

An Object SLAM Framework for Association, Mapping, and High-Level Tasks

Yanmin Wu, Yunzhou Zhang, Delong Zhu, Zhiqiang Deng, Wenkai Sun, Xin Chen, and Jian Zhang

Abstract—Object SLAM is considered increasingly significant for robot high-level perception and decision-making. Existing studies fall short in terms of data association, object representation, and semantic mapping and frequently rely on additional assumptions, limiting their performance. In this paper, we present a comprehensive object SLAM framework that focuses on object-based perception and object-oriented robot tasks. First, we propose an ensemble data association approach for associating objects in complicated conditions by incorporating parametric and nonparametric statistic testing. In addition, we suggest an outlier-robust centroid and scale estimation algorithm for modeling objects based on the iForest and line alignment. Then a lightweight and object-oriented map is represented by estimated general object models. Taking into consideration the semantic invariance of objects, we convert the object map to a topological map to provide semantic descriptors to enable multi-map matching. Finally, we suggest an object-driven active exploration strategy to achieve autonomous mapping in the grasping scenario. A range of public datasets and real-world results in mapping, augmented reality, scene matching, relocalization, and robotic manipulation have been used to evaluate the proposed object SLAM framework for its efficient performance.

Index Terms—Visual SLAM, Data Association, Semantic Mapping, Augmented Reality, Robotics.

I. INTRODUCTION

THE fundamental issues in terms of the accuracy and efficiency of visual SLAM have been vastly improved over the past two decades, which enables a wide application of visual SLAM in robots, autonomous driving, and augmented reality. The next generation of SLAM will require support for more intelligent tasks with a better capacity that we call “geometric and semantic Spatial AI perception” [1]. This will greatly extend the scope of traditional geometric localization and mapping.

In terms of geometric perception (*e.g.*, point-based appearance modeling and handcrafted feature-based localization),

This work was supported by National Natural Science Foundation of China (No. 61973066), Major Science and Technology Projects of Liaoning Province (No.2021JH1/10400049), Fundation of Key Laboratory of Aerospace System Simulation (No.6142002200301), Fundation of Key Laboratory of Equipment Reliability (No.WD2C20205500306) and Fundamental Research Funds for the Central Universities (N2004022).

Yanmin Wu and Xin Chen are with Faculty of Robot Science and Engineering, Northeastern University, Shenyang 110819, China (Email: wuyanminmax@gmail.com).

Yunzhou Zhang, Zhiqiang Deng, and Wenkai Sun are with College of Information Science and Engineering, Northeastern University, Shenyang 110819, China. (**Corresponding author:** Yunzhou Zhang, Email: zhangyunzhou@mail.neu.edu.cn).

Delong Zhu is with the Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong SAR, China.

Jian Zhang is with the School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School, Shenzhen 518055, China.

more visual landmarks, such as the line [2], edge [3], and plane [4], are exploited to overcome environmental and motion challenges. Omnidirectional geometric perception is achieved by multi-sensor fusion of visual, thermal, inertia, LiDAR, GNSS, and UWB [5]–[7]. In onboard applications, these versatile and robust algorithms are extensively employed. However, due to the absence of semantic cues, geometric clues alone are insufficient for intelligent robot interaction and active decision-making, such as semantic mapping, object goal navigation, and object searching. This article focuses on another aspect of the next-generation SLAM: semantic perception, aiming at representing and understanding environmental information at a semantic level, which extends beyond the basic geometric appearance and position perception.

In semantic SLAM, the semantic cues provided by deep learning technology play an essential role in various sub-components, *e.g.*, localization, mapping, loop closure, and optimization. In this work, we focus on **semantic-aided mapping and exploring multiple high-level applications based on the semantic map**. Popular semantic mapping pipelines [8]–[10] parallelize the geometric SLAM workflow and learning-based semantic segmentation, and then annotate 3D point clouds (or volume, mesh) with 2D image segmentation labels. Finally, the multi-frame segmentation results are fused with probabilistic approaches to build a global semantic map. Although, these point clouds-based semantic maps are visually appealing, they are not detailed and lack sufficient instance-specific information to assist the robot in performing fine-grained tasks. Therefore, the first insight of this article is that **a helpful semantic map for robot operation should be instance- and object-oriented**.

Object SLAM is an object-oriented branch of semantic SLAM that focus on constructing the map with objects as central entities and typically takes instance-level segmentation or object detection as the semantic network. Most studies on sparse SLAM [11], [12] associate point clouds with object landmarks and take the centroids of point clouds as the positions of objects. Other studies [13]–[16] on dense SLAM improve the mapping results by denser point clouds and more precise segmentation/detection, enabling object-level reconstruction and dense semantic representation of objects. Nonetheless, these studies focus on the accuracy of object position, while the orientation and size of objects are not investigated, which are indeed indispensable for robotic tasks like manipulation and navigation. The second point of view presented in this article is that **the object’s position, orientation, and size in the map should all be parameterized**.

Object parameterization or representation is one of the

primary missions of object SLAM. To address this problem, typical studies [17]–[19] usually include object models as a prior and the point clouds or shapes of target objects are known. The pose estimation of objects is then achieved by model retrieval and matching. The prior model is also integrated into the map and engaged in object-level bundle adjustment. Studies [20]–[23] are examples that focus on categorized object models, which only take a partial knowledge of the object, such as the structure and shape, as the prior and use requires one model to represent one category. Although the object parameters are well encoded in the prior instance model or category model, obtaining the prior knowledge is difficult and expensive. In addition, the generalization capability of these models is limited. The third observation made in this work is that **objects should be represented by general models with a high degree of generality and a low cost prior**, such as the cube, cylinder, and quadric.

To summarize, this work aims to present an object SLAM framework that generates an object-oriented map with general models, which can parameterize the position, orientation, and size of objects in the map. In addition, we further explore high-level applications based on the object-oriented map. Some previous studies [24], [25] pursued a similar objective but encountered the following challenges. 1) The data association algorithms are insufficiently robust and accurate for dealing with complex settings involving various classes and numbers of objects. 2) Object parametrization is sloppy, typically depending on strict assumptions or achieving only incomplete modeling, both of which are difficult to achieve in practice. 3) Most studies focus on creating the object or semantic map, but the application in downstream tasks is not explored, nor is the map's utility demonstrated. Instead, we discuss **not only the fundamental techniques of object mapping but also high-level and object map-oriented applications**.

In this paper, we propose an object SLAM framework to achieve the desired objective while overcoming the aforementioned challenges. Firstly, we integrate the parametric and nonparametric statistic tests and the traditional IoU-based method to conduct model ensembling for data association. Compared with conventional methods, our approach sufficiently exploits the nature of different statistics, *e.g.*, Gaussian, non-Gaussian, 2D, and 3D measurements, hence exhibiting significant advantages in association robustness. Then, for object parametrization, we offer an algorithm for centroid, size, and orientation estimation and an object pose initialization approach based on the iForest (isolation forest) and line alignment. The proposed methods are robust to outliers and exhibit high accuracy, which significantly facilitates the joint pose optimization process. Finally, an object-oriented map is constructed using the general models taking cubes and quadrics as representations. Based on the map, we develop an augmented reality system to enable virtual-real fusion and interaction, transplant a framework for the robot arm to realize common objects' modeling and grasping, and propose a novel object descriptor for sub-scene matching and relocalization.

This article extends our previous works [26], [27]. Extensions include semantic descriptor-based scene matching/relocalization (Sec. VI and Sec. VIII-F) and expanded experiments

and analysis (Sec. IX). The contributions are summarized:

- We propose an ensemble data association strategy that can effectively aggregate different measurements of the objects to improve association accuracy.
- We propose an object pose estimation framework based on the iForest and line alignment, which is robust to outliers and can accurately estimate the pose and size of objects.
- We build a lightweight and object-oriented map with general models, upon which we develop an augmented reality application aware of occlusion and collisions.
- We extend the object map to a topological map and design a semantic descriptor based on the parameterized object information to enable multiple scene matching and object-based relocalization.
- We integrate object SLAM with robotic grasping tasks to propose an object-driven active exploration strategy that accounts for object observation completeness and pose estimation uncertainty, achieving accurate object mapping and complex robotic grasping.
- We propose a comprehensive object SLAM framework that explores the key challenges and powerfully demonstrates its utility in various scenarios and tasks.

