# Swimming Tango

An underwater navigation mechanism for deep-sea divers

Rishabh Ananthan
Anupam Goli

ECE 4012 L7A

# Project in a Nutshell

- Creating a means of underwater navigation (similar to a mobile GPS).
- Hardware size limited to the size of a small tablet.
- End goal is to produce a visualization of the underwater environment.

# Objective

- Design a device that attempts to reconstruct Google’s Project Tango using a Micron DST SONAR instead of the Kinect.
  - Kinect doesn’t work well in low-light conditions.
- Waterproof and will complement the diver’s breathing apparatus.
- Also enables divers to communicate with each other.

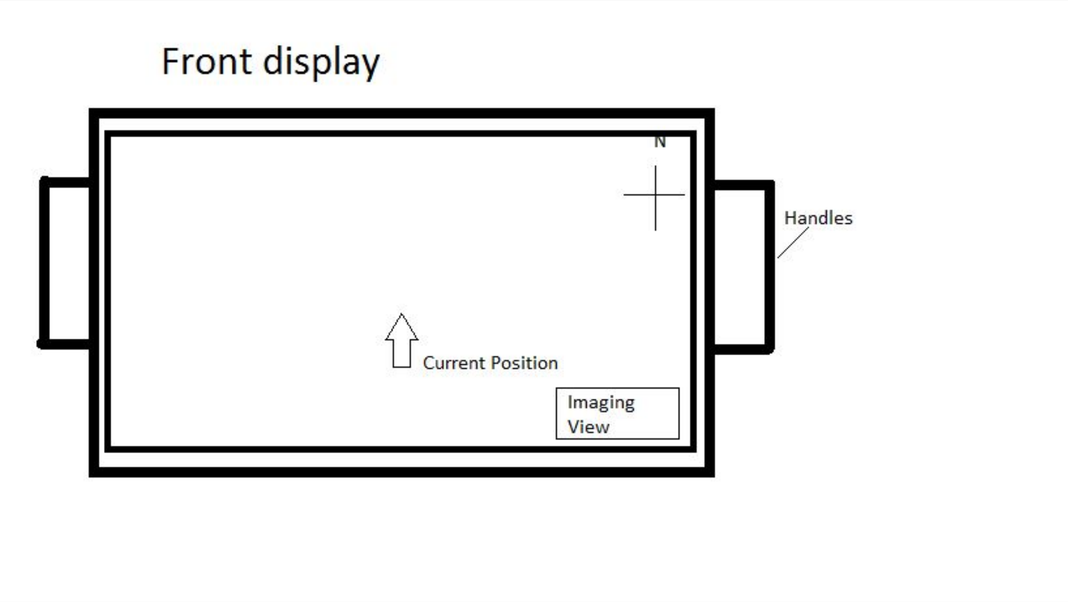
# Motivation

- Helps divers communicate in case of emergency.
- Allows divers to be found in case they lose their way.
- Enable autonomous exploration while mapping.

