S3M: Semantic Segmentation Sparse Mapping for UAVs with RGB-D Camera

Thanh Nguyen Canh[1,3], Van-Truong Nguyen[2], *Xiem HoangVan[1], Armagan Elibol[3], Nak Young Chong[3]
[1]University of Engineering and Technology, Vietnam National University
Hanoi, Vietnam ({canhthanh, xiemhoang}@vnu.edu.vn)
[2]Department of Mechatronics Engineering, Hanoi University of Industry
Hanoi 159999, Vietnam (nguyenvantruong@haui.edu.vn)
[3]School of Information Science, Japan Advanced Institute of Science and Technology
Nomi, Ishikawa 923-1292, Japan ({aelibol, nakyoung}@jaist.ac.jp)

***Abstract*—Unmanned Aerial Vehicles (UAVs) hold immense potential for critical applications, such as search and rescue operations, where accurate perception of indoor environments is paramount. However, the concurrent amalgamation of localization, 3D reconstruction, and semantic segmentation presents a notable hurdle, especially in the context of UAVs equipped with constrained power and computational resources. This paper presents a novel approach to address challenges in semantic information extraction and utilization within UAV operations. Our system integrates state-of-the-art visual SLAM to estimate a comprehensive 6-DoF pose and advanced object segmentation methods at the back end. To improve the computational and storage efficiency of the framework, we adopt a streamlined voxel-based 3D map representation - OctoMap to build a working system. Furthermore, the fusion algorithm is incorporated to obtain the semantic information of each frame from the front-end SLAM task, and the corresponding point. By leveraging semantic information, our framework enhances the UAV's ability to perceive and navigate through indoor spaces, addressing challenges in pose estimation accuracy and uncertainty reduction. Through Gazebo simulations, we validate the efficacy of our proposed system and successfully embed our approach into a Jetson Xavier AGX unit for real-world applications.**

***Index Terms*—Semantic Mapping, S3M, UAVs, ROS, SLAM.**

I. Introduction

Over the past decades, UAVs have significantly impacted geoinformation acquisition in areas like firefight rescue, inspection, and agriculture. Understanding the environment is crucial for advancing autonomous capabilities, including the UAV's self-location awareness and semantic 3D map creation. Semantic mapping, which combines environmental geometry estimation with semantic labeling, goes beyond traditional geometric mapping, enhancing UAVs' situational understanding and interactions. For instance, in the rescue mission, a robot relying solely on a traditional SLAM-generated map encounters challenges in performing complex tasks, such as: "maneuvering around the desk to locate a victim beside the bed". Besides that, this task is still challenging due to: (1) the inaccuracy of GPS indoors, (2) the cluttered environments, (3) the real-time process demands, and (4) the complexity of semantic maps.

To address the aforementioned challenges, the adoption of Simultaneous Localization and Mapping (SLAM) techniques, especially Visual SLAM [1]–[3] emerges as a compelling solution in the realm of drone applications, which is characterized by its compact design and cost-effectiveness, offering a wealth of information to comprehend the field of view. Various algorithms have been developed for this task such as KinectFusion [4], RGBD_SLAM [5], ORB-SLAM [6], PTAM [7], DSO_SLAM [3], LSD_SLAM [2], and SVO_SLAM [8]. Their effectiveness however varies depending on each scenario and the environments in which the robot operates such as localization, mapping, and real-time processes. Additionally, the research on the fusion of the semantic segmentation CNNs with the visual SLAM has been investigated with some notable works, including Semantic Fusion [9], Mask Fusion [10], SCFusion [11], Co-Fusion [12], and DS-SLAM [13]. Despite these advancements, achieving semantic reconstruction for UAVs remains challenging. Hence, our research aims to establish mapping with semantic data, vital for enabling UAVs to perform advanced autonomous tasks. In this paper, we proposed an efficient Semantic Segmentation Sparse Mapping (S3M) SLAM system for incrementally constructing an object-level map using a localized RGB-D camera. The proposed system is organized into two main components: an RGB-D SLAM framework based on propagation utilizing Visual Odometry (VO) estimation and object instance segmentation-based semantic sparse map creation. To summarize, the main contributions of this work can be summarized as follows:

- A S3M SLAM system that has faster fully 6-DoF pose tracking and the capability to construct a semantic sparse map based on object segmentation information.
- A semantic fusion strategy based on geometric and semantic descriptions to incrementally update objects.
- An efficient representation and storage method using OctoMap of the front-end system, a memory-efficient alternative to point cloud data.
- The demonstrated capability of constructing semantically sparse maps in real-time on a compact, computation-limited platform via experiments on the

