



Implementasi YOLOv8 Pada Sistem Smart Mirror Untuk Deteksi dan Pengingat Barang Pribadi

Tugas Akhir

**Oleh:
Anjunius M (4212431015)**

**Program Studi Teknik Mekatronika
Jurusan Teknik Elektro
Politeknik Negeri Batam
2022**

Pernyataan Keaslian Tugas Akhir

Saya yang bertandatangan dibawah ini menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya yang berjudul : "Implementasi Yolov8 Pada Sistem Smart Mirror Untuk Deteksi dan Peningat Barang Pribadi" adalah hasil karya sendiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan, dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri. Semua referensi yang dikutip atau dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan saya ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Batam, Juli 2026



Anjunius M
NIM: 4212431015

Lembar Pengesahan

Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Terapan Teknik (S.Tr.T)
di
Politeknik Negeri Batam

Oleh:
Anjunius M (4212431015)

Tanggal Sidang: 06 Juli 2026

Disetujui oleh :



1. Diono, S.Tr., M.Sc
NIK:120243



1. Daniel Sutopo Pamungkas, S.T.,
M.T., Ph.D
NIK:100006



2. Dr. Abdurahman
Dwijotomo, S.ST., M.Sc
NIK:122257

Implementasi YOLOv8 Pada Sistem Smart Mirror Untuk Deteksi dan Pengingat Barang Pribadi

Abstrak

Penelitian ini bertujuan mengembangkan sistem *smart mirror* yang mampu mendeteksi dan mengingatkan pengguna terhadap barang-barang pribadi seperti masker, kacamata dan kartu identitas kerja menggunakan metode *object detection* dengan algoritma YOLOv8 (*You Only Look Once* versi 8). Sistem dibangun menggunakan dataset yang dikumpulkan dan dianotasi secara mandiri, kemudian dilatih melalui platform Google Colab, dengan pembagian data 80% *training* dan 20% *validation*. Kamera terintegrasi pada cermin akan menangkap citra secara *real-time*, dianalisis oleh model YOLOv8, dan sistem akan memberikan notifikasi suara secara otomatis apabila salah satu barang tidak terdeteksi. Hasil pengujian menunjukkan model mencapai tingkat presisi (*precision*) sebesar 78,60% secara keseluruhan, dengan rincian per kelas objek yaitu masker sebesar 97,65%, kacamata sebesar 80,26%, dan kartu identitas sebesar 74,21%. Presisi terendah pada kartu identitas disebabkan oleh permukaannya yang bersifat reflektif, sehingga menjadi salah satu tantangan sekaligus bahan pengembangan untuk penelitian selanjutnya. Sistem ini terbukti dapat membantu pengguna mengecek kelengkapan barang pribadi secara efisien sebelum beraktivitas.

Kata kunci: YOLOv8, Smart Mirror, Object Detection, Barang Pribadi.

Implementation of Yolov8 on System Smart Mirror For Personal Item Detection and Reminder System

Abstract

This research aims to develop a smart mirror system capable of detecting and reminding users about personal items such as masks, glasses, and work identification cards using an object detection method with the YOLOv8 (You Only Look Once version 8) algorithm. The system was built using a dataset collected and annotated independently, then trained on the Google Colab platform, with a data split of 80% training and 20% validation. A camera integrated into the mirror captures images in real-time, which are then analyzed by the YOLOv8 model, and the system automatically issues a voice notification whenever one of the items is not detected. Testing results show that the model achieved an overall precision of 78.60%, with class-wise precision values of 97.65% for face masks, 80.26% for eyeglasses, and 74.21% for identification cards. The lower precision for the identification card class was mainly caused by its reflective surface, which poses a challenge for object detection and provides an opportunity for future improvements. This system has proven effective in helping users check the completeness of their personal items efficiently before starting their activities.

Keywords: YOLOv8, Smart Mirror, Object Detection, Personal Items.

Kata Pengantar

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan kasih karunia-Nya, sehingga penuli dapat menyelesaikan Laporan Tugas Akhir ini dengan baik. Laporan Tugas Akhir ini disusun untuk memenuhi salah satu syarat memperoleh gelar Sarjana Terapan Teknik (S.Tr.T) Program Studi Teknik Mekatronika di Politeknik Negeri Batam.

Laporan ini membahas implementasi sistem *Smart Mirror* untuk mendeteksi dan mengingatkan pengguna terhadap kelengkapan barang pribadi seperti masker, kacamata, dan kartu identitas kerja, menggunakan metode *object detection* dengan algoritma YOLOv8 (*You Only Look Once* versi 8). Sistem ini dirancang untuk membantu pengguna memastikan kelengkapan barang penting sebelum beraktivitas, sehingga dapat mengurangi risiko tertinggalnya barang-barang tersebut secara otomatis dan efisien.

Penulis mengucapkan terima kasih kepada:

1. Tuhan yang Maha Esa atas segala nikmat yang diberikan.
2. Kedua orang tua dan keluarga atas doa dan dukungannya.
3. Bapak Indra Hardian Mulyadi, S.T., M.Eng, selaku Kepala Jurusan Teknik Elektro.
4. Bapak Diono, S.Tr. T.,M.Sc. selaku Kepala Program Studi Teknik Mekatronika.
5. Bapak Daniel Sutopo Pamungkas, S.T., M.T., Ph.D selaku Dosen pembimbing tugas akhir.
6. Semua pihak yang tidak dapat disebutkan satu per satu yang telah memberikan bantuan kepada penulis, baik secara langsung maupun tidak langsung.

Penulis menyadari bahwa laporan ini masih memiliki kekurangan. Oleh karena itu, saran dan kritik yang membangun sangat diharapkan.

Batam, Juni 2026

Anjunius M

Daftar Isi

Pernyataan Keaslian Tugas Akhir	i
Lembar Pengesahan	ii
Abstrak	iii
<i>Abstract</i>	iv
Kata Pengantar	v
Daftar Isi	vi
Daftar Gambar	viii
Daftar Tabel	ix
Bab 1. Pendahuluan	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan	2
1.4. Manfaat	2
1.5. Batasan	2
Bab 2. Tinjauan Pustaka	3
2.1. Penelitian Terkait	3
2.2. Smart Mirror	6
2.3. Pengolahan Citra	6
2.4. Machine Learning	6
2.5. Algoritma YOLOv8	7
2.6. Sistem Notifikasi Otomatis	8
2.7. Kamera Webcam	8
2.8. Metrik Evaluasi Model	9
2.8.1 Precision	9
2.8.2 Recall	9
2.8.3 <i>Mean Average Precision (mAP)</i>	10
2.9. Anotasi Data (Data Labeling)	10
2.10. Pelatihan Model (Training)	10

Bab 3. Metodologi Penelitian / Metode Pelaksanaan	12
3.1. Perancangan Penelitian	12
3.1.1. Studi Literatur	13
3.1.2. Pengambilan Gambar	13
3.1.3. Anotasi Data	14
3.1.4. Pelatihan Model	14
3.1.5 Perancangan Perangkat Keras dan Perangkat Lunak.....	15
3.2. Perancangan Sistem Deteksi Barang Pribadi	18
3.3 Arsitektur Model YOLOv8 Pada Sistem Smart Mirror.....	20
3.4. Pengujian.....	21
3.4.1. Pengujian Fungsional.....	21
3.4.2. Evaluasi Presisi Model	22
3.4.3 Pengujian Performa Sistem Deteksi terhadap Pengguna	22
Bab 4. Hasil dan Pembahasan	23
4.1. Pengambilan Gambar	23
4.1.1 Anotasi Gambar Menggunakan Label Studio	23
4.1.2 Training Dataset	24
4.2. Perancangan Sistem	25
4.3. Hasil Evaluasi Model.....	26
4.4. Hasil Pengujian Sistem	28
4.5. Hasil Pengujian Sistem Terhadap Pengguna.....	29
4.5.1 Pengujian sistem terhadap pengguna Anjunius	31
4.5.2 Pengujian sistem terhadap pengguna Rafif	33
4.5.3 Pengujian sistem terhadap pengguna Fandi.....	34
Bab 5. Kesimpulan dan Saran	36
5.1. Kesimpulan.....	36
5.2. Saran	36
Daftar Pustaka	37
Biodata	39

Lampiran	40
----------------	----

Daftar Gambar

Gambar 1. Arsitektur YOLOv8	7
Gambar 2. Kamera WebCam	9
Gambar 3. Diagram Alir Rancangan Penelitian.....	12
Gambar 4. Objek Dataset	14
Gambar 5. Blok Diagram Sistem	15
Gambar 6. Rancangan Fisik Smart Mirror.....	16
Gambar 7. Tampilan GUI Aktif.....	17
Gambar 8. Diagram Alir Sistem Deteksi Barang Pribadi	18
Gambar 9. Contoh Pengambilan Gambar	23
Gambar 10. Hasil Anotasi	24
Gambar 11. Training Dataset	24
Gambar 12. Gambar Perancangan Sistem.....	25
Gambar 13. Grafik Hasil Pelatihan Model	27
Gambar 14. Tampilan Sistem Saat Melakukan Deteksi	28
Gambar 15. Pengujian sistem terhadap pengguna Anjunius	31
Gambar 16. Grafik Hasil Pengujian Anjunius	32
Gambar 17. Pengujian sistem terhadap pengguna Rafif	33
Gambar 18. Grafik Hasil Pengujian Rafif.....	33
Gambar 19. Pengujian sistem terhadap pengguna Fandi.....	34
Gambar 20. Grafik Hasil Pengujian Fandi	35

Daftar Tabel

Tabel 1. Penelitian Terkait	3
Tabel 2. Daftar Perangkat Keras	16
Tabel 3. Penjelasan Panel GUI	17
Tabel 4. Penjelasan Tahap Alur Deteksi Sistem	19
Tabel 5. Hasil Evaluasi Model Keseluruhan	27
Tabel 6. Hasil Evaluasi Precision Per Kelas Objek	28
Tabel 7. Hasil Pengujian Fungsional	29

Bab 1. Pendahuluan

1.1. Latar Belakang

Smart Mirror adalah solusi inovatif yang menggabungkan perangkat keras seperti kamera dan layar monitor dengan perangkat lunak berbasis *computer vision* untuk menampilkan informasi atau melakukan deteksi objek [1]. Berdasarkan observasi di lingkungan kerja yang diperkuat melalui penyebaran kuisioner kepada sejumlah karyawan. Diketahui bahwa sebanyak 75% sejumlah karyawan menyatakan pernah lupa membawa minimal satu barang pribadi saat akan berangkat kerja, dengan barang yang sering terlupakan adalah masker, kacamata dan kartu identitas kerja. Hal ini dapat menyebabkan keterlambatan atau hambatan saat bekerja. Oleh karena itu, dibutuhkan sistem pengingat otomatis yang mampu mendeteksi keberadaan barang penting dan memberikan peringatan kepada karyawan yang sering lupa membawa barang tersebut, guna mengurangi risiko gangguan atau keterlambatan dalam aktivitas kerja [2]. Untuk mencapai tujuan tersebut, dibutuhkan metode deteksi objek *real-time* yang cepat dan andal. Salah satu algoritma yang sering digunakan adalah YOLOv8 (*You Only Look Once versi 8*) yang terkenal dengan performa tinggi dan efisiensi pada berbagai platform komputasi [3].

Penelitian ini didasarkan pada rangkaian penelitian sebelumnya yang telah dilakukan oleh peneliti-peneliti terdahulu. Pertama penelitian yang berjudul “Penggunaan Algoritma YOLOv8 Dalam Mendeteksi Keberadaan Peternak Pada Area Peternakan Ayam” oleh Fhatiroy. Berdasarkan hasil pengujian, sistem mampu mencapai akurasi sebesar 94,1% dengan presisi 88%, menunjukkan bahwa model tersebut cukup baik untuk digunakan dalam dunia nyata [4].

