama siswa yang tidak memakai masker di kelas dari video langsung yang diambil dari kamera yang dipasang di kelas.

Yao, dkk. [7] menggunakan model YOLOv5. Tujuan dari penelitian ini adalah mendeteksi cacat pada buah kiwi. Hasilnya model tersebut memiliki akurasi 94-98%. Model YOLOv5 juga digunakan pada [8]. Model tersebut digunakan untuk mendeteksi masker wajah. Hasilnya memiliki akurasi sebesar 96,5%. Model YOLOv5 juga digunakan pada [9]. Model tersebut digunakan untuk mendeteksi penggunaan *safety helmet*. Hasilnya memiliki akurasi sebesar 94,7%.

B. YOLO

You Only Look Once (YOLO) merupakan model yang menggunakan *Region-based convolutional neural network* (R-CNN) untuk mendeteksi objek. Model ini dapat melakukan banyak deteksi objek, memprediksi *class* yang dibuat, serta mengidentifikasi lokasi dari objek tersebut. YOLO diklaim sebagai arsitektur yang sangat cepat dan akurat. Namun, akurasi arsitektur tersebut dipengaruhi oleh beberapa variabel[4].

YOLO memiliki beberapa versi, yaitu:

1. YOLOv1
YOLOv1 merupakan versi pertama dari model YOLO. pada versi ini citra *input* hanya diproses sekali, parameter kernel konvolusi dibagikan setiap kali, dan fitur yang berbeda diekstraksi melalui beberapa lapisan konvolusi. Model ini memiliki kecepatan deteksi objek yang sangat cepat. Namun, kelemahannya adalah tidak dapat mendeteksi objek kecil dan akurasi pendeteksian posisi rendah [10].
2. YOLOv2
YOLOv2 memperbaiki YOLOv1. Jaringan *backbone* ditingkatkan. YOLOv2 menggunakan rata-rata *pooling*, klasifikasi *SoftMax* dan *Anchor prediction box*, dan

juga menggabungkan metode pelatihan klasifikasi dan target. Karena beberapa peningkatan tersebut, akurasi meningkat untuk deteksi objek kecil [11].

3. YOLOv3

YOLOv3 memiliki beberapa peningkatan dibanding YOLOv2, lapisan konvolusi meningkat 2,8 kali lipat dari YOLOv2 dan *SoftMax classifiers* juga diganti dengan *multiple Logistic classifiers*. Hal tersebut meningkatkan kedalaman dan ketebalan jaringan sehingga akurasi model meningkat[12].

4. YOLOv4

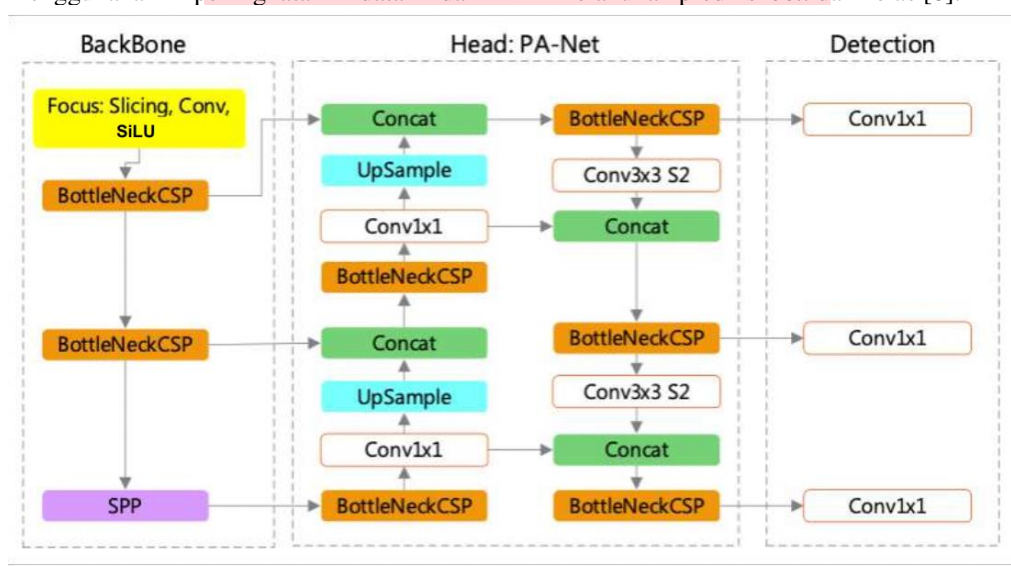
YOLOv4 muncul pada tahun 2019. Tujuan utamanya adalah untuk merancang sistem deteksi objek cepat yang dapat diterapkan di lingkungan kerja nyata dan dapat dioptimalkan secara paralel. Ini menggunakan peningkatan data dan

beberapa trik jaringan *deep learning* terbaru dalam beberapa tahun terakhir[13].

5. YOLOv5 merupakan versi terbaru dari YOLO. YOLOv5 memiliki basis yang sama dengan YOLOv4 dengan beberapa peningkatan, hasilnya *running speed* sangat meningkat dengan ukuran lebih kecil, dan berat file hampir 90% lebih kecil dari YOLOv4[3].

C. YOLOv5

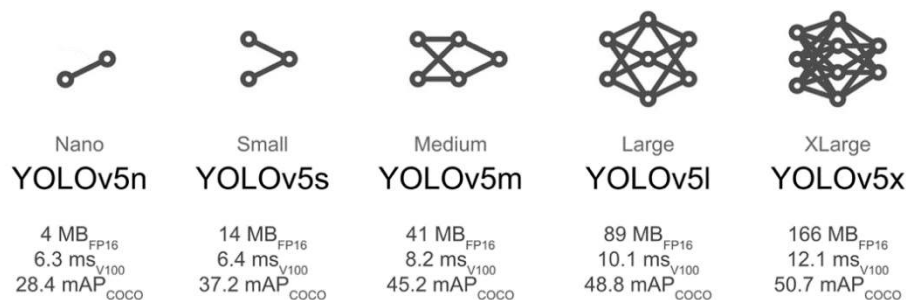
YOLOv5 memiliki tiga komponen utama yaitu: *backbone*, *head*, dan *detection*. *Backbone* merupakan CNN yang bertugas untuk mengumpulkan dan membentuk fitur citra pada perincian berbeda. *Head* merupakan serangkaian lapisan yang bertugas untuk menggabungkan fitur citra untuk proses prediksi. *Detection* merupakan proses yang menggunakan fitur dari *head* dan melakukan prediksi *box* dan kelas [8].



