**Các bước thực hiện:**

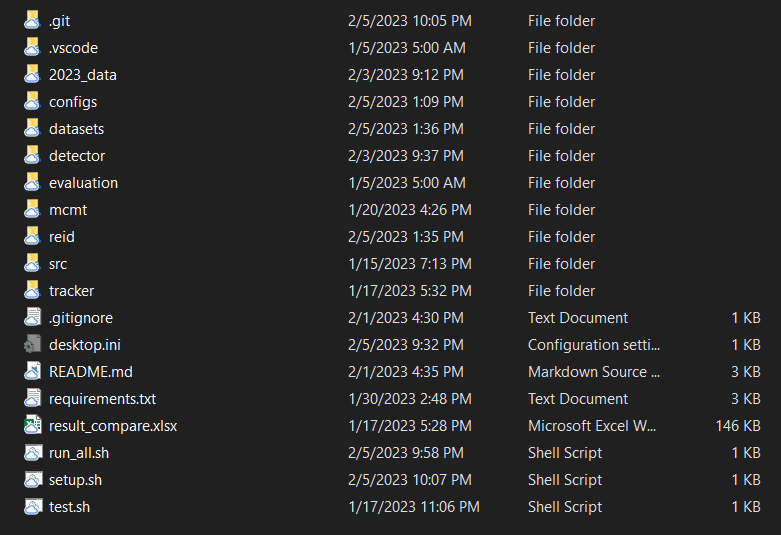
1. Tạo virtual enviroment với conda: (torch 1.13.1, cuda 11.7)

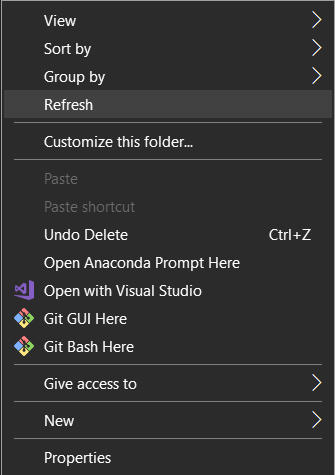
**conda create env -n mcmt python=3.9**

**conda activate mcmt**

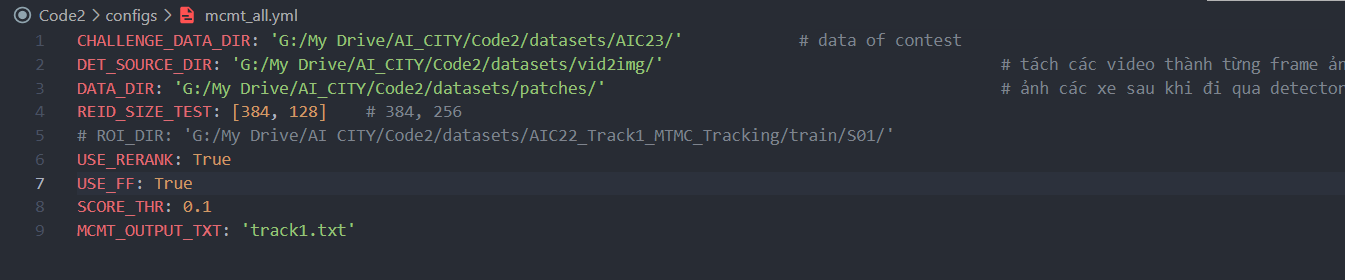
**pip install requirements.txt**

1. Chạy file “setup.sh”: Nếu trên window có thể dùng **git bash** để chạy

****

****

1. Thay đổi các đường dẫn có trong folder **“config”:**

****

File **mcmt\_all.yml**

**CHALLENGE\_DATA\_DIR**: đường dẫn đến dataset của BTC trong máy của mình (đường dẫn tuyệt đối)

**DET\_SOURCE\_DIR**: đường dẫn đến folder để lưu các frame được tách ra từ các video cuộc thi. Không cần tạo trước folder đó, chỉ cần để đúng đường dẫn tuyệt đối như của **CHALLENGE\_DATA\_DIR** và thay thành **vid2img**

**DATA\_DIR:** đường dẫn đến folder lưu các patches. Patches là các đối tượng được cắt ra khỏi frame dựa vào bounding box từ detector (sau khi đi qua yolov7)

