

IMPLEMENTASI TRANSFER LEARNING DENGAN FINE-TUNING PADA DETEKSI OBJEK MULTI-KELAS MENGGUNAKAN YOLO (STUDI KASUS CAR FREE DAY JALAN EL TARI KOTA KUPANG)

Hendrikus Samuel Ola Sogen^{1*}, Erna Rosani Nubatonis², Hasibun Asikin³

^{1,2,3}Teknik Informatika, STIKOM Uyelindo Kupang

^{1*}riosogen1@gmail.com, ²ernarosaninubatonis@gmail.com, ³doktorbento@gmail.com

DOI: <https://doi.org/10.46880/mtk.v12i2.5923>

ABSTRACT

Object detection is a computer vision technology used to recognize and determine the location of objects in images or videos. This study aims to implement the transfer learning method with fine-tuning on the YOLOv8m model to detect multi-class objects consisting of persons, vehicles, and umbrellas in the Car Free Day environment on El Tari Street, Kupang City, as well as to develop a web-based object detection system capable of automatically detecting objects in images and videos. The research dataset consisted of 315 images obtained through field documentation and annotated using Roboflow, which was then increased to 757 images through the augmentation process before being divided into training, validation, and testing datasets. The YOLOv8m pretrained model based on the COCO dataset was trained using Google Colab for 100 epochs with the transfer learning and fine-tuning approach. The results showed that the model achieved a precision of 0.903, a recall of 0.801, an mAP50 of 0.868, and an mAP50-95 of 0.625. In addition, the developed web-based system was able to automatically detect objects in images and videos and display bounding boxes, confidence scores, object counts, detection result graphs, and model evaluation metrics. The results of the study indicate that the application of fine-tuning to YOLOv8m is capable of improving the model's adaptability to the characteristics of the local Car Free Day environment, thereby potentially supporting more effective public activity monitoring.

Keywords: Car Free Day, Computer Vision, Deteksi Objek, Fine-Tuning, Transfer Learning, YOLOv8

I. PENDAHULUAN

Deteksi objek merupakan salah satu kemampuan utama dalam pengolahan citra digital yang berfungsi untuk mengenali dan menentukan posisi berbagai objek dalam suatu gambar atau video. Teknologi ini menjadi semakin penting ketika diaplikasikan pada lingkungan nyata yang memiliki dinamika tinggi, seperti ruang publik atau area dengan aktivitas masyarakat yang beragam [1]. Variasi kondisi pencahayaan, pergerakan objek, serta kerumunan orang sering kali menjadi tantangan dalam menghasilkan deteksi yang akurat. Oleh karena itu, diperlukan metode yang mampu beradaptasi dengan karakteristik lingkungan tersebut, terutama ketika model deteksi akan digunakan untuk mengidentifikasi banyak jenis objek sekaligus di area yang kompleks.

Di antara berbagai metode deteksi yang telah dikembangkan, You Only Look Once (YOLO) muncul sebagai salah satu algoritma yang unggul karena kemampuannya mengenali objek secara cepat dan akurat hanya dalam satu kali proses komputasi. Berbagai penelitian sebelumnya telah membuktikan efektivitas algoritma YOLO dalam mendeteksi berbagai jenis objek pada kondisi lingkungan yang berbeda. Penerapan YOLOv5 pada sistem tingkat pintar berbasis Odroid menunjukkan bahwa model ini mampu melakukan deteksi objek secara cepat dan akurat meskipun dijalankan pada perangkat dengan sumber daya terbatas [2]. Sementara itu, penelitian lain menggunakan YOLOv8 untuk mengidentifikasi hingga sepuluh jenis sampah anorganik dalam satu citra, dan hasilnya menunjukkan tingkat akurasi yang sangat tinggi pada proses klasifikasi multi-objek [3].

Hasil-hasil penelitian tersebut menunjukkan bahwa YOLO merupakan algoritma deteksi objek yang

efisien, fleksibel, dan mampu beradaptasi terhadap berbagai konteks penerapan di dunia nyata. Selain digunakan pada perangkat berdaya tinggi, algoritma YOLO juga terbukti mampu diimplementasikan secara efektif pada perangkat dengan sumber daya terbatas. Hasil penelitian menunjukkan bahwa YOLOv5 Nano memiliki tingkat akurasi yang baik dalam mendeteksi objek pada perangkat ESP32-CAM, meskipun membutuhkan waktu pemrosesan yang lebih lama dibandingkan algoritma ringan lainnya seperti FOMO [4]. Di sisi lain, penelitian mengenai evaluasi performa YOLOv8 menunjukkan bahwa variasi pencahayaan dan kondisi lingkungan secara signifikan memengaruhi akurasi deteksi, di mana model YOLOv8m memberikan hasil terbaik pada kondisi pencahayaan optimal, tetapi mengalami penurunan performa pada pencahayaan rendah [5].

Temuan tersebut menegaskan pentingnya proses fine-tuning agar model YOLO dapat beradaptasi dengan kondisi lingkungan nyata dan menghasilkan deteksi objek yang konsisten di berbagai situasi. Perkembangan mutakhir dalam mengoptimalkan algoritma ini terus dilakukan melalui berbagai pendekatan struktural dan pengayaan data lapangan. Niu dkk. mengembangkan metode YOLOv8-CGRNet yang menggabungkan modul penurun sampel CGNet dan arsitektur Res2Net untuk meningkatkan pemahaman konteks spasial objek pada perangkat mobile, di mana model ini mampu mengoptimalkan mAP50 hingga 4,3% pada dataset publik [6]. Pendekatan lain oleh Kumar dan Muhammad berfokus pada penggabungan data (data merging) dari dataset kondisi cuaca ekstrem seperti ACDC dan DAWN untuk melatih model YOLOv8 secara tangguh, yang membuktikan bahwa pemuatan variasi fitur cuaca

secara masif mampu meningkatkan adaptabilitas model dibandingkan bobot standar [7].

Kendati demikian, penelitian optimasi struktur tersebut masih mengalami kendala berupa kompleksitas beban komputasi serta penurunan akurasi pada objek mikro [6], sementara model penggabungan data cuaca berkendara masih rentan mengalami kegagalan deteksi pada objek berjarak jauh dengan resolusi visual yang kecil [7]. Melalui tinjauan literatur terkini tersebut, terdapat celah (gap) penelitian yang ingin ditutupi melalui studi kasus ini. Penelitian terdahulu umumnya menguji generalisasi model pada dataset lalu lintas berkendara standar atau dataset publik global, namun belum mengeksplorasi tantangan spesifik berupa kerumunan padat pejalan kaki dan lapak temporal di ruang publik lokal, seperti kegiatan Car Free Day (CFD) Jalan El Tari Kota Kupang. Lingkungan CFD ini memiliki masalah penelitian yang lebih spesifik, seperti tantangan saling menutupi antar-objek (occlusion) yang tinggi akibat padatnya pengunjung, ukuran objek kecil (small objects) seperti payung dan lapak pedagang yang berada jauh dari sudut pandang kamera, serta variasi pencahayaan yang berubah dinamis seiring transisi waktu pelaksanaan CFD dari pagi hingga siang hari.

