

BAB III

HASIL ANALISIS DAN PEMBAHASAN

3.1 Pengumpulan Data

Penelitian ini menggunakan data video perekaman kendaraan yang keluar masuk area parkir Universitas Muhammadiyah Kalimantan Timur (UMKT), yang diperoleh melalui metode dokumentasi. Pengumpulan data dilakukan pada hari Rabu, tanggal 8 Mei 2024. Proses perekaman video dilakukan dengan memperhatikan berbagai aspek teknis seperti resolusi dan konfigurasi kamera untuk memastikan kualitas video yang dihasilkan sesuai dengan standar CCTV. Hal ini dilakukan agar hasil perekaman memiliki kesamaan dalam hal kualitas dan dapat digunakan secara efektif dalam penelitian ini.

Pada hari tersebut, dilakukan perekaman sebanyak tiga video yang mencakup keseluruhan aktivitas di area parkir. Video-video ini kemudian diolah dan dibagi menjadi tiga kategori utama, yaitu data latih, data uji, dan data validasi. Proses pembagian ini dilakukan dengan cermat untuk memastikan bahwa setiap kategori memiliki representasi yang cukup dari berbagai kondisi dan situasi yang terekam. Tujuannya adalah untuk melatih model YOLOv8 dengan data yang beragam sehingga meningkatkan akurasi deteksi dan klasifikasi kendaraan.

Video yang direkam awalnya beresolusi Full HD (1080p). Namun, untuk menyesuaikan dengan standar resolusi CCTV yang umumnya digunakan, resolusi video tersebut dikonversi menjadi HD (720p). Konversi ini dilakukan tanpa mengurangi kualitas signifikan dari video, sehingga tetap memberikan detail yang cukup untuk keperluan pelabelan dan pelatihan model.

Selain itu, penyesuaian resolusi ini membantu dalam memastikan bahwa model yang dilatih dapat diterapkan pada sistem CCTV yang sebenarnya, yang biasanya memiliki resolusi serupa.

Proses perekaman ini juga melibatkan penyesuaian sudut kamera dan penempatan strategis untuk menangkap seluruh aktivitas kendaraan di area parkir. Perekaman dilakukan secara kontinu untuk mendapatkan data yang cukup banyak dan variatif, mencakup berbagai jenis

kendaraan seperti mobil dan motor, serta aktivitas keluar masuk kendaraan yang terjadi. Dengan data yang komprehensif ini, penelitian dapat menghasilkan model deteksi yang lebih akurat yang mampu mengidentifikasi dan menghitung kendaraan secara efektif dalam berbagai kondisi pencahayaan dan situasi. Salah satu frame data latih dapat dilihat pada gambar 3.1.