II. RELATED WORK

A. Data Association

Data association establishes the 2D-3D relationship between objects in image frames and the global map and the 2D-2D correspondence of objects between sequential frames. The most popular strategy considers it an object-tracking issue [11], [28], [29]. Li *et al.* [30] project 3D objects to the image plane and then perform association using the projected 2D bounding boxes via the Hungarian object tracking algorithm. Some approaches [16], [31]–[33] use Intersection over Union (IoU) algorithm to track objects between frames, while tracking-based approaches are prone to create erroneous priors in complicated contexts resulting in wrong association results.

Some studies increase the utilization of shared information. Liu *et al.* [34] create a descriptor representing the topological relationships between objects, and instances with the greatest number of the shared descriptors are considered identical. Instead, Yang *et al.* [24] suggest using the number of matched map points on detected objects as an association criterion. Grinvald *et al.* [15] preset a measurement of semantic label similarity, while Ok *et al.* [35] propose to leverage the hue saturation histogram correlation. Sünderhauf *et al.* [14] compare the distance between distinct instances more directly. Typically, the designed criteria are inadequately general, exhaustive, or robust, leading to incorrect associations.

In terms of learning-based studies, Xiang *et al.* [36] suggest utilizing recurrent neural networks to achieve semantic label data association between consecutive images. However, they only focus on pixel-level associations. Similarly, Li *et al.* [37] use an attention-based GNN to maintain the detected 2D and 3D attributes. Merrill *et al.* [38] propose a keypoint-based object-level SLAM system that projects the 3D key points to the image as the prior of the objects in the next frame.

However, this method is not verified on the SLAM dataset and cannot be generalized to previously unseen objects. Using a deep graph convolutional network, Xing *et al.* [39] extract object features and perform feature matching. Nevertheless, this method is only suitable for well-constructed maps and is challenging for incremental maps of real-time SLAM.

Another viable option is the probabilistic-based solution. Bowman *et al.* [20] use a probabilistic method to model the data association process and leverage the EM algorithm to identify correspondences between observed landmarks. Subsequent studies [40], [41] extend the concept to associate dynamic objects or perform dense semantic reconstructions. However, their efficiency is limited by the high cost of the EM optimizers. Weng *et al.* [13] present a nonparametric Dirichlet process for semantic data association, which can address the challenges that arise when the statistics do not follow a Gaussian distribution. Later, Zhang *et al.* [42] and Ran *et al.* [43] introduce two variations of the hierarchical Dirichlet method for lowering association uncertainty. Iqbal *et al.* [12] also demonstrate the efficiency of nonparametric data association. However, this strategy cannot properly address statistics with Gaussian distributions and is thus incapable of adequately leveraging diverse data in SLAM. We combine the parametric and nonparametric methods to execute model ensembling, which exhibits superior association performance in complex scenarios with numerous object categories.

B. Object Representation

Object representation in object SLAM can be divided into shape reconstruction-based and model-based methods. For the former category, Sucar *et al.* [23] infer object volume from images using a Variational Auto Encoder and then jointly optimize object shape and pose. Wang *et al.* [32] adopt DeepSDF [44] as shape embedding, minimizing the surface consistency and depth rendering loss by observed point clouds. Similarly, Xu *et al.* [33] train a shape completion network based on the pre-trained DeepSDF to achieve complete shape reconstruction of partially seen objects. However, these methods are data-driven and significantly dependent on large-scale shape priors.

Model-based object representations are classified broadly into three types: prior instance-level models [17]–[19], [45], category-specific models, and general models. Prior instance-level models rely on a well-established or trained database, such as detailed point clouds or CAD models. Since such models must be known in advance, their application scenarios are limited. In addition, studies [21]–[23] on category-specific models focus on identifying category-level characteristics. Parkhiya *et al.* [21] and Joshi *et al.* [22] represent different categories through the combination of line segments, but the category-specific feature is insufficiently general and is impossible to describe an excessive number of classes.

The general object models are represented by simple geometric elements, *e.g.*, cube, quadric, and cylinder, which are the most efficient models. There are two typical modeling method. The first type infers the 3D pose from the 2D detection result. Yang *et al.* [24] leverage the vanishing point to sample 3D cube proposal from a single view, and then optimize the object pose using geometric measurements.

Nicholson *et al.* [25] combine multi-view observation to parametrize object landmarks as constrained dual quadrics. Subsequent studies [35], [46] refine quadric representation by incorporating shape and semantic priors and plane constraints. However, this inference from the 2D object has a poor precision with significant errors. Li *et al.* [37] apply superquadric to tune between 3D boxes and quadrics adaptively. However, they rely on additional 3D object detection. Another type of methods resolve the 3D object pose by 3D point cloud measurements. Some studies [11]–[13] portray object position using point cloud centers, which is an imprecise way of expressing object properties. Runz *et al.* [47] get a dense object reconstruction result using more accurate instance and geometry-based segmentation. While the object's position and size are viable, the orientation is ignored. Some other studies [24], [48], [49] involving direction estimation use the geometric characteristics of images or point clouds for orientation sampling and analysis. However, they face the problem of insufficient robustness. In contrast, studies [50], [51] use learning-based methods from orientation regression from the image, but there are issues in terms of accuracy and generalization. In this work, based on the general object model, we propose an outlier-robust object pose estimation algorithm using the iForest and line alignment method for better parametrization of object size and orientation.

C. Semantic Scene Matching

Scene matching is critical for robot relocalization, loop closure, and multi-agent collaboration. Conventional studies [52], [53] rely on keyframes and geometric features, which are vulnerable to failure when faced with changes in viewpoint, illumination, and appearance. Conversely, semantic-based scene matching is more efficient because of the time and space invariance of the semantic information (*e.g.*, label and size).

Gawel *et al.* [54] focus on global scene matching for multi-view robots and they propose a random-walk-based semantic descriptor to enable global localization by semantic graph match. Guo *et al.* [55] investigate large-scale scene matching problem and suggest a semantic histogram-based fast graph matching algorithm, resulting in more accurate and faster localization and map merging. However, these methods only account for the global matching of large scenes, disregarding the local information. Additionally, the semantic information is not at the object level. Liu *et al.* [34] are interested in the issue of localization when environmental appearance changes. They suggest characterizing the scene with a dense semantic topology map and performing 6-DOF object localization by matching object descriptors. Similarly, Li *et al.* [30] focus on the relocalization of perspective changes. They use object landmarks to establish the correspondence between different views and conduct relocalization through graph matching based on the Hungarian algorithm. However, given the limited number of objects and the fact that they are not well-parameterized, their method is doubtful in a complicated setting with several repeating objects. To address the loop closure problem in multi-object scenes, Qin *et al.* [56] propose to generate semantic sub-graphs using objects' semantic labels and then leverage Kuhn–Munkres to align

sub-graphs for estimating the transformation. However, the semantic clues are only used to determine the resemblance of scenes, while the translation between them is still calculated by geometric measures instead of semantic measures. In this work, we focus on scene matching and scene translation with multiple objects. Similar to previous studies, we create a topological map and design an object descriptor. In the map, the objects are fully parameterized, and the matching strategy based on object descriptors are also improved.

D. Active Perception and Object Map-based Grasping

Active perception is the process of actively adjusting sensor states by analyzing existing data to gather more valuable information for executing specific tasks, which is a critical characteristic of robot autonomy. Zhang *et al.* [57] leverage Fisher information to predict the optimal sensor position to reduce localization uncertainty. Zeng *et al.* [58] exploit prior knowledge between objects to establish a semantic link graph for active object search. More specifically, active mapping is a specific type of active perception task concerned with autonomous map construction. Charrow *et al.* [59] utilize the quadratic mutual information to guide 3D dense mapping. Wang *et al.* [60] also leverage the mutual information to perform Next-Best View (NBV) selection on a sparse road map, which subsequently acts as a semantic landmark to aid the mapping process. Kriegel *et al.* [61] propose a surface reconstruction method for single unknown objects. In addition to the information gain, they also integrate the measurement of reconstruction quality into the objective function, achieving high accuracy and completeness. The key to active mapping is defining the measurement and strategy to guide the agent moving autonomously. We propose an information entropy-based uncertainty quantification and an object-driven active exploration strategy. Another significant difference from other methods is that the output of our suggested method is an object map compatible with complex robot manipulation tasks.

