Language-EXtended Indoor SLAM (LEXIS): A Versatile System for Real-time Visual Scene Understanding

Christina Kassab, Matias Mattamala, Lintong Zhang, and Maurice Fallon

Abstract*— Versatile and adaptive semantic understanding would enable autonomous systems to comprehend and interact with their surroundings. Existing fixed-class models limit the adaptability of indoor mobile and assistive autonomous systems. In this work, we introduce LEXIS, a real-time indoor Simultaneous Localization and Mapping (SLAM) system that harnesses the open-vocabulary nature of Large Language Models (LLMs) to create a unified approach to scene understanding and place recognition. The approach first builds a topological SLAM graph of the environment (using visual-inertial odometry) and embeds Contrastive Language-Image Pretraining (CLIP) features in the graph nodes. We use this representation for flexible room classification and segmentation, serving as a basis for room-centric place recognition. This allows loop closure searches to be directed towards semantically relevant places. Our proposed system is evaluated using both public, simulated data and real-world data, covering office and home environments. It successfully categorizes rooms with varying layouts and dimensions and outperforms the state-of-the-art (SOTA). For place recognition and trajectory estimation tasks we achieve equivalent performance to the SOTA, all also utilizing the same pre-trained model. Lastly, we demonstrate the system's potential for planning. Video at: `https://youtu.be/gRqF3euDfX8`

I. Introduction

Scene understanding is a long-standing problem in robot perception. Over the last decade, SLAM systems have shifted from building purely geometric representations for localization, to semantic and interpretable representations for interaction [1], [2]. Semantic SLAM and object-based perception have made significant advances — powered by progress in the machine learning and computer vision communities. *3D scene graphs* [3], [4] have more recently emerged as a unifying representation to integrate structure and semantics [5], [6]. Nonetheless, the usage of fixed-class semantic models in these applications limits the versatility of these systems.

The progress of LLM research offers a solution to this challenge, as they can bridge the gap between visual and textual information with their open vocabularies. Methods such as CLIP [7] and ViLD [8] have been used to enrich 3D reconstructions with semantics, as demonstrated by methods such as OpenScene [9], ConceptFusion [10], and NLMap [11]. These methods can identify objects and scene properties; and can even carry out navigation using human instructions [12]. However, open questions remain about integrating this capability into the modules of a robotic system. In particular, can embedded semantic understanding be harnessed for tasks such as place recognition and localization?

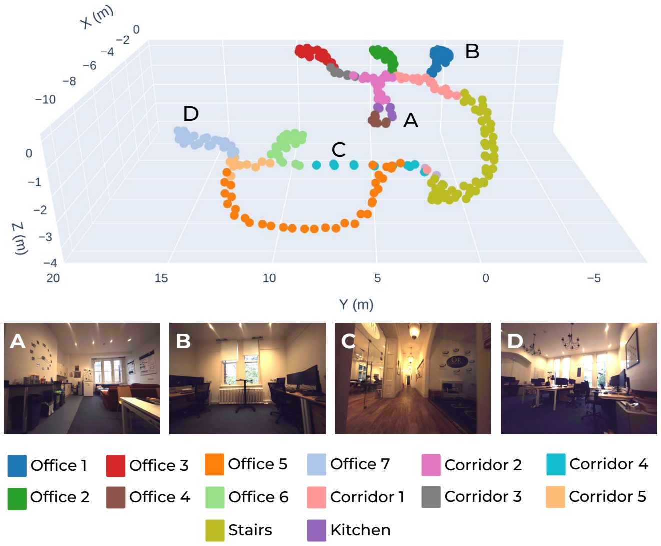

Fig. 1: LEXIS enables pose graph segmentation from natural language. By exploiting the open-vocabulary capabilities of CLIP, we can segment room instances such as `office`, `kitchen`, and `corridor` directly from the pose graph without fine-tuning. The above dataset is from a two floor office environment and contains 7 rooms as well as 2 corridors and stairs.

In this work, we combine the open-vocabulary capabilities of LLMs with classical localization and mapping methods to develop LEXIS (Language-EXtended Indoor SLAM). Unlike conventional approaches which employ separate models for room classification, place recognition and semantic understanding, our approach uses a single pre-trained model to efficiently execute all of these functions. The output is a semantically segmented pose graph as shown in Fig. 1. Our specific contributions are:

- A lightweight topological pose graph representation embedded with CLIP features.
- A method to leverage semantic features to achieve online room segmentation, capable of accomodating different room sizes, layouts and open-floor plans.
- A place recognition approach building on these room segmentations to propose hierarchical, room-aware loop closures.
- Extensive evaluation of the system for indoor real-time room segmentation and classification, place recognition, and as a unified visual SLAM system using standard and custom multi-floor datasets, with a demonstration for planning tasks.

