

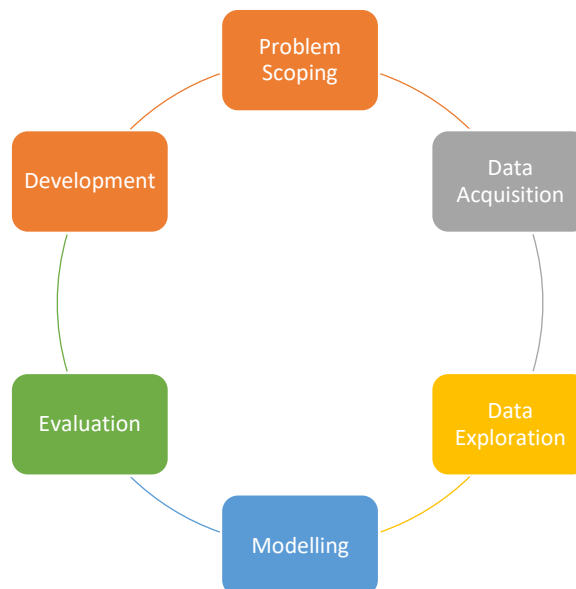
BAB III

METODE PENELITIAN

Dalam bab ini, penulis membagi metode menjadi dua bagian utama. Pertama, metode pengembangan sistem deteksi objek menggunakan *AI Project Cycle*, yang melibatkan tahapan mulai dari *problem scoping* hingga *deployment*. Kedua, metode pengujian sistem deteksi objek pada kategori objek, kondisi ruangan, dan jarak yang berbeda. Pengujian ini bertujuan mengevaluasi kinerja sistem di berbagai skenario nyata, termasuk pengaruh pencahayaan dan variasi jarak, untuk memastikan efektivitas sistem dalam situasi yang beragam.

3.1 Metode dan Desain Pengembangan Sistem Deteksi Objek

Penelitian ini mengimplementasikan *AI Project Cycle* sebagai pendekatan metodologi. *AI Project Cycle* merupakan serangkaian tahapan proses yang digunakan dalam pengembangan proyek kecerdasan buatan secara menyeluruh. Metode ini terstruktur untuk memastikan pencapaian hasil yang diharapkan, termasuk dalam tahap-tahap seperti *problem scoping*, *data acquisition*, *data exploration*, *modelling*, *evaluation*, dan *development* (Ariza & Baez, 2022).

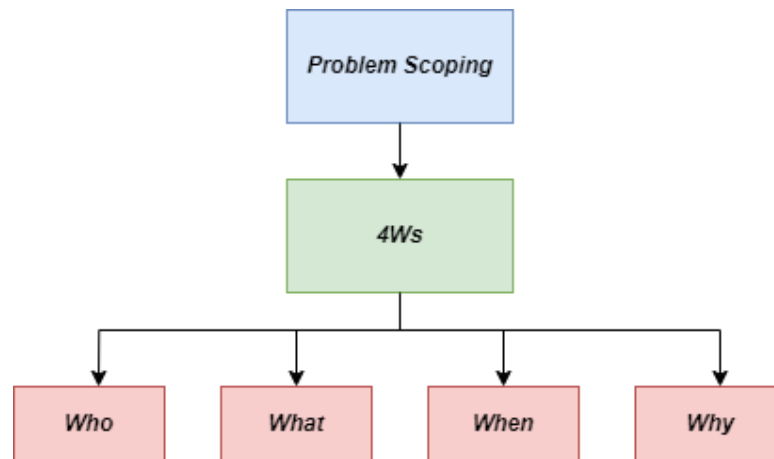


Gambar 3. 1 Proses Metode *AI Project Cycle*

Sumber: Dokumentasi Pribadi

3.1.1 Problem Scoping

Problem scoping adalah langkah awal dalam *AI Project Cycle* yang berfungsi untuk mengidentifikasi masalah dan memetakan batasan masalah yang ingin diselesaikan. Tujuan utama dari *problem scoping* adalah untuk membuat tujuan atau target menjadi jelas dan terarah sehingga lebih mudah untuk menemukan solusi. Tahapan ini membantu dalam menyelesaikan masalah dengan menggunakan pendekatan pemetaan menggunakan konsep *4W*. Berikut ini adalah detailnya:



Gambar 3. 2 Diagram *4W* dari *Problem Solving*

Sumber: Dokumentasi Pribadi

Pada gambar 3.2 mengilustrasikan konsep *4W* dalam tahap *problem scoping*. Pendekatan menggunakan *4W* ini bermanfaat dalam mengidentifikasi masalah yang akan dianalisis serta merumuskan solusinya. Indikator *Who* menerangkan subjek dalam masalah, *What* menguraikan masalah utama yang dihadapi serta faktor-faktor yang mendukung terjadinya masalah tersebut, *When* mengidentifikasi kapan dan saat apa masalah tersebut hadir, *Why* menjelaskan mengapa masalah tersebut perlu diselesaikan dan apa manfaatnya.

Penulis merumuskan *4W* untuk membantu dalam memahami dan mengidentifikasi masalah yang ada serta menemukan solusinya. Konsep *4W* digunakan sebagai kerangka kerja dalam penyusun masalah deteksi

penggunaan helm, topi, dan kacamata hitam di ruangan ATM menggunakan metode YOLO. Berikut adalah tabel yang merangkum *4W* dari *problem scoping* ini:

Tabel 3. 1 *Problem Statement 4W of Problem Solving*

<i>4W</i>	Keterangan
<i>What</i>	Pengembangan sistem deteksi objek penutup wajah seperti helm, topi, dan kacamata hitam yang dilarang digunakan di ruangan ATM.
<i>Why</i>	Masalah ini penting karena penggunaan benda-benda tersebut di ruangan ATM dapat menimbulkan risiko keamanan.
<i>Who</i>	Orang yang menggunakan helm, topi, dan kacamata hitam.
<i>When</i>	Pada saat orang menggunakan helm, topi, dan kacamata hitam.

Tabel 3.1 menjabarkan *problem statement* dari permasalahan yang dirumuskan menggunakan pendekatan *4W* dalam *problem scoping*. Permasalahan yang diteliti melibatkan pengembangan sistem deteksi untuk objek-objek yang dilarang digunakan di ruangan ATM, seperti helm, topi, dan kacamata hitam, karena penggunaan objek-objek ini dapat menimbulkan risiko keamanan yang signifikan. Tujuan dari sistem ini adalah untuk mengidentifikasi secara efektif dan mencegah penggunaan helm, topi, dan kacamata hitam guna mengurangi potensi ancaman terhadap keamanan. Deteksi dilakukan saat orang menggunakan objek-objek tersebut di dalam ruangan ATM.

3.1.2 *Data Acquisition*

Dalam tahap ini, penulis fokus pada pengumpulan dataset yang diperlukan untuk pengembangan sistem deteksi. Pada penelitian ini, dataset yang dikumpulkan berupa gambar-gambar dari tiga kategori objek yaitu helm, topi, dan kacamata hitam. Untuk objek topi terdapat dua kelas yaitu

yaitu *topi1* dan *topi2*. Kelas *topi1* berupa *baseball cap* dan *topi2* berupa *topi bundar*. Pembagian kelas *topi* tersebut dimaksudkan untuk memberikan variasi pada kelas *topi*. Pengumpulan dataset objek yaitu sebanyak 400 gambar pada setiap kelas. Untuk mengumpulkan gambar-gambar tersebut, penulis menggunakan Bing Image Search API.

Bing Image Search API adalah layanan dari Microsoft yang memungkinkan pengguna untuk mencari dan mengakses gambar dari internet melalui mesin pencari Bing. Pengguna dapat mencari gambar dengan kata kunci dan menyesuaikan parameter seperti ukuran gambar. API ini memberikan hasil berupa URL gambar, deskripsi, dan metadata terkait. Penggunaan Bing Image Search API sangat berguna dalam pengembangan sistem deteksi untuk penelitian ini.

