```
    # Exit if q is pressed.
    if cv2.waitKey(1) == ord('q'):
        break

# Release Capture and destroy windows
cap.release()
cv2.destroyAllWindows()
```

DeepSort Tracking:

```
from deep_sort import generate_detections as gdet
from deep_sort.tracker import Tracker
from deep_sort.detection import Detection
from deep_sort import nn_matching
import time
from yolov3.configs import *
from yolov3.utils import Load_Yolo_model, image_preprocess,
postprocess_boxes, nms, draw_bbox, read_class_names
import tensorflow as tf
import numpy as np
import cv2
import os
os.environ['CUDA_VISIBLE_DEVICES'] = '0'



def Object_tracking(Yolo, input_size=416, show=False,
CLASSES=YOLO_COCO_CLASSES, score_threshold=0.3, iou_threshold=0.45,
rectangle_colors='', Track_only=[]):
    # Definition of the parameters
    max_cosine_distance = 0.7
    nn_budget = None

    #initialize deep sort object
    model_filename = 'model_data/mars-small128.pb'
    encoder = gdet.create_box_encoder(model_filename, batch_size=1)
    metric = nn_matching.NearestNeighborDistanceMetric(
        "cosine", max_cosine_distance, nn_budget)
    tracker = Tracker(metric)

    times, times_2 = [], []
```