CDE2310 Report G1
Concept of Operations

Group 6
Chong Kai Jie
Diwakar Vidya Adhavan
Li Zhuangda
Toh Ee Sen, Izen
Zhang Yuening

Executive Summary **3**
1.0 Problem Definition and Requirement **3**
1.1 Background 4
1.2 Assumptions and Constraints 4
2.0 Description of Architecture **4**
2.1 Real-Time Environment Mapping 5
2.2 Thermal Sensor 5
2.3 Flare Launcher (Ping-pong Launcher) 6
2.4 Mission Control 7
3.0 Risks and Potential Issues **8**
3.1 BOGAT 8
4.0 Preliminary Design **10**
References **11**
Appendix A: Glossary of Acronyms **12**
Appendix B: Glossary of Terms **13**

Executive Summary

This report integrates steps 1-5 of the EG2310 V-Model with the NASA CONOPs framework, into a concept of operations for our CDE2310 project. This report aims to detail the mission objective, our proposed solution, and specific implementation plans of this project.

1.0 Problem Definition and Requirement

Tasked to design an autonomous search and rescue (S&R) robot to be deployed in an unknown area to search for heat signals and fire flares to alert rescue teams for survivor assistance. Key features are listed in the following table.

Feature	**Requirement**
Fully autonomous robot	1. Using LiDAR sensor to map the surroundings. 2. Able to detect heat signals. 3. It must not collide with any obstacle. 4. Robot needs to approach the heat signal proximity. 5. Does not navigate to the same heat signal and fire flares. 6. The robot needs to be able to be fully autonomous and not remote controlled.
Fire flares to signal S&R teams for immediate survivor assistance	1. Launching mechanism must be able to fire 2 flares at set time interval 2. Flares need to be self reloadable as it is fully autonomous 3. Flares need to be fired high enough for S&R teams to be able to see. 4. Potentially able to unjam itself if jamming of flares occurs.
Power system (battery, rasp pi & other electronic components)	1. Battery needs to last long enough for the whole mission. 2. Needs to power the launching mechanism, rasp pi, motors, LiDAR and OpenCR. 3. Motors require a sufficient amount of battery remaining or it will stop working.
Mechanical structure	1. Robot needs to have a relatively low and central center of gravity such that it can navigate uneven areas (such as ramps) without toppling over. 2. Robot needs to be not bulky as it needs to navigate through tight spaces. 3. Robot needs to be mounted with a launching mechanism with a magazine for flare firing.

1.1 Background

The mission objective is to navigate a maze and identify two heat signals located at random positions within the maze. Once the heat signals are located, the robot will have to fire two ping-pong balls (replacement for flares) upwards at a specified time interval. An additional mission objective is a third heat source, located atop a ramp which the robot will have to climb.

The mission is a simulation of an earthquake search-and-rescue scene, with the heat signals being survivors and the ping-pong balls shot by the robot being flares to alter rescue authorities.

1.2 Assumptions and Constraints

Assumptions:

1. In this simulation, the robot operates in a controlled environment with level terrain and flat acrylic walls.
2. The only heat signals are those mimicking the survivors, there are no alternate heat signals that can possibly confuse the robot.
3. 2D LiDAR is sufficient to map the whole area without issues.

Constraints:

1. The time limit for this mission is 25 minutes.
2. Limited battery capacity
3. Budget: SGD$80

2.0 Description of Architecture (Concept Design)

In order to achieve the task of finding all casualties and launching flares (ping-pong balls), we are focusing development on 2 key capabilities: real-time environmental mapping and a ping-pong ball launching mechanism. Overall, this requires the integration of several key systems (see Fig. 2.0.1).

