

## Perbandingan Arsitektur YOLOv7 dan YOLOv8 untuk Mendeteksi Kerah Baju

Calvin Danyalson <sup>1</sup>, Margi Cahyanti <sup>2</sup>, Ericks Rachmat Swedia <sup>3</sup>, dan Mochammad Wisuda Sarjono <sup>4</sup>

<sup>1,3,4</sup>Informatika, Universitas Gunadarma

<sup>2</sup>Sistem Informasi, Universitas Gunadarma

<sup>1,2,3,4</sup>Jalan Margonda Raya No.100, Pondok Cina, Depok

E-mail: [5042030@student.gunadarma.ac.id](mailto:5042030@student.gunadarma.ac.id)<sup>1</sup>, [margi@staff.gunadarma.ac.id](mailto:margi@staff.gunadarma.ac.id)<sup>2</sup>, [ericks\\_rs@staff.gunadarma.ac.id](mailto:ericks_rs@staff.gunadarma.ac.id)<sup>3</sup>, [moch\\_wisuda@staff.gunadarma.ac.id](mailto:moch_wisuda@staff.gunadarma.ac.id)<sup>4</sup>

### ABSTRAK

Kerah baju merupakan salah satu aspek utama yang diawasi dalam pelaksanaan ujian *online* pada program studi pascasarjana di Universitas Gunadarma. Setiap peserta ujian diwajibkan mengenakan pakaian formal kemeja berkerah. Berdasarkan ketentuan tersebut, diperlukan penelitian yang bertujuan untuk mengembangkan metode deteksi kerah baju guna mempermudah proses pengawasan ujian *online* tersebut. Penelitian ini melibatkan analisis perbandingan antara dua arsitektur deteksi, yaitu dengan menggunakan *You Only Look Once* (YOLO) versi 7 (YOLOv7) dan versi 8 (YOLOv8), untuk menentukan arsitektur yang paling efektif dalam mendeteksi kerah baju sesuai dengan *dataset* yang digunakan dalam penelitian. Model deteksi yang dikembangkan dari kedua versi arsitektur ini diimplementasikan dalam aplikasi berbasis *website* dan diuji untuk mengevaluasi tingkat akurasi dan efisiensinya. Berdasarkan pengujian terhadap model yang dihasilkan, YOLOv7 menunjukkan akurasi rata-rata sebesar 95%, yang lebih unggul dibandingkan dengan YOLOv8 yang memiliki akurasi rata-rata sebesar 75%. Namun, meskipun akurasi YOLOv8 lebih rendah dibandingkan YOLOv7, YOLOv8 memiliki keunggulan dalam hal kecepatan deteksi, dengan waktu rata-rata 2,27 detik, yang jauh lebih cepat daripada YOLOv7 yang memerlukan waktu rata-rata 22,42 detik. Jika dibandingkan dari segi akurasi dan kecepatan, YOLOv7 masih menunjukkan performa terbaik dalam penelitian ini. Namun, tidak menutup kemungkinan bagi YOLOv8 untuk dapat mengungguli YOLOv7 di masa mendatang, terutama jika akurasi deteksi YOLOv8 dapat ditingkatkan secara signifikan.

**Kata Kunci:** Baju, Deteksi, Kemeja, Kerah, Ujian, Yolo.

## Comparison of YOLOv7 and YOLOv8 Architectures for Detecting Shirt Collars

### ABSTRACT

*The shirt collar is one of the primary aspects monitored during online examinations in the postgraduate program at Gunadarma University. Examinees are required to wear formal, collared attire. Based on these regulations, a study was conducted to develop a collar detection method to facilitate the online exam monitoring process. This research involves a comparative analysis of two detection architectures: You Only Look Once (YOLO) version 7 (YOLOv7) and version 8 (YOLOv8), to determine the most effective architecture for detecting shirt collars using the dataset provided in the study. Detection models developed from both architectures were implemented in a web-based application and tested to evaluate their accuracy and efficiency. The testing results showed that YOLOv7 achieved an average accuracy of 95%, outperforming YOLOv8, which had an average accuracy of 75%. However, despite YOLOv8's lower accuracy, it excelled in detection speed, with an average processing time of 2.27 seconds, significantly faster than YOLOv7's average processing time of 22.42 seconds. Considering both accuracy and speed, YOLOv7 demonstrated the best overall performance in this study. Nonetheless, it is possible that YOLOv8 could surpass YOLOv7 in the future if significant improvements are made to its detection accuracy.*

**Keywords:** Collar, Detection, Exam, Shirt, YOLO

### 1. PENDAHULUAN

Pemrosesan citra digital sudah mulai memiliki popularitas yang besar pada berbagai bidang seperti *fotogrametri*, *remote sensing*, dan *computer vision* (Atik

et al., 2022). *Object detection* merupakan sebuah teknik yang akan mendeteksi objek seperti buah, baju, dan lainnya. Saat ini tersedia banyak arsitektur yang dapat digunakan untuk dapat membuat sebuah mesin *object*

