Zeshan Khalid

PGTE 5566 Creative Coding

SHIRLEY LEUNG

**Project Description**

This project is about setting up a smart camera system that can detect and highlight objects in real-time using a special recognition tool called YOLO.

**Code Explanation**

In my code, I have two functions: parse\_arguments and main. What parse\_arguments does is it receives input from my webcam to determine the resolution of the video. This way, I get the frame width and height, which I need for the rest of my code. In the main function, I call parse\_arguments to obtain the video resolution. I create boxes to track objects in a live video feed using this information. This live video is streamed using the OpenCV library, which in Python is cv2.

parse\_arguments specifically retrieves the webcam resolution from my system. Then, based on the dimensions of the incoming video—like video width, height, frame width, frame height—it creates a window to display the video captured by the webcam. This is all done using the cv2 library. Essentially, Python is interfacing with my webcam now, allowing me to get a live video input of my surroundings, which is then displayed in a window on my laptop.

Next, I'm initializing YOLO from the Ultralytics library in my code. I also have a line for box\_annotator, which draws boxes around all the objects in my video. The zone\_polygon function, I believe, scales this based on the size of my window.

When I get to the while loop, it’s continuously capturing the live stream of video. By setting it to while True, it captures video indefinitely, frame by frame, at a certain rate. The result variable feeds the incoming video into the model, allowing it to identify what's in each frame. The model then recognizes all objects in the video and draws boxes around them. For example, in my screenshots, you can see boxes around objects like a teddy bear or a TV monitor. The model also shows a confidence score for each identification. If the score is high, like 0.81 for the TV monitor, it means the model is 81% sure of its identification. Lower scores, like 0.31 or 0.56 for a pencil, indicate less certainty. That’s basically how my code is operating the webcam to capture and analyze live video.

**How ChatGPT explains my Code**

Setting Up a Camera on a Computer:

The code is like instructions for setting up a camera connected to a computer. It tells the computer how to use the camera and at what quality to capture images (like adjusting the camera to take clearer pictures).

Using a Special Camera Feature to Recognize Objects:

Imagine the camera can not only take pictures but also recognize different things in the picture, like a person, a car, or a dog. The code uses a special tool (called YOLO, short for "You Only Look Once") to give the camera this smart ability. This tool can look at the camera's images and point out, "Hey, there's a person here!" or "Look, a dog!"

Drawing Boxes Around Recognized Objects:

Once the camera, with the help of YOLO, recognizes something, the code then draws a box around it on the screen. So, if it sees a person, it draws a box around the person, making it easy for us to see what the camera has recognized.

Setting Up an Area of Interest:

The code also defines a specific area on the camera's view, like drawing an imaginary square on the floor. If any recognized object, like a person or a dog, moves into this square, the code takes special notice of that.

Displaying the Camera's View on the Screen:

All these images captured by the camera, along with the boxes around recognized objects and the special area, are shown live on the computer screen. It's like watching a video where certain things get highlighted automatically.

Stopping the Program:

Finally, if you want to stop this whole process, you just press a specific key on the keyboard (like the 'Esc' key), and the program stops running.

In summary, this code is like giving a computer the ability to understand and highlight what it sees through a camera, in real time. It's a bit like giving the computer a pair of smart glasses that can recognize and point out things around it.