The object map encoded with object pose is available for robot object manipulation tasks, such as object placement and arrangement. Wada *et al.* [62] propose reconstructing objects by incremental object-level voxel mapping. Voxel points initialize the object pose, and the ICP algorithm is then used to align the initialized object with the CAD model to optimize the pose further, which is heavily dependent on the CAD model's registration accuracy. In NodeSLAM [23], the object is regarded as a landmark and is involved in joint optimization to help generate an accurate object map. The primary deficiency of this method is that the model requires a tedious category-level training process for each object. Labb   *et al.* [19] present a single-view 6-DoF object pose estimation method and utilize the object-level bundle adjustment in the SLAM framework to optimize the object map. However, this method only focuses on known objects. Almeida *et al.* [63] leverage the SLAM framework to densely map unknown objects for accurate grasping point detection, but the object pose is not estimated. In this work, we use the proposed SLAM framework to generate the object map actively, which enables the global perception to aid the robot in performing more

intelligent tasks autonomously. Additionally, unlike previous studies, we focus on the pose estimation of unknown objects.

III. SYSTEM OVERVIEW

The proposed object SLAM framework is demonstrated in Fig. 1 including four parts. The **tracking module** builds upon the ORB-SLAM2 [52], which generates incremental sparse point clouds and estimates camera pose by extracting and matching multi-view features. Our main contributions lie in the remaining three parts. The **semantic module** employs YOLO [64] as the object detector to provide semantic labels and bounding boxes which are then combined with point cloud measurements to associate the 2D detected objects with 3D global objects. After that, the iForest and the line alignment algorithms are applied to refine the point clouds and 2D lines generated by the tracking module. Based on the association and refinement results, the objects are parameterized using the cube and quadric models.

The **object map** comprises of multiple parameterized objects and achieves a lightweight representation of the environment, which is a vital component of the application module. For the **augmented reality** application, virtual models' 3D registration is based on the real-world object pose rather than the conventional point-based approach. Additionally, we convert the object map to a topological map, a graph representation of the objects and their relative poses. Based on this map, a semantic descriptor is designed to enable multi-scene matching and relocalization tasks.

The accurate object pose is encoded in the object map, which provides the fundamental clues (*e.g.*, grasping points) for **robotic grasping** applications. Notably, the object map is created actively, as depicted in the **exploration module**. Here, we propose an uncertainty measurement model to predict the next best view for exploration. The manipulator then actively moves to scan the table with the best view until building up a complete and accurate object map.

In short, the proposed object SLAM leverages geometric and semantic measurements to simultaneously realize camera localization and object map building, resulting in a comprehensive system that addresses various challenges in this field and facilitates many intelligent and fascinating applications. The remainder of our paper is organized as follows: IV and V present the principal theories of data association and object parameterization. The semantic descriptor and scene-matching method are defined in Section VI. The active exploration strategy is introduced in Section VII. Section VIII demonstrates the performance of our system through comprehensive experiments. Section IX provides the discussion and analysis, and Section X provides the conclusion.

IV. OBJECT-LEVEL DATA ASSOCIATION

Fig. 2 presents the pipeline of the proposed data association strategy. The **local object** is a 3D instance observed in the current single view (t), where the point clouds correspond to ORB features that lie in the 2D bounding box, and the centroid is the mean of the points. The **global object** is an entity observed by multiple frames (before t) and already exists on

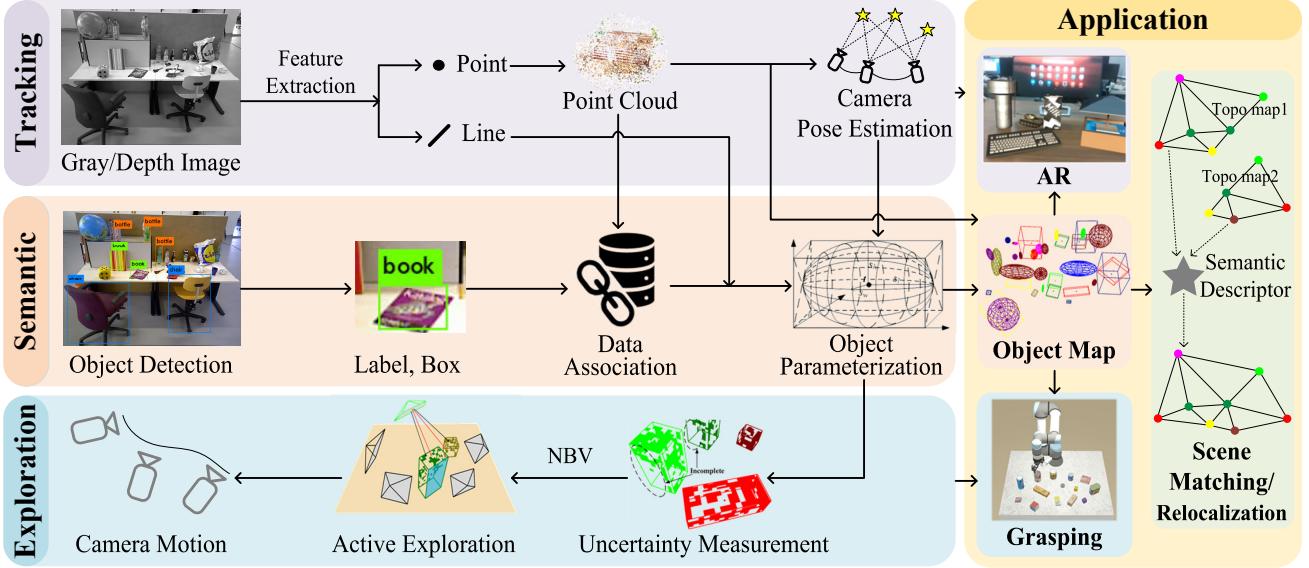


Fig. 1: The proposed object SLAM framework.

the map, where point clouds and centroids also come from incremental measurement of multi-views. Data association aims to determine which global object in the map is associated with the local object in the current view. As shown in the pipeline, the camera motion IoU (M-IoU), nonparametric (NP) test, single sample- t (S- t) test, and project IoU (P-IoU) will be used to determine whether or not the association is successful. In the experiment, the successful case should satisfy the fourth item and any of the first three items. If so, the existing global object will be updated; otherwise, a new global object will be created. Finally, the double sample- t (D- t) test is utilized to check whether duplicates exist.

Throughout this section, the following notations are used:

- $P \in \mathbb{R}^{3 \times |P|}, Q \in \mathbb{R}^{3 \times |Q|}$ - the point clouds of the local object and the global object.
- \mathcal{R} - the rank (position) of a data point in a sorted list.
- $\mathbf{c} \in \mathbb{R}^{3 \times 1}$ - the currently observed local object centroid.
- $C = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_{|C|}] \in \mathbb{R}^{3 \times |C|}$ - a series of centroids of a global object observed by historical views. C_1, C_2 are similar.
- $f(\cdot)$ - the probability function used for statistic test.
- $m(\cdot), \sigma(\cdot) \in \mathbb{R}^{3 \times 1}$ - the mean and variance functions.

A. Intersection over Union (IoU) Model

If a global object is observed in the previous two frames ($t-1$ and $t-2$), we then predict the bounding box in the current frame (t) based on the hypothesis of uniform motion, and calculate the IoU between the predicted box of a global object and the detected box of a local object, which we defined as Motion-IoU (See Fig. 2 M-IoU part). If the IoU value is large enough, there may be a potential association between the two objects. After NP and S- t (see Sections IV-B and IV-C), the Project-IoU will validate this association by projecting 3D point clouds of the global object to 2D points on the current frame and fitting a box to these points. After that, we calculate the IoU between the projected and detected boxes (See Fig. Fig. 2 P-IoU part).