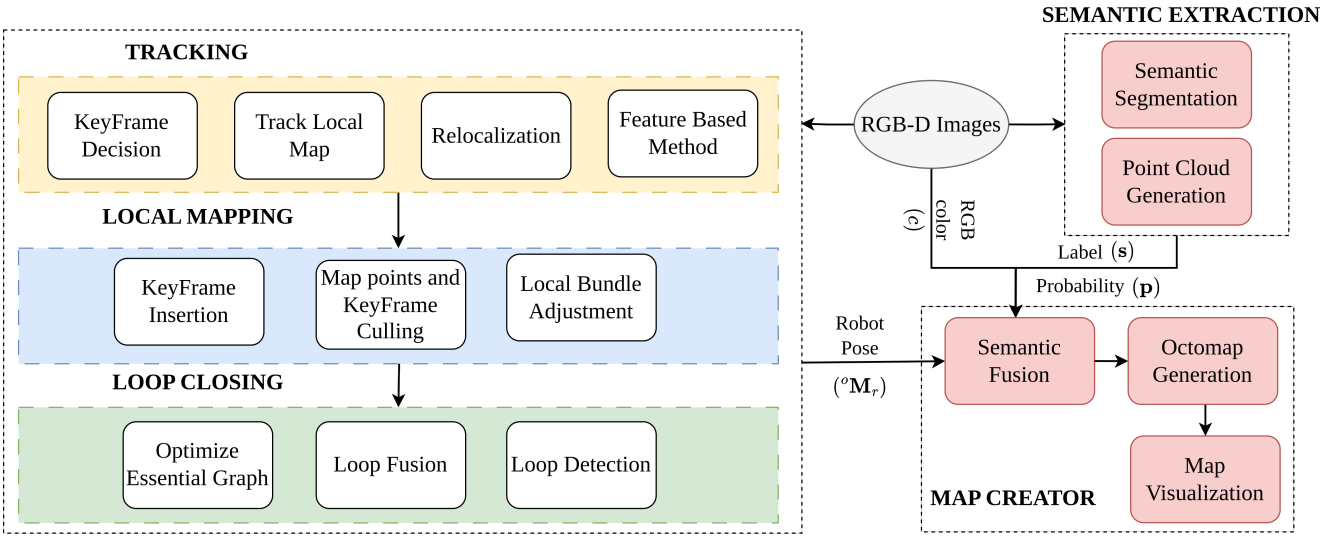


Fig. 1: **Proposed S3M SLAM Architecture**: The system is composed of three units: a full 6 DoF pose estimation of the drone through ORB-SLAM3 (Tracking part - yellow, Local Mapping part - blue, Loop Closing part - green), a 3D semantic segmentation branch, and a semantic fusion scheme

Jetson Xavier AGX embedded computer.

The remainder of this paper is organized as follows: Sec II describing the proposed system based on RGB-D SLAM and object segmentation. The experiments conducted and results analysis are presented in Sec III. Finally, Sec IV draws a conclusion with future works.

II. Methodology

The proposed S3M SLAM pipeline is illustrated in Fig. 1, which takes RGB-D sequences as input and progressively constructs a volumetric map enhanced with object instances. To achieve this, the RGB-D images undergo initial processing via a UAV pose tracking framework (Section II-A). Subsequently, an object instance segmentation method is applied to detect and extract semantic 3D objects from individual frames (Section II-B). These identified objects are then integrated into a volumetric mapping framework to generate a dense map at the object level (Section II-C). To enhance map quality, Octomap is utilized for noise removal and voxel grid downsampling to save space, with optimization for improved visual representation (Section II-D). The system is implemented within a ROS framework, the widely used platform in the robotics community, the system leverages open-source tools, libraries, and conversions to simplify the development of intricate and robust robotics behaviors.

A. Pose estimation

The accurate estimation of the UAV's pose is a critical step in our S3M SLAM pipeline. We employ the ORB-SLAM3 algorithm [1] for robust and real-time camera pose estimation from RGB-D images. ORB-SLAM3 utilizes a monocular camera model and extends it to support stereo and RGB-D setups, making it well-suited for our UAV's sensor configuration. It encompasses three parallel threads: (1) Tracking, (2) Local Mapping, and (3) Loop Closing [14]. The pose estimation problem involves determining the position (x, y, z) and orientation (ϕ, θ, ψ) of the UAV in a global coordinate system. ORB-SLAM3 solves this problem by tracking a set of distinctive features in consecutive frames and establishing the correspondences between them. The estimated pose is obtained by minimizing the reprojection error between the observed feature locations and their predicted locations in the camera frame. Mathematically, given a set of N observed 2D feature points $\mathbf{p_i}$ in the current RGB-D frame and their corresponding 3D points $\mathbf{P_i}$ in the world frame, the estimated camera pose

