**Oakland University**

**School of Electrical & Computer Engineering**

**Winter 2023**

**ECE 5415**

**PROJECT PROPOSAL :**

**ADAPTIVE CRUISE CONTROL WITH AUTOMATIC SPEED REDUCTION IN CASE OF TRAFFIC SPEED VIOLATION**

**Team Members:** Maya Karuthedath

Niranjan Chennasamudram Balaji

**Submitted:** Mar 12, 2023

**ABSTRACT**

The proposed project aims to develop an autonomous car with advanced driver assistance systems (ADAS) using Arduino and Pixy 2 vision sensors. The ADAS systems include forward collision control, adaptive cruise control, lane keeping, traffic light recognition, and speed reduction in case of traffic speed violation. We are using speed control here , for demonstrating a closed loop control system.

We plan to build on our first project , i.e, object follower robot , with additional sensors for feedback , PID control algorithm using MATLAB for speed control , to accomplish this. Now that we have familiarized ourselves with pixy 2 , we would like to explore more of its features like detection and tracking of lines , which can be used in our lane keeping feature. It will also be used in traffic speed sign detection . The forward collision control will be accomplished with an ultrasonic sensor.

TABLE OF CONTENTS

[List of Figures 1](#_Toc129351713)

[List of Tables 2](#_Toc129351714)

[Chapter 1. INTRODUCTION 3](#_Toc129351715)

[Chapter 2. OBJECTIVES 6](#_Toc129351716)

[Chapter 3. APPROACH 7](#_Toc129351717)

[Chapter 4. POTENTIAL BENEFITS 8](#_Toc129351718)

[Chapter 5. MILESTONES AND SCHEDULE 9](#_Toc129351719)

[Chapter 6. PERSONNEL 10](#_Toc129351720)

[Chapter 7. FACILITY AND RESOURCE 11](#_Toc129351721)

[Chapter 8. BUDGET 12](#_Toc129351722)

[Chapter 9. REFERENCES 13](#_Toc129351723)

List of Figures

[Figure 1.1 Adaptive cruise control 3](#_Toc129351899)

[Figure 1.2 Forward collision control 3](#_Toc129351900)

[Figure 1.3 Lane keeping 4](#_Toc129351901)

[Figure 1.4 Closed loop control system 4](#_Toc129351902)

List of Tables

[Table 5.1 9](#_Toc127358821)

[Table 6.1 10](#_Toc127358822)

[Table 8.1 12](#_Toc127358823)

[Table 9.1 13](#_Toc127358824)

# INTRODUCTION

Adaptive Cruise Control (ACC) is an advancement of cruise control system. It’s an automotive feature allows the vehicle to adopt set vehicle's speed to the traffic environment. Adaptive cruise control can also improve fuel efficiency by optimizing the speed of the vehicle based on traffic conditions.

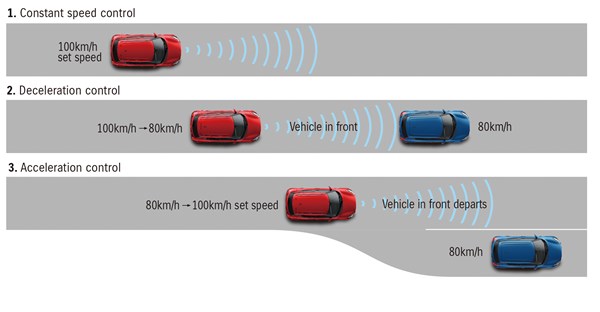


Figure . Adaptive cruise control

Forward collision control is a feature that uses sensors to detect objects in front of the vehicle , closer than a critical distance and warns the driver or automatically applies the brakes to avoid a collision. This system can help prevent accidents caused by distracted or fatigued driving, as well as reduce the severity of accidents by slowing down the car before impact.

A picture containing text, monitor, screenshot

Description automatically generated

Figure . Forward collision control

Lane keeping features are a set of technologies that help drivers stay within their lane on the road. This system uses cameras and sensors to detect the lane markings on the road and provides visual or audio alerts if the vehicle begins to drift out of the lane. Some systems can also automatically adjust the steering to keep the vehicle within the lane.

A group of cars on a road

Description automatically generated with low confidence

Figure . Lane keeping

For all these features , there’s a need to set a particular speed and achieve that value. We are employing a sensor and using PID control for the resulting closed loop feedback system.

Diagram

Description automatically generated

Figure . Closed loop control system

We are planning to implement the speed control using PID control. As we know by now , the PID controller uses three main terms: the proportional term, the integral term, and the derivative term, to adjust the input in order to reduce the error and maintain a stable motor speed. The proportional term responds to the current error, the integral term accumulates past errors, and the derivative term predicts future errors.



PID control involves:

* Selecting an appropriate PID algorithm (P, PI, or PID)
* Tuning controller gains
* Simulating the controller against a plant model
* Implementing the controller on our system

Arduino has a built-in pid library and code , to which these values can be input.

# OBJECTIVES

Through this experiment we aim to build a car that supports several ADAS features and used closed loop control system with feedback for achieving this. Our objectives for this experiment are:

1. Implement basic speed control using PID algorithm , i.e , match a set speed.
2. Implement Adaptive cruise control and adjust vehicle speed to match the environment.
3. Implement forward collision control using ultrasonic sensor.
4. Implement lane keeping using pixy 2’s line tracking feature.
5. Implement speed reduction on traffic speed violation using pixy for reading traffic speed sign using speed control to reduce speed.
6. Learn how to apply all the theory learned in class about control theory and closed loop control system into an actual system.
7. Learn how to effectively tune PID values for the system.
8. Test and validate all features.