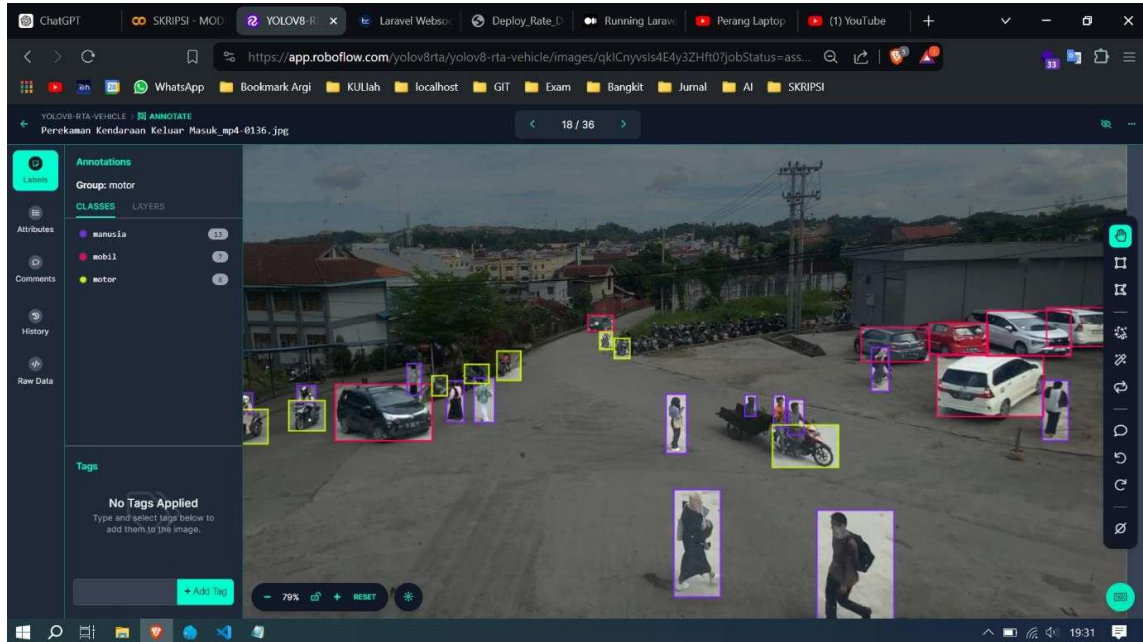
Gambar 3.1 Pengumpulan Data

3.2 Pelabelan Data

Setelah mengumpulkan video data latih dengan durasi 2 menit 51 detik, dilakukan proses ekstraksi frame sehingga didapatkan sejumlah frame yang siap untuk dianotasi. Proses ekstraksi frame ini bertujuan untuk mengonversi video menjadi serangkaian gambar statis yang kemudian dapat dianalisis secara individual. Setelah frame-frame tersebut diekstraksi, langkah berikutnya adalah proses anotasi yang sangat penting untuk pelatihan model deteksi.

Proses anotasi ini dilakukan dengan menggunakan perangkat lunak Roboflow, yang merupakan alat yang sangat berguna untuk membuat bounding box dan melabeli objek dalam gambar. Roboflow memudahkan penulis untuk melakukan pelabelan secara efisien dan akurat. Berikut adalah salah satu proses pelabelan data pada Roboflow yang ditampilkan pada gambar 3.2. Pada tahap ini, setiap frame dianotasi dengan memperhatikan tiga kategori utama, yaitu

motor, mobil, dan manusia. Kategori-kategori ini dipilih berdasarkan relevansi dan tujuan penelitian untuk mendeteksi dan mengklasifikasikan jenis-jenis kendaraan serta keberadaan manusia di area parkir.