Berdasarkan berbagai penelitian tersebut, dapat disimpulkan bahwa YOLO memiliki potensi besar dalam penerapan deteksi objek multi-kelas pada berbagai kondisi dan perangkat. Namun, setiap lingkungan memiliki karakteristik unik yang dapat memengaruhi tingkat akurasi deteksi, seperti pencahayaan, kepadatan objek, dan variasi latar belakang. Oleh karena itu, dibutuhkan proses transfer learning dan fine-tuning untuk menyesuaikan model dengan kondisi data lokal agar performa deteksi tetap optimal. Penelitian ini berfokus pada implementasi transfer learning dengan fine-tuning menggunakan algoritma YOLO untuk mendeteksi objek multi-kelas di area Car Free Day Jalan El Tari Kota Kupang. Penelitian ini memberikan beberapa kontribusi nyata secara eksplisit untuk membedakannya dari penelitian sebelumnya.

Pertama, naskah ini menyusun dataset lokal baru dari dokumentasi lapangan CFD Jalan El Tari yang dianotasi mandiri lewat Roboflow untuk mendeteksi kelas manusia (*person*), kendaraan lapak jualan (*vehicle*), dan payung (*umbrella*). Kedua, dilakukan optimasi kinerja pada arsitektur YOLOv8m dengan mengintegrasikan algoritma object tracking ByteTrack serta virtual counting area berbentuk huruf U guna menstabilkan pelacakan objek dinamis dan mereduksi perhitungan berulang (*overcounting*). Ketiga, model terlatih ini diimplementasikan langsung ke dalam antarmuka sistem berbasis web menggunakan Python, sehingga pengguna dapat mengunggah file gambar maupun video dan memperoleh visualisasi grafik hasil deteksi beserta tabel evaluasi model secara real-time. Harapan penelitian ini menghasilkan sistem deteksi objek yang adaptif, akurat, serta mampu beradaptasi dengan kondisi lingkungan nyata di lapangan.

II. KAJIAN TERKAIT

Citra Digital

Citra digital merupakan representasi dua dimensi dari suatu objek atau adegan nyata yang tersusun atas sekumpulan elemen gambar atau piksel (picture element). Komputer tidak mengenali citra secara visual, melainkan mengolahnya sebagai matriks nilai numerik. Dalam pengolahan citra digital, terdapat tiga jenis citra utama, yaitu: citra warna (RGB) yang memiliki tiga kanal warna dasar (merah, hijau, biru), citra grayscale yang hanya memiliki satu kanal intensitas keabuan (dengan rentang nilai 0 hingga 255), serta citra biner yang merupakan hasil penyederhanaan menjadi dua nilai piksel, yakni 0 (hitam) dan 1 (putih) [8].

Deep Learning dan Convolutional Neural Network (CNN)

Deep learning adalah cabang machine learning yang menggunakan arsitektur jaringan saraf berlapis (multi-layer network) untuk meniru cara kerja otak manusia dalam mempelajari representasi data kompleks secara otomatis tanpa desain fitur manual [9]. Salah satu arsitektur deep learning yang paling mendominasi pemrosesan citra adalah Convolutional Neural Network (CNN).

CNN mengekstraksi fitur secara hierarkis melalui tiga lapisan utama [10].

1. Lapisan Konvolusi (*Convolutional Layer*)

Berfungsi mengekstraksi ciri citra masukan menggunakan filter/kernel untuk mendeteksi pola dasar seperti garis dan tekstur untuk membentuk peta fitur (feature map).

2. Lapisan Pooling (*Pooling Layer*)

Berfungsi mereduksi dimensi ukuran spasial dari peta fitur untuk mengefisienkan komputasi jaringan tanpa menghilangkan informasi penting, seperti melalui teknik max pooling atau average pooling.

3. Lapisan *Fully Connected* (FC)

Merupakan lapisan akhir yang menggabungkan seluruh fitur hasil ekstraksi menjadi keluaran klasifikasi akhir yang sesuai.

Deteksi Objek Multi-Kelas

Object detection merupakan sub-bidang computer vision yang berfokus pada pendeteksian dan pengenalan lokasi suatu objek secara otomatis menggunakan algoritma cerdas [11]. Sementara itu, Multi-Class Detection merupakan pendekatan di mana sistem dirancang untuk mengenali serta mengklasifikasikan lebih dari dua jenis objek secara bersamaan dalam satu proses pemrosesan, bahkan saat kondisi lingkungan cukup kompleks dan objek saling tumpang tindih [12].

Algoritma You Only Look Once (YOLO) dan YOLOv8

You Only Look Once (YOLO) adalah algoritma berbasis CNN untuk pendeteksian objek real-time secara menyeluruh (*single-shot detection*). YOLO membagi citra masukan ke dalam grid berukuran $S \times S$, dan setiap grid langsung memprediksi keberadaan objek menggunakan kotak pembatas (bounding box) beserta tingkat keyakinan (*confidence score*) [1].

Sejak diperkenalkan pada tahun 2015, algoritma ini terus berevolusi. Pada tahun 2023, diperkenalkan varian YOLOv8 yang menerapkan pendekatan anchor-free detection, yang menjadikan prediksi kotak pembatas

sangat efisien karena menghilangkan ketergantungan pada anchor box [13]. Arsitektur YOLOv8 memanfaatkan modul C2f pada blok backbone untuk ekstraksi fitur yang solid, serta menerapkan struktur decoupled head pada bagian keluarannya untuk memisahkan jalur tugas klasifikasi objek dengan regresi kotak pembatas, sehingga memberikan keseimbangan optimal antara kecepatan komputasi dan akurasi [14].

Transfer Learning dan Fine-Tuning

Transfer learning memungkinkan sebuah model untuk memanfaatkan pengetahuan yang dipelajari pada satu ranah data (seperti model pretrained berbasis dataset skala besar) agar dapat diaplikasikan pada dataset atau domain baru yang memiliki relasi serupa [15]. Pendekatan ini mengatasi masalah keterbatasan komputasi dan waktu training yang masif [16].

Fine-tuning melengkapi metode tersebut dengan melakukan pelatihan ulang pada sebagian atau keseluruhan parameter lapisan pada jaringan secara perlahan, biasanya dengan nilai learning rate yang lebih kecil. Penyesuaian presisi ini akan membuat model beradaptasi penuh terhadap fitur spesifik dari lingkungan lokal yang baru [17].

III. METODE

Penelitian ini dilaksanakan di area Car Free Day (CFD) Jalan El Tari, Kota Kupang, Nusa Tenggara Timur, dengan durasi penelitian terhitung dari bulan September 2025 hingga Juli 2026.