$$^{\bar{o}}\mathbf{M}_c = \begin{bmatrix} ^{\bar{o}}\mathbf{R}_c & ^{\bar{o}}\mathbf{T}_c \\ 0_{1\times3} & 1 \end{bmatrix} = \begin{bmatrix} ^{\bar{o}}r_{c_{00}} & ^{\bar{o}}r_{c_{01}} & ^{\bar{o}}r_{c_{02}} & ^{\bar{o}}t_{c_{00}} \\ ^{\bar{o}}r_{c_{10}} & ^{\bar{o}}r_{c_{11}} & ^{\bar{o}}r_{c_{12}} & ^{\bar{o}}t_{c_{10}} \\ ^{\bar{o}}r_{c_{22}} & ^{\bar{o}}r_{c_{22}} & ^{\bar{o}}r_{c_{22}} & ^{\bar{o}}t_{c_{20}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

can be obtained by solving the optimization problem:

$$^{\bar{o}}\mathbf{M}_c = \underset{^{\bar{o}}\mathbf{M}_c}{\operatorname{argmin}} \sum_{i=1}^{N} ||\mathbf{p_i} - \pi(^{\bar{o}}\mathbf{M}_c \times \mathbf{P_i})||^2 \tag{1}$$

where $^{\bar{o}}\mathbf{R}_c, ^{\bar{o}}\mathbf{T}_c, ^{\bar{o}}\mathbf{M}_c$ are the rotation matrix, the translation matrix, and the transformation matrix between the world's coordinate frame ($\bar{\mathbf{O}}_{xyz}$) and the camera's coordinate frame in ORB-SLAM3, respectively. $\pi(\cdot)$ is the projection function from 3D to 2D points, and $||\cdot||$ represents the Euclidean distance.

Since camera odometry obtained from Eq. 1 and robot odometry have distinct world coordinates, we performed a calibration process. Denote $^{o}\mathbf{M}_r, ^{o}\mathbf{M}_c$ respectively as the transformation matrix representing the robot pose and camera pose relative to the robot's world frame ($\mathbf{O}_{xyz}$). The transformation $^{o}\mathbf{M}_c$ is then computed as:

$$^{o}\mathbf{M}_c = \begin{bmatrix} ^{\bar{o}}r_{c_{00}} & ^{\bar{o}}r_{c_{02}} & ^{\bar{o}}r_{c_{21}} & ^{\bar{o}}t_{c_{20}} \\ ^{\bar{o}}r_{c_{20}} & ^{\bar{o}}r_{c_{11}} & ^{\bar{o}}r_{c_{01}} & ^{\bar{o}}t_{c_{00}} \\ ^{\bar{o}}r_{c_{12}} & ^{\bar{o}}r_{c_{10}} & ^{\bar{o}}r_{c_{22}} & ^{\bar{o}}t_{c_{10}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

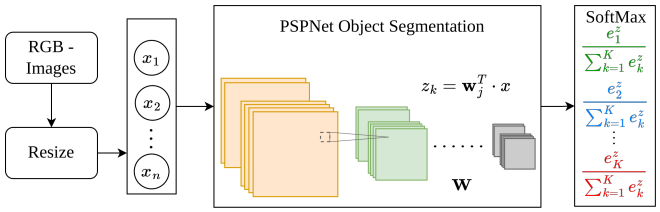


Fig. 2: Structure of semantic segmentation model

Let ${}^c\mathbf{M}_r$ represent the transformation matrix between the camera and the UAV's frame. The UAV's pose is given by:

$$ {}^o\mathbf{M}_r = {}^o\mathbf{M}_c \times {}^c\mathbf{M}_r \tag{3} $$

B. Semantic segmentation

In our methodology, semantic segmentation plays a vital role in extracting meaningful object instances from RGB-D images. This process is illustrated in Fig. 2. Firstly, a color image is resized to the input size of the network. For our semantic segmentation network, we adopted the Pyramid Scene Parsing Network (PSPNet) [15] due to its proven effectiveness in generating accurate pixel-level semantic labels. PSPNet employs a multi-step process involving feature extraction with ResNet, pyramid pooling, convolutions on pooled feature maps, fusion of feature maps, and final convolutions to generate a class score map. This map assigns probabilities p_i to pixels, enabling precise identification of object instances and their semantic labels. To finalize the process, a softmax activation is applied to the class score map, producing a probability distribution. Each pixel, along with its probability, is then selected and fused with the point cloud's pose.