The authors are with the Oxford Robotics Institute at the University of Oxford, UK. {`christina, matias, lintong, mfallon`}`@robots.ox.ac.uk`

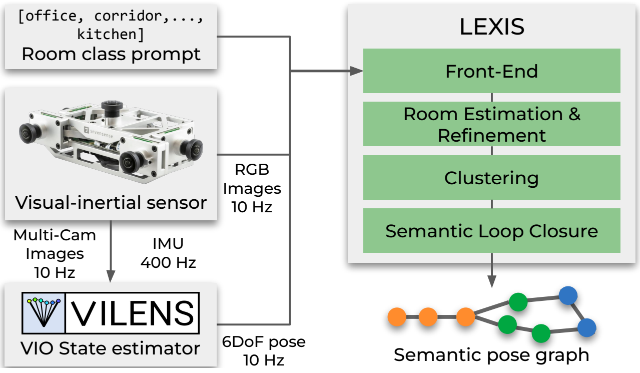

Fig. 2: LEXIS system overview: The only inputs are RGB images and an odometry estimate from a visual-inertial state estimator, as well as a prompt list of potential room classes. The output is a semantic pose graph that encodes room information.

II. Related Work

A. Semantic Scene Representations

The use of semantic information is motivated by the limitations of purely geometric representations to encode interpretable information and to support higher-level tasks.

Early research used boosting and Hidden Markov Models to label indoor locations based on vision and laser range data [13]. Later works shifted towards Convolutional Neural Network (CNN)-based methods for room classification and scene understanding. Goeddel et al. [14] used CNNs to classify LiDAR maps into rooms, corridors and doorways. Sunderhauf et al. [15] overcame the closed-set limitations of CNNs using a series of one-vs-all classifiers to allow recognition of new semantic classes, such as allowing generalization of door recognition to diverse settings.

These techniques emphasize the extraction of semantic information but do not capture contextual and higher-level understanding, such as relationships between objects or rooms. More recent studies are directed towards the incorporation of these semantic attributes directly into hierarchical map models, such as 3D scene graphs [3], [4]. The multi-layered graph represents entities such as objects, rooms, or buildings as graph nodes, while semantic relationships are established through graph edges. Hydra [5] presented a five-layered scene graph with a metric-semantic 3D mesh layer, object and agents layers, as well as, obstacle-free locations, rooms, and buildings. Semantics are obtained through a pretrained HR-Net [16] and Graph Neural Networks (GNNs) models to encode object relationships. S-Graphs+ [6] used a similar four-layered graph and employed geometry-based room segmentation using free-space clusters and wall planes, without using an explicit semantic segmentation method.

These methods rely on fixed-class models for tasks like room classification, and face challenges in generalizing to new environments [14], [17], leading to reduced performance and difficulties with unfamiliar room types. Approaches like Hydra need to segment the representation prior to classification and depend on geometric data (e.g., walls and doorways). This limits segmentation in open-floor plans or multi-functional spaces. Moreover, current 3D scene graph representations require multiple models for semantic segmentation, room classification, and place recognition. This requires extensive training data and further diminishes adaptability to varied environments.

With LEXIS we aim to address these limitations by exploiting the information encoded in LLMs. Their open-vocabulary features enable an arbitrary number of classes, and allow LEXIS to adapt to diverse indoor environments without the need for pre-training or fine-tuning. Our system does not require geometric information to perform room segmentation, enabling us to accommodate varying room sizes and layouts, and allowing us to segment open plan spaces effectively. Additionally, we leverage the same model for place recognition, thereby fully capitalizing on the capabilities of LLMs throughout the entire SLAM pipeline.

B. LLM-powered Representations

Our system is inspired by other recent works exploiting LLMs for scene representation.

OpenScene [9] is an offline system that enhances 3D metric representations using CLIP visual-language features. Other approaches, such as LERF [18] and CLIP-Fields [19], embed visual-language features into neural fields to achieve 3D semantic segmentations from open-vocabulary queries. ConceptFusion [10] further advances these approaches by building a representation featuring multi-modal features from vision, audio, and language on top of a differentiable SLAM pipeline [20].

These representations have proven useful when interacting with 3D scenes, especially for planning and navigation tasks. Natural language commands have been employed to guide navigation tasks in indoor environments, combining natural language plan specifications with classical state estimation and local planning systems for navigation [19], [11], [12]. Other 3D scene understanding tasks, such as completing partially observed objects and localizing hidden objects have been explored [21].

LEXIS differs from the previous methods as they heavily rely on a metric representation of the environment, necessitating the embedding and fusion of LLMs features into a 2D or 3D map. This fusion process often needs to be performed offline or is limited to single-room environments.

In contrast, our system utilizes a topological representation—a pose graph—which streamlines feature embedding while preserving the ability to use natural language queries for segmentation in an online manner. Moreover, it allows us to apply well-established loop closing and pose graph optimization techniques to handle trajectory drift effectively.