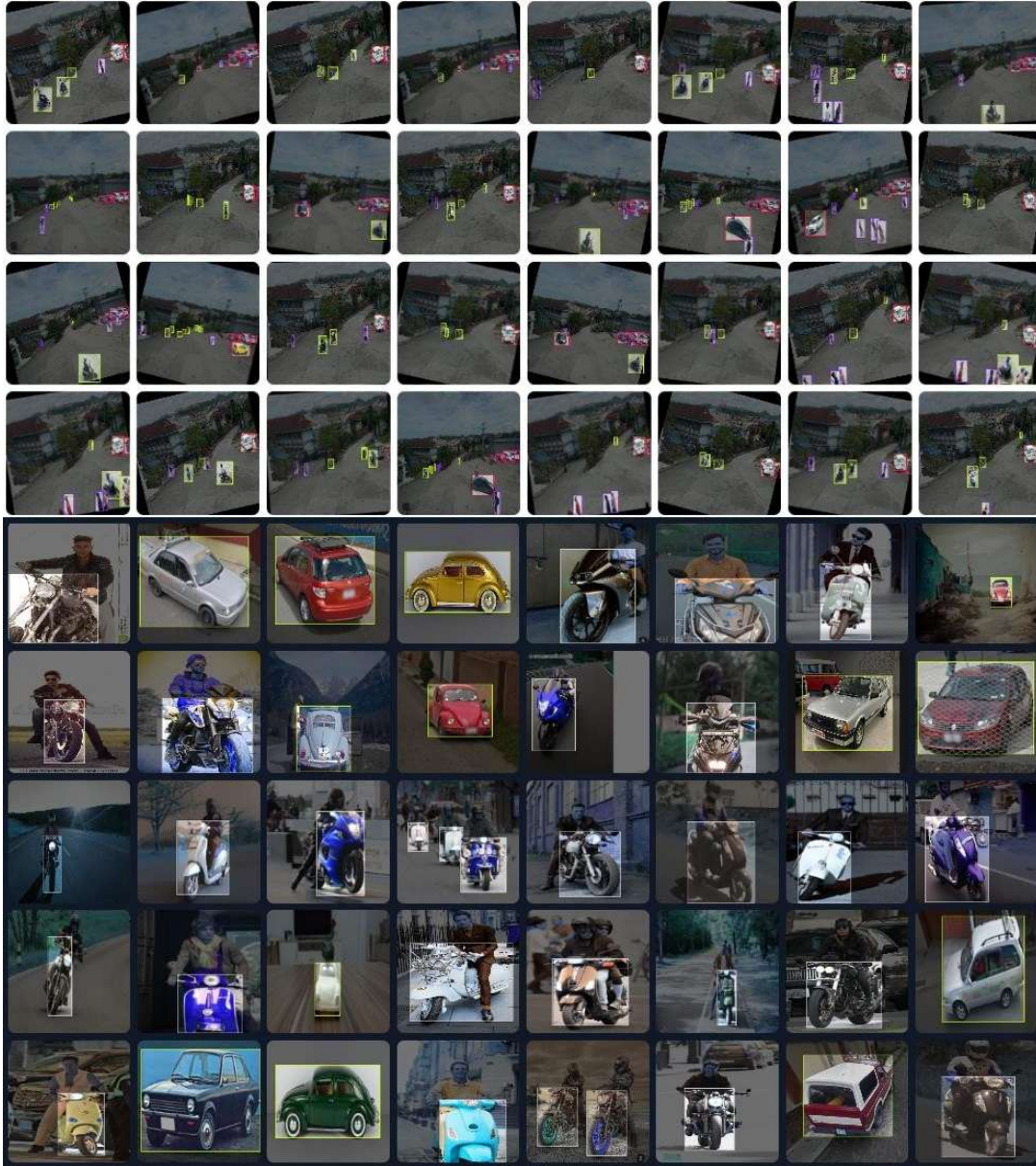


Gambar 3.2 Pelabelan Data

Setiap objek yang masuk ke dalam salah satu kategori tersebut diberi bounding box yang mengelilingi objek tersebut. Bounding box ini merupakan kotak pembatas yang digunakan untuk menunjukkan posisi dan ukuran objek dalam frame. Selain itu, setiap bounding box diberi label sesuai dengan kategori yang relevan, seperti "motor" untuk sepeda motor, "mobil" untuk kendaraan roda empat, dan "manusia" untuk orang yang berada di area parkir. Proses pelabelan ini dilakukan secara manual untuk memastikan bahwa setiap objek terannotasi dengan tepat dan akurat.

Hasil dari proses anotasi ini adalah dataset yang terstruktur dengan baik, dimana setiap frame memiliki informasi bounding box dan label yang sesuai. Dataset ini sangat penting untuk melatih model YOLOv8 karena memberikan data yang jelas tentang posisi dan kategori objek yang perlu dideteksi. Dataset ini kemudian dibagi menjadi tiga bagian untuk keperluan pelatihan dan pengujian model klasifikasi dan deteksi kendaraan. Proporsi pembagian dataset

adalah 70% untuk data latih, 20% untuk data validasi, dan 10% untuk data uji. Kemudian setiap frame akan dilakukan augmentasi dengan dirotasi sebesar negatif & positif 15 derajat. Berikut gambar 3.3 adalah sampel dataset yang dihasilkan pada tahap ini.



