Semantic pySLAM: Unifying semantic mapping approaches under the same framework

David Morilla-Cabello
Universidad de Zaragoza
Zaragoza, Spain
davidmc@unizar.es

Eduardo Montijano
Universidad de Zaragoza
Zaragoza, Spain
emonti@unizar.es

Fig. 1. Semantic segmentation results for the KITTI (left), Scannet (right-up), and Replica (right-down) datasets. Map points in 3D are colored with the associated semantic class following the Cityscapes (left), and NYU40 (right) color labeling. Semantic pySLAM enables seamless integration of multiple semantic mapping methods within the SLAM pipeline and datasets. Although the main pySLAM pipeline works with sparse points, the output of the volumetric integration module has been added for visualization.

I. Introduction

Simultaneous Localization and Mapping (SLAM) using images can be considered now a mature field. The efforts of the community to consolidate solutions to the SLAM problem have resulted in recent proposals such as pySLAM [1], or VSLAM-LAB [2]. In parallel, images have been widely used to extract semantic information from scenes. The advances made in this field have also prompted the appearance of SLAM solutions incorporating different variants of semantic features [3]–[7]. However, the research on semantic mapping still poses some problems.

First, semantic mapping presents a higher level of abstraction than geometric mapping. Additionally, the field is less mature, and semantic reasoning from neural networks evolves fast. Thus, some of the theoretical bases are yet to be established. How can we extract semantic knowledge from images? How should semantic features be interpreted? How should multiple observations be combined? How should the proposed methods be evaluated in a fair and rigorous manner? These questions still remain unanswered, partially due to the lack of standardized frameworks for the integration and evaluation of semantic mapping implementations.

This work has been funded by the DGA project T45_23R, MCIN/AEI/ERDF/European Union NextGenerationEU/PRTR project PID2021-125514NB-I00, ONR grant N62909-24-1-2081 and grant FPU20-06563.

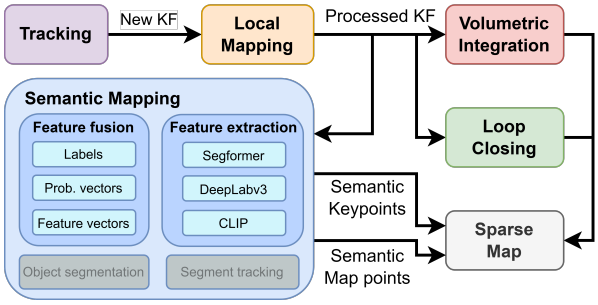

Fig. 2. Overview of the pySLAM architecture with the newly integrated semantic mapping module. The semantic mapping module implements common semantic mapping functionalities in a unified manner while integrating seamlessly with the SLAM pipeline. Components shown in gray are not yet implemented but are compatible with the framework.

The contribution of this work is a general semantic mapping framework designed to integrate within the SLAM pipeline, enabling a unified implementation and evaluation of semantic mapping methods. To this end, we analyze current research on semantic mapping, identify common approaches, and implement the necessary modules within pySLAM, allowing the use of existing datasets. We apply the proposed framework to incorporate typical semantic mapping methods and evaluate them on the ScanNet dataset [8]. With this article and the proposed system, we aim to foster discussion on how semantic knowledge should be incorporated into maps, ultimately reaching consensus that can drive progress in semantic scene understanding, similar to the evolution seen in SLAM.

II. Semantic pySLAM

To incorporate semantic mapping into the SLAM pipeline, we leverage the pySLAM framework. pySLAM is a Python implementation of a Visual SLAM pipeline with different variants of common components (i.e., feature extraction and matching, tracking, mapping, loop closing, and volumetric reconstruction) and integration with several datasets, enabling benchmarking and experimenting with visual SLAM techniques in a unified way. We include a semantic mapping module within this framework that runs in a parallel thread after the local mapping step, processing refined keyframes and adding semantic features to keypoints and sparse map points used by the geometric SLAM module.