III. Method

A system overview of LEXIS is presented in Figure 2. The main inputs are high-frequency 6 DoF odometry (for which we use our previous work Multi-Camera VILENS [22]), a stream of wide field-of-view (FoV) RGB images, and a list of potential room classes (for example: `office`, `kitchen`,

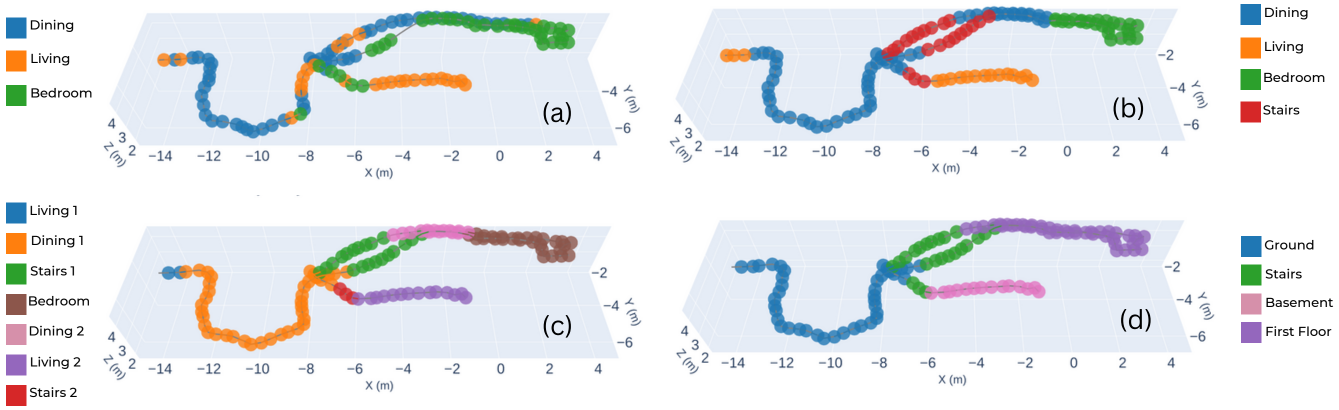

Fig. 3: Room segmentation and refinement on a pose graph with data from the uHumans2 Apartment scene (*uH2-Apt*). (a) Initial room labels are given by CLIP. (b) The room labels post refinement. (c) Clustering into room instances. (d) Segmentation into floors.

`corridor`). The output of the system is a CLIP-enhanced semantically-segmented topological map of the environment. The main modules of LEXIS are explained in the following sections.

A. *Front-end*

Using the high-frequency odometry estimate and RGB image stream, LEXIS builds an incremental pose graph of equally-spaced keyframes, based on a pre-set distance threshold. The state of the system at time t_i is defined as $\mathbf{T}^{i}_{\mathtt{WB}} \in SE(3)$, where `W` is the fixed world frame, and `B` is the moving base frame.

As well as the pose, each node also contains CLIP image encodings for semantic understanding which we define as $\mathbf{f}_{\text{CLIP}}$. We also extract AKAZE [23] local features, $\mathbf{f}_{\text{AKAZE}}$, for loop closure registration. We extract text encodings, denoted as $\mathbf{f}_{\text{TEXT}}$, from the prior list of potential room classes by utilizing the same CLIP model. It is important to emphasize that the room labels are not limited to predefined categories associated with any particular dataset.

B. *Room Estimation and Refinement*

As we build the graph, we compare the image encodings, $\mathbf{f}_{\text{CLIP}}$, to the room text encodings, $\mathbf{f}_{\text{TEXT}}$, using the cosine similarity defined as:

$$S_c(\mathbf{f}_{\text{CLIP}}, \mathbf{f}_{\text{TEXT}}) = \frac{\mathbf{f}_{\text{CLIP}} \cdot \mathbf{f}_{\text{TEXT}}}{\|\mathbf{f}_{\text{CLIP}}\| \cdot \|\mathbf{f}_{\text{TEXT}}\|} \tag{1}$$

This provides an initial room segmentation for the pose graph, as shown in Fig. 3 (a). As this module is executed on a per-image basis without contextual information, it can make incorrect classifications, particularly in areas with room transitions or when images lack distinct semantic content. To mitigate this, we employ a nearest neighbour refinement.

The refinement approach is inspired by the Label Propagation algorithm [24], a well-known technique for finding communities in network structures. Considering the C closest neighbors of each node, Label Propagation identifies the most common label among them. If this label differs from the node's present label, the algorithm updates the node's label. However, in contrast to Label Propagation, which updates all the labels until convergence, we only run one forward pass every K new keyframes. In our experiments, we found that values of C between 3 and 7, and K between 7 and 12, produced comparable results across diverse indoor environments, including open-floor plans.

We also use height change (in the z-axis) to detect and segment staircases, even when they are not visibly present in the immediate surroundings. The outcome of the full refinement module, consisting of a room label for each pose in the pose graph, is shown in Fig. 3 (b).