Các tham số còn lại có thể để nguyên



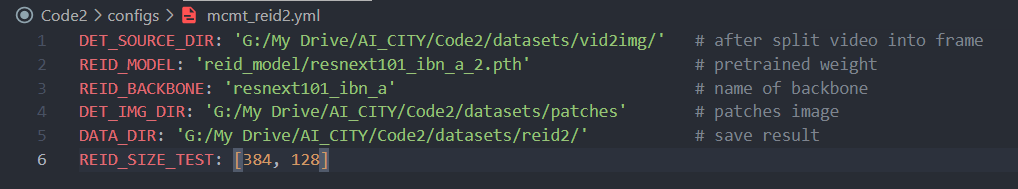
File **mcmt\_reid1.yml**

**DET\_SOURCE\_DIR:** giống của **mcmt\_all.yml**

**REID\_MODEL, REID\_BACKBONE:** đường dẫn đến pretrained weight của model và tên của model. Không cần phải thay đổi

**DET\_IMG\_DIR:** giống **DATA\_DIR** của **mcmt\_all.yml**

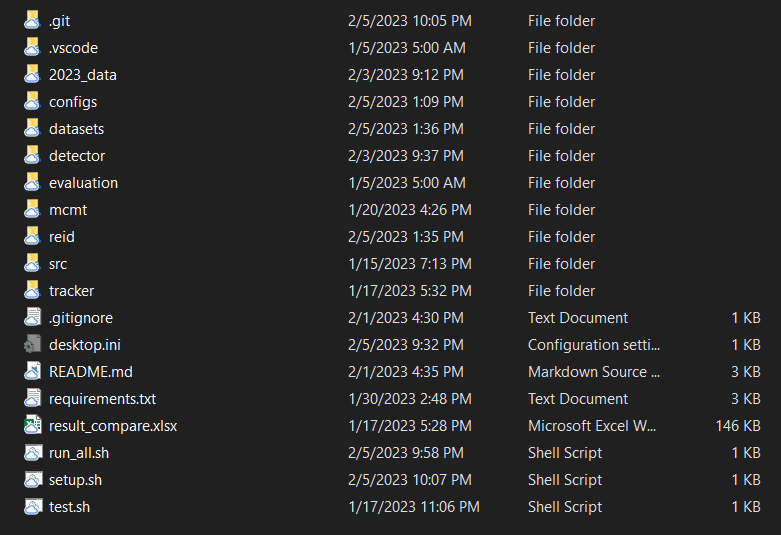
**DATA\_DIR:** đường dẫn lưu kết quả, chỉ cần thay đường dẫn trước /dataset/reid1. Không cần tạo trước, code tự tạo folder này.



File **mcmt\_reid2.yml**

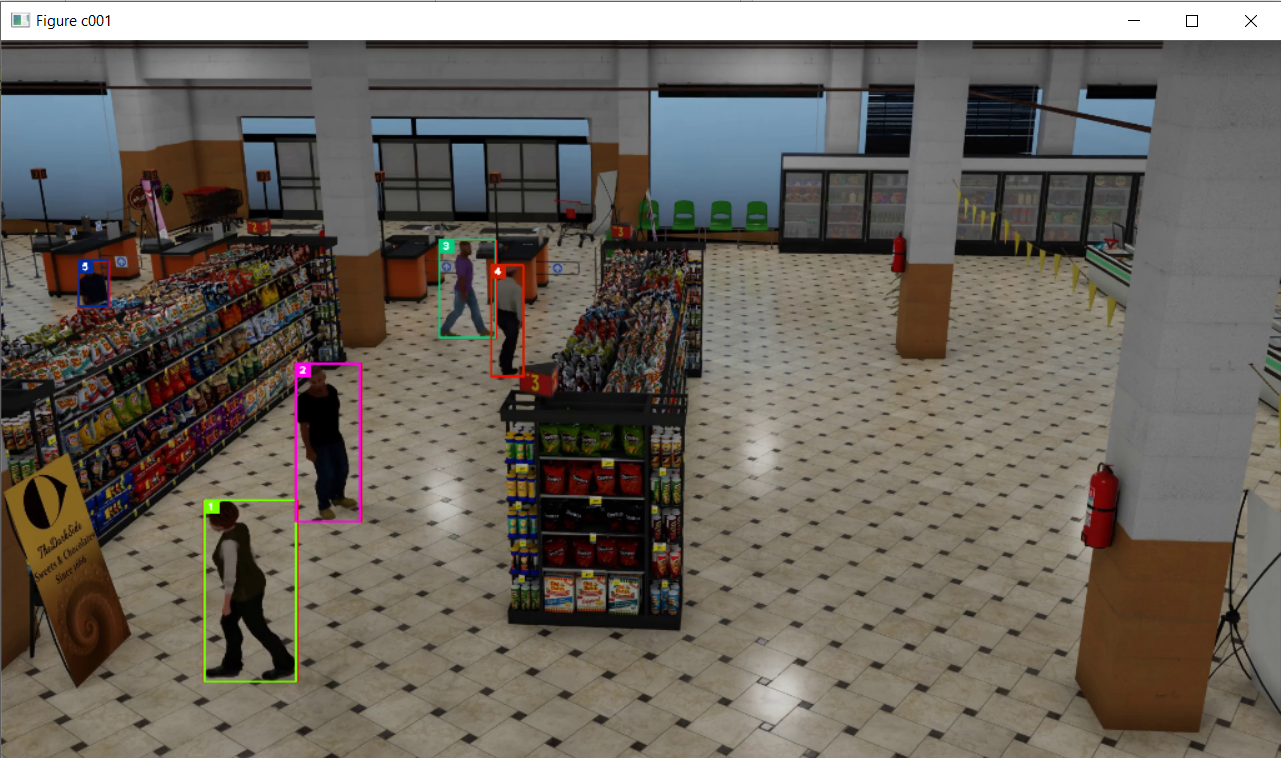
Tất cả giớng với **mcmt\_reid1.yml,** chỉ thay lại tên của model và đường dẫn đến pretrained model đó. Các đường dẫn còn lại có thể cop từ **mcmt\_reid1.yml**

1. Sau đó ta sẽ chạy file **run\_all.sh**:

****

1. Ví dụ output:

* ‘space: để pause video
* ‘s’: để chạy và dừng lại ở từng frame (step)
* ‘esc’: thoát.

****