*School of Science, Computing and Engineering Technologies*

A black and red square with white text

Description automatically generated

**COS40007**

**Artificial Intelligence for Engineering**

**Portfolio Assessment-5: “Deep learning using YOLO v5'”**

Studio 6 – Portfolio 6

**Students’ Name:** Duc Thinh Pham - 104169675

**Lecturer:** Dr. Trung Luu

**Date:** October 20th, 2024

Contents

[Abstract 3](#_Toc180332461)

[Developing deep learning model using YOLO v5 and Pytorch 3](#_Toc180332462)

[Appendix 10](#_Toc180332463)

# Abstract

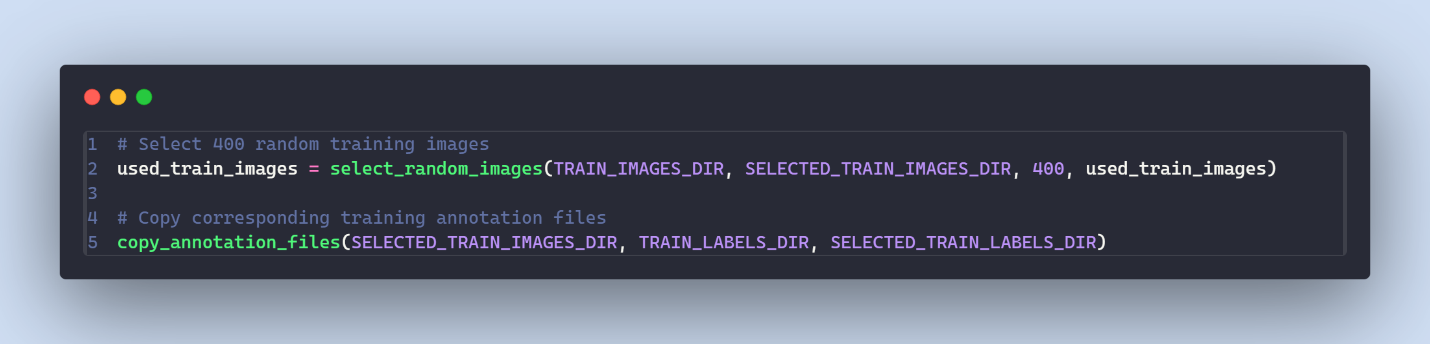
This portfolio demonstrates the development of a deep learning model using YOLOv5 and PyTorch for graffiti detection. The primary goal is to iteratively train a YOLO model using a subset of images to detect graffiti while optimizing performance through the Intersection over Union (IoU) metric. The model is trained using 400 random images from the training set in each iteration, and tested on 40 random images, with a CSV file generated to record image names, confidence values, and IoU values. The process is repeated until 80% of test images achieve an IoU value over 90%, or until the entire dataset is utilized. Furthermore, the final model is applied to real-time video data for graffiti detection. The portfolio includes code for data annotation conversion, iterative model training, and evaluation, along with detection results on video files.

# Developing deep learning model using YOLO v5 and Pytorch

1. **Write a function to convert given annotation format in training labels to YOLO annotation format.**



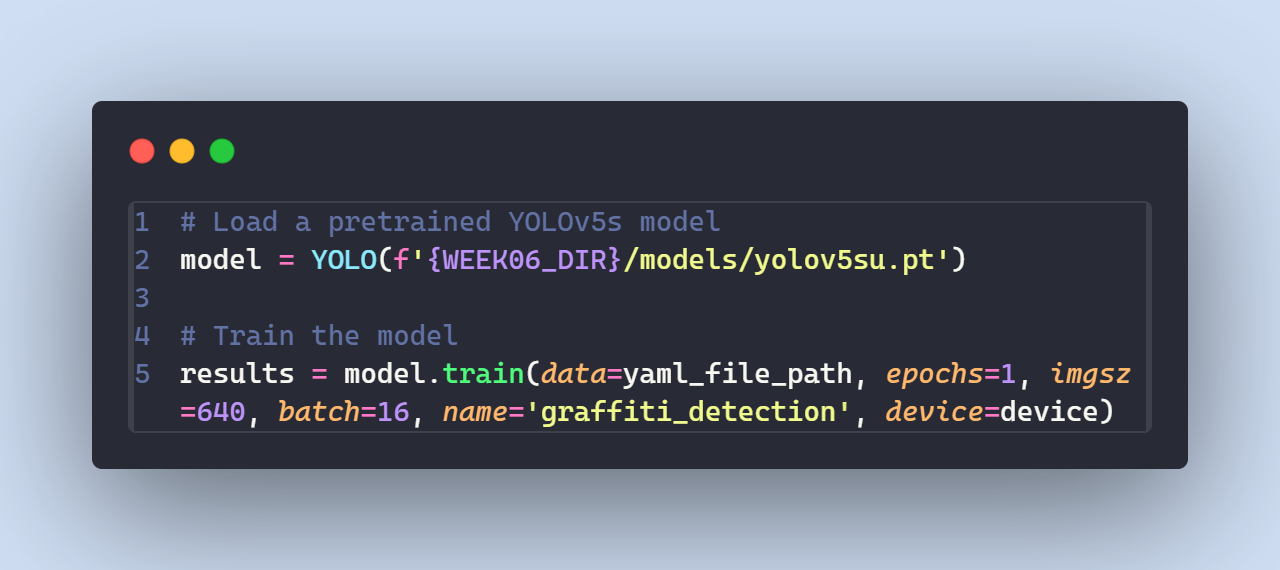
1. **Train and create a YOLO model by randomly taking 400 images from train data which can detect graffiti in the image**
   1. Select randomly 400 images from train data