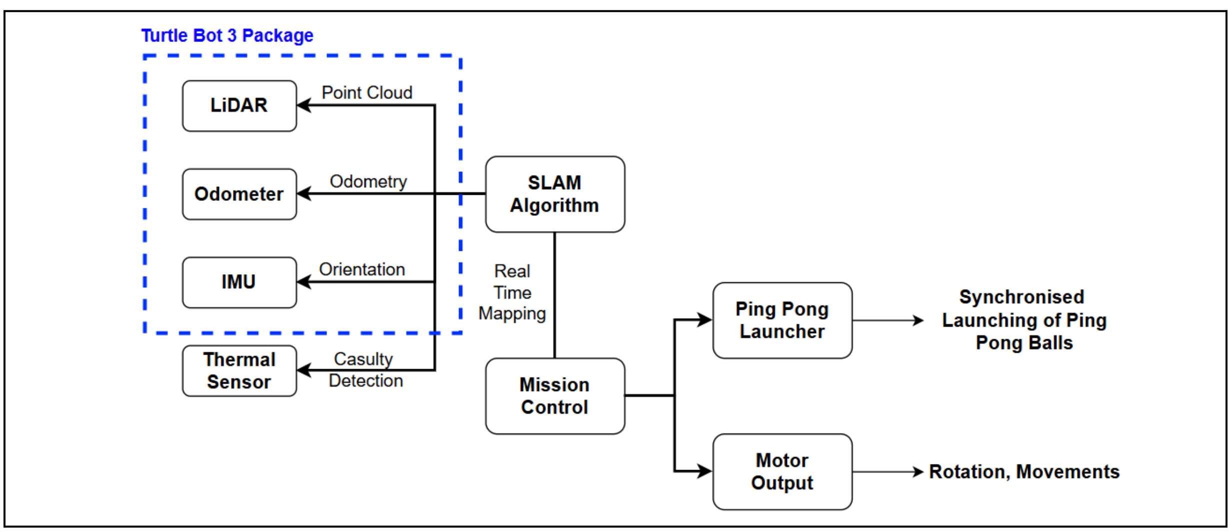

Fig. 2.0.1: System Architecture Diagram

2.1 Real-Time Environment Mapping

Our system interfaces with its physical environment using 2 main sensors, a LiDAR sensor and a thermal sensor, as well as the odometer packaged with the turtlebot's motors.
The sensor fusion framework described in Multisensor Data for Localization and Mapping in Agricultural Environments[1] as well as Hard Sensor Fusion: LiDAR and IR[2] serves as a useful reference for our task in integrating thermal and LiDAR data into a single map, which enables path finding algorithms in navigation towards objectives.

2.2 Thermal Sensor

As a means for detecting casualties and mapping out their position in SLAM, our team proposes an OTS rudimentary thermal camera[3], which provides low resolution thermal imaging and have **proven capability for detecting medium sized heat sources** (see Fig. 2.2.1), as seen in numerous past projects conducted by hobbyists,[4] and even in medical fields.[5] An alternative lower cost solution our group has considered was building an IR Sensor array from scratch, as further discussed below in section 3.1 BOGAT.

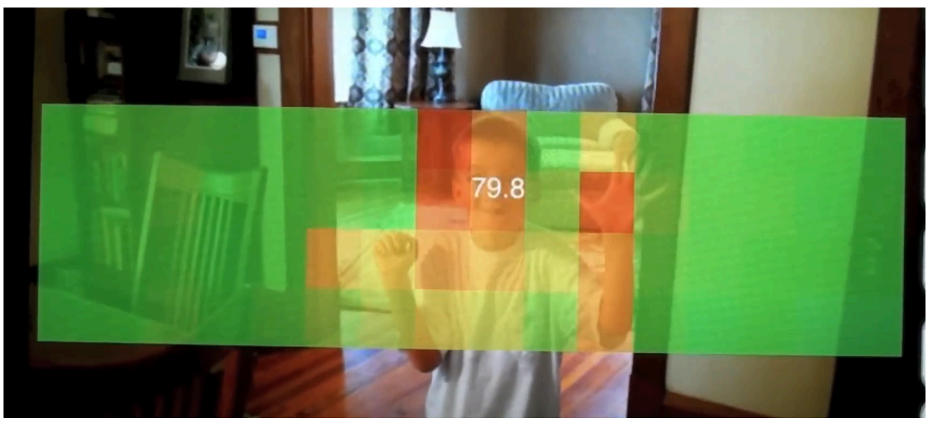

Fig. 2.2.1: Using a MLX90620 thermopile for detecting humans

Source: Benchoff, B. Hackaday.