Gambar 3.3 Sampel Dataset

Data latih digunakan untuk melatih model agar dapat mengenali dan mengklasifikasikan objek-objek dalam kategori yang telah ditentukan. Data validasi digunakan untuk mengukur kinerja model selama pelatihan dan melakukan penyesuaian parameter jika diperlukan.

Sedangkan data uji digunakan untuk mengevaluasi kinerja akhir model setelah proses pelatihan selesai. Dengan pembagian dataset ini, diharapkan model yang dihasilkan akan memiliki kemampuan yang baik dalam mendeteksi dan mengklasifikasikan kendaraan dan manusia di area parkir.

3.3 Pelatihan Model

Setelah menyelesaikan proses anotasi data, langkah berikutnya adalah melatih model YOLOv8 menggunakan dataset yang telah teranotasi. Pada tahap ini, data yang digunakan terdiri dari 70% data latih, 20% data uji validasi serta 10% data uji, sesuai dengan yang telah direncanakan pada bab 2. Data latih dan data uji diatur dalam direktori terpisah. Data latih berisi 70% dari total frame yang telah dianotasi, sedangkan data validasi berisi 20% dan data uji berisi 10% sisanya. Dataset ini berisi berbagai rekognisi frame per frame yang memuat objek motor, mobil dan manusia dengan anotasi bounding box. Gambar 3.4 menunjukkan hasil yang dihasilkan YOLOv8 setelah training dataset dengan nilai epoch 70.

```

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
69/70   2.16G    1.058     0.5296    0.9686    120        640: 100% 46/46 [00:15<00:00, 2.99it/s]
      Class  Images  Instances  Box(P    R          mAP50  mAP50-95): 100% 3/3 [00:00<00:00, 3.75it/s]
      all    69      715      0.891    0.825     0.888   0.497

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
70/70   2.16G    1.06      0.5291    0.9708    116        640: 100% 46/46 [00:14<00:00, 3.20it/s]
      Class  Images  Instances  Box(P    R          mAP50  mAP50-95): 100% 3/3 [00:00<00:00, 3.29it/s]
      all    69      715      0.897    0.826     0.886   0.494

70 epochs completed in 0.357 hours.
Optimizer stripped from runs/detect/train/weights/last.pt, 6.3MB
Optimizer stripped from runs/detect/train/weights/best.pt, 6.3MB

Validating runs/detect/train/weights/best.pt...
Ultralytics YOLOv8.2.42 Python-3.10.12 torch-2.3.0+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 3006233 parameters, 0 gradients, 8.1 GFLOPs
      Class  Images  Instances  Box(P    R          mAP50  mAP50-95): 100% 3/3 [00:01<00:00, 2.52it/s]
      all    69      715      0.857    0.87      0.897   0.51
      manusia  57      233      0.819    0.811     0.831   0.403
      mobil   69      228      0.996    0.985     0.994   0.648
      motor   69      254      0.755    0.815     0.866   0.48

Speed: 0.2ms preprocess, 3.2ms inference, 0.0ms loss, 3.0ms postprocess per image
Results saved to runs/detect/train

