**TRAFFIC MANAGEMENT**

import cv2

cap = cv2.VideoCapture(0) # 0 for default camera

while True:

ret, frame = cap.read()

# Perform traffic analysis on 'frame' here

if not ret:

break

cv2.imshow('Traffic Camera', frame)

if cv2.waitKey(1) & 0xFF == 27: # Press 'Esc' to exit

break

cap.release()

cv2.destroyAllWindows()

**VECHICLE DETECTION COUNTING**

import cv2

import numpy as np

# Load YOLO model

net = cv2.dnn.readNet("yolov3.weights", "yolov3.cfg") # Update with your YOLO model files

# Load COCO class labels

classes = []

with open("coco.names", "r") as f: # Update with your class names file

classes = f.read().strip().split("\n")

# Function to detect vehicles and count them

def detect\_and\_count\_vehicles(frame):

height, width, \_ = frame.shape

# Convert frame to a blob

blob = cv2.dnn.blobFromImage(frame, 0.00392, (416, 416), (0, 0, 0), True, crop=False)

net.setInput(blob)

outs = net.forward(net.getUnconnectedOutLayersNames())

class\_ids = []

confidences = []

boxes = []

for out in outs:

for detection in out:

scores = detection[5:]

class\_id = np.argmax(scores)

confidence = scores[class\_id]

if confidence > 0.5 and class\_id == 2: # Class 2 corresponds to vehicles in COCO dataset

center\_x = int(detection[0] \* width)

center\_y = int(detection[1] \* height)

w = int(detection[2] \* width)

h = int(detection[3] \* height)

x = int(center\_x - w / 2)

y = int(center\_y - h / 2)

class\_ids.append(class\_id)

confidences.append(float(confidence))

boxes.append([x, y, w, h])

indexes = cv2.dnn.NMSBoxes(boxes, confidences, 0.5, 0.4)

vehicle\_count = len(indexes)

for i in range(vehicle\_count):

x, y, w, h = boxes[i]

label = str(classes[class\_ids[i]])

confidence = confidences[i]

color = (0, 255, 0) # Green

cv2.rectangle(frame, (x, y), (x + w, y + h), color, 2)

cv2.putText(frame, f"{label} {confidence:.2f}", (x, y - 10), cv2.FONT\_HERSHEY\_SIMPLEX, 0.5, color, 2)

return frame, vehicle\_count

# Capture video from a file or camera

cap = cv2.VideoCapture("video.mp4") # Update with your video source

while cap.isOpened():

ret, frame = cap.read()

if not ret:

break

processed\_frame, vehicle\_count = detect\_and\_count\_vehicles(frame)

cv2.imshow("Vehicle Detection", processed\_frame)

if cv2.waitKey(1) & 0xFF == ord("q"):

break

cap.release()

cv2.destroyAllWindows()