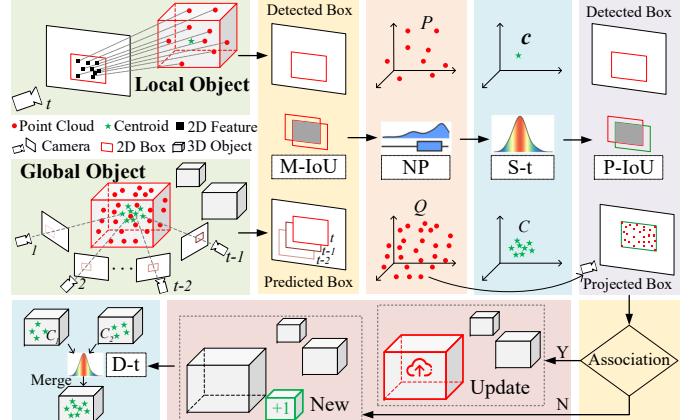


Fig. 2: The pipeline of object-level data association.

B. Nonparametric Test Model

The m-IoU model provides a straightforward and efficient way of dealing with the scenario of consecutive frames. However, it will malfunction when 1) the object is missed by the detector, 2) the object is occluded, or 3) the object disappears from the camera view.

The Nonparametric test model does not require continuous observations of the object, and can be directly applied to process two sets of point clouds, P and Q (see Fig. 2 NP part), based on the hypothesis that point clouds follow a non-Gaussian distribution (which will be demonstrated in Section VIII-A). Theoretically, if P and Q represent the same object, they should follow the same distribution, i.e., $f_P = f_Q$. We use the Wilcoxon Rank-Sum test [65] to verify whether the null hypothesis holds.

We first mix the two point clouds $X = [P|Q] = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{|X|}] \in \mathbb{R}^{3 \times (|P|+|Q|)}$, and then sort X in three dimensions respectively. Define $W_P \in \mathbb{R}^{3 \times 1}$ as follows,

$$W_P = \left\{ \sum_{k=1}^{|X|} \mathcal{R}(1\{\mathbf{x}_k \in P\}) - \frac{|P|(|P|+1)}{2} \right\}, \quad (1)$$

and W_Q is with the same formula. The Mann-Whitney statistics is $W = \min(W_P, W_Q)$, which is proved to follow a Gaussian distribution asymptotically [66], [67]. Herein, we essentially construct a Gaussian statistics using the non-Gaussian point clouds. The mean and variance of W are calculated:

$$m(W) = (|P||Q|)/2, \quad (2)$$

$$\sigma(W) = \frac{|P||Q|\Delta^+}{12} - \frac{|P||Q|(\sum_i \tau_i^3 - \sum_i \tau_i)}{12(|P| + |Q|)\Delta^-}, \quad (3)$$

where $\Delta^+ = |P| + |Q| + 1$, $\Delta^- = |P| + |Q| - 1$, and $\tau \in P \cap Q$. τ represents the number of shared points between two objects; because its value is small, the complicated and low-contributing second term in Eq. (3) is ignored in our implementation.

To make the null hypothesis stand, W should meet the following constraints:

$$f(W) \geq f(r_r) = f(r_l) = \alpha/2, \quad (4)$$

where α is the significance level, $1 - \alpha$ is the confidence level, and $[r_l, r_r] \approx [m - s\sqrt{\sigma}, m + s\sqrt{\sigma}]$ defines the confidence region. The scalar $s > 0$ is defined on a normalized Gaussian distribution $\mathcal{N}(s|0, 1) = \alpha$. In summary, if the Mann-Whitney statistics W of two point clouds P and Q satisfies Eq. (4), we temporarily assume they come from the same object.

C. Single-sample and Double-sample T-test Model

The single-sample t -test is used to process object centroids observed in different views (see Fig. 2 S-t part), which typically follow a Gaussian distribution (see Section VIII-A).

Suppose the null hypothesis is that C and c are from the same object, and define t statistics as follows,

$$t = \frac{m(C) - c}{\sigma(C)/\sqrt{|C|}} \sim t(|C| - 1). \quad (5)$$

For the null hypothesis to hold, t should satisfy:

$$f(t) \geq f(t_{\alpha/2,v}) = \alpha/2, \quad (6)$$

where $t_{\alpha/2,v}$ is the upper $\alpha/2$ quantile of the t -distribution of v degrees of freedom, and $v = |C| - 1$. If t statistics satisfy (6), we temporarily assume c and C come from the same object.

Some existing objects may be misidentified as new due to the above-described strict data association strategy, poor observation views, or erroneous object detection, resulting in duplicates. Consequently, a double-sample t -test is leveraged to determine whether to merge the two objects by analyzing their historical centroids (see Fig. 2 D-t part).

Construct t -statistics for C_1 and C_2 as follows,

$$t = \frac{m(C_1) - m(C_2)}{\sigma_d} \sim t(|C_1| + |C_2| - 2), \quad (7)$$

$$\sigma_d = \sqrt{\frac{(|C_1| - 1)\sigma_1^2 + (|C_2| - 1)\sigma_2^2}{|C_1| + |C_2| - 2} \left(\frac{1}{|C_1|} + \frac{1}{|C_2|} \right)}, \quad (8)$$

where σ_d is the pooled standard deviation of the two objects. Similarly, if t satisfies (6), $v = |C_1| + |C_2| - 2$, it means that C_1 and C_2 belong to the same object, then we merge them.

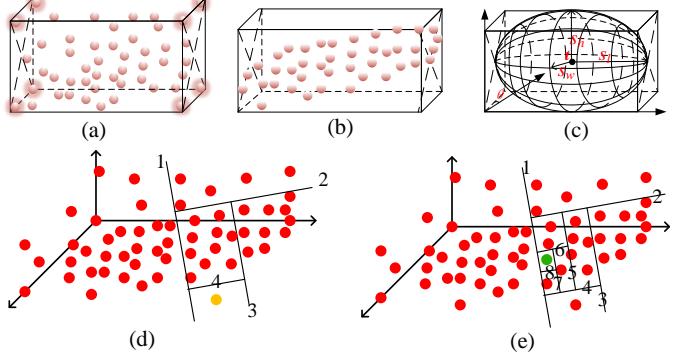


Fig. 3: (a-c) Demonstration of object parameterization. (d-e) Demonstration of iForest.

V. OBJECT PARAMETERIZATION

Data association provides the global object with multi-view measurements that ensure more observations for parameterization to model objects effectively. Throughout this section, the following notations are used:

- $\mathbf{t} = [t_x, t_y, t_z]^T$ - the translation (location) of object frame in world frame.
- $\boldsymbol{\theta} = [\theta_r, \theta_y, \theta_p]^T$ - the rotation of object frame w.r.t. world frame. $R(\boldsymbol{\theta})$ is matrix representation.
- $T = \{R(\boldsymbol{\theta}), \mathbf{t}\}$ - the transformation of object frame w.r.t. world frame.
- $\mathbf{s} = [s_l, s_w, s_h]^T$ - half of the side length of a 3D bounding box, i.e., the scale of an object.
- $P_o, P_w \in \mathbb{R}^{3 \times 8}$ - the coordinates of eight vertices of a cube in object and world frame, respectively.
- $Q_o, Q_w \in \mathbb{R}^{4 \times 4}$ - the quadric parameterized by its semiaxis in object and world frame, respectively, where $Q_o = \text{diag}\{s_l^2, s_w^2, s_h^2, -1\}$.
- $\alpha(\cdot)$ - calculate the angle of line segments in the image.
- K, T_c - the intrinsic and extrinsic parameters of camera.
- $\mathbf{p} \in \mathbb{R}^{3 \times 1}$ - the coordinates of a point in world frame.