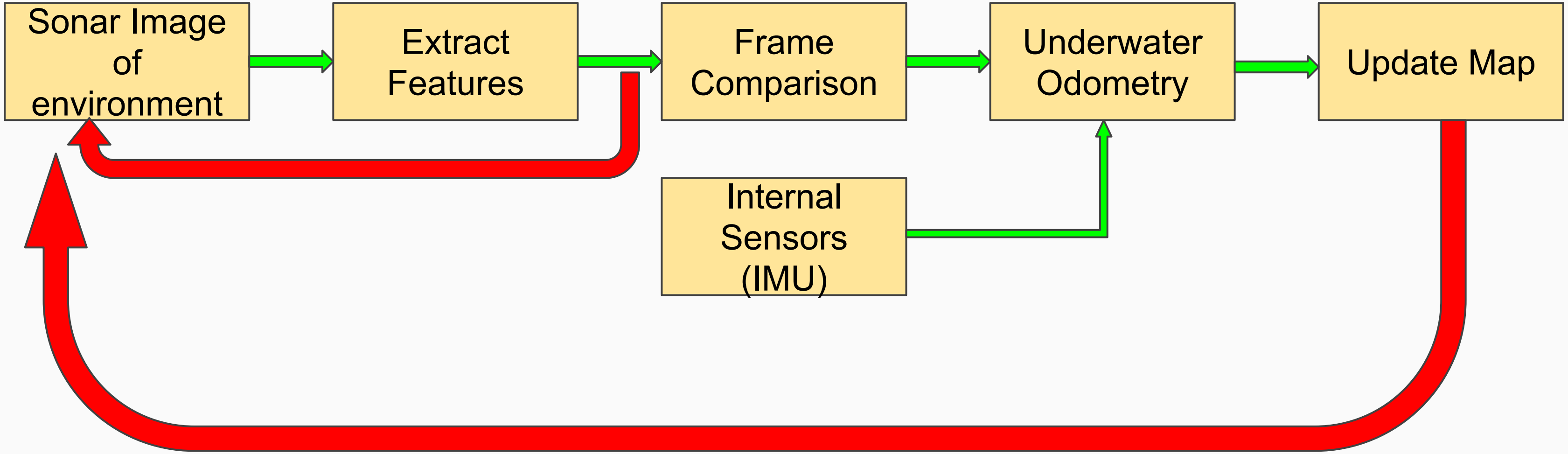
# Background

- Project Tango makes use of computer vision to help mobile devices to detect their position relative to their surroundings.
- Key algorithm used is SLAM. Tango uses an IMU and Kinect-like laserscan sensor. We intend to use the SONAR.

# Concept of Operations



# Concept of Operation Cont.



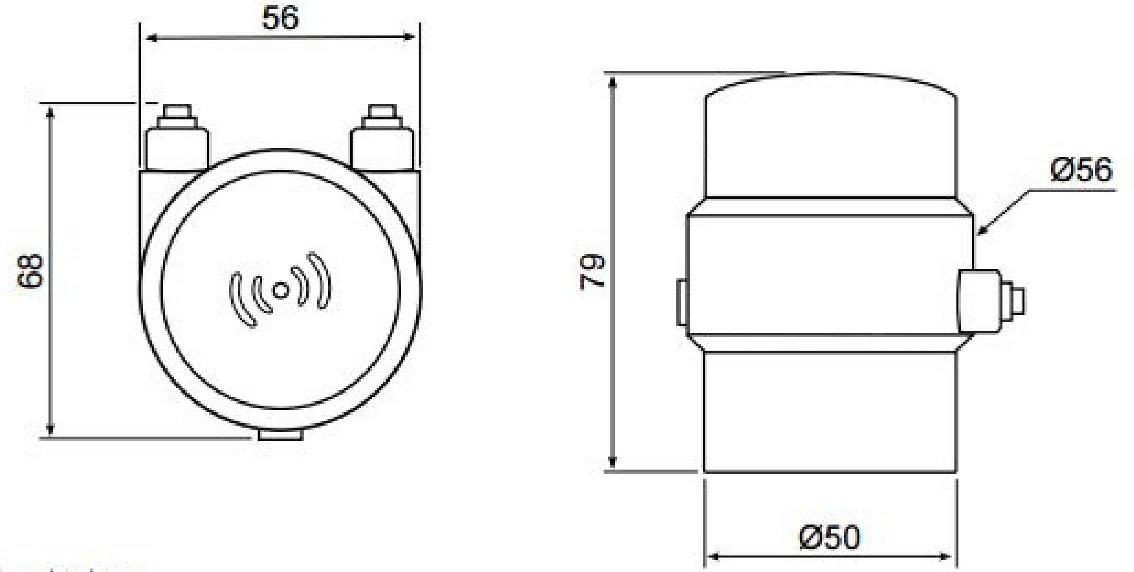
# Technical Specs of the Device

| Specification | Feature |
|---|---|
| Screen size | ideally between 4-7 inches |
| Thickness | preferably not more than 9mm (excluding SONAR) |
| OS | Linux with ROS |
| Max. depth | 66 Feet of Seawater (FSW) |
| Max. number of divers supported | 56 on one network |
| Max. communications range | 1000m |
| Navigations equipment | SONAR, IMU |

# Technical Specs of the SONAR

| Specification | Feature |
| --- | --- |
| Size | Smallest digital CHIRP SONAR in the world. Dimensions shown in the figure on the next slide. |
| Max. depth | 750m (around 2460 feet) |
| Max. range | 75m |
| Min. range | 0.3m |
| Power requirement | 12-48V DC at 4 VA average power |
| Communications protocols | RS-232, RS-485 (twisted pair) |