C. *Clustering*

Once we have allocated a room label to each pose graph node, the next step is to group the nodes into clusters representing individual rooms such as `office 1` and `office 2`. For each new node with a room label not encountered previously, a new cluster is formed. Nodes are then added to the cluster if they possess the same room label and are within a certain distance threshold of the cluster's mean position. This approach enables continuous updates to the clusters during the refinement module, as allocated room labels evolve over time.

When dealing with rooms of significantly varying sizes, it is possible for multiple clusters to emerge within a single room. In these cases, we merge clusters that have the same room label and do not have an intermediate space between them such as a corridor. The clustering outcome is presented in Fig. 3 (c). Furthermore, by identifying clusters labelled as `stairs` we can further segment the pose graph into distinct floors (Fig. 3 (d)).

By employing this strategy, our system can organize the pose graph into a structured representation of meaningful room instances which can also enhance subsequent localization and loop closure modules. This adaptive clustering approach ensures robust segmentation and can accommodate various room layouts and sizes commonly encountered in real-world indoor environments.

D. Semantic Loop Closure Detection

The semantic information encoded in the LEXIS graph allows for efficient place retrieval without using a dedicated place recognition model. Because of this we can reuse this information for loop closure candidate detection.

For each new keyframe added to the graph, we first determine its corresponding candidate room label using the image encoding, $\mathbf{f}_{\text{CLIP}}$. We then search for candidate rooms by querying all the room clusters sharing the same label.

We use the current localization estimate provided by the odometry to choose the closest room cluster, and attempt geometric verification against all the keyframes within the room using PnP [25]. For efficiency, the query node's image encoding, $\mathbf{f}_{\text{CLIP}}$, can also be compared to nodes within the cluster using cosine similarity, further refining the candidate set. All successful localization attempts are then added as loop closure edges in the pose graph, which is later optimized. The optimised poses are defined as $\mathcal{X} := \{\mathbf{T}^1_{\mathtt{WB}}, ..., \mathbf{T}^n_{\mathtt{WB}}\}$ with the optimization formulated as a least squares minimization with a robust DCS loss $\rho(\cdot)$ [26]:

$$\mathcal{X} = \underset{\mathcal{X}}{\operatorname{argmin}} \sum_{i} \|\mathbf{r}_{\text{odom}}\|^2 + \sum_{i,j} \rho\left(\mathbf{r}_{\text{loop}}\right) \tag{2}$$

where, $\mathbf{r}_{\text{odom}}$ refers to odometry edges and $\mathbf{r}_{\text{loop}}$ refers to loop closures.

IV. Experiments and Results

In this section, we demonstrate the capabilities of the system as applied to room classification, place recognition and as a unified SLAM system using indoor real-world and simulated datasets. We conclude with a demonstration of a mission planning application.

A. Experimental Setup

LEXIS runs in real-time on a mid-range laptop with an Intel i7 11850H @ 2.50GHz x 16 with an Nvidia RTX A3000 GPU. The only module that requires GPU compute is the CLIP feature extractor. All other modules run on the CPU.

There are several pre-trained CLIP models which use different variants of a ResNet (RN) or a Vision Transformer (ViT) as a base. Two of the ResNet variants follow a EfficientNet-style model scaling and use approximately 4x and 16x the compute of ResNet-50. A full list of the models evaluated are available in Tab. I.

We evaluated LEXIS on three datasets:

- *uHumans2* is a Unity-based simulated dataset provided by the authors of Kimera [27]. It has two indoor scenes: a small apartment (*uH2-Apt* [49m, 4 rooms, 3 floors]) and an office (*uH2-Off* [264m, 4 rooms, 1 floor]). The dataset provides visual-inertial data, ground truth trajectories and ground-truth bounding boxes for each room.
- *ORI* [253m, 7 rooms, 2 floors] is a real-world dataset collected at the Oxford Robotics Institute and it includes offices, staircases and a kitchen. It was collected using a multi-sensor unit consisting of the Sevensense Alphasense Multi-Camera kit (Fig. 2) integrated with a Hesai Pandar LiDAR.
- *Home* [118m, 7 rooms, 2 floors] is a dataset collected from a home environment, including kitchen, bedrooms, bathroom, living and dining areas, and a garden. This dataset was recorded and labeled using the same approach as the *ORI*.

TABLE I: Mean classification accuracy and standard deviation (over 5 runs). Inference time for extracting both image and text encodings for the available CLIP models is also provided. Models with inference time of over 40 ms were disregarded from further analysis.

