Laser Triangulation SLAM
Initial Plan

Author	*Samuel Martin*
Author ID	*C1319546*
Module	*One Semester Individual Project*
Module Code	*CM3203*
Module Credits	*40 Credits*
Project Supervisor	*David Marshall*

Description

Problem Definition

“Build a 3D map of an unknown environment using an affordable SLAM solution mounted to a small mobile platform”

Why is this is a problem? There are very few cheap, high resolution 3D SLAM systems available for amateur robotics platforms.
The closest sensor I could find to solve this problem was a £500 2D LIDAR module.
This is where most of the cost goes when implementing a mobile SLAM system.
The first priority is therefore to develop a cheap sensor made from readily available parts.
Next it would be necessary to develop a SLAM system to take advantage of the sensor’s data to map and connect multiple scans together. The final task would be to create a mobile robotic platform to test and prove that the entire SLAM system achieves its goal of mapping a small local environment.

Context

Usages

Being able to build a 3D model of an environment is exceptionally useful especially in restricted spaces which are inaccessible to humans. This ranges from small spaces to hazardous environments.
Example applications could include drain inspection, cave modeling and interior map building. For this project, I will be concentrating on interior map building.

Sensor Choice

Robotic platforms regularly rely on LIDAR, sonar and GPS for positioning. However, scanners for smaller robotic platforms are usually inaccurate, expensive or complex to implement.
A selection of SLAM sensors are listed below

- Impact based
  - Must make contact with their surroundings
  - Very low resolution and low accuracy
  - Very cheap and simple to develop
- Sonar
  - Low resolution and accuracy
  - Acoustic error when it comes to complex environments
- LIDAR
  - Expensive
  - Smaller units are usually 2D
- Camera motion
  - Requires environment to have enough matching feature points
  - Requires the platform to move through unknown environment
  - Requires high processing power and image analysis tools
  - Cheap and simple to implement
- WiFi / GPS
  - Only suitable for location acquisition
  - Sporadic error
  - Only works within range of transmitters
- Structured Light
  - Can be fabricated with cheap, readily available parts
  - Only works with IR or visible light in a dark room
  - Can be slow to complete a full scan
  - 3D high resolution

Processor Choice

Processing solutions are usually a trade-off between price and computing power.
Other factors to consider include power consumption, availability and physical footprint.

- Arduino
    - Small footprint and inexpensive
    - Low processing power for image analysis
    - Low connectivity
- Intel Galileo
    - Better connectivity than the Arduino
    - More expensive than the Pi
    - Still not enough processing power
- Raspberry Pi
    - Great connectivity
    - Enough processing power for most image analysis tasks
    - Supports webcams and its own Raspberry Pi Camera
    - Can't deal with real-time/analogue I/O
- Laptop
    - Excellent processing power and connectivity
    - Self contained
    - Too large for small projects and can be expensive
- Online processing
    - Requires peer to peer connectivity
    - Relies on a continuous connection to the controller
    - High connectivity overhead but unbeatable processing power

Project Aims and Objectives

Success Criteria

- **Accuracy**
  - The SLAM world model needs to be accurate enough to navigate around messy environments. For example a desk covered in books and papers.
- **Dynamicity**
  - The system should adapt to new environments automatically.
- **Speed**
  - The robotic platform should be able to scan an environment within a reasonable amount of time. For example, one scan and one 30cm route plan traversal should take no more than two minutes to complete.
- **Price**
  - The entire system's component value should be no more than £100

Tangible Milestones

- **Hardware - Mobile robotics platform development**
  - Design and build a capture rig to scan an environment using structured laser light
  - Design and build a rotating platform to gather a panoramic scan
  - Design and build a mobile platform to mount the scanner
  - Connect Raspberry Pi, power source, motors and sensors together
  - Set up a wireless connection and portable power supply
- **Stage 1 - 3D mesh construction**
  - Retrieve camera data quickly to ensure a fast scan
  - Analyse the depth component of the image
  - Convert the depth space to world space
  - Capture a panorama using a turntable system
- **Stage 2 - Navigation**
  - Break the 3D scan down into a voxel grid and calculate each voxel's traversability.
  - Create a point to point cartesian navigation system to move the platform around the environment safely. This should use the voxel grid defined above
- **Stage 3 - Multi-scan environment mapping**
  - Create a scan stitching algorithm to minimise odometry drift between scans
  - Create a world map consisting of multiple scans
- **Stage 4 - Autonomous navigation**
  - Develop an automated waypoint generation algorithm to autonomously explore an environment.

