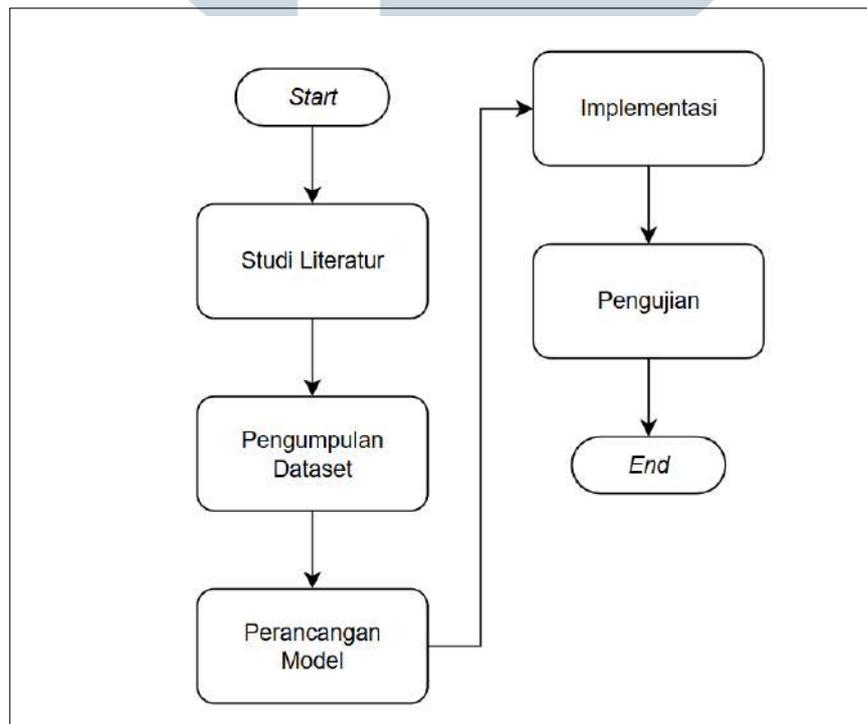


BAB 3 METODOLOGI PENELITIAN

3.1 Metode Penelitian

Metode penelitian yang digunakan terdiri dari beberapa tahapan yang saling berkaitan seperti pada gambar 3.1. Proses dimulai dengan identifikasi masalah untuk menentukan fokus utama yang akan dikaji. Setelah itu, dilakukan studi literatur guna memperoleh dasar teori dan referensi yang mendukung penelitian. Tahap berikutnya adalah pengumpulan dataset dari berbagai macam sumber untuk variasi yang dilakukan untuk melatih model. Selanjutnya adalah perancangan model, yang bertujuan untuk mengatur proses pelatihan dan uji pada model. Setelah model selesai dirancang, tahap selanjutnya adalah implementasi, di mana rancangan pembangunan aplikasi utama dimulai. Tahap pengujian dilakukan untuk memastikan bahwa model yang diimplementasikan berfungsi dengan baik.



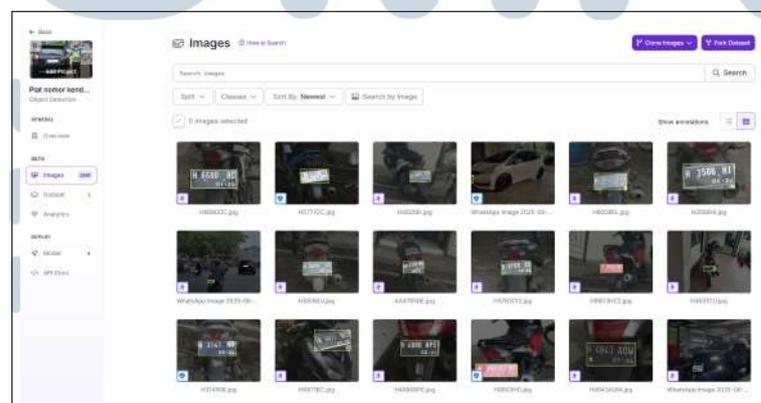
Gambar 3.1. Diagram langkah-langkah penelitian

3.2 Studi Literatur

Studi literatur dilakukan dengan menelusuri dan mengkaji secara mendalam berbagai penelitian yang relevan dengan komponen-komponen utama dalam sistem ini. Fokus utama kajian adalah pada arsitektur deteksi objek menggunakan *YOLOv8* dan model segmentasi citra dengan *Segment Anything Model 2 (SAM2)* untuk memahami keunggulan serta keterbatasan masing-masing teknologi dalam konteks pemrosesan video. Selain itu, dilakukan pula analisis mendalam terhadap teknik *fine-tuning* pada model *deep learning*, yang menjadi strategi inti untuk meningkatkan akurasi deteksi plat nomor kendaraan secara optimal. Kajian ini juga mencakup analisis metode pengaburan berbasis *masking* yang presisi sebagai pendekatan untuk menjaga privasi tanpa mengorbankan integritas visual dari video secara signifikan. Referensi dari jurnal dan penelitian serupa dieksplorasi guna mendapatkan wawasan tentang implementasi serupa, tantangan yang dihadapi, serta solusi yang telah diterapkan dalam penelitian-penelitian sebelumnya.