	Home (%)	*uH2-Apt* (%)	Inference time (ms)
RN50	73.40 ± 6.42	53.77 ± 1.56	21.62 ± 0.04
RN101	70.66 ± 2.65	52.49 ± 3.98	27.28 ± 0.75
RN50x4	74.58 ± 5.40	55.12 ± 5.75	28.37 ± 1.53
RN50x16	75.85 ± 2.97	**57.36 ± 1.27**	37.14 ± 0.37
RN50x64	-	-	63.57 ± 0.25
ViT-B/32	74.96 ± 4.92	55.51 ± 2.05	20.74 ± 0.17
ViT-B/16	77.55 ± 3.66	56.90 ± 3.51	22.36 ± 0.09
ViT-L/14	**78.92 ± 3.01**	**57.47 ± 1.81**	35.61 ± 0.78
ViT-L/14@336px	-	-	46.41 ± 0.25

For both *ORI* and *Home*, the LiDAR sensor was used to generate ground truth. It was not used in LEXIS. Ground truth trajectories were determined via LiDAR ICP registration against prior maps built with a Leica BLK360, room labels were hand-labeled using the LiDAR map.

B. Results

1) Room Segmentation and Classification: We define classification accuracy as the ratio of accurately classified nodes relative to the total number of nodes in a dataset. A node is considered accurately classified if the bounding box that it falls into has the same room label as the node itself. The reported accuracy is an average of five runs.

An evaluation of room classification accuracy on a real dataset (*Home*) and a simulated one (*uH2-Apt*) using the available CLIP models is shown in Table I. RN50x64 and ViT-L/14@336px were excluded due to their larger size and longer inference times. Refinement parameters C and K were tuned for each model and dataset, with the best performing models being RN50x16 and ViT-L/14. For further evaluation, we selected RN50x16 for the uHumans2 datasets, as it required less refinement and more effectively preserved small-scale changes in open-floor plans compared to ViT-L/14. We used ViT-L/14 for all evaluations on the *Home* and *ORI* datasets.

Table II presents a comparison of LEXIS using both the initial segmentation from CLIP (LEXIS - Baseline) and the refined outcome (LEXIS - Refined) with Hydra [17]. Hydra employs two different 2D semantic segmentation models, using the ADE20k dataset label space [28]. The two

TABLE II: Mean room classification accuracy and standard deviation over 5 runs.

%	*uH2-Apt*	*uH2-Off*	*ORI*	*Home*
Hydra - HRNet	38.0 ± 21.7	28.4 ± 6.9	-	-
Hydra - OneFormer	45 ± 11.2	27.0 ± 10.1	-	-
LEXIS - Baseline	51.31 ± 3.24	68.99 ± 1.07	68.09 ± 1.64	61.21 ± 1.12
LEXIS - Refined	**57.36 ± 1.27**	**76.03 ± 1.69**	**79.22 ± 4.23**	**78.92 ± 3.01**

segmentation models are HRNet [16] and OneFormer [29]. We use the same hand-labeled ground-truth bounding boxes as Hydra, but please note that Hydra's evaluation method computes accuracy over clustered room nodes as opposed to the pose nodes used in our method.

Although the evaluation method differs slightly and it is difficult to precisely compare, our results still show a marked improvement on both Hydra variants. Across the datasets, the refinement procedure improves classification accuracy by an average of 10%. The key advantage of our open-vocabulary approach is its ability to avoid the constraints of fixed class sets, facilitating effective generalization to diverse environments and accurate segmentation of open-floor plans using semantics rather than geometry. For example, in *uH2-Apt*, the algorithm successfully segmented the `living room` and `dining room` despite the open floor-plan (Fig. 3). Similarly, within the *ORI* dataset, LEXIS divided the `kitchen` area into `kitchen` and `office` spaces as it contains typical kitchen equipment as well as whiteboards and tables (Fig. 1).

The variation in the performance of LEXIS on *uH2-Apt* and *uH2-Off* can be attributed to the ground-truth bounding boxes provided with the dataset, where stairs are not considered a separate class but instead included as part of the `dining room`. Moreover, we hypothesize that the performance difference between Hydra and LEXIS on the *uH2-Off* dataset can be attributed to Hydra considering only the object-room graph in training and classification. For instance, in the *uH2-Off* dataset, objects such as chairs and water dispensers are positioned within the corridors. This can be misclassified as `office` when only considering the objects in the scene, especially if the objects are out-of-distribution from the original training set.

Our results on the *uH2-Off* dataset are visualized in Fig. 4, with orange regions 🟧 indicating misclassifications. Incorrect classifications are typically clustered around room edges, e.g., when the camera faces into a room but is actually located within a corridor (Example A).

2) Semantic Place Recognition: We compared LEXIS' place recognition method to DBoW [30], and NetVLAD [31]. DBoW provides a framework for feature quantization and indexing of large-scale visual vocabularies. We fed DBoW with ORB features [32], as used in the place recognition systems of ORB-SLAM [33] and Hydra [5]. NetVLAD is a neural network architecture pre-trained on Pitts30k [34].

