Pathway

Damanvir Singh | Munib Syed

**Abstract**

Our project is an indoor navigation system/application, with the possible addition of an Indoor Positioning System (IPS) using Wi-Fi triangulation. We hope to create a user-friendly web-based app that will work in a variety of indoor environments, giving the user short directions to their goal destination. Since this is a web-app, it will be available on all platforms including OSX, Windows, IOS, Android, etc.

The project delves into the areas of computer vision, and OpenCV has been used to make portions of this task more achievable. We hope to achieve an efficient, scalable algorithm that can work universally as long as sufficient map data is provided, that can recognize and avoid as many obstacle types as possible.

**Purpose**

The reason that that the project exists is that there do not seem to be many apps out there specifically for indoor navigation. It also seems like indoor positioning is an unexplored area, which is why this project should exist.

The purpose of the project is to solve the issue of quickly navigating large indoor environments, such as a school or office building. Users can find efficient ways to reach their destinations and avoid getting lost in an unfamiliar building.

**Outcome**

The goal of our project is to deliver an efficient and scalable application which can work in any building as long as accurate map data (floor plans) is provided. The application will be able to read the map data, detect locations (defined by user), detect obstacles on the map including walls, boundaries or anything defined by the user as an obstacle, and then place nodes connected with edges in the walkable space on the map. This map data will then be imported into the path finding application. From here, the application will be ready to use for indoor navigation in the building the map was provided for.

\*If an IPS is implemented into the project, the user of the application will also be able to look at their location on the map and navigate to another location.

**Team Member Capabilities**

Daman - HTML, CSS, JavaScript, jQuery, Python, Flask, Path Finding Algorithms (Dijkstra’s Algorithm), Mapping libraries (Mapael and Leaflet)

Munib - Python, OpenCV, Geometry and Coordinates, Computer Vision, Path finding algorithms

**Initial Issue**

There are many initial issues, many of which are already being looked into. One is a large knowledge gap, as our project addresses some very difficult parts of computer science such as image analysis (computer vision) and networking (IPS). Knowing that building programs like this ourselves would be near impossible, we are making good use of free computer-vision centred libraries including OpenCV and Scikit-Image. We are already making progress with OpenCV, but there are still knowledge gaps present, such as working with NumPy arrays. Another massive gap has to do with the IPS, given that we know very little about networking and how our school’s Wi-Fi system works (which is our testing grounds). Building an IPS will be very difficult without an understanding of our capabilities and what is even possible in the first place.

**Deliverables and Timeframes**

|  |  |  |
| --- | --- | --- |
| **Deliverables** | **Duration** | **Completion Date** |
| Figure out how to generate a grid of nodes, and create an algorithm that can remove/ignore the nodes within obstacles (using coordinates of corners) to create a walkable space. Make sure we can deal with some simple obstacles/shapes, at the minimum. Test out different mapping libraries to serve our purposes. | 3 Weeks | End Week 5 |
| Research into more path finding algorithms in order to determine which serves our  purposes the best while still being efficient (Dijkstra’s and A\*).  Work on mapping libraries, and make sure custom maps can be used and manipulated. | 1 Week | End Week 5 |
| Have a prototype for our application working, which will be able to read in map data (floor maps), place nodes and edged on it while avoiding simple-shaped obstacle (squares and rectangles).  Have an efficient path finding algorithm working.  Be able to use custom maps on the mapping library | 1 Week | End Week 6 |
| Work on complex obstacle detection and complex map reading. Work on algorithm efficiency and be able to handle large amounts of map data.  Work on making the path finding application work with multiple floors in a building. Add more features to path finding application. Work on the application’s cross-platform ability. | 4 Weeks | End Week 10 |
| Research into IPS and test with different resources and methods. | 1 Week | End Week 11 |
| Build a IPS | 4 Weeks | End Week 15 |
| Integrate the IPS into the main application.  Work on the documentation | 1 Week | End Week 16 |

**\*There are high chances of encountering major problems in this project as it uses advanced computer science topics including computer vision and networking, topics that we have minimal or no knowledge in. So far our work in computer vision to read and process map data looks promising. However, some aspects of computer vision related work could take longer than mentioned in the “Deliverables and Timeframes”. If the computer vision related work takes longer than predicted, then the approach would be to drop the IPS from our project and focus purely on the computer vision related work and integrating that into the path finding application.**