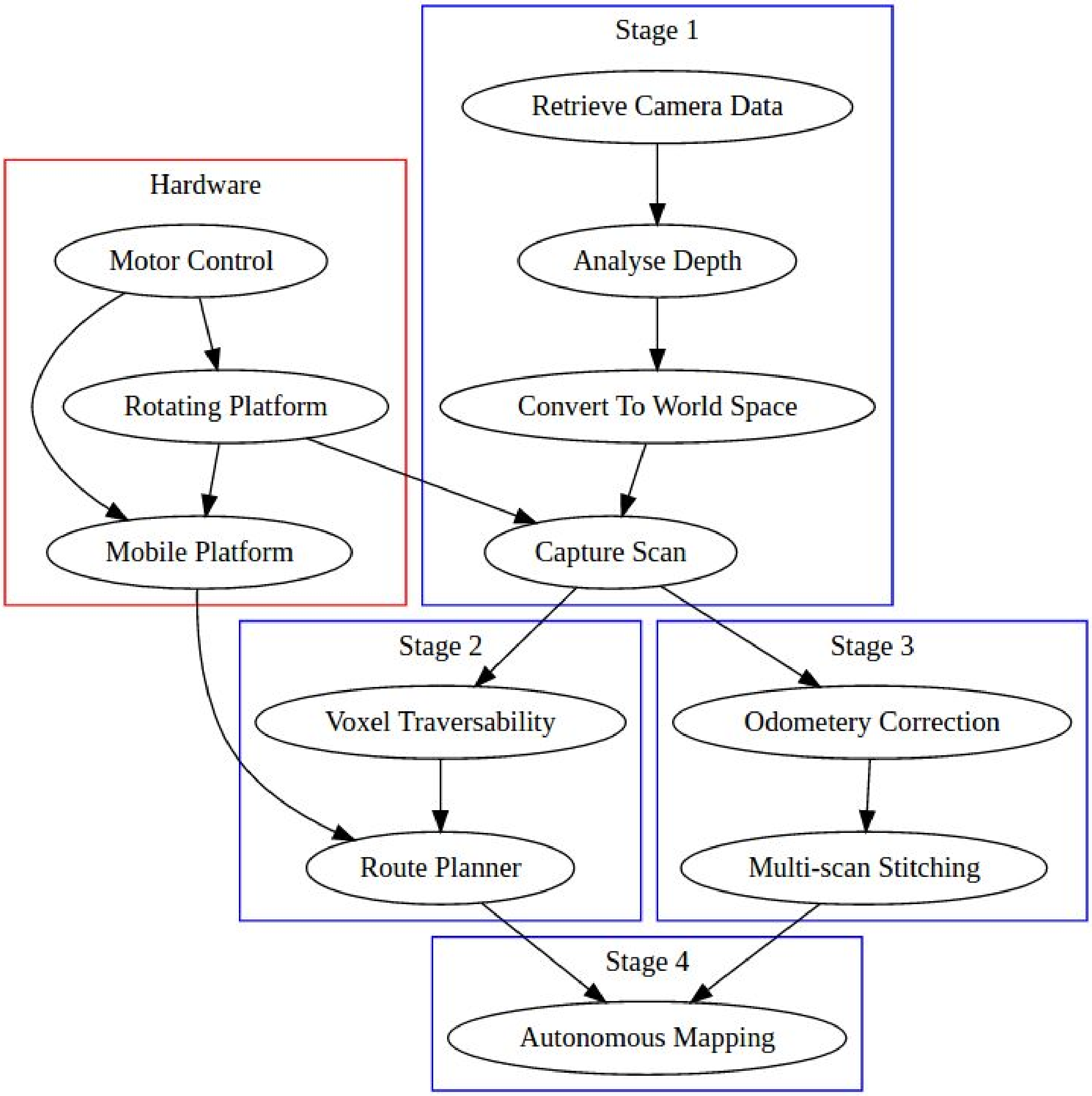
Work Plan

Supervisor meetings occur weekly at midday on Friday to discuss progress on the last week's work objectives and speculate the next week's challenges.
A diary will be held on a daily basis to record successes and failures of the system as well as the reasoning behind each decision.

- **Week 1 - 2**
    - Hardware
    - Stage 1
    - Deliverables
        - Demonstrate the sensor creating a single full 3D scan of an environment
        - Demonstrate the mobile platform being controlled by a wireless connection.
- **Week 3 - 4**
    - Stage 2
    - Deliverables
        - Demonstrate the planner taking a scan and outputting a traversability grid which the route planner can utilise to find a route to the next manually set waypoint.
- **Week 5 - 7**
    - Stage 3
    - Deliverables
        - Demonstrate the system compiling multiple scans into one world model.
        - Demonstrate the odometry correction algorithm to minimise drift.
- **Week 8 - 9**
    - Stage 4
    - Bug testing and edge case scenario testing
    - Deliverables
        - Demonstrate the system automatically scanning and traversing an unknown environment without human interaction.
- **Week 10**
    - Run extensive success criteria tests on the system
    - Summate the data and draw a conclusion on the project's success
- **Week 11 - 13**
    - Gather test data, diary, references and other material into the final report
    - Deliverables
        - Test Data
        - Final Project Report

Work Plan Dependency Graph

This graph shows each stage and their dependencies. This allows concurrent work on multiple sections if needed. It also shows choke-points which need to be overcome to progress in the project.



References

Calonder, M. EKF SLAM vs FastSLAM - A Comparison [Online]. Lausanne: Swiss Federal Institute of Technology. Available at: https://infoscience.epfl.ch/record/146805/files/ekf_fastslam_comp.pdf [Accessed: 29 Jan 2017].

Riisgaard, S et al. SLAM For Dummies [Online]. Avaliable at: https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-412j-cognitive-robotics-spring-2005/projects/1aslam_blas_repo.pdf [Accessed: 30 Jan 2017].

Abrate, F et al. Experimental EKF-based SLAM for mini-rovers with IR sensor only [Online]. Dipartimento di Automatica e Informatica: Politecnico di Torino, Italy. Available at: http://s3.amazonaws.com/academia.edu.documents/43367001/ECMR07_0020.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1485610018&Signature=bprkZDhso5%2FJaov3FsWvFrOp2SU%3D&response-content-disposition=inline%3B%20filename%3DExperimental_EKF-based_SLAM_for_mini-rov.pdf [Accessed: 30 Jan 2017]