A. Object Representation

In this work, we leverage the cubes and quadrics/cylinders to represent objects, rather than the complex instance-level or category-level model. For objects with regular shapes, such as the book, keyboard, and chair, we use cubes (encoded by their vertices P_o) to represent them. For non-regular objects without an explicit direction, such as the ball, bottle, and cup, the quadric/cylinder (encoded by its semiaxis Q_o) is used for representation, and its orientation parameter is ignored. Here, P_o and Q_o are expressed in the object frame and only depend on the scale s . To register these elements to the global map, we also need to estimate their translation \mathbf{t} and orientation $\boldsymbol{\theta}$ w.r.t. the global frame. Cubes and quadrics in the global frame are expressed as follows:

$$P_w = R(\boldsymbol{\theta})P_o + \mathbf{t}, \quad (9)$$

$$Q_w = TQ_oT^T. \quad (10)$$

These two models can be switched conveniently, as shown in fig.3(c). Assuming that objects are placed parallel with the ground, as in other works [68], [69], i.e., $\theta_r = \theta_p = 0$, we only need to estimate $[\theta_y, \mathbf{t}, \mathbf{s}]$ for a cube and $[\mathbf{t}, \mathbf{s}]$ for a quadric.

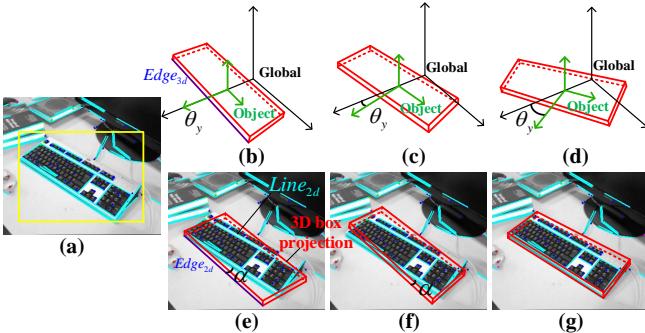


Fig. 4: Line alignment to initialize object orientation. (a) Object and line detection in 2D image. (b-d) Angle sampling in 3D space; (e-g) Projection of angle sampling process in 2D images.

B. Estimation of translation(t) and scale(s)

Assuming that the object point clouds X are in the global frame, we follow conventions and denote its mean by t , based on which the scale can be calculated by $s = (\max(X) - \min(X))/2$, as shown in Fig. 3(a). The main challenge here is that X is typical with many outliers, which will introduce a substantial bias to t and s . One of our major contributions in this paper is the development of an outlier-robust centroid and scale estimation algorithm based on the iForest [70] to improve the estimation accuracy. The detailed procedure of our algorithm is presented in Alg. 1.

The key idea of the algorithm is to recursively separate the data space into a series of isolated data points, and then take the easily isolated ones as outliers. The philosophy is that, normal points are typically located more closely and thus need more steps to isolate, while the outliers usually scatter sparsely and can be easily isolated with fewer steps. As indicated by the algorithm, we first create t isolated trees (the iForest) using the point cloud of an object (lines 2 and 14-33), and then identify the outliers by counting the path length of each point $\mathbf{x} \in X$ (lines 3-9), in which the score function is defined as follows:

$$s(\mathbf{x}) = 2 \exp \frac{-E(h(\mathbf{x}))}{C}, \quad (11)$$

$$C = 2H(|X| - 1) - \frac{2(|X| - 1)}{|X|}, \quad (12)$$

where C is a normalization parameter, H is a harmonic number $H(i) = \ln(i) + 0.5772156649$, $h(\mathbf{x})$ is the height of point \mathbf{x} in the isolated tree, and E is the operation to calculate the average height. As demonstrated in Fig. 3(d)-(e), the yellow point is isolated after four steps; hence its path length is 4, whereas the green point has a path length of 8. Therefore, the yellow point is more likely to be an outlier. In our implementation, points with a score greater than 0.6 are removed and the remaining are used to calculate t and s (lines 10-12). Based on s , we can initially construct the cubics and quadratics in the object frame, as shown in Fig. 3(a)-(c).

C. Estimation of orientation(θ_y)

The estimation of θ_y is divided into two steps, namely, to find a good initial value for θ_y first and then conduct numerical optimization based on the initial value. Since pose estimation is a non-linear process, a good initialization is very important to help improve the optimality of the estimation

Algorithm 1 Centroid and Scale Estimation Based on iForest

Input: X - The point cloud of an object, t - The number of iTrees in iForest, ψ - The subsampling size for an iTree.

Output: \mathcal{F} - The iForest, a set of iTrees, t - The origin of local frame, s - The initial scale of the object.

```

1: procedure PARAOBJECT( $X, t, \psi$ )
2:    $\mathcal{F} \leftarrow \text{BUILDFOREST}(X, t, \psi)$ 
3:   for point  $\mathbf{x}$  in  $X$  do
4:      $E(h) \leftarrow \text{averageDepth}(\mathbf{x}, \mathcal{F})$ 
5:      $s \leftarrow \text{score}(E(h), C)$             $\triangleright$  Eq. (11) and (12)
6:     if  $s > 0.6$  then              $\triangleright$  an empirical value
7:       remove( $\mathbf{x}$ )                  $\triangleright$  remove  $\mathbf{x}$  from  $X$ 
8:     end if
9:   end for
10:   $t \leftarrow \text{meanValue}(X)$ 
11:   $s \leftarrow (\max(X) - \min(X)) / 2$ 
12:  return  $\mathcal{F}, t, s$ 
13: end procedure

14: procedure BUILDFOREST( $X, t, \psi$ )
15:    $\mathcal{F} \leftarrow \phi$ 
16:    $l \leftarrow \text{ceiling}(\log_2 \psi)$      $\triangleright$  maximum times of iterations
17:   for  $i = 1$  to  $t$  do
18:      $X^{(i)} \leftarrow \text{randomSample}(X, \psi)$ 
19:      $\mathcal{F} \leftarrow \mathcal{F} \cup \text{BUILDTREE}(X^{(i)}, 0, l)$ 
20:   end for
21:   return  $\mathcal{F}$ 
22: end procedure

23: procedure BUILDTREE( $X, e, l$ )
24:   if  $e \geq l$  or  $|X| \leq 1$  then
25:     return exNode $\{\|X\|\}$            $\triangleright$  record the size of  $X$ 
26:   end if
27:    $i \leftarrow \text{randomDim}(1, 3)$          $\triangleright$  get one dimension
28:    $q \leftarrow \text{randomSplitPoint}(X[i])$ 
29:    $X_l, X_r \leftarrow \text{split}(X[i], q)$ 
30:    $L \leftarrow \text{BUILDTREE}(X_l, e + 1, l)$      $\triangleright$  get child pointer
31:    $R \leftarrow \text{BUILDTREE}(X_r, e + 1, l)$ 
32:   return inNode $\{L, R, i, q\}$ 
33: end procedure
```

result. Conventional methods [30], [47] usually neglect the initialization process, which typically yields inaccurate results.

The detail of orientation initialization algorithm is presented in Alg. 2. The inputs are obtained as follows: 1) LSD (Line Segment Detector [71]) segments are extracted from t consecutive images, and those falling in the bounding boxes are assigned to the corresponding objects (see Fig. 4(a)); 2) The initial pose of an object is assumed to be consistent with the global frame, *i.e.*, $\theta_0=0$ (see Fig. 4b). In the algorithm, we first uniformly sample thirty angles within $[-\pi/2, \pi/2]$ (line 2). For each sample, we then evaluate its score by calculating the accumulated angle errors between LSD segments Z_{lsd} and the projected 2D edges of 3D edges Z of the cube (lines 3-12). The error is defined as follows:

$$\begin{aligned} e(\boldsymbol{\theta}) &= \|\alpha(\hat{Z}(\boldsymbol{\theta})) - \alpha(Z_{lsd})\|^2, \\ \hat{Z}(\boldsymbol{\theta}) &= KT_c(R(\boldsymbol{\theta})Z + \mathbf{t}). \end{aligned} \quad (13)$$

Algorithm 2 Initialization for Object Pose Estimation

Input: Z_1, Z_2, \dots, Z_t - Line segments detected by LSD in t consecutive images, θ_0 - The initial guess of yaw angle.