Selain itu, Penelitian yang dilakukan oleh Stevanus Wijaya dan koleganya mengenai “Analisis Rancang Bangun Pada Sistem Pendeteksi Wajah untuk Keamanan Otomatis Berbasis IoT” menunjukkan bahwa sistem ini efektif untuk tujuan keamanan otomatis. Dalam pengujian menggunakan tiga wajah berbeda, sistem ini berhasil mengidentifikasi dan membedakan mereka dengan akurasi 100%. Pengujian dilakukan tanpa memperhitungkan jarak antara pengguna dan kamera, serta dalam kondisi pencahayaan standar dalam ruangan [5].

Berdasarkan latar belakang tersebut, khususnya terkait permasalahan yang sering dialami oleh karyawan dalam melupakan barang-barang penting seperti masker, kacamata dan tanda kartu identitas kerja, penelitian ini bertujuan untuk mengembangkan sistem *smart mirror* berbasis YOLOv8 yang dapat mendeteksi barang-barang pribadi secara *real-time*. Jika barang tidak terdeteksi, sistem akan mengirimkan notifikasi kepada pengguna melalui output suara. Dengan demikian, diharapkan dapat membantu pengguna agar tidak lupa membawa barang penting.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang dijelaskan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana mengimplementasikan algoritma YOLOv8 pada sistem *smart mirror* untuk mendeteksi barang-barang pribadi secara *real-time*?
2. Bagaimana sistem dapat mengidentifikasi ketiadaan barang tertentu dan memberikan notifikasi otomatis berupa output suara?
3. Seberapa akurat dan responsif sistem dalam mendeteksi barang pribadi dalam berbagai kondisi pencahayaan dan posisi objek?

1.3. Tujuan

Berdasarkan perumusan masalah yang telah diidentifikasi, tujuan dari penelitian ini adalah sebagai berikut:

1. Mengembangkan sistem *smart mirror* yang dilengkapi kamera dan algoritma YOLOv8 untuk mendeteksi barang pribadi seperti masker, kacamata, dan kartu identitas kerja secara *real-time*.
2. Merancang sistem pengingat otomatis yang mengirimkan notifikasi kepada pengguna jika barang tidak terdeteksi dalam kurun waktu tertentu.
3. Menganalisis performa sistem dalam hal akurasi dan presisi dari sistem yang telah dibuat.

1.4. Manfaat

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Membantu mengingatkan pengguna terhadap barang-barang pribadi penting sebelum melakukan aktivitas, sehingga mengurangi risiko kelupaan.
2. Memberikan gambaran mengenai integrasi sistem kamera, *object detection*, dan notifikasi dalam satu aplikasi *smart mirror*.

1.5. Batasan

Untuk memfokuskan ruang lingkup penelitian, batasan masalah yang ditetapkan adalah:

1. Sistem hanya dirancang untuk mendeteksi barang pribadi yang telah melalui proses pelabelan dan pelatihan menggunakan model YOLOv8.
2. Kamera hanya menangkap area tertentu di depan cermin dengan jarak dan kondisi pencahayaan yang telah di uji pada penelitian ini.

Bab 2. Tinjauan Pustaka

2.1. Penelitian Terkait

Berikut beberapa jurnal telah di jadikan acuan dalam pengembangan sistem *smart mirror* berbasis YOLOv8 untuk mendeteksi dan pengingat barang pribadi, yaitu sebagai berikut:

Tabel 1. Penelitian Terkait

No	Tahun	Penulis	Jurnal	Metode	Kelemahan
1	2023	Stevanus Wijaya, Sholihin	Analisis Rancang Bangun Pada Sistem Pendeteksi Wajah Untuk Keamanan Otomatis Berbasis IoT	Internet Of Things	Sistem Berbasis IoT Cenderung Sangat Tergantung Pada Koneksi Internet Sehingga ketika koneksi tidak ada akan mengganggu sistem.
2	2021	Hoppy Suraya	Rancang Bangun Alat Pendeteksi Kantuk Pada Mobil Berbasis IoT Menggunakan Kamera	Pengolahan Citra (Image Processing)	Sangat Bergantung Pada Kualitas Gambar, Keterbatasan Dalam Mendeteksi Tanda-Tanda Kantuk Dengan Akurat
3	2024	Andi Fahreza	Penggunaan Algoritma YOLOv8 Dalam Mendeteksi Keberadaan Peternak Pada Area Peternakan Ayam	Pengolahan Citra (Image Processing)	Sangat Bergantung Pada Kualitas Gambar dan membutuhkan dataseet yang banyak
4	2024	Mohammad Afif Muhajir, dkk	Lifestyle Assistance	Internet of Things	Bergantung pada koneksi internet,

No	Tahun	Penulis	Jurnal	Metode	Kelemahan
			Using Smart Mirror for Better Physcal and Spiritual	sebagai Planner jadwal harian	ketika koneksi internet buruk maka dapat mengganggu fungsional sistem
5	2024	Edhy Poerwandono	Implementasi YOLOv8 Untuk Mendeteksi Pelanggaran Lalu Lintas	Convolutional Neural Network	Membutuhkan dataset yang representatif, dan keterbatasan jangkauan area deteksi, pencahayaan dan akurasi.
6	2022	Zhevia Bella	Prototype Smart Mirror Dengan Goggle Nest Berbasis IoT	Internet of Things	Sangat bergantung pada Internet, dan tidak memiliki penyimpanan data sendiri sehingga pengingat jadwal harian bergantung pada goggle nest jika koneksi tidak stabil maka sistem akan terganggu
7	2024	Aufa Faiz Setyawan	Identifikasi Penyakit Pada Daun Mangga Menggunakan Algoritma YOLOv8	Pengolahan Citra (Image Processing)	Masih menggunakan dataseet yang sudah ada di kaggle sehingga menyulitkan mendeteksi berbagai jenis penyakit pada mangga yang berbeda

No	Tahun	Penulis	Jurnal	Metode	Kelemahan
8	2023	Jonathan Valentino	Analisis Perbandingan Optical Character Recognition Goggle Vision dengan Microsoft Computer Vision Pada Pembacaan KTP-el	Optical Character Recognition (OCR)	Ketidakmampuan mengenali tanda tangan dan tanda khusus, keterbatasan akurasi pengenalan karakter
9	2017	Erwin Susanto	Cermin Pintar Dengan Pengolahan Citra Berbasis Mikrokomputer	Pengolahan Citra (Image Processing)	Pada percobaan yang dilakukan selama lima kali percobaan dan hasil nya kotak wajah hanya bisa mendeteksi 40 x 40 pixel, dan sangat bergantung pada sensor PIR yang mendeteksi keberadaan objek pada cermin.
10	2024	Ega Hegarini, dkk	Penerapan Algoritma YOLOv5 Untuk Deteksi Kualitas Buah Alpukat Menggunakan Python Berbasis Web	Pengolahan Citra (Image Processing)	Sistem hanya mendeteksi buah alpukat yang masih utuh saja karena dataset yang dimiliki di ambil dari roboflow yang sudah ada.

2.2. Smart Mirror

Smart Mirror merupakan perangkat cermin yang diintegrasikan dengan teknologi digital, seperti layar tampilan, kamera, dan sensor, sehingga mampu menampilkan informasi tambahan sekaligus tetap berfungsi sebagai cermin konvensional. Konsep ini memanfaatkan sifat *two-way mirror* (cermin dua arah), yaitu material reflektif yang mampu memantulkan cahaya seperti cermin biasa, namun juga bersifat transparan terhadap cahaya dari layar monitor yang diletakkan di baliknya, sehingga tampilan digital dapat terlihat menyatu dengan permukaan cermin [6].

Secara umum, *smart mirror* terdiri dari beberapa komponen utama, yaitu cermin dua arah sebagai permukaan tampilan, monitor yang diletakkan di balik cermin untuk menampilkan antarmuka digital, serta unit pemroses (komputer atau *microcontroller*) yang menjalankan aplikasi tertentu. Beberapa *smart mirror* juga dilengkapi dengan kamera dan sensor tambahan untuk mendukung fitur interaktif, seperti pengenalan wajah, pengukuran waktu, penampil cuaca, hingga sistem *object detection*. Berdasarkan karakteristik tersebut, *smart mirror* pada penelitian ini tidak hanya berfungsi sebagai media tampilan informasi, tetapi juga sebagai sistem pengingat barang pribadi. Kamera webcam digunakan untuk menangkap citra pengguna, kemudian citra diproses menggunakan algoritma YOLOv8 untuk mendeteksi keberadaan masker, kacamata, dan kartu identitas. Selanjutnya sistem memberikan notifikasi suara apabila salah satu objek tidak terdeteksi.

2.3. Pengolahan Citra

Citra merupakan salah satu bentuk informasi penting yang sering digunakan selain teks, suara, dan video. Informasi visual ini tidak hanya berperan dalam komunikasi antar manusia, tetapi juga mendukung interaksi antara manusia dan mesin. Pemahaman terhadap isi dari sebuah citra bisa berbeda-beda tergantung pada individu yang menginterpretasikannya. Data yang dihasilkan dari sistem perekaman dapat berupa gambar optik seperti foto, sinyal video analog seperti yang terlihat di layar televisi, atau dalam bentuk data digital yang dapat langsung disimpan pada media penyimpanan [7].

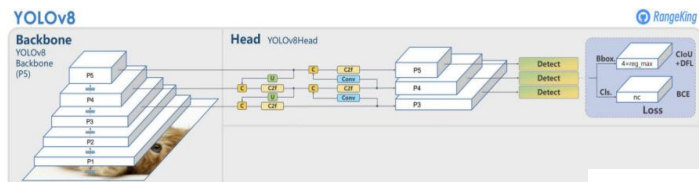
2.4. Machine Learning

Machine Learning merupakan salah satu cabang dari kecerdasan buatan (Artificial Intelligence/AI) yang menggabungkan prinsip-prinsip ilmu komputer dan statistik untuk membentuk model yang mampu mengenali pola dalam data[8]. Teknologi ini bertujuan untuk mengembangkan algoritma yang memungkinkan komputer belajar dari data tanpa instruksi pemrograman yang

eksplisit untuk setiap tugas tertentu. Berbagai pendekatan digunakan dalam *machine learning*, seperti regresi, klasifikasi, dan optimasi, guna menganalisis data dan menghasilkan prediksi atau keputusan berdasarkan data yang belum pernah dilihat sebelumnya. Penerapannya sangat luas, meliputi pengenalan wajah, pengenalan suara, pemrosesan bahasa alami, sistem rekomendasi, analisis risiko finansial, serta deteksi pola dalam berbagai konteks[9].

2.5. Arsitektur YOLOv8

YOLO (*You Only Look Once* versi 8) merupakan salah satu algoritma deteksi objek real-time yang paling populer. YOLO bekerja dengan cara memproses gambar hanya satu kali (*single pass*) untuk menghasilkan bounding box dan klasifikasi objek secara simultan. Versi terbaru, YOLOv8, dikembangkan oleh Ultralytics dan menghadirkan peningkatan dari sisi akurasi, efisiensi model, serta kecepatan proses. YOLOv8 mendukung berbagai ukuran model seperti *nano*, *small*, *medium*, dan *large*, sehingga dapat disesuaikan dengan kebutuhan perangkat keras[10].



Gambar 1. Arsitektur YOLOv8

Arsitektur YOLOv8 terdiri atas beberapa komponen utama, yaitu *input layer*, *backbone*, *neck*, *head*, dan *output layer*. Setiap komponen memiliki fungsi yang saling berkaitan dalam proses deteksi objek sehingga model mampu menghasilkan prediksi secara cepat dan akurat [11]. Proses deteksi diawali pada input layer, yaitu tahap ketika citra diterima sebagai masukan oleh model. Sebelum diproses lebih lanjut, citra terlebih dahulu melalui tahap *preprocessing* yang meliputi normalisasi nilai piksel dan penyesuaian ukuran citra (*resize*) agar sesuai dengan ukuran input yang digunakan oleh model.