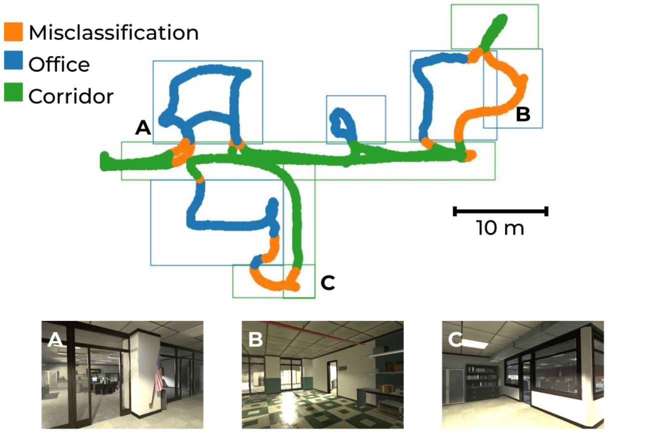

Fig. 4: Segmentations produced by LEXIS for the uHumans2 office (*uH2-Off*) dataset. Also shown are the ground-truth bounding boxes used in Hydra's evaluation. Misclassifications occur during room transitions (example A and B); or areas with fewer features (C).

We evaluated performance by counting true positives and false positives (Fig. 5). For each query, if N matches were situated within a distance/angle threshold, we counted it as a true positive; otherwise, a false positive count was registered. We conducted evaluations at $N = 1$, 3, and 5. We used the *Home* and *ORI* datasets, with the true positive distance threshold set at 1 m and angular threshold of 0.5 rad.

As illustrated in Fig. 5, our approach achieved more true positives and less false positives than DBoW across both datasets. The increased number of true positives can be attributed to the refinement of our search for loop closures to a relevant room or corridor.

Interestingly, our method retrieved a similar number of true positives to NetVLAD despite relying on CLIP, a pre-

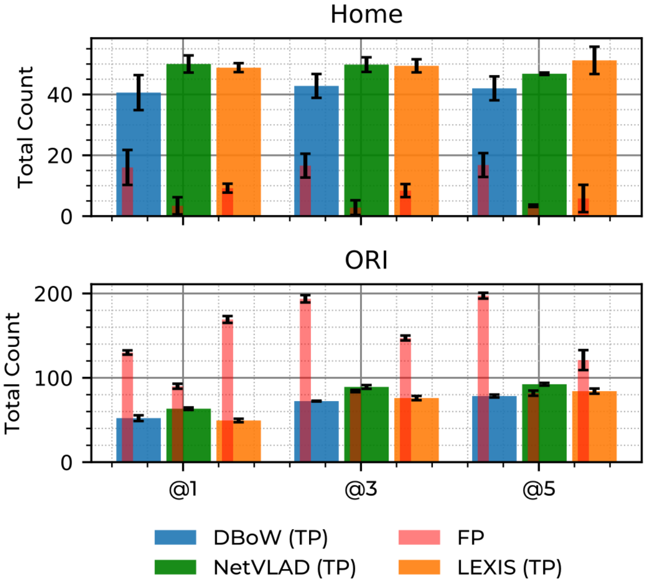

Fig. 5: Number of true positives and false positives (red 🟥) using three different VPR methods: DBoW, NetVLAD and LEXIS on the *Home* (left) and *ORI* (right) dataset averaged over 5 runs.

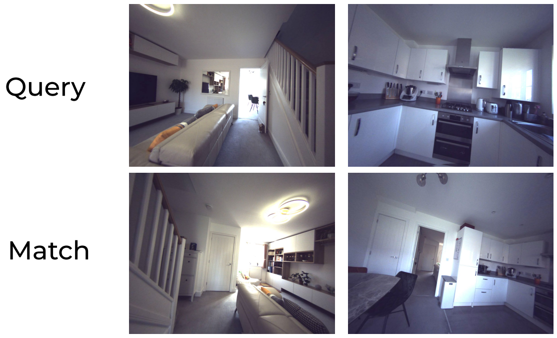

Fig. 6: Examples of loop closures provided by CLIP in the *Home* dataset. CLIP is able to provide matches from opposing viewpoints (left) and with significant viewpoint variations (right) as it relies on semantic information.

trained model, with no specific training for place recognition. We also found that LEXIS produced a slightly higher number of false positives than NetVLAD. This is primarily due to the viewpoint variations in the suggested matches produced by our method, as demonstrated in Fig. 6. Notably, due to CLIP relying solely on semantic information, opposing viewpoints are presented as potential matches.

The high number of false positives across all methods in the *ORI* dataset could be attributed to there being many visually similar offices in the dataset. However, it is worth noting, particularly with robust graph optimization techniques and PnP verification, prioritizing the accurate identification of enough valid loop closures (true positives) is more important than avoiding incorrect loop closures (false positives) [35].