Output: θ - The estimation result of yaw angle, e - The estimation errors.

```

1:  $\mathcal{S}, \mathcal{E} \leftarrow \phi$ 
2:  $\Theta \leftarrow \text{sampleAngles}(\theta_0, 30)$             $\triangleright$  see Fig. 4 (b)-(d)
3: for sample  $\theta$  in  $\Theta$  do
4:    $s_\theta, e_\theta \leftarrow 0$ 
5:   for  $Z$  in  $\{Z_1, Z_2, \dots, Z_t\}$  do
6:      $s, e \leftarrow \text{score}(\theta, Z)$             $\triangleright$  Eq. (13) and (14)
7:      $s_\theta \leftarrow s_\theta + s$ 
8:      $e_\theta \leftarrow e_\theta + e$ 
9:   end for
10:   $\mathcal{S} \leftarrow \mathcal{S} \cup \{s_\theta\}$ 
11:   $\mathcal{E} \leftarrow \mathcal{E} \cup \{e_\theta\}$ 
12: end for
13:  $\theta^* \leftarrow \text{argmax}(\mathcal{S})$ 
14: return  $\theta^*, e_{\theta^*}$ 
```

The demonstration of the calculation of $e(\theta)$ is visualized in Fig. 4(e)-(g). The score function is defined as follows:

$$\text{Score} = \frac{N_p}{N_a} (1 + 0.1(\xi - E(e))), \quad (14)$$

where N_a is the total number of line segments of the object in the current frame, N_p is the number of line segments that satisfy $e < \xi$, ξ is a manually defined error threshold (five degrees here), and $E(e)$ is the average error of these line segments with $e < \xi$. After evaluating all the samples, we choose the one that achieves the highest score as the initial yaw angle for optimization (line 13).

D. Object pose optimization

After obtaining the initial s and θ_y , we then jointly optimize object and camera poses:

$$\{O, T_c\}^* = \underset{\{\theta_y, s\}}{\text{argmin}} \sum (e(\theta) + e(s)) + \underset{\{T_c\}}{\text{argmin}} \sum e(p), \quad (15)$$

where the first term is the object pose error defined in Eq. (13) and the scale error $e(s)$ is defined as the distance between the projected edges of a cube and their nearest parallel LSD segments. The second term $e(p)$ is the commonly-used reprojection error in the traditional SLAM framework.

VI. OBJECT DESCRIPTOR ON THE TOPOLOGICAL MAP

After the step of object parameterization, we obtain the label, size, and pose information of a single object. To present the relationship between objects and that between objects and the scene, we create a topological map. The map is then used to generate an object descriptor for scene matching.

A. Semantic Topological Map

The topological map is an abstract representation of the scene. In this work, to construct the semantic topological map, the 3D object centroid is used to represent the node N that

encodes the semantic label l and the object parameters t, θ, s . Then, under the distance and number constraints, we generate the undirected edge E between objects, which includes the distance d and angle α of two objects:

$$N = \langle l, t, \theta, s \rangle, E = \langle d, \alpha \rangle. \quad (16)$$

Fig. 5(a) presents a real-world scene with multiple objects. Fig. 5(b) shows the object modeling result by the method of Section V, which is then used to create a semantic topological map (Fig. 5(c)) that expresses the scene in an abstract way and shows the connection relationship between objects as symbolized in Eq. (16).

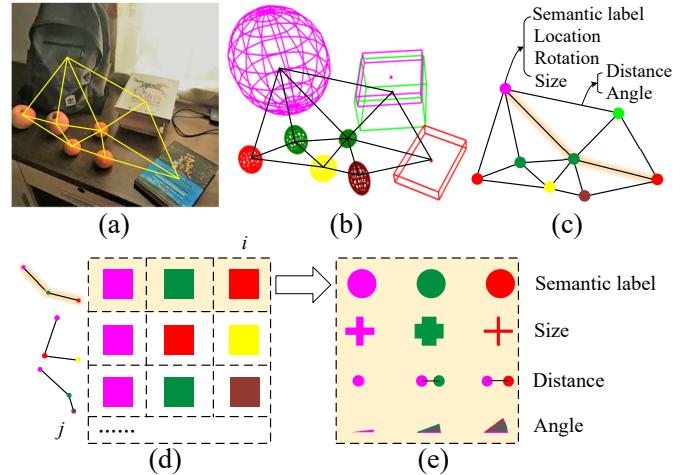


Fig. 5: (a) Real-world scene. (b) Object-level map. (c) Semantic topological map. (d) Random walk descriptor. (e) 3D matrix visualization of a single descriptor.

B. Semantic Descriptor

Since the object information, including semantic label, position, and scale, is not unique, the computation for undirected graph matching, an NP problem [72], is extremely high. To reduce the computational complexity and enhance the matching accuracy, we introduce a random-walk descriptor that weights multi-neighborhood measurements to describe an object, improving object uniqueness and the relationship with the scene.

The random-walk descriptor is represented by a 2D matrix, as shown in Fig. 5(d), with each row storing a walking route that starts at the described object, and randomly points to the next object. It is worth noting that each object only appears once in a route and the process ends when reaching a certain depth i or time j limit.

The previous work [54] only considers the semantic label $l = (l_1, l_2, \dots, l_i)$ as the descriptor. Benefiting from the above accurate object parameterization, we add three additional measurements, object size $s = (s_1, s_2, \dots, s_i)$, distance $d = (d_{11}, d_{12}, \dots, d_{1i})$, and angle $\alpha = (\alpha_{11}, \alpha_{12}, \dots, \alpha_{1i})$, to improve the robustness of the descriptor. As shown in Fig. 5(e), thus we transfer the random-walk descriptor to a 3D matrix form:

$$v = (r_1, r_2, \dots, r_j)^T, r_j = (l, s, d, \alpha)^T. \quad (17)$$

Algorithm 3 Scene matching based on object descriptor

Input: T_1, T_2 - Two sub-topo maps, i, j - threshold of depth and number of random-walk.
Output: \mathbb{T} - Transformation between two maps.

```

1: procedure POSESOLVE( $T_1, T_2, i, j$ )
2:    $\mathcal{V}_1, \mathcal{V}_2, \mathcal{M} \leftarrow \phi$ 
3:    $\mathcal{V}_1 \leftarrow \text{OBJECTDESCRIPTOR}(T_1, i, j)$ 
4:    $\mathcal{V}_2 \leftarrow \text{OBJECTDESCRIPTOR}(T_2, i, j)$ 
5:   for object  $v_1$  in  $\mathcal{V}_1$  do
6:      $\mathcal{M} \leftarrow \mathcal{M} \cup \text{MATCH}(v_1, \mathcal{V}_2)$ 
7:   end for
8:   return  $\mathbb{T} \leftarrow \text{SVD}(\mathcal{M})$ 
9: end procedure

10: procedure OBJECTDESCRIPTOR( $T, i, j$ )
11:   for object  $o$  in  $T$  do
12:      $v.\text{row} \leftarrow \text{random-walk from } o \text{ to the } i_{th} \text{ object}$ 
13:      $v.\text{col} \leftarrow \text{repeat random-walk } j \text{ times}$ 
14:      $\mathcal{V} \leftarrow \mathcal{V} \cup v$             $\triangleright$  Eq. 17 and Fig. 5(d,e)
15:   end for
16:   return  $\mathcal{V}$ 
17: end procedure

18: procedure MATCH( $v_1, \mathcal{V}_2$ )
19:    $v_2 \leftarrow \maxScore(v_1, \mathcal{V}_2)$ 
20:   return  $(v_1, v_2)$ 
21: end procedure
```

In our implementation, the additional measurement does not increase the computation. Instead, it accelerates the matching process by eliminating irrelevant candidates with more clues, such as label and size.

Alg. 3 describes the procedure for scene matching. Firstly, each object's semantic descriptor is generated in two independent sub-topological maps (lines 3-4, 10-17). Then find the best matching object-pair by scoring the similarity of each element (l, s, d, α) (lines 5-7, 18-21). Finally, the transformation between two scenes is solved by singular value decomposition (SVD) according to the multiple object pairs (line 8).

