Machine Learning Engineer Nanodegree

Capstone Proposal

Luke(Xu) Lin
Jan 1st, 2018

Proposal

Domain Background

Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Well-researched domains of object detection include face detection and pedestrian detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance.
[https://en.wikipedia.org/wiki/Object_detection]

Problem Statement

I participated in a system which help user to organize their receipts. So within this system, there are tons of receipts image need to be cropped and do OCR. I would like to handle cropping part and let Google do the OCR part. I would like to use machine learning to automatically crop receipt out for me. The ideal successful rate should be more than 90%.

Datasets and Inputs

The dataset is coming from this receipt management system. All the receipt images are uploaded from users. But we also need manually input the expect corners of the receipts.

Solution Statement

So basically, it is an object detection issue.

- 1 rezise all the receipts in to same size.
- 2 provide expected corners location
- 3 feed training set into training algrithm
- 4 test outcome algrithm with test set, get successful rate.

Benchmark Model

The TensorFlow Models GitHub repository has a large variety of pre-trained models for various machine learning tasks, and one excellent resource is their object detection API. The object detection API makes it extremely easy to train your own object detection model for a large variety of different applications. Whether you need a high-speed model to work on live stream high-frames-per-second (fps) applications or high-accuracy desktop models, the API makes it easy

to train and export a model.

Evaluation Metrics

Accept rate should be more than 90%.

Project Design

From the desceition and the problem statement it can be inferred that TenseFlow Object Detection can be one solution for it.

We need to pre-process all the images first.

- 1 rezise all the receipts in to same size.
- 2 provide expected corners location
- 3 feed training set into training algrithm
- 4 test outcome algrithm with test set, get successful rate.