C. Semanic fusion

To achieve comprehensive scene understanding, it becomes imperative to integrate semantic labels across multiple views with translations. In addition to position and RGB data, semantics information is encoded within a point cloud. We represent this information using the vector $\mathbf{Q} = \begin{bmatrix}\mathbf{t} & c & \mathbf{s} & \mathbf{p}\end{bmatrix}^T$, where $\mathbf{t} \in \mathbb{R}^3$ and $c \in \mathbb{R}^1$ denote the 3D position and RGB color of a point cloud. This information is derived from RGB images and depth images. Furthermore, $\mathbf{s} \in \mathbb{R}^k$ and $\mathbf{p} \in \mathbb{R}^k$ symbolize k semantic colors with the highest probability and their respective confidence scores associated with a point cloud. In each observation O_i, we calculate the probability for each semantic color within a given semantic set. Subsequently, the point featuring the highest probability is selected as the final decision. This approach ensures that semantic information is effectively incorporated into the point cloud, enabling a richer and more nuanced understanding of the scene across multiple viewpoints and translations. Algorithm 1 outlines the semantic fusion process.

D. Semantic map creation

In our approach, each keyframe retains the 3D point clouds, while the segmented 3D point clouds are preserved in alignment with the respective objects. However, point cloud-based maps often demand substantial storage space, rendering them unsuitable for modeling large-scale environments with limited memory and lack of structures to efficiently store each point, hindering search operations. To address these challenges, we adopted OctoMap [16], a probabilistic 3D mapping framework based on octrees. OctoMap presents a more efficient solution for storing occupancy status compared to point cloud maps, significantly reducing storage demands. Leveraging octrees, OctoMap divides spaces into small cubes, further subdivided into eight smaller cubes. Leaf nodes represent the smallest voxels, and a probabilistic model tackles issues like noise and range measurement errors by assigning probabilities to occupied or free states. This makes OctoMap an ideal choice for creating maps in our system, as it overcomes the limitations posed by traditional point cloud-based approaches. When a new 3D point is inserted, the log odds value for the voxel i at time t ($L(i|Z_{1:t-1})$ is computed using the log odds value accumulated up to time $t-1$ ($L(i|Z_{1:t-1})$:

$$ L(i|Z_{1:t}) = L(i|Z_{1:t-1}) + L(i|Z_t) \tag{4} $$

where,

$$ L(i) = \log\left[\frac{p(i)}{1-p(i)}\right] \tag{5} $$

Here, Z_t represents the observed for a voxel at time t. $p(i)$ is the probability that the voxel i contains an object or obstacle.

Algorithm 1 Semantic Segmentaion Fusion Approach

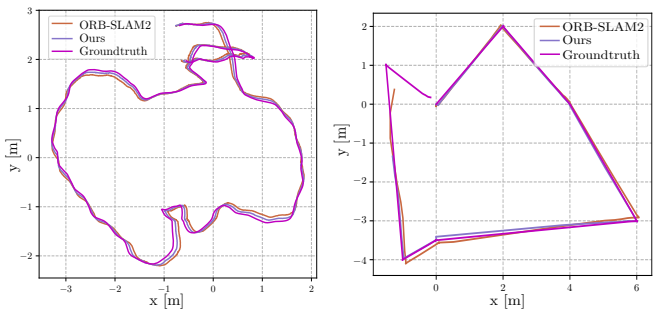
Input: $\mathbf{Q}_1$ ▷ Point cloud in Oservation 1
$\mathbf{Q}_2$ ▷ Point cloud in Oservation 2
α ▷ Trade of coefficient
Output: $\mathbf{Q}_{fusion}$
if $\mathbf{Q}_1.\mathbf{s} = \mathbf{Q}_2.\mathbf{s}$ **then**
  $\mathbf{Q}_{fusion} = \mathbf{Q}_1$
else
  ▷ Probability for other unknown colors
  $\bar{p_1} = 1 - \sum(\mathbf{Q}_1.\mathbf{p})$
  $\bar{p_2} = 1 - \sum(\mathbf{Q}_2.\mathbf{p})$
  ▷ Synchronize data from $\mathbf{Q}_1$ to $\mathbf{Q}_2$
  for each $label$ in $\mathbf{Q}_1.\mathbf{s}$ not in $\mathbf{Q}_2.\mathbf{s}$ **do**
    $(\mathbf{Q}_2.\mathbf{s}).push\_back(label)$
    $(\mathbf{Q}_2.\mathbf{p}).push\_back(\alpha \times \bar{p_2})$
    $\bar{p_2} = 1 - \sum(\mathbf{Q}_2.\mathbf{p})$
  end for
  ▷ Synchronize data from $\mathbf{Q}_2$ to $\mathbf{Q}_1$
  for each $label$ in $\mathbf{Q}_2.\mathbf{s}$ not in $\mathbf{Q}_1.\mathbf{s}$ **do**
    $(\mathbf{Q}_1.\mathbf{s}).push\_back(label)$
    $(\mathbf{Q}_1.\mathbf{p}).push\_back(\alpha \times \bar{p_1})$
    $\bar{p_1} = 1 - \sum(\mathbf{Q}_1.\mathbf{p})$
  end for
  $\mathbf{Q}_{fusion} = \mathbf{Q}_1$
  ▷ Nomalize to probability distribution
  $\mathbf{Q}_{fusion}.\mathbf{p} = \big(\mathbf{Q}_1.\mathbf{p} \times \mathbf{Q}_2.\mathbf{p}\big) / \big(\sum(\mathbf{Q}_1.\mathbf{p} \times \mathbf{Q}_2.\mathbf{p})\big)$
end if

