

TUGAS KELOMPOK

Proyek Deteksi Objek menggunakan YOLO: Eksperimen pada Data Gambar SIP 107 Rancang Bangun Sistem Berbasis AI

NAMA ANGGOTA KELOMPOK (NIM):

- Runi Dwi Jiasta (202304560001)
 - Teresa Kaena Dharmanyoto (202304560014)
 - Caroline Evarista Den Lau (202304560027)
 - Andrew Riza Rafael (202304560030)
-

I. PENDAHULUAN

Proyek ini adalah bagian dari tugas mata kuliah SIP 107: Rancang Bangun Sistem berbasis AI. Tujuan proyek adalah untuk membuat, melatih, dan menguji sebuah model *deep learning* yang dapat mendekripsi dan mengklasifikasikan tiga jenis buah: jeruk, pisang, dan apel. Algoritma yang digunakan model adalah YOLOv8, algoritma deteksi objek *real-time* yang terkenal dengan kecepatan dan akurasinya.

II. METODOLOGI

A. Dataset

Eksperimen kami menggunakan *dataset* dengan judul "[*Fruit Images for Object Detection*](#)" oleh MuhammedBuyukkinaci dari situs Kaggle. *Dataset* ini memiliki 300 gambar yang telah dilabeli dengan 3 kelas, yakni **apple**, **banana**, dan **orange**.

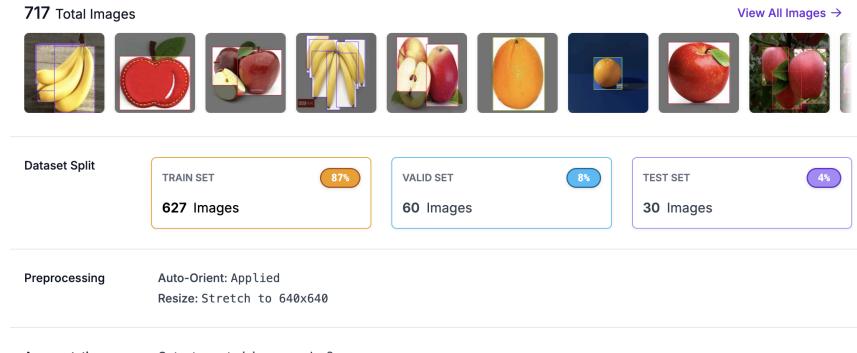
B. Roboflow

Dataset diunggah ke [**Roboflow**](#) untuk manajemen dan *preprocessing*. Data secara otomatis dibagi menjadi 3 set berikut:

- **Training Set** (70%) dengan **210** gambar,
- **Validation Set** (20%) dengan **60** gambar,
- **Testing Set** (10%) dengan **30** gambar.

Preprocessing yang diterapkan merupakan **Auto-Orient** untuk memperbaiki orientasi gambar, serta **Resize** untuk mengubah ukuran gambar menjadi 640 x 640 *pixel*, sesuai dengan rasio optimal YOLOv8.

Augmentasi juga diterapkan pada Training Set untuk meningkatkan variasi data. Diantaranya **Flip (Horizontal)** dan **Brightness** (diantara -15% sampai + 15%). Augmentasi ini menghasilkan 3 *output* per contoh training. Pada akhirnya, terdapat **627 gambar** untuk **Training Set**, dan **717 total gambar** dalam **keseluruhan dataset**.



Gambar 1. Rincian Dataset di Roboflow

C. Environment dan Training

Training dilakukan menggunakan **Google Colab** untuk memanfaatkan GPU (Tesla T4). Library utama yang digunakan ialah Ultralytics (YOLOv8). Model dilatih dengan *pre-trained weights yolov8n.pt*. Proses dijalankan sebanyak 50 epochs, dan memakan waktu 0.191 jam.

```

Epoch      GPU_mem   box_loss   cls_loss   dfl_loss   Instances      Size
50/50     3.44G    0.4162    0.3494    0.9781      3           640: 100% ----- 40/40 3.5it/s 11.5s
          Class     Images   Instances   Box(P       R           mAP50   mAP50-95: 100% ----- 2/2 3.9it/s 0.5s
          all       60        88         0.816     0.902      0.859     0.625
50 epochs completed in 0.191 hours.
Optimizer stripped from /content/drive/MyDrive/Tugas_Fruit_Detection/runs/detect/train/weights/last.pt, 6.2MB
Optimizer stripped from /content/drive/MyDrive/Tugas_Fruit_Detection/runs/detect/train/weights/best.pt, 6.2MB

```

Gambar 2. Penyelesaian 50 Epoch

III. HASIL DAN ANALISIS

Setelah selesai, model dievaluasi dengan *Validation Set* dan *Test Set*.