```

Gambar 3.4 Hasil Training Model YOLOv8

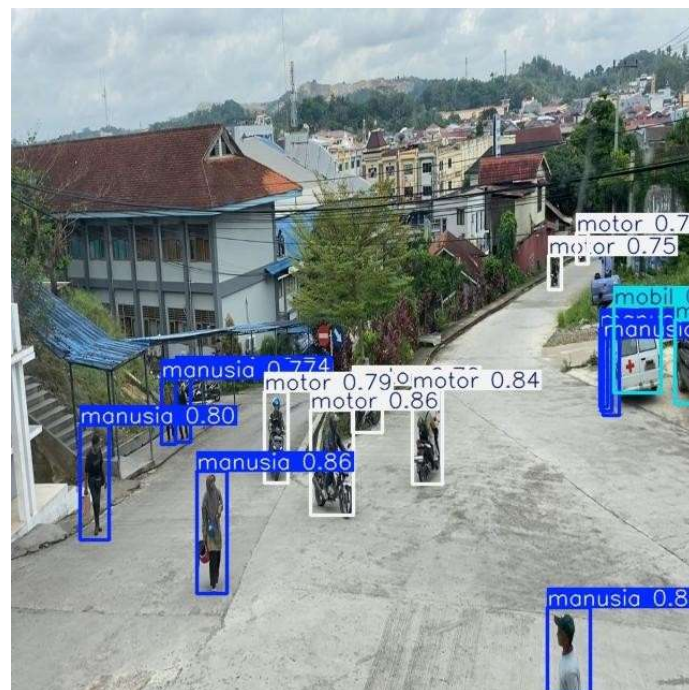
Model YOLOv8 dilatih menggunakan *framework* Ultralytics. Konfigurasi pelatihan termasuk parameter-parameter seperti jumlah epoch dan ukuran gambar yang disesuaikan untuk mendapatkan performa terbaik. Proses pelatihan dilakukan pada platform Google Colab yang mendukung GPU untuk mendukung komputasi.

Model yang dilatih mencapai hasil yang cukup memuaskan dengan nilai mAP yang tinggi pada data latih. Secara umum, model ini menunjukkan performa yang baik untuk kelas "mobil" dengan precision dan recall yang tinggi, sementara kelas "motor" memiliki performa yang lebih rendah dibandingkan kelas "mobil". Performa keseluruhan model bisa diterima dengan mAP50 sekitar 89.7%, tetapi masih bisa ditingkatkan terutama untuk kelas "manusia" dan "motor". Detail dari hasil pelatihan data dapat dilihat pada tabel 3.1 berikut.

Tabel 3.1 Hasil Pelatihan Data

Metrik	Nilai (%)
Precision	85.7
mAP50	89.7

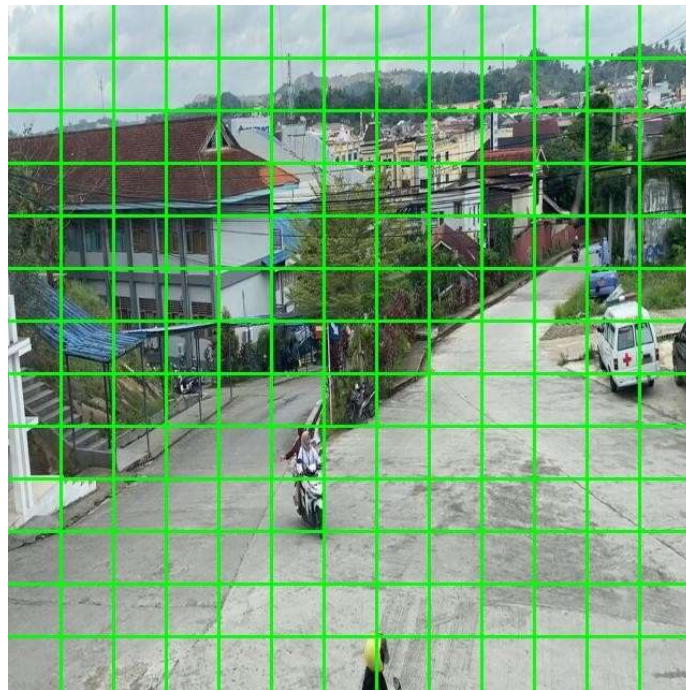
Setelah pelatihan selesai, dilakukan visualisasi hasil deteksi pada beberapa frame dari data uji. Hasilnya menunjukkan bahwa bounding box yang dihasilkan oleh model YOLOv8 secara akurat mengelilingi objek motor, mobil dan manusia. Berikut adalah contoh hasil deteksi model YOLOv8 pada frame dari data uji yang dapat dilihat pada gambar 3.5.