*detection* seperti menggunakan arsitektur *You Only Look Once* atau biasa dikenal dengan istilah YOLO (Fauzan & Wibowo, 2021). YOLO yang menggunakan pendekatan satu tahap proses deteksi memiliki kemampuan untuk dapat memproduksi hasil deteksi objek yang lebih cepat dibandingkan dengan deteksi objek lainnya yang menggunakan dua tahap proses deteksi (Putro et al., 2020). Arsitektur YOLO ini dapat diimplementasikan untuk mendeteksi dari objek kendaraan mobil dan motor memperoleh besar akurasi sebesar 86% dalam melakukan deteksinya (Hayati et al., 2023). Mendeteksi waktu panen dari tanaman pakcoy menghasilkan nilai prediksi terbesar dengan nilai 98% (Ibrahim & Latifa, 2023). Mendeteksi objek plat nomor kendaraan berhasil memperoleh akurasi sebesar 92.38% (Illmawati & others, 2023). Mendeteksi objek kapal lain Batam sebesar 93% pada kapal Ferry, 85% pada Kapal Nelayan Indonesia, 89% pada Kapal Nelayan Thailand, 91% pada Kapal Nelayan Vietnam, 94% pada speedboat, dan 83% pada Tanker (Marsil & Caniango, 2024). Mendeteksi korban bencana alam memperoleh nilai precision sebesar 92.9% (Sarosa & Muna, 2021). Melakukan deteksi dan klasifikasi kecatatan pada produk masker mampu memiliki nilai akurasi saat pengujian sebesar 97.1% (Arvio et al., 2024). Melakukan deteksi riasan yang ada pada adat nusantara mampu memiliki nilai rata-rata akurasi sebesar 95.20% dan rata-rata waktu yang dibutuhkan model dalam melakukan deteksi adalah 327ms (Jannah & Sutanto, 2022). Mendeteksi pada makanan khas Indonesia memiliki akurasi sebesar 98.6% (Utami et al., 2023).

Mendeteksi kerah baju merupakan salah satu faktor yang perlu diperhatikan dalam pelaksanaan ujian *online* pada program studi pasca sarjana Universitas Gunadarma yang dilaksanakan di halaman website <https://ujian.pasca.gunadarma.ac.id/>. Selama proses ujian, peserta akan diawasi oleh pengawas secara langsung. Melalui adanya kebutuhan hal tersebut, maka perlu dilakukan penelitian untuk mendeteksi kerah baju sehingga memudahkan proses pengawasan ujian.

Melalui penelitian yang dilakukan, akan dibangun dua buah model dari arsitektur YOLOv7 dan YOLOv8. Kedua model ini akan dilakukan *training* untuk dapat melakukan deteksi pada objek pada bagian leher baju dan melakukan klasifikasi apakah baju tersebut memiliki kerah (baju berkerah) atau tidak (kaus). Fokus dari penelitian ini adalah untuk menentukan pilihan model terbaik dalam melakukan deteksi kerah baju berdasarkan *dataset* yang telah dikumpulkan kemudian mengimplementasikan menggunakan kedua model ini sebagai alat untuk dapat mendeteksi baju berkerah pada ujian pasca sarjana di Universitas Gunadarma.

## 2. RUANG LINGKUP

Melalui penelitian yang telah dilakukan, berikut beberapa batasan yang dapat diuraikan menjadi beberapa hal sebagai berikut.

1. Penelitian melibatkan total 2572 gambar yang diperoleh dari internet melalui pencarian *Google* yang

sebagian besar diambil melalui *e-commerce* seperti Zalora dan Matahari

2. Penelitian akan menggunakan arsitektur YOLOv7 dan YOLOv8 sebagai arsitektur deteksi objek yang digunakan untuk pengujian perbandingan.
3. Pengujian perbandingan akan menggunakan perangkat CPU.
4. Sebagai simulasi *webcam* dalam sebuah ujian, proses pengujian akan menggunakan gambar dengan ukuran rasio 19:6 yang berisi seseorang yang menggunakan baju berkerah sebanyak 15 gambar dan seseorang yang menggunakan baju tidak berkerah sebanyak 15 gambar.
5. Pengujian perbandingan akan dilakukan pada nilai akurasi yang dihasilkan dan waktu yang dibutuhkan oleh model.
6. Implementasi model yang telah dihasilkan dari pelatihan YOLOv7 dan YOLOv8 akan diimplementasikan dalam bentuk aplikasi berbasis *website* yang terdiri dari *Backend API* menggunakan bahasa pemrograman Python dan *Frontend App* menggunakan bahasa pemrograman Javascript.

Dalam penelitian ini juga terdapat tujuan yang mencakup :

1. Menganalisis model pada akurasi dan waktu saat pelatihan dan saat melakukan deteksi objek kerah baju.
2. Mencari model terbaik dari kedua arsitektur.
3. Melakukan implementasi kedua model kedalam bentuk aplikasi berbasis *website*.

## 3. BAHAN DAN METODE

Pada bagian ini menjelaskan bahan-bahan dan metode apa yang akan digunakan dalam penelitian ini, diantaranya :

### 3.1 Sistem *You Only Look Once* (YOLO)

YOLO merupakan sebuah algoritma yang dapat digunakan untuk mendeteksi objek dengan memanfaatkan metode *Single Shot Detector* (SSD) untuk dapat mendeteksi objek dalam satu kali proses konvolusional. Selain menerapkan SSD, YOLO juga menggunakan teknologi OCR yang dimana merupakan suatu metode untuk dapat memperoleh suatu citra digital dan menjadikannya sebagai karakter sehingga dapat di olah oleh komputasi digital (Illmawati & others, 2023).