Fig. 3: UAV and gazebo environment simulation

III. Experimental Results

A. UAVs Simulation

The experimental evaluation of our proposed S3M SLAM system was conducted on the Hummingbird UAV platform equipped with a RealSense camera as shown in Fig. 3. The Hummingbird UAV [17] is characterized by its lightweight design, enabling agile flight maneuvers and precise navigation in dynamic and challenging environments, such as those encountered in search and rescue operations. It is equipped with state-of-the-art flight control algorithms, ensuring stable and controlled flight behavior during the experiments. The RealSense D455 camera complements the UAV's capabilities by providing RGB-D data, which is crucial for accurate pose estimation and semantic information extraction.

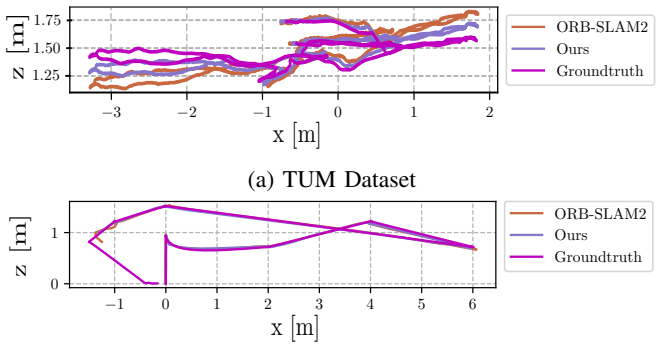


(a) TUM Dataset (b) Gazebo Dataset

Fig. 4: The comparison of trajectory for ORB-SLAM2, Our system and ground truth in X-Y axis

B. Pose estimation evaluation

To evaluate the pose estimation accuracy of the S3M system, we conducted evaluations on two distinct types of datasets: 1) TUM publicly available RGB-D data sequences [18], 2) the simulation dataset obtained from Gazebo. The precision of 6-DoF pose estimation was evaluated using the Root Mean Square Error (RMSE) of Absolute Trajectory Error (ATE) and Relative Pose Error (RPE). Figs. 4 and 5 show experiment results for pose trajectories, comparing ORB-SLAM2, our proposed system, and the ground truth across both datasets. Both ORB-SLAM2 and our system demonstrated accurate pose estimation and smooth movement within the environments, as depicted in Fig. 6. Finally, Fig. 7 presents the RMSE of ATE for all frames of both frameworks, reaffirming our system still ensures pose estimation performance.



(a) TUM Dataset

(b) Gazebo Dataset

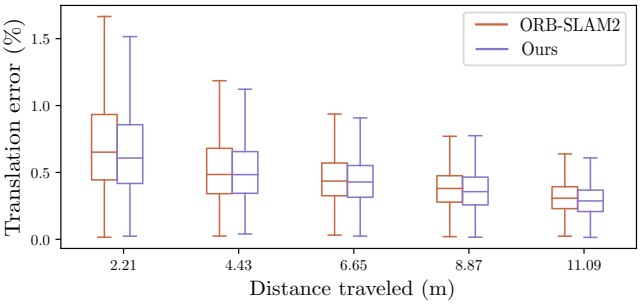
Fig. 5: The comparison of trajectory for ORB-SLAM2, our sytem and ground truth in X-Z axis

C. Training and evaluation on SUNRGBD dataset

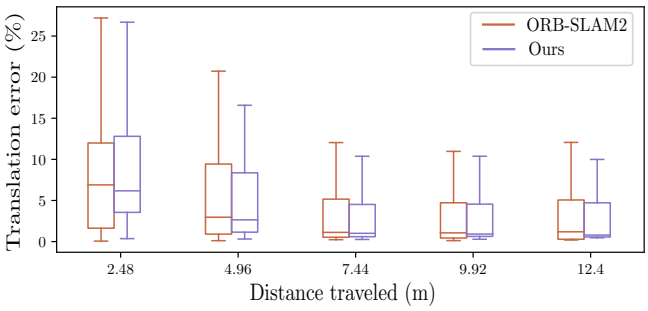
SUNRGBD [19] stands as a widely adopted benchmark for evaluating semantic scene understanding. This dataset encompasses a total of $10,335$ images distributed across 38 semantic classes, with $5,285$ images earmarked for training and $5,050$ for validation. We selected 6 types of networks for training PSPNet [15], ICNet [20], SegNet [21], UNet [22], FRRNs [23], and FCNs [24]. In the case of FCNs, two network variants, denoted as 8s, and 16s, were utilized. Similarly, FRRNs were employed in both A and B settings. Each model underwent training with a maximum of 100 epochs and batch size of 2 on an Nvidia T4, and the best-performing model was chosen. For optimization, standard stochastic gradient descent was utilized, featuring a weight decay of 1e-3, a momentum of 0.9, and a learning rate of 0.01. The experiment results for each model tested in SUNRGBD are depicted in Fig. 8. Among the models, PSPNet exhibited superior accuracy performance, prompting its selection as the segmentation model for integration into our system.

