Carnegie Mellon University

Monocular SLAM for

24-783: Advanced Engineering Computation

Short Presentation of Project Proposal

Team: Will Code for Food

Aditya Ramakrishnan
Neel Pawar
Samiran Gode
Shri Kiran Srinivasan

Project Overview

SLAM is the computational problem of constructing or updating a map of an unknown environment while simultaneously keeping track of an agent's location within it.

What will our program do?

- **Small-scale monocular SLAM** system for static environment.
- **Primary functionality:**

  Feature detection, feature tracking, get camera poses, and then mapping the landmarks using factor graph optimization (GTSAM)

📌 Impact

Why did we choose this problem?

- **SLAM** is being increasingly deployed in a variety of real-world settings, from self-driving cars to mobile devices.

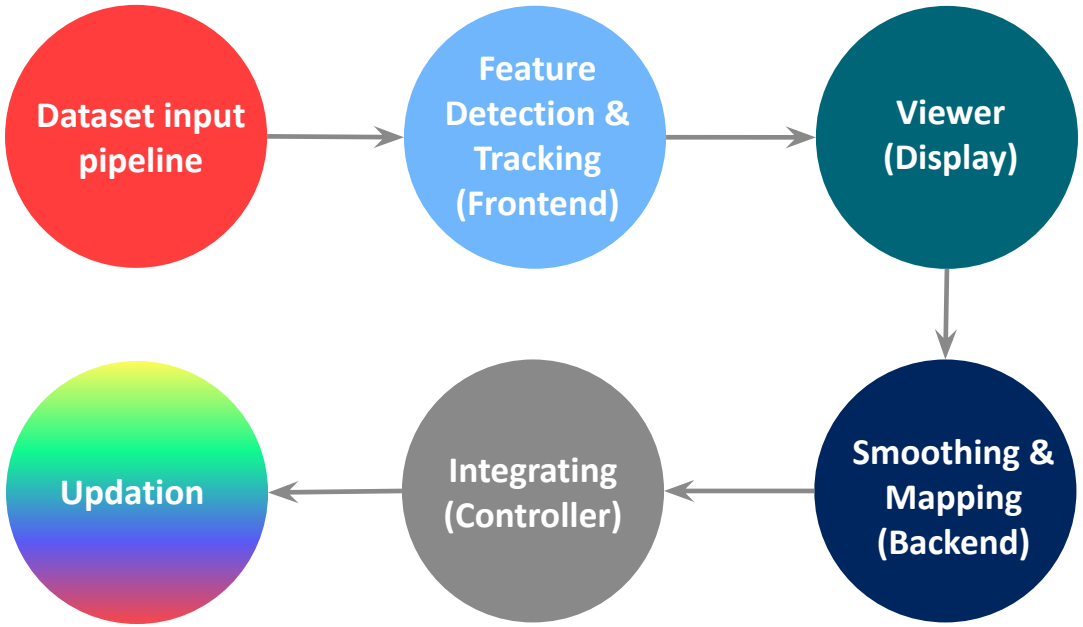
- **SLAM techniques** will be increasingly relied upon to provide reliable metric positioning in situations where infrastructure based solutions such as GPS are unavailable or do not provide sufficient accuracy.

