Technology Review and Implementation Plan

Team (Group 32) name: Teaching AutoPilot to Dodge

Author: Xilun Guo

Team members: Basil Al Zamil and Tanner Fry

CS 461: Senior Capstone Fall 2016

Oregon State University

Abstract

The aim of my tech review research is to figure out the ways to test the image recognition algorithms within several different conditions. There are environment conditions(sunny, rainy, and snowy) testing, camera blind spot(too close to see stationary object) conditions testing, objects testing(avoid moving object), and the ways to quantify and verify the tests for our project. The three main technologies are described in each condition; beside, there are two additional technologies for data images organization as well as setup testing scripts.

CONTENTS

1 Environment Conditions Testing

1.1 Introduction

We gather on road data set in a different area with various weather conditions, and filter useful categories based on camera setup and professorial quality video outcome. In the environment conditions testing, we will divide into three main severe weather categories, which could break the algorithm with higher possibility then normal weather. There are Rainy, Snowy, and Sunny.

1.2 Rainy Condition

Similar to human eyes, video can not clearly catch the situations while heavy raining condition, but camera is even worse than human eyes. Considering ultrasonic and camera working together in the self-driving car, but the video is more precise to predict the next movement, we need to test if the video can at least record basic information to avoid crashing or any serious action by applying the algorithms. Considering the worst case, which is the continuing heavy raining condition, a camera might be limited by the discontinuously recording on the road. We are going to test if the algorithm can handle if the camera cannot see very 0.5 second in 1 second by taking out half image data continuously. Basically, if there is any object in the image can't be recognized due to rainy condition, we will get the specific image out and put it into fail result file.

1.3 Snowy Condition

Snowy condition is similar to rainy condition, but it makes camera harder to see. We will delete the image record by 0.5+ second in every 1 second, to see how often does the algorithm rely on the picture, and how poor condition it could still make the safe outcome. In the case of some subjects can't be seen even in the actual image due to windshield is completely blocked out by snow, we won't put the completely black color result images into the fail result file. The situation should be handled by ultrasonic in the real world.

1.4 Sunny Condition

Lighting may affect the camera can not catch some partial objects in the images. For example, some part of a car can't be seen clearly. we need to see if the algorithm could store the image data before camera cannot see and make some safe reaction. In this case, we are going to take out long consistent period, which is around 5+ seconds, to see if the algorithm can still make the reaction while no an extended period of video record. On the other hand, we also want to see if the algorithm can recognize a half part of a car as a car. Since we have approximately 35 colors to determine different objects, all we need to do is analyze if the half part of cars are shown as the color, as which a car should be recognized.

2 Camera Blind Spot Conditions Testing

2.1 Introduction

It is hard to see the objects outside of the main sight line; however, if the car is not getting too close, the camera can catch the block before approaching it. Therefore, the main goal for these conditions is to test if the algorithm can predict if the vehicle can pass or not before too close to see the block. By passing this situation, we need to see if some specific objects can be recognized in the result images before it is out of images. There are only two situations: too wide and too high to record it while approaching.

2.2 Narrow Street With Guardrail On Two Sides Passing(Limited Wide)

We are going to set a chart and mark down the statistics of how wide base on the speed the camera can not record in how many meters the vehicle approach to it. Using the algorithm, we want to test if it can predict the vehicle can or can not pass before approaching by recognizing the wide between two object. To pass this situation, we have to compare the actual images and result images and analyze if the whole part of objects can be recognized so that there are much less error while calculating the length of the wide.

2.3 Low Bridge Oassing(Limited High)

It is the same situation as the above; however, it is recording the statistics of the height base on the speed the camera can not record. Using the similar technology in order to see if the algorithm pass the case.

3 Unusual Objects Testing

3.1 Introduction

Comparing to unmoving objects, moving objects require more reaction from the algorithm. There are many unpredictable conditions can lead accidents from moving objects. We are going to focus on testing a couple situations: little kids suddenly cross the street and the vehicle with a strip of a bar in front of our car.

3.2 Little Kids

While driving down the street, suddenly insert a little object, representing kids, in x meters front of the car. Base on the speed of reaction from the algorithm, testing what is the range of distance that the car can stop and avoid hit the object. In order to pass this conditions, we need to test if the result images are able to recognize the objects we set in no time.

3.3 Small Objects

Small objects could be animals or some unusual small things that moving in the street, and we mainly test how quick the algorithm can make the reaction to avoid those objects that cause accidents. By testing if the algorithm can recognize some unusual object on the car with possibility leading accident. We need to setup a color for those objects different from cars and see if there are showing in the result images.

4 Additional Technology

4.1 Technology For Organizing Data Images

Applying database technologies, we can insert all image data we collect into the database and select each different conditions we aim to test without overlap. We are able to use this technology in order to point the actual images to result images, and it will make analysis process much easier. Moreover, it allows us to modify any data such as deleting, changing scale, and editing ratio in a convenient way.

4.2 Technology For Setting Up Testing Scripts

As such a huge amount of dataset, it would take a long period of extra time if we run them manually. Using unit testing technology and setting up some scripts in order to let algorithms automatically grape the actual images and put the result images into specific files after running through. More importantly, due to algorithms are smart enough to improve ability by running more and more data, we need to apply some default module from Cityscape to make the algorithms run much enough data before we test our own image data. Therefore, it is significant technology for our project to get the expected result and saving our time for more testing.

5 Tests Prov Plan With Analysis

It is critical to have the technology to verify each section of tests. Setting a plan for all ideal and achievable results that are proving if the test pass, then the algorithm reacts safely with the conditions we tested. By doing that, we will create a matrix of potential conditions as the X-axis and unusual objects as the y-axis. It allows us to analyze the result much more clearer, and we can find the specific objects, which can’t be recognized by the algorithm with specific conditions. On the other hand, if the current test fail to pass and out of setting safe range, then the conditions could possibility break the algorithm and we will plan further to fix it up or upload as a file for the designers to improve the algorithms. Moreover, we might need addition testing to confirm that the fixed algorithm can help hundred percent handle the conditions. Not just recognizing all objects as specific color, but do much enough calculation and analysis to make it sure.

6 Change Log

Overall: Revised the whole document because I thought the algorithms can test videos, but we have to extract from videos to images, which the algorithms can run with.

Redesign the technologies we would use for each condition testing.

Add more details on how we analyze expected result images by comparing to the actual images.

Improve some formating stuffs and grammar issues.

More clearly describe how we organize data, process testing, and apply technologies to analyze.

References

[1] https://github.com/mcordts/cityscapesScripts. *Part of the Cityscapes team* Marius Cordts, Mohamed Omran.

[2] https://www.cityscapes-dataset.com/ *Cityscapes subset of Daimler* 2016 Cityscapes Dataset by Marius Cordts

[3] http://pedrokroger.net/choosing-best-python-ide/ *Choosing the Best Python IDE* Dr. Pedro Kroger