3.1.3 Data Exploration

Pada tahap ini, dilakukan eksplorasi terhadap dataset yang telah dikumpulkan guna mempersiapkannya untuk proses pelatihan model deteksi. Untuk memastikan dataset dalam kondisi yang optimal sebelum digunakan, dilakukan proses yang mencakup anotasi data, *preprocessing data*, dan augmentasi data.

3.1.3.1 Anotasi Data

Langkah pertama dalam *data exploration* adalah melakukan anotasi data. Ini berarti menandai atau memberi label data dengan informasi yang relevan, seperti menandai lokasi dan jenis objek dalam gambar. Anotasi data dilakukan pada *platform* Roboflow, yang menyediakan alat dan fitur untuk menandai objek-objek yang ada dalam gambar. Proses anotasi ini sangat penting karena model deteksi objek memerlukan data latih yang sudah dianotasi dengan benar untuk belajar mengenali dan mendeteksi objek yang diinginkan. Setiap gambar dalam dataset diberi anotasi dengan *bounding box* yang menandai posisi dan jenis objek (*helm*, *kacamata hitam*, *topi1* dan *topi2*).

3.1.3.2 *Preprocessing Data*

Preprocessing data adalah tahap penting untuk memastikan bahwa data siap digunakan dalam pelatihan model. Dalam langkah ini, semua gambar diterapkan *auto-orient* untuk memastikan orientasinya benar. Hal ini dilakukan karena beberapa gambar mungkin memiliki metadata orientasi yang menyebabkan gambar tidak ditampilkan dengan benar. Selain itu, semua gambar diubah ukurannya menjadi 640 x 640 piksel, sesuai dengan persyaratan model YOLOv8. Ukuran ini dipilih untuk memastikan model dapat bekerja secara optimal, karena model deteksi objek sering memerlukan input dengan dimensi tertentu.

3.1.3.3 *Augmentasi Data*

Augmentasi data bertujuan untuk meningkatkan kemampuan generalisasi model dengan memperkaya variasi dataset. Beberapa teknik augmentasi diterapkan, seperti *flipping* gambar secara horizontal atau vertikal untuk membantu model mengenali objek dari berbagai sudut pandang. Teknik *cropping* juga digunakan untuk memungkinkan model belajar mengenali objek meskipun hanya sebagian yang terlihat. Selain itu, *shear* digunakan untuk mengubah bentuk objek dalam gambar, yang membantu model dalam mengenali objek dari perspektif yang berbeda. Teknik lain termasuk mengubah gambar menjadi skala abu-abu atau *grayscale*, yang membantu model fokus pada bentuk dan tekstur tanpa tergantung pada warna. Selain itu, penyesuaian kecerahan atau *brightness* dilakukan untuk memastikan model dapat mengenali objek dalam berbagai kondisi pencahayaan. Semua teknik ini bertujuan untuk memperluas kemampuan model dalam menghadapi situasi dunia nyata yang bervariasi.