[1] Pire, Taihú & Mujica, Martin & Civera, Javier & Kofman, Ernesto. (2019). The Rosario dataset: Multisensor data for localization and mapping in agricultural environments. The International Journal of Robotics Research. 38. 027836491984143. 10.1177/0278364919841437.
[2] Hall, Tutwiler, & Rimland. (n.d.). *Hard sensor fusion: LiDAR and IR*. Center for Multisource Information Fusion. https://www.eng.buffalo.edu/~nagi/MURI/MURI/Year_2_files/PDF%20Presentations/6%20-%20David%20-%20Hard%20Data%20Fusion%201%20LIDAR%20and%20IR.pdf
[3] Infrared detector arrays low cost thermal imaging. (2022). In https://www.boselec.com. BostonElectronics. https://www.boselec.com/wp-content/uploads/Linear/Heimann/HeimannLiterature/Heimann-Arrays-01-11-24.pdf
[4] Benchoff, B. (2012, November 12). Building a thermal imaging sensor from scratch. Hackaday. https://hackaday.com/2012/11/12/building-a-thermal-imaging-sensor-from-scratch/
[5] Muthukumar, K. A., Bouazizi, M., & Ohtsuki, T. (2022). An Infrared Array Sensor-Based Approach for Activity Detection, Combining Low-Cost Technology with Advanced Deep Learning Techniques. Sensors (Basel, Switzerland), 22(10), 3898. https://doi.org/10.3390/s22103898