Gambar 3.5 Pendeteksian Objek Dari Hasil Model

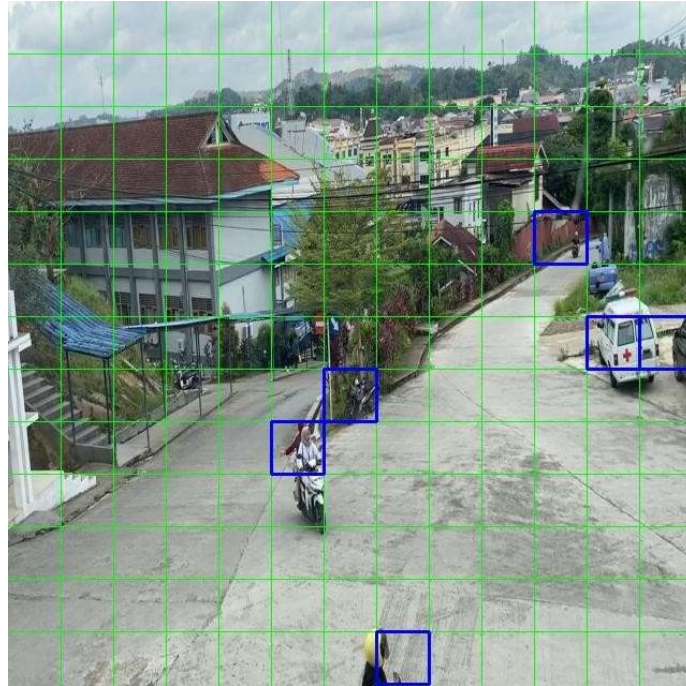
Gambar di atas menunjukkan bahwa model mampu mendeteksi objek motor, mobil dan manusia dengan tingkat akurasi yang cukup tinggi, sesuai dengan anotasi yang diberikan selama tahap pelabelan data. Hasil pelatihan ini menunjukkan bahwa model YOLOv8 yang dilatih dengan dataset yang relevan dan teranotasi dengan baik mampu memberikan performa deteksi yang sesuai dengan tujuan penelitian.

YOLOv8 melakukan tahapan mulai dari pembagian grid hingga menjadi hasil output yang memiliki bounding box serta nilai kepercayaan. Pada tahap ini, grid 13x13 dipilih karena cukup untuk mendeteksi objek dengan berbagai skala objek keluar masuk area parkir. Gambar 3.6 merupakan hasil dari tahap pembagian grid yang dilakukan oleh algoritma YOLOv8.