GAMBAR 2 ARSITEKTUR YOLOV5 [8]

YOLOv5 memiliki beberapa model yaitu: YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x. Perbedaan model yolo terdapat pada GAMBAR 3. Untuk penggunaan *mobile*

direkomendasikan menggunakan YOLOv5s atau YOLOv5m dan untuk penggunaan *cloud* direkomendasikan menggunakan YOLOv5l atau YOLOv5x [16].



GAMBAR 3 JENIS MODEL YOLOV5 [16]

YOLOv5 menggunakan fungsi aktivasi Sigmoid-weighted Linear Units(SiLU) pada persamaan berikut.

$$a_k(z_k) = z_k \sigma(z_k) \tag{1}$$

$$z_k = \sum_i \omega_{ik} S_i + b_k \tag{2}$$

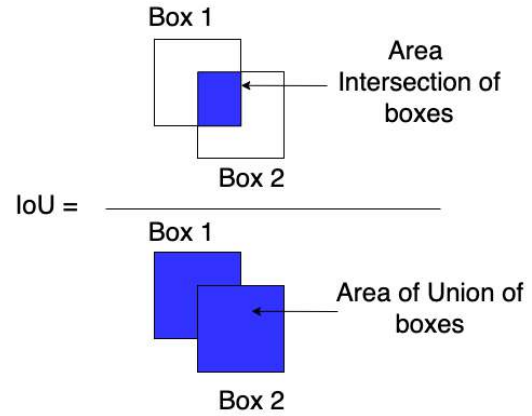
$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

Dimana z_k adalah input *hidden unit k*, $\sigma(\cdot)$ merupakan fungsi sigmoid, ω_{ik} adalah beban penghubung dan *hidden unit k* dan b_k beban bias untuk *hidden unit*.

D. Intersection Over Union

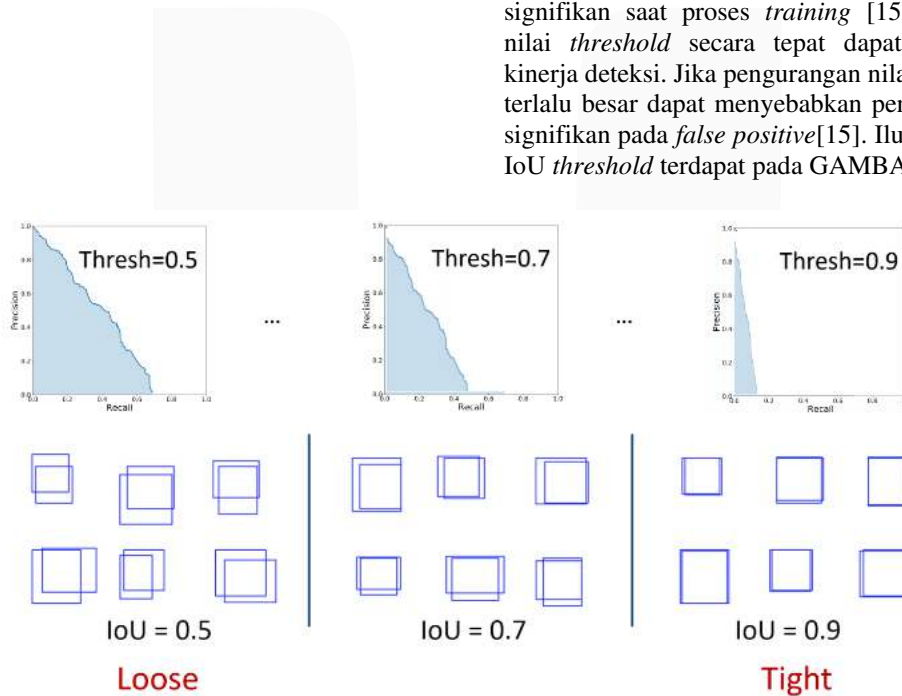
Intersection Over Union (IoU) merupakan nilai berdasarkan statistik kesamaan dan keragaman set sampel bertujuan untuk mengevaluasi area tumpang tindih (area yang beririsan) antara dua *bounding box*, yaitu *bounding box* hasil prediksi model dan *bounding box* data asli. IoU memiliki nilai minimum agar hasil prediksi dianggap benar positif yang disebut *IoU threshold*. Berikut terdapat persamaan IoU.

$$IoU = \frac{\text{Area of Intersection of two boxes}}{\text{Area of Union of two boxes}} \tag{4}$$



GAMBAR 4. PERSAMAAN IOU

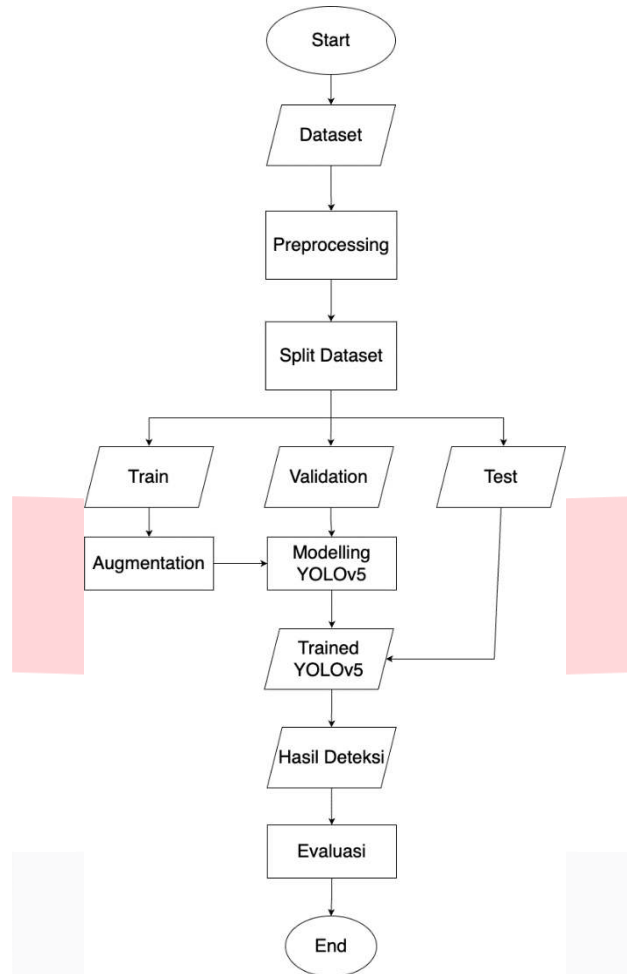
IoU threshold merupakan nilai ambang batas nilai IoU suatu citra yang berhasil di deteksi oleh model yang digunakan untuk menentukan apakah *bounding box* termasuk *true positive* atau *false positive*. Nilai *IoU threshold* dapat mempengaruhi nilai mAP. *IoU threshold* memiliki nilai antara 0 sampai 1. Jika menggunakan nilai *IoU threshold* yang rendah maka model memiliki jumlah *false positive* yang besar. Sebaliknya, jika menggunakan nilai *threshold* yang lebih tinggi dapat menyebabkan pengurangan *True positive* yang sangat signifikan saat proses *training* [15]. Pengurangan nilai *threshold* secara tepat dapat meningkatkan kinerja deteksi. Jika pengurangan nilai *IoU threshold* terlalu besar dapat menyebabkan pengurangan yang signifikan pada *false positive*[15]. Ilustrasi mengenai *IoU threshold* terdapat pada GAMBAR 5.