To maintain generality, we design the semantic mapping module to support most existing approaches using semantic features. From the literature, we identify a taxonomy based on two main aspects. First, **3D entities**: semantic features can be assigned to each point of the map using *pixel-level* semantic segmentation [3]–[5], [9]–[12], or to object-level clusters through object detection and segmentation to create 3D *segments* [13]–[21]. Second, the **type of features**: semantic features output by neural networks can be interpreted in different ways, which also determines how they are fused. Interpreting them as point estimates yields categorical *labels*, which can be combined by simple counting [17]. When treated as *probability vectors*, fusion can be performed using Bayesian approaches [3], [4], [9], or averaging [18]. More recently, research on language-aligned feature spaces such as CLIP [22] has led to the use of latent *feature vectors*, which allow matching against natural language descriptions for open-vocabulary segmentation. Fusion, in this case, remains an open problem, with current strategies ranging from averaging to selecting the most representative vector, or learning-based solutions [5]–[7], [10], [11].

The proposed interface allows seamless integration of semantic mapping into the SLAM pipeline. We implement the *pixel-level* semantic mapping approach as a module supporting all feature types and their fusion methods (i.e., labels, probabilities, and feature vectors). Segment-based methods are left for future work. We include support for deep learning libraries, *torchvision* and *transformers*, including two off-the-shelf closed-set semantic segmentation models, DeepLabV3 [23] and Segformer [24]. We also integrate an open-vocabulary method based on CLIP using the *f3rm* library [11]. Functionality for visualizing semantic features and evaluating the semantic mapping task is also implemented. The proposed framework, along with all implemented functionality, is currently being integrated into pySLAM [1].

III. Results

To highlight the research opportunities enabled by the proposed framework, we evaluate and compare multiple semantic mapping methods integrated within the SLAM pipeline. This unified setup supports consistent evaluation and facilitates the development of new approaches, such as open-vocabulary mapping, across different datasets and representations.

First, we compare various semantic representations following a strategy similar to SemanticFusion [3]. We integrate the ScanNet dataset into pySLAM, which provides RGB-D sequences from diverse indoor scenes along with ground-truth camera trajectories and semantic labels based on the NYU40 class set. This experiment investigates whether fusing semantic information across keyframes improves accuracy compared to single-image inference. For each keyframe, we project the corresponding 3D map points and compare their semantic descriptors with the ground-truth labels of the frame. We compare the classification results against only using the semantic predictions obtained directly from the RGB image named *2D seg.*

[1]Check: https://github.com/luigifreda/pyslam/pull/177

TABLE I. Semantic mapping results on ScanNet using sequence *_00 for all scenes. Reported values are macro-averaged precision.

Model	Feature	Scene					
		568	578	435	100	488	Avg.
Segformer	2D seg.	0.538	0.546	0.344	0.381	0.531	0.473
	Label	0.549	0.585	0.349	0.407	0.601	0.498
	Prob. vector	0.535	0.579	0.375	0.400	0.673	0.512
CLIP	2D seg.	0.270	0.349	0.148	0.108	0.213	0.220
	Label	0.314	0.357	0.185	0.113	0.299	0.253
	Prob. vector	0.354	0.394	0.188	0.121	0.256	0.263
	Feat. vector	0.357	0.374	0.181	0.120	0.327	0.272

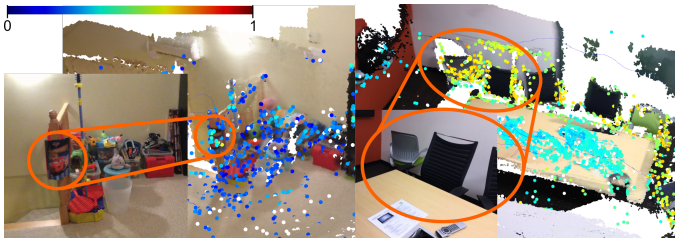

Fig. 3. Similarity of 3D features to *"rayo mcqueen"*, and *"something to sit on"* respectively with the associated image that observed it.

We employ two models for semantic inference, Segformer trained on ADE20k and the pixel-based CLIP model for closed and open-set segmentation respectively, using multiple semantic representations termed *Label, Probability vector, and Feature vector*. Notably, Segformer is applied off-the-shelf by mapping ADE20k labels to NYU40, showing competitive performance despite the domain shift. We report the macro-averaged precision for several scenes and their overall average in Table I. Results show that using a fused 3D map often leads to more accurate segmentation, although the improvement depends on the chosen representation.

Beyond quantitative evaluation, we provide qualitative results for open-vocabulary mapping. Figure 3 shows maps queried using natural language, with relevance visualized as a heatmap based on similarity to the text query. This example illustrates how the framework enables seamless integration of new methods and paves the way for future research that leverages semantics within the SLAM pipeline in a more unified and extensible manner.

