

BAB II

METODE PENELITIAN

2.1 Lokasi Penelitian

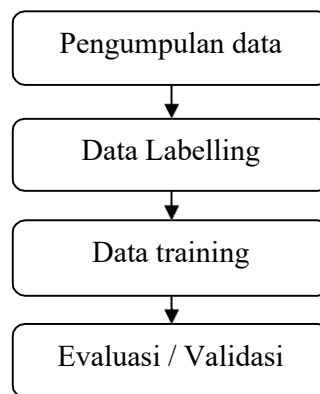
Penelitian ini dilakukan di Universitas Muhammadiyah Kalimantan Timur (UMKT) yang berlokasi di Jl. Ir. H. Juanda No. 15 Samarinda.

2.2 Peralatan Penelitian

Agar penelitian berjalan dengan baik, diperlukan peralatan untuk membantu kelancaran penelitian ini. Peralatan tersebut dibagi menjadi dua bagian yaitu perangkat keras dan perangkat lunak. Perangkat keras yang dibutuhkan meliputi Gawai sebagai perekam, Laptop Asus Vivobook A412DA dengan RAM 12 GB dengan penyimpanan 1 TB HDD dan 250 GB SSD. Sementara itu, perangkat lunak yang diperlukan mencakup Sistem Operasi Windows 10, Visual Studio Code, Google Colab, PyTorch, Roboflow, Bahasa Pemrograman Python dan YOLOv8.

2.3 Tahapan Penelitian

Pada bagian ini akan menjelaskan tahapan penelitian yang akan dilakukan, tahapan penelitian dijabarkan melalui blok diagram seperti pada gambar 2.1.

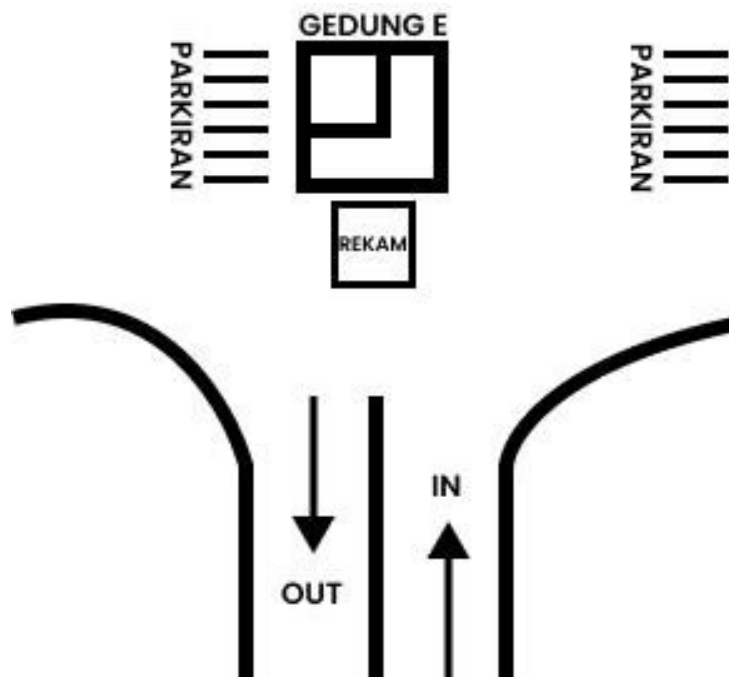


Gambar 2.1 Blok Diagram Metode Penelitian

2.3.1 Pengumpulan Data

Untuk memperoleh data yang akurat dan relevan perlu dilakukan teknik pengumpulan data yaitu menggunakan metode dokumentasi, karena dokumentasi adalah metode yang digunakan untuk menelusuri data historis (Asmi & Permata, 2020). Data yang digunakan pada penelitian ini adalah data yang diperoleh langsung dari pada saat dokumentasi lapangan dari perekaman video kendaraan yang menuju parkir.

Posisi kamera sangat berpengaruh terhadap kualitas hasil dan deteksi (Leriansyah & Kurniawardhani, 2020). Oleh karena itu pengambilan video dilakukan di samping gedung E pada simpang tanjakan yang menuju area parkir. Dengan sudut pengambilan video yang cukup jelas agar objek yang diteliti dapat terlihat sepenuhnya. Berikut adalah sketsa letak gawai yang digunakan untuk pengumpulan data pada gambar 2.2.



Gambar 2.2 Letak Pengumpulan Data

2.3.2 Pelabelan Data

Setelah data terkumpul akan dilakukan anotasi / pelabelan dengan memperhatikan contoh objek yang dilatih. Untuk melakukan anotasi, penulis menggunakan perangkat lunak bernama

Roboflow yang mendukung anotasi *bounding box*. Perangkat lunak ini dipilih karena kemudahannya memberikan akses untuk proses pelabelan, anotasi gambar, pre-processing, proses augmentasi dan kelebihan lainnya untuk menangani dataset (Hidayah et al., 2022). Data akan dilabelkan berdasarkan kategori kendaraan yang relevan dengan tujuan penelitian yakni Manusia, Motor dan Mobil. Proses pelabelan ini bertujuan untuk menghasilkan dataset yang sudah teranotasi dengan benar sehingga dapat digunakan sebagai data uji untuk pengujian model klasifikasi dan deteksi kendaraan.

