**ADC Semi-autonomous AI Research**

### **Training YOLOv5 Using a Custom Dataset from Roboflow using Bash commands in VsCode**

#### **Step 1: Install Dependencies and Set Up YOLOv5**

1. Install **Python** (version 3.8 or later).
2. Install **Git** and clone the YOLOv5 repository:

Bash command



Bash command: git clone https://github.com/ultralytics/yolov5 cd yolov5

1. Create a virtual environment (optional but recommended):
2. Install the required dependencies(important):

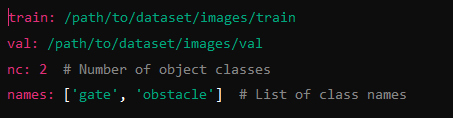


pip install -r requirements.txt

#### **Step 2: Download pre-annoated dataset from Roboflow**

YOLO format uses .txt files where each line represents an object’s class\_id, x\_center, y\_center, width, and height, normalized to the image size.

* Download the dataset in yolov5 format from here: <https://universe.roboflow.com/yolov8robomaster/aims-03p1o/browse?queryText=&pageSize=50&startingIndex=0&browseQuery=true>
* The downloaded dataset contains a data.yaml that looks like this:



* Unzip the dataset and move to ‘data’ folder within the yolov5 folder. Find where yolov5 is installed.

**Step 3: Train the YOLOv5 Model**

Run the following bash command in vscode to start training:



python train.py --img 640 --batch 16 --epochs 50 --data data.yaml --weights yolov5s.pt

* --img 640: Resizes images to 640x640 pixels.
* --batch 16: Number of images processed at once (depends on GPU memory).
* --epochs 50: Number of training iterations through the dataset.
* --data data.yaml: Path to the dataset configuration file.
* --weights yolov5s.pt: Start with pretrained YOLOv5 weights.

