ECE417/ECE598 Final Project

Vikas Dhiman

April 6, 2022

Final Project goals

The ECE417 final project will be done in teams of 2. You will choose your own project partner. You can post on Brightspace that you are looking for teammates if you cannot find one in The goal of the ECE417 final project is to demonstrate your skills in the following areas,

1. Ability to use and apply Pinhole-camera model, transformations of coordinate frames, geometry and in general linear algebra to 3D vision and robotics problems.
2. Ability to use and integrate third-party ROS packages into your own code.
3. Team work and time management.
4. Curiosity to learn new concepts on your own and ability to learn independently.

Your project report will justify how you have achieved the above objectives. Furthermore, your project

1. must consist of substantially your own created work. Utilizing supporting code to realize your design is acceptable, but it must be *clearly demarcated* what is your code and what is supporting code you have integrated. You must clearly state the source of supporting code. Failure to follow this practice will count as academic dishonesty. Ideally, your code and supporting code will be part of different ROS packages, but you can also borrow supporting code and comment which lines of the code are yours and what were borrowed along with the source.
2. must integrate some functionality into your project that we have not explicitly covered in class.

Final project ideas and options

There are a few options that are applications of the concepts that we have learned in the class or the labs,

1. Implementation of camera calibration using two mutually orthogonal checkerboards (as done in the sample exam). If you are an ECE-598 student, you will read, understand and implement [1] instead. You are allowed to use, hack and modify the existing ROS camera-calibration node, but you have to implement your own math.
2. Implementation of Birds eye view system with four cameras. You will extend the Bird's eye view lab to implement 360 degree parking view in Gazebo. To accomplish this, you will need to edit the prius-light.urdf file and enable four cameras instead of one; one on each side of the car. Then you will use the rotations and transformations between the four cameras to combine four bird's eye-views into one. ECE-598 students are additionally expected to park the gazebo car when the parking spot is in a visible range without any obstacles. You will use M-city in Gazebo for the testing.
3. Recommended for ECE-598 students only: Monocular road plane estimation. This is an implementation of Midterm 2, problem 1(b) in the Gazebo car demo setting. As the car is moving, can you estimate the road plane from two consecutive frames? You will have to use feature matching to find correspondence points. You can use a third-party road segmentation algorithm, like this one and

4. Recommended for ECE-598 students only: Lane following. Use a third-party lane detection implementation like this one and then translate 2D lanes to 3D lanes, and manipulate the `prius_msgs/Control` topic to move the car along the lane in the gazebo.

You can choose another project, but it will be an exception rather than a rule. If you are already working on a project (capstone or graduate project), and you can find a way to integrate the goals of this final project within your capstone or graduate project, then you are encouraged to integrate the two projects. In both the cases, please do not wait for project proposal deadlines, talk to me in person during my office hours or during the class. Also, don't be scared with the scant project description. Help will always be given at UMaine to those who ask for it.

Final Project submissions

You will maintain a single running document that will be submitted every week on Monday before class at 1pm. This document will act as your planning document, your project logbook, and final project report.

You will submit the following documents

1. Project planning document (due April 11)
2. Project progress and reassessment document (due April 18)
3. Project progress and reassessment document. Q&A ready. (due April 25)
4. Final Project documentation, report and video (due May 2nd)

Project planning document (due April 11)

The project team, proposal, design and planning document will consist of a PDF file addressing the following questions,

1. What is the end-goal of your project? In a proposal, you should not specify a single outcome, but a range of projects, from minimal to ambitious. The minimal outcome should make sure that you have something to show, while ambitious outcome will be the perfect project of your dreams. You should develop the timeline by making minimal outcome ready as early as possible, and then work towards adding more features towards the ambitious outcome.
2. Who is in your team?
3. How will you break down the project into smaller and manageable parts (function, class or ROS node)? Which parts of your project will you implement, and which parts will be borrowed from the internet or third-party library. It is recommended that you convey these parts of the project using a block diagram. How will each part interact with each other?
4. How can each part (function, class or ROS node) be tested? What is the expected range of inputs? What is the expected output for each input?
5. Anticipated timeline for the project, including research on finding the desired modules; learning new ROS packages; evaluating, adapting, testing and debugging the existing modules; anticipated man-hours for each step. Since the project is spread over 3-weeks, we expect you to spend total 6 hours per week per person. A sample timeline could look liks this:

Project progress report (due April 18 and then on April 25)

1. Code with test procedures for the functions, classes, or ROS nodes that have been implemented. Each function, successfully implemented (passes your own specification of the function), will earn you grades towards the final project.
2. Updates to the project proposal timeline. Scaling up or scaling down the project expectations.

Task num	**Task**	**Person time (36 hrs)**	**Dependency**	**Schedule**
T1	Arranging for checkerboards	1 hr		Apr 11
T2	Detection and identification of checkboard in the image	11 hrs	T1	Apr 11-14
T3	Understanding the code of ROS camera calibration	7 hrs		April 18-20
T4	Implementation to find matrix P	5 hrs	T3	Apr 20-25
T5	Implementation to find matrix K from P	2 hrs	T4	Apr 25-29
T6	Connect everything together and final report	10 hrs	T2,T3,T5	Apr 25-29

3. Sticking points and challenges. Things that you will try next to address the challenges.

4. Your project should be almost done by April 25th. So that you can answer questions about your project.

Final Project report, demo and video (due May 2nd)

1. A 10-min Q&A will be scheduled with you to evaluate your understanding of the code that you wrote. The Q&A for each team will be scheduled only once between April 25-26th.

2. You must provide documentation of the finished project. This should consist of your initial design documents, a detailed description of the functionality of your final project, a listing of all of your code, a description of your code, and a brief conclusion covering any changes to the final project from your initial design and any future modifications you would like to make to the project. The description for the code at minimum should describe the functionality of each part, and a description of how your code in the modules accomplishes your functionality.

3. A final report and video demo of your project.

4. Reflections on your planning document. How could you have planned better? What are the future ideas you will like to try to add on to this project?

Grading

1. Your grade will depend on the C++ code successfully implemented, their complexity, and a final project Q&A session about your implementation (total 30% of the final grade).
   (a) Project proposal (5% of the final grade).
   (b) Project progress report 1 (5% of the final grade).
   (c) Project progress report 2 (5% of the final grade).
   (d) Q&A and final report (15% of the final grade).

References

[1] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334, 2000.