Sebagai perbandingan, pada penelitian yang dilakukan oleh Amwin (2021) dengan menggunakan video dari CCTV yang kemudian diolah menjadi gambar-gambar yang akan diproses anotasi citra dengan menggunakan perangkat lunak LabelImg dengan menggunakan format anotasi YOLO. Hasilnya adalah data yang terdapat informasi letak kotak pembatas dengan labelnya dalam format .txt.

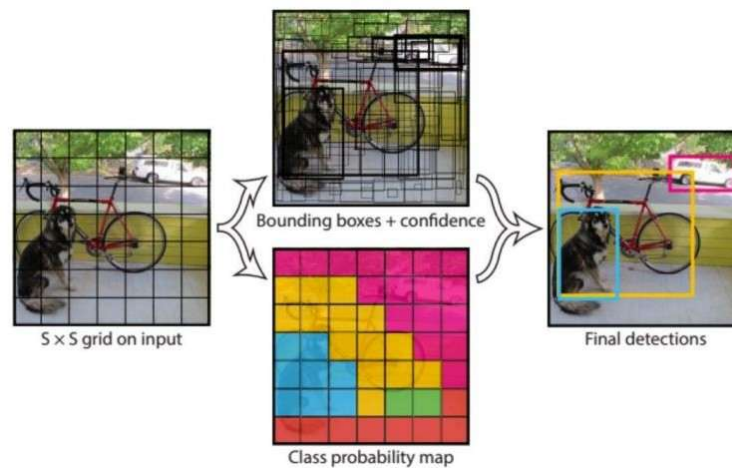
2.3.3 Data Latih

Penelitian ini akan menggunakan data latih langsung dari hasil perekaman video pada saat pengumpulan data. Hal tersebut dilakukan untuk melatih algoritma YOLOv8 dalam merancang model yang sesuai dengan klasifikasinya (Zhang et al., 2022). Dalam data latih ini terdapat banyak rekognisi frame per frame sebagai bahan untuk algoritma agar dapat mendeteksi objek. Untuk pembagian, sebanyak 70% data akan digunakan sebagai data latih, 20% akan digunakan sebagai data validasi serta 10% digunakan sebagai data uji. Data tersebut sudah teranotasi pada tahap pelabelan data agar objek yang lewat nantinya akan mudah untuk diklasifikasi.

Gambar-gambar yang telah diberi label dan diklasifikasikan kemudian akan digunakan untuk melatih model dengan menggunakan algoritma YOLOv8 dengan Google Colab. Google Colab adalah sebuah IDE untuk pemrograman Python dimana pemrosesan akan dilakukan oleh server Google yang memiliki perangkat keras dengan performa yang tinggi (Handayanto dan Herlawati, 2020). Sedangkan YOLO (*You Only Look Once*) adalah algoritma pendeteksi objek

yang dirilis pada tahun 2015 dibuat oleh Joseph Redmon dan Ali Farhadi. YOLO memproses seluruh citra dalam satu lintasan maju melalui jaringan saraf konvolusional (CNN) dan memprediksi kotak pembatas objek dan probabilitas kelas secara *real-time* (Rahma et al., 2021).

Deep learning memungkinkan mesin untuk belajar mengklasifikasikan gambar, audio, teks atau video (Nugroho et al., 2020). YOLOv8 merupakan bagian dari deep learning dan merupakan algoritma *one-stage detection* yang melakukan deteksi lokasi objek dan klasifikasi objek dalam satu tahap (Choudhary, 2022). YOLOv8 beroperasi dengan memindai seluruh area gambar sekali dan kemudian melewati jaringan saraf hanya sekali untuk langsung mendeteksi objek yang ada. Berikut gambar 2.3 merupakan langkah-langkah dalam proses deteksi objek menggunakan YOLOv8.