### 3.2 YOLOv7

YOLOv7 merupakan versi ke-7 dari algoritma YOLO yang memiliki beberapa pengembangan dari beberapa iterasi versi sebelumnya (Maulida & others, 2023). Salah satu pengembangan yang paling mencolok dalam algoritma ini adalah pengembangan pada penggunaan *anchor box* ketika melakukan deteksi objek. Melalui pengembangan dari *anchor box*, memungkinkan model untuk dapat mengenail objek dalam berbagai bentuk dan ukuran (Hasan et al., 2023).

### 3.3 YOLOv8

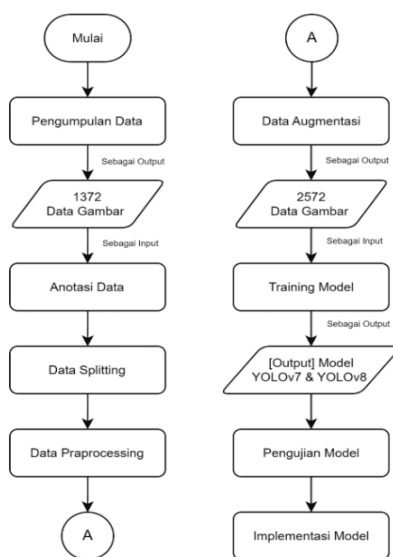
YOLOv8 merupakan algoritma deteksi objek yang dikembangkan oleh Ultralytics yang memiliki beberapa keunggulan seperti proses *training* yang lebih mudah dengan API baru dan dapat berjalan lebih cepat pada CPU maupun GPU (Hasan et al., 2023). Dalam perkembangannya, YOLO telah mendapat perhatian dari komunitas visi komputer dan beberapa versi sebelumnya dipelihara dalam bahasa C menggunakan *Darknet*. Pengembangan YOLOv8 dimulai ketika penulisnya, Glenn Jocher, mengikuti YOLOv3 di PyTorch dan akhirnya meluncurkan YOLOv5 yang menjadi “*State-of-the-Art*” berkat struktur Python yang fleksibel. YOLOv8 merupakan hasil penelitian dan pengembangan selama enam bulan terakhir dan resmi diluncurkan pada Januari 2023 (Satya et al., 2023).

### 3.4 API dengan Bahasa Pemrograman Python

*Application Programming Interface* (API) sendiri adalah program yang menjembatani dua aplikasi untuk saling berkomunikasi (*interface*). API memudahkan koneksi untuk beberapa sistem yang memiliki arsitektur berbeda. Aplikasi ini biasanya berjalan di web server, yang berfungsi untuk memberikan layanan terhadap permintaan dari client (Swedia et al., 2022).

### 3.5 Gambaran Umum

Melalui penelitian ini, akan dilakukan pembuatan sebuah model dengan memanfaatkan arsitektur dari YOLOv8 dan YOLOv7 dengan harapan dapat melakukan deteksi pada kerah pada baju. Proses perbandingan antara hasil deteksi kedua arsitektur akan dilakukan untuk mengetahui perbedaan performa akurasi kecepatan dalam melakukan deteksi objek. Penelitian akan dilakukan dalam beberapa tahap yang dijelaskan secara visual pada Gambar 1.



**Gambar 1. Flowchart Alur Penelitian**  
*Figure 1. Research Flow Flowchart*

### 3.6 Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data dalam bentuk gambar yang dikumpulkan melalui *platform search engine* seperti *google.com*, *bing.com*, dan lainnya. Data yang dikumpulkan merupakan data manusia yang menggunakan pakaian yang memiliki kerah dan tidak memiliki kerah. Kategori kelas pada *dataset* yang dibangun akan memiliki dua kelas yaitu baju berkerah dan baju tidak berkerah. Gambar 2 merupakan contoh dari data yang diperoleh dari internet.



**Gambar 2. Data Baju Berkerah dan Tidak Berkerah**  
*Figure 2. Collared and Collarless Shirt Data*

Melalui proses pengumpulan data, berhasil diperoleh sejumlah 1372 total gambar dengan pembagian baju berkerah dan baju tidak berkerah masing-masing sejumlah 686. Proses pengumpulan dan organisasi data memanfaatkan alat bantu *software* yang bernama Roboflow. Melalui *dataset* ini, akan dilakukan pembangunan model dari YOLOv7 dan YOLOv8 untuk menjadi bahan utama dalam penelitian.