Bahan dan Alat Penelitian

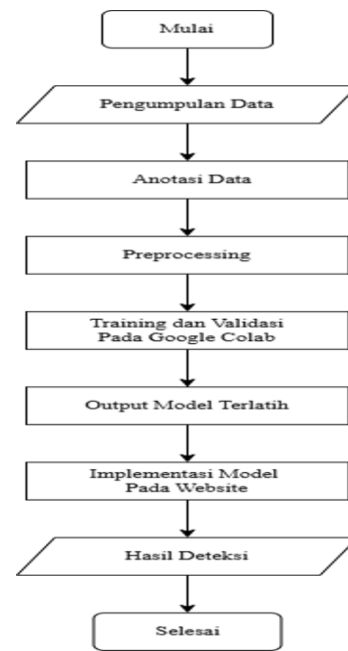
Bahan utama yang digunakan meliputi data visual berupa gambar dan video kegiatan CFD yang direkam secara langsung, serta dataset Common Objects in Context (COCO) yang difungsikan sebagai pre-trained data pada model YOLO sebelum tahap penyesuaian lokal. Perangkat keras yang digunakan untuk proses perancangan meliputi laptop dengan prosesor AMD Ryzen 3 3250U, RAM 8 GB, dan media penyimpanan SSD 512 GB, serta smartphone Redmi 10 untuk akuisisi data visual. Sementara itu, perangkat lunak yang dipergunakan mencakup sistem operasi Windows 11, bahasa pemrograman Python dengan Visual Studio Code, serta platform web Roboflow untuk keperluan anotasi citra.

Teknik Pengumpulan Data

Pengumpulan data lapangan di area CFD dilakukan melalui tahapan berikut:

1. Observasi, dilakukan untuk mengidentifikasi karakteristik lingkungan serta memetakan kelas objek target yang spesifik, yaitu manusia (person), kendaraan lapak (vehicle), dan payung (umbrella).
2. Wawancara, bertujuan memperoleh informasi terkait puncak waktu keramaian di lokasi guna menentukan waktu pengambilan data yang representatif terhadap kondisi sepadat mungkin.
3. Pengambilan Data, dokumentasi secara langsung melalui perekaman gambar dan video menggunakan kamera smartphone saat kegiatan publik berlangsung.

Prosedur Penelitian



Gambar 1. Flowchart Penelitian

Berdasarkan alur flowchart penelitian pada Gambar 1, maka dapat dijelaskan sebagai berikut.

1. Pengumpulan Data

Data berupa gambar kegiatan Car Free Day (CFD) di Jalan El Tari Kota Kupang dikumpulkan menggunakan kamera smartphone pada berbagai kondisi lingkungan. Dataset terdiri atas tiga kelas objek, yaitu person (manusia), vehicle (kendaraan yang digunakan sebagai lapak), dan umbrella (payung).

2. Anotasi Data

Dataset dianotasi menggunakan platform Roboflow dengan membuat bounding box dan memberikan label pada setiap objek sesuai dengan kelas yang telah ditentukan.

3. Preprocessing

Tahap preprocessing meliputi penyesuaian ukuran gambar (resizing), augmentasi data, serta pembagian dataset menjadi data training, validation, dan testing untuk meningkatkan kualitas dan variasi data. Untuk menyamakan resolusi visual dan mengoptimalkan komputasi memori, seluruh citra melalui tahap resize ke dimensi standar 640 x 640 piksel. Sebelum dilakukan augmentasi, total dataset 315 gambar dibagi secara statis dengan proporsi 70% untuk data pelatihan (training), 20% untuk data validasi (validation), dan 10% untuk data pengujian (testing). Pembagian ini mengalokasikan 221 citra training, 63 citra validation, dan 31 citra testing. Selanjutnya, augmentasi data dilakukan lewat platform Roboflow untuk memperbanyak variasi data latih secara acak dengan menghasilkan 3 keluaran gambar baru per satu contoh sampel pelatihan (outputs per training example: 3). Parameter augmentasi terkontrol yang diterapkan secara eksklusif hanya meliputi pembalikan horizontal (horizontal flip) guna melatih sensitivitas model terhadap arah pergerakan objek pejalan kaki maupun lapak pedagang, serta pemotongan skala (crop) dengan batas transformasi antara 0%

minimum zoom hingga 23% maximum zoom untuk mereplikasi variasi jarak kedekatan kamera terhadap objek target di lapangan. Melalui implementasi kedua teknik augmentasi statis ini, jumlah data latih meningkat secara signifikan dari 221 gambar menjadi 663 gambar, sehingga total keseluruhan dataset berkembang menjadi 757 gambar. Sebaliknya, kelompok data validasi (63 gambar) dan data pengujian (31 gambar) dipertahankan tetap konsisten tanpa augmentasi untuk memastikan penilaian akurasi berjalan objektif.

Tabel 1. Distribusi Kelompok Data dan Jumlah Citra Dataset

No	Kelas Objek	Kelompok Data	Jumlah Citra Sebelum Augmentasi	Jumlah Citra Setelah Augmentasi
1	Person	Data Training	221	663
2	Umbrella	Data Validation	63	63
3	Vehicle	Data Testing	31	31
		Total	315	757

4. Training dan Validasi pada Google Colab

Dataset diekspor dari Roboflow dalam bentuk API dan dimuat langsung ke dalam Google Colab untuk proses transfer learning serta fine-tuning menggunakan model dasar YOLOv8m.pt yang berbasis COCO Dataset. Pelatihan dijalankan selama 100 epoch menggunakan dukungan hardware akselerasi GPU Tesla T4. Proses pelatihan dikonfigurasi dengan menetapkan ukuran tumpukan (batch size) sebesar 8 citra per iterasi. Pembaruan bobot parameter model dikontrol penuh oleh optimizer Stochastic Gradient Descent (SGD) dengan nilai laju pembelajaran awal (initial learning rate) ditetapkan secara presisi pada angka 0,01.

Penurunan laju pembelajaran diatur berkala menggunakan skema learning rate schedule cosine annealing hingga laju pembelajaran akhir menyentuh batas 0,01 dari nilai awal pada epoch ke-100 agar konvergensi konvolusi jaringan berjalan stabil. Selama pengolahan berlangsung, fungsi on-the-fly augmentation bawaan Ultralytics seperti teknik mosaic data augmentation (menggabungkan empat gambar acak di memori) aktif secara otomatis berdampingan dengan parameter preprocessing yang dikirim dari Roboflow. Kinerja model dipantau melalui grafik loss dan dievaluasi secara kuantitatif menggunakan metrik Precision, Recall, Mean Average Precision pada threshold IoU konstan 0,5 (mAP50), serta mAP pada rentang batas 0,5 hingga 0,95 dengan kenaikan langkah sebesar 0,05 (mAP50-95). Metrik mAP50 bertindak sebagai indikator kelayakan utama dalam menilai ketepatan model mengenali keberadaan objek multi-kelas secara keseluruhan, sedangkan mAP50-95 mengukur tingkat presisi penempatan koordinat kotak pembatas model terhadap objek nyata di lapangan.

5. Output Model Terlatih

Proses pelatihan menghasilkan model bobot terbaik

bernama best.pt yang dipilih secara otomatis sebagai model akhir berdasarkan pencapaian nilai tertinggi pada metrik mAP50 selama fase pelatihan dan validasi berlangsung. Sistem secara cerdas mengamankan parameter bobot optimal ini dan mengabaikan model epoch terakhir (last.pt) guna mencegah terjadinya penurunan performa deteksi akibat overfitting.