GAMBAR 5. ILUSTRASI IOU THRESHOLD [17]

III. METODE

Penelitian ini menggunakan YOLOv5m untuk membuat sistem yang mendeteksi penggunaan masker wajah. Rancangan dari sistem deteksi penggunaan masker wajah menggunakan YOLOv5 terdapat pada GAMBAR 6.



GAMBAR 6. PERANCANGAN SISTEM

A. Dataset

Penelitian ini menggunakan dataset *Face Mask Detection*. Dataset tersebut berupa citra dengan format png. Total dari dataset sebesar 848 citra. Dataset dibagi menjadi tiga kelas, yaitu *with_mask*, *without_mask*, *mask_wearred_incorrect*. Citra pada

dataset ini sudah dilakukan anotasi sehingga tidak perlu melakukan anotasi terlebih dahulu. Contoh citra yang ada pada dataset yang akan digunakan pada penelitian ini terdapat pada GAMBAR 7.



GAMBAR 7. CONTOH DATASET FACE MASK DETECTION

B. Preprocessing

Preprocessing merupakan tahap yang terdiri dari *split* dataset dan melakukan *resize* pada citra. *Split* dataset bertujuan untuk memisahkan antara data *train*, *validation*, dan *test*. *Resize* pada citra dilakukan

untuk mengubah ukuran pada citra menjadi 640 x 640 piksel yang bertujuan untuk menyamakan semua ukuran citra dan menyesuaikan kebutuhan model.

C. Augmentasi

Augmentasi yang dilakukan pada citra data *train* berupa memutar citra antara -20° sampai dengan $+20^\circ$, menambah *noise* hingga 5%, serta dilakukan

zoom maksimal 25% pada citra. Contoh citra data *train* yang sudah dilakukan augmentasi terdapat pada GAMBAR 8.



GAMBAR 8.
CONTOH DATASET FACE MASK DETECTION YANG TELAH DIAUGMENTASI

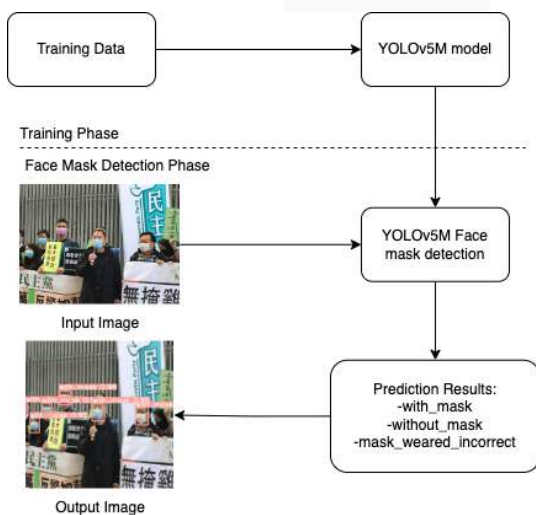
D. Pemodelan menggunakan YOLOv5

Pada tahapan ini pemodelan dilakukan dengan implementasi menggunakan model YOLOv5M. Model dibagi menjadi dua bagian yaitu model pelatihan dan model deteksi masker wajah. Dalam model pelatihan, dataset asli (*original*) digunakan untuk melatih model. Selain itu, ada model pelatihan lain yang dilatih menggunakan dataset yang telah diaugmentasi. Model yang dilatih akan menghasilkan model deteksi masker wajah dengan citra dari dataset *original* sebagai input dan diproses sehingga dapat memprediksi tiga kelas yaitu *with_mask*, *without_mask*, dan *mask_wearred_incorrect*.

akhirnya citra tersebut menjadi feature map berukuran $320 \times 320 \times 32$. Modul CBL merupakan modul konvolusi dasar. Modul ini terdiri dari Conv2D + BatchNormal + SiLU [9].

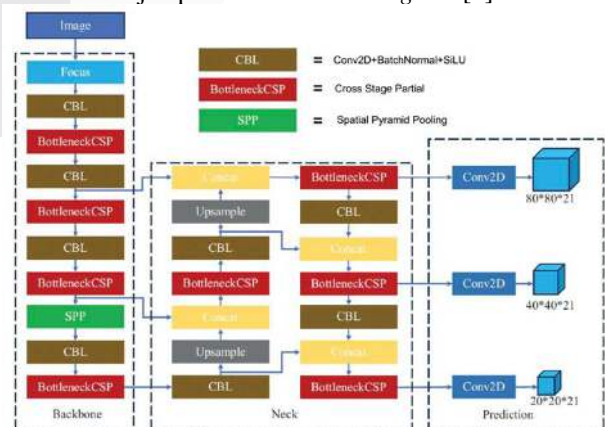
Modul BottleneckCSP melakukan ekstraksi fitur pada feature map, mengekstraksi banyak informasi dari citra. Struktur BottleneckCSP dapat mengurangi duplikasi informasi gradien dalam proses optimasi CNN. Modul SPP dapat memperbanyak bidang penerima dari network dan memperoleh fitur pada skala berbeda [9].

YOLOv5 menambahkan bottom-up feature pyramids structure berdasarkan struktur FPN. Dengan kombinasi ini, lapisan FPN menyampaikan fitur semantik yang kuat dari atas ke bawah, dan piramida fitur menyampaikan fitur pemosisian yang kuat dari bawah ke atas. Gabungan agregasi fitur dari lapisan fitur yang berbeda untuk meningkatkan kemampuan jaringan dalam mendeteksi target skala yang berbeda. outputnya adalah hasil klasifikasi dan koordinat objek pada citra dataset *original* [9].