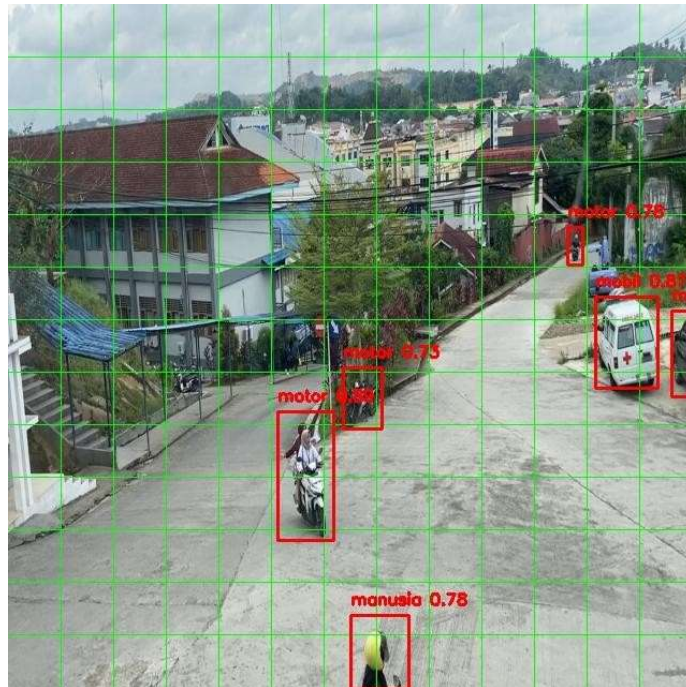
Gambar 3.6 Pembagian Grid 13x13

Tahap berikutnya setelah pembagian grid yaitu prediksi *bounding box* dari grid yang telah dihasilkan. Ukuran *bounding box* menyesuaikan dengan ukuran grid sebelumnya, hasil dari tahap ini dapat dilihat pada gambar 3.7.



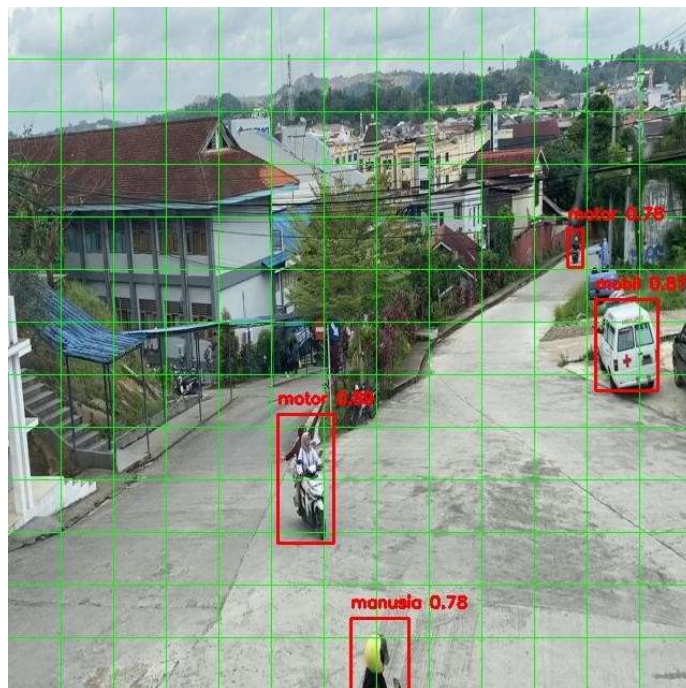
Gambar 3.7 Prediksi Bounding Box

Tahap berikutnya prediksi label yang dilakukan oleh YOLO, setelah mendapatkan *bounding box*, YOLO akan melakukan prediksi terhadap *bounding box* tersebut dengan faktor model yang telah dihasilkan dari training sebelumnya dengan nilai keyakinan algoritma bahwa *bounding box* yang diprediksi berisi objek yang ingin dideteksi dengan hasil pada gambar 3.8.



Gambar 3.8 Bounding Box Dengan Label & Confident

Selanjutnya YOLO melakukan tahap *Non-Maximum Suppression* (NMS) dengan membatasi nilai *confident* minimal dengan nilai 40% agar objek yang dideteksi valid, hal ini disebut penyaringan. Berikut gambar 3.9 menunjukkan proses YOLO dalam tahap NMS.



Gambar 3.9 Tahap Non-Maximum Suppression

Setelah dilakukan penyaringan, YOLO menghasilkan objek-objek yang berhasil dideteksi bersama dengan koordinat bounding box dan kelas objek seperti pada gambar 3.10 berikut.



Gambar 3.10 Tahap Akhir Hasil Deteksi Algoritma YOLOv8

3.4 Evaluasi & Validasi

Pada tahap ini, penulis mengevaluasi kinerja model YOLOv8 menggunakan metrik validasi yang telah ditentukan pada bab sebelumnya, yaitu *Precision* dan *mean Average Precision* (mAP). Pada hasil validasi mendapatkan hasil yang memuaskan *Precision* dan mAP yang cukup memuaskan dan sedikit lebih tinggi dibandingkan hasil data latih. Tabel 3.2 berikut menampilkan hasil evaluasi model yang didapatkan setelah validasi data.