6. Implementasi Model Pada Website

Model visual terbaik (best.pt) yang telah lulus uji generalisasi kemudian ditransfer dan diintegrasikan ke dalam arsitektur sistem berbasis website. Sistem interface ini dibangun menggunakan bahasa pemrograman Python dengan Visual Studio Code sebagai editor teks utamanya, sehingga sistem siap menerima instruksi interaktif dari pengguna untuk memproses pengujian deteksi objek.

7. Hasil Deteksi

Tahap akhir menampilkan hasil deteksi menggunakan model best.pt berupa visualisasi bounding box, label kelas, confidence score, jumlah masing-masing objek yang terdeteksi, serta grafik hasil deteksi. Pada pengoperasiannya di dalam website, batas ambang keyakinan minimum (confidence threshold) dikonfigurasi secara ketat sebesar 0,25, di mana kotak pembatas hanya akan diproduksi jika skor keyakinan model berada di atas 25%. Guna mengeliminasi redundansi kemunculan kotak pembatas yang saling tumpang tindih pada objek manusia yang sama di area kerumunan padat, diaplikasikan metode pasca-pemrosesan Non-Maximum Suppression (NMS) dengan IoU threshold NMS dikonfigurasi sebesar 0,7.

Prosedur NMS ini menekan visualisasi kotak pembatas lain yang memiliki nilai overlap melebihi 70% dari kotak utama yang ber-score tertinggi sehingga tampilan pengawasan tetap bersih dan presisi. Sistem deteksi objek multi-kelas pada penelitian ini secara konsisten mengidentifikasi tiga kelas objek, yaitu person (manusia), vehicle (kendaraan yang digunakan sebagai lapak), dan umbrella (payung). Khusus untuk pemrosesan file uji berupa video dinamis, model best.pt dikombinasikan dengan metode object tracking ByteTrack serta penerapan area hitung virtual berbentuk huruf U (virtual counting area) melewati deteksi garis (line crossing detection) untuk menstabilkan pelacakan identitas objek, mencegah ID switch, serta mereduksi kesalahan perhitungan berulang (overcounting) di lingkungan Car Free Day.

IV. HASIL DAN PEMBAHASAN

Pembuatan Model

Pembuatan model dalam penelitian ini merupakan tahapan awal yang bertujuan untuk membangun sistem deteksi objek multi-kelas menggunakan metode YOLO dengan pendekatan transfer learning dan fine-tuning.

1. Pengumpulan Data: Proses pengambilan dataset pada penelitian ini dilakukan dengan menggunakan kamera ponsel untuk mengambil gambar pada kegiatan Car Free Day di Jalan El Tari Kota Kupang. Pengambilan data mencakup berbagai kondisi lingkungan, seperti tingkat keramaian, variasi pencahayaan, serta sudut pengambilan gambar yang berbeda. Objek yang menjadi fokus dalam pengambilan dataset meliputi (person), kendaraan (vehicle), dan payung (umbrella). Secara

keseluruhan, dataset yang berhasil dikumpulkan berjumlah 315 gambar.

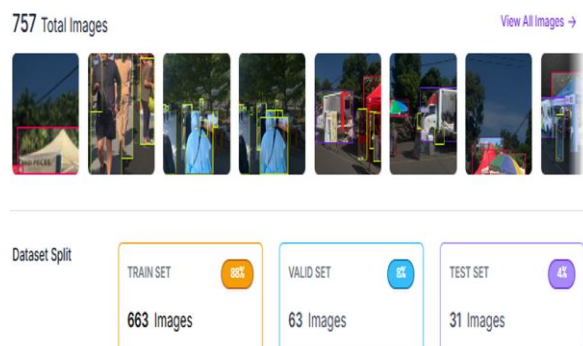
2. *Anotasi Data*: Data yang telah dikumpulkan kemudian dianotasi menggunakan platform Roboflow. Sebelum proses anotasi dilakukan, seluruh dataset terlebih dahulu diunggah ke dalam Roboflow. Selanjutnya dibuat tiga kelas objek yang akan digunakan dalam penelitian, yaitu person, vehicle, dan umbrella. Proses anotasi dilakukan dengan membuat bounding box pada setiap objek yang terdapat pada gambar serta memberikan label sesuai dengan kelas yang telah ditentukan. Hasil dari tahap ini berupa dataset yang telah dianotasi dan siap digunakan pada proses selanjutnya.



Gambar 2. Proses Anotasi Data

1. Preprocessing

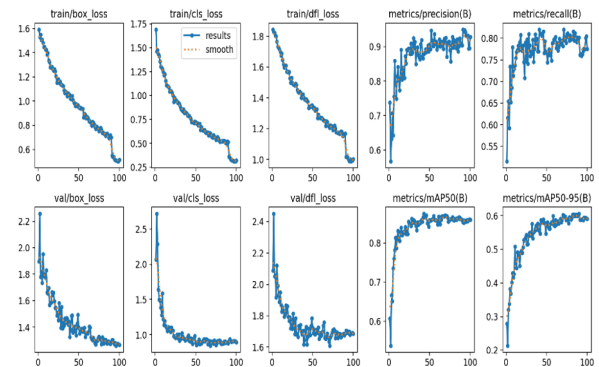
Pada tahap preprocessing dilakukan pengolahan data untuk mempersiapkan dataset sebelum digunakan dalam proses pelatihan model. Tahapan ini meliputi penyesuaian ukuran gambar (resizing), augmentasi data untuk meningkatkan variasi dataset, serta pembagian dataset menjadi data training, validation, dan testing. Sebelum dilakukan proses augmentasi, total dataset yang digunakan berjumlah 315 gambar. Dataset tersebut kemudian dibagi menjadi 221 gambar (70%) untuk data training, 63 gambar (20%) untuk data validation, dan 31 gambar (10%) untuk data testing. Setelah dilakukan proses augmentasi, total dataset meningkat menjadi 757 gambar. Dataset tersebut kemudian dibagi menjadi 663 gambar untuk data training, 63 gambar untuk data validation, dan 31 gambar untuk data testing.



Gambar 3. Split Dataset Setelah Augmentasi

3. *Training dan Validasi pada Google Colab*: Pelatihan model dilakukan menggunakan Google Colab dengan memanfaatkan GPU Tesla T4. Dataset diakses melalui API Roboflow, kemudian model YOLOv8m pretrained berbasis COCO Dataset di-fine-tuning menggunakan

dataset penelitian selama 100 epoch. Selama proses pelatihan, nilai box loss, classification loss, dan distribution focal loss terus menurun, yang menunjukkan bahwa model mampu mempelajari karakteristik objek dengan baik. Perkembangan proses pelatihan diamati melalui grafik perubahan nilai loss dan metrik evaluasi pada setiap epoch.



Gambar 4. Grafik Hasil Training Dan Validasi

Berdasarkan Gambar 4, nilai box loss, classification loss, dan distribution focal loss mengalami penurunan secara bertahap seiring bertambahnya epoch. Sebaliknya, nilai precision, recall, mAP50, dan mAP50-95 menunjukkan tren peningkatan hingga mencapai kondisi yang relatif stabil pada akhir proses pelatihan. Hal ini menunjukkan bahwa model memiliki proses pembelajaran yang baik dan mampu beradaptasi dengan karakteristik dataset lokal tanpa mengalami gejala overfitting yang signifikan. Model terbaik (best.pt) dipilih secara otomatis berdasarkan nilai mAP50 tertinggi selama proses pelatihan dan validasi. Selanjutnya, model divalidasi menggunakan data yang tidak digunakan pada proses pelatihan untuk mengukur kemampuan generalisasi model dalam mendeteksi objek pada data baru.