2.3 Flare Launcher (Ping-pong Launcher)

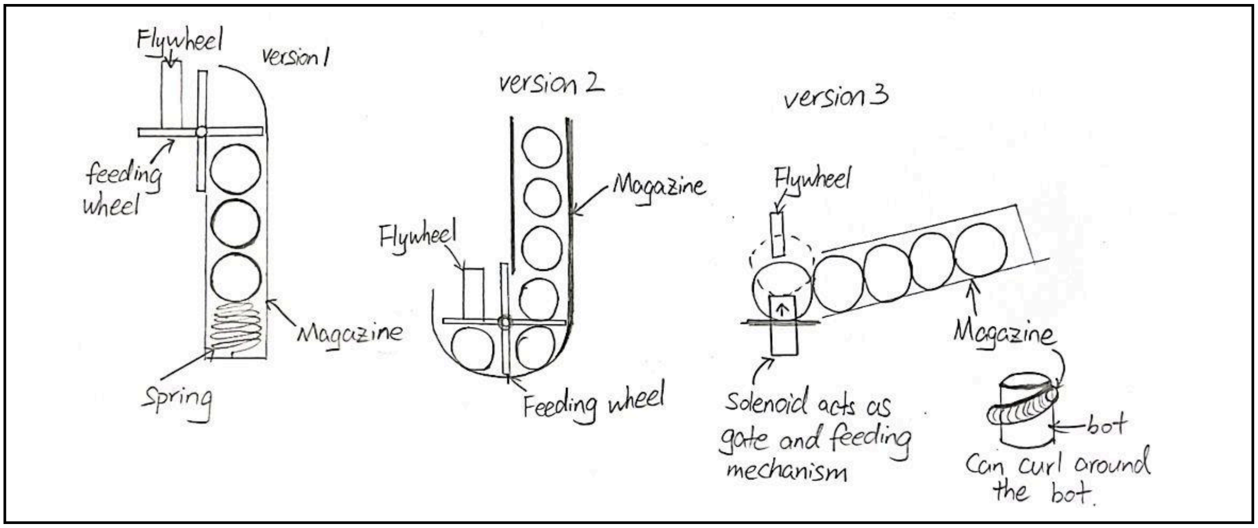

Fig. 2.3.1: Sketches of Ping-pong Ball Loading Mechanism

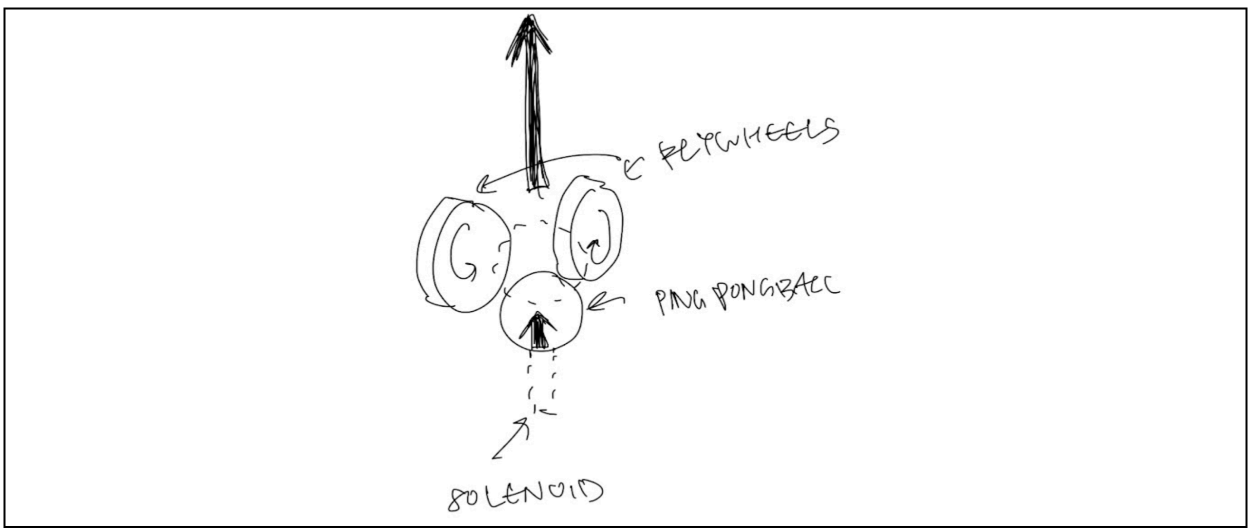

Fig. 2.3.2: Sketch of Flywheel Mechanism

For carrying the ping-pong balls, we intend to design and 3D print a tube-shaped launcher (see Fig. 2.3.1) that will wrap around the TurtleBot, ensuring that the ping-pong balls are carried securely while also not interfering with any of the TurtleBot's other systems (e.g LiDAR sensor). The launching mechanism will comprise of two flywheels, inspired by the work of Engineering Dads,[6] with more detailed design references developed by Butler N.[7] Finally, a solenoid actuator located at the intersection of the holder and the flywheels will be programmed to feed the ping-pong balls into the flywheels when the TurtleBot is in the correct position and at the correct time intervals.

---

[6] Engineering Dads. (2023, June 6). *Powerful Ping-Pong launcher* [Video]. YouTube. https://www.youtube.com/watch?v=MX3i9wQe3_g

[7] Butler, N. (2022, January 7). Automatic Ping-Pong Launcher - Nollaig Butler - Medium. *Medium*. https://nollaigengineering.medium.com/automatic-ping-pong-launcher-c77dc140d33b

2.4 Mission Control

Overall, our robot will proceed with the mission in 3 phases. 1. Search 2. Rescue 3. Final Task. This is detailed in the following table:

Phase	Activity
1. Search	Frontier-based exploration on the TurtleBot3 architecture[8] is well researched. This is crucial for the robot to obtain the full layout of the disaster area, taking notes of the positions of the casualties for later. The exploration algorithms are further discussed below in section 3.1 BOGAT.
2. Rescue	The robot navigates towards each of its objectives in a systematic matter, choosing its path based on the A* path finding algorithm.[9] Upon reaching each task, it will fire the flares (ping-pong balls), following the predetermined interval as per S&R SOPs.
3. Final Task	The robot navigates towards the exfiltration point (ramp) and scans for the final casualty, before firing its final flare and completing the mission.

3.0 Risks and Potential Issues

Technical Failures: Robots are complex machines with many moving parts, sensors, and software systems. Any malfunction in hardware or software—like a sensor failure, power issues, or programming errors—could cause the robot to behave unpredictably, potentially causing damage to people, property, or the robot itself.

Legal and Liability Issues: If a robot causes harm, it can be difficult to assign legal responsibility. Is it the developer, the user, or the manufacturer who is accountable? This ambiguity could make it complicated to resolve disputes or seek compensation in case of accidents.

Complex Internal Structures: This Robot has complex inside structures to allow its function, which may not function in the real circumstance under adverse environments during/ after earthquakes as all the testings are done indoors.

Heat Detection Limitations: The robot identifies survivors through heat sources which may be inaccurate if there are other heat sources such as machines buried under building pieces or other robots.