D. Semantic Map

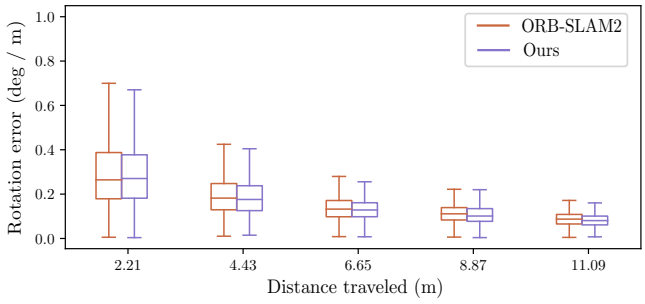
Fig. 9 shows the sequential processing stages and corresponding outcomes achieved by the proposed S3M system. As observed, our proposed method can integrate incoming semantic segmentation information (Fig. 9b) from input images (Fig. 9a) into the map volume, and the sparse map creation (Fig. 9d) process finalizes the OctoMap reconstruction by utilizing the point cloud containing semantic information. The implementation of the proposed system on the Jetson Xavier AGX platform, operating at 2Hz, where the object segmentation phase consumes 40ms per frame. The mapping results underscore the system's prowess in achieving real-time semantic mapping capabilities.



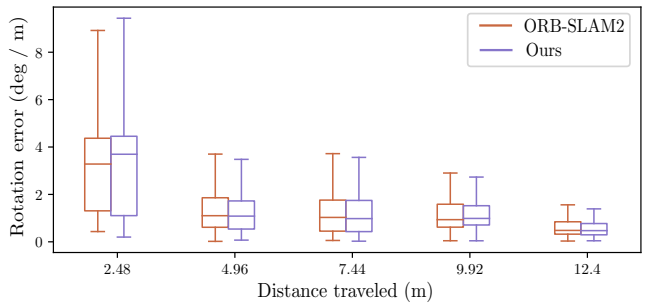
(a) Translation Error in TUM dataset



(b) Translation Error in Gazebo dataset



(c) Rotation Error in TUM dataset

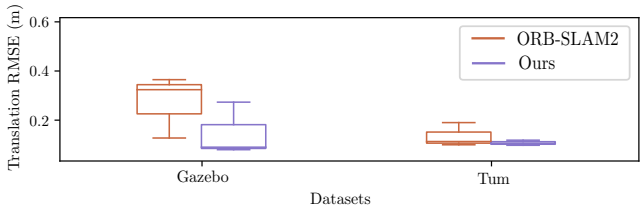


(d) Rotation Error in Gazebo dataset

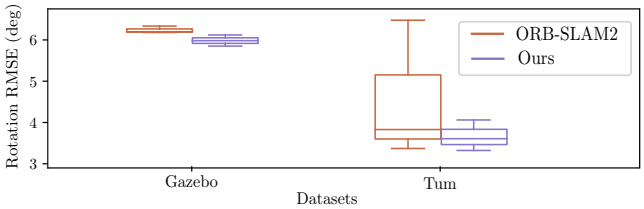
Fig. 6: Comparison of Relative Rose Error (RPE) between ORB-SLAM2 and Our system

IV. Conclusion

In this paper, we introduced a novel approach for Semantic Sparse Mapping (S3M) in Unmanned Aerial Vehicles (UAVs) based on RGB-D camera data. Our proposed S3M SLAM framework addresses the challenge of integrating semantic information into UAV mapping operations, enabling enhanced perception and understanding of the environment. By fusing