3) Full System Evaluation: We conducted a comparison using LEXIS as a complete SLAM system, benchmarked against two state-of-the-art alternatives: ORB-SLAM3 [36] and VINS-Fusion [37]. In our experiments, we used the stereo-inertial configurations with loop closures enabled and assessed performance using the Absolute Trajectory Error (ATE). The results are summarized in Table III.

TABLE III: Comparison of ATE in the *ORI* and *Home* datasets.

ATE (m)	*ORI*	*Home*
ORB-SLAM3	0.22	0.10
VINS-Fusion	0.10	0.08
LEXIS	0.16	0.10

Despite the streamlined and minimal design of LEXIS — combining the Multi-Camera VILENS VIO system [22], with classical pose graph optimization and our CLIP-based semantic place recognition module, it still achieves comparable performance to that of ORB-SLAM3 and VINS-Fusion. The incorporation of the Multi-Camera system, which analyzes images from two front-facing and two lateral-facing cameras, provides benefits as the system can avoid tracking issues in confined indoor environments.

4) Planning Application: Finally, we demonstrated that the representation produced by LEXIS can be used for mission planning in a real-world environment encompassing multiple floors and rooms. From the pose graph, we constructed an adjacency matrix that establishes connections between consecutive nodes and nodes within the same cluster. We then computed the shortest path between initial and goal room labels using Dijkstra's algorithm [38]. An example path on the *Home* dataset is illustrated in Fig. 7.

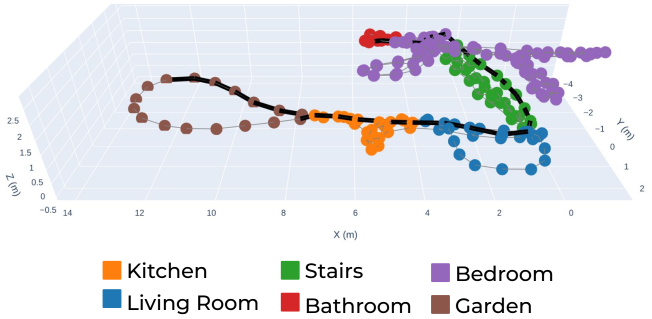

Fig. 7: Segmentation of the *Home* dataset with a topological plan, shown in black ■, from the `bathroom` to the `garden`.

V. Conclusion

This work presents LEXIS, a real-time semantic visual SLAM system enhanced by open-vocabulary language models. Our system constructs a topological model of indoor environments that is enriched with embedded semantic understanding. This allows us to properly segment rooms and spaces across diverse contexts. Leveraging this representation, we demonstrated room-aware place recognition which achieves performance equivalent with established place recognition methods such as NetVLAD and DBoW. We evaluated our SLAM system in home and office environments and achieved comparable ATE to established systems (ORB-SLAM3 and VINS-Fusion). Finally, we demonstrated an example of how our representation can be used for other robotics tasks such as room-to-room planning. This work showcases how open-vocabulary models can enable autonomous systems to interact naturally with their environment. Future work will focus on enhancing room classification by integrating LEXIS with dense reconstruction techniques and considering uncertainty in the estimation for long term use of the system. We also intend to investigate per-pixel adaptations of the CLIP model.

Acknowledgments

This work is supported in part by a Royal Society University Research Fellowship (Fallon, Kassab), and the ANID/BECAS CHILE/2019-72200291 (Mattamala). We thank Nathan Hughes for providing the ground truth for the uHumans2 dataset. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission.

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Trans. Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] A. J. Davison, "FutureMapping: The Computational Structure of Spatial AI Systems," *CoRR*, vol. abs/1803.11288, 2018.

[3] I. Armeni, Z.-Y. He, J. Gwak, A. R. Zamir, M. Fischer, J. Malik, and S. Savarese, "3d scene graph: A structure for unified semantics, 3d space, and camera," in *IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2019, pp. 5664–5673.

[4] U.-H. Kim, J.-M. Park, T.-j. Song, and J.-H. Kim, "3-D Scene Graph: A Sparse and Semantic Representation of Physical Environments for Intelligent Agents," *IEEE Trans. Cybern.*, vol. 50, no. 12, pp. 4921–4933, 2020.

[5] N. Hughes, Y. Chang, and L. Carlone, "Hydra: A Real-time Spatial Perception System for 3D Scene Graph Construction and Optimization," in *Robotics: Science and Systems (RSS)*, 2022.

[6] H. Bavle, J. L. Sanchez-Lopez, M. Shaheer, J. Civera, and H. Voos, "S-Graphs+: Real-Time Localization and Mapping Leveraging Hierarchical Representations," *IEEE Robot. Autom. Lett. (RA-L)*, vol. 8, no. 8, pp. 4927–4934, 2023.

[7] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," *CoRR*, vol. abs/2103.00020, 2021.

[8] X. Gu, T. Lin, W. Kuo, and Y. Cui, "Open-vocabulary Object Detection via Vision and Language Knowledge Distillation," in *Intl. Conf. on Learning Representations (ICLR)*, 2022.