Robot Movement: The movement of robots in the earthquake environment may cause movement of building fragments, and may lead to further harm to survivors.

[8] Topiwala, Inani, & Kathpal. (n.d.). *Frontier based exploration for autonomous robot*. https://arxiv.org/pdf/1806.03581

[9] HaiderAbasi. (n.d.). GitHub - HaiderAbasi/ROS2-Path-Planning-and-Maze-Solving: Developing a maze solving robot in ROS2 that leverages information from a drone or Satellite’s camera using OpenCV algorithms to find its path to the goal and solve the maze. :). GitHub. https://github.com/HaiderAbasi/ROS2-Path-Planning-and-Maze-Solving

3.1 BOGAT

Maze-mapping algorithms: A primary objective that the robot needs to fulfill is the ability to navigate a maze while looking for the heat signals. This requires the implementation of an appropriate maze-mapping algorithm. Our team discussed 2 main algorithms that we could use to map the maze:

Breadth-First Search (BFS)

Attached below is a sketch explaining briefly how the BFS algorithm works:

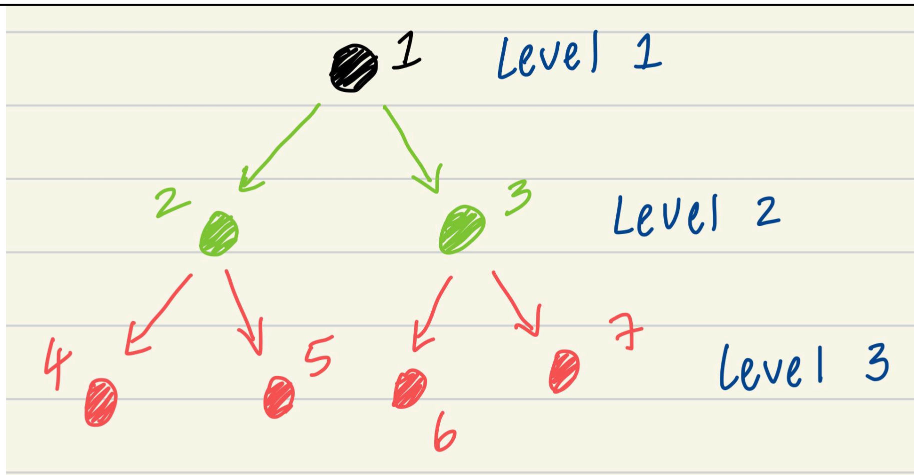

Figure 3.1.1: A simple diagram illustrating the working of the BFS algorithm

As inferred from the above diagram, the BFS algorithm explores a given graph (in this case, the maze) by dividing the constituent nodes into levels (based on the distance from the starting point). It then explores every node at a given level before moving on to the nodes at the next level[10]. In terms more specific to the objective's maze, BFS explores every cell adjacent to the starting cell, before moving on to the cells adjacent to those and so on.

BFS has the advantage of always finding the solution (if it exists), and always determining the shortest path to it. However, its time complexity ($O(V+E)$) and space complexity ($O(V)$), where V and E are the numbers of vertices and edges respectively, can lead the algorithm to be slow or take up a considerable amount of memory space when there are a large number of vertices and edges in the input graph.

A* Search Algorithm

The A* algorithm is a popular search algorithm that uses a cost function to streamline its search in such a way to prevent exploring potentially unnecessary parts of a given graph. This cost function, say $f(n)$, is a combination of two other functions. One is the distances traversed from

[10] "Breadth First Search Tutorials & Notes: Algorithms." *HackerEarth*, www.hackerearth.com/practice/algorithms/graphs/breadth-first-search/tutorial/. Accessed 4 Feb. 2025.

the start to the current location, say *g(n)*.[11] The other is a heuristic function, *h(n),* that provides an estimate of the "cost" from the current location to the end goal. For the mission, the heuristic function could be designed to factor in the data from the IR sensors, guiding the robot towards the heat signals.

Due to the use of a heuristic function, A* is often more efficient than algorithms such as BFS, because it skips unnecessary parts of the graph. However, its space and time complexities are dependent on the heuristic and cost functions, and in the worst case, they can be exponential. Additionally, and more critically, the effectiveness of A* is heavily reliant on the heuristic function, and a bad heuristic can lead to the algorithm malfunctioning severely.