Selanjutnya, citra diproses oleh bagian backbone yang berfungsi melakukan ekstraksi fitur. Pada YOLOv8, proses ekstraksi fitur dilakukan menggunakan beberapa lapisan konvolusi (*Convolution*), modul C2f, dan Spatial Pyramid Pooling Fast (SPPF). Kombinasi komponen tersebut memungkinkan model memperoleh representasi fitur yang lebih kaya sehingga objek dapat dikenali dengan lebih baik.

Fitur yang telah diekstraksi kemudian diteruskan ke bagian neck. Bagian ini menggunakan kombinasi Feature Pyramid Network (FPN) dan Path Aggregation

Network (PAN) untuk menggabungkan informasi dari berbagai tingkat resolusi. Penggabungan fitur tersebut membantu model mendeteksi objek dengan ukuran yang beragam serta meningkatkan kualitas hasil deteksi.

Tahap berikutnya adalah head, yaitu bagian yang bertugas menghasilkan prediksi deteksi objek. Pada tahap ini, model menentukan posisi objek dalam bentuk *bounding box*, memprediksi kelas objek, serta menghitung nilai *confidence score* yang menunjukkan tingkat keyakinan model terhadap hasil prediksi. Hasil prediksi kemudian diteruskan ke output layer yang menghasilkan informasi berupa koordinat *bounding box*, probabilitas kelas, dan nilai *confidence score*. Informasi tersebut digunakan sebagai dasar dalam menentukan objek yang berhasil dikenali oleh model.

Pada penelitian ini, arsitektur YOLOv8 digunakan untuk mendeteksi tiga kelas objek, yaitu kacamata, masker, dan kartu identitas. Hasil deteksi tersebut kemudian dimanfaatkan oleh sistem Smart Mirror untuk memeriksa kelengkapan barang pribadi pengguna. Apabila salah satu objek tidak terdeteksi, sistem akan memberikan peringatan dalam bentuk suara.

2.6. Sistem Notifikasi Otomatis

Sistem notifikasi otomatis merupakan fitur yang digunakan untuk memberikan peringatan kepada pengguna secara langsung. Dalam konteks penelitian ini, notifikasi *output* suara digunakan ketika barang pribadi tidak terdeteksi oleh sistem. Notifikasi bertujuan untuk mengingatkan pengguna agar tidak melupakan barang sebelum beraktivitas[12].

Sebelum notifikasi dikirim, sistem pembacaan objek harus sangat presisi agar dapat mengetahui barang-barang pribadi yang dibutuhkan sebelum meninggalkan rumah. Pembacaan ini dilakukan dengan menggunakan OpenCV (*Open Source Computer Vision Library*), yaitu pustaka *open source* yang digunakan untuk pemrosesan citra dan video. OpenCV mendukung berbagai fungsi seperti pembacaan kamera, deteksi wajah, pelacakan objek, dan anotasi *bounding box*. Dalam penelitian ini, OpenCV dimanfaatkan untuk menangkap citra secara *real-time* dari kamera dan menampilkan hasil deteksi objek menggunakan YOLOv8.

2.7. Kamera Webcam

Webcam merupakan perangkat kamera digital yang terhubung ke komputer atau laptop melalui koneksi USB atau jaringan nirkabel [13]. Perangkat ini berfungsi untuk menangkap citra atau video secara langsung dan mengirimkannya ke sistem untuk diproses. Dalam penelitian ini, webcam digunakan sebagai alat bantu untuk menangkap citra secara *real-time*, yang kemudian dianalisis menggunakan algoritma YOLOv8 untuk mendeteksi

keberadaan barang-barang pribadi seperti masker, kacamata dan kartu identitas kerja.



Gambar 2. Kamera WebCam

2.8. Metrik Evaluasi Model

Dalam mengevaluasi performa model *object detection*, terdapat beberapa metrik standar yang digunakan untuk mengukur seberapa baik model dalam mendeteksi objek secara benar. Metrik-metrik ini dihitung berdasarkan empat kemungkinan hasil prediksi, yaitu *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN) [14].

2.8.1 Precision

Precision mengukur seberapa tepat prediksi yang dihasilkan oleh model, yaitu proporsi prediksi positif yang benar-benar merupakan objek yang dimaksud. Nilai *precision* yang tinggi menunjukkan bahwa model jarang melakukan kesalahan deteksi (*false positive*), seperti mendeteksi objek yang sebenarnya tidak ada [15]. *Precision* dihitung dengan rumus:

$$Precision = \frac{TP}{TP + FP} \quad (2.1)$$

2.8.2 Recall

Recall mengukur seberapa baik model dalam menemukan seluruh objek yang seharusnya terdeteksi, yaitu proporsi objek yang berhasil dideteksi dari seluruh objek yang sebenarnya ada. Nilai *recall* yang tinggi menunjukkan bahwa model jarang melewatkan objek yang seharusnya terdeteksi (*false negative*) [16]. *Recall* dihitung dengan rumus:

$$Recall = \frac{TP}{TP + FN} \quad (2.2)$$

2.8.3 Mean Average Precision (mAP)

Mean Average Precision (mAP) merupakan metrik utama yang digunakan untuk mengevaluasi performa keseluruhan model *object detection* pada seluruh kelas objek yang dideteksi. mAP dihitung dengan merata-ratakan nilai *Average Precision* (AP) dari setiap kelas objek, di mana AP sendiri merepresentasikan tingkat kesesuaian antara hasil prediksi model dengan kondisi objek yang sebenarnya (*ground truth*) [17]. Semakin tinggi nilai mAP, semakin baik kemampuan model dalam mendeteksi seluruh kelas objek secara keseluruhan.

$$mAP = \frac{1}{n} \sum_{i=1}^n AP_i \quad (2.3)$$

2.9. Anotasi Data (Data Labeling)

Anotasi data merupakan proses pemberian label atau penandaan pada data mentah agar dapat dikenali dan dipelajari oleh model *machine learning*, khususnya dalam tugas *object detection* [18]. Pada kasus deteksi objek, anotasi dilakukan dengan menggambar *bounding box* (kotak pembatas) di sekeliling objek yang ingin dideteksi pada setiap citra, kemudian memberikan label kelas sesuai jenis objeknya. Proses ini penting karena model YOLOv8 belajar mengenali pola visual suatu objek berdasarkan contoh yang telah diberi label secara manual, sehingga kualitas dan konsistensi anotasi sangat memengaruhi akurasi.

Dalam penelitian ini, proses anotasi dilakukan menggunakan Label Studio, yaitu perangkat lunak *open source* berbasis web yang memungkinkan pengguna melakukan anotasi data secara manual untuk berbagai jenis tugas, termasuk *object detection*, klasifikasi teks, dan segmentasi gambar. Label Studio dipilih karena mendukung ekspor hasil anotasi langsung dalam format YOLO, sehingga memudahkan proses integrasi dengan tahap pelatihan model.

2.10. Pelatihan Model (Training)

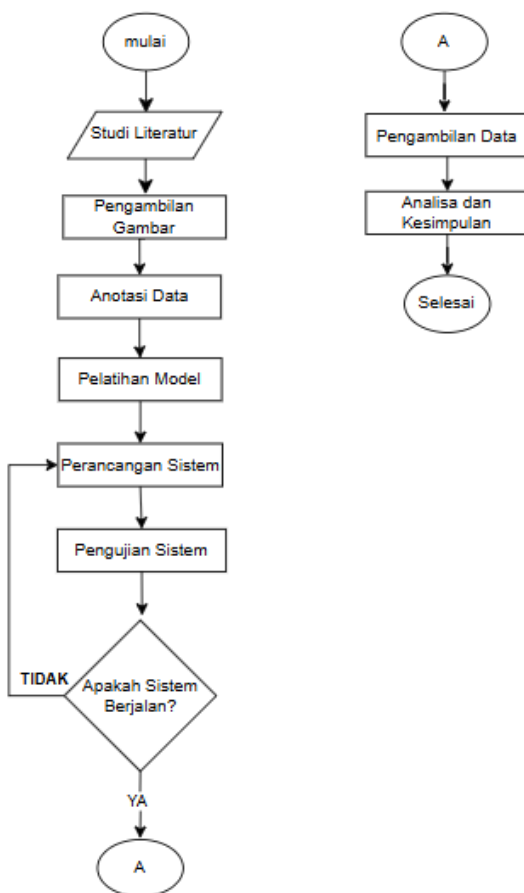
Pelatihan model (*training*) merupakan proses di mana model *machine learning* "belajar" mengenali pola dari data yang telah dianotasi, dengan cara menyesuaikan parameter internal model secara berulang hingga mampu memprediksi kelas objek dengan akurat [19]. Proses ini dilakukan dalam beberapa siklus yang disebut *epoch*, di mana pada setiap *epoch*, seluruh data *training* diproses oleh model satu kali penuh. Selain itu, terdapat parameter *batch size* yang menentukan jumlah data yang diproses secara bersamaan dalam satu iterasi pelatihan, serta pembagian dataset menjadi data *training* (untuk

melatih model) dan data *validation* (untuk mengevaluasi performa model pada data yang belum pernah dipelajari) [20].

Dalam penelitian ini, proses pelatihan model YOLOv8 dilakukan menggunakan Google Colab (*Google Colaboratory*), yaitu layanan berbasis *cloud* yang disediakan oleh Google untuk menulis dan menjalankan kode Python secara *online* melalui *browser*, tanpa memerlukan instalasi atau konfigurasi perangkat keras khusus di komputer pengguna. Salah satu keunggulan utama Google Colab adalah ketersediaan akses GPU (*Graphics Processing Unit*) secara gratis, yang dapat mempercepat proses pelatihan model *deep learning* secara signifikan dibandingkan menggunakan CPU biasa

Bab 3. Metodologi Penelitian / Metode Pelaksanaan

3.1. Perancangan Penelitian



Gambar 3. Diagram Alir Rancangan Penelitian

Pada Gambar 3, agar penelitian dapat berjalan dengan baik, dibutuhkan perencanaan dan perancangan sistem yang matang. Tahapan perancangan ini mencakup berbagai langkah yang diperlukan untuk menjawab rumusan masalah yang telah ditetapkan. Adapun langkah-langkah tersebut dijabarkan dalam proses perancangan berikut ini.

3.1.1. Studi Literatur

Studi literatur dilakukan dengan mencari dan memahami berbagai literatur yang relevan dengan topik penelitian ini, yaitu implementasi YOLOv8 pada sistem *smart mirror* untuk deteksi dan pengingat barang pribadi. Sumber informasi yang digunakan mencakup buku, artikel jurnal ilmiah, artikel internet, media cetak, dan sumber informasi lainnya yang mendukung penelitian ini. Dari studi literatur yang dilakukan, penulis memperoleh berbagai informasi penting sebagai landasan penelitian, yaitu pemahaman tentang karakteristik dan cara kerja algoritma YOLOv8 sebagai algoritma deteksi objek *real-time*, konsep dan komponen dasar *smart mirror* sebagai integrasi antara cermin fisik dengan perangkat digital, proses anotasi data menggunakan Label Studio yang menjadi tahap penting sebelum pelatihan model, mekanisme pelatihan model *deep learning* pada platform Google Colab dengan dukungan GPU, serta metrik evaluasi model deteksi objek, seperti *precision*, *recall*, dan *mean Average Precision* (mAP). Selain itu, penulis juga mempelajari beberapa penelitian sebelumnya yang berkaitan, sebagaimana telah dijelaskan pada Subbab 2.1 (Penelitian Terkait), yang menjadi acuan untuk membangun sistem yang lebih baik dari penelitian-penelitian sebelumnya.

3.1.2. Pengambilan Gambar

Dalam penelitian ini, proses pengambilan gambar dilakukan menggunakan kamera *smartphone* sedangkan pengujian sistem *Smart Mirror* secara *real-time* menggunakan kamera *webcam* yang terpasang pada perangkat. Kamera berfungsi untuk menangkap citra secara *real-time* yang menampilkan barang-barang pribadi pengguna, seperti masker, kacamata, dan kartu identitas kerja. Gambar-gambar tersebut digunakan untuk membangun dataset sebagai bahan pelatihan model YOLOv8 dan untuk pengujian sistem deteksi objek.