There are some points worth mentioning: **1) Scale ambiguity**: Two maps are initialized with different depths resulting in distinct scales. While object size, like Li *et al.* [50], provides a scale by length, width, and height, it is insufficiently robust. Instead, we find the matched object pair between two maps, then calculate the scale factor by averaging the ratio of the distance d . **2) Anomalous object**: The mismatch resulting from the error object or novel object may cause a considerable inaccuracy in the resolution of the translation; therefore, the RANSAC algorithm is used to eliminate the disturbance caused by anomalous objects.

VII. OBJECT-DRIVEN ACTIVE EXPLORATION

Object parameterization is good for quantifying the incompleteness of the object or map, and the incompleteness provides a driving force for active exploration. We consider the robotic grasping scene as an example. As shown in Fig. 6, the robot arm is fitted with a camera, the motion module controls the robot to execute observation commands. The

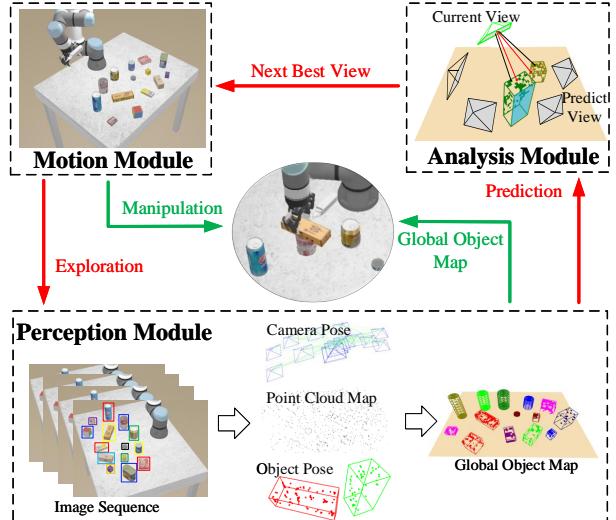


Fig. 6: The active mapping framework.

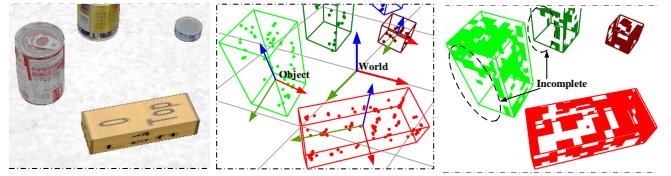


Fig. 7: Illustration of observation completeness measurement. Left: Raw image. Center: Objects with point cloud. Right: Objects with surface grids.

perception module parametrized the object map by Section IV and Section V. The analysis module measures object uncertainty and predicts different camera views' information gains. The view with the greatest information gain is selected as the Next Best View (NBV) and passed to the motion module to enable active exploration. We aim to incrementally build a global object map with the minimum effort and the maximum accuracy for robotic grasping.

A. Observation Completeness Measurement

We focus on active map building and regard the incompleteness of the map as a motivating factor for active exploration. Existing studies usually take the entire environment as the exploration target [59], [73] or focus on reconstructing a single object [61], [74], neither of which is ideal for building the object map required by robotic grasping. The reasons are as follows: 1) The insignificant environmental regions will interfere with the decisions made for exploration and misguide the robot into the non-object area; 2) it will significantly increase the computational cost and thus reduce the efficiency of the whole system. We propose an object-driven active exploration strategy for building the object map incrementally. The strategy is designed based on the observation completeness of the object, which is defined as follows.

As demonstrated in Fig. 7, the point clouds of an object are translated from the world frame to the object frame and then projected onto the five surfaces of the estimated 3D cube. Here, the bottom face is not considered. Each of the five surfaces is discretized into a surface occupancy grid map [75] with cell size $m * m$ ($m = 1\text{cm}$ in our implementation). Each grid cell can be in one of three states:

- **unknown:** the grid is not observed by the camera;
- **occupied:** the grid is occupied by the point clouds;
- **free:** the grid can be seen by the camera but is not occupied by the point clouds.

We use information entropy [76] to determine the completeness of observations based on the occupancy grid map, as information entropy has the property of symptomatizing uncertainty. The entropy of each grid cell is defined by a binary entropy function:

$$H_{grid}(p) = -p \log(p) - (1-p) \log(1-p), \quad (18)$$

where p is the probability of a grid cell being occupied and its initial value before exploration is set to 0.5. The total entropy is therefore defined as

$$H_{obj} = \sum_{o \in \mathbb{O}} H_o + \sum_{f \in \mathbb{F}} H_f + \sum_{u \in \mathbb{U}} H_u, \quad (19)$$

and the normalized total entropy is

$$\bar{H}_{obj} = H_{obj}/(|\mathbb{O}| + |\mathbb{F}| + |\mathbb{U}|), \quad (20)$$

where H_o, H_f, H_u are the entropy of occupied, free, and unknown grids, $\mathbb{O}, \mathbb{F}, \mathbb{U}$ are sets of the occupied, free, and unknown grid cells, respectively. $|\mathbb{X}|$ represents the size of \mathbb{X} . As objects continue to be explored, the number of unknown grid cells is gradually reduced, making all grids' normalized entropy \bar{H}_{grid} a smaller value. The lower the \bar{H}_{grid} is, the higher the observation completeness is. The exploration objective is to minimize \bar{H}_{grid} .

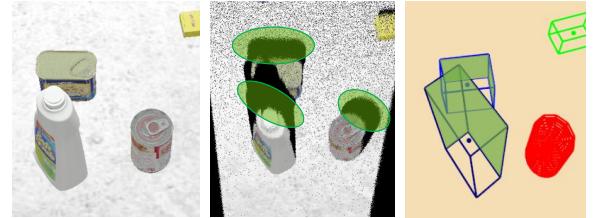
B. Object-Driven Exploration

Information Gain Definition: As illustrated in Fig. 8(b), object-driven exploration aims to predict the information gain of different candidate camera views and then select the one to explore that maximizes the information gain, *i.e.*, the NBV. The information in this work is defined as the uncertainty of the map, as mentioned in Section VII-A. The information gain is thus defined as the measurement of uncertainty reduction and accuracy improvement after the camera is placed at a specific pose. Conventionally, information gain is defined based on the area of unknown regions of the environment, *e.g.*, the black holes in the medium subfigure of Fig. 8(a), which may mislead the object map building. Compared with the conventional one, our proposed information gain is built on the observation completeness measurement of the object, shown in the right subfigure of Fig. 8(a), and incorporates the influence on object pose estimation.

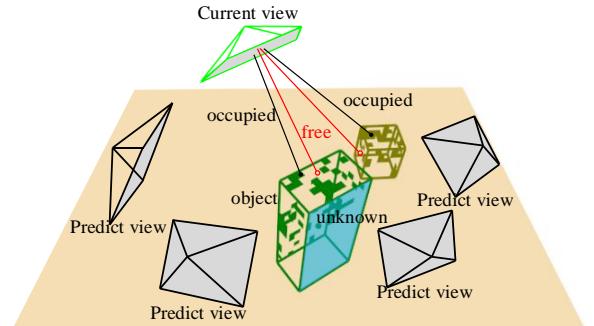
Information Gain Modeling: As indicated by the definition, information gain is contingent on many factors; thus, we create a utility function to model the information gain by manually designing a feature vector to parameterize those factors. The following is the design of the feature vector used to characterize the object \mathbf{x} ,

$$\mathbf{x} = (H_{obj}, \bar{H}_{obj}, R_o, R_{IoU}, \bar{V}_{obj}, s), \quad (21)$$

where H_{obj} , and \bar{H}_{obj} are defined by Eq. (18) - (20), R_o is the ratio of occupied grids to the total grids of the object, which indicates the richness of its surface texture, R_{IoU} is the 2D mean IoU with adjacent objects used for modeling occlusion



(a) Different definitions of information gain in exploration.



(b) Information gain under different camera views.