# SONAR Dimensions

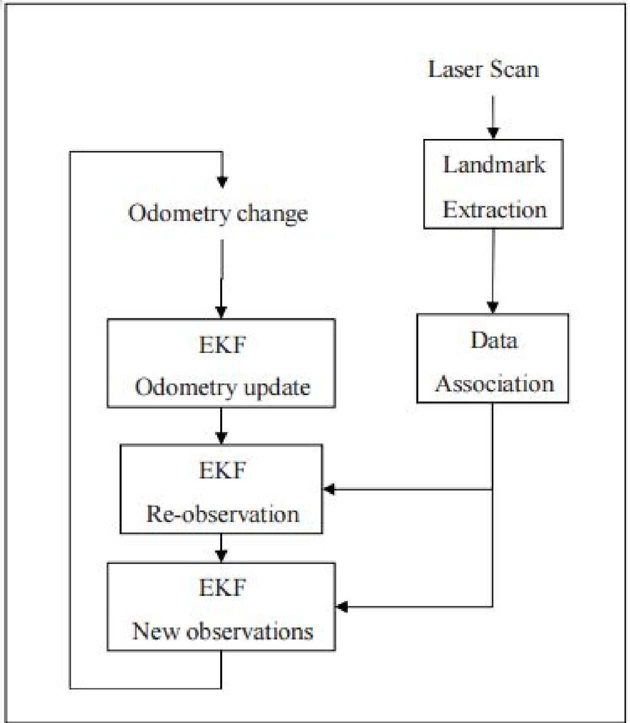



Not to scale, dimensions in mm.

# Simultaneous Localization and Mapping: SLAM

- Developed by Hugh Durrant-Whyte and John J Leonard.
  - Originally termed SMAL.
- Consists of:
  - Landmark extraction
  - Data association
  - State estimation
  - State update
  - Landmark update
- Requires a mobile robot and a range-estimation device (our SONAR).

# SLAM algorithm



# Visual Odometry

- Use features from different frames to discern distance traveled.
- Project Tango uses Visual Odometry to generate a map
- Useful to complement SLAM algorithm to map path traveled more accurately

# Robotic Operating System: ROS

- Open-source meta-operating system for robots.
  - Provides services such as hardware abstraction, low-level device control and package management.
  - Provides tools and libraries for obtaining, building, writing and running code across multiple computers.
- Custom sensor libraries.
  - Many open source implementations of various sensors in ROS exist.

# Mission Oriented Operating Suite: MOOS

- C++ cross-platform middleware for robotics research.
- Very useful for underwater robots.
- Can be bridged with ROS to add even more sensor integration.

# Codes and Standards

- Protocols for interfacing of sensors:
  - SPI
  - $I^2C$
  - USB
  - RS-232
  - RS-485
- FCC standards when interfacing with communications device.
- GNU GPL when using open-source software like ROS and MOOS.