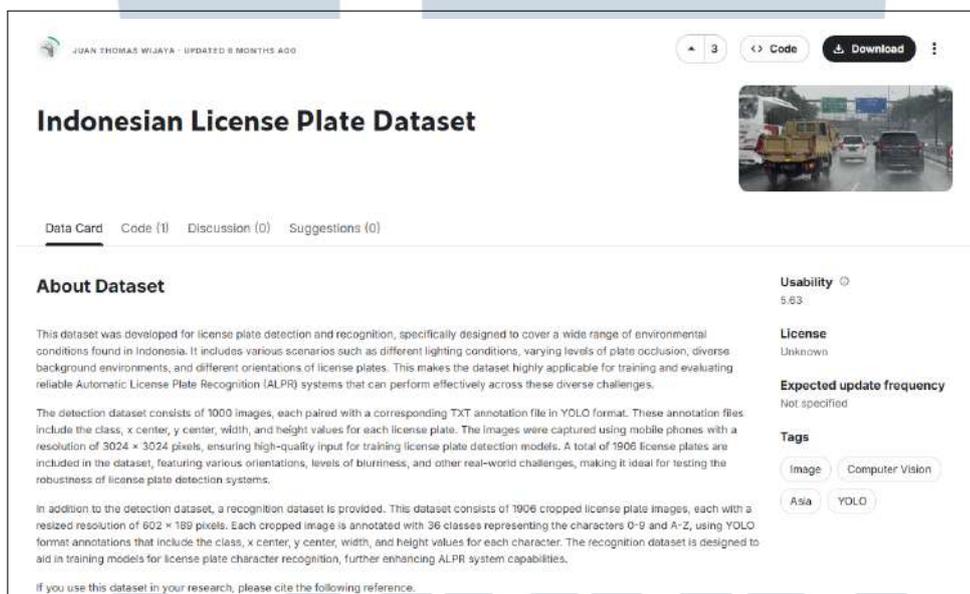
3.3 Pengumpulan Dataset

Tahap pengumpulan data merupakan fondasi utama dalam penelitian ini untuk melatih dan mengevaluasi model deteksi plat nomor yang andal. Untuk memastikan keragaman dan representasi data yang komprehensif, penelitian ini menggunakan pendekatan penggabungan dataset dari tiga sumber berbeda, yaitu koleksi gambar pribadi penulis serta dua dataset publik dari platform Kaggle. Penggabungan ini bertujuan untuk menciptakan sebuah dataset akhir yang lebih kokoh, mencakup berbagai jenis kendaraan dan kondisi visual yang ada di Indonesia, dengan total keseluruhan mencapai 1.830 gambar.



Gambar 3.2. *Dataset* plat nomor kendaraan Indonesia

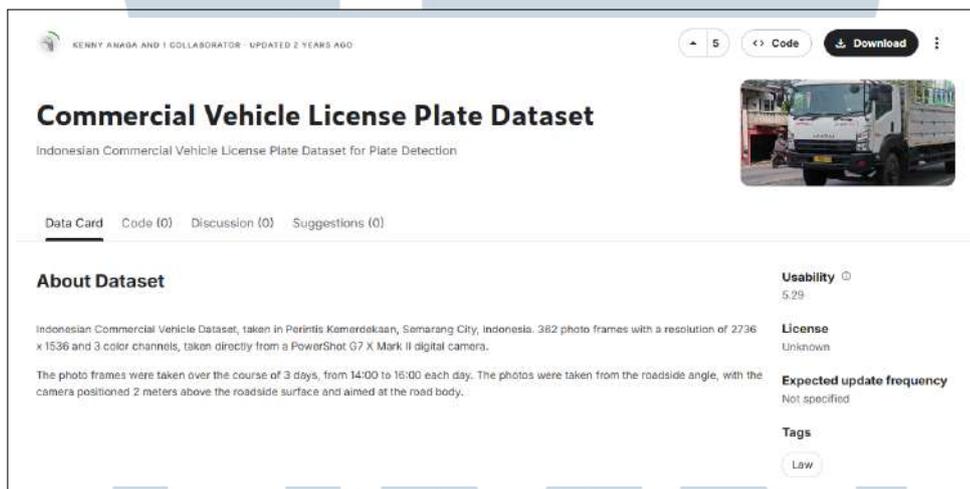
Dataset publik pertama yang digunakan adalah *Indonesian License Plate Dataset* seperti pada gambar 3.3 yang dikembangkan oleh Juan Thomas Wijaya. Dataset ini dirancang secara spesifik untuk mencakup berbagai skenario dan kondisi lingkungan di Indonesia, seperti variasi pencahayaan, tingkat oklusi plat, serta orientasi yang beragam. Untuk keperluan deteksi, dataset tersebut menyediakan 1.000 gambar berkualitas tinggi dengan resolusi 3024 x 3024 piksel, yang masing-masing dilengkapi dengan *file* anotasi berformat TXT YOLO. Anotasi tersebut mencakup informasi kelas dan koordinat *bounding-box* untuk total 1.906 plat nomor, menjadikannya sumber daya yang sangat relevan untuk melatih model deteksi yang tangguh.