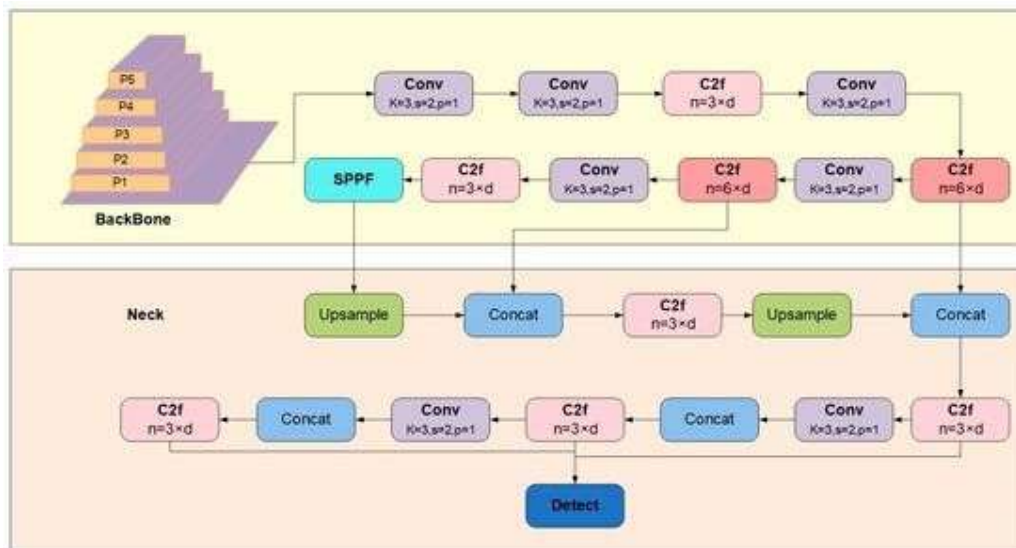


Gambar 2.3 Langkah Kerja Algoritma YOLOv8

Dari gambar tersebut, terdapat beberapa tahap kerja algoritma YOLOv8. (i) Pertama frame terbagi menjadi grid yang berukuran $S \times S$. (ii) Setelah frame dibagi menjadi grid, setiap grid akan memprediksi *bounding box*, koordinat x , y , lebar, tinggi, serta skor kepercayaan. Skor kepercayaan ini mencerminkan keyakinan algoritma bahwa *bounding box* yang diprediksi berisi objek yang ingin dideteksi (Zayed et al., 2021). (iii) Prediksi kelas objek akan dilakukan yang terdapat di dalam setiap *bounding box* dengan. Algoritma ini dapat memprediksi multiple objek dengan berbagai kelas yang berbeda dalam satu frame. (iv) Setelah melakukan prediksi

bounding box, dilakukan penyaringan dengan metode *Non-Maximum Suppression* (NMS) untuk menghapus *bounding box* yang tumpang tindih atau redundan dengan hanya mempertahankan *bounding box* dengan prediksi yang tinggi. (v) Hasil akhirnya adalah YOLO menghasilkan objek-objek yang berhasil dideteksi bersama dengan koordinat *bounding box* dan kelas objeknya yang dapat digunakan untuk berbagai tujuan seperti pelacakan objek, analisis visual hingga pengenalan objek.

YOLOv8 adalah *single-stage real-time object detector* yang dirilis pada Januari 2024. Arsitektur Algoritma YOLOv8 terdiri dari tiga bagian utama, yaitu Backbone, Neck dan Detect. Gambar 2.4 merupakan tampilan Arsitektur YOLOv8.



Gambar 2.4 Arsitektur YOLOv8

Backbone adalah bagian yang bertanggung jawab untuk mengekstrak fitur dari gambar. Tingkat P masing-masing memiliki resolusi yang sesuai dengan piramida, misalnya tingkat 1 (P1) memiliki resolusi tertinggi dan bertanggung jawab untuk mendeteksi objek besar dengan bantuan SPPF (*Spatial Pyramid Pooling Fast*).

Setelah dari Backbone, Neck adalah bagian yang bertanggung jawab untuk menggabungkan fitur dari Backbone menjadi fitur yang lebih abstrak. Struktur Neck menjembatani jaringan backbone dan lapisan output, memfasilitasi integrasi dan fusi fitur pada

berbagai skala dan tingkat abstraksi. Dengan menggunakan struktur *Feature Pyramid Network* (FPN) dan *Path Aggregation Network* (PANet), bagian Neck menangkap informasi kontekstual untuk meningkatkan kemampuan representasi objek yang berbeda dalam ukuran dan posisi (Du et al., 2023).

Pada lapisan output, prediksi deteksi objek dihasilkan. Langkah ini melibatkan penerapan teknik *Non-Maximum Suppression* (NMS) untuk menyaring hasil yang kurang tinggi dari nilai kepercayaan deteksi, sehingga hanya hasil prediksi yang andal dan akurat yang dipertahankan. Setelah fitur dari Backbone dan Neck digabungkan, fitur-fitur tersebut diproses oleh Head untuk menghasilkan bounding box dan skor kepercayaan bagi setiap objek yang terdeteksi.

2.3.4 Evaluasi & Validasi

Pada tahapan ini merupakan tahapan yang digunakan untuk mengukur kesesuaian antara hasil klasifikasi model dengan kelas yang sebenarnya. Mempertimbangkan kebutuhan untuk identifikasi secara langsung, penelitian ini menggunakan *Precision* dan *mean Average Precision* (mAP) sebagai indikator evaluasi model klasifikasi karena kedua nilai tersebut memberikan gambaran tentang kemampuan model deteksi dalam mengenali objek secara presisi (Surya et al., 2024). Selain itu, validasi manual juga akan dilakukan untuk memperoleh status algoritma dalam memproses deteksi setiap objek yang diidentifikasi, validasi ini berupa tabel sebagai perbandingan data aktual yang ada dengan data yang diperoleh dari deteksi algoritma YOLOv8.