# APPROACH

We already have a motor chassis built , for our first project and enough familiarity with pixy 2 , to focus on all the other feature related tasks. This will allow us to delve straight into the project , without much background work. Based on the described objectives , our implementation approach is :

1. Learn Pid control techniques using MATLAB and decide on the values for our system speed control.
2. Implement forward collision control using ultrasonic sensor.
3. Implement Adaptive cruise control.
4. Implement lane keeping feature.
5. Implement traffic speed sign recognition and speed reduction.
6. Integrate PID control to the code to ensure set values are matched.
7. Test each feature.
8. Test the systems with and without PID control and analyze results.

# POTENTIAL BENEFITS

There are many benefits to building an autonomous car with Adapative cruise control , forward collision avoidance , lane keeping , traffic sign detection and speed reduction. In a broader sense ,

1. Increased safety: Forward collision control and adaptive cruise control can help prevent accidents caused by human error . Lane-keeping features can help prevent accidents caused by unintentional lane drifting. Traffic sign recognition and speed reduction can help prevent accidents caused by over speeding and may even help us escape a traffic ticket .
2. Improved traffic flow: Adaptive cruise control and traffic speed sign detection and speed reduction features can help reduce over speeding , traffic congestion and maintain smooth traffic flow.
3. Reduced emissions: The autonomous car with adaptive cruise control can help reduce emissions by maintaining a consistent speed, reducing unnecessary acceleration and deceleration, and optimizing fuel efficiency.
4. Increased accessibility: Autonomous cars can increase accessibility for individuals with disabilities, senior citizens, and others who may have difficulty driving or navigating traditional vehicles.

On a personal note ,

* It would be satisfying to get to know and create autonomous driver assistance features that we have enjoyed using .
* Getting to know and implement PID control for our system would help us gain a greater understanding of the theory behind it and would be helpful for us in the future when building other systems.

# MILESTONES AND SCHEDULE

|  |  |
| --- | --- |
| **MILESTONE** | **SCHEDULE** |
| PID algorithm development and testing | 3/20/22 |
| Implement forward collision control using ultrasonic sensor and test | 3/24/22 |
| First progress report | 3/27/22 |
| Implement adaptive cruise control and test | 3/29/22 |
| Implement lane keeping feature and test | 5/4/22 |
| Implement traffic sign detection and speed reduction and test | 5/7/22 |
| Second progress report | 5/10/22 |
| Integrate PID control and test all features | 5/11/22 |
| Complete demo ,ppt report | 5/15/22 |

Table . Schedule

# PERSONNEL

Maya Karuthedath: Masters major is embedded systems. Experience in automotive embedded product development and design. Experience in working on real-time systems , programming knowledge and testing and debugging skills , will be of use in this project.

Niranjan Chennasamudram Balaji : Masters major is embedded systems design. Experienced as a data analyst, developed data warehouse concepts. The experience gained helps to use the data collected from sensors and use the data to make decisions about how to control a system in real time.

|  |  |
| --- | --- |
| **Task** | **Team member** |
| PID algorithm development and testing | Maya |
| Implement forward collision control using ultrasonic sensor and test | Maya |
| First progress report | Maya, Niranjan |
| Implement adaptive cruise control and test | Niranjan |
| Implement lane keeping feature and test | Niranjan |
| Implement traffic sign detection and speed reduction and test | Niranjan |
| Second progress report | Maya, Niranjan |
| Integrate PID control | Maya |
| Test and validate all features | Maya |
| Complete demo ,ppt report | Maya, Niranjan |

Table . Work break-up

# FACILITY AND RESOURCE

We need the following parts for our project:

1. DC Motors + Wheels + Chassis:
2. Arduino UNO
3. L2938N Motor driver
4. Pixy 2 camera
5. LiPo Battery -Sypom 1000mAh, 11.1V
6. HC SR04 ultrasonic sensor
7. KY-040 rotary encoder

We already have the required DC motors , wheels and the chassis , Arduino UNO. Professor has graciously agreed to lend us a pixy 2 camera for the project. We will buy the L2938N motor driver . And we also have the Li Po battery, ultrasonic sensor. We will buy the rotary encoder.

# BUDGET

|  |  |
| --- | --- |
| **Part** | **Cost** |
| DC Motors + Wheels + Chassis: | NA |
| Arduino UNO | $40 |
| L2938N Motor driver | $5 |
| Pixy 2 camera | NA |
| Battery | NA |
| HC SR04 ultrasonic sensor | NA |
| KY-040 rotary encoder | $10 |

Table . Parts and expenses

# REFERENCES

|  |  |
| --- | --- |
| Computer vision | <https://www.sas.com/en_us/insights/analytics/computer-vision.html> |
| Pixy 2 | <https://dronebotworkshop.com/pixy2-camera/> |
| Easy Object Following Robot using Arduino and PixyCam | <https://www.youtube.com/watch?v=w_krOCBk1DE> |
| Autonomous car | <https://www.slideshare.net/nadaashraf12/final-graduadtion-book-autonomous-car> |
| DC motor speed control | <https://www.youtube.com/watch?v=HRaZLCBFVDE> |

Table . References