GAMBAR 9.
PROSES DETEKSI MASKER MENGGUNAKAN YOLOV5M

Untuk tahapan proses deteksi, pada backbone citra asli dataset dengan resolusi $640 \times 640 \times 3$ sebagai input dipecah menjadi feature map berukuran $320 \times 320 \times 12$ dan kemudian digabungkan melalui operasi *concat*. Setelah itu, dilakukan convolution operation dari 32 convolution kernels



GAMBAR 10.
NETWORK STRUCTURE YOLOV5

E. Evaluasi Model

Tahap ini bertujuan untuk mengevaluasi kinerja model dengan cara mengukur *precision*, yaitu rasio antara objek yang diprediksi benar berbanding dengan keseluruhan hasil yang diprediksi oleh model, *recall*, yaitu rasio antara objek yang diprediksi benar berbanding keseluruhan hasil aktualnya [7], *mean average precision*(mAP) dihitung menggunakan nilai *precision* dan *recall*. *Precision*(P), *recall*(R), *average precision*(AP), *mean average precision*(mAP), dapat dihitung dengan menggunakan persamaan dibawah ini.

$$Precision = \frac{TP}{(TP + FP)} \quad (5)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (6)$$

$$mAP = \frac{1}{11} \sum_{Recall_i} Precision(Recall_i) \quad (7)$$

$$mAP = \frac{1}{11} \sum_{(0,0.1,0.2,0.3,\dots,0.9,1)} Precision(Recall_i) \quad (7)$$

TP merupakan *true positive* yang berarti data aktual benar dan model memprediksi dengan benar positif. TN merupakan *true negative* yang berarti data aktual salah dan model memprediksi dengan benar negatif. FP merupakan *False Positive* yang berarti data aktual salah tetapi model memprediksi dengan benar positif. FN merupakan *false negative* yang berarti data aktual benar tetapi model memprediksi dengan benar negatif [9].

IV. HASIL DAN PEMBAHASAN

Bagian ini berisi dua sub-bagian, yaitu Hasil Pengujian dan Analisis Hasil Pengujian. Pengujian dan analisis yang dilakukan selaras dengan tujuan TA sebagaimana dinyatakan dalam Pendahuluan.

A. Hasil Pengujian

Pada penelitian ini penulis melakukan pengujian menggunakan model YOLOv5M dengan menggunakan dataset *original* dan dataset yang diaugmentasi. Dataset di *split* menjadi tiga bagian yaitu: *training*, *validation*, *testing*. Dataset dilakukan augmentasi pada bagian data *training*. Oleh karena itu, terdapat perbedaan antara jumlah data *train* telah dilakukan augmentasi dengan data *train original*.

Perbedaan jumlah citra dan label pada dataset *original* dan dataset yang diaugmentasi terdapat pada TABEL 1 dan TABEL 2.

TABEL 1
RINCIAN JUMLAH CITRA DATASET

Jenis Data	Dataset
------------	---------

	Original	Augmentasi
Training	593 Citra	1782 Citra
Validation	170 Citra	169 Citra
Testing	85 Citra	85 Citra
Total	848 Citra	2036 Citra

TABEL 2
RINCIAN JUMLAH LABEL DATASET

Training	Dataset	
	Original	Augmentasi
with_mask	565 Label	1603 Label
without_mask	192 Label	552 Label
mask_wearred_incorrect	68 Label	200 Label
Total	825 Label	2355 Label
Validation	Dataset	
	Original	Augmentasi
with_mask	153 Label	153 Label
without_mask	60 Label	56 Label
mask_wearred_incorrect	18 Label	20 Label
Total	231 Label	229 Label
Testing	Dataset	
	Original = Augmentasi	
with_mask	74 Label	
without_mask	36 Label	
mask_wearred_incorrect	6 Label	
Total	116 Label	

Pengujian dilakukan dengan menggunakan dataset original dan dataset yang telah diaugmentasi. Dengan jumlah *epoch* yang sama sebesar 100, *batch size* 20, *image size* 640x640 piksel, dan *IoU threshold* yang berbeda mulai dari 0,1; 0,2; 0,3; 0,5 dan 0,7.

1. Hasil Pengujian menggunakan Dataset *Original*

TABEL 3
HASIL PENGUJIAN UNTUK SEMUA KELAS MENGGUNAKAN DATASET *ORIGINAL*

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,899	0,803	0,885	0,874	0,776	0,840
0,2	0,912	0,787	0,869	0,861	0,792	0,818
0,3	0,791	0,857	0,872	0,916	0,843	0,876
0,5	0,811	0,818	0,860	0,827	0,802	0,837
0,7	0,805	0,722	0,737	0,814	0,723	0,745

Pada

TABEL 3 terdapat hasil pengujian pada semua kelas. Pada *validation*, nilai mAP tertinggi sebesar

0,885 didapat dari penggunaan IoU *threshold* sebesar 0,1. Pada *testing*, nilai mAP tertinggi sebesar 0,876 didapat dari penggunaan IoU *threshold* sebesar 0,3.

TABEL 4
HASIL PENGUJIAN UNTUK KELAS MASK_WEARED_INCORRECT MENGGUNAKAN DATASET *ORIGINAL*

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,860	0,630	0,768	0,733	0,667	0,663
0,2	0,872	0,593	0,708	0,740	0,635	0,628
0,3	0,625	0,704	0,716	0,863	0,778	0,760
0,5	0,750	0,630	0,721	0,706	0,667	0,668
0,7	0,717	0,481	0,513	0,693	0,556	0,528

Pada

TABEL 4 terdapat hasil pengujian pada kelas *mask_wearred_incorrect*. Pada *validation*, nilai mAP

tertinggi sebesar 0,768 didapat dari penggunaan IoU *threshold* sebesar 0,1. Pada *testing*, nilai mAP tertinggi sebesar 0,760 didapat dari penggunaan IoU *threshold* sebesar 0,3.