Detecting Casualties: Likely to be the most technically heavy subsystem of our project, finding the most **cost-effective** solution to casualty detection will be an important decision. We discussed 2 main considerations for this subsystem:

OTS Thermal Camera, MLX90640	**IR Array built from Scratch, FC-51**
\$30-\$80[12]	<\$10[13]
Less setup required	Requires building extra circuitry
Well documented example projects	Lack of information available on the web, needs additional experimentation

Given that we likely will not spend as much money on other components of our project, we find that investing in a well researched and documented OTS product should be a worthy investment that will also save us time in implementation. It is worth noting the **many well documented projects involving the MLX90640**[14] specifically (also see section 2.2 Thermal Sensor), which would accelerate the development of our casualty detection subsystem. Since we **likely can afford only one** thermal sensor of limited FOV, we also discussed how to integrate only one Thermal Camera in our mapping algorithm in order to locate all casualties. We propose that phase 1. Search will be split into 2 parts. Firstly, the TurtleBot will use LiDAR only to map the whole course. Afterwards, the bot will make a second run through the course scanning using both LiDAR and IR in order to pinpoint casualty locations.

Ping-pong launcher: Another key component to the success of the mission, it is crucial that the ping-pong launcher is **compact and lightweight** enough to be mounted on the TurtleBot, as well as **robust** enough so as to enable consistent synchronised firing of the ping-pong balls. The

---

[11] Kumar, Rajesh. "The A* Algorithm: A Complete Guide." *DataCamp*, DataCamp, 7 Nov. 2024, www.datacamp.com/tutorial/a-star-algorithm.
[12] *Far Infrared Thermal Sensor Array (32x24 RES)*. (n.d.). Melexis. https://www.melexis.com/en/product/MLX90640/Far-Infrared-Thermal-Sensor-Array
[13] https://www.aliexpress.com/item/1005006244747703.html
[14] Admin, & Admin. (2024, May 4). DIY Thermal Imaging Camera with MLX90640 & Raspberry Pi. How to Electronics. https://how2electronics.com/diy-thermal-imaging-camera-with-mlx90640-raspberry-pi/

following are some solutions we have considered:

Spring-loaded launcher	Compressed air cannon	Dual flywheel launcher
+ Simpler system compared to the other proposed mechanisms	+ Provides considerable power with relatively small amount of compressed air	+ Powerful, can launch to a considerable height
+ Reliable and provides consistent performance, requires less power	+ No electrical components involved in the launching mechanism, so less prone to hardware failure	+ No priming or reloading required between individual launches
- Less powerful compared to the other proposed systems	- Requires the mounting and integration of pneumatic systems in addition to mechanical and electrical ones	- Energy-intensive, so will draw considerable power from the battery. Performance can become inconsistent if the battery is drained.
- Requires priming the spring between each launch		- Requires precise parts to function as intended, 3D-printed parts may cause malfunctions.

4.0 Preliminary Design

Outcome	Technologies/Methods Required
Integration of heat sensors with robot	1. Sensor fusion with LiDAR point cloud 2. Integration with cartographer to produce a map with walls coloured using thermal data
Ability to map environment autonomously	1. Frontier-based exploration and Monte-Carlo Localisation[15] 2. LiDAR point cloud, IMU data and Odometry
Ping-pong ball launcher	1. Flywheel system (taken from OTS ping-pong ball launchers) 2. Loading mechanism (spring/gravity fed)

[15] Nakamura (n.d.). https://kainakamura.com/project/slam-robot

References

[1] Pire, Taihú & Mujica, Martin & Civera, Javier & Kofman, Ernesto. (2019). The Rosario dataset: Multisensor data for localization and mapping in agricultural environments. The International Journal of Robotics Research. 38. 027836491984143. 10.1177/0278364919841437.