Tabel 3.2 Hasil Validasi Data Model




Metrik	Nilai (%)
Precision	86.5
mAP50	89.8

Dari tabel di atas, terlihat bahwa model YOLOv8 yang dilatih memiliki *precision* yang tinggi, menunjukkan bahwa model dapat akurat dalam mendeteksi objek yang relevan. mAP

juga menunjukkan performa yang baik, terutama pada threshold 0.5 yang sering digunakan sebagai standar evaluasi.

Selain validasi performa model, peneliti juga melakukan validasi manual dengan melakukan perbandingan antara hasil deteksi algoritma YOLOv8 dengan data aktual yang telah terannotasi. Validasi ini dilakukan pada 3 frame acak dari data uji untuk memastikan keakuratan deteksi. Hasil validasi manual menunjukkan bahwa sebagian besar deteksi yang dilakukan oleh model sesuai dengan posisi dan kategori objek yang sebenarnya. Tabel 3.3 berikut menyajikan sampel hasil dari validasi manual.

Tabel 3.3 Validasi Manual

Frame	Detik	Objek Aktual	Deteksi Model	Status
	42	1 manusia, 2 mobil, 3 motor	1 manusia, 2 mobil, 3 motor	Valid
	1:12	1 manusia, 2 mobil, 5 motor	1 manusia, 2 mobil, 5 motor	Valid
	2:23	2 manusia, 3 motor, 2 mobil	4 motor, 2 mobil	Tidak Valid

Dari tabel di atas, dapat dilihat bahwa model YOLOv8 umumnya memberikan deteksi yang tepat, namun ada beberapa kasus di mana terjadi kesalahan deteksi. Kesalahan deteksi pada validasi manual biasanya terjadi pada kondisi objek yang tumpang tindih. Analisis ini

memberikan wawasan tentang area yang perlu diperbaiki dalam dataset atau model untuk meningkatkan akurasi lebih lanjut.

Selain evaluasi performa dari klasifikasi, penulis juga memodifikasi program menggunakan model YOLOv8 agar dapat menghitung jumlah kendaraan keluar masuk area parkir UMKT. Dengan memanfaatkan OpenCV, area masuk parkir dibuat sebuah garis yang menjadi tolak ukur kendaraan tersebut melintas masuk atau keluar. Dari garis tersebut program mampu menjumlahkan kendaraan keluar masuk seperti yang dapat dilihat pada gambar 3.11 berikut.



Gambar 3.11 Hasil Deteksi & Perhitungan Kendaraan Keluar Masuk

Hasil perhitungan jumlah kendaraan keluar masuk pada program ini masih dapat dikembangkan. Dilihat dari perhitungannya, program ini masih sedikit kesulitan menghitung kendaraan dikarenakan tidak menggunakan algoritma tambahan seperti DeepSORT. Masih terdapat beberapa kesalahan yang dihasilkan oleh program perhitungan jumlah kendaraan keluar masuk yang masih sangat mungkin untuk dikembangkan kedepannya.

Hasil evaluasi dan validasi menunjukkan bahwa model YOLOv8 yang telah dilatih memiliki performa yang baik dalam mendeteksi objek motor, mobil dan manusia. Hasil model

standar klasifikasi kendaraan pada YOLOv8 mendapatkan tingkat *confidence* yang tinggi dibandingkan tingkat *confidence* klasifikasi kendaraan pada hasil perhitungan kendaraan keluar masuk, hal ini dikarenakan pada model standar YOLOv8 tidak terganggu oleh kodingan tambahan untuk perhitungan jumlah kendaraan keluar masuk. Ini memberikan dasar yang kuat untuk implementasi pada penelitian berikutnya.