Tabel 2. Hasil Validasi Model YOLOv8m pada Data Validasi Dataset Car Free Day

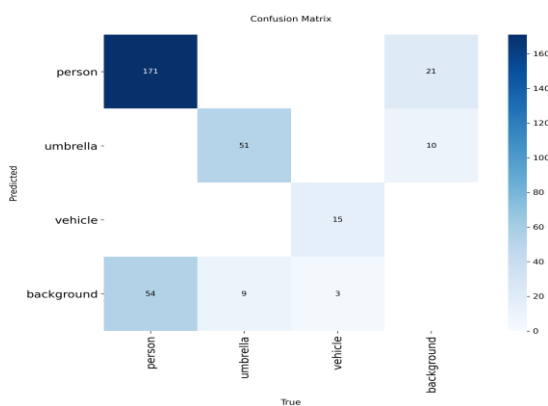
Kelas	Jumlah Sampel	Precisi on (P)	Recall (R)	mAP 50	mAP50-95
all	303	0.903	0.801	0.868	0.625
person	225	0.900	0.721	0.821	0.518
umbrel	60	0.854	0.850	0.936	0.671
la vehicle	18	0.956	0.833	0.848	0.687

Berdasarkan Tabel 2, jumlah sampel pada data validasi menunjukkan distribusi yang tidak seimbang, di mana kelas person memiliki jumlah objek paling banyak, yaitu 225 objek, sedangkan kelas umbrella dan vehicle masing-masing hanya terdiri atas 60 dan 18 objek. Perbedaan jumlah sampel ini dapat memengaruhi proses pembelajaran dan evaluasi model, terutama pada kemampuan generalisasi untuk setiap kelas objek.

Berdasarkan hasil validasi, model memperoleh nilai precision sebesar 0,903, recall sebesar 0,801, mAP50 sebesar 0,868, dan mAP50-95 sebesar 0,625. Nilai precision yang tinggi menunjukkan bahwa sebagian besar objek yang dideteksi oleh model merupakan objek yang benar, sedangkan nilai recall menunjukkan bahwa

model mampu menemukan sebagian besar objek yang terdapat pada data pengujian. Nilai mAP50 sebesar 0,868 menunjukkan bahwa model memiliki kemampuan deteksi yang baik pada threshold IoU 0,5, sedangkan nilai mAP50-95 sebesar 0,625 menunjukkan bahwa model masih mampu mempertahankan performa deteksi pada berbagai tingkat threshold IoU yang lebih ketat. Berdasarkan evaluasi per kelas, objek umbrella memperoleh nilai mAP50 tertinggi sebesar 0,936, diikuti oleh vehicle sebesar 0,848 dan person sebesar 0,821. Tingginya performa pada kelas umbrella dan vehicle menunjukkan bahwa kedua kelas memiliki karakteristik visual yang relatif mudah dibedakan oleh model. Sebaliknya, performa pada kelas person lebih rendah dibandingkan dua kelas lainnya, terutama pada nilai recall sebesar 0,721. Nilai tersebut menunjukkan bahwa masih terdapat sejumlah objek person yang tidak berhasil dideteksi oleh model.

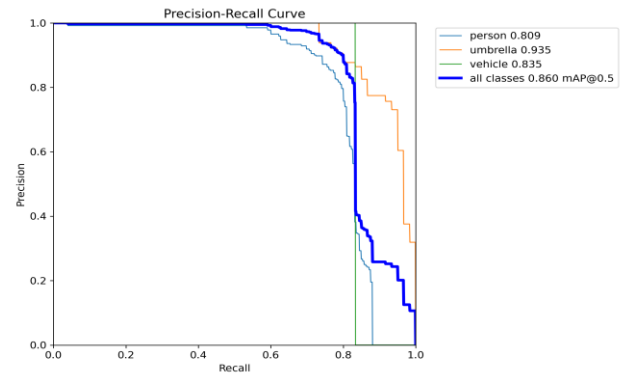
Untuk memperoleh analisis yang lebih mendalam terhadap kemampuan klasifikasi model, dilakukan evaluasi menggunakan confusion matrix.



Gambar 5. Confusion Matrix Hasil Validasi Model

Berdasarkan Gambar 5, model berhasil mengklasifikasikan 171 objek person, 51 objek umbrella, dan 15 objek vehicle dengan benar. Selain itu, masih terdapat sejumlah objek yang tidak berhasil dideteksi oleh model dan diklasifikasikan sebagai background, yaitu sebanyak 54 objek person, 9 objek umbrella, dan 3 objek vehicle. Hasil tersebut menunjukkan bahwa kelas person merupakan kelas yang paling sulit dideteksi oleh model. Sebanyak 24% objek person tidak berhasil dideteksi, sedangkan pada kelas umbrella dan vehicle masing-masing hanya sebesar 15% dan 17%. Kondisi ini menunjukkan bahwa performa deteksi pada kelas person masih dipengaruhi oleh kompleksitas lingkungan Car Free Day. Rendahnya performa pada kelas person disebabkan oleh beberapa faktor, antara lain tingginya tingkat keramaian yang menyebabkan terjadinya saling menutupi antarobjek (occlusion), variasi pose manusia yang sangat beragam, serta keberadaan objek manusia yang berukuran kecil dan berada jauh dari kamera. Pada beberapa kasus, hanya sebagian tubuh manusia yang terlihat sehingga fitur yang diperoleh model menjadi kurang lengkap dan menyebabkan objek gagal dideteksi (false negative). Selain itu, ditemukan beberapa kasus false positive ketika model mendeteksi objek lain yang memiliki

karakteristik visual menyerupai manusia, seperti bagian kendaraan, bayangan, atau objek di sekitar area pengamatan. Meskipun demikian, secara keseluruhan model masih menunjukkan kemampuan deteksi yang baik pada ketiga kelas objek.



Gambar 6. Precision-Recall Curve Hasil Validasi Model

Berdasarkan Gambar 6, model memperoleh nilai Average Precision (AP) sebesar 0,809 pada kelas person, 0,935 pada kelas umbrella, dan 0,835 pada kelas vehicle, dengan nilai mAP 0.5 keseluruhan sebesar 0,860. Nilai tersebut menunjukkan bahwa model memiliki kemampuan deteksi yang baik pada seluruh kelas objek. Kelas umbrella menunjukkan performa terbaik karena memiliki karakteristik visual yang khas dan relatif mudah dibedakan dari objek lain. Kelas vehicle juga menunjukkan performa yang baik karena ukuran objek yang cenderung lebih besar dan memiliki bentuk yang konsisten. Sebaliknya, kelas person memiliki nilai Average Precision yang paling rendah dibandingkan kelas lainnya. Penurunan performa pada kelas person dipengaruhi oleh tingginya jumlah false negative akibat kondisi keramaian, saling menutupi antarobjek (occlusion), serta ukuran objek yang relatif kecil pada beberapa citra. Selain itu, variasi bentuk, pose, orientasi tubuh, dan perbedaan skala manusia menyebabkan model lebih sulit memperoleh fitur yang konsisten dibandingkan pada objek umbrella dan vehicle. Kondisi tersebut menyebabkan beberapa objek person tidak berhasil dideteksi atau memiliki nilai confidence yang rendah sehingga berdampak pada penurunan nilai recall. Secara keseluruhan, hasil evaluasi menunjukkan bahwa metode transfer learning dan fine-tuning pada model YOLOv8m mampu meningkatkan kemampuan adaptasi model terhadap karakteristik lingkungan lokal dan menghasilkan performa deteksi yang baik untuk diterapkan pada sistem pemantauan aktivitas di area Car Free Day Jalan El Tari Kota Kupang.