TABEL 5
HASIL PENGUJIAN UNTUK KELAS WITH_MASK MENGGUNAKAN DATASET *ORIGINAL*

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,928	0,924	0,971	0,977	0,931	0,979
0,2	0,934	0,894	0,958	0,964	0,961	0,975
0,3	0,909	0,935	0,962	0,964	0,955	0,972

0,5	0,877	0,944	0,961	0,935	0,961	0,970
0,7	0,890	0,880	0,898	0,921	0,902	0,918

Pada

TABEL 5 terdapat hasil pengujian pada kelas *with_mask*. Pada *validation*, nilai mAP tertinggi sebesar 0,971 didapat dari penggunaan *IoU threshold*

sebesar 0,1. Pada *testing*, nilai mAP tertinggi sebesar 0,979 didapat dari penggunaan *IoU threshold* sebesar 0,1.

TABEL 6
HASIL PENGUJIAN UNTUK KELAS WITHOUT_MASK MENGGUNAKAN DATASET ORIGINAL

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,910	0,856	0,917	0,913	0,729	0,878
0,2	0,928	0,873	0,942	0,880	0,780	0,851
0,3	0,839	0,932	0,939	0,922	0,796	0,897
0,5	0,807	0,881	0,897	0,839	0,780	0,873
0,7	0,810	0,805	0,801	0,829	0,712	0,788



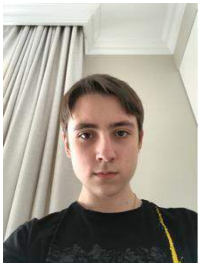
Pada

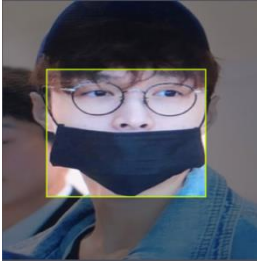
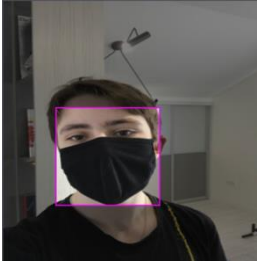
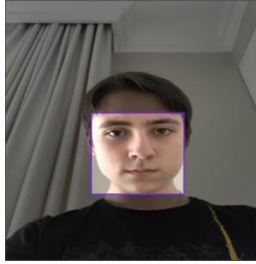

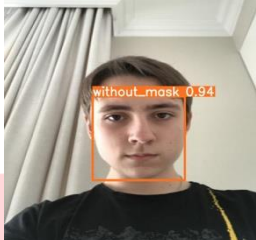

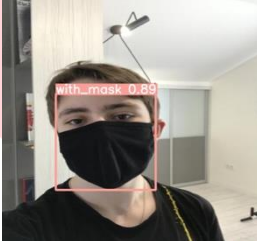
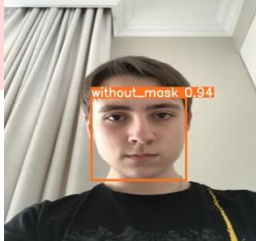

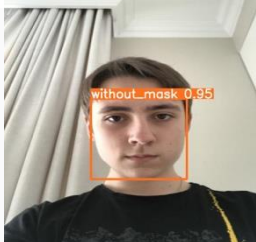

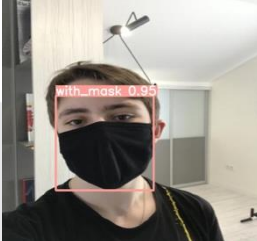
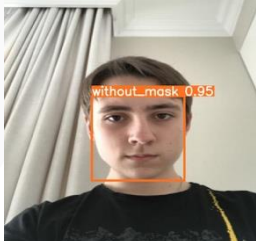

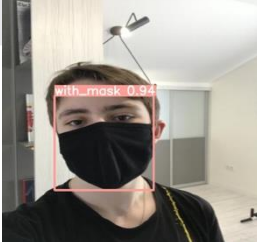
TABEL 6 terdapat hasil pengujian pada kelas *without_mask*. Pada *validation*, nilai mAP tertinggi

sebesar 0,942 didapat dari penggunaan *IoU threshold* sebesar 0,2. Pada *testing*, nilai mAP tertinggi sebesar 0,897 didapat dari penggunaan *IoU threshold* sebesar 0,3.

TABEL

7 HASIL PREDIKSI MODEL YANG MENGGUNAKAN DATASET ORIGINAL

Dataset: Original	Sample 1	Sample 2	Sample 3
Citra Input			

<p>Ground Truth</p>			
<p>IoU_T 0,1</p>			
<p>IoU_T 0,2</p>			
<p>IoU_T 0,3</p>			
<p>IoU_T 0,5</p>			
<p>IoU_T 0,7</p>			

Pada

TABEL 7 terdapat citra input, ground truth, dan hasil prediksi model.

2. Hasil Pengujian menggunakan Dataset yang diaugmentasi

TABEL 8
HASIL PENGUJIAN UNTUK SEMUA KELAS

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,914	0,729	0,833	0,849	0,721	0,809
0,2	0,858	0,735	0,824	0,968	0,677	0,811
0,3	0,920	0,737	0,823	0,817	0,788	0,822
0,5	0,854	0,759	0,815	0,911	0,692	0,849
0,7	0,769	0,605	0,634	0,896	0,624	0,674

TABEL 8 terdapat hasil pengujian

pada semua kelas. Pada *validation*, nilai mAP tertinggi sebesar 0,833 didapat dari penggunaan IoU *threshold* sebesar

Pada

0,2. Pada *testing*, nilai mAP tertinggi sebesar 0,849 didapat dari penggunaan IoU *threshold* sebesar 0,5

TABEL 9
HASIL PENGUJIAN UNTUK KELAS MASK_WEARED_INCORRECT

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,922	0,519	0,671	0,727	0,500	0,625
0,2	0,871	0,444	0,630	0,977	0,500	0,615
0,3	1,000	0,461	0,627	0,793	0,642	0,672
0,5	0,821	0,519	0,646	0,900	0,500	0,805
0,7	0,724	0,370	0,430	1,000	0,496	0,513

Pada

TABEL 9 terdapat hasil pengujian pada kelas *mask_wearred_incorrect*. Pada *validation*, nilai mAP tertinggi sebesar 0,671 didapat dari penggunaan IoU

threshold sebesar 0,1. Pada *testing*, nilai mAP tertinggi sebesar 0,805 didapat dari penggunaan IoU *threshold* sebesar 0,5.