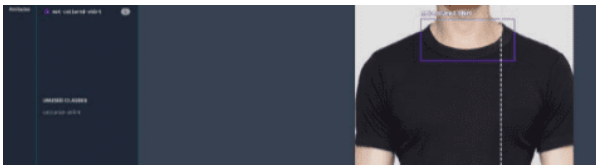
### 3.7 Anotasi Data

Proses anotasi gambar dilakukan pada 1372 total gambar yang telah dikumpulkan dengan menggunakan alat bantu *software* bernama Roboflow. Pada proses ini akan memberikan anotasi pada objek kerah baju pada setiap gambar dengan menyesuaikan kelasnya. Kelas yang dimaksud dalam anotasi ini adalah kelas “baju berkerah” dan “baju tidak berkerah”. Gambar 3 merupakan proses anotasi untuk gambar dengan manusia menggunakan baju berkerah.



**Gambar 3. Proses Anotasi untuk Gambar Manusia dengan Baju Berkerah**  
*Figure 3. Annotation Process for Human Images with Collars Shirts*

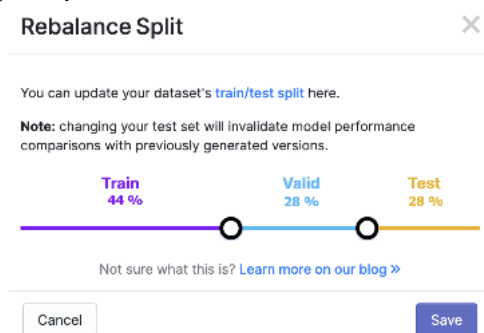
Selain dari proses anotasi untuk data dengan kelas “baju berkerah”, dilakukan juga proses anotasi untuk kelas “baju tidak berkerah”. Gambar 4 merupakan proses anotasi untuk gambar manusia yang menggunakan baju tidak berkerah.



**Gambar 4. Proses Anotasi untuk Gambar Manusia dengan Baju Tidak Berkerah**  
*Figure 4. Annotation Process for Human Images with Collarless Shirts*

### 3.8 Data Splitting

Proses ini merupakan perlakuan kepada *dataset* yang sudah dikumpulkan dengan membagi porsi dari data yang digunakan untuk data *train*, *valid*, dan *test*. Proses ini dilakukan dengan menggunakan bantuan tools pada Roboflow di menu *Rebalance Splits* sebagaimana yang ditunjukkan pada Gambar 5.



**Gambar 5. Proses Pengaturan Porsi Data Gambar dengan Rebalance Splits**

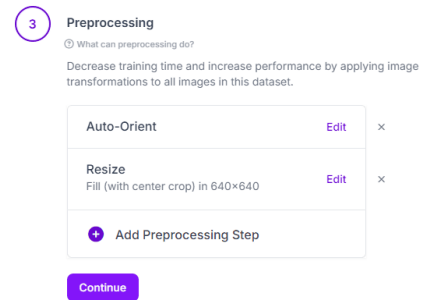
*Figure 5. Image Data Portion Arrangement Process with Rebalance Splits*

Melalui Gambar 5 dapat diketahui pemecahan yang dilakukan adalah dengan perbandingan 44:28:28 dari total 100% data. Selain menggunakan bantuan *menu Rebalance Splits*, dilakukan juga pemindahan data secara manual pada data baju berkerah dan baju tidak berkerah dari kelompok *train* ke *valid*, kelompok *valid* ke *test*, dan kelompok *test* ke *train*. Hal ini bertujuan untuk memastikan porsi data baju berkerah dan baju tidak berkerah sama dengan satu sama lain dalam suatu kelas. Setelah proses ini, data pada *train* akan memiliki total 600 data gambar dengan 300 data gambar dari baju berkerah, dan 300 data gambar untuk baju tidak berkerah, kemudian dengan total 386 data gambar untuk kelompok *valid* dan *test* dengan pembagian masing masing kelasnya berjumlah 193 data gambar.

### 3.9 Data Preprocessing

Pada proses data *preprocessing* akan dilakukan persiapan data gambar sebelum digunakan untuk proses *training*, dan lainnya. Pada penelitian ini, *dataset* gambar yang telah dikumpulkan akan dilakukan operasi *resize* menjadi ukuran 640 x 640 piksel dan operasi *auto orient* yang bertujuan untuk memastikan orientasi yang dimiliki setiap gambar memiliki orientasi yang konsisten dan

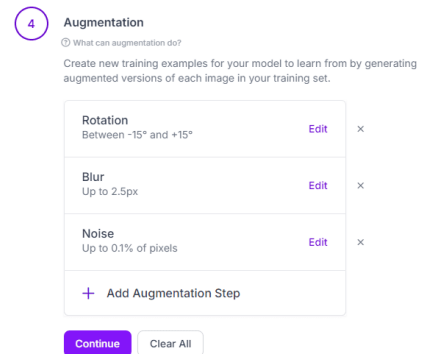
sesuai. Gambar 6 merupakan proses kegiatan yang dilakukan dalam proses data *preprocessing*.



**Gambar 6. Proses Melakukan Data Preprocessing**  
*Figure 6. Data Preprocessing Process*

### 3.10 Data Augmentasi

Pada proses ini dilakukan augmentasi pada data gambar yang sebelumnya sudah disediakan. Proses ini akan membuat sebuah gambar duplikasi baru namun dengan versi gambar yang sudah di augmentasi. Gambar 7 merupakan proses melakukan data augmentasi pada Roboflow.