(a) Translation Error



(b) Rotation Error

Fig. 7: The comparison of ORB-SLAM2 and Our system based on the RMSE of ATE

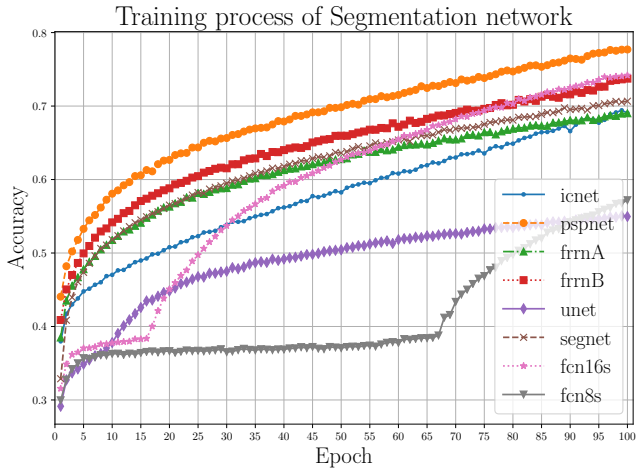


Fig. 8: Training Models Assessment

object instance segmentation with Octomap-based mapping, we achieve the creation of a semantic map that captures both spatial occupancy and object semantics. Future work could explore the integration of additional sensors to decrease cost and machine learning techniques to further enhance the UAV's perception capabilities.

Acknowledgment

This work was supported by the Asian Office of Aerospace Research and Development under Grant/Cooperative Agreement Award No. FA2386-22-1-4042.

References

[1] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[2] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[3] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[4] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison *et al.*, "Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera," in *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 2011, pp. 559–568.

[5] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the rgb-d slam system," in *2012 IEEE international conference on robotics and automation*. IEEE, 2012, pp. 1691–1696.

[6] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[7] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE, 2007, pp. 225–234.

[8] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 15–22.

[9] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfusion: Dense 3d semantic mapping with convolutional neural networks," in *2017 IEEE International Conference on Robotics and automation (ICRA)*. IEEE, 2017, pp. 4628–4635.

[10] M. Runz, M. Buffier, and L. Agapito, "Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects," in *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2018, pp. 10–20.

[11] S.-C. Wu, K. Tateno, N. Navab, and F. Tombari, "Scfusion: Real-time incremental scene reconstruction with semantic completion," in *2020 International Conference on 3D Vision (3DV)*. IEEE, 2020, pp. 801–810.

[12] M. Rünz and L. Agapito, "Co-fusion: Real-time segmentation, tracking and fusion of multiple objects," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 4471–4478.

[13] C. Yu, Z. Liu, X.-J. Liu, F. Xie, Y. Yang, Q. Wei, and Q. Fei, "Ds-slam: A semantic visual slam towards dynamic environments," in *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2018, pp. 1168–1174.

[14] L. Zhang, L. Wei, P. Shen, W. Wei, G. Zhu, and J. Song, "Semantic slam based on object detection and improved octomap," *IEEE Access*, vol. 6, pp. 75 545–75 559, 2018.

[15] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[16] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3d mapping framework based on octrees," *Autonomous robots*, vol. 34, pp. 189–206, 2013.

[17] R. Wall, "Hummingbird uav begins flight test program," *Aviation Week & Space Technology*, vol. 156, no. 5, pp. 37–37, 2002.

[18] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[19] S. Song, S. P. Lichtenberg, and J. Xiao, "Sun rgb-d: A rgb-d scene understanding benchmark suite," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 567–576.

[20] G. Li, Z. Liu, and H. Ling, "Icnet: Information conversion network for rgb-d based salient object detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 4873–4884, 2020.

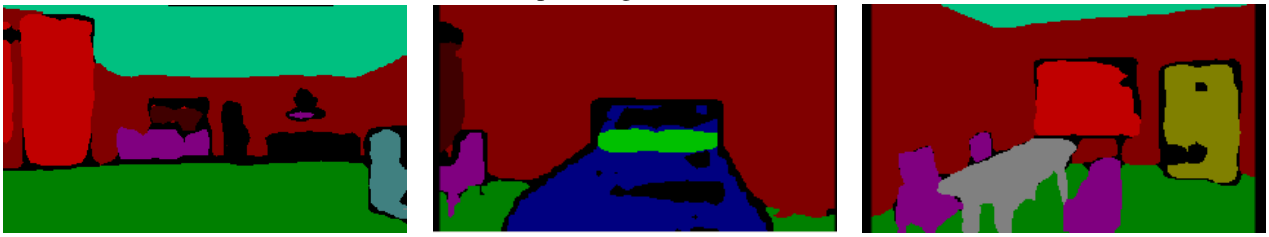
[21] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