TABEL 10 HASIL PENGUJIAN UNTUK KELAS WITH_MASK

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,924	0,903	0,961	0,927	0,874	0,942
0,2	0,857	0,943	0,958	0,959	0,855	0,943
0,3	0,884	0,944	0,961	0,848	0,911	0,937
0,5	0,889	0,941	0,954	0,940	0,862	0,920
0,7	0,812	0,830	0,831	0,842	0,786	0,819

TABEL 10 terdapat hasil pengujian pada kelas *with_mask*. Pada *validation*, nilai mAP tertinggi

Pada

sebesar 0,961 didapat dari penggunaan IoU *threshold* sebesar 0,1 dan 0,3. Pada *testing*, nilai mAP tertinggi sebesar 0,943 didapat dari penggunaan IoU *threshold* sebesar 0,2


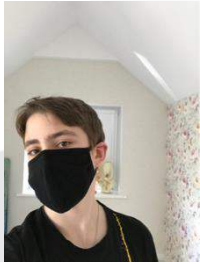
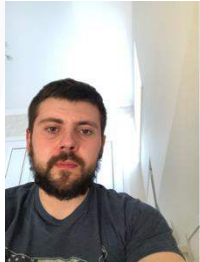

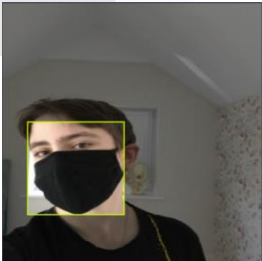
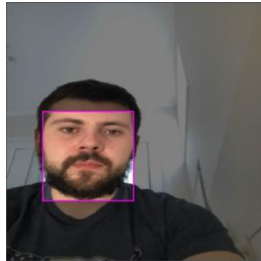
TABEL 11
HASIL PENGUJIAN UNTUK KELAS WITHOUT_MASK

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,897	0,765	0,865	0,892	0,788	0,861
0,2	0,847	0,818	0,883	0,966	0,676	0,874
0,3	0,875	0,806	0,882	0,810	0,810	0,858
0,5	0,852	0,818	0,846	0,892	0,714	0,821
0,7	0,771	0,615	0,640	0,847	0,591	0,691

Pada TABEL 11 terdapat hasil pengujian pada kelas *without_mask*. Pada *validation*, nilai mAP tertinggi sebesar 0,897 didapat dari penggunaan IoU *threshold*

sebesar 0,1. Pada *testing*, nilai mAP tertinggi sebesar 0,874 didapat dari penggunaan IoU *threshold* sebesar 0,2.

TABEL 12
HASIL PREDIKSI MODEL

Dataset: <i>Augmented</i>	<i>Sample 1</i>	<i>Sample 2</i>	<i>Sample 3</i>
<i>Citra Input</i>			
<i>Ground Truth</i>			

IoU_T 0,1			
IoU_T 0,2			
IoU_T 0,3			
IoU_T 0,5			
IoU_T 0,7			

Pada

TABEL 12 terdapat citra *input*, *ground truth*, dan hasil prediksi model.

3. Hasil Pengujian menggunakan model YOLOv3

Berikut ini merupakan hasil pengujian model YOLOv3 yang dilatih menggunakan dataset yang diaugmentasi. Pengujian dilakukan menggunakan IoU *threshold* berbeda mulai dari 0,1;0,2;0,3;0,5 dan 0,7. Dengan jumlah *epoch* yang sama sebesar 100,

batch size 20, image size 640x640 piksel. Pengujian ini dilakukan untuk melihat perbedaan hasil antara YOLOv3 dan YOLOv5M jika dilatih menggunakan

dataset yang diaugmentasi. Berikut hasil pengujian dan prediksi model YOLOv3.

TABEL 13
HASIL PENGUJIAN UNTUK SEMUA KELAS

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,840	0,760	0,837	0,893	0,723	0,815
0,2	0,918	0,707	0,814	0,921	0,709	0,827
0,3	0,894	0,774	0,838	0,874	0,807	0,857
0,5	0,915	0,715	0,801	0,926	0,709	0,784
0,7	0,771	0,659	0,676	0,821	0,601	0,638

TABEL 13 terdapat hasil pengujian untuk semua kelas. Pada validation, nilai mAP tertinggi sebesar

Pada

0,838 didapat dari penggunaan IoU threshold sebesar 0,3. Pada testing, nilai mAP tertinggi sebesar 0,857 didapat dari penggunaan IoU threshold sebesar 0,3.

TABEL 14
HASIL PENGUJIAN UNTUK KELAS MASK_WEARED_INCORRECT

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,875	0,518	0,676	0,824	0,500	0,604
0,2	0,894	0,481	0,642	0,895	0,481	0,643
0,3	0,927	0,556	0,685	0,827	0,833	0,759
0,5	1,000	0,400	0,610	0,941	0,500	0,572
0,7	0,854	0,432	0,487	0,725	0,500	0,454

TABEL 14 terdapat hasil pengujian untuk kelas mask_weared_incorrect. Pada validation, nilai mAP tertinggi sebesar 0,685 didapat dari penggunaan IoU

Pada

threshold sebesar 0,3. Pada testing, nilai mAP tertinggi

sebesar 0,759 didapat dari penggunaan IoU threshold sebesar 0,3.

TABEL 15
HASIL PENGUJIAN UNTUK KELAS WITH_MASK

IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,844	0,951	0,963	0,938	0,884	0,942
0,2	0,928	0,915	0,957	0,928	0,916	0,960
0,3	0,897	0,943	0,961	0,938	0,884	0,945
0,5	0,878	0,934	0,957	0,923	0,865	0,918
0,7	0,745	0,845	0,842	0,886	0,757	0,811

TABEL 15 terdapat hasil pengujian untuk kelas with_mask. Pada validation, nilai mAP tertinggi sebesar 0,963 didapat dari

Pada

penggunaan IoU threshold sebesar 0,1.

Pada testing, nilai mAP tertinggi sebesar 0,960 didapat dari penggunaan IoU threshold sebesar 0,2.