Adapun spesifikasi umum pengambilan gambar adalah sebagai berikut:

- Kelas objek : masker, kacamata dan kartu identitas kerja.
- Kondisi pencahayaan: pengambilan gambar dilakukan di dalam ruangan menggunakan pencahayaan lampu ruangan, sehingga kondisi cahaya relatif konsisten tanpa dipengaruhi oleh waktu siang atau malam di luar ruangan.
- Jarak pengambilan gambar: dilakukan pada beberapa variasi jarak (dekat dan jauh) dari kamera untuk mendapatkan variasi ukuran objek dalam citra.

- Posisi dan sudut objek: setiap objek diambil dalam berbagai sudut dan orientasi (misalnya kacamata dipakai dengan posisi wajah lurus dan menyamping, kartu identitas dipegang dari berbagai sisi), guna melatih model agar dapat mengenali objek meski tidak selalu berada pada posisi yang sama.



Gambar 4. Objek Dataset

3.1.3 Anotasi Data

Pada tahap ini, dilakukan anotasi terhadap seluruh citra hasil pengambilan gambar. Anotasi dilakukan menggunakan Label Studio, yaitu perangkat lunak *open source* berbasis web yang memungkinkan pengguna melakukan pelabelan data secara manual untuk berbagai jenis tugas *machine learning*, termasuk *object detection*. Label Studio dipilih karena mendukung ekspor hasil anotasi langsung dalam format YOLO, sehingga memudahkan proses integrasi dengan tahap pelatihan model YOLOv8. Proses anotasi dilakukan dengan cara menggambar *bounding box* (kotak pembatas) di sekeliling setiap objek yang ingin dideteksi (masker, kacamata, dan kartu identitas kerja), kemudian memberikan label kelas sesuai jenis objeknya. Setiap gambar harus dianotasi secara manual dan konsisten, karena kualitas dan konsistensi anotasi akan menentukan seberapa baik model dapat mengenali objek tersebut nantinya. Setelah seluruh gambar selesai dianotasi, hasil anotasi akan dibagi menjadi dua bagian, yaitu 80% sebagai data *training* dan 20% sebagai data *validation*, kemudian diekspor dalam format YOLO untuk digunakan pada tahap Pelatihan Model.

3.1.4 Pelatihan Model

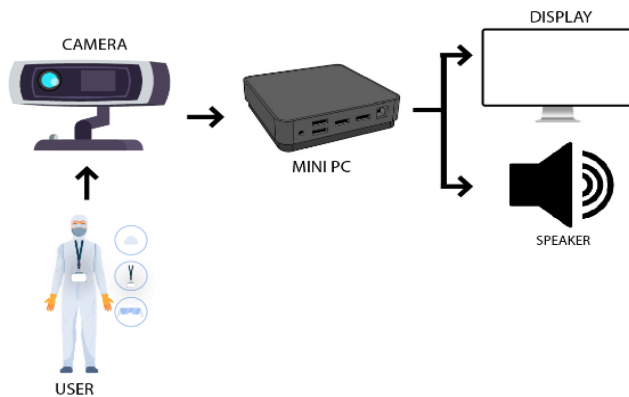
Setelah proses anotasi data selesai dan dataset telah dibagi menjadi data *training* dan *validation*, tahap berikutnya adalah pelatihan model YOLOv8 menggunakan dataset tersebut. Tahap ini bertujuan agar model mampu mengenali pola visual dari ketiga objek yang dideteksi, yaitu masker, kacamata, dan kartu identitas kerja, sehingga dapat digunakan pada sistem *smart mirror* untuk melakukan deteksi objek secara *real-time*.

Pelatihan model dilakukan menggunakan platform Google Colab, yaitu layanan berbasis *cloud* yang disediakan oleh Google untuk menulis dan menjalankan kode Python secara *online* melalui *browser*. Google Colab dipilih

karena menyediakan akses GPU (*Graphics Processing Unit*) secara gratis, yang mampu mempercepat proses pelatihan model *deep learning* secara signifikan dibandingkan menggunakan CPU biasa. Pemanfaatan GPU sangat penting dalam pelatihan model YOLOv8, karena proses pelatihan melibatkan komputasi yang cukup berat dan berulang dalam jumlah besar.

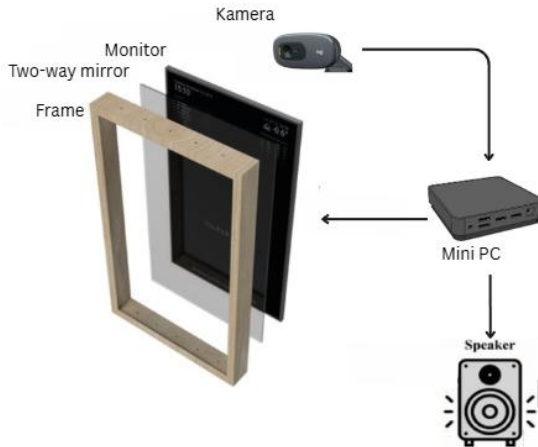
3.1.5 Perancangan Perangkat Keras dan Perangkat Lunak

Sistem *smart mirror* dalam penelitian ini dirancang sebagai perangkat terintegrasi yang menggabungkan komponen perangkat keras (*hardware*) dan perangkat lunak (*software*). Alur data antar komponen sistem ditunjukkan dalam bentuk blok diagram pada Gambar 4.



Gambar 5. Blok Diagram Sistem

Berdasarkan Gambar 5, pengguna (*user*) berdiri di depan sistem dengan menggunakan atau membawa barang-barang pribadi (masker, kacamata, dan kartu identitas kerja). Kamera *webcam* menangkap citra pengguna secara *real-time*, kemudian citra tersebut dikirimkan ke Mini PC yang berfungsi sebagai unit pemroses utama. Pada Mini PC, model YOLOv8 dijalankan untuk mendeteksi objek pada citra. Hasil deteksi kemudian ditampilkan pada *display* (monitor), sedangkan notifikasi suara dikeluarkan melalui *speaker* eksternal apabila terdapat objek yang tidak terdeteksi.



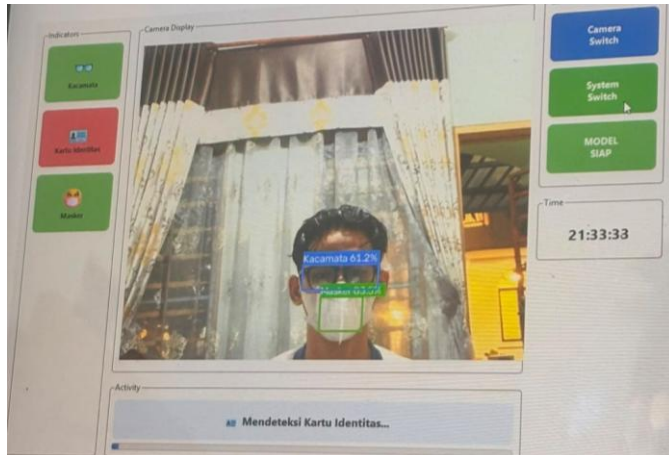
Gambar 6. Rancangan Fisik Smart Mirror

Berdasarkan Gambar 6, seluruh komponen sistem disatukan dalam sebuah *frame* kayu yang berfungsi sebagai rangka penopang. Monitor diletakkan di balik *two-way mirror*, sehingga tampilan digital dari monitor dapat terlihat menyatu dengan permukaan cermin. Kamera *webcam* Logitech dipasang di bagian atas rangka untuk menangkap citra pengguna, sedangkan Mini PC dan *speaker* diletakkan terpisah namun tetap terhubung ke sistem. Adapun daftar perangkat keras yang digunakan pada sistem ini ditunjukkan pada Tabel 2.

Tabel 2. Daftar Perangkat Keras

No	Komponen	Fungsi
1	<i>Frame</i> Kayu	Rangka Penopang seluruh komponen sistem
2	<i>Two-way mirror</i>	Permukaan cermin yang memantulkan bayangan sekaligus menampilkan tampilan digital dari monitor di baliknya.
3	Monitor	Menampilkan antarmuka pengguna (GUI) sistem
4	Kamera <i>Webcam</i> Logitech	Menangkap citra pengguna secara <i>real-time</i>
5	Mini PC	Unit pemroses utama yang menjalankan model YOLOv8 dan seluruh aplikasi sistem
6	<i>Speaker</i> eksternal	Mengeluarkan notifikasi suara ketika ada barang tidak terdeteksi

Selain perangkat keras, sistem *smart mirror* ini juga dilengkapi dengan perangkat lunak berupa antarmuka pengguna (*Graphical User Interface / GUI*) yang dirancang untuk menampilkan proses deteksi barang pribadi secara langsung kepada pengguna. Tampilan GUI sistem *smart mirror* ditunjukkan pada Gambar 6.



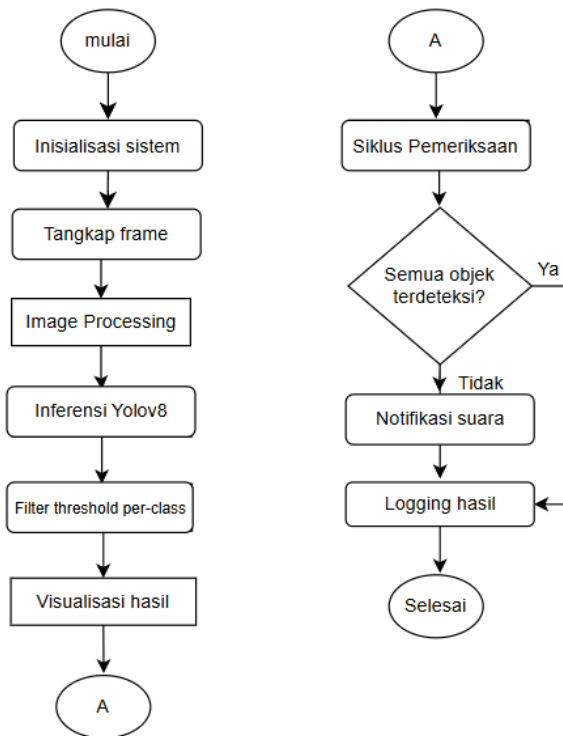
Gambar 7. Tampilan GUI Aktif

Berdasarkan Gambar 7, GUI sistem dibangun dengan tampilan *dashboard* yang terbagi menjadi empat panel utama untuk memudahkan pengguna dalam memahami status deteksi. Adapun penjelasan dari masing-masing panel ditunjukkan pada Tabel 3.