Gambar 3.3. *Dataset* plat nomor kendaraan Indonesia

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Untuk memperkaya variasi data, khususnya pada kendaraan komersial, penelitian ini juga memanfaatkan *Commercial Vehicle License Plate Dataset* seperti pada gambar 3.4 yang dikembangkan oleh Kenny Anaga dan Risang Baskoro. Dataset ini terdiri dari 362 *frame* foto kendaraan komersial Indonesia yang diambil secara langsung di Kota Semarang menggunakan kamera digital PowerShot G7 X Mark II dengan resolusi 2736 x 1536 piksel. Pengambilan data dilakukan selama tiga hari pada sore hari untuk menangkap kondisi pencahayaan yang konsisten dari sudut pandang sisi jalan. Penggunaan dataset ini secara spesifik bertujuan untuk meningkatkan kemampuan model dalam mengenali plat nomor pada jenis kendaraan komersial yang seringkali memiliki karakteristik visual dan kondisi fisik yang berbeda.

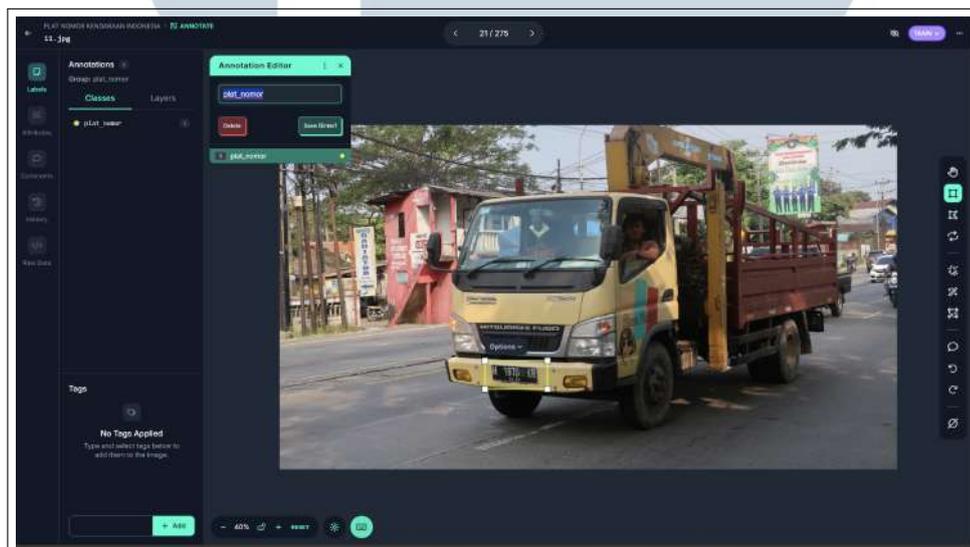


Gambar 3.4. *Dataset* plat nomor kendaraan Indonesia

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.3.1 Proses Anotasi

Menindaklanjuti tahap pengumpulan data, keseluruhan 1.830 gambar masukan menjadi subjek proses anotasi manual yang bertujuan untuk menetapkan label *ground truth* bagi setiap objek plat nomor. Proses pelabelan tersebut diimplementasikan menggunakan platform *online Roboflow* yang menyajikan antarmuka serta perangkat untuk pelabelan *object detection*. Metodologi anotasi yang diaplikasikan adalah *bounding-box*, di mana sebuah kotak pembatas didefinisikan secara presisi untuk melingkupi seluruh area spasial plat nomor yang terlihat pada gambar seperti pada contoh 3.5. Guna menjaga konsistensi data dan spesifisitas model, seluruh anotasi diklasifikasikan ke dalam satu kelas tunggal, yaitu `plat_nomor`. Penetapan label ini esensial sebagai data acuan bagi model YOLOv8 dalam mempelajari dan mengenali batasan spasial dari objek target.