TABEL 16
HASIL PENGUJIAN UNTUK KELAS WITHOUT_MASK

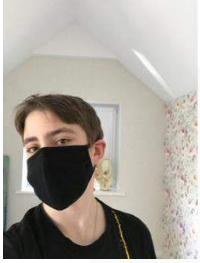
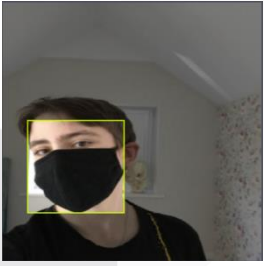
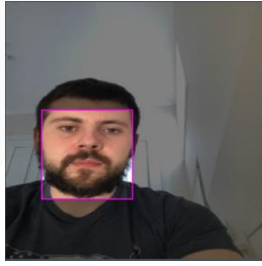
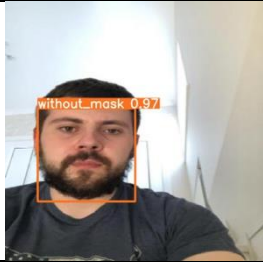
IoU_T	Validation			Testing		
	Prec	Rec	mAP	Prec	Rec	mAP
0,1	0,802	0,812	0,873	0,917	0,786	0,900
0,2	0,932	0,724	0,843	0,939	0,729	0,877
0,3	0,859	0,823	0,869	0,855	0,703	0,866
0,5	0,868	0,812	0,836	0,914	0,762	0,861
0,7	0,714	0,700	0,699	0,852	0,547	0,648

Pada

TABEL 16 terdapat hasil pengujian untuk kelas *without_mask*. Pada *validation*, nilai mAP tertinggi sebesar 0,873 didapat dari

penggunaan IoU *threshold* sebesar 0,1. Pada *testing*, nilai mAP tertinggi sebesar 0,900 didapat dari penggunaan IoU *threshold* sebesar 0,1.

TABEL 17
HASIL PREDIKSI MODEL

Dataset: <i>Augmented</i>	<i>Sample 1</i>	<i>Sample 2</i>	<i>Sample 3</i>
<i>Citra Input</i>			
<i>Ground Truth</i>			
IoU_T 0,1			

IoU_T 0,2			
IoU_T 0,3			
IoU_T 0,5			
IoU_T 0,7			

Pada TABEL 17 terdapat hasil prediksi model. Tabel ini berisi citra *input*, *ground truth*, dan hasil prediksi model.

TABEL 3, dilihat dari mAP pada *testing* untuk semua kelas, jika menggunakan dataset *original* dengan IoU *threshold* sebesar 0,3 memberikan hasil paling baik diantara penggunaan IoU *threshold* lainnya yang menggunakan dataset *original* dengan nilai mAP pada *testing* untuk semua kelas sebesar 0,876. Berdasarkan hasil pengujian pada

B. Analisis Hasil Pengujian

1. Nilai mAP Berdasarkan IoU *Threshold*

Berdasarkan hasil pengujian pada

TABEL 8, dilihat dari mAP pada *testing* untuk semua kelas, jika menggunakan dataset yang telah diaugmentasi dengan nilai IoU *threshold* sebesar 0,5 memberikan hasil paling baik diantara penggunaan IoU *threshold* lainnya yang menggunakan dataset yang telah di augmentasi dengan nilai mAP pada *testing* untuk semua kelas sebesar 0,849.

Jika melihat hasil nilai mAP untuk setiap kelas yang menggunakan dataset *original* terdapat pola naik turun, contohnya ketika menggunakan IoU *threshold* sebesar 0,1 memiliki nilai mAP pada *testing* untuk setiap kelas sebesar 0,840. Lalu ketika menggunakan IoU *threshold* sebesar 0,2 nilai turun menjadi 0,818. Lalu ketika menggunakan IoU *threshold* sebesar 0, nilai tersebut naik menjadi 0,876.

Berbeda dengan dataset yang telah diaugmentasi, hasil nilai mAP untuk setiap kelas yang TABEL 13, model YOLOv3 yang dilatih menggunakan dataset yang diaugmentasi nilai mAP tertinggi pada *testing* didapatkan ketika menggunakan IoU *threshold* sebesar 0,3 dengan nilai mAP untuk setiap kelas sebesar 0,857. Hasil nilai mAP pada pengujian ini cenderung naik pada saat penggunaan nilai IoU *threshold* sebesar 0,1 sampai 0,3 lalu turun pada saat penggunaan IoU *threshold* sebesar 0,5 sampai 0,7

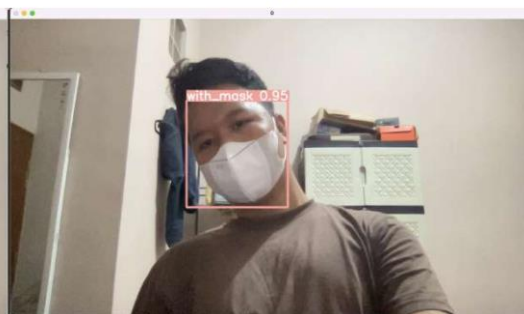
TABEL 3 dan

TABEL 8 jika menggunakan dataset *original*, hasil terbaik didapatkan dengan IoU *threshold* sebesar 0,3. Jika menggunakan dataset yang diaugmentasi, hasil terbaik didapatkan dengan IoU *threshold* sebesar 0,5. jika menggunakan model YOLOv3 dan dilatih menggunakan dataset yang diaugmentasi, hasil terbaik didapatkan dengan IoU *threshold* sebesar 0,3. Terlihat bahwa dengan melakukan augmentasi pada data *train* dapat menghasilkan model yang lebih baik karena citra data latih menjadi lebih banyak dan bervariasi sehingga menghasilkan model deteksi objek yang memiliki kemampuan yang lebih baik.

IoU_T 0,5 Original



IoU_T 0,5 Augmented



GAMBAR 11
CONTOH PERBANDINGAN HASIL DETEKSI

menggunakan dataset yang telah diaugmentasi cenderung naik pada saat menggunakan IoU *threshold* antara 0,2 sampai 0,5 dengan nilai mAP pada *testing* tertinggi sebesar 0,849 didapat pada penggunaan IoU *threshold* sebesar 0,5 dan turun menjadi 0,674 untuk nilai mAP pada *testing* pada saat menggunakan IoU *threshold* sebesar 0,7.

Oleh karena itu, tidak menutup kemungkinan bahwa penggunaan IoU *threshold* antara 0,3 sampai 0,5 dapat menghasilkan model yang lebih baik untuk dataset *original* dan untuk dataset yang diaugmentasi dapat menghasilkan model yang lebih baik dengan penggunaan IoU *threshold* antara 0,3 sampai kurang dari 0,7.

Berdasarkan hasil pengujian pada

Oleh karena itu tidak menutup kemungkinan bahwa model YOLOv3 jika dilatih menggunakan dataset yang diaugmentasi dapat menghasilkan model yang lebih baik dengan penggunaan IoU *threshold* antara 0,3 sampai kurang dari 0,5.