Tabel 3. Penjelasan Panel GUI

Panel	Lokasi	Fungsi
Indicator	Kiri	Menampilkan status tiap objek (hijau =terdeteksi, merah = belum)
Camera Display	Tengah	Menampilkan Kamera <i>real-time</i> + <i>bounding box</i> deteksi
Control dan Data	Kanan	Tombol kontrol (Camera Switch, System Switch) + status model + waktu
Activity	Bawah	Status siklus deteksi + <i>progress bar</i>

3.2. Perancangan Sistem Deteksi Barang Pribadi



Gambar 8. Diagram Alir Sistem Deteksi Barang Pribadi

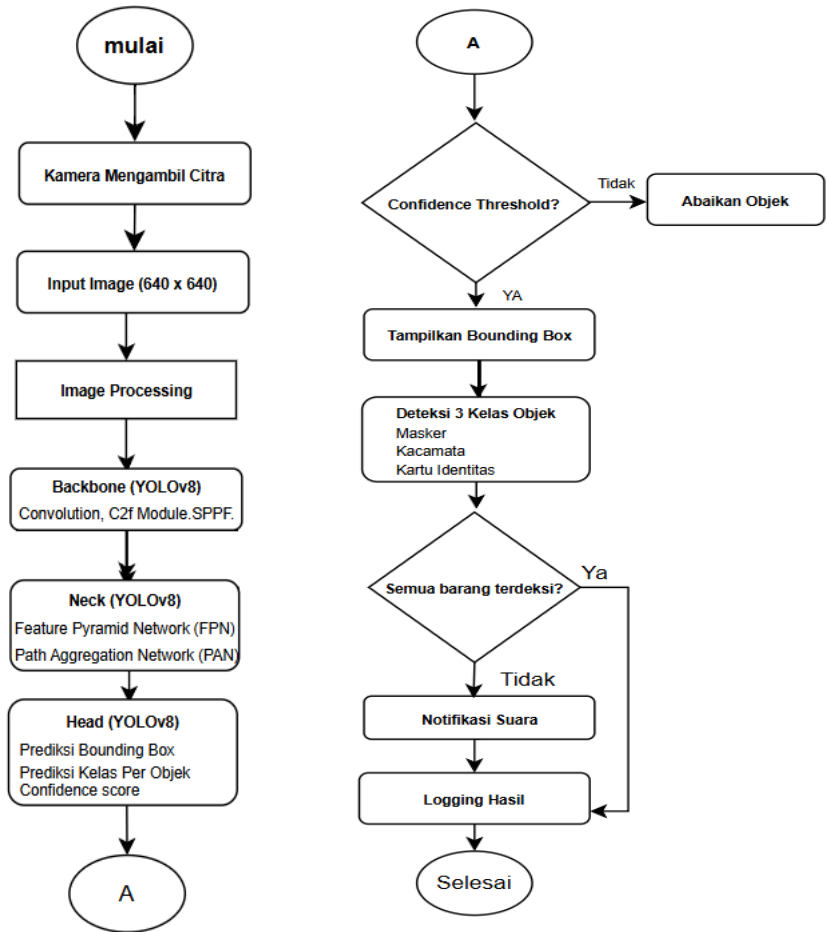
Diagram alir pada Gambar 8, menggambarkan alur kerja sistem deteksi barang pribadi pada *smart mirror* secara menyeluruh, mulai dari inisialisasi sistem hingga pencatatan hasil deteksi. Alur ini berjalan secara berulang selama sistem aktif, sehingga sistem dapat melakukan deteksi secara *real-time* dan memberikan respons langsung kepada pengguna.

Tabel 4. Penjelasan Tahap Alur Deteksi Sistem

No	Proses	Keterangan
1	Inisialisasi Sistem	Membuat model Yolov8 dan Komponen GUI
2	Tangkap Frame	Kamera menangkap citra <i>real-time</i>
3	<i>Image Processing</i>	Resize 640 x 640 + konversi BGR ke RGB
4	Inferensi YOLOv8	Model memprediksi objek dan <i>confidence</i>
5	Filter Threshold	Menyaring hasil per-kelas objek
6	Visualisasi Hasil	<i>Bounding box</i> & label pada GUI
7	Siklus Pemeriksaan	Cek Masker, Kacamata, Kartu Identitas
8	Evaluasi Kelengkapan	Semua objek terdeteksi?
9	Notifikasi Suara	<i>Text-to-speech</i> jika tidak lengkap
10	Pencatatan (<i>Logging</i>)	Simpan ke <i>log_deteksi.xlsx</i>

Berdasarkan Tabel 4, terlihat bahwa sistem tidak hanya melakukan proses deteksi objek secara langsung, tetapi juga melalui beberapa tahap pemrosesan penting seperti *image processing*, filter *threshold* per-kelas, dan siklus pemeriksaan bertahap. Tahap *image processing* diperlukan untuk memastikan citra dari kamera sesuai dengan format input yang dibutuhkan model YOLOv8, sedangkan filter *threshold* per-kelas digunakan untuk menyesuaikan tingkat sensitivitas deteksi pada masing-masing objek. Objek dengan karakteristik yang lebih sulit dideteksi, seperti kartu identitas kerja yang memiliki permukaan reflektif, diberikan ambang batas yang lebih rendah agar tetap dapat terdeteksi dengan baik. Selain proses deteksi, sistem juga dilengkapi dengan fitur pencatatan hasil (*logging*) yang secara otomatis menyimpan setiap hasil deteksi ke dalam berkas *log_deteksi.xlsx*. Fitur ini penting untuk keperluan evaluasi dan pengujian sistem, karena memungkinkan pengambilan data secara sistematis tanpa harus mencatat manual setiap kali pengujian dilakukan.

3.3 Arsitektur Model YOLOv8 Pada Sistem Smart Mirror



Gambar 9. Arsitektur YOLOv8 Pada Sistem Smart Mirror

Arsitektur model YOLOv8 yang digunakan pada penelitian ini terdiri atas beberapa tahapan, yaitu preprocessing citra, backbone, neck, dan head. Proses diawali dengan kamera yang menangkap citra pengguna, kemudian citra diproses

melalui tahap preprocessing berupa perubahan ukuran citra menjadi 640×640 piksel dan normalisasi data sebelum dimasukkan ke dalam model YOLOv8.

Pada tahap backbone, model melakukan ekstraksi fitur menggunakan lapisan Convolution, modul C2f, dan Spatial Pyramid Pooling Fast (SPPF). Selanjutnya, fitur-fitur tersebut diteruskan ke bagian neck yang menggunakan Feature Pyramid Network (FPN) dan Path Aggregation Network (PAN) untuk menggabungkan informasi dari berbagai skala sehingga model mampu mendeteksi objek dengan ukuran yang berbeda.

Tahap head menghasilkan prediksi berupa lokasi objek (bounding box), kelas objek, dan nilai confidence. Hasil prediksi kemudian dibandingkan dengan nilai threshold yang telah ditentukan. Objek yang memiliki nilai confidence di atas threshold akan ditampilkan pada layar Smart Mirror beserta label kelasnya. Pada penelitian ini, model hanya mendeteksi tiga kelas objek, yaitu kaca mata, masker, dan kartu identitas. Selanjutnya, sistem memeriksa apakah ketiga objek tersebut telah terdeteksi. Jika seluruh objek berhasil dikenali, sistem menyatakan bahwa barang pribadi telah lengkap. Sebaliknya, apabila salah satu objek tidak terdeteksi, sistem memberikan pengingat dalam bentuk suara kepada pengguna.

3.4. Pengujian

Pengujian sistem dilakukan untuk mengevaluasi performa dan keandalan sistem *smart mirror* dalam mendeteksi keberadaan barang-barang pribadi menggunakan algoritma YOLOv8. Tujuan dari pengujian ini adalah untuk memastikan bahwa sistem mampu mendeteksi objek dengan presisi tinggi, memberikan respons yang cepat, serta mengirimkan notifikasi apabila objek tidak terdeteksi. Pengujian yang dilakukan mencakup pengujian fungsional, evaluasi presisi model, dan pengujian terhadap pengguna.

3.4.1. Pengujian Fungsional

Pengujian fungsional dilakukan untuk memastikan sistem dapat mendeteksi kombinasi kelengkapan barang serta memberikan notifikasi suara yang sesuai. Pengujian ini dilakukan dengan mengamati sistem terhadap kondisi kehadiran objek pada tiga skenario utama, yaitu (1) seluruh objek tersedia, (2) satu atau lebih objek tidak tersedia, dan (3) seluruh objek tidak tersedia. Pengujian dilakukan secara *real-time* menggunakan kamera *webcam* Logitech yang terhubung ke Mini PC, dan hasilnya diamati dari visualisasi *bounding box*, indikator status pada antarmuka sistem, serta pengiriman notifikasi suara. Validasi keberhasilan dilakukan dengan mencocokkan hasil deteksi sistem dengan kondisi nyata pada saat pengujian.

3.4.2. Evaluasi Presisi Model

Evaluasi presisi model dilakukan untuk mengukur seberapa tepat model YOLOv8 dalam mendeteksi objek. Metrik utama yang digunakan adalah Precision, sebagaimana telah dijelaskan pada Subbab 2.8. *Precision* dipilih sebagai metrik utama karena metrik ini secara langsung menunjukkan seberapa besar tingkat kebenaran sistem dalam mengenali objek, yaitu proporsi objek yang dideteksi model yang benar-benar sesuai dengan objek yang dimaksud. Semakin tinggi nilai *precision*, semakin dapat diandalkan sistem dalam memberikan notifikasi yang akurat kepada pengguna

3.4.3 Pengujian Performa Sistem Deteksi terhadap Pengguna

Pengujian terhadap pengguna dilakukan untuk mengetahui konsistensi dan kemampuan generalisasi sistem dalam mendeteksi barang pribadi pada individu yang berbeda. Pengujian dilakukan terhadap tiga pengguna dengan karakteristik fisik yang berbeda-beda, seperti bentuk wajah, model kacamata yang dikenakan, dan cara membawa kartu identitas kerja. Masing-masing pengguna menggunakan sistem selama satu minggu (tujuh hari), dan hasil deteksi tercatat secara otomatis pada berkas *log* sistem (*log_deteksi.xlsx*).

Bab 4. Hasil dan Pembahasan

4.1. Pengambilan Gambar

Proses pengambilan gambar yang telah menghasilkan sejumlah citra yang mencakup tiga kelas objek, yaitu masker, kacamata, dan kartu identitas kerja. Pengambilan gambar dilakukan menggunakan kamera *smartphone* dengan variasi jarak, posisi, dan sudut pengambilan, guna memperoleh keragaman data yang cukup untuk proses pelatihan model.

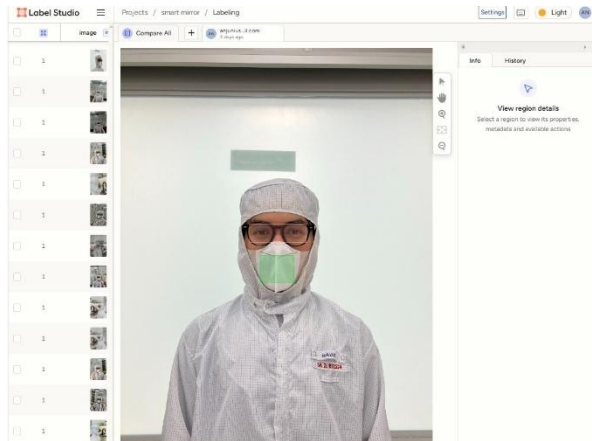


Gambar 10. Contoh Pengambilan Gambar

Pengambilan gambar dilakukan dengan fokus pada objek yang akan dideteksi (masker, kacamata, dan kartu identitas kerja), dengan mempertimbangkan variasi jarak antara objek dan kamera untuk mendapatkan variasi ukuran objek dalam citra. Selain itu, gambar diambil dari berbagai sudut dan orientasi objek, seperti kacamata yang difoto dari posisi wajah lurus maupun menyamping, serta kartu identitas yang dipegang dari berbagai sisi. Seluruh proses pengambilan gambar dilakukan di dalam ruangan dengan pencahayaan lampu ruangan yang relatif konsisten, sehingga kondisi cahaya tidak dipengaruhi oleh waktu siang maupun malam di luar ruangan.

4.1.1 Anotasi Gambar Menggunakan Label Studio

Setelah proses pengambilan gambar selesai, langkah selanjutnya adalah melakukan anotasi data menggunakan platform Label Studio. Proses anotasi dilakukan dengan menggambar *bounding box* pada setiap objek (masker, kacamata, dan kartu identitas kerja) di setiap citra, kemudian memberikan label kelas sesuai jenis objeknya. Setiap gambar diberikan anotasi secara manual untuk menandai posisi objek secara presisi. Anotasi yang akurat sangat penting untuk membantu model mengenali objek dengan baik selama proses pelatihan. Setelah seluruh gambar selesai dianotasi, dataset dibagi menjadi dua bagian, yaitu data *training* (80%) dan data *validation* (20%), kemudian diekspor dalam format YOLO untuk digunakan pada tahap pelatihan model.