A. Hasil Training

Model menunjukkan performa yang sangat baik pada 60 gambar di *Validation Set*. Metrik pengukur yang digunakan ialah **mean Average Precision (mAP)** pada **IoU 0.5**.

```

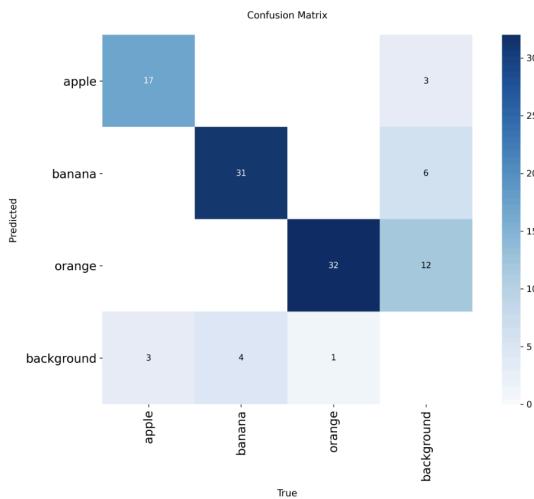
Validating /content/drive/MyDrive/Tugas_Fruit_Detection/runs/detect/train/weights/best.pt...
Ultralytics 8.3.221 ✘ Python-3.12.12 torch-2.8.0+cu126 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 72 layers, 3,006,233 parameters, 0 gradients, 8.1 GFLOPs
          Class     Images   Instances   Box(P       R           mAP50   mAP50-95: 100% ----- 2/2 2.0it/s 1.0s
          all       60        88         0.823     0.875      0.882     0.641
          apple     16        20         0.84      0.85      0.861     0.657
          banana    26        35         0.883     0.866      0.932     0.591
          orange    22        33         0.745     0.909      0.853     0.675
Speed: 0.2ms preprocess, 2.9ms inference, 0.0ms loss, 4.3ms postprocess per image
Results saved to /content/drive/MyDrive/Tugas_Fruit_Detection/runs/detect/train
💡 Learn more at https://docs.ultralytics.com/modes/train

```

Gambar 3. Hasil Training

Secara keseluruhan, model memiliki **akurasi total (mAP50)** sebesar 0.882 atau **88.2%**. Dengan akurasi kelas **apple** sebesar **86.1%**, **banana** **93.2%**, dan **orange** **85.3%**.

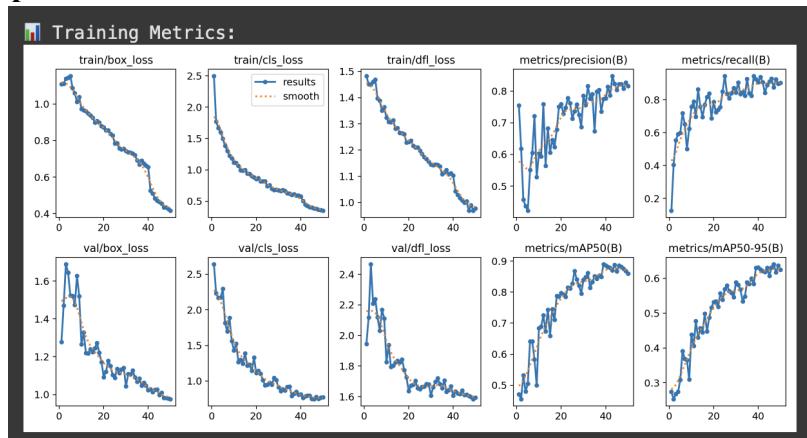
Confusion matrix berikut menunjukkan detail akurasi prediksi untuk setiap kelas:



Gambar 4. *Confusion Matrix* pada *Validation Set*

B. Grafik *Training*

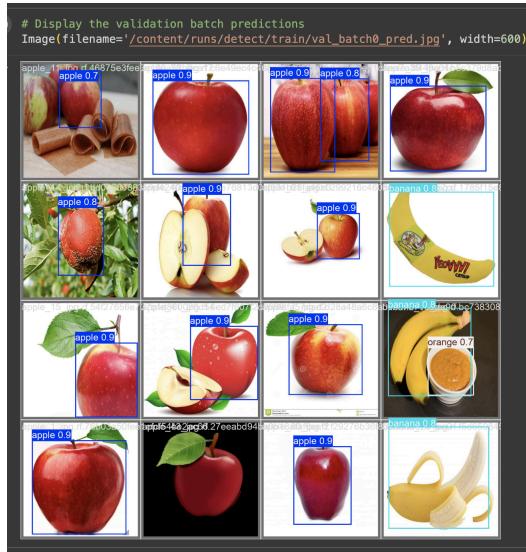
Grafik berikut menampilkan progres *Training* selama 50 epoch. **Loss** (box_loss, cls_loss, dfl_loss) **menurun** secara **konsisten**, sementara metrik **precision**, **recall**, dan **mAP** **meningkat** hingga mencapai **konvergensi** di sekitar **epoch ke-40**.



Gambar 5. Grafik Loss dan Metrik *Training*

C. Hasil Deteksi Visualisasi (Validasi)

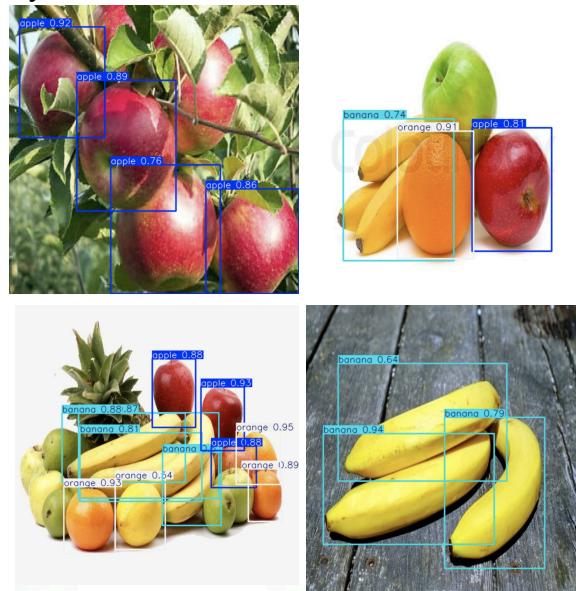
Model mampu menemukan apel, pisang, dan jeruk dengan kotak pelabelan yang akurat, dengan *confidence score* yang tinggi pada gambar validasi.



Gambar 6. Contoh Deteksi pada *Validation Set*

D. Pengujian pada *Test Set*

Model kemudian diuji dengan 30 gambar Test Set yang belum pernah dilihatnya. Pada akhirnya model dapat memproses semua gambar dan menyimpan hasilnya



Gambar 7. Contoh Deteksi pada *Test Set*

IV. KESIMPULAN

Proyek mendeteksi objek buah dengan YOLOv8 telah diselesaikan dengan sukses, dan hasilnya memuaskan. Model yang dilatih menggunakan dataset 300 gambar Kaggle mampu mendeteksi dan mengklasifikasikan dengan sangat akurat tiga jenis buah: apel, pisang, dan jeruk. Pada set validasi, mereka mencapai mAP50 sebesar 88.2%.

Hasil menunjukkan bahwa pisang memiliki akurasi tertinggi (93.2%), diikuti oleh apel (86.0%) dan jeruk (85.3%). Variasi warna dan bentuk yang lebih besar dibandingkan dengan pisang mungkin menjadi penyebab akurasi yang sedikit lebih rendah pada kelas jeruk dan apel.

Model ini dapat diterapkan pada sistem deteksi buah otomatis di supermarket, pasar tradisional, atau sistem sortir buah di industri pertanian. Untuk kemajuan lebih lanjut, dataset dengan variasi kondisi pencahayaan yang lebih besar dan sudut pengambilan gambar yang lebih beragam serta eksplorasi model YOLOv8 yang lebih besar dapat ditambahkan.