Object Detection Documentation

**Name:** Tushar Chauhan

**Date:** 06 April 2024

**Introduction**

Object detection specially on the streets is really important for making smart driving systems and react to the surroundings, like noticing a pedestrian, a cycle, or a bike. The technology is helpful not just in enhancing driving system but also in public and workplace safety.

This document explores the application of the object detection models for real-time identification of potholes, traffic lights, signboards, vehicles, pedestrians, and more. The approaches include use of both pretrained models, as well as a model I trained myself using a custom dataset from the Roboflow database.

**Literature Review**

There are numerous studies have been done for object detection. Different methods for identifying objects have been explored, including pre-trained models like YOLO and MobileNet SSD, R-CNN, designing custom models, and vision transformers. Research papers on identifying specific items like potholes and signboards have also been studied. Researched deep learning algorithms that can learn from video, allowing them to understand and predict movements and changes in scenes over time.

**Methodologies**

The approaches used are listed below. Each part talks about a different strategy or model used in the experiments to detect objects.

1. **MobileNetSSD pretrained model**

MobileNet SSD is a fast and efficient pre-trained model designed for object detection on mobile devices. It combines the MobileNet architecture, which is lightweight yet powerful, with the Single Shot Detector (SSD) framework to quickly identify objects in images.

*Location:* Repository - Method 1

*Contents:* This folder contains all input videos along with the output files, presenting the predicted object frames.

*Outcome:* The results yielded by this method were less than satisfactory.

**Pros:**

MobileNetSSD is known for being lightweight and fast, making it suitable for real-time detection and applications with limited computational resources.

**Cons:**

The accuracy was found to be lacking, potentially due to the less complex architecture not being robust enough for the diversity of the dataset.

1. **Ultralytics YOLOv5 pretrained Implementation**

Ultralytics YOLOv5 is a fast and accurate model for detecting objects in images or videos. It's part of the YOLO (You Only Look Once) family, known for looking at an image once to find different objects. YOLOv5 is easy to use and can be customized for various tasks. It's great for projects needing quick and reliable object detection, like spotting objects in drone footage or monitoring areas for specific activities.

*Location:* Repository - Method 2

*Contents:* This includes scripts for the detection process, as well as the input and output files.

*Performance:* Exhibited promising results in identifying objects like vehicles, person, sign boards within video streams. There were, however, instances of misidentification.

**Pros:**

YOLOv5 has demonstrated high precision and recall rates, making it a reliable model for accurate detection in various conditions. The speed of detection is quite efficient, which is beneficial for real-time applications.

**Cons:**

Misclassifications occurred, such as confusing stop and speed signs with unrelated objects, indicating a possible need for more training data or hyperparameter tuning. Might require substantial computational resources for training and may not be as efficient as other models when deployed on lower-end hardware.

1. **Ultralytics YOLOv8n Custom Training**

*Location:* Repository - Method 3

Structure:

* Downloaded pothole and traffic signs dataset from Roboflow.
* Obtained data for an additional 12 classes from Roboflow, encompassing traffic-related imagery.
* Developed scripts for data loading and transformations in Python.
* Validated the compatibility of the dataset with the YOLOv8n model.
* Trained the YOLOv8n model on a T4 Tesla GPU.
* Stored the trained model in TorchScript format.
* Performed object prediction in images using the trained model.
* Used the output in conjunction with a pothole detection script since the fine-tuned model had not fully learned to detect potholes due to insufficient training epochs.

**Pros:**

YOLOv8n is designed to push the boundaries of accuracy and speed, potentially surpassing previous versions.

The use of a Tesla T4 GPU for training implies the ability to handle complex models and large datasets efficiently.

**Cons:**

The model's training was not fully completed, which resulted in inadequate detection of potholes. Additional training epochs may be necessary for improved performance.

Since it is a more recent and possibly complex model, it may require more data and a better training strategy to reach its full potential.

**Suggestion to improve the performance**

**Two-layer model (Pretrained YOLOv5 + pothole detection model)**

The two-layer model is crafted to precisely identify traffic signs, vehicles, and previously missed or inaccurately detected potholes. Initially, videos or images are processed through a pre-trained YOLOv5 model to spot vehicles and traffic signs. Subsequently, they are routed through a specialized pothole detection model, ensuring that potholes are also accurately identified.

**Pros:**

The model excels at identifying potholes that were previously undetected, significantly improving the overall accuracy of the algorithm.

**Cons:**

The model excels at identifying potholes that were previously undetected, significantly improving the overall accuracy of the algorithm.

**Conclusions and Future work**

The documented methods constitute an extensive exploration into enhancing the performance of object detection systems. The advantages and limitations of each method have been scrutinized, with each possessing unique strengths and areas for improvement. Continuous refinement and adaptation of these methods are imperative for the advancement of object detection accuracy and for their application to specific recognition tasks.