

**LAPORAN UAS**  
**“OBJECT DETECTION MENGGUNAKAN METODE YOLO V8”**  
**Pengenalan Pola**



**Ivan Luis Simarmata**

**2008561019**

**PROGRAM STUDI INFORMATIKA**  
**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**  
**UNIVERSITAS UDAYANA**

**2023**

## Manual

Berikut adalah cara untuk menggunakan program “Object Detection YOLO v8”:

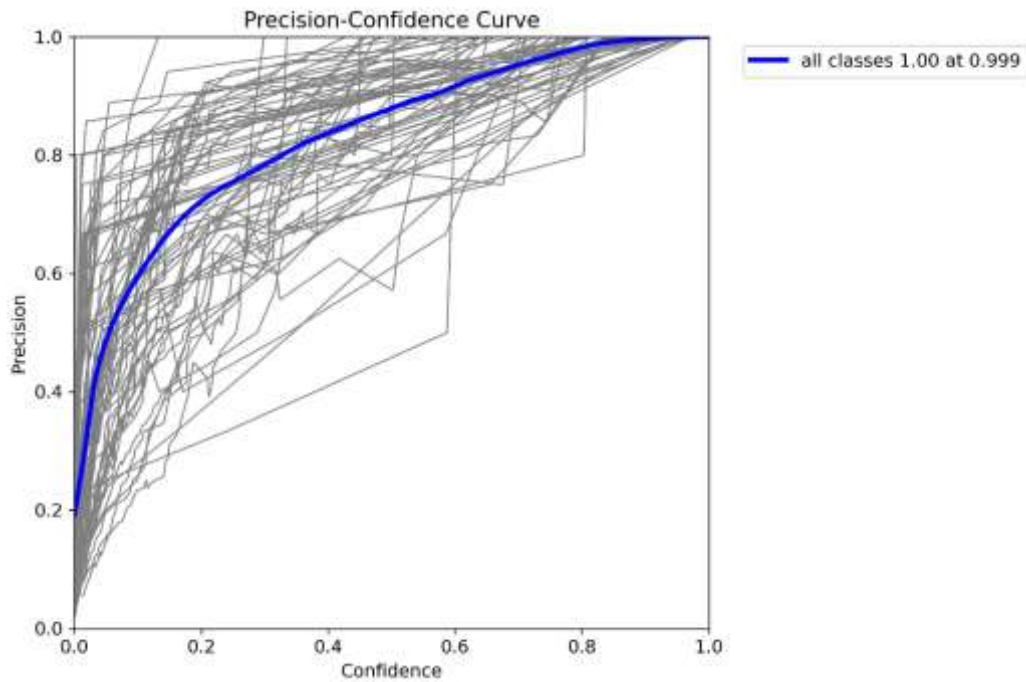
1. Download program dari link github berikut: <https://github.com/tedzies/yolov8-detection>
2. Ekstrak folder tersebut
3. Buka folder tersebut dalam VSCode atau menggunakan CLI (Pastikan direktori dalam CLI adalah direktori folder contoh: C:\Downloads\yolov8-detection)
4. Jika menggunakan VSCode klik run dipojok kanan atas, jika menggunakan CLI masukkan perintah berikut: `"c:\Downloads\yolov8-detection\object_detector.py"` (sesuaikan direktori)
5. Jika sudah, maka buka browser kemudian masukkan “localhost:8080”
6. Pencet choose file, kemudian masukkan foto yang akan diuji untuk *Object Detection*.

## Analisa

Untuk program ini, digunakan dataset Coco128. Dataset ini berisi objek-objek harian sebanyak 128 data. Data ini digunakan oleh karena jumlahnya yang sedikit untuk menghemat waktu pelatihan. Untuk pelatihan, karena menggunakan Google Colab, maka digunakan 100 epoch. Waktu pelatihan memakan waktu yang lama, sekitar 5 jam. Setelah melakukan pelatihan maka model terbaik disimpan dan dapat dilihat pada file “100best.pt” pada github.

