Unusual Objects on Road

Team (Group 32) name: Teaching AutoPilot to Dodge

Basil Al Zamil, Xilun Guo, and Tanner Fry

CS 461: Senior Capstone Fall 2016

Oregon State University

Abstract

The primary goal of the project is to collect a dataset of real-world driving scenarios (images, video, and sensor data) where potentially problematic objects are in a frame along with bad weather and lighting scenarios. For example wildlife such as deer, cows, and birds need to be correctly identified, along with landmarks of odd shapes such as sculptures in all sorts of lighting and weather conditions. Then we will test whether current self-driving car software can detect, recognize, and accurately maneuver around unexpected and or extraordinary objects. Specifically, we look at autonomous driving algorithms capability to comprehend objects given particular world conditions (such as lighting, rain, water, snow, etc).

CONTENTS

1 Introduction

1.1 Main requirements

Our main requirements are limited by the bugs we find. The hard goal we have is to test the algorithms used by CityScape in autonomous vehicles. Assuming there are issues with the image recognition algorithms under certain conditions, we will collect our own data, and we must then collect the errors we find into a single dataset that the system fails to recognize certain objects in the images, which extracted from videos recorded on the road. This can be set as a baseline of failure and be forwarded to the CityScape team. This will also open us up to the possibility of modifying the algorithms used ourselves to try and fix the errors that we found.

We are required to use some strategies to collect as much data as possible in the amount of time for proving if there is a situation which could break the algorithms. For example, trying to find some volunteers from different areas that allow us to place cameras onto their vehicles, thus we can collect the video feed from their driving experience. As a result, we will apply the algorithms from CityScape to analyze the data we collected. Specifically, the data we are looking for is wide angle front image of a vehicle while driving within a bad digital visual field. In addition, some objects are placed on too high or too low, which the camera might view but because of the angle or camera field of view the algorithms do not analyze the image correctly.

They also could not respond quickly enough to react to driving conditions, which may lead to a car crash or to a dangerous situations. This however is not in the scope of our project due to us not having availability of the hardware that the self driving system do. So while speed of the algorithm is something to consider with any code changes we make we do not a measurable metric to be limited by. Also consider that the speed the algorithm would need to function at to make the right decision is based on the speed at which the vehicle is moving.

2 Nature of the Project Requirements

2.1 Environment and Characteristics of Requirements

The project is research based and is focused on testing more so than code development. To that end we are required to have a thorough process for collecting data, processing data sets, and testing the software.

2.2 Functionality of Testing

Our end objective is to methodically test the Cityscape image recognition algorithms to either confirm areas of fault or determine that the algorithm is flawed. To do this we will create a matrix of potential lighting and weather situations as the X-axis and unusual objects as the y-axis. This will allow us to create a matrix with a varying number of rows depending on the object we find that are unrecognizable by the algorithm.

2.3 External Interfaces and Test Constraints

We have no concern for application of hardware or specific interfaces as we are just testing the software algorithm. That being said we need to verify quality of any cameras used, along with any online data sets we use. The only limitations of testing are those that go out of the bounds of a normal driving scenario. Such as driving off road or areas where the sensor systems will not function.

3 Requirement Specifications

3.1 Methodology

Creating video, and extracting to image datasets and apply them to the CityScape algorithms for testing. Setting up algorithms running environment, and having them to learn by applying some file from CityScape before running them. Then verify that all the datasets that we applied to the algorithm either succeed of fail. Thus we can safely assume the algorithm is relatively problem free or has flaws. Comparing the algorithm results and actual results helps to find any unusual objects can't be recognized. Using the matrix system to record and gather data we can then verify if the object is the issue or the lighting/weather is to blame (or both). We can do this by running datasets with both cases individually and then in parallel to compare behavior and response of the algorithm. Thus have better feedback and understanding of the algorithm faults. In order to analyze the results we got, we are required to do every single steps carefully so that we will be able to narrow down the possibilities and figure out what makes the unexpected results. Along with that it will give us a better idea of what to change should we choose to edit the algorithm ourselves.
As an example of some of the targets we want to hit:

- Direct light on the camera
- Direct light reflections off wet road
- Heavy rain (over .5 inches an hour)
- Snow on road and environment

3.2 Stretch Requirements

Improve and correct the image recognition algorithm so it does not fail to recognize unusual object(s) or lighting errors we find. The corrected algorithms should recognize the unusual object under the certain conditions that it was failing to recognize in the previous algorithm version. For example, if we found that the algorithm fails to recognize the side of a white semi-truck when the side of the semi-truck is reflecting sunlight, the corrected algorithm would recognize the side of the semi-truck under the same lightning condition.
Next, we would run the new algorithms under the same tests that was ran on the previous algorithm, to insure that the improved algorithm did not create new bugs. For example, the improved algorithms may fail to recognize stop signs, where the previous algorithm recognizes stop signs.