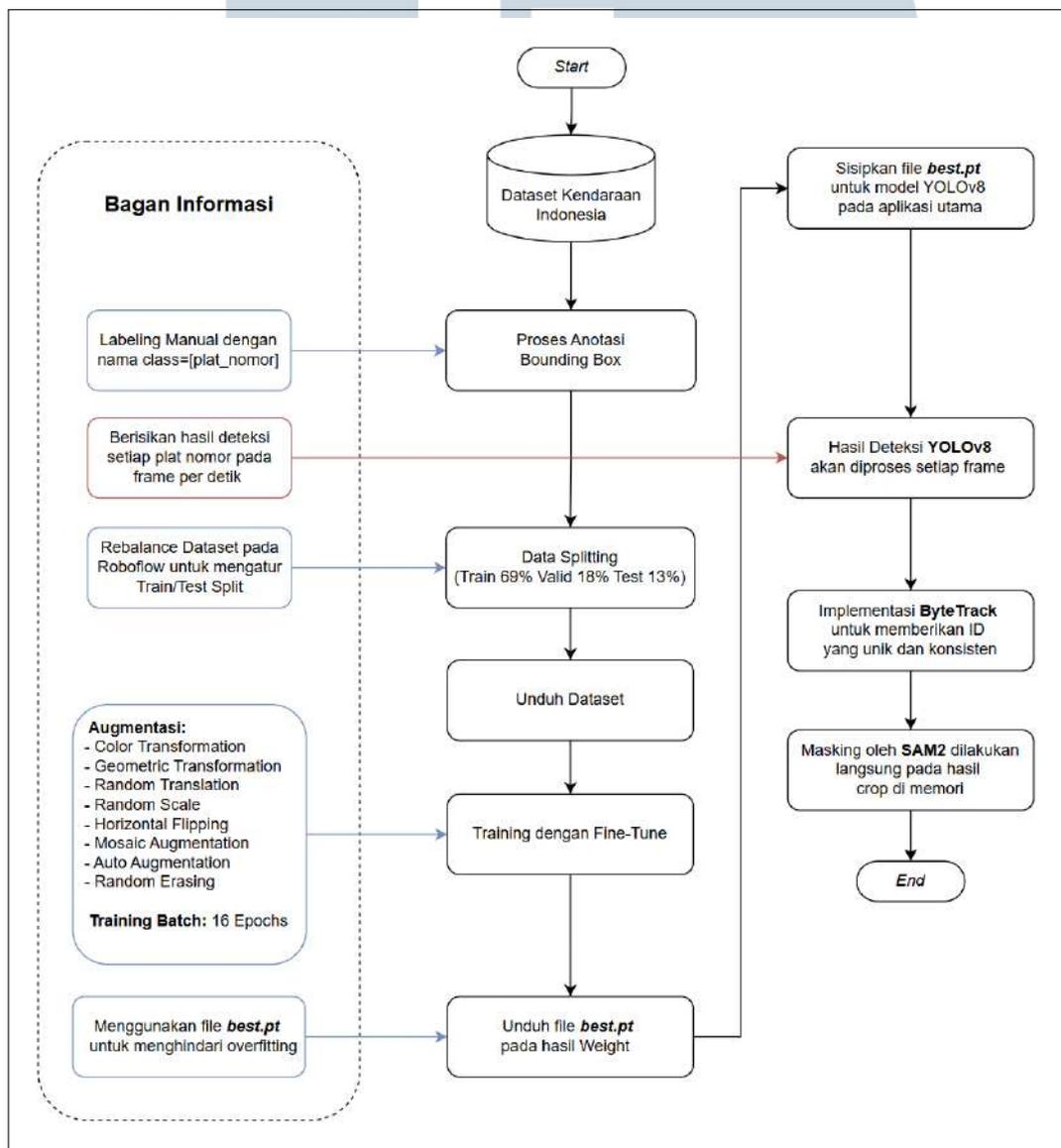
Gambar 3.5. Anotasi Dataset pada Roboflow