Sebelumnya dilakukan pelatihan model menggunakan 5 epoch, yang memakan waktu sekitar 20 menit. Nilai loss yang didapatkan cukup tinggi yaitu 1,1. Dan saat model diuji menggunakan foto dengan banyak orang sebagai objek, objek yang di deteksi cukup sedikit. Kemudian dilakukan pelatihan kembali menggunakan 10 epoch dan nilai loss yang didapatkan masih di atas 1, dan perubahan tidak terlalu signifikan saat diuji. Namun sayangnya data tersebut tidak bisa ditampilkan oleh karena runtime google habis sehingga menghapus semua file.

Karena hasil yang kurang baik, maka dilakukan pelatihan menggunakan 100 epoch. Setelah melakukan pelatihan, loss untuk class detection dan box detection mencapai dibawah 1, sekitar 0,8. Dan saat pengujian menggunakan foto dengan banyak orang sebagai objek, model mampu mendeteksi lebih banyak objek yaitu 25 objek. Untuk metrik pengukuran yang digunakan adalah precision dimana dengan menggunakan 100 epoch, maka precision yang didapatkan mendekati 1. Berikut adalah grafik dari precision untuk model yolo yang dilatih menggunakan 100 epoch.



Oleh karena itu, model yang digunakan untuk program adalah model yang dilatih menggunakan 100 epoch.

## Hasil

Berikut adalah hasil pengujian program menggunakan foto-foto yang di luar dari dataset. Mengingat dataset yang digunakan untuk melatih relatif kecil (128 data) dan jumlah pelatihan epoch hanya 100, maka hasil mungkin tidak seakurat yang diharapkan apabila terdapat banyak objek atau kualitas foto kurang bagus.

1. Foto dengan banyak orang sebagai objek



Dapat dilihat bahwa semua objek yang dideteksi memiliki kotak hijau, dan apabila di zoom akan terlihat pada pojok kiri atas tiap kotak adalah nama dari objek yang digunakan. Namun dari banyaknya objek yang ada, model hanya mampu mendeteksi 25 orang dan salah mendeteksi beton tiang sebagai tenis raket.

```
0: 448x640 25 persons, 1 tennis racket, 332.6ms  
Speed: 20.0ms preprocess, 332.6ms inference, 18.5ms postprocess per image at shape (1, 3, 448, 640)
```

## 2. Foto Bus dengan beberapa orang sebagai objek



Pada foto berikut terdapat 4 objek yaitu 1 bus dan 3 orang. Karena terdapat lebih sedikit objek, maka model mampu mendeteksi semua objek yang ada.

```
0: 640x480 3 persons, 1 bus, 280.1ms  
Speed: 6.0ms preprocess, 280.1ms inference, 5.5ms postprocess per image at shape (1, 3, 640, 480)
```

## Kesimpulan

Yolo adalah neural network yang dikembangkan untuk mendeteksi object pada media, baik itu foto maupun video. Pada program yang dibuat, model Yolo yang digunakan adalah yang terbaru yaitu v8 dan model ini dilatih untuk mendeteksi objek pada media foto. Pelatihan dilakukan menggunakan dataset Coco128 yang berisi 128 foto objek hari-harian seperti orang, mobil, truk dan yang lainnya. Pelatihan dilakukan 3 kali dengan parameter tuning berbeda yaitu, 5 epoch, 10 epoch dan 100 epoch.

Dari semua pelatihan, model yang dilatih menggunakan 100 epoch memiliki kinerja yang lebih baik dengan nilai loss yang rendah dan precision yang lebih tinggi. Dan saat pengujian, terbukti bahwa model dengan 100 epoch lebih mampu dalam mendeteksi objek dengan baik. Untuk model yang lebih baik, disarankan untuk pelatihan menggunakan dataset yang lebih besar dan epoch yang lebih banyak, namun untuk pelatihan object detection ini memakan banyak komputasi dan disarankan menggunakan GPU dibanding CPU.