Assignment 5.1 Exercises

Video and Real-Time Analysis

Part 1- Video Read/Write

We will start with reading a video in Python, and apply some basic tools on it. The video chosen for this part is from *pixabay.com*, a resource of free video/image and sound clips.

a) Read the file "Bangkok.mp4" file using OpenCV. Find the relevant commands to find the video's metadata and print out the following information about the video: Frame Per second (fps), Total number of frames, Height, and Width of the video.

b) There are many ways to play this file inside Google Colab. If you just use OpenCV, it will be shown frame by frame. Another approach is to use an embedded version using HTML library. For the rest of the steps, it would be easier to use HTML version for playing the video.

c) Let's apply some filters on the video file. Use adaptive_theshold_mean and threshold_binary_inv, to create a sketch version of the video. You can use the Videowriter command to save the output it takes quite some time. Instead, only print 3 frames of the video.


Part 2- Pose Estimation in Video

Return to the functions that were provided in previous assignments to estimate the pose in images. This time, we want to experience it in a video file. For that, we will follow the approach in https://github.com/tensorflow/hub/tree/master/examples and estimate the pose of the girl in video. The format of the video this time is gif.

Part 3- Using Webcam Video

Now, we want to use a webcam and make object detection in real time. As you have seen in previous modules, the first step is to have a pre-trained model and use it for future classifications. My recommendation is to use the YOLO8 modell for this part (*yolov8n.pt*).

Another piece that we need is a script to connect our webcam to Python to make a live object detection application. We will be using the code snippet for Camera Capture which runs JavaScript code to utilize your computer's webcam. Please refer to the previous assignment to copy the code here.

Use a code to detect the camera and take a photo, and finally detect the objects in the photo.
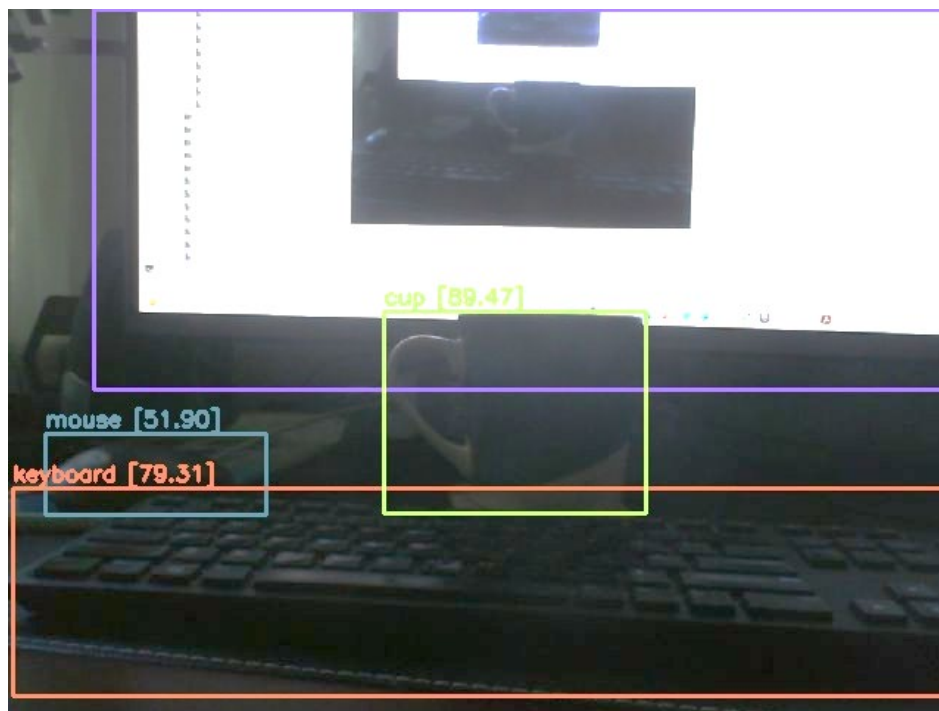
Extra credit: Instead of taking photos, design a code that can detect the objects in real-time while you are seeing the video on Notebook.