2. Pengaruh Augmentasi Data terhadap Nilai IoU *Threshold*

Berdasarkan hasil pengujian pada

Selain itu, model yang dilatih menggunakan dataset yang telah diaugmentasi memiliki kemampuan mendeteksi objek miring lebih baik. Contoh perbandingan hasil deteksi terdapat pada

GAMBAR 11. Saat mendeteksi pengguna masker yang memiringkan wajahnya, tingkat keyakinan (*confidence score*) model yang dilatih dengan data *train* teraugmentasi lebih tinggi dibandingkan model yang dilatih dengan dataset *original*.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan pengujian yang dilakukan pada penelitian tugas akhir ini, metode YOLOv5 dapat melakukan deteksi penggunaan masker pada wajah dengan memanfaatkan citra dari dataset *Face mask*

detection sebagai data latihan, validasi, dan uji. Selain itu, percobaan mengubah nilai IoU *threshold* pada model yang dilatih menggunakan dataset *original* dan dataset yang telah diaugmentasi pada data latihan juga memiliki dampak yang cukup signifikan. Model yang memiliki hasil terbaik yang menggunakan dataset *original* dengan IoU *threshold* sebesar 0,3 yang dengan nilai mAP pada *testing* untuk setiap kelas sebesar 0,876. Sedangkan model yang memiliki hasil terbaik yang menggunakan dataset yang diaugmentasi dengan IoU *threshold* sebesar 0,5 dengan nilai mAP pada *testing* untuk setiap kelas sebesar 0,849. Dengan adanya perbedaan nilai IoU *threshold* antar skenario model terbaik, menunjukkan bahwa variasi dan jumlah data berpengaruh terhadap nilai IoU *threshold* yang digunakan. Selain itu, model yang dilatih menggunakan dataset yang telah diaugmentasi memiliki kemampuan mendeteksi objek miring lebih baik dibandingkan model yang dilatih menggunakan dataset *original* dengan nilai IoU *threshold* yang sama. Model YOLOv3 yang dilatih dengan dataset yang diaugmentasi mendapatkan hasil terbaik pada saat penggunaan IoU *threshold* sebesar 0,3 dengan nilai mAP pada *testing* sebesar 0,857. Dengan adanya perbedaan nilai mAP antar IoU *threshold*, menunjukkan bahwa YOLOv5 lebih baik dari YOLOv3.

B. Saran

Saran yang dapat diberikan berdasarkan hasil percobaan pada Tugas Akhir ini adalah sebagai berikut:

1. Menambahkan dataset citra pada kelas `mask_worn_incorrect`
2. Menambahkan variasi citra dengan jarak lebih jauh
3. Melakukan augmentasi citra berupa zoom out

REFERENSI

- [1] G. J. Chowdary, N. S. Punna, S. K. Sonbhadra, and S. Agarwal, "Face Mask Detection using Transfer Learning of InceptionV3," Sep. 2020, doi: 10.1007/978-3-030-66665-1_6.
- [2] R. Chiragkumar Shah and R. Jignesh Shah, "Detection of Face Mask using Convolutional Neural Network." [Online]. Available: <https://doi.org/10.11>
- [3] Y. Liu, B. Lu, J. Peng, and Z. Zhang, "Research on the Use of YOLOv5 Object Detection Algorithm in Mask Wearing Recognition," *World Scientific Research Journal*, vol. 6, p. 2020, doi: 10.6911/WSRJ.202011_6(11).0038.
- [4] D. Giancini, E. Yulia Puspaningrum, Y. Vita Via, U. Pembangunan Nasional, and J. Timur, "Seminar Nasional Informatika Bela Negara (SANTIKA) Identifikasi Penggunaan Masker Menggunakan Algoritma CNN YOLOv3-Tiny".
- [5] Z. Cao, M. Shao, L. Xu, S. Mu, and H. Qu, "Maskhunter: Real-time object detection of face masks during the COVID-19 pandemic," *IET Image Processing*, vol. 14, no. 16, Dec. 2020, doi: 10.1049/iet-ipr.2020.1119.
- [6] K. Nithiyasree and T. Kavitha B A Student, "Face Mask Detection in Classroom using Deep Convolutional Neural Network," 2021.
- [7] J. Yao, J. Qi, J. Zhang, H. Shao, J. Yang, and X. Li, "A real-time detection algorithm for kiwifruit defects based on yolov5," *Electronics (Switzerland)*, vol. 10, no. 14, Jul. 2021, doi: 10.3390/electronics10141711.
- [8] J. Ieamsaard, S. N. Charoensook, and S. Yammen, "Deep Learning-based Face Mask Detection Using YoloV5," in *Proceeding of the 2021 9th International Electrical Engineering Congress, iEECON 2021*, Mar. 2021, pp. 428–431. doi: 10.1109/iEECON51072.2021.9440346.
- [9] F. Zhou, H. Zhao, and Z. Nie, "Safety Helmet Detection Based on YOLOv5," in *Proceedings of 2021 IEEE International Conference on Power Electronics, Computer Applications, ICPECA 2021*, Jan. 2021, pp. 6–11. doi: 10.1109/ICPECA51329.2021.9362711.
- [10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection." [Online]. Available: <http://pjreddie.com/yolo/>
- [11] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," Dec. 2016, [Online]. Available: <http://arxiv.org/abs/1612.08242>
- [12] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," Apr. 2018, [Online]. Available: <http://arxiv.org/abs/1804.02767>
- [13] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020, [Online]. Available: <http://arxiv.org/abs/2004.10934>
- [14] M. Harahap, L. Kusuma, M. Suryani, C. E. Situmeang, and J. F. Purba, "Identification of Face Mask With YOLOv4 Based on Outdoor Video," *Sinkron*, vol. 6, no. 1, pp. 127–134, Oct. 2021, doi: 10.33395/sinkron.v6i1.11190.
- [15] J. Yan, H. Wang, M. Yan, W. Diao, X. Sun, and H. Li, "IoU-adaptive deformable R-CNN: Make full use of IoU for multi-class object detection in remote sensing imagery," *Remote Sensing*, vol. 11, no. 3, Feb. 2019, doi: 10.3390/rs11030286.
- [16] G. Jocher.(2021, 1 April). Tips For Best Training Results. Diakses 1 Juni 2022 dari <https://github.com/ultralytics/yolov5/wiki/Tips-for-Best-Training-Results>
- [17] H. Kumar.(2019, 20 September). Evaluation metrics for object detection and segmentation: mAP. Diakses 26 Juli 2022 dari <https://kharshit.github.io/blog/2019/09/20/evalua>

tion-metrics-for-object-detection-and-segmentation