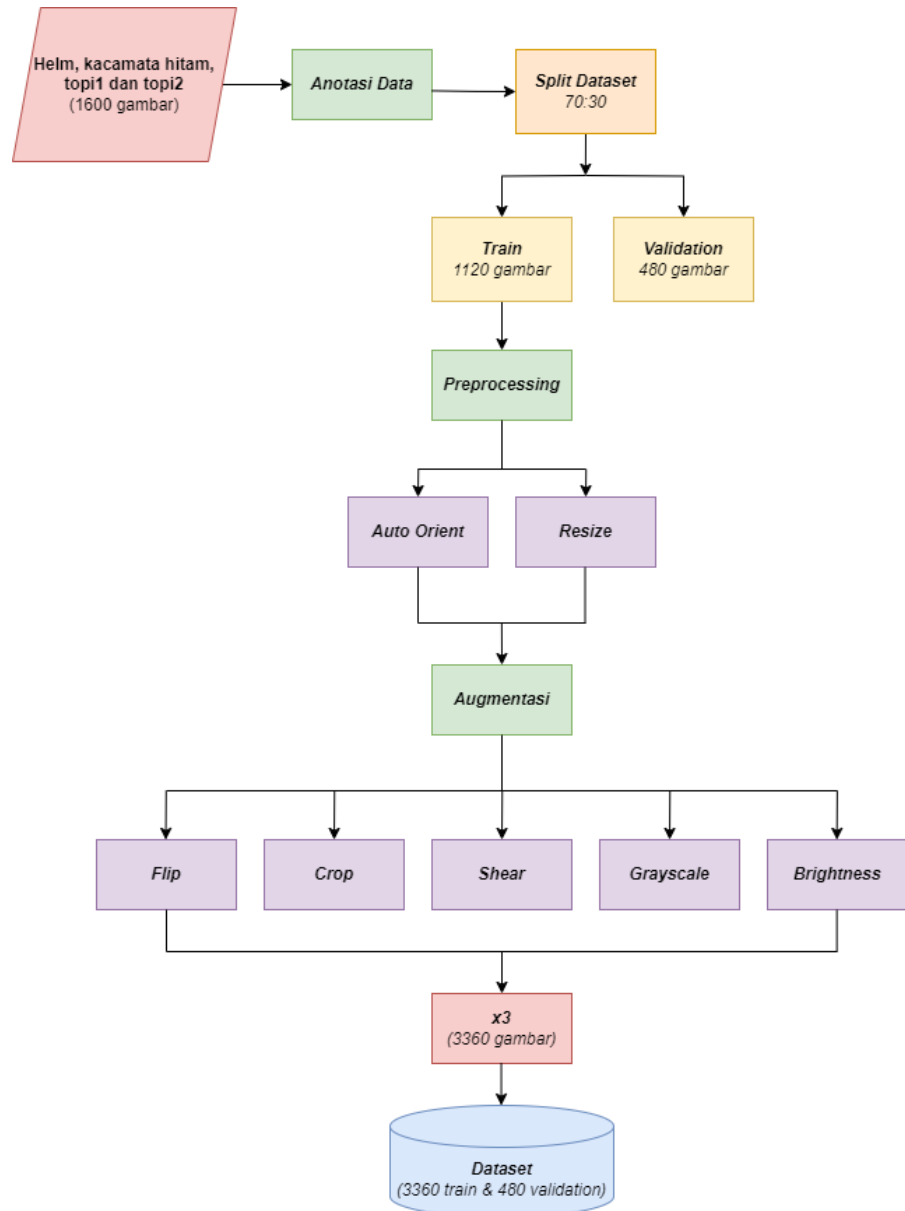
3.1.3.4 *Hasil Anotasi, Preprocessing, dan Augmentasi Data*

Dataset yang telah dilakukan anotasi, *preprocessing*, dan augmentasi pada awalnya terdiri dari 1.600 gambar, dengan masing-masing kelas (helm, kacamata hitam, topi1 dan topi2) memiliki 400

gambar. Setelah proses anotasi selesai, dataset ini kemudian dibagi menjadi dua bagian, yaitu data *train* dan *data validation*, dengan perbandingan 70:30. Hal ini menghasilkan 1.120 gambar untuk data *train* dan 480 gambar untuk data *validation*.

Selanjutnya, data *train* yang terdiri dari 1.120 gambar ini dilakukan *preprocessing* untuk memastikan bahwa gambar-gambar tersebut dalam kondisi optimal sebelum masuk ke tahap *training* model. Dan untuk meningkatkan kemampuan generalisasi model dan memperkaya variasi dataset, dilakukan augmentasi data pada data *train*. Proses augmentasi ini penting untuk membantu model mengenali objek dari berbagai sudut pandang, kondisi pencahayaan, dan perspektif yang berbeda.