Fig. 8: Demonstration of the object-driven exploration.

under a specific camera view, \bar{V}_{obj} is the current volume of the object, and s is a binary value used for indicating whether the object is fully explored.

The utility function for NBV selection then is defined as:

$$f = \sum_{\mathbf{x} \in I} ((1 - R_o)H_{obj} + \lambda(H_{IoU} + H_V)) s(\mathbf{x}), \quad (22)$$

where I is the predicted camera view, λ is a weight coefficient ($\lambda = 0.2$ in our implementation), and H_{IoU}, H_V share the same formula,

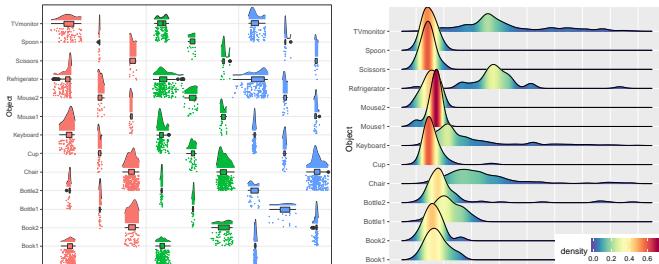
$$H = -p \log(p). \quad (23)$$

The first item $\sum_{\mathbf{x} \in I} (1 - R_o)H_{obj}$ in Eq. (22) is used to model the total weighted uncertainty of the object map under the predicted camera view. Here we give more weight to the unknown grids and the free ones by using $1 - R_o$. The reason is to encourage more explorations in free regions to find more image features that are neglected by previous sensing.

The second item $\sum_{\mathbf{x} \in I} H_{IoU}$ in Eq. (22) defines the uncertainty of object detection, which is one of the critical factors affecting object pose estimation. The uncertainty is essentially caused by occlusions between objects. We use this item to encourage a complete observation of the object. The variable in Eq. (23) is the rescaled 2D IoU, *i.e.*, $p = R_{IoU}/2$.

The third item $\sum_{\mathbf{x} \in I} H_V$ in Eq. (22) models the uncertainty of object pose estimation. Under different camera views, the estimated object poses are usually different and induce the changes in object volume. Here, we first fit a standard normal distribution using the normalized history volumes $\{\bar{V}_{obj}^{(0)}, \bar{V}_{obj}^{(1)}, \dots, \bar{V}_{obj}^{(t)}\}$ of each object, and then take the probability density of $\bar{V}_{obj}^{(t)}$ as the value p in Eq. (23). This item essentially encourages the camera view that can converge the pose estimation process.

The $s(\mathbf{x})$ in Eq. (22) indicates whether the object should be considered during the calculation of the utility function. Set $s(\mathbf{x})=0$, if the following condition is satisfied: $(\bar{H}_{grid} <$



(a) Position distribution of point clouds in three directions. (b) Distance error distribution of centroids.

Fig. 9: Distributions of different statistics in data association.

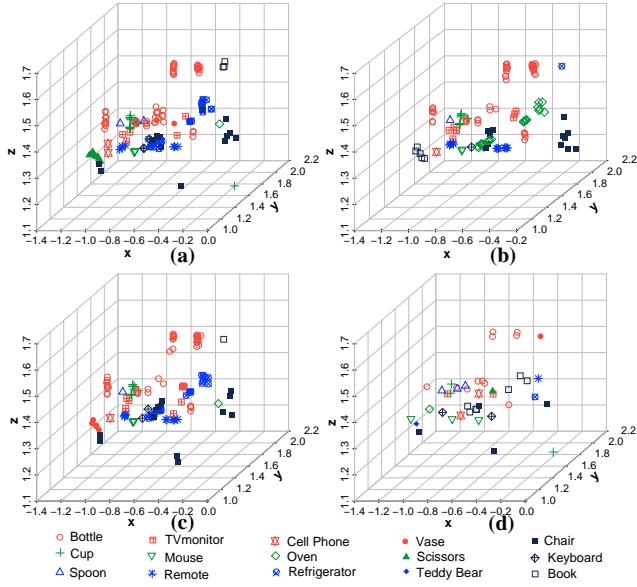


Fig. 10: Qualitative comparison of data association results. (a) IoU method. (b) IoU and nonparametric test. (c) IoU and t-test. (d) our ensemble method.

$0.5 \vee R_o > 0.5) \wedge p(\bar{V}_{obj}^{(t)}) > 0.8$. If this condition holds for all the objects, or the maximum tries are achieved (10 in this work), the exploration will be finished.

Based on the utility function, the NBV that maximizes f is continuously selected and leveraged to guide the exploration process, during which the global object map is also incrementally constructed, as depicted in Fig. 6.

VIII. EXPERIMENT

The experiment will demonstrate the performance of essential techniques such as data association, object parameterization, and active exploration. In addition, the proposed object SLAM framework will be evaluated by various applications, such as object mapping, augmented reality, scene matching, relocalization, and robotic grasping, .

A. Distributions of Different Statistics

For data association, the adopted 3D statistics for statistical testing include the point clouds and their centroids of an object. To verify our hypothesis about the distributions of different statistics, we analyze a large amount of data and visualize their distributions in Fig. 9.

Fig. 9 (a) shows the distributions of the point clouds from 13 objects during the data association in the TUM RGB-D

TABLE I: DATA ASSOCIATION RESULTS

	IoU	IoU+NP	IoU+t-test	Ours	GT
Fr1_desk	62	47	41	14	16
Fr2_desk	83	64	52	22	25
Fr3_office	150	128	130	42	45
Fr3_teddy	32	17	21	6	7

fr3_long_office sequence [77]. Obviously, these statistics do not follow a Gaussian distribution. The distributions are related to specific characteristics of the objects, and do not show consistent behaviors. Fig. 9 (b) shows the error distribution of object centroids, which typically follow the Gaussian distribution. This error is computed between the centroids of objects detected in each frame and the object centroid in the final, well-constructed map. This result verifies the reasonability of applying the nonparametric *Wilcoxon Rank-Sum test* for point clouds and the t-test for object centroids.

B. Object-level Data Association Experiments

We compare our method with the commonly-used Intersection over Union (IoU) method, nonparametric test (NP), and t-test. Fig. 10 shows the association results of these methods in the TUM RGB-D fr3_long_office sequence. It can be seen that some objects are not correctly associated in (a)-(c). Due to the lack of association information, existing objects are often misrecognized as new ones by these methods once the objects are occluded or disappear in some frames, resulting in many unassociated objects in the map. In contrast, our method is much more robust and can effectively address this problem (see Fig. 10(d)). The results of other sequences are shown in Table I, and we use the same evaluation metric as [12], [78], which measures the number of objects that are finally present in the map. The *GT* represents the ground-truth object number. As we can see, our method achieves a high success rate of association, and the number of objects in the map goes closer to GT, which significantly demonstrates the effectiveness of the proposed method.

The results of our comparison with [12], [78], which is based on the nonparametric test, are reported in II. As indicated, our method can significantly outperform [12], [78]. Especially in the TUM dataset, the number of successfully associated objects by our method is almost twice that by [12], [78]. The advantage in Microsoft RGBD [79] and Scenes V2 [80] is not apparent since the number of objects is limited. Reasons for the inaccurate association of [12], [78] lie in two folds: 1) The method does not exploit different statistics and only uses non-parametric statistics, thus resulting in many unassociated objects; 2) A clustering algorithm is leveraged to tackle the abovementioned problem, but it removes most of the candidate objects.

C. Qualitative Assessment of Object Parameterization

To demonstrate the accuracy of object parameterization, We superimpose the cubes and quadrics of objects on semi-dense maps for qualitative evaluation. Fig. 11 is the 3D top view of a keyboard (Fig. 4(a)) where the cube characterizes its pose. Fig. 11(a) is the initial pose with large-scale error; Fig. 11(b) is the result after using iForest; Fig. 11(c) is the final pose after our joint pose estimation. Fig. 12 presents the pose estimation

TABLE II: QUANTITATIVELY ANALYZED DATA ASSOCIATIONS

Seq	TUM				Microsoft RGBD					Scenes V2				