# Design Constraints

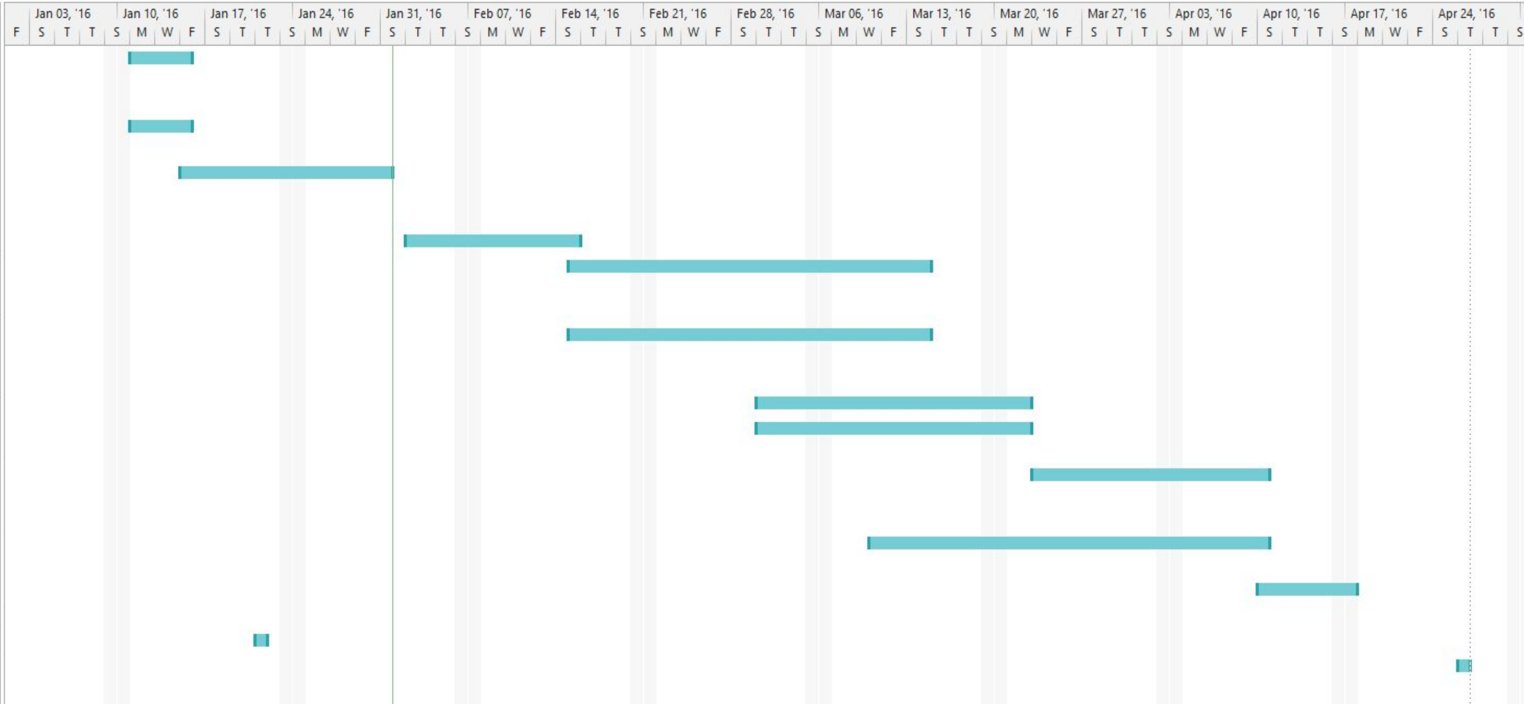
- Cost
  - Unsure of the budget provided by the sponsor.
  - The sensor will need to be bought
- Time
  - Only one semester available.
  - Unlikely to be sufficient to design the entire system from scratch.
- Size
  - Affects the choice of sensors.

# Potential Trade-Offs

- Size-related
  - Smaller, less-accurate sensors vs Larger, more-accurate sensors
- Power related
  - Battery life vs number and quality of sensors
  - Computational power vs battery life
  - Computational power vs quality of sensors

# Gantt Chart

| | Task Name | Start | Finish |
|---|---|---|---|
| 1 | Research and find an IMU and position sensor | Mon 01/11/16 | Fri 01/15/16 |
| 2 | Recreate Project Tango on Desktop | Mon 01/11/16 | Fri 01/15/16 |
| 3 | Set up ROS on test PC and try to collect data | Fri 01/15/16 | Sun 01/31/16 |
| 4 | Order parts | Tue 02/02/16 | Mon 02/15/16 |
| 5 | Implement Micron DST SONAR with ROS and MOOS | Mon 02/15/16 | Mon 03/14/16 |
| 6 | Implement IMU and position sensor with ROS | Mon 02/15/16 | Mon 03/14/16 |
| 7 | Create UI for device | Tue 03/01/16 | Tue 03/22/16 |
| 8 | Implement mapping algorithm with ROS | Tue 03/01/16 | Tue 03/22/16 |
| 9 | Gather feedback on UI and make changes | Wed 03/23/16 | Sun 04/10/16 |
| 10 | Implement SLAM and visualization | Thu 03/10/16 | Sun 04/10/16 |
| 11 | Test device | Sun 04/10/16 | Sun 04/17/16 |
| 12 | | | |
| 13 | Proposal Presentatio | Thu 01/21/16 | Thu 01/21/16 |
| 14 | Capstone Design Exp | Tue 04/26/16 | Tue 04/26/16 |



# Current Status

- Working on setting up research environment.
  - Project Tango
  - ROSpy
  - DST packages
- Installing ROS on a small laptop and setting up ROSpy.
- Found a McGill University Robotics GitHub repository for the DST Micron which will be used as a starting point.

# Resources Used

- SLAM
  - “SLAM for Dummies: A Tutorial Approach to Simultaneous Localization and Mapping”. http://ocw.mit.edu/courses/aeronautics-and-astronautics/16-412j-cognitive-robotics-spring-2005/projects/1aslam_blas_repo.pdf
  - “Simultaneous Localization and Mapping (SLAM): Part I The Essential Algorithms” http://www.cs.berkeley.edu/~pabbeel/cs287-fa09/readings/Durrant-Whyte_Bailey_SLAM-tutorial-I.pdf

- ROS
  - http://wiki.ros.org/ROS/Introduction

- MOOS