[2] Hall, Tutwiler, & Rimland. (n.d.). *Hard sensor fusion: LiDAR and IR*. Center for Multisource Information Fusion. https://www.eng.buffalo.edu/~nagi/MURI/MURI/Year_2_files/PDF%20Presentations/6%20-%20David%20-%20Hard%20Data%20Fusion%201%20LIDAR%20and%20IR.pdf

[3] Infrared detector arrays low cost thermal imaging. (2022). In https://www.boselec.com. BostonElectronics. https://www.boselec.com/wp-content/uploads/Linear/Heimann/HeimannLiterature/Heimann-Arrays-01-11-24.pdf

[4] Benchoff, B. (2012, November 12). Building a thermal imaging sensor from scratch. Hackaday. https://hackaday.com/2012/11/12/building-a-thermal-imaging-sensor-from-scratch/

[5] Muthukumar, K. A., Bouazizi, M., & Ohtsuki, T. (2022). An Infrared Array Sensor-Based Approach for Activity Detection, Combining Low-Cost Technology with Advanced Deep Learning Techniques. Sensors (Basel, Switzerland), 22(10), 3898. https://doi.org/10.3390/s22103898

[6] Engineering Dads. (2023, June 6). *Powerful Ping-Pong launcher* [Video]. YouTube. https://www.youtube.com/watch?v=MX3i9wQe3_g

[7] Butler, N. (2022, January 7). Automatic Ping-Pong Launcher - Nollaig Butler - Medium. *Medium*. https://nollaigengineering.medium.com/automatic-ping-pong-launcher-c77dc140d33b

[8] Topiwala, Inani, & Kathpal. (n.d.). *Frontier based exploration for autonomous robot*. https://arxiv.org/pdf/1806.03581

[9] HaiderAbasi. (n.d.). *GitHub - HaiderAbasi/ROS2-Path-Planning-and-Maze-Solving: Developing a maze solving robot in ROS2 that leverages information from a drone or Satellite's camera using OpenCV algorithms to find its path to the goal and solve the maze. :)*. GitHub. https://github.com/HaiderAbasi/ROS2-Path-Planning-and-Maze-Solving

[10] "Breadth First Search Tutorials & Notes: Algorithms." *HackerEarth*, www.hackerearth.com/practice/algorithms/graphs/breadth-first-search/tutorial/. Accessed 4 Feb. 2025.

[11] Kumar, Rajesh. "The A* Algorithm: A Complete Guide." *DataCamp*, DataCamp, 7 Nov. 2024, www.datacamp.com/tutorial/a-star-algorithm.

[12] *Far Infrared Thermal Sensor Array (32x24 RES)*. (n.d.). Melexis. https://www.melexis.com/en/product/MLX90640/Far-Infrared-Thermal-Sensor-Array

[13] https://www.aliexpress.com/item/1005006244747703.html

[14] Admin, & Admin. (2024, May 4). DIY Thermal Imaging Camera with MLX90640 & Raspberry Pi. How to Electronics. https://how2electronics.com/diy-thermal-imaging-camera-with-mlx90640-raspberry-pi/

[15] Nakamura (n.d.). https://kainakamura.com/project/slam-robot

Appendix A: Glossary of Acronyms

ConOps	Concept of Operations
BOGAT	Bunch of Guys Around Table
SLAM	Simultaneous Localisation and Mapping
LiDAR	Light Detection and Ranging
OTS	Off the Shelf
OpenCR	Open Source Control Module for ROS
ROS2	Robot Operating System (Version 2)
S&R	Search and Rescue
SOP	Standard Operating Procedure
BFS	Breadth-First Search

Appendix B: Glossary of Terms

Rasp Pi	Raspberry Pi is a series of small single-board computers (SBCs) developed in the United Kingdom.
IMU	An inertial measurement unit (IMU) is an electronic device that measures and reports a body's specific force, angular rate, and sometimes the orientation of the body, using a combination of accelerometers, gyroscopes, and sometimes magnetometers.
Synchronised	Operate at the same time or rate
Sensor Fusion	Sensor fusion is the process of integrating data from multiple sensors to obtain more accurate, reliable, and comprehensive information about an environment or system.
Odometry	Odometry is the use of data from motion sensors to estimate change in position over time.
Solenoid Actuator	Solenoid actuators are solenoid Equipment used to convert electrical signals or electric currents into mechanical linear motion.
A* Algorithm	A search algorithm used to find the shortest path between an initial and a final point.
Thermopile	A thermopile is an electronic device that converts thermal energy into electrical energy