**Gambar 7. Proses Operasi Augmentasi pada Dataset Gambar**

*Figure 7. Augmentation Operation Process on Image Dataset*

Berdasarkan Gambar 7, beberapa teknik augmentasi yang dilakukan adalah *rotation* atau memutar objek dengan derajat putaran sebesar 15 derajat, *blur* objek dengan tingkat keburaman hingga 2.3 pixel, penambahan *noise* pada objek. Gambar 8 merupakan contoh gambar dari hasil augmentasi *rotation*, *blur*, dan *noise*.



**Gambar 8. Contoh Gambar Hasil Augmentasi**  
*Figure 8. Example of Augmentation Result Image*

## 4. PEMBAHASAN

Pada bagian ini menjelaskan bahan-bahan dan metode apa yang akan digunakan dalam penelitian ini, diantaranya :



#### 4.1 Training Model Arsitektur YOLOv7

Training untuk model YOLOv7 ini tidak menggunakan *early stopping* namun akan mengikuti jumlah *epoch* yang dicapai oleh training YOLOv8 yang menggunakan *early stopping*. Sehingga jumlah *epochs* yang digunakan pada training YOLOv7 ini berjumlah 51 *epochs*. Gambar 9 merupakan gambar dari logs yang dihasilkan dari training YOLOv7.

Epoch	gpu_mem	box	obj	cls	total	Labels	img_size
46/50	12.7%	0.02137	0.002717	0.000253	0.02432	12	640: 100% 113/113 [01:37:00-00, 1.161t/s]
Class Images Labels P R mAP <sub>0.5</sub> mAP <sub>0.5:0.95</sub> 100% 13/13 [00:00:00-00, 1.961t/s]							
all 386 405 0.998 0.998 0.995 0.734							

Gambar 9. Epocs Terakhir pada Training YOLOv7  
Figure 9. Last Epocs on YOLOv7 Training

#### 4.2 Training Model Arsitektur YOLOv8

Training untuk model YOLOv8 ini memanfaatkan fitur *early stopping* yang ditandai pada penggunaan parameter *patience*. Sehingga jika nilai *loss* tidak mengalami perubahan dalam 10 *epochs*, maka training akan dihentikan dan model dapat langsung digunakan. Melalui proses yang dilakukan, model YOLOv8 berhasil di *train* dan berhenti pada *epochs* 51/100. Gambar 10 merupakan gambar dari logs yang dihasilkan dari training YOLOv8.

Epoch	gpu_mem	box_loss	cls_loss	obj_loss	Instances	Size
46/200	4.14%	0.0005	0.0001	0.0001	0	640: 100% 113/113 [00:42:00-00, 2.081t/s]
Class Images Instances box? cls? obj? 100% 13/13 [00:00:00-00, 2.101t/s]						
all 386 405 0.999 0.999 0.995 0.732						

Gambar 10. Epocs Terakhir pada Training YOLOv8  
Figure 10. Last Epocs on YOLOv8 Training

#### 4.3 Pengujian Model

Pengujian model merupakan hal yang diperlukan untuk mengetahui performa dari sebuah model yang sudah di *train*. Pengujian ini dilakukan dengan data pada kelompok *valid* dengan total gambar sebanyak 386 gambar yang masing-masing kelas nya terdiri dari 193 gambar. Gambar 11 merupakan proses pengujian yang dilakukan dari model yang sudah di *train*

Class	Images	Instances	box?P	cls?P	obj?P	mAP <sub>0.5</sub>	mAP <sub>0.5:0.95</sub>
collared-shirt	386	405	0.99	0.986	0.995	0.752	0.752
not-collared-shirt	386	312	0.99	0.979	0.995	0.730	0.730
all	386	405	0.991	0.984	0.994	0.735	0.735

Gambar 11. Proses Pengujian dan Validasi  
Figure 11. Model Testing and Validation

Berdasarkan Gambar 11, dapat diketahui bahwa model berhasil melakukan proses deteksi objek dengan metrik rata-rata pada *precision* sebesar 0.99, *recall* sebesar 0.986, mAP50 sebesar 0.995, dan mAP50-90 sebesar 0.752.

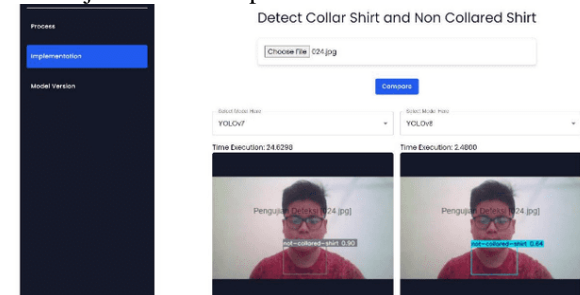
Class	Images	Labels	P	R	mAP <sub>0.5</sub>	mAP <sub>0.5:0.95</sub>
collared-shirt	386	405	0.995	0.995	0.995	0.734
not-collared-shirt	386	312	0.995	0.995	0.995	0.734
all	386	405	0.999	0.995	0.995	0.737

Gambar 12. Proses Pengujian dan Validasi Model  
Figure 12. Model Testing and Validation Process

Berdasarkan Gambar 12, dapat diketahui bahwa model YOLOv7 berhasil melakukan proses deteksi objek dengan metrik rata-rata pada *precision* sebesar 0.999, *recall* sebesar 0.995, mAP50 sebesar 0.995, dan mAP50-90 sebesar 0.737.