Setelah proses augmentasi, data *train* yang awalnya terdiri dari 1.120 gambar dikalikan tiga kali lipat, sehingga jumlah gambar data *train* menjadi 3.360. Dengan demikian, total dataset keseluruhan menjadi 3.840 gambar, yang terdiri dari 3.360 gambar untuk data *train* dan 480 gambar untuk data *validation*. Adapun alur dari *data exploration* dijelaskan pada gambar berikut ini:



Gambar 3. 3 Alur dari Proses *Data Exploration*

Sumber: Dokumentasi Pribadi

3.1.4 Modelling

Pada sub bab ini, dijelaskan tahap pemodelan dalam *AI Project Cycle*. Pemodelan merupakan tahap yang sangat penting karena di sinilah algoritma atau model pembelajaran mesin diterapkan untuk menyelesaikan permasalahan yang telah dirumuskan sebelumnya. Untuk model sistem deteksi penggunaan helm, topi, dan kacamata hitam di ruangan ATM, dipilih model *You Only Look Once* (YOLO) versi 8. YOLOv8 adalah salah satu

algoritma deteksi objek yang populer dan efisien. Algoritma ini mampu melakukan deteksi objek dengan akurasi yang tinggi. YOLOv8 bekerja dengan cara membagi gambar menjadi *grid*, kemudian memprediksi *bounding boxes* dan kelas untuk setiap *grid cell* secara langsung dalam satu pass melalui jaringan neural. Ini membuat YOLOv8 sangat cepat dibandingkan metode deteksi objek lainnya yang memerlukan beberapa langkah terpisah untuk mendeteksi dan mengklasifikasikan objek.

Proses pemodelan dilakukan dengan menggunakan *platform* Kaggle dan model YOLOv8 dari Ultralytics. Kaggle menyediakan lingkungan yang ideal untuk melatih model karena memiliki infrastruktur komputasi yang kuat. Ultralytics menyediakan implementasi YOLOv8 yang mudah digunakan dan telah dioptimalkan, sehingga mempermudah dalam melakukan *training* dan evaluasi model.

3.1.5 Evaluation

Tahap evaluasi bertujuan untuk mengukur kinerja model yang telah dilatih. Pada tahap ini, model yang telah melalui proses pelatihan diujikan menggunakan dataset pengujian yang telah disiapkan sebelumnya. Evaluasi dilakukan untuk memastikan bahwa model mampu mendeteksi objek dengan akurasi tinggi pada data yang belum pernah dilihat sebelumnya.

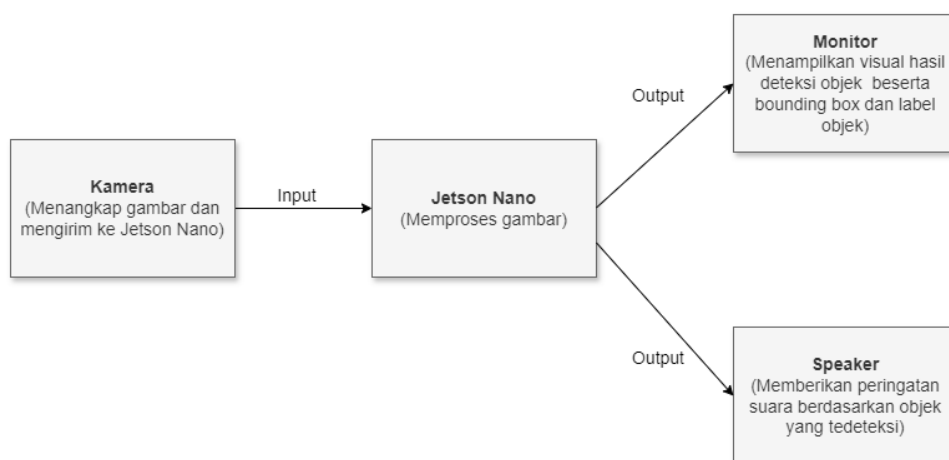
Beberapa metrik evaluasi yang umum digunakan dalam deteksi objek adalah *accuracy*, *precision*, *recall*, dan *F1-Score*. Pada tahap evaluasi ini, model YOLOv8 yang telah dilatih akan diujikan pada dataset pengujian yang terdiri dari 3360 gambar. Setiap prediksi model akan dibandingkan dengan anotasi *ground truth* untuk menghitung metrik-metrik evaluasi tersebut. Jika hasil evaluasi menunjukkan kinerja yang kurang memuaskan, proses *fine-tuning* dilakukan dengan menyesuaikan parameter model atau menambahkan data pelatihan yang lebih beragam. Tujuannya adalah untuk meningkatkan akurasi model hingga mencapai tingkat yang diinginkan.