IV. Conclusions

This work presents a unified implementation for integrating semantic information into the pySLAM framework, enabling consistent evaluation and rapid prototyping of semantic mapping methods. The integration supports diverse research directions, including the use of semantics within SLAM for tasks such as evaluating descriptor quality. Our results emphasize the need for further investigation into semantic representations and fusion. The framework also opens new possibilities for high-level applications requiring contextual understanding of the environment.

REFERENCES

[1] Luigi Freda. pyslam: An open-source, modular, and extensible framework for slam, 2025.

[2] Alejandro Fontan, Tobias Fischer, Javier Civera, and Michael Milford. Vslam-lab: A comprehensive framework for visual slam methods and datasets. *arXiv preprint arXiv:2504.04457*, 2025.

[3] John McCormac, Ankur Handa, Andrew Davison, and Stefan Leutenegger. SemanticFusion: Dense 3D semantic mapping with convolutional neural networks. In *IEEE Int. Conf. on Robotics and Automation*, pages 4628–4635, 2017.

[4] Antoni Rosinol, Marcus Abate, Yun Chang, and Luca Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2020.

[5] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19729–19739, 2023.

[6] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical Open-Vocabulary 3D Scene Graphs for Language-Grounded Robot Navigation. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024.

[7] Tomas Berriel Martins, Martin R Oswald, and Javier Civera. Ovo-slam: Open-vocabulary online simultaneous localization and mapping. *arXiv preprint arXiv:2411.15043*, 2024.

[8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[9] David Morilla-Cabello, Lorenzo Mur-Labadia, Ruben Martinez-Cantin, and Eduardo Montijano. Robust fusion for bayesian semantic mapping. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 76–81, 2023.

[10] Mingrui Li, Shuhong Liu, Heng Zhou, Guohao Zhu, Na Cheng, Tianchen Deng, and Hongyu Wang. Sgs-slam: Semantic gaussian splatting for neural dense slam. In *European Conference on Computer Vision*, pages 163–179. Springer, 2024.

[11] William Shen, Ge Yang, Alan Yu, Jansen Wong, Leslie Pack Kaelbling, and Phillip Isola. Distilled feature fields enable few-shot language-guided manipulation. In *7th Annual Conference on Robot Learning*, 2023.

[12] Kunyi Li, Michael Niemeyer, Nassir Navab, and Federico Tombari. Dns-slam: Dense neural semantic-informed slam. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7839–7846, 2024.

[13] Martin Runz, Maud Buffier, and Lourdes Agapito. Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 10–20, 2018.

[14] Berta Bescos, José M. Fácil, Javier Civera, and José Neira. Dynaslam: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robotics and Automation Letters*, 3(4):4076–4083, 2018.

[15] Irene Ballester, Alejandro Fontán, Javier Civera, Klaus H. Strobl, and Rudolph Triebel. Dot: Dynamic object tracking for visual slam. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11705–11711, 2021.

[16] Binbin Xu, Wenbin Li, Dimos Tzoumanikas, Michael Bloesch, Andrew Davison, and Stefan Leutenegger. Mid-fusion: Octree-based object-level multi-instance dynamic slam. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 5231–5237, 2019.

[17] M. Grinvald, F. Furrer, T. Novkovic, J. J. Chung, C. Cadena, R. Siegwart, and J. Nieto. Volumetric Instance-Aware Semantic Mapping and 3D Object Discovery. *IEEE Robotics and Automation Letters*, 4(3):3037–3044, July 2019.

[18] John McCormac, Ronald Clark, Michael Bloesch, Andrew Davison, and Stefan Leutenegger. Fusion++: Volumetric Object-Level SLAM. In *Int. Conf. on 3D Vision*, pages 32–41, 2018.

[19] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. 2023.

[20] Lukas Schmid, Marcus Abate, Yun Chang, and Luca Carlone. Khronos: A unified approach for spatio-temporal metric-semantic slam in dynamic environments. In *Proc. of Robotics: Science and Systems (RSS)*, Delft, Netherlands, July 2024.

[21] Ayca Takmaz, Alexandros Delitzas, Robert W. Sumner, Francis Engelmann, Johanna Wald, and Federico Tombari. Search3d: Hierarchical open-vocabulary 3d segmentation. *IEEE Robotics and Automation Letters*, 10(3):2558–2565, 2025.

[22] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[23] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[24] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Neural Information Processing Systems (NeurIPS)*, 2021.