Sumber: Roboflow

UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.3.2 Perancangan Model

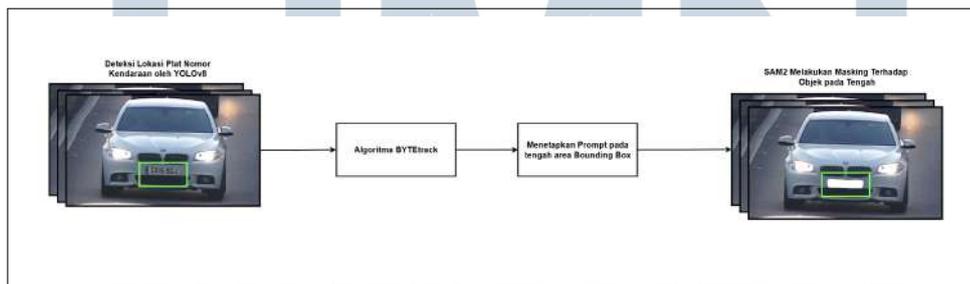
Pada penelitian ini, sistem yang dirancang bertujuan untuk mendeteksi, melacak, dan mengekstraksi plat nomor kendaraan dari rekaman video secara otomatis. Proses tersebut melibatkan beberapa model kecerdasan buatan yang bekerja secara berurutan untuk memastikan hasil yang akurat dan stabil. Diagram 3.6 menggambarkan alur rancangan sistem deteksi dan segmentasi plat nomor kendaraan.



Gambar 3.6. Alur rancang model penelitian

Proses dimulai dengan Model Deteksi YOLOv8 yang digunakan untuk mendeteksi plat nomor kendaraan dari dataset yang telah dilabeli dengan *bounding-box*. Dataset tersebut dibagi menjadi *Train Dataset* untuk melatih model dan *Test Dataset* untuk menguji performa model. Setelah plat nomor terdeteksi, Model Segmentasi SAM2 digunakan untuk melakukan segmentasi pada area plat nomor. Selanjutnya, Algoritma BYTETrack diimplementasikan untuk melacak lokasi plat nomor secara konsisten dalam video atau *frame* berurutan. Hasil akhir dari proses tersebut adalah Output Hasil yang mencakup deteksi dan segmentasi plat nomor, menghasilkan sistem yang mampu mengidentifikasi dan memproses plat nomor kendaraan Indonesia dengan akurasi tinggi.

Diagram 3.7 menggambarkan alur proses deteksi dan masking plat nomor kendaraan yang dimulai dengan deteksi lokasi plat nomor menggunakan YOLOv8, sebuah model deteksi objek yang cepat dan akurat. Setelah plat nomor terdeteksi, algoritma BYTETrack digunakan untuk melacak objek tersebut secara konsisten dalam video, memastikan plat nomor yang sama dapat diidentifikasi dari frame ke frame. Selanjutnya, *prompt* (titik referensi) ditetapkan di tengah area *bounding-box* yang mengelilingi plat nomor, yang kemudian digunakan oleh SAM2 untuk melakukan masking pada objek di tengah *bounding-box* tersebut. Proses masking tersebut bertujuan untuk mengaburkan atau menyembunyikan plat nomor guna melindungi privasi, sehingga menghasilkan alur kerja yang efektif dari deteksi, pelacakan, hingga pengaburan plat nomor kendaraan.



Gambar 3.7. Alur deteksi dan segmentasi

A YOLOv8

YOLO melakukan *cropping* pada area plat nomor dengan mendeteksi *bounding-box* di sekitar objek plat nomor menggunakan model yang telah dilatih. Setelah YOLO memproses setiap *frame* video, model ini menghasilkan koordinat *bounding-box* (x_1, y_1, x_2, y_2) yang menunjukkan lokasi plat nomor dalam *frame*. Dengan koordinat tersebut, area plat nomor dipotong (*cropping*) dari *frame* asli menggunakan operasi *slicing* pada *array* gambar, menghasilkan gambar yang hanya berisi plat nomor. Gambar yang dipotong ini kemudian dapat disimpan untuk keperluan debugging atau diproses lebih lanjut, seperti segmentasi menggunakan model SAM2 untuk memisahkan plat nomor dari latar belakangnya. Dengan demikian, YOLO memungkinkan ekstraksi area plat nomor secara otomatis dan akurat untuk analisis lebih lanjut.



Gambar 3.8. Area deteksi plat nomor dengan *bounding-box*

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Untuk mengevaluasi kinerja model YOLOv8 dalam mendeteksi plat nomor kendaraan, digunakan metrik evaluasi seperti *Precision*, *Recall*, dan *F1-Score*. Metrik tersebut membantu mengukur sejauh mana model dapat mendeteksi objek dengan akurasi tinggi dan meminimalkan kesalahan deteksi.