4 Overall Description

4.1 Planned Testing Methods

Since different weather conditions are a prime part of our testing objectives, a dataset from different weather conditions must be collected. Now we also need to consider different environments around the world, not just limited to our climate here in Oregon. We will do this by obtaining datasets from different environments, either from notable sources on the internet, or through contacting other individuals around the world. To control as many of the variables as possible, we are required to clearly understand which kinds of weather could deviate the normal algorithm function. Heavy rain, snow, sunny day, darkness, and direct sunlight and vehicle headlights into the camera all must be tested. Also we will focus primarily on country or side roads rather than inside city limits as most datasets have been primarily tested in busier settings. We will attach a decent quality camera to the inside of the car dashboard to collect video while driving. If possible we would also collect GPS data.

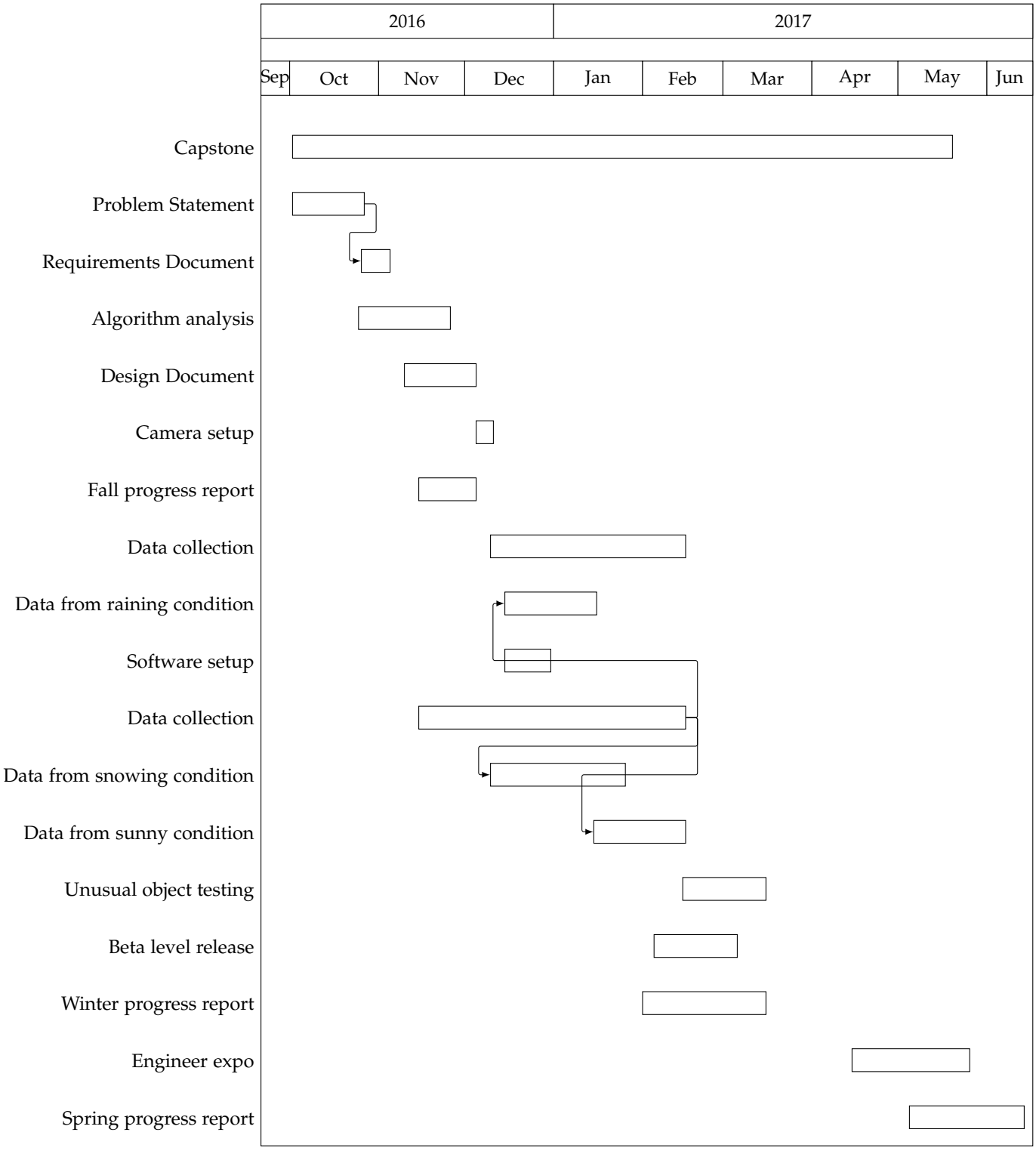

Fig. 1: Gantt Chart: Timeline of Project Tasks

5 CHANGE LOG

Updated the requirements of collecting our own data, and what we need to teat specifically in section 1.

Minor grammar issues fixed in section 1 and 2.

Updated the specific steps we are also required to do but we didn't realized before in section 3.

Minor changes on the timeline based on what we have done till now

Added change log section.