Implementasi Antarmuka Website

Sistem deteksi objek dikembangkan berbasis website menggunakan bahasa pemrograman Python dengan mengimplementasikan model YOLOv8m (best.pt) yang telah dilatih menggunakan metode transfer learning dan fine-tuning. Sistem menerima masukan berupa gambar (.jpg, .jpeg, .png) maupun video (.mp4, .avi, .mkv), kemudian secara otomatis melakukan deteksi objek dan menampilkan hasil berupa bounding box, label kelas, confidence score, jumlah objek yang terdeteksi, serta grafik hasil deteksi. Sistem dirancang untuk mendeteksi

tiga kelas objek, yaitu person, vehicle, dan umbrella.



Gambar 7. Tampilan Halaman Upload Data Uji

Halaman ini digunakan untuk mengunggah gambar atau video yang akan digunakan sebagai data uji sebelum proses deteksi dilakukan.



Gambar 8. Tampilan Halaman Preview Data Uji

Setelah data berhasil diunggah, sistem menampilkan halaman pratinjau data uji sebagai bentuk konfirmasi visual kepada pengguna. Untuk input berupa citra, sistem menampilkan gambar yang telah diunggah, sedangkan untuk input berupa video, sistem menampilkan pemutar video (video player). Pada halaman ini juga tersedia tombol Mulai Deteksi untuk menjalankan proses deteksi objek serta tombol Stop Detection untuk menghentikan proses deteksi yang sedang berlangsung.



Gambar 9. Tampilan Halaman Hasil Deteksi

Halaman ini merupakan bagian utama dari sistem yang menampilkan hasil proses deteksi objek. Hasil ditampilkan dalam bentuk visualisasi bounding box yang mengelilingi objek terdeteksi, disertai label kelas dan nilai confidence score. Selain visualisasi, sistem juga menampilkan informasi tambahan berupa jumlah objek pada masing-masing kelas, grafik distribusi objek, serta tabel evaluasi model yang mencakup nilai precision, recall, mAP50, dan mAP50-95. Nilai evaluasi yang ditampilkan diperoleh dari model YOLO yang telah dilatih sebelumnya menggunakan metode transfer learning dan fine-tuning, sehingga nilai tersebut bersifat tetap dan tidak berubah terhadap setiap input data uji.

Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode Black Box Testing untuk memastikan seluruh fungsi sistem, seperti unggah data, pratinjau, deteksi objek, dan penanganan kesalahan, berjalan sesuai dengan yang diharapkan.

Tabel 3. Hasil Black Box Testing

N	Skenario	Data	Hasil	Status
o	Pengujian	Input	Pengujian	
1	Upload Gambar	File .jpg, .jpeg, .png	Pratinjau gambar tampil pada area preview	Berhasil
2	Upload Video	File .mp4, .avi, .mkv.	Pemutar video tampil pada area preview	Berhasil
3	Proses Deteksi Gambar	Gambar valid	Hasil deteksi gambar tampil dengan benar	Berhasil
4	Proses Deteksi Video	Video valid	Deteksi video berjalan, objek terdeteksi dan ter-tracking, serta counting (person, umbrella, vehicle) berdasarkan counting line	Berhasil
5	Penghentian Deteksi Video	Menekan tombol Stop Detection	Proses deteksi berhenti	Berhasil
6	Penyimpanan Hasil Gambar	Gambar valid	File hasil tersimpan	Berhasil
7	Penyimpanan Hasil Video	Video valid	File hasil tersimpan	Berhasil

Hasil Pengujian Website

Pada tahap ini dilakukan pengujian sistem untuk mengetahui kemampuan model dalam mendeteksi objek multi-kelas pada data uji berupa gambar dan video.

1. Hasil Deteksi Menggunakan Gambar

Pada tahap ini dilakukan pengujian sistem untuk mengetahui kemampuan model dalam mendeteksi objek multi-kelas pada data uji berupa gambar. Pengujian dilakukan menggunakan data gambar dengan kondisi yang berbeda.

Tabel 4. Hasil Deteksi Gambar

Data Aktual	Hasil Deteksi
Person (6)	
Umbrella(1)	
Vehicle(1)	
	 Person  Umbrella  Vehicle 6 1 1
Person(5)	
Umbrella(3)	
Vehicle(1)	
	 Person  Umbrella  Vehicle 5 3 1
Person(6)	
Umbrella(2)	
Vehicle(1)	
	 Person  Umbrella  Vehicle 6 2 1

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem mampu mendeteksi objek multi-kelas berupa person, umbrella, dan vehicle dengan cukup baik pada berbagai kondisi lingkungan. Pada gambar pertama, sistem berhasil mendeteksi 6 objek person dengan nilai confidence 0.81–0.90, 1 objek umbrella dengan nilai confidence tertinggi sebesar 0.92, serta 1 objek vehicle dengan nilai confidence 0.65. Nilai

confidence yang relatif tinggi pada gambar ini dipengaruhi oleh kondisi objek yang terlihat cukup jelas dan tidak terlalu padat. Pada gambar kedua, sistem berhasil mendeteksi 5 objek person dengan nilai confidence 0.49–0.90, 3 objek umbrella dengan nilai confidence 0.82–0.90, dan 1 objek vehicle dengan nilai confidence 0.81. Variasi nilai confidence pada gambar ini dipengaruhi oleh posisi objek yang saling berdekatan, pencahayaan, ukuran objek, serta adanya beberapa objek yang tertutup sebagian (*occlusion*). Sementara itu, pada gambar ketiga, sistem berhasil mendeteksi 6 objek person dengan nilai confidence 0.70–0.90, 2 objek umbrella dengan nilai confidence 0.88–0.91, dan 1 objek vehicle dengan nilai confidence 0.94. Variasi nilai confidence dipengaruhi oleh kondisi keramaian, tumpang tindih antar objek (*occlusion*), posisi objek terhadap kamera, serta ukuran objek yang berbeda-beda. Meskipun demikian, model tetap mampu mendeteksi objek dengan cukup baik sehingga menunjukkan performa yang efektif pada kondisi lingkungan *Car Free Day* (CFD).

