Active Simultaneous Localization and Mapping in Unstructured Environment with a Quadrotor

Siming He[1, 2], Mentor: Prof. Pratik Chaudhari[2]
[1]The Wharton School of The University of Pennsylvania, [2]University of Pennsylvania School of Engineering and Applied Science
Sponsored by the Wharton Summer Program of Undergraduate Research

Introduction

- An agent performing **Simultaneous Localization and Mapping (SLAM) constructs a map of the environment while estimating its location at the same time.** They can be formulated as a smoothing problem of belief state (finding the best estimation of agent location and landmarks in the environment):

$$X^*, \Theta^* = \text{argmax}_{X,\Theta}\, Bel(X,\Theta) = \text{argmax}_{X,\Theta} P(X,\Theta \mid Z, U)$$

$$= \text{argmax}_{X,\Theta} P(x_0) \cdot \prod_{i=1}^{T} P(x_i \mid x_{i-1}, u_{i-1})$$

$$\cdot \prod_{i=0}^{n} P(\theta_i) \prod_{j=0}^{T} P(z_j \mid x_j, \theta_j)\ \mathbf{1}_{z_j}(\theta_i)$$

where X are states of the robot over time, Θ are locations of each landmarks, Z are robot's observations of landmarks, and U are robot odometry data. The joint belief can be represented as a graphical model and can be optimized through graph-based optimization approaches.

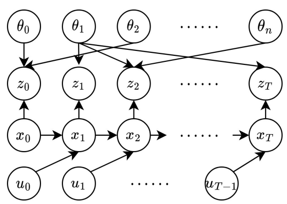

Figure 1: Graphical Model of Joint Belief of States

- **These algorithms do not typically address how an agent should explore an unknown environment to build a map efficiently.** This ability for active exploration is important for autonomous robots to work in unknown, unstructured environments such as forests or caves.

- **This paper proposes an active SLAM system that allows an agent to explore its surroundings, using visual-inertial data from an RGBD camera.** We formalize this problem as taking actions that maximize the amount of information obtained from the scene.

References

[1] J. A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, and J. A. Castellanos, "A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers," arXiv:2207.00254 [cs.RO].
[2] A. Rosinol, M. Abate, Y. Chang and L. Carlone, "Kimera: an OpenSource Library for Real-Time Metric-Semantic Localization and Mapping," in IEEE Intl. Conf. on Robotics and Automation (ICRA), 2020.
[3] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3D Euclidean Signed Distance Fields for OnBoard MAV Planning," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017.
[4] F. Dellaert, "Factor Graphs and GTSAM: A Hands-on Introduction," Technical Report number GT-RIM-CPR-2012-002.
[5] L. M. Schmid, M. Pantic, R. Khanna, L. Ott, R. Y. Siegwart, and J. I. Nieto, "An Efficient Sampling-Based Method for Online Informative Path Planning in Unknown Environments," IEEE Robotics and Automation Letters, 2020, vol. 5, pp. 1500-1507.
[6] R. Border, J. D. Gammell, P. Newman, "Surface Edge Explorer (SEE): Planning Next Best Views Directly from 3D Observations," IEEE Press, 10.1109/ICRA.2018.8461098.
[7] J. Amanatides and A. Woo, "A Fast Voxel Traversal Algorithm for Ray Tracing," In Eurographics '87, 1987, pp. 3-10. z
[8] I. Brand, J. Roy, A. Ray, J. Oberlin and S. Oberlix, "PiDrone: An Autonomous Educational Drone Using Raspberry Pi and Python," 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 1-7, doi: 10.1109/IROS.2018.8593943.

System Design

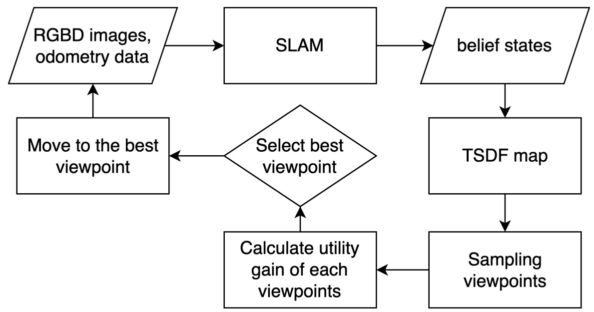

Figure 2: Overall System Design

- **Truncated Signed Distance Field (TSDF) Map** is used to represent the environment. It discretizes the environment into voxels (3D cubes that is comparable to pixels of images). Each voxel contains its distance to the closest surface and how certain is this distance estimation. This representation is chosen since it contains information that measures how well each voxel is estimated.

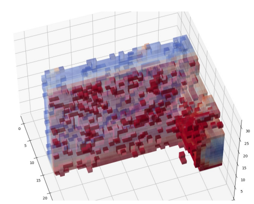
Figure 3: TSDF Visualization

- **Sampling viewpoints**: States of the robot is sampled uniformly for multiple times. For each of the sampled viewpoint, we want to measure the amount of utility gain after moving to this viewpoint.

- **Utility gain**: Utility gain is designed with two considerations. Firstly, utility gain should be higher if moving to the viewpoint reduces more uncertainty of our mapping of the environment. Secondly, utility gain should be lower if the robot need to move longer to reach the viewpoint. Hence, we defined the following utility function:

$$U(V) = \frac{U_{surface}(V) + U_{new}(V) + U_{frontier}(V)}{amoun\ of\ translationa\ and\ rotation}$$

where $U_{surface}(V)$, $U_{new}(V)$, and $U_{frontier}(V)$ are the amount of information gain of observing surface voxels, new voxels, and frontier voxels.

Experiments

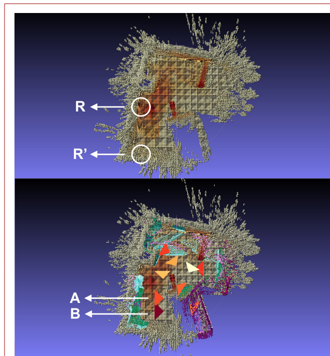

Figure 4: TSDF, Sampled Viewpoints, and Utility Gain

In the TSDF, redder voxels are voxels with more certainty. The triangles are sampled viewpoints. Triangles with redder color are viewpoints with more information gain.

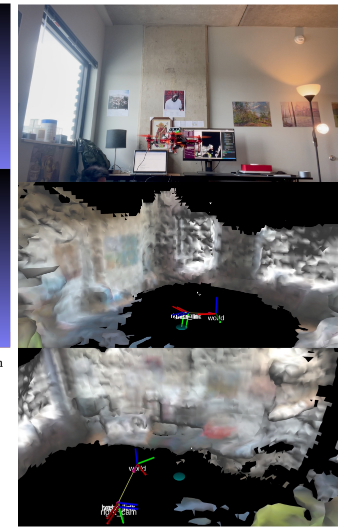
Figure 5: Mapping Result by Intel RealSense camera mounted on a custom-built quadrotor

Conclusion

We implemented an active SLAM system and test it on a drone. The experiment results show that the system can explore the environment and construct a map. However, the map is not detailed, and improvements can be made in the future. The following are some assumptions in our implementation that need to be changed for improvements:

1. The assumption that the agent has perfect knowledge of its pose is unrealistic and localization uncertainty would cause inaccurate mapping results.
2. Unexplored regions have a fixed information gain in this system. It is not a good approximation since it doesn't show which unexplored region to go to especially in larger state space. Prediction of unseen regions or information gain of unseen regions would be needed.