Gambar 11. Hasil Anotasi

4.1.2 Training Dataset

Setelah dataset selesai dianotasi dan diekspor dalam format YOLO, tahap berikutnya adalah proses pelatihan model. Pelatihan dilakukan menggunakan platform Google Colab dengan memanfaatkan akselerasi GPU Tesla T4 yang disediakan secara gratis. Sebelum pelatihan dimulai, dataset yang telah diekspor dari Label Studio terlebih dahulu diunggah ke Google Drive, kemudian dihubungkan (*mount*) ke Google Colab. Proses instalasi *library* Ultralytics dan persiapan dataset.

```
[9] ✓ Os
data_yaml_content = """
path: /content/dataset/dataset
train: images/train
val: images/val

names:
  0: Kacamata
  1: Kartu Identitas
  2: Masker
"""

with open("/content/data.yaml", "w") as f:
    f.write(data_yaml_content)

print("data.yaml berhasil diperbarui")

# Verifikasi isi file
with open("/content/data.yaml", "r") as f:
    print(f.read())

data.yaml berhasil diperbarui
path: /content/dataset/dataset
train: images/train
val: images/val

names:
  0: Kacamata
  1: Kartu Identitas
  2: Masker

[10] ✓ Os
import os

train_path = "/content/dataset/dataset/images/train"
val_path = "/content/dataset/dataset/images/val"

print("train folder ada?", os.path.isdir(train_path), "- ", loader)
print("val folder ada?", os.path.isdir(val_path), "- ", loader)

Train folder ada? True - 488 file
Val folder ada? True - 121 file

[10] from ultralytics import YOLO

model = YOLO("yolov8n.pt")

model.train(
    data="/content/data.yaml",
    epochs=100,
    imgsz=100,
    batch=10,
    project="/content/drive/MyDrive/smart_mirror/runs",
    name="train1"
)

Epoch      GPU_mem  box_loss  cls_loss  dfl_loss  Instance
57/100      2.47G    1.071     0.7855    1.112     0.8
Class      Images
all         121      171     0.801
Epoch      GPU_mem  box_loss  cls_loss  dfl_loss  Instance
58/100      2.47G    1.068     0.7853    1.100     0.8
Class      Images
all         121      171     0.793
Epoch      GPU_mem  box_loss  cls_loss  dfl_loss  Instance
59/100      2.47G    1.080     0.6889    1.004     0.8
Class      Images
all         121      171     0.822
Epoch      GPU_mem  box_loss  cls_loss  dfl_loss  Instance
60/100      2.47G    1.078     0.7888    1.090     0.8
Class      Images
all         121      171     0.833
```

Gambar 12. Training Dataset

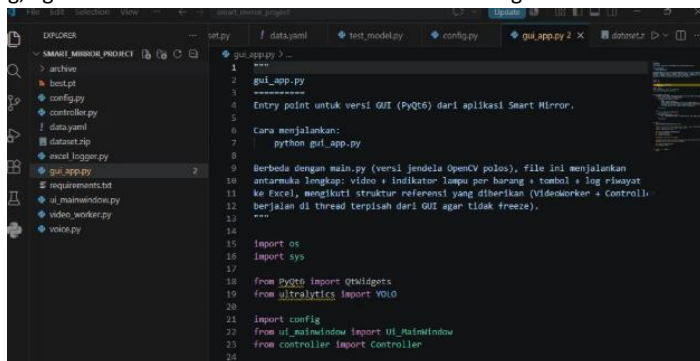
Beberapa parameter penting dalam proses pelatihan model adalah sebagai berikut :

- Ukuran Gambar : Model dilatih menggunakan gambar berukuran 640 × 640 piksel untuk menyeimbangkan antara akurasi deteksi dan kecepatan proses.
- Batch Size : Pelatihan dilakukan dengan memproses 16 gambar sekaligus dalam setiap iterasi untuk efisiensi.
- Jumlah epoch : proses pelatihan berlangsung selama 100 pengulangan (*epoch*) agar model dapat mempelajari pola pada dataset secara maksimal.
- Lokasi Dataset : Berkas data.yaml digunakan untuk menentukan lokasi dataset serta konfigurasi kelas objek (Kacamata, Kartu Identitas, dan Masker).

Selama proses pelatihan, sistem memantau performa model, termasuk tingkat akurasi deteksi objek dan penurunan *error*. Dengan memanfaatkan GPU Tesla T4 pada Google Colab, pelatihan dapat dilakukan dengan lebih cepat meskipun jumlah dataset cukup besar. Setelah pelatihan selesai, diperoleh model dengan bobot terbaik (*best.pt*) yang mencapai performa terbaik pada data *validation*, yang selanjutnya digunakan pada sistem *smart mirror*.

4.2. Perancangan Sistem

Perancangan sistem *smart mirror* ini diimplementasikan menggunakan bahasa pemrograman Python dengan pendekatan pemrograman berorientasi objek (*Object-Oriented Programming*). Sistem dirancang secara modular, yaitu dipisahkan ke dalam beberapa berkas program berdasarkan fungsinya masing-masing, agar kode lebih terstruktur dan mudah dikembangkan.



Gambar 13. Gambar Perancangan Sistem

Adapun fungsi dari masing-masing modul program adalah sebagai berikut:

- `Config.py` : Menyimpan seluruh parameter konfigurasi sistem, seperti *path* model, ambang batas *confidence* per kelas objek, serta pengaturan suara dan tampilan.
- `detector.py` / `video_worker.py`: Bertugas memuat model YOLOv8 (*best.pt*) dan melakukan deteksi objek pada setiap *frame* citra dari kamera.
- `voice.py`: Menangani notifikasi suara (*text-to-speech*) menggunakan *library* `pyttsx3` secara *offline*.
- `controller.py`: Mengatur logika utama sistem, termasuk siklus pemeriksaan barang dan penghubung antar modul.
- `ui_mainwindow.py`: Mengatur tampilan antarmuka pengguna (GUI) berbasis `PyQt6`.
- `gui_app.py`: Berkas utama yang menjalankan keseluruhan aplikasi.

Tujuan dari perancangan modular ini adalah untuk memastikan sistem dapat berjalan dengan baik dan mudah dikembangkan lebih lanjut, termasuk kemungkinan migrasi ke perangkat lain seperti Mini PC yang digunakan pada sistem *smart mirror* ini.

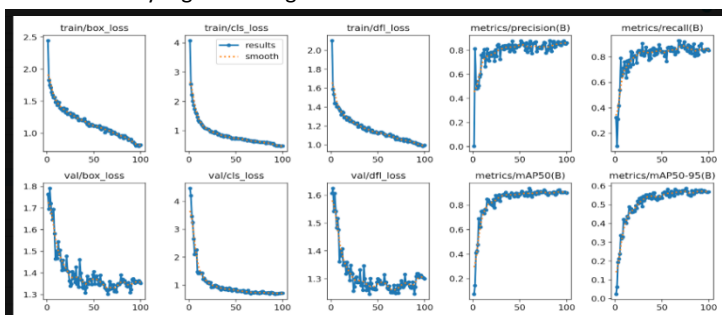
4.3. Hasil Evaluasi Model

Setelah proses pelatihan model YOLOv8 selesai, dilakukan evaluasi terhadap performa model menggunakan data *validation* yang telah dipisahkan pada tahap pelatihan. Evaluasi ini bertujuan untuk mengukur seberapa baik model dalam mendeteksi ketiga kelas objek, yaitu masker, kacamata, dan kartu identitas kerja. Metrik utama yang menjadi fokus evaluasi adalah *Precision*, karena metrik ini secara langsung menggambarkan tingkat kebenaran sistem dalam mendeteksi objek — semakin tinggi nilai *precision*, semakin dapat diandalkan sistem dalam memberikan notifikasi yang akurat kepada pengguna. Selain *precision*, ditampilkan pula nilai *recall* dan *mean Average Precision* (mAP) sebagai metrik pendukung untuk memberikan gambaran menyeluruh terkait performa model. Setelah proses pelatihan selesai, model dievaluasi menggunakan data *validation* untuk mengukur performanya. Evaluasi dilakukan berdasarkan metrik *precision*, *recall*, dan *mean Average Precision* (mAP).

Tabel 5. Hasil Evaluasi Model Keseluruhan

Matrik	Nilai
Precision	86,63%
Recall	87,62%
MAP50	92,59%

model mencapai nilai *precision* sebesar 86,63%, yang menunjukkan bahwa sebagian besar objek yang terdeteksi oleh model merupakan deteksi yang benar. Nilai *recall* sebesar 87,62% menunjukkan bahwa model mampu menemukan sebagian besar objek yang seharusnya terdeteksi. Nilai mAP50 sebesar 92,59% menunjukkan performa deteksi yang tergolong baik pada standar evaluasi yang umum digunakan.



Gambar 14. Grafik Hasil Pelatihan Model

Grafik hasil pelatihan model ditunjukkan pada Gambar 12. Grafik tersebut menampilkan perkembangan nilai *loss* dan metrik performa model selama 100 *epoch* pelatihan. nilai *loss* pada data *training* (*box_loss*, *cls_loss*, dan *dfl_loss*) mengalami penurunan yang konsisten seiring bertambahnya jumlah *epoch*. Penurunan serupa juga terjadi pada *loss* data *validation*, yang kemudian stabil pada nilai rendah tanpa mengalami kenaikan kembali. Hal ini menunjukkan bahwa model tidak mengalami *overfitting*, yaitu kondisi di mana model hanya menghafal data pelatihan namun gagal mengenali data baru. Sementara itu, nilai *precision*, *recall*, *mAP50*, dan *mAP50-95* menunjukkan peningkatan yang signifikan pada *epoch* awal, kemudian meningkat secara bertahap hingga mencapai kondisi stabil (*plateau*) di akhir pelatihan. Nilai *mAP50* mencapai sekitar 0,92 (92%), yang menunjukkan bahwa model memiliki kemampuan deteksi yang baik. Kondisi kurva yang telah stabil di akhir pelatihan menunjukkan bahwa jumlah 100 *epoch* sudah memadai untuk melatih model secara optimal.

Tabel 6. Hasil Evaluasi Precision Per Kelas Objek

Kelas	Precision
Masker	97,65%
Kacamata	80,26%
Kartu Identitas	74,21 %

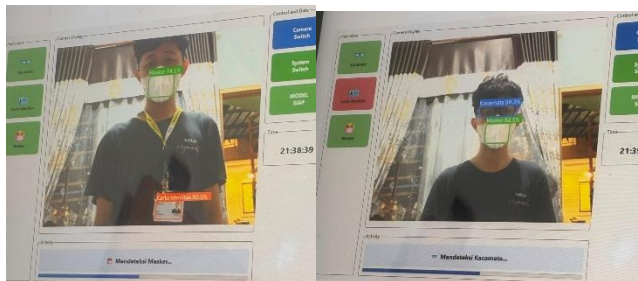
Berdasarkan Tabel 6, terdapat perbedaan tingkat presisi yang cukup signifikan antar kelas objek. Kelas Masker menunjukkan tingkat presisi tertinggi sebesar 97,65%, yang berarti hampir seluruh deteksi masker oleh sistem adalah benar. Hal ini disebabkan bentuk masker yang relatif seragam dan mudah dikenali oleh model, sehingga jarang terjadi kesalahan deteksi.

Kelas Kacamata memperoleh presisi sebesar 80,26%. Meskipun tergolong cukup baik, nilai ini lebih rendah dibandingkan Masker karena adanya variasi bentuk dan model kacamata yang beragam, sehingga sistem kadang mendeteksi objek lain sebagai kacamata (*false positive*).

Kelas Kartu Identitas memperoleh presisi paling rendah, yaitu sebesar 74,21%. Rendahnya presisi pada kelas ini disebabkan oleh permukaan kartu identitas yang bersifat reflektif, sehingga pada kondisi pencahayaan tertentu dapat menimbulkan pantulan cahaya yang membuat sistem salah mendeteksi objek lain sebagai kartu identitas. Temuan ini menjadi dasar bahwa kartu identitas merupakan objek paling menantang untuk dideteksi dalam sistem ini, dan menjadi salah satu bahan pengembangan untuk penelitian selanjutnya.

4.4. Hasil Pengujian Sistem

Pengujian sistem dilakukan secara *real-time* menggunakan kamera *webcam* yang terhubung ke Mini PC, untuk mengevaluasi kemampuan sistem dalam mendeteksi barang pribadi serta memberikan notifikasi kepada pengguna.



Gambar 15. Tampilan Sistem Saat Melakukan Deteksi

Berdasarkan Gambar 15, sistem berhasil mendeteksi objek yang dikenakan pengguna, ditandai dengan munculnya *bounding box* dan label pada tampilan kamera, serta perubahan warna indikator menjadi hijau untuk objek yang terdeteksi. Panel *Activity* juga menampilkan proses siklus pemeriksaan barang yang sedang berjalan, sesuai dengan tahapan deteksi masker dan kacamata pada saat pengujian.

Tabel 7. Hasil Pengujian Fungsional

No	Kondisi Pengujian	Hasil Deteksi	Notifikasi Suara	Keterangan
1	Seluruh Objek		Semua Barang Sudah Lengkap	Sesuai
2	Masker tidak tersedia		Jangan Lupa Memakai Masker	Sesuai
3	Kacamata tidak tersedia		Jangan Lupa Memakai Kacamata	Sesuai
4	Kartu Identitas tidak tersedia		Jangan Lupa Membawa Kartu Identitas	Sesuai
5	Seluruh Objek tidak tersedia		Jangan Lupa Membawa masker, kacamata dan kartu identitas	Sesuai

4.5. Hasil Pengujian Sistem Terhadap Pengguna

Pengujian sistem terhadap pengguna dilakukan untuk mengetahui konsistensi dan kemampuan generalisasi sistem dalam mendeteksi barang pribadi pada individu yang berbeda. Pengujian dilakukan terhadap tiga pengguna, yaitu Anjunius, Rafif, dan Fandi, yang masing-masing menggunakan sistem selama satu minggu (tujuh hari). Setiap hari, hasil deteksi tercatat secara otomatis pada berkas *log* sistem. Pengujian ini bertujuan untuk memastikan bahwa sistem mampu bekerja secara konsisten dalam penggunaan sehari-hari pada pengguna dengan karakteristik fisik yang berbeda-beda. Selain karakteristik fisik pengguna, kondisi lingkungan pengujian juga menjadi faktor yang perlu diperhatikan karena dapat memengaruhi performa deteksi sistem. Beberapa parameter yang diamati dalam pengujian ini meliputi lokasi pengujian, waktu pengujian, intensitas cahaya, serta jarak pengguna terhadap kamera. Intensitas cahaya diukur menggunakan aplikasi *lux meter* pada perangkat *smartphone*, sedangkan jarak pengguna diukur menggunakan meteran. Parameter pengujian pada masing-masing pengguna ditunjukkan pada Tabel 8.

Tabel 8. Parameter Pengujian Per Pengguna

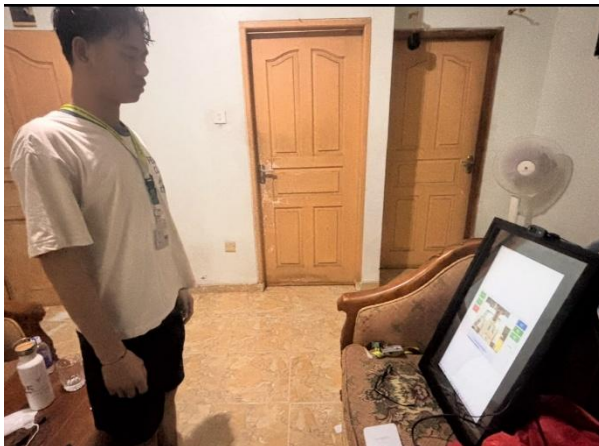
No	Pengguna	Lokasi	Waktu	Intesitas Cahaya (lux)	Jarak Ke kamera (cm)
1	Anjunius	Indoor (dalam rumah)	Pagi hari	221 (lx)	150 cm
2	Rafif	Indoor (dalam rumah)	Pagi hari	160 (lx)	150 cm
3	Fandi	Indoor (dalam rumah)	Malam hari	90 (lx)	160 cm

Berdasarkan Tabel 8, seluruh pengujian dilakukan pada kondisi *indoor* dengan sumber pencahayaan dari lampu ruangan, namun terdapat perbedaan pada waktu pengujian dan intensitas cahaya. Anjunius dan Rafif melakukan pengujian pada pagi hari dengan intensitas cahaya 221 lux dan 160 lux, sedangkan Fandi melakukan pengujian pada malam hari dengan intensitas cahaya 90 lux. Perbedaan intensitas cahaya ini disebabkan oleh perbedaan waktu pengujian dan lokasi ruangan masing-masing pengguna.

Berdasarkan Standar Nasional Indonesia (SNI 03-6197-2000) tentang konservasi energi pada sistem pencahayaan, intensitas cahaya untuk aktivitas dalam ruangan seperti ruang kerja umum minimal 200 lux, sedangkan untuk aktivitas sederhana seperti ruang tamu 120–150 lux. Dengan demikian, kondisi pengujian Anjunius (221 lux) telah memenuhi standar ruang kerja, kondisi Rafif (160 lux) berada pada standar ruang tamu, sedangkan kondisi Fandi (90 lux) berada di bawah standar dan tergolong redup.

Untuk parameter jarak, Anjunius dan Rafif melakukan pengujian pada jarak 150 cm dari kamera, sedangkan Fandi pada jarak 160 cm. Jarak pengujian yang relatif sama ini bertujuan untuk memastikan agar posisi seluruh objek (masker, kacamata, dan kartu identitas kerja) tetap masuk ke dalam bingkai kamera secara utuh.

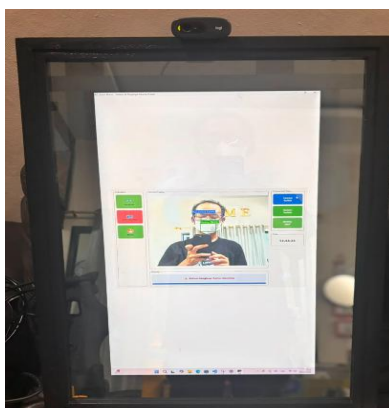
Hasil pengujian pada masing-masing pengguna menunjukkan pola yang menarik untuk dianalisis. Pengguna Fandi yang diuji pada kondisi pencahayaan paling rendah (90 lux) justru memperoleh akurasi tertinggi sebesar 85,71%, sedangkan Anjunius dengan pencahayaan tertinggi (221 lux) memperoleh akurasi 71,43%, dan Rafif dengan pencahayaan menengah (160 lux) memperoleh akurasi terendah sebesar 66,67%. Pola ini menunjukkan bahwa intensitas cahaya bukan satu-satunya faktor yang memengaruhi kinerja sistem.



Gambar 16. Contoh Kesalahan Deteksi (False Positive)

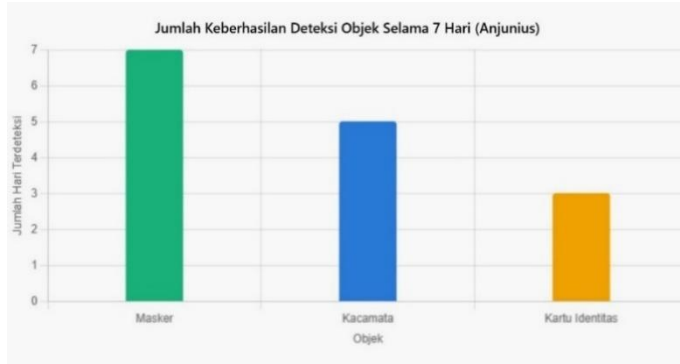
Selama pengujian, ditemukan beberapa kasus *false positive* pada objek kacamata, di mana sistem mendeteksi kacamata meskipun pengguna tidak memakainya. Kesalahan ini biasanya terjadi ketika pengguna berada pada jarak yang cukup jauh dari kamera (di atas 200 cm), sehingga model salah mengenali area sekitar mata sebagai kacamata. Fenomena ini sejalan dengan nilai *precision* kacamata sebesar 80,26% yang menunjukkan bahwa sekitar 20% deteksi kacamata bukan merupakan deteksi yang benar.

4.5.1 Pengujian sistem terhadap pengguna Anjunius



Gambar 17. Pengujian sistem terhadap pengguna Anjunius

Gambar 17, menunjukkan proses pengujian sistem Smart Mirror terhadap pengguna bernama Anjunius. Pada tahap ini, pengguna berdiri di depan Smart Mirror untuk melakukan pengujian kemampuan sistem dalam mendeteksi tiga objek, yaitu kacamata, masker, dan kartu identitas menggunakan model YOLOv8 yang telah dilatih. Hasil deteksi ditampilkan secara langsung pada layar Smart Mirror dalam bentuk bounding box dan label pada setiap objek yang berhasil dikenali. Pengujian ini bertujuan untuk memastikan bahwa sistem dapat mengenali barang-barang pribadi pengguna sesuai dengan kondisi sebenarnya.



Gambar 18. Grafik Hasil Pengujian Anjunius

Berdasarkan Gambar 18, dari tujuh hari pemakaian sistem oleh Pengguna Anjunius, objek Masker berhasil terdeteksi sebanyak 7 kali, Kacamata sebanyak 5 kali, dan Kartu Identitas sebanyak 3 kali. Tingkat keberhasilan deteksi keseluruhan dapat dihitung sebagai berikut:

$$Akurasi = \frac{Jumlah\ Deteksi\ Keberhasilan\ (15)}{Jumlah\ total\ Percobaan\ (21)} \times 100\% = 71,43\% \quad (2.4)$$

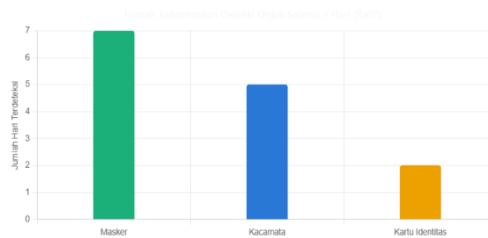
Nilai tersebut diperoleh dari 15 deteksi berhasil dari total 21 percobaan (7 hari × 3 objek). Objek Masker memiliki tingkat keberhasilan tertinggi karena bentuknya yang relatif seragam dan mudah dikenali oleh model. Sebaliknya, Kartu Identitas memiliki tingkat keberhasilan paling rendah, yang disebabkan oleh permukaannya yang bersifat reflektif sehingga pada kondisi pencahayaan tertentu dapat mengurangi kejelasan objek saat dideteksi. Meskipun demikian, sistem tetap berfungsi sesuai tujuannya, yaitu memberikan notifikasi pengingat ketika salah satu barang tidak terdeteksi.

4.5.2 Pengujian sistem terhadap pengguna Rafif



Gambar 19. Pengujian sistem terhadap pengguna Rafif

Gambar 19, menunjukkan proses pengujian sistem Smart Mirror terhadap pengguna bernama Rafif. Pengujian dilakukan untuk mengetahui kemampuan sistem dalam mendeteksi objek pada pengguna yang berbeda. Pada tahap ini, Rafif berada di depan Smart Mirror sambil mengenakan barang-barang pribadi yang menjadi objek deteksi, yaitu kacamata, masker, dan kartu identitas. Kamera kemudian menangkap citra pengguna dan model YOLOv8 melakukan proses deteksi terhadap ketiga objek tersebut secara real-time. Hasil deteksi ditampilkan pada layar Smart Mirror dalam bentuk bounding box dan label objek yang berhasil dikenali. Berdasarkan hasil pengujian, sistem mampu mendeteksi objek yang digunakan oleh Rafif dengan baik sehingga menunjukkan bahwa model tidak hanya bekerja pada satu pengguna, tetapi juga dapat mengenali objek pada pengguna lain.



Gambar 20. Grafik Hasil Pengujian Rafif

Berdasarkan Gambar 20, dari tujuh hari pemakaian sistem oleh Pengguna Rafif, objek Masker berhasil terdeteksi sebanyak 7 kali, Kacamata sebanyak 5 kali, dan Kartu Identitas sebanyak 2 kali. Tingkat keberhasilan deteksi keseluruhan dihitung sebagai berikut:

$$Akurasi = \frac{Jumlah\ Deteksi\ Keberhasilan\ (14)}{Jumlah\ total\ Percobaan\ (21)} \times 100\% = 66,67\% \quad (2.5)$$

Berdasarkan hasil pengujian selama tujuh hari, sistem berhasil mendeteksi objek Masker sebanyak 7 kali, Kacamata sebanyak 5 kali, dan Kartu Identitas sebanyak 2 kali, dengan tingkat akurasi keseluruhan sebesar 66,67%. Objek Masker menunjukkan performa deteksi paling stabil, sedangkan Kartu Identitas memiliki tingkat keberhasilan paling rendah akibat permukaannya yang bersifat reflektif. Pola hasil ini konsisten dengan pengujian pada Pengguna Anjunius, yang menunjukkan bahwa performa sistem relatif stabil dan tidak bergantung pada individu tertentu. Selain itu, sistem terbukti berfungsi sesuai tujuannya, yaitu memberikan notifikasi pengingat setiap kali terdapat barang yang tidak terdeteksi.

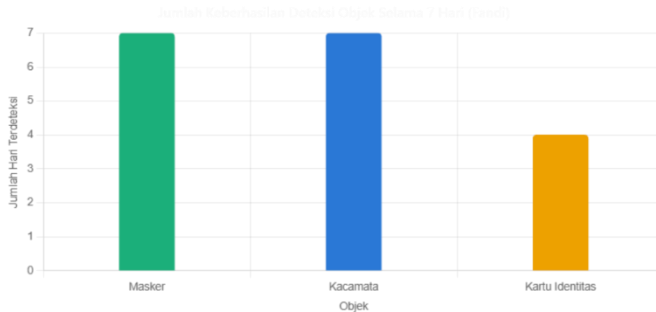
4.5.3 Pengujian sistem terhadap pengguna Fandi



Gambar 21. Pengujian sistem terhadap pengguna Fandi

Gambar 21, menunjukkan proses pengujian sistem Smart Mirror terhadap pengguna bernama Fandi. Pengujian dilakukan dengan meminta pengguna berdiri di depan Smart Mirror sambil mengenakan barang-barang pribadi yang menjadi objek deteksi, yaitu kacamata, masker, dan kartu identitas. Kamera kemudian mengambil citra pengguna, selanjutnya model YOLOv8 melakukan

proses deteksi terhadap objek-objek tersebut secara real-time. Hasil deteksi ditampilkan pada layar Smart Mirror dalam bentuk bounding box dan label kelas pada setiap objek yang berhasil dikenali. Berdasarkan hasil pengujian, sistem mampu mendeteksi objek yang dikenakan oleh pengguna dengan baik sehingga menunjukkan bahwa model dapat bekerja pada pengguna yang berbeda secara konsisten.



Gambar 22. Grafik Hasil Pengujian Fandi

Berdasarkan Gambar 22, dari tujuh hari pemakaian sistem oleh Pengguna Fandi, objek Masker dan Kacamata masing-masing berhasil terdeteksi sebanyak 7 kali, sedangkan Kartu Identitas sebanyak 4 kali. Tingkat keberhasilan deteksi keseluruhan dihitung sebagai berikut:

$$Akurasi = \frac{\text{Jumlah Deteksi Keberhasilan (18)}}{\text{Jumlah total Percobaan (21)}} \times 100\% = 85,71\% \quad (2.5)$$

Pengguna Fandi memperoleh tingkat akurasi tertinggi di antara ketiga pengguna, dengan objek Masker dan Kacamata yang terdeteksi secara sempurna. Meskipun demikian, Kartu Identitas tetap menjadi objek dengan tingkat keberhasilan paling rendah, yang kembali menegaskan bahwa objek dengan permukaan reflektif lebih menantang untuk dideteksi secara konsisten.

Bab 5. Kesimpulan dan Saran

5.1. Kesimpulan

Berdasarkan hasil penelitian, perancangan, dan pengujian yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Sistem *smart mirror* berbasis YOLOv8 berhasil dirancang dan diimplementasikan untuk mendeteksi barang pribadi berupa masker, kacamata, dan kartu identitas kerja secara *real-time*, serta memberikan notifikasi suara kepada pengguna apabila salah satu barang tidak terdeteksi.
2. Model YOLOv8 yang dilatih menggunakan dataset yang dikumpulkan dan dianotasi secara mandiri memperoleh performa yang baik, dengan tingkat presisi (*precision*) sebesar 86,63% secara keseluruhan. Adapun presisi per kelas objek adalah masker sebesar 97,65%, kacamata sebesar 80,26%, dan kartu identitas sebesar 74,21%.
3. Hasil pengujian terhadap tiga pengguna yang berbeda (Anjunius, Rafif, dan Fandi) selama satu minggu menunjukkan tingkat akurasi deteksi rata-rata sebesar 74,60%. Selain karakteristik fisik pengguna, kondisi lingkungan seperti intensitas cahaya dan jarak pengguna terhadap kamera juga memengaruhi hasil deteksi.
4. Sistem terbukti mampu bekerja secara konsisten pada pengguna dengan karakteristik fisik yang berbeda-beda, serta berhasil menjalankan fungsi utamanya, yaitu mengingatkan pengguna terhadap kelengkapan barang pribadi sebelum beraktivitas.

5.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran untuk pengembangan sistem di masa mendatang, yaitu:

1. Perlu dilakukan penambahan jumlah dan variasi dataset, khususnya untuk objek kartu identitas kerja, guna meningkatkan akurasi deteksi pada objek dengan permukaan reflektif.
2. Sistem dapat dikembangkan dengan menambahkan pengaturan pencahayaan yang lebih optimal atau teknik pengolahan citra tambahan untuk mengurangi pengaruh pantulan cahaya pada objek reflektif.
3. Sistem dapat dikembangkan agar mampu mendeteksi lebih dari satu pengguna secara bersamaan (*multi-user detection*) untuk penggunaan pada lingkungan yang lebih luas.
4. Penambahan jenis barang pribadi lain yang dapat dideteksi dapat dipertimbangkan agar sistem lebih fleksibel dan sesuai dengan kebutuhan pengguna yang beragam.

Daftar Pustaka

- [1] D. K. J, D. P. A. Vijaya and N. Awasthi, "Design and Implementation of Smart Mirror as a Personal Assistant using Raspberru Pi," *International Research Journal of Engineering and Technology (IRJET)*, vol. 05, no. 05, pp. 438-441, 2018.
- [2] N. Ramsari and A. Rifaldi, "Rancang Bangun aplikasi Penjadwalan Kegiatan Akademik Disertai Sistem Reminder Berbasis Responsive Web Design," *Fiki*, vol. IX, no. 1, pp. 1-11, 2018.
- [3] Ilham, Andi, Mutmainnah Muctar, and Jayanti Yusmah Sari. 2024. "Mask Detection Using the YOLO (You Only Look Once) Method." *Media Informasi Teknologi* 1(1):1-12.
- [4] A.F. Fhatiroy, "Penggunaan Algoritma YOLOv8 dalam Mendeteksi Keberadaan Peternak pada Area Peternakan Ayam," *Univ.Muhammadiyah Makassar*, 2024.
- [5] S. Wijaya and Sholihin, "Analisis Rancang Bangun Pada Sistem Pendeteksi Wajah Untuk Keamanan Otomatis Berbasis IoT," *J. Inform. MULTI*. Vol.1, no. 4, pp. 362-371, 2023.
- [6] Dhamanigi, Joshua Roshan, et all. (2017). "Smart Mirror - A Home Automation System Implemented Using Ambient Artificial Intelligence". Bangalore, India, Vol. 6, Issue 7, July 2017.
- [7] T. Susim and C. Darujati, "Pengolahan Citra untuk Pengenalan Wajah (Face Recognition) Menggunakan OpenCV," *J. Syntax Admiration*, vol.2, no.3, pp.534-545, 2021, doi:10.46799/jsa. V 2i3.202.
- [8] A. S. Handayani, S. Soim, T. E. Agusdi, Rumiasih, and A. Nurdin, "Klasifikasi Kualitas Udara Dengan Metode Support Vector Machine," *JIRE (Jurnal Inform. Rekayasa Elektron.*, vol. 3, no. 2, pp. 187-199, 2020.
- [9] Q. Zhang, Q. Yang, X. Zhang, Q. Bao, J. Su, and X. Liu, "Waste image classification based on transfer learning and convolutional neural network," *Waste Manag.*, vol. 135, no. May, pp. 150-157, 2021, doi: 10.1016/j.wasman.2021.08.038.
- [10] B. P. Nugroho, Y. Prihati, and S. T. Galih, "Implementasi Algoritma Yolo V5 Dalam Rancangan Aplikasi Pendeteksi Plat Nomor Kendaraan," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 7, no. 3, pp. 851-859, 2024, doi: 10.31539/intecom. V 7i3.10376.
- [11] B. D. Prasetya, E. P. Amalia, and P. D. Juliana, "Implementasi Object Detection Berbasis Web untuk Klasifikasi Jenis Sampah Menggunakan YOLOv8," vol. 4, pp. 60-65, 2025.
- [12] M. S. Nafiz, S. S. Das, M. K. Morol, A. Al Juabir, and D. Nandi, "ConvoWaste: An Automatic Waste Segregation Machine Using Deep Learning," *Int. Conf. Robot. Electr. Signal Process. Tech.*, vol. 2023-Janua, pp. 181-186, 2023, doi: 10.1109/ICREST57604.2023.10070078.
- [13] H. Suraya, I. Ziad, J. T. Elektro, F. Teknik, and D. Telekomunikasi, "Rancang Bangun Alat Pendeteksi Kantuk Pada Mobil Berbasis IoT Menggunakan Raspberry Pi Dan Kamera," *J. Ilm. Komputasi*, vol. 20, no. 3, pp. 385-391, 2021, doi: 10.32409/jikstik.20.3.2797.
- [14] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. da Silva, "A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit," *Electronics*, vol. 10, no. 3, p. 279, 2021.

- [15] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.
- [16] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2022.
- [17] A. F. Oklilas, Sukemi, and R. Apriliyanto, "Model YOLO Versi 4 pada Pengenalan Kendaraan di Jalan Raya Kota Palembang," *Transmisi: Jurnal Ilmiah Teknik Elektro*, vol. 25, no. 3, pp. 136–139, 2023.
- [18] A. I. Pradana, H. Harsanto, and W. Wijiyanto, "Deteksi Rambu Lalu Lintas Real-Time di Indonesia dengan Penerapan YOLOv11: Solusi untuk Keamanan Berkendara," *Jurnal Algoritma*, vol. 21, no. 2, pp. 145–155, 2024.
- [19] D. A. Prabowo, A. Nugroho, dan R. R. Isnanto, "Implementasi Algoritma YOLOv8 untuk Deteksi Objek pada Citra Digital," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 2, pp. 345–353, 2024.
- [20] M. R. Hidayat, A. Wibowo, dan E. Prasetyo, "Implementasi Deep Learning Menggunakan YOLOv8 untuk Deteksi Objek Secara Real-Time," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 11, no. 1, pp. 55–64, 2024.

Biodata

Pas Foto
2 x 3

Nama : Anjunius M
TTL : Batam, 25 Januari 2001
Agama : Kristen
Alamat : Perum.Nusa Indah blok d6 No.05

Email : anjuniusmanurung@gmail.com
Riwayat Pendidikan SMA/SMK : SMKN 3 BATAM
SMP : SMPN 16 BATAM

Lampiran