****

* 1. Generate a YAML configuration file for a YOLO training process.

****

* 1. Train YOLO model

****

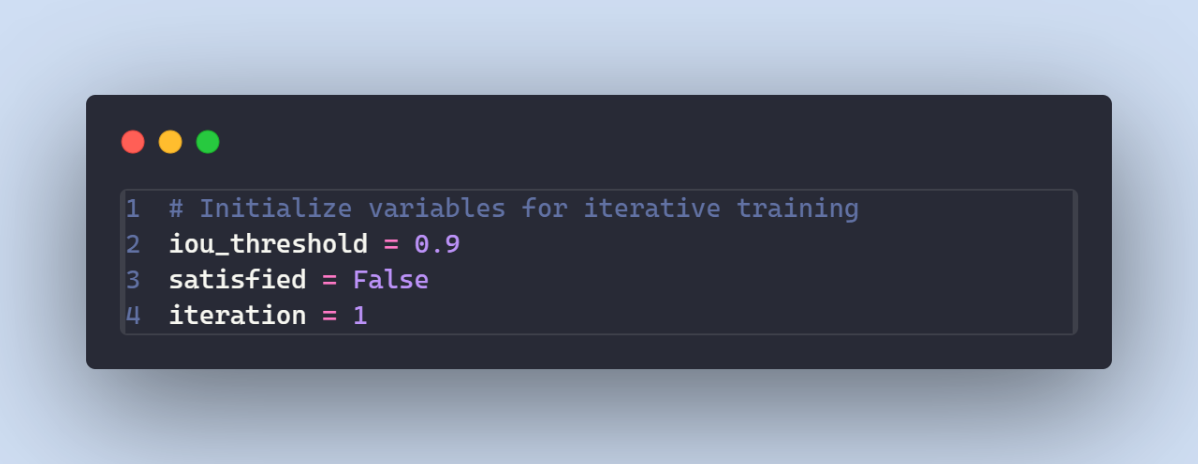
1. **Randomly take 40 images from test data and compute IoU for each and generate a CSV file containing 3 columns [image\_name, confidence value, IoU value]. If no graffiti is detected for an image, then its IoU will be 0.**
   1. Select randomly 40 images from test data

**A screenshot of a computer

Description automatically generated**

* 1. Evaluate YOLO model and compute IoU



1. **Until IoU value of 80% images in your test data is over 90% or all images are utilized for training and testing purpose, you need to iteratively train and test the model with a new set of 400 training and 40 test images. Make sure you use the model of previous iteration as the pre-trained model for new iteration.** 
   1. Initialize the threshold for iterative training****

4.2 Train and evaluate loop

****

1. **Use your final model to detect graffiti in real-time video data.**
   1. Load the best trained model

****

* 1. Fetch video from Pexels using its API

****

* 1. Fetch the video and apply the model for prediction

****

# Appendix

My GitHub repository: [thinhpham1807/COS40007---Artificial-Intelligence-for-Engineering (github.com)](https://github.com/thinhpham1807/COS40007---Artificial-Intelligence-for-Engineering)

Submission folder: <https://drive.google.com/drive/folders/1CVvGfAItkBKUBpUSdN3uE5NhuYuFgymw?usp=drive_link>

1. Code for step 1 - <https://drive.google.com/drive/folders/1lsUKSxrDJk85IDKP9ILPCwxdoXRDGqvs?usp=drive_link>

2. The best.pt model of each iteration - <https://drive.google.com/drive/folders/1cDfL7MiMDe92b0k0kBVe1OoxK9RdTUW7?usp=drive_link>

3. The CSV file of outcome for each iteration, and 2 good sample of detected images with

bounding box. Separate by folder for each iteration - <https://drive.google.com/drive/folders/1Ka5FvlJi1Dz5W8wJr11dPELNErkWVE2j?usp=drive_link>

4. Detection outcomes of 5 videos in (5) - <https://drive.google.com/drive/folders/1RrWrYXF_PsQN3-aWQePZceHPmlbGpC6r?usp=drive_link>