Berdasarkan hasil evaluasi model dan pengujian menggunakan gambar, masih terdapat beberapa keterbatasan pada proses deteksi, terutama pada kelas person. Hal ini ditunjukkan oleh nilai recall kelas person yang lebih rendah dibandingkan kelas umbrella dan vehicle, yang mengindikasikan bahwa masih terdapat beberapa objek manusia yang tidak berhasil dideteksi (*false negative*). Kondisi tersebut kemungkinan disebabkan oleh ukuran objek yang relatif kecil, posisi objek yang berada jauh dari kamera, serta adanya tumpang tindih antarobjek (*occlusion*) pada kondisi keramaian. Selain itu, lingkungan *Car Free Day* yang dinamis dan memiliki latar belakang yang kompleks berpotensi menyebabkan kesalahan deteksi (*false positive*) pada kondisi tertentu. Meskipun demikian, secara keseluruhan model tetap mampu mendeteksi objek multi-kelas dengan baik dan menunjukkan kemampuan yang memadai untuk diterapkan pada lingkungan *Car Free Day*.

2. Hasil Deteksi Menggunakan Video

Pada tahap ini dilakukan pengujian sistem menggunakan data video untuk mengetahui kemampuan model dalam mendeteksi objek multi-kelas pada kondisi keramaian nyata. Pengujian dilakukan menggunakan satu video berdurasi 20 detik yang dipilih untuk menguji kemampuan sistem dalam melakukan deteksi, tracking, dan perhitungan objek secara dinamis.

Proses deteksi video dilakukan dengan memanfaatkan metode object tracking menggunakan ByteTrack sehingga sistem tidak hanya mendeteksi objek pada setiap frame, tetapi juga mampu mempertahankan identitas objek selama objek masih berada di dalam area pengamatan. Selain itu, sistem juga menerapkan virtual counting area berbentuk huruf U untuk menghitung jumlah objek yang melewati batas area deteksi. Objek akan dihitung ketika centroid bounding box melewati garis area yang telah ditentukan.

Tabel 5. Hasil Deteksi Video

Data Aktual	Hasil Deteksi						
Person (19)							
Umbrella (1)							
Vehicle(1)							
	<table><tr><td> Person</td><td> Umbrella</td><td> Vehicle</td></tr><tr><td>19</td><td>1</td><td>1</td></tr></table>	 Person	 Umbrella	 Vehicle	19	1	1
 Person	 Umbrella	 Vehicle					
19	1	1					

Berdasarkan hasil pengujian video, sistem berhasil mendeteksi objek multi-kelas dengan cukup baik pada kondisi keramaian nyata. Hasil pengamatan aktual diperoleh berdasarkan jumlah objek yang melewati virtual counting area pada video pengujian. Berdasarkan hasil pengamatan tersebut, terdapat 19 objek person, 1 objek umbrella, dan 1 objek vehicle yang melewati garis area deteksi. Setelah dilakukan proses deteksi menggunakan model YOLOv8 yang dipadukan dengan metode tracking ByteTrack dan virtual counting area, sistem berhasil mendeteksi seluruh objek dengan jumlah yang sesuai dengan data aktual.

Dengan demikian, hasil deteksi sistem menunjukkan nilai yang sama dengan hasil pengamatan aktual, yaitu 19 objek person, 1 objek umbrella, dan 1 objek vehicle. Keberhasilan sistem dalam menghasilkan jumlah deteksi yang sesuai menunjukkan bahwa penerapan object tracking dan virtual counting area mampu membantu proses perhitungan objek secara lebih stabil serta mengurangi terjadinya perhitungan berulang (overcounting). Selain itu, objek hanya akan dihitung ketika centroid bounding box melewati virtual counting area yang telah ditentukan sehingga objek yang terdeteksi tetapi tidak melewati garis area virtual tidak akan masuk ke dalam proses counting.

Meskipun demikian, proses deteksi video masih dapat dipengaruhi oleh beberapa kondisi, seperti perpindahan objek yang cepat, kondisi saling menutupi antar objek (occlusion), maupun perubahan identitas objek (ID switch) selama proses tracking berlangsung. Kondisi tersebut berpotensi menyebabkan objek terdeteksi lebih dari satu kali atau kehilangan identitas tracking pada frame tertentu. Selain kondisi tersebut, performa deteksi pada video juga dipengaruhi oleh ukuran objek dan kepadatan kerumunan. Objek yang berukuran kecil atau berada pada jarak yang cukup jauh dari kamera memiliki kemungkinan lebih besar untuk tidak terdeteksi (false negative) karena informasi fitur yang diperoleh model menjadi lebih terbatas. Pada kondisi keramaian yang tinggi, objek person juga lebih rentan mengalami occlusion dibandingkan kelas

umbrella dan vehicle, sehingga menyebabkan proses deteksi dan tracking menjadi lebih sulit dilakukan secara konsisten. Selain itu, perubahan pencahayaan dan latar belakang yang kompleks berpotensi menimbulkan kesalahan deteksi (false positive) pada kondisi tertentu. Meskipun demikian, hasil pengujian menunjukkan bahwa model YOLOv8 yang dipadukan dengan metode ByteTrack dan virtual counting area mampu menghasilkan deteksi dan perhitungan objek yang stabil serta menunjukkan performa yang baik pada lingkungan Car Free Day.

V. KESIMPULAN

Penelitian ini berhasil mengimplementasikan sistem deteksi objek multi-kelas berbasis website menggunakan metode transfer learning dan fine-tuning pada model YOLOv8m untuk pemantauan wilayah Car Free Day (CFD) Jalan El Tari Kota Kupang. Kontribusi utama penelitian ditunjukkan oleh performa model yang baik pada data validasi dengan nilai precision sebesar 0,903, recall sebesar 0,801, mAP50 sebesar 0,868, dan mAP50-95 sebesar 0,625. Selain itu, integrasi algoritma ByteTrack dan virtual counting area berbentuk huruf U mampu mendukung proses pelacakan dan perhitungan objek secara otomatis serta mengurangi terjadinya perhitungan berulang (overcounting) pada data video.

Evaluasi per kelas menunjukkan bahwa objek umbrella memperoleh performa terbaik dengan nilai mAP50 sebesar 0,936, diikuti oleh vehicle sebesar 0,848 dan person sebesar 0,821. Performa yang lebih rendah pada kelas person dipengaruhi oleh kondisi keramaian yang menyebabkan terjadinya occlusion, variasi pose manusia yang tinggi, serta ukuran objek yang relatif kecil dan berada jauh dari kamera. Dari sisi implementasi praktis, sistem menunjukkan potensi untuk diterapkan pada pemantauan aktivitas di area publik secara real-time. Pada penelitian ini, proses deteksi dan pelacakan dijalankan menggunakan GPU Tesla T4 dengan vRAM 8 GB dan menghasilkan performa yang baik. Oleh karena itu, perangkat dengan kemampuan komputasi yang setara direkomendasikan

untuk memperoleh kinerja real-time yang optimal. Selain itu, proses pengambilan data sebaiknya dilakukan dari sudut pandang elevasi yang lebih tinggi (high-angle) untuk mengurangi dampak occlusion serta pada kondisi pencahayaan yang memadai agar akurasi deteksi tetap terjaga.