Tahap evaluasi ini sangat penting karena hasilnya akan menentukan apakah model siap untuk diimplementasikan dalam aplikasi nyata atau perlu

dilakukan perbaikan lebih lanjut. Melalui evaluasi yang tepat, kita dapat memastikan bahwa model deteksi yang dikembangkan mampu beroperasi dengan baik dan memberikan hasil yang akurat dalam kondisi yang sebenarnya.

3.1.6 Deployment

Tahap *deployment* atau pengembangan sistem deteksi objek dilakukan pada perangkat Jetson Nano. Jetson Nano dipilih karena kemampuannya untuk menjalankan aplikasi AI secara efisien dengan daya komputasi yang cukup tinggi dalam bentuk yang kecil dan hemat energi. Perangkat ini dilengkapi dengan GPU NVIDIA yang mendukung inferensi AI yang memadai, menjadikannya pilihan ideal untuk aplikasi deteksi objek di lingkungan yang memerlukan respon cepat. Pengembangan sistem ini ditunjukkan pada gambar 3.4 yang menjelaskan mengenai blok diagram sistem.



Gambar 3. 4 Blok Diagram Sistem

Blok diagram diatas menunjukkan komponen utama yang terlibat dalam proses deteksi dan aliran data antar komponen. Komponen-komponen tersebut meliputi:

1. Kamera: Kamera digunakan untuk menangkap gambar dan mengirimkannya ke Jetson Nano. Kamera ini harus memiliki resolusi yang cukup tinggi dan kemampuan untuk menangkap

gambar dalam berbagai kondisi pencahayaan untuk memastikan deteksi yang akurat.

2. *Jetson Nano*: *Jetson Nano* adalah inti dari sistem ini, yang memproses gambar menggunakan model YOLO yang telah dilatih sebelumnya. *Jetson Nano* menjalankan model YOLO untuk mendeteksi objek yang relevan seperti helm, kacamata hitam, dan topi. Setelah deteksi dilakukan, informasi mengenai objek yang terdeteksi dikirim ke komponen berikutnya.
3. *Monitor*: *Monitor* digunakan untuk menampilkan visual hasil deteksi objek. Hasil deteksi objek ini termasuk gambar yang diambil oleh kamera dengan *bounding box* dan label objek yang terdeteksi. *Monitor* memberikan visualisasi yang jelas kepada pengguna mengenai objek apa saja yang terdeteksi oleh sistem.
4. *Speaker*: *Speaker* digunakan untuk memberikan peringatan suara berdasarkan objek yang terdeteksi. Misalnya, jika sebuah helm terdeteksi, sistem dapat memberikan peringatan suara untuk menginformasikan pengguna mengenai keberadaan helm tersebut. Fungsi ini sangat penting dalam aplikasi keamanan untuk memberikan respons cepat terhadap deteksi objek tertentu.