[9] S. Peng, K. Genova, C. M. Jiang, A. Tagliasacchi, M. Pollefeys, and T. Funkhouser, "OpenScene: 3D Scene Understanding with Open Vocabularies," in *IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2023.

[10] K. M. Jatavallabhula, A. Kuwajerwala, Q. Gu, M. Omama, T. Chen, S. Li, G. Iyer, S. Saryazdi, N. Keetha, A. Tewari, J. B. Tenenbaum, C. M. de Melo, M. Krishna, L. Paull, F. Shkurti, and A. Torralba, "ConceptFusion: Open-set Multimodal 3D Mapping," in *Robotics: Science and Systems (RSS)*, 2023.

[11] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, "Open-vocabulary Queryable Scene Representations for Real World Planning," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 11 509–11 522.

[12] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual Language Maps for Robot Navigation," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023.

[13] A. Rottmann, O. Mozos, C. Stachniss, and W. Burgard, "Semantic Place Classification of Indoor Environments with Mobile Robots Using Boosting." in *Proceedings of the National Conference on Artificial Intelligence*, vol. 3, 01 2005, pp. 1306–1311.

[14] R. Goeddel and E. Olson, "Learning semantic place labels from occupancy grids using CNNs," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 3999–4004.

[15] N. Sünderhauf, F. Dayoub, S. McMahon, B. Talbot, R. Schulz, P. Corke, G. Wyeth, B. Upcroft, and M. Milford, "Place Categorization and Semantic Mapping on a Mobile Robot," *CoRR*, vol. abs/1507.02428, 2015.

[16] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep High-Resolution Representation Learning for Visual Recognition," *CoRR*, vol. abs/1908.07919, 2019.

[17] N. Hughes, Y. Chang, S. Hu, R. Talak, R. Abdulhai, J. Strader, and L. Carlone, "Foundations of Spatial Perception for Robotics: Hierarchical Representations and Real-time Systems," *CoRR*, vol. abs/2305.07154, 2023.

[18] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik, "LERF: Language Embedded Radiance Fields," *CoRR*, vol. abs/2303.09553, 2023.

[19] N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam, "CLIP-Fields: Weakly Supervised Semantic Fields for Robotic Memory," in *Robotics: Science and Systems (RSS)*, 2023.

[20] K. M. Jatavallabhula, G. Iyer, and L. Paull, "Grad SLAM: Dense SLAM meets Automatic Differentiation," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2020, pp. 2130–2137.

[21] H. Ha and S. Song, "Semantic Abstraction: Open-World 3D Scene Understanding from 2D Vision-Language Models," *CoRR*, vol. abs/2207.11514, 2022.

[22] L. Zhang, D. Wisth, M. Camurri, and M. F. Fallon, "Balancing the Budget: Feature Selection and Tracking for Multi-Camera Visual-Inertial Odometry," *IEEE Robot. Autom. Lett. (RA-L)*, vol. 7, no. 2, pp. 1182–1189, 2022.

[23] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "KAZE Features," in *Eur. Conf. on Computer Vision (ECCV)*, vol. 7577, 2012, pp. 214–227.

[24] X. Zhu and Z. Ghahramani, "Learning from Labeled and Unlabeled Data with Label Propagation," Carnegie Mellon University, Tech. Rep., 2002.

[25] M. A. Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Commun. ACM*, vol. 24, no. 6, p. 381–395, 1981.

[26] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2013, pp. 62–69.

[27] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, and L. Carlone, "Kimera: from SLAM to Spatial Perception with 3D Dynamic Scene Graphs," *Intl. J. of Robot. Res.*, 2021.

[28] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene Parsing through ADE20K Dataset," in *IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2017, pp. 5122–5130.

[29] J. Jain, J. Li, M. Chiu, A. Hassani, N. Orlov, and H. Shi, "OneFormer: One Transformer to Rule Universal Image Segmentation," in *IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2023, pp. 2989–2998.

[30] D. Galvez-López and J. D. Tardos, "Bags of Binary Words for Fast Place Recognition in Image Sequences," *IEEE Trans. Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[31] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2016, pp. 5297–5307.

[32] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Intl. Conf. on Computer Vision (ICCV)*, D. N. Metaxas, L. Quan, A. Sanfeliu, and L. V. Gool, Eds., 2011, pp. 2564–2571.

[33] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," *IEEE Trans. Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[34] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual Place Recognition with Repetitive Structures," in *IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2013, pp. 883–890.

[35] S. Schubert, P. Neubert, S. Garg, M. Milford, and T. Fischer, "Visual Place Recognition: A Tutorial," *CoRR*, vol. abs/2303.03281, 2023.

[36] C. Campos, R. Elvira, J. J. Gómez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM," *IEEE Trans. Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[37] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[38] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.