1. Precision

Precision merupakan metrik evaluasi yang digunakan untuk mengukur seberapa akurat model dalam melakukan prediksi terhadap kelas positif. Nilai precision menunjukkan proporsi dari prediksi positif yang benar-benar relevan (benar). Rumus ini menghitung precision sebagai rasio antara jumlah *True Positive* (TP), yaitu prediksi benar untuk kelas positif, terhadap total prediksi positif, yaitu *True Positive* ditambah *False Positive* (FP). Semakin tinggi nilai precision, semakin sedikit kesalahan prediksi positif yang dilakukan oleh model.

Rumus 3.1 menunjukkan cara perhitungan *Precision*:

$$Precision = \frac{TP}{TP + FP} \quad (3.1)$$

2. Recall

Recall mengukur sejauh mana model mampu mendeteksi semua instance dari kelas positif yang sebenarnya ada dalam data. Nilai recall dihitung dari rasio antara jumlah *True Positive* (TP) terhadap jumlah semua instance positif yang sebenarnya, yaitu *True Positive* ditambah *False Negative* (FN). Recall menjadi penting ketika tujuan utama adalah menangkap sebanyak mungkin kasus positif, seperti dalam sistem deteksi penyakit atau keamanan. Nilai recall yang tinggi menunjukkan bahwa model jarang melewatkan kasus positif.

Rumus 3.2 menunjukkan cara perhitungan *Recall*:

$$Recall = \frac{TP}{TP + FN} \quad (3.2)$$

3. F1 Score

F1 Score adalah metrik gabungan yang mempertimbangkan keseimbangan antara *precision* dan *recall*. Metrik ini sangat berguna ketika terdapat ketidakseimbangan antara jumlah data positif dan negatif dalam dataset. *F1 Score* dihitung sebagai rata-rata harmonik dari *precision* dan *recall*, sehingga hanya akan bernilai tinggi jika kedua metrik tersebut juga tinggi. Dengan demikian, *F1 Score* memberikan ukuran performa yang lebih menyeluruh dibandingkan hanya menggunakan *precision* atau *recall* secara terpisah.

Rumus 3.3 menunjukkan cara perhitungan *F1 Score*:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3.3)$$

4. Mean Average Precision (mAP)

Mean Average Precision (mAP) merupakan metrik evaluasi paling komprehensif untuk tugas *object detection*, karena secara simultan mengukur ketepatan klasifikasi dan presisi lokalisasi *bounding-box*. Nilai mAP pada dasarnya adalah rata-rata dari *Average Precision* (AP) untuk seluruh kelas objek yang dievaluasi, sebagaimana ditunjukkan pada Rumus 3.4.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3.4)$$

Di mana:

- N adalah jumlah total kelas objek.
- AP_i adalah nilai *Average Precision* untuk kelas ke- i .

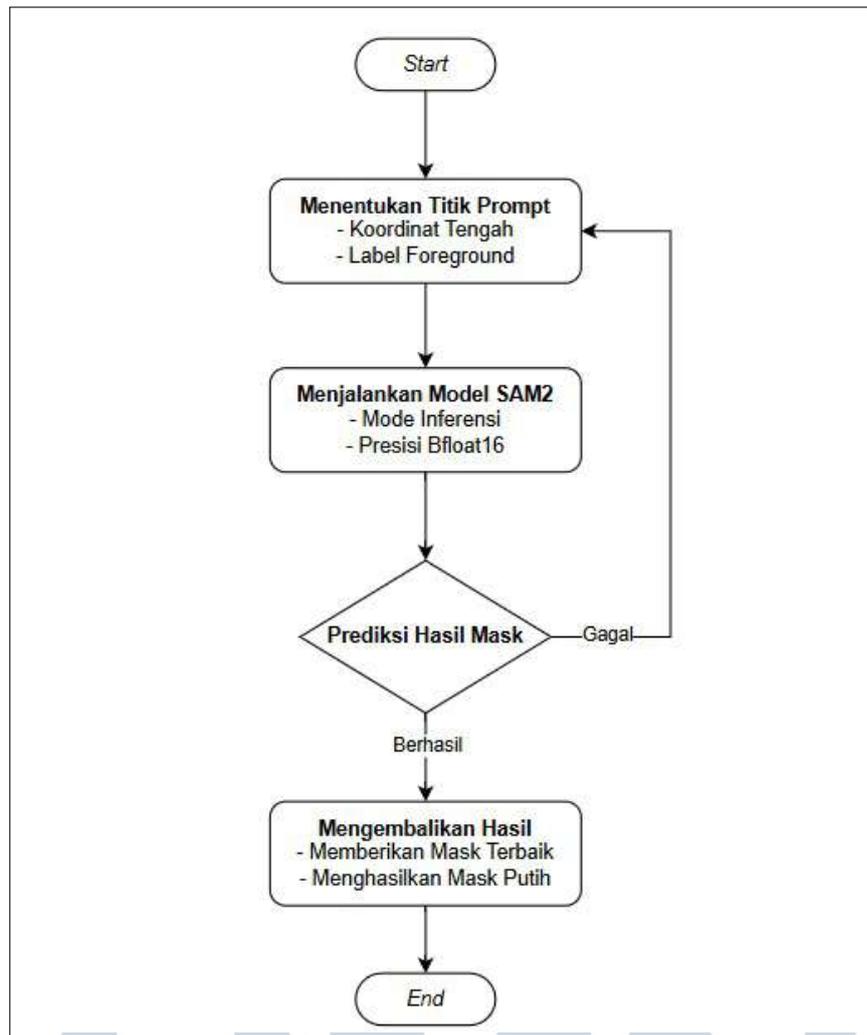
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

