

Documentation/Notes on Object Detection and Tracking:

Author: *uditsangule* [uditsangule@gmail.com]

Note: Please read the README.md file before running the code for understanding the parameters and Output.

1. Getting Started:

The code and observations are made by running YOLOv8 segmentation model trained on COCO2017 Dataset with 80 classes.

I have used Lenovo G5080 Laptop with Corei5- Gen5 (2015) model , with Samsung SSD installed.Also I have Provided a CodeProfiler in the end of code termination which you can check when code ends.

2. Time sheet of Overall Work:

Date	TimeTaken (hrs)	Work	Status
23/01	2 - 3	Understanding Problem statement , Plannning and Strategizing Approach	Done
25/01	5 - 6	Coding Work, File Structuring and Implenting the version 1.0 (task 1 and 2)	Done
26/01	2 - 3	implementation Version 2.0 (Optional Tasks)	Done
27/01	3 - 4	Bug Fixes and Further Optimization of Codes	Done
28/01	1 - 2	Documentation work and Code Observations	Done
Overall	13 - 15		

3. Observations :

- YOLOv8 ObjectDetections works with more confidence if images around 1k or less is provided, but 5-10 % less if direct 2k frames are passed, downscaling of by factor of 4 is recommended.
- Inference Time is unaffected or (negligible difference) if we pass images greater than 640 x 480 size.
- Edges around Close viewed cell phone fails to detect some part of screen , if it contains some object that itself is a class of ObjectDetection Model.
- There is no way to track the liveliness of tracker in Yolo which is covered in Improvement part [Chp4] below.

4. Improvements in Object Detection and Tracking:

I am suggesting some improvement without retraining the models or any new specified Dataset models. Here are some post processing methodologies to get better object detections and tracking:

A. Improvement in Tracking by Adding Parent to child relationship on Classids to get less false detections with respect to concurrent frame.

Eg1 : cup , jug , mug can be given a new parentname such as 'cup'. So even if same cup is detected as cup in frame1 , mug in frame2, it will be recognized as cup as it shares same parent.

B. we can have better understanding of detected object by Adding image to depth neural network , which converts image to its respective 3D image.

C. improve the edges or segmentation of detected object by Adding voting algorithm to each pixel of segmented mask of two - three consecutive frames to get best pixel out of all 3 and then assigning the segmentation mask to object.

5. Future Version / Changes:

- GPU support can be provided in code , which can reduce inference time by 10x.
- Model selection automation code that detects the frame size and hardware capabilities to select model size of its own.

6. References:

- opencv documention - <https://docs.opencv.org/4.9.0/>
- ultralytics Yolo documentation-
<https://docs.ultralytics.com/models/yolov8/>