Final Project Proposal
COMP4102A

Automotive Safety Suite

Authors:

CONNER BRADLEY – 101073585
CHRISTIAN BELAIR – 101078744
ADAM PAYZANT – 101082175

Carleton University

February 5, 2020

1 Summary

The goal of this project is to create an automotive safety suite using computer vision. Our goals are to create a suite of services to track pedestrians, road signs and stop lights, as well as tracking the driver's eye movements to detect distracted driving. A secondary goal is to make the system highly modular, so new components can be easily added in or create interactions between modules.

2 Background

2.1 Gaze Detection and Tracking

Gaze detection and tracking has many positive implications for driver safety, namely to detect if a driver is distracted or drowsy. The US department of transportation found that in years 2011-2015 an overall 2.5% of fatalities were caused by drowsy driving [3], and for distracted driving in 2018 it was found that 8% of fatalities were distraction-affected [4]. Currently there are a wide variety of well-established and novel techniques for gaze tracking as found by a survey by Chennamma and Xiaohui [1].

2.2 Pedestrian Tracking

OpenCV includes the implementation of the Histograms of Oriented Gradients (HOG). The implementation details in terms of human detection are found in this paper[2]

2.3 Road Sign Tracking

3 The Challenge

This space has a number of challenges. While any one component of this project has many implementations even within the computer vision space, making them all work together in real time on mid to low end hardware. In addition, gaze tracking with a single camera is a more difficult area with no simple, of-the-shelf implementation in openCV.

4 Goals and Deliverables

4.1 Primary Goals

The primary goals for this project is to create a suite of car safety features which are solved using computer vision. The three main safety features involves gaze tracking, road sign and traffic light detection, and pedestrian detection. Each play a role in distracted driving incidents where a driver's gaze might not be focused on a pedestrian, road signs, or traffic lights when a driver should be.

4.2 Stretch Goals

The primary stretch goal for us to add an interaction between the modules. As a proof of concept, we would implement a system to ensure a driver has actually looked in the direction of a pedestrian or sign. An additional goal would be to validate it running on a Raspberry Pi Zero as a proof it can be embedded into an actually.

4.3 Evaluation

As this system is made up of many components, each element must be evaluated individually for success.

- Gaze Tracking
  -
- Pedestrian Tracking
  -
- Road Sign Tracking
  - Accuracy: 80% of signs correctly detected

- Be able to detect a sign in <500ms after appearing in frame

- Raspberry Pi Port
  - All targets must still be met when running the code on a Raspberry Pi Zero
- Module Interaction
  - Using the pedestrian or sign tracking, the system can detect if the driver has looked in the direction of the detected object after 5 seconds.

5 Schedule for Completion

Generally speaking, this project is split into 3 sub-projects for gaze tracking, pedestrian tracking, and road sign tracking. These 3 components will be developed seperately; however, near the end they will all be integrated together into a comprehensive suite of computer vision technologies to aid with automotive safety.

Week	Task Summary	
Week 1	**Adam**	a,b,c
	Christian	Finding papers regarding person detection
	Conner	Preliminary research, scope out papers in eye tracking
Week 2	**Adam**	a,b,c
	Christian	Start implementing pedestrian detection system, test using web-cam
	Conner	Scaffold project structure, determine entities related to task, build a common object model
Week 3	**Adam**	a,b,c
	Christian	Finish implementing pedestrian detection system, test using web-cam
	Conner	Begin implementing eye tracking, gather test data
Week 4	**Adam**	a,b,c
	Christian	Refactor pedestrian detection implementation to fit within modular design goal
	Conner	Continue implementing eye tracking, integrate with test data to establish a test bed
Week 5	**Adam**	a,b,c
	Christian	a,b,c
	Conner	Finish implementing eye tracking, provide a high-level API for working with eye tracking functionality
Week 6	**Adam**	a,b,c
	Christian	a,b,c
	Conner	Perform testing, gather result data for tracking accuracy, determine edge cases, make improvements where applicable
Week 7	**Adam**	a,b,c
	Christian	a,b,c
	Conner	Integrate high-level API with main car safety application, begin making UI for working with feature
Week 8	**Adam**	a,b,c
	Christian	a,b,c
	Conner	Finsih UI for interacting with feature, build notifications
Week 9	**Adam**	a,b,c
	Christian	a,b,c
	Conner	Continue week 8 work
Week 10	**Adam**	a,b,c
	Christian	a,b,c
	Conner	Work on strech goals / rasberry pi port, prep for final demo
Week 11	**Adam**	a,b,c
	Christian	a,b,c
	Conner	Package final deliverables and perform demo

References

[1] HR Chennamma and Xiaohui Yuan. A survey on eye-gaze tracking techniques. *arXiv preprint arXiv:1312.6410*, 2013.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. 1:886–893 vol. 1, 2005.

[3] US Department of Transportation. Crash stats: Drowsy driving. *Traffic Safety Facts*, DOT HS 812 446, 2015.

[4] US Department of Transportation. Research note: Distracted driving. *Traffic Safety Facts*, DOT HS 812 926, 2018.