Berdasarkan keterbatasan penelitian ini, beberapa saran untuk penelitian selanjutnya adalah menambahkan dan memperkaya dataset dengan variasi objek, kondisi keramaian, serta pencahayaan yang lebih beragam untuk meningkatkan kemampuan deteksi dan memperluas cakupan kelas objek, mengembangkan model menggunakan parameter yang lebih optimal atau arsitektur YOLO yang lebih mutakhir untuk meningkatkan akurasi, terutama pada objek berukuran kecil dan kondisi occlusion, mengembangkan metode object tracking yang lebih stabil untuk mengurangi kesalahan perhitungan objek pada data video akibat ID switch, occlusion, maupun objek yang keluar dan masuk kembali ke dalam frame dan melakukan evaluasi dan pembaruan model secara berkala agar sistem tetap optimal dan mampu beradaptasi terhadap perubahan kondisi lingkungan serta karakteristik objek.

VI. REFERENSI

- [1] H. Herdianto, H. Hafni, D. Nasution, and S. Ramadhan, "Implementasi Metode YOLO pada Deteksi Objek Manusia," *Method. J. Manaj. Inform. Komputerisasi Akunt.*, vol. 8, no. 2, p. 234–240, 2024, doi: <https://ejurnal.methodist.ac.id/index.php/methomika/article/view/3139>
- [2] M. E. L. Ramadany, N. R. Y. T. Dyiono, and A. Sucipto, "Deteksi Objek Tongkat Pintar Menggunakan YOLOv5 Berbasis Odroid," *J. Ilm. Res. Dev. Student*, vol. 3, no. 1, p. 256–265, 2025, doi: <https://jurnal.alimspublishing.co.id/index.php/jis/article/view/1136/863>
- [3] Visen and N. Charibaldi, "Penerapan Object Detection Menggunakan Deep Learning YOLOv8 untuk Mengidentifikasi Sampah Anorganik (Maksimal Sepuluh Objek) dalam Satu Citra," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 1, p. 195–202, 2025, doi: <https://jtiik.ub.ac.id/index.php/jtiik/article/view/9012>
- [4] F. Firdaus, M. Wibowo, R. Tullah, and W. Ricesa, "Perbandingan Algoritma YOLO dan FOMO untuk Object Detection pada Perangkat ESP32-CAM," *INSECT J. Teknol. Inf. dan Komput.*, vol. 11, no. 1, p. 44–54, 2025, doi: <https://ejournal.umsorong.ac.id/index.php/insect/article/view/4289/2150>
- [5] T. Taufiqurrahman, A. P. Hadi, and R. E. Siregar, "Evaluasi Performa YOLOv8 dalam Deteksi Objek di Depan Kendaraan dengan Variasi Kondisi Lingkungan," *J. Minfo Polgan*, vol. 13, no. 2, p. 1755–1771, 2024, doi: <https://www.jurnal.polgan.ac.id/index.php/jmp/article/view/14228/2894>
- [6] Y. Niu, W. Cheng, C. Shi, and S. Fan, "YOLOv8-CGRNet: A Lightweight Object Detection Network Leveraging Context Guidance and Deep Residual Learning," *Electronics*, vol. 13, no. 1, p. 43, 2024, doi: <https://doi.org/10.3390/electronics13010043>
- [7] D. Kumar and N. Muhammad, "Object Detection in Adverse Weather for Autonomous Driving through Data Merging and YOLOv8," *Sensors*, vol. 23, no. 20, p. 8471, 2023, doi: <https://doi.org/10.3390/s23208471>
- [8] A. Syahfitri, K. Ibutama, and D. Suherdi, "Mendeteksi Tingkat Kesegaran Daging Sapi Menggunakan Metode Transformasi Ruang Warna HIS (Hue, Intensity, dan Saturation)," *J. Sist. Inf. TGD*, vol. 2, no. 6, p. 923–934, 2023, doi: <https://ojs.trigunadharma.ac.id/index.php/jsi/article/download/9040/2294>
- [9] F. Badri, M. T. Alawiy, and E. M. Yuniarno, "Deep Learning Architecture Based on Convolutional Neural Network (CNN) on Animal Image Classification," *J. KURSUSOR*, vol. 12, no. 2, p. 45–54, 2023, doi: <https://kursorjournal.org/index.php/kursor/article/view/349/153>
- [10] P. Harsani, R. Yulianto, and N. M. Sari, "Identifikasi Citra untuk Membedakan Uang Asli dan Palsu Menggunakan Algoritma Convolutional Neural Network (CNN)," *J. Sains dan Teknol.*, vol. 13, no. 2, p. 328–337, 2024, doi: <https://ejournal.undiksha.ac.id/index.php/JST/article/view/83416>
- [11] D. Reinaldo, Y. F. Riti, and N. P. T. Christian, "Perbandingan Performa Model SSD Mobilenet V2 dan FPNLite dalam Deteksi Helm Pengendara Sepeda Motor," *Komputika J. Sist. Komput.*, vol. 13, no. 1, p. 131–138, 2024, doi: <https://ojs.unikom.ac.id/index.php/komputika/article/view/10333>
- [12] S. Diantika, H. Nalatissifa, R. Supriyadi, N. Maulidah, and A. Fauzi, "Implementasi Multi-Class Gradient Boosting untuk Mengklasifikasikan Jenis Hewan pada Kebun Binatang," *ANTIVIRUS J. Ilm. Tek. Inform.*, vol. 17, no. 1, p. 39–46, 2023, doi: <https://ejournal.unisbablitar.ac.id/index.php/antivirus/article/download/2812/1543/9131>
- [13] M. Hussain, "YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection," *Machines*, vol. 11, no. 7, p. 4–14, 2023, doi: <https://www.mdpi.com/2075-1702/11/7/677>
- [14] C. Jia, D. Wang, J. Liu, and W. Deng, "Performance Optimization and Application Research of YOLOv8 Model in Object Detection," *Asian J. Sci. Technol.*, vol. 10, no. 1, p. 325–329, 2024, doi: <https://drpress.org/ojs/index.php/ajst/article/view/19314>
- [15] O. Saputra, D. I. Mulyana, and M. B. Yel, "Implementasi Algoritma Convolutional Neural Network (CNN) untuk Klasifikasi Senjata Tradisional di Jawa Tengah dengan Metode Transfer Learning," *J. Sist. Komput. dan Kecerdasan Buatan*, vol. 5, no. 2, p. 45–52, 2022, doi: <https://jurnal.tau.ac.id/index.php/siskomkb/article/view/282>
- [16] A. Mufidatuzzainiya and M. Faisal, "Penggunaan Teknik Transfer Learning pada Metode CNN untuk Pengenalan Tanaman Bunga," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 10, no. 2, p. 195–206, 2025, doi: <https://ejournal.uinsuka.ac.id/saintek/JISKA/article/download/4559/2912/16081>
- [17] A. G. Pradana, D. R. I. M. Setiadi, and A. R. Muslikh, "Fine-Tuning Model Convolutional Neural Network EfficientNet-B4 dengan Augmentasi Data untuk Klasifikasi Penyakit Kakao," *J. Inf. Syst. Appl. Dev.*, vol. 2, no. 1, p. 1–4, 2024, doi: <https://jurnal.unmer.ac.id/index.php/jisad/article/view/1899>