Setelah mengetahui blok diagram sistem deteksi, langkah selanjutnya adalah menjelaskan langkah-langkah dalam pengembangan model deteksi objek pada *Jetson Nano*:

1. *Set Up Jetson Nano*

Langkah pertama adalah *set up* atau menyiapkan lingkungan pengembangan pada *Jetson Nano*. Ini melibatkan instalasi sistem operasi yang disediakan oleh NVIDIA, yang mencakup driver GPU, *library* CUDA, dan toolkit pengembangan lainnya yang diperlukan untuk menjalankan model sistem deteksi.

2. *Transfer Model ke Jetson Nano*

Model YOLO yang telah dilatih dan dievaluasi dipindahkan ke *Jetson Nano*. Proses ini meliputi konversi model ke format yang

dioptimalkan untuk eksekusi di Jetson Nano, yaitu ONNX untuk meningkatkan kinerja inferensi.

3. Implementasi Kode Deteksi

Kode deteksi dikembangkan dan diimplementasikan pada Jetson Nano. Ini termasuk integrasi model YOLO dengan kamera atau sumber video yang digunakan untuk mendeteksi helm, kacamata hitam, dan topi. Kode ini mengatur proses pengambilan gambar, pra-pemrosesan, inferensi, dan tampilan hasil deteksi. Proses ini melibatkan penggunaan Jetson Nano sebagai perangkat komputasi utama yang menangani inferensi. Kamera berfungsi sebagai perangkat input yang mengumpulkan data visual yang kemudian diproses oleh model YOLO untuk mendeteksi objek yang relevan. Hasil deteksi ditampilkan pada monitor untuk memberikan visualisasi kepada pengguna, sementara speaker digunakan untuk memberikan peringatan suara saat objek yang ditargetkan terdeteksi.

3.2 Metode Pengujian Sistem Deteksi Objek

Pengujian sistem deteksi objek dilakukan untuk mengevaluasi kinerja dan akurasi sistem dalam berbagai kondisi. Pengujian ini melibatkan 10 orang sebagai subjek penelitian yang diminta menggunakan objek-objek yang dibagi menjadi tiga kategori: objek utama, kombinasi objek, dan objek yang menyerupai objek utama. Pengujian dilakukan dalam berbagai kondisi pencahayaan dan jarak untuk menguji keandalan sistem dalam situasi yang berbeda.

3.2.1 Kategori Objek

1. Objek Utama: Terdiri dari helm, topi, dan kacamata hitam yang merupakan objek dari penelitian.
2. Kombinasi Objek: Melibatkan kombinasi topi dan kacamata hitam serta helm dan kacamata hitam. Kombinasi ini digunakan untuk menguji kemampuan sistem dalam mendeteksi dan membedakan antara penggunaan tunggal dan penggunaan ganda objek.

3. Objek yang Menyerupai Objek Utama: Meliputi helm sepeda, topi proyek, dan kacamata bening. Objek-objek ini digunakan untuk menilai kemampuan sistem dalam membedakan antara objek utama dan objek yang menyerupai objek utama.

3.2.2 Kondisi Pencahayaan

Pengujian dilakukan dalam tiga kondisi pencahayaan yang berbeda untuk mengevaluasi kinerja sistem dalam berbagai intensitas cahaya:

1. Netral: Kondisi pencahayaan normal tanpa ada tambahan cahaya atau pengurangan cahaya, dengan intensitas antara 100 hingga 600 lux.
2. Lampu: Kondisi pencahayaan dengan tambahan lampu untuk meningkatkan intensitas cahaya, dengan intensitas antara 300 hingga 500 lux.
3. Gelap: Kondisi pencahayaan minim atau gelap untuk menguji kinerja sistem dalam kondisi kurang cahaya, dengan intensitas antara 40 hingga 110 lux.

3.2.3 Kondisi jarak

Subjek penelitian diuji dalam tiga jarak yang berbeda dari kamera untuk mengevaluasi kinerja sistem dalam mendeteksi objek pada berbagai jarak yaitu jarak 1 meter, 2 meter, dan 3 meter.