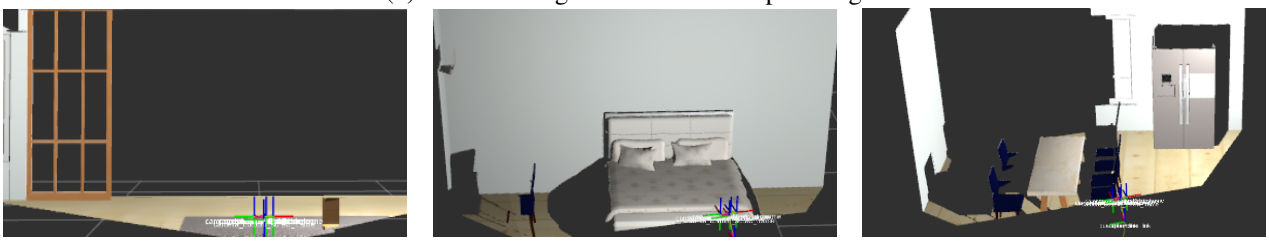
[23] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe, "Full-resolution residual networks for semantic segmentation in street scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4151–4160.

[24] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

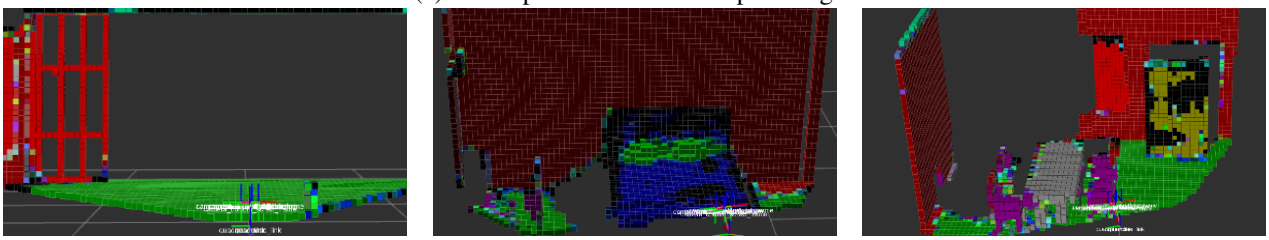
(a) Input image from camera



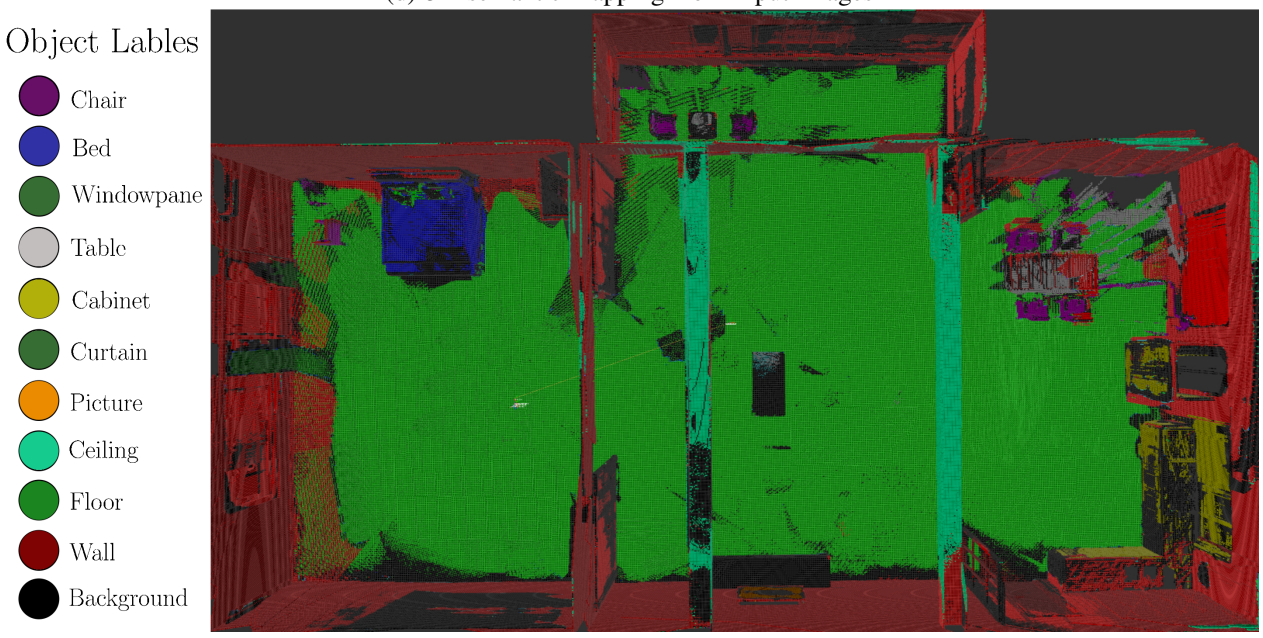
(b) Semantic segmentation from input images



(c) Color point cloud from input images



(d) 3D semantic mapping from input images



(e) Overall 3D semantic mapping

Fig. 9: 3D visual representation of the obtained semantic maps