

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Gambaran Umum Penelitian

Penelitian ini dilakukan dengan menghitung jumlah kendaraan yang melintas dengan menerapkan algoritma YOLO, SSD dan *Haar Cascade*. Penggunaan algoritma YOLO menggunakan YOLOv3 dan algoritma SSD menggunakan versi SSD300. Ketiga algoritma tersebut di uji menggunakan file video dengan resolusi dan nilai *brightness* yang berbeda yang diambil di jalan raya. Hasil perhitungan jumlah kendaraan dengan ketiga algoritma tersebut kemudian di uji dengan menggunakan evaluasi model yaitu nilai akurasi.

#### 3.2 Tahapan Penelitian

Tahap-tahap pada penelitian ini adalah sebagai berikut:



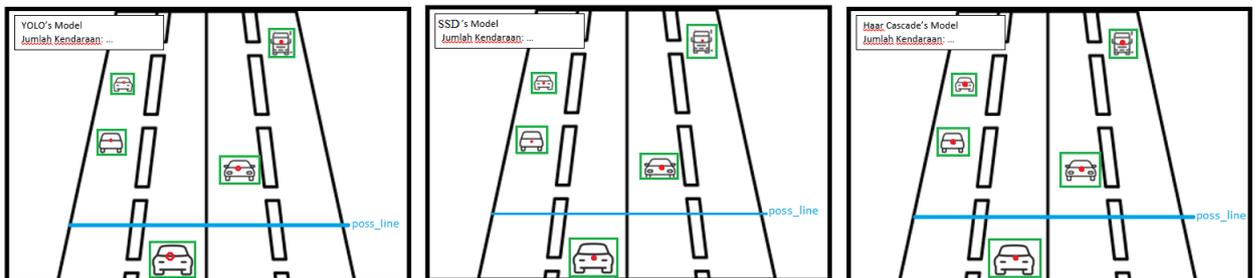
**Gambar 3.1** Tahapan Penelitian

### 3.2.1 Pengumpulan Data

Data yang akan digunakan pada penelitian ini berupa video yang di ambil di jalan raya dengan resolusi yang diubah menjadi 1440p, 720p dan 360p. 3 video dengan resolusi berbeda kemudian di proses kembali dengan mengatur nilai *brightness* menjadi 3 kondisi yang berbeda yaitu gelap (nilai *brightness* = 0,5), normal (nilai *brightness* = 1) dan terang (nilai *brightness* = 1,5).

### 3.2.2 Perancangan Model

Model yang akan dibuat menggunakan bahasa pemrograman Python menggunakan *environment* Google Colab dengan *runtime type* T4. Berikut adalah gambaran model yang akan dibuat dengan menggunakan Algoritma YOLO, SSD dan *Haar Cascade*.



**Gambar 3.2** Visualisasi Implementasi Model

Masing-masing model akan menampilkan kotak berwarna hijau (*bounding box*) untuk area objek yang terdeteksi yaitu kendaraan dan titik merah merupakan pusat objek. Apabila objek melewati garis berwarna biru maka model akan menghitung jumlah objek. *Poss\_line* berfungsi sebagai sensor apabila objek melaluinya.

### 3.2.3 Training

Pada tahap ini data berupa video akan di impor kedalam model yang telah dibuat satu persatu. Kemudian model akan mendeteksi dan menghitung objek (kendaraan) yang melewati *pos\_line* dan hasil perhitungannya akan di tampilkan pada video yang telah diproses oleh model masing-masing algoritma. Setiap video akan di *training* masing-masing sebanyak sepuluh kali untuk setiap Model Algoritma YOLO, SSD dan *Haar Cascade*. Output yang akan dihasilkan adalah berupa video dengan durasi yang sama dengan input dan di dalam setiap *frame* terdapat *pos line*, *bounding box* dan titik pusat pada objek terdeteksi.

### 3.2.4 Analisis Perbandingan Hasil Deteksi

Hasil perhitungan jumlah kendaraan dengan menerapkan algoritma YOLO, SSD dan *Haar Cascade* akan di hitung kemudian dianalisis dan dibandingkan berdasarkan jumlah hasil deteksi.

### 3.2.5 Perhitungan Performa

Perhitungan performa dengan menggunakan nilai akurasi dari Algoritma YOLO, SSD dan Haar Cascade. Akurasi merupakan rasio prediksi benar dengan keseluruhan data. Berikut adalah persamaan untuk menghitung nilai akurasi:

$$Accuracy = \frac{TP(True\ Positif)}{Total\ Dataset}$$

### **3.2.6 Kesimpulan**

Nilai akurasi yang telah di hitung dari Algoritma YOLO, SSD dan *Haar Cascade* kemudian dibandingkan dan algoritma dengan nilai akurasi terbesar merupakan algoritma yang paling efektif untuk menghitung jumlah kendaraan yang melintas.