B SAM2

Model SAM2 yang digunakan adalah `sam2.1.hiera.small` yang dimana sudah memberi dukungan yang efisien dan model segmentasi yang cukup pada skala proyek yang dikerjakan. SAM2 dalam proyek tersebut digunakan untuk melakukan segmentasi pada area plat nomor kendaraan yang telah dipotong dari gambar asli. Proses segmentasi dilakukan dengan memilih titik pusat dari gambar yang telah dipotong, karena umumnya plat nomor berada di bagian tengah dan bawah kendaraan. Model kemudian menghasilkan mask biner, yang secara spesifik memisahkan area plat nomor dari latar belakang. Dengan pendekatan ini, sistem dapat mengekstraksi plat nomor dengan lebih akurat, sehingga mempermudah langkah-langkah pemrosesan selanjutnya seperti identifikasi atau pelacakan.

Gambar 3.9 menjelaskan alur proses Segment Anything Model (SAM2) dimulai dengan menentukan titik *prompt* di tengah gambar (*foreground*) menggunakan koordinat pusat dari *bounding-box* yang telah di-*crop*. Titik tersebut diberi label sebagai *foreground* (`point_labels = np.array([1])`). Selanjutnya, model SAM2 dijalankan dengan menggunakan mode inferensi dan presisi `bfloat16` pada GPU untuk efisiensi komputasi. Model ini menerima input berupa gambar yang telah di-*crop* (`cropped_plate_rgb`), *bounding-box* sebagai *prompt* (`prompt_box`), serta koordinat dan label titik tengah. Proses prediksi menghasilkan *mask* segmentasi (`masks`), skor kepercayaan (`scores`), dan informasi tambahan. *Mask* dengan skor tertinggi dipilih sebagai hasil segmentasi terbaik (`best_mask`). Jika *mask* berhasil dihasilkan, *mask* tersebut dikembalikan; jika tidak, fungsi mengembalikan `None`. Proses tersebut memungkinkan segmentasi objek (plat nomor) dengan akurasi tinggi berdasarkan *prompt* yang diberikan.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.9. Diagram proses SAM2

C BYTETrack

BYTETrack berperan penting dalam menjaga konsistensi pelacakan plat nomor kendaraan yang terdeteksi dalam video. Algoritma tersebut digunakan untuk mengatasi masalah *flickering*, di mana deteksi YOLO terkadang tidak konsisten dan dapat menghilangkan objek pada frame tertentu. Dengan menerapkan BYTETrack, sistem dapat mempertahankan *bounding-box* plat nomor meskipun YOLO gagal mendeteksinya dalam beberapa *frame*, sehingga pelacakan menjadi lebih stabil dan akurat. Hal tersebut memungkinkan identifikasi plat nomor yang lebih handal dan berkelanjutan, terutama dalam kondisi pencahayaan yang kurang ideal atau pergerakan kendaraan yang cepat.

3.3.3 Implementasi

Tahap implementasi merupakan realisasi teknis dari arsitektur sistem yang telah dirancang, meliputi penulisan kode, konfigurasi model, dan penyiapan lingkungan pengembangan untuk digunakan pada aplikasi utama. Sistem tersebut dibangun menggunakan bahasa pemrograman Python. Komponen deteksi objek memanfaatkan pustaka *Ultralytics* untuk memuat dan melatih ulang (*fine-tuning*) model YOLOv8. Selanjutnya, untuk menjaga konsistensi identitas objek, diintegrasikan algoritma pelacakan BYTETrack, sementara tahap segmentasi presisi menggunakan Segment Anything Model 2 (SAM2) yang dipandu oleh hasil deteksi YOLOv8.