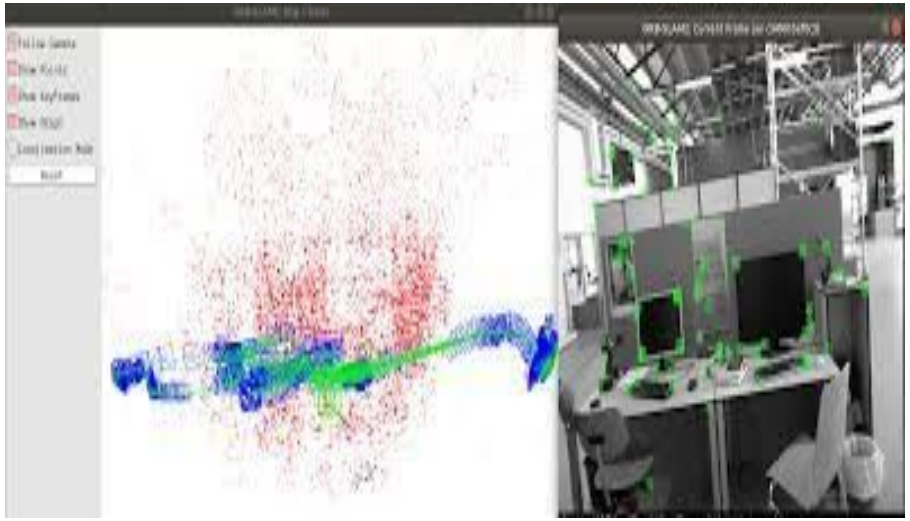
📌 Reasonable difficulty for the course

- This project will cover a vast amount of topics that we learn in this course.
- It will involve integrating a lot of low-level external libraries and multiple component and thus the build system will be fairly complex.
- Since there are multiple components some of which will be working together and others independently; multithreading will have to be utilised.
- Our data contains multiple frames of images, hence, data structures will have to be used effectively to ensure the program is fast.
- System will need to be designed choosing the right design pattern(tbd) and each class will have to be made using optimal constructors(move/copy) for efficient storage.

Our Process

Project Pipeline:





Source: Link

Carnegie Mellon University

Monocular SLAM for

24-783: Advanced Engineering Computation

Project Proposal

Team: Will Code for Food

Aditya Ramakrishnan
Neel Pawar
Samiran Gode
Sri Kiran

Project Requirements

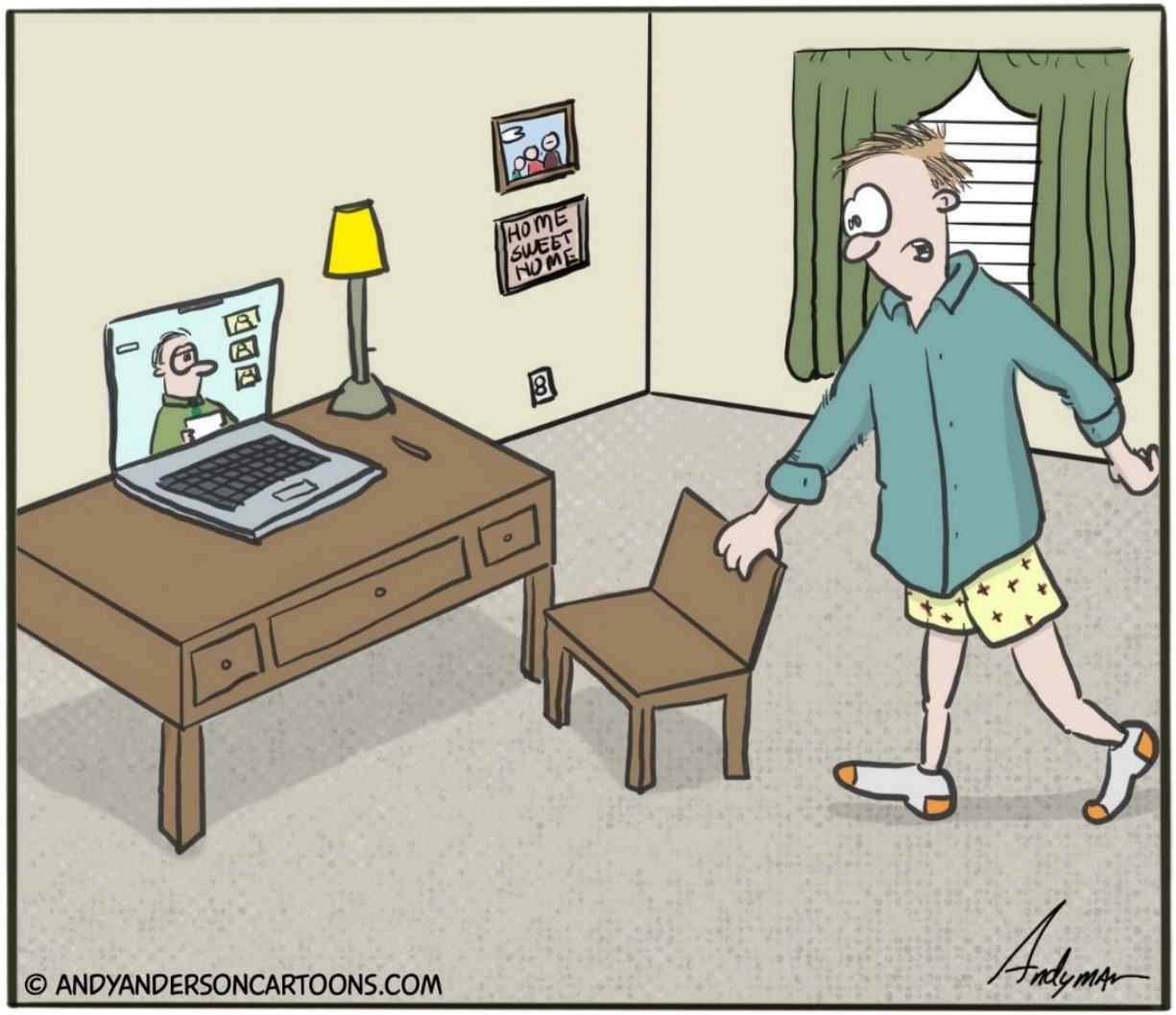
Project Description

- **Find and solve an engineering problem with C++. It can be a problem that you deal with on a day to day basis. Or, it can be part of your research project. For example, you may be doing repetitive and painful manual operations over and over again for your research, which may not be a focus of your research. Such manual operations may be made more ecient by writing a program that automates the process. Another example can be an modeling tool specialized for your problem. We often use a general-purpose CAD program to generate test cases, sample models, etc. However, a general-purpose CAD program may not be most ecient for your needs. Writing a program that quickly create those models may expedite overall research. Those are good candidates for your Option-A project.**

Overview

Web based mapping platforms can be unreliable!

- **Estimated Time of Arrival (ETA)** is never accurate!
- Heavy reliance on the **applications ability to track location accurately.**
- **Current solutions are not as robust** - real-time traffic prediction and navigation planning.



SORRY I'M LATE FOR WORK. TRAFFIC ON THE STAIRS WAS BRUTAL.

⚠ Existing Gaps

What are the current Issues?

- Efficient use of resources – **at the cost of accuracy** ☹
- **Urban development** presents a technical problem!
- **Technological advancements** (5G) still have **hiccups.**
- Inability to distinguish between **real and fake data-points.**

📌 Impact

User Groups & Pain Points

- **Professional workers** (Uber / Lyft / Delivery driver) :

  # Frequent navigation to different location

  - What are the shortest routes?

  - How can I get there in the quickest manner?

- **Local users :**

  # Looking for directions to travel locally

  - Navigation routes that adapt to real time traffic data

Product Opportunity

Where do we come in?

- Solution already exists!
- Augment the algorithm by providing reliable data points
- Account for real-time discrepancies

Goal

- Organize information and make it more useful
- Focus is on Customer Engagement
  - -> Revenue
  - -> Impact

Previous Work

- **Area mapping** algorithm based on OpenCV and SURF.
- **Image Matching Using SIFT, SURF, BRIEF and ORB:** Performance Comparison for Distorted Images
- **Automatic traffic density estimation** using Single Shot Detection (SSD)
- **Optical Flow Based Moving Object Detection** and Tracking for Traffic Surveillance

Our Process

Project Pipeline:

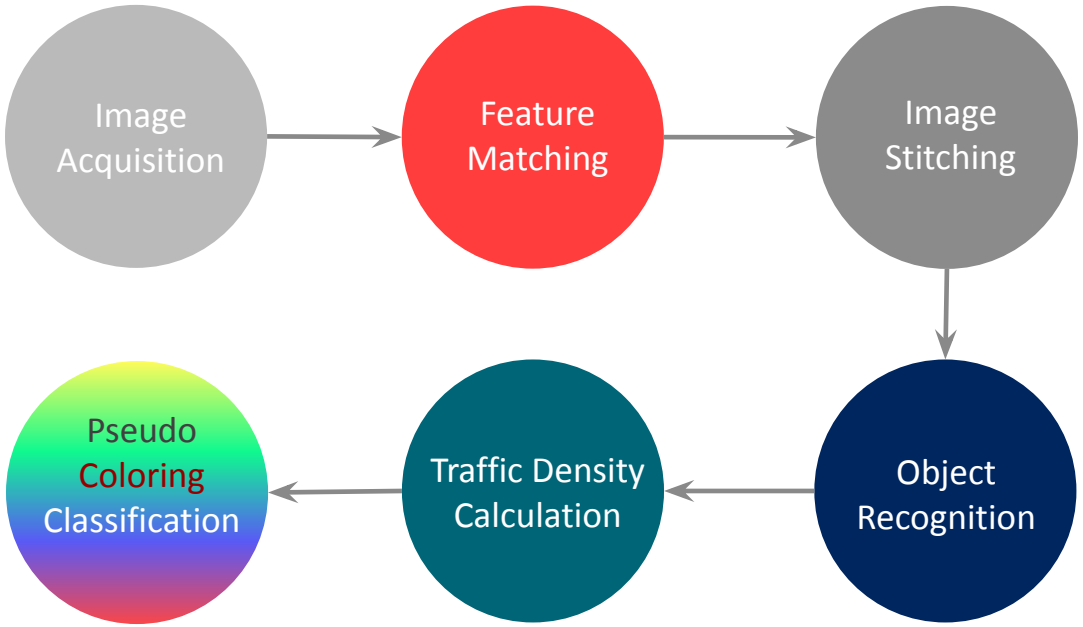


Image Acquisition

ORB Feature Matching

- **Compute Keypoints :** corner/edge/contour detection
- **Extract features :** Brute Force ORB
- **Compute distances :** between every descriptor pair

  Hamming Distance (ORB uses binary string based descriptors)

- **Select best matches**

Image Stitching

- **Estimation of Homography Matrix**
- **Warp source images :** Realignment for stitching
- **Stitch wrapped Images :** about every descriptor pair

Object Recognition

- **YOLO :** Deep learning based Object detection algorithm.
- **Dataset :** Microsoft Common Objects in Context (COCO Dataset)
- **Train Algorithm :** to detect various vehicle classes in a given image
- **Classify and Store :** the number of vehicles found in the image

Traffic Density Calculation

- **Calculate road_area :** the area of road with respect to the input image
- **Calculate Total_vehicle_area :** the sum of individual vehicle areas
- **Compute percentage occupancy:** by dividing the two areas

$$Road\ Percent\ Occupancy = \frac{Total\_vehicle\_area}{Road\ Area}$$

- **Note:** The Road area may differ based on the type of road (single lane, multi-lane, incoming, and outgoing traffic)

Future Scope

- **Pseudo-coloring (Mobility Heatmap)**
  - **Dataset showing the volume of traffic flow**

- **Data-driven city planning**
  - Dataset measuring street speeds of traffic.

Value proposition

Customers Satisfaction

- Reliable traffic density data
- Privacy protection (Big Brother data harvesting)

Economic Impact

- Increase in productivity and business revenue generation for local and professional consumers

Urban Infrastructure Planning

- Highways, Parking Lots, Metro Lines, Malls ...

Thank You!

- **Any Questions?**

  Please find us at WillCodeForFood@gmail.com

- **Please don't forget to provide your feedback!** ☺️