#### 4.4 Implementasi

Pada tahapan ini menampilkan hasil dari penelitian yang diintegrasikan dengan API. Pada gambar 13 menunjukkan hasil implementasi.

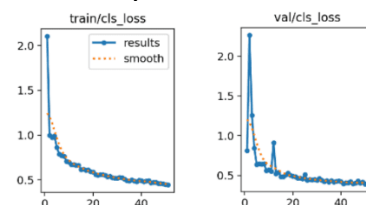


Gambar 13. Tampilan Implementasi  
Figure 13. Application Interface

#### 4.5 Hasil

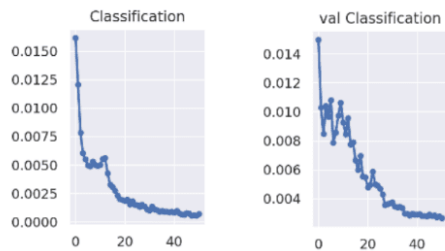
Melalui proses pelatihan yang dilakukan berhasil diperoleh sebuah model dari masing-masing kedua arsitektur yang akan diuji perbandingannya yaitu model dari arsitektur YOLOv8 dan YOLOv7.

Gambar 14 merupakan grafik *metric class loss* yang diperoleh berdasarkan pelatihan YOLOv8.



Gambar 14. Metric Loss Pelatihan Model YOLOv8  
Figure 14. YOLOv8 Model Training Loss Metric

Berdasarkan Gambar 14, dapat diketahui pada grafik *class loss* pada model YOLOv8 selama proses *training* maupun *validation* mengalami menurun seiring bertambahnya *epochs*. Hal ini menunjukkan bahwa model yang di *train*, mengalami peningkatan dalam hal mengklasifikasikan objek dengan benar. Melalui jumlah *epochs* yang diperoleh dari pelatihan YOLOv8 dengan menggunakan *early stopping*, model YOLOv7 juga di *train* dengan jumlah *epochs* yang sejumlah dengan pelatihan YOLOv8. Gambar 15 merupakan grafik *metric* yang diperoleh berdasarkan pelatihan YOLOv7.



**Gambar 15. Metric Loss Pelatihan Model YOLOv7**

*Figure 15. YOLOv7 Model Training Loss Metric*

Berdasarkan Gambar 15, dapat diketahui juga bahwa sama halnya dengan *training* YOLOv8, model YOLOv7 juga memiliki grafik *classification loss* yang menurun seiring berjalannya *epochs*. Hal ini menunjukkan bahwa model yang dilatih semakin dalam melakukan klasifikasi objek dengan benar selama pertambahan *epoch*.

Tabel 1 merupakan hasil dari pengujian untuk kelas *collared-shirt* atau baju berkerah yang telah dilakukan.

**Tabel 1. Pengujian Deteksi Kelas *collared-shirt* (Baju Berkerah)**

*Table 1. Collared-shirt Class Detection Testing*

Gambar Hasil Deteksi	Kelas	Waktu (Detik)	Status
	<b>YoloV7</b> Berkerah	22.99	Benar
	<b>YoloV8</b> Berkerah	4.04	Benar
	<b>YoloV7</b> Berkerah	23.26	Benar
	<b>YoloV8</b> Tidak Terdeteksi	2.23	Salah
	<b>YoloV7</b> Berkerah	22.51	Benar
	<b>YoloV8</b> Berkerah	2.39	Benar
	<b>YoloV7</b> Berkerah	24.68	Benar
	<b>YoloV8</b> Tidak Terdeteksi	3.91	Salah
	<b>YoloV7</b> Berkerah	23.99	Benar
	<b>YoloV8</b> Tidak Terdeteksi	2.32	Salah
	<b>YoloV7</b> Berkerah	22.07	Benar
	<b>YoloV8</b> Berkerah	2.14	Benar

Gambar Hasil Deteksi	Kelas	Waktu (Detik)	Status
	<b>YoloV7</b> Berkerah	21.14	Benar
	<b>YoloV8</b> Berkerah	2.22	Benar
	<b>YoloV7</b> Tidak Terdeteksi	20.18	Salah
	<b>YoloV8</b> Tidak Terdeteksi	2.35	Salah
	<b>YoloV7</b> Berkerah	23.05	Benar
	<b>YoloV8</b> Berkerah	2.31	Benar
	<b>YoloV7</b> Berkerah	20.40	Benar
	<b>YoloV8</b> Berkerah	2.22	Benar
<b>YoloV7</b> Hasil dan Rata-Rata Waktu		9 : 1	Waktu : 22.42
<b>YoloV8</b> 6 : 4			Waktu : 2.61

Berdasarkan hasil pengujian dapat diketahui juga bahwa YOLOv7 membutuhkan waktu rata-rata selama pengujian sebesar 22.42 detik, sedangkan YOLOv8 membutuhkan waktu 2.61 detik. Melalui pengujian yang telah dilakukan dengan *dataset* yang sudah digunakan dapat disimpulkan bahwa model YOLOv7 membutuhkan waktu yang lama dibandingkan dengan YOLOv8 namun mampu memperoleh deteksi objek kerah baju dengan baik sedangkan YOLOv8 mampu melakukan deteksi dengan cepat dibandingkan YOLOv7 namun dalam beberapa kasus kondisi gambar, YOLOv8 tidak dapat melakukan deteksi pada objek kelas *collared-shirt*.