Keseluruhan alur kerja tersebut kemudian digabungkan ke dalam sebuah aplikasi berbasis web yang dibangun menggunakan *framework Streamlit* guna menyediakan antarmuka pengguna yang interaktif. Aplikasi tersebut dirancang agar pengguna dapat mengunggah video untuk selanjutnya diproses oleh *pipeline* deteksi, pelacakan, dan segmentasi secara berurutan. Lingkungan utama untuk pengembangan dan eksekusi yang digunakan dalam penelitian adalah Google Colab, yang didukung oleh T4 *Graphics Processing Unit* (GPU) untuk mengakselerasi proses pelatihan maupun inferensi model.



3.3.4 Pengujian

Untuk melakukan validasi terhadap sistem yang telah dibangun, serangkaian uji coba dilakukan dalam dua skenario berbeda. Pengujian tersebut bertujuan untuk mengukur performa deteksi dan kualitas segmentasi model pada kondisi yang beragam.

A Uji pada rekaman CCTV

Pengujian sistem akan dilakukan menggunakan data rekaman lalu lintas dari berbagai sumber CCTV di Indonesia untuk mengukur akurasi model dalam mendeteksi variasi plat nomor dari beragam wilayah. Rekaman CCTV tersebut akan disimpan dalam bentuk video klip pendek yang kemudian akan diunggah ke dalam sistem untuk proses pengujian. Gambar 3.11 menunjukkan contoh klip siaran dari CCTV Balitower di Gelora Bung Karno Jakarta, Indonesia pada tahun 2025 mengenai lalu lintas di wilayah perkotaan sebagai representasi data uji.



Gambar 3.10. Contoh rekaman CCTV lalu lintas di Gelora Bung Karno, Jakarta
Sumber: CCTV Balitower

UNIVERSITAS
MULTIMEDIA
NUSANTARA

B Uji pada rekaman smartphone pada jarak dekat

Uji rekaman jarak dekat dengan menggunakan *smartphone* digunakan untuk melakukan uji coba apakah model-model tersebut berhasil dalam segmentasi plat nomor kendaraan yang ada. Sumber rekaman tersebut berasal dari *smartphone* milik Penulis yaitu Samsung Galaxy A73 dan beberapa sumber lainnya dari Pinterest dimana rekaman yang diambil adalah video situasi jalanan dengan menyoroti berbagai kendaraan bermotor di Indonesia. Video yang diuji bertujuan untuk menangkap fleksibilitas kemampuan model dalam melakukan pengaburan dari jarak dekat hingga jauh.



Gambar 3.11. Contoh rekaman jarak dekat menggunakan smartphone

C Uji Coba Sudut Pandang Pengambilan Rekaman

Sebuah sistem yang andal harus mampu bekerja secara efektif dalam berbagai kondisi, termasuk saat merekam dari sudut pandang yang berbeda. Oleh karena itu, dilakukan pengujian performa sistem secara spesifik menggunakan skenario perekaman dari arah depan, belakang dan atas. Pengujian ini bertujuan untuk memastikan sistem dapat memberikan hasil yang akurat dan konsisten, meskipun posisi kendaraan yang terekam tidak dalam kondisi lurus sempurna menghadap kamera.

Gambar 3.12 menunjukkan pengujian dengan rekaman dari sudut pandang depan.



Gambar 3.12. Contoh rekaman dengan sudut pandang depan

Sumber: Youtube Fitra Eri

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Gambar 3.13 menunjukkan pengujian dengan rekaman dari sudut belakang.



Gambar 3.13. Contoh rekaman dengan sudut pandang belakang

Gambar 3.14 menunjukkan pengujian dengan rekaman dari sudut atas.



Gambar 3.14. Contoh rekaman dengan sudut pandang atas

Sumber: CCTV Dishub Kota Bandung

D Uji Coba Kondisi Pencahayaan

Faktor pencahayaan merupakan salah satu variabel eksternal yang paling berpengaruh terhadap kualitas citra yang dihasilkan dan kinerja sistem secara keseluruhan. Perbedaan drastis antara kondisi gelap dan terang dapat menyebabkan tantangan seperti gambar buram, *noise*, atau cahaya berlebih (*overexposure*), yang semuanya berdampak pada akurasi deteksi. Oleh karena itu, pengujian sistem wajib dilakukan dalam skenario pencahayaan yang berbeda untuk memvalidasi kemampuannya dalam beradaptasi.

Gambar 3.15 menunjukkan contoh rekaman dalam kondisi pencahayaan gelap atau pada malam hari.



Gambar 3.15. Contoh rekaman pada kondisi pencahayaan gelap

Gambar 3.16 menunjukkan contoh rekaman dalam kondisi pencahayaan terang atau pada siang hari.



Gambar 3.16. Contoh rekaman pada kondisi pencahayaan terang