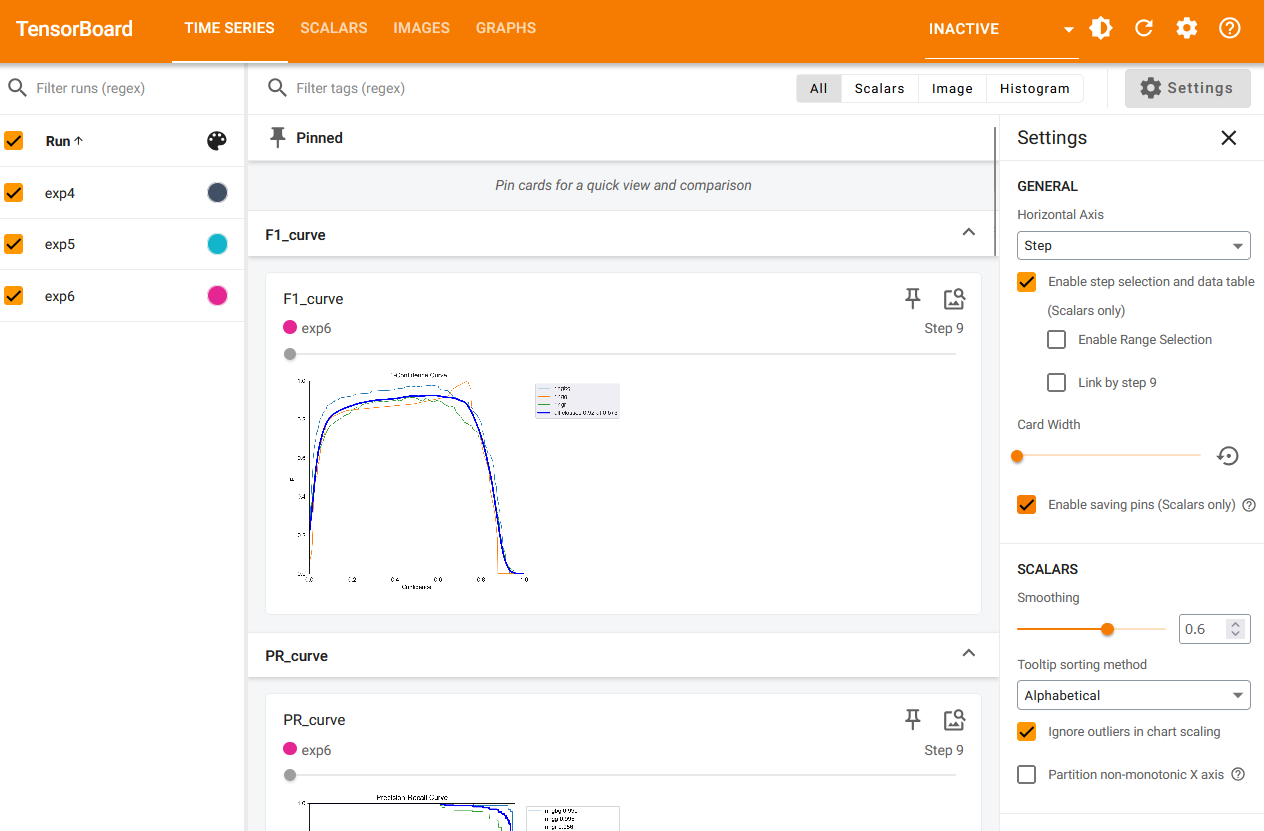
**Step 4: Monitor Training**

1. You can monitor training using **TensorBoard**:



tensorboard --logdir runs/train

1. Open the URL provided (http://localhost:6006) to view training metrics like loss, mAP, and more. It looks like this:



1. After training is completed, the best weights are saved in the runs/train/exp/weights/best.pt file.

**Testing the Trained Model on a Video File**

1. Locate the Best Weights, it is going to be here runs/train/exp/weights/best.pt file.
2. **Run Inference on a Video File:** Use the detect.py script to test the model on a video file:

python detect.py --weights runs/train/exp/weights/best.pt --img 640 --source /path/to/video.mp4

1. View Results: After running the command, the detection results will be saved in runs/detect/exp
2. Open the output video file and watch the magic.

**Reusing the Best Weights (best.pt) on Another Computer for Real-Time Detection with a Drone Camera**

1. **Install Python**, **Git**, and clone the YOLOv5 repository:

Bash command stated earlier: git clone https://github.com/ultralytics/yolov5 cd yolov5



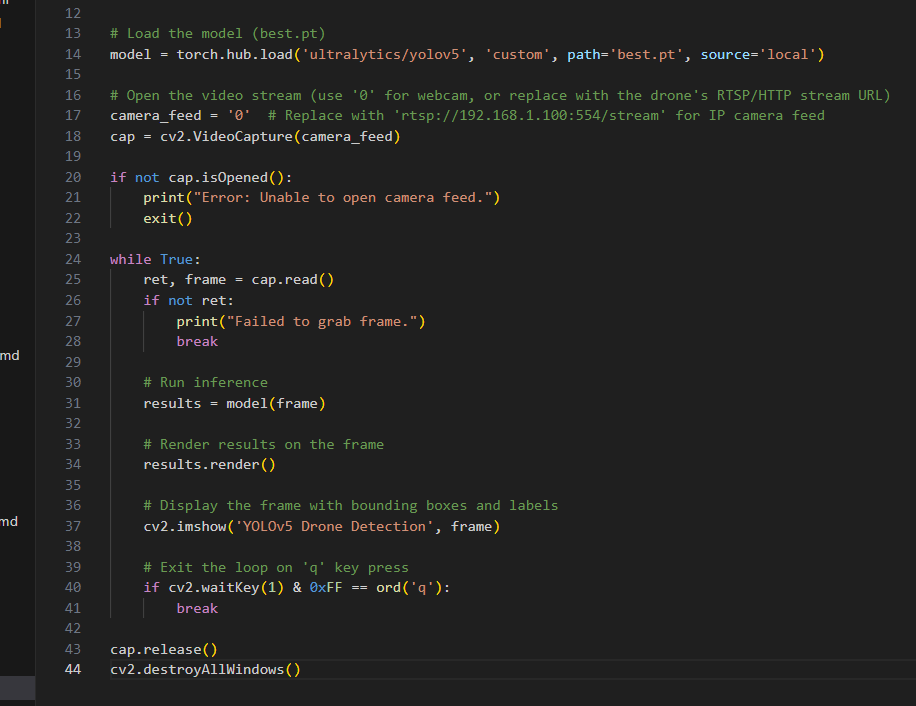
1. Install the required dependencies as stated above. Bash command: pip install -r requirements.txt



1. Copy the best.pt Weights to the New Computer
2. Place the best.pt file in the yolov5 directory within the runs>train>exp4>weights>
3. Set Up the Drone Camera Feed

**Real-Time Sample Detection Script**

1. Create a Python script (drone\_detect.py) to capture the live feed and run inference:



**Run Real-Time Detection**

1. Run the drone\_detect.py script to start live detection on the drone camera feed. Bash command:



python drone\_detect.py

1. The output will be displayed in a window with detected objects (bounding boxes and labels) in real-time.