Tabel 2 merupakan hasil pengujian untuk kelas *not-collared-shirt* atau baju tidak berkerah yang telah dilakukan.

**Tabel 2. Pengujian Deteksi Kelas *not-collared-shirt* (Baju Tidak Berkerah)**

*Table 2. Not Collared Shirt Class Detection Testing*

Gambar Hasil Deteksi	Kelas	Waktu (Detik)	Status
	<b>YoloV7</b> Tidak Berkerah	22.38	Benar
	<b>YoloV8</b> Tidak Berkerah	2.31	Benar
	<b>YoloV7</b> Tidak Berkerah	24.95	Benar
	<b>YoloV8</b> Tidak Berkerah	2.40	Benar
	<b>YoloV7</b> Tidak Berkerah	21.78	Benar
	<b>YoloV8</b> Tidak Berkerah	2.17	Benar
	<b>YoloV7</b> Tidak Berkerah	20.66	Benar
	<b>YoloV8</b> Tidak Berkerah	2.28	Benar
	<b>YoloV7</b> Tidak Berkerah	19.98	Benar

Gambar Hasil Deteksi	Kelas	Waktu (Detik)	Status
	<b>YoloV8</b> Tidak Berkerah	2.12	Benar
	<b>YoloV7</b> Tidak Berkerah	20.85	Benar
	<b>YoloV8</b> Tidak Berkerah	2.28	Benar
	<b>YoloV7</b> Tidak Berkerah	22.06	Benar
	<b>YoloV8</b> Tidak Berkerah	2.30	Benar
	<b>YoloV7</b> Tidak Berkerah	25.41	Benar
	<b>YoloV8</b> Tidak Berkerah	2.3	Benar
	<b>YoloV7</b> Tidak Berkerah	23.91	Benar
	<b>YoloV8</b> Tidak Berkerah	2.37	Benar
	<b>YoloV7</b> Tidak Berkerah	21.62	Benar
	<b>YoloV8</b> Tidak Berkerah	2.26	Salah
Hasil dan Rata-Rata Waktu	<b>YoloV7</b> 10 : 0	Waktu : 22.37	
	<b>YoloV8</b> 9 : 1	Waktu : 2.27	

Melalui pengujian yang telah dilakukan, dalam deteksi dan klasifikasi kelas baju tidak berkerah (baju kaus), model YOLOv7 mampu melakukan 10 deteksi benar dan tidak memiliki deteksi yang salah, sedangkan model YOLOv8 mampu melakukan deteksi sejumlah 10 deteksi benar. Selanjutnya, berdasarkan hasil pengujian dapat diketahui juga bahwa YOLOv7 membutuhkan waktu rata-rata selama pengujian sebesar 22.37 detik, sedangkan model YOLOv8 mampu melakukan deteksi sejumlah 9 deteksi benar dan 1 gambar tidak terdeteksi. Melalui pengujian yang telah dilakukan dengan *dataset* yang sudah digunakan untuk kelas *not-collared-shirt* dapat disimpulkan bahwa model YOLOv7 membutuhkan waktu yang lama dibandingkan dengan YOLOv8 namun mampu memperoleh deteksi objek bagian leher baju kaus dengan baik, YOLOv8 mampu melakukan deteksi dengan cepat dibandingkan YOLOv7 namun dalam beberapa kasus kondisi gambar YOLOv8 tidak dapat melakukan deteksi pada objek kelas *not-collared-shirt* dengan baik. Tabel 3 menampilkan hasil dari perbandingan untuk YOLOv7 dan YOLOv8.

**Tabel 3. Perbandingan Akurasi Model YOLOv7 dan YOLOv8**

*Table 3. Comparison of YOLOv7 and YOLOv8 Model Accuracy*

Arsitektur	Baju Berkerah	Akurasi	Baju Tidak Berkerah	Akurasi
YoloV7	Benar : 9	90%	Benar : 10	100%
YoloV8	Benar : 6	60%	Benar : 9	90%

## 5. KESIMPULAN

Dalam penelitian ini model YOLOv7 dan model YOLOv8 mampu memahami objek dengan baik seiring berjalannya *epochs* pelatihan dengan YOLOv7 membutuhkan waktu yang lebih lama dalam pelatihnnya dibandingkan dengan YOLOv8. Kemudian, berdasarkan pengujian yang telah dilakukan, diperoleh juga informasi bahwa dengan menggunakan *dataset* yang dikumpulkan dalam penelitian ini, YOLOv7 mampu mengungguli YOLOv8 dalam akurasi deteksi dengan nilai akurasi rata-rata sebesar 90% untuk baju berkerah dan 100% untuk baju tidak berkerah, sedangkan YOLOv8 memperoleh sebesar 60% untuk baju berkerah dan 90% untuk baju tidak berkerah. Selain itu, jika diperhatikan pada waktu deteksi, waktu yang dibutuhkan oleh YOLOv7 lebih lama dibandingkan YOLOv8 selama melakukan deteksi dengan nilai rata-rata waktu deteksi pada YOLOv7 yaitu 22.42 detik untuk baju berkerah dan 22.37 untuk baju tidak berkerah, sedangkan YOLOv8 sebesar 2.61 detik untuk baju berkerah dan 2.27 untuk baju tidak berkerah. Selain itu, kedua model yaitu model dari YOLOv7 dan YOLOv8 yang juga berhasil diimplementasikan kedalam aplikasi berbasis website dengan tujuan untuk melakukan pengujian setelah diintegrasika kedalam aplikasi website.

## 6. SARAN

Pengembangan yang dapat dilakukan untuk penelitian lebih lanjut yaitu penambahan jumlah *dataset* dan generalisasi data. Melalui pengembangan ini dapat memungkinkan kedua model dapat mampu memahami lebih baik pada objek yang perlu dilakukan deteksi dalam keadaan gambar seperti apapun terutama pada model YOLOv8. Hal ini dikarenakan waktu yang dibutuhkan YOLOv8 sangat signifikan mengungguli YOLOv7 dalam penelitian ini, sehingga jika YOLOv8 mampu memiliki akurasi yang lebih tinggi, maka model ini dapat mengungguli model YOLOv7.

## 7. REFERENSI

- Arvio, Y., Kusuma, D. T., & Sangadji, I. B. M. (2024). PENDEKATAN ALGORITMA YOLO V5 UNTUK MENDETEKSI CACAT PRODUK MASKER. *Jurnal Ilmiah Dinamika Rekayasa*, 20(1), 11–17.
- Atik, M. E., Duran, Z., & Özgünlük, R. (2022). Comparison of YOLO versions for object detection from aerial images. *International Journal of Environment and Geoinformatics*, 9(2), 87–93.
- Fauzan, M. R., & Wibowo, A. P. W. (2021). Pendeteksian Plat Nomor Kendaraan Menggunakan Algoritma You Only Look Once V3 Dan Tesseract. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 8(1), 57–62.
- Hasan, R. H., Hassoo, R. M., & Aboud, I. S. (2023). *Yolo Versions Architecture*.
- Hayati, N. J., Singasatia, D., & Muttaqin, M. R. (2023). Object Tracking Menggunakan Algoritma You Only Look Once (Yolo) V8 Untuk Menghitung

- Kendaraan. *Komputa: Jurnal Ilmiah Komputer Dan Informatika*, 12(2), 91–99.
- Ibrahim, M., & Latifa, U. (2023). PENERAPAN ALGORITMA YOLOV8 DALAM DETEKSI WAKTU PANEN TANAMAN PAKCOY BERBASIS WEBSITE. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(4), 2489–2495.
- Illmawati, R., & others. (2023). YOLO v5 untuk Deteksi Nomor Kendaraan di DKI Jakarta. *Jurnal Ilmu Komputer Dan Agri-Informatika*, 10(1).
- Jannah, Z. S., & Sutanto, F. A. (2022). Implementasi Algoritma YOLO (You Only Look Once) Untuk Deteksi Rias Adat Nusantara. *Jurnal Ilmiah Universitas Batanghari Jambi*, 22(3), 1490–1495.
- Marsil, M. A., & Caniago, D. P. (2024). Sistem Pencegahan Illegal Fishing di Laut Batam menggunakan YOLOv7 berbasis Notifikasi Telegram. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 12(1), 175.
- Maulida, S., & others. (2023). Analisis Akurasi Pada Simbol Abjad Sistem Isyarat Bahasa Indonesia (SIBI) Menggunakan Metode CNN dan YOLO (You Only Look Once). UIN Ar-Raniry Banda Aceh.
- Putro, E. C., Awangga, R. M., & Andarsyah, R. (2020). *Tutorial Object Detection People With Faster region-Based Convolutional Neural Network (Faster R-CNN)* (Vol. 1). Kreatif.
- Sarosa, M., & Muna, N. (2021). Implementasi algoritma you only look once (YOLO) untuk deteksi korban bencana alam. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 8(4), 787–792.
- Satya, L., Septian, M. R. D., Sarjono, M. W., Cahyanti, M., & Swedia, E. R. (2023). SISTEM PENDETEKSI PLAT NOMOR POLISI KENDARAAN DENGAN ARSITEKTUR YOLOV8. *Sebatik*, 27(2), 753–761.
- Swedia, E. R., Fitriani, R. R., Cahyanti, M., & Septian, M. R. D. (2022). *Feed Forward Neural Network untuk Prediksi Data: Implementasi dengan Pyhton dan Flask API pada Sistem Operasi Windows*. <https://books.google.co.id/books?id=NY2KEAAAQBAJ>
- Utami, G. C., Widiawati, C. R., & Subarkah, P. (2023). Detection of Indonesian Food to Estimate Nutritional Information Using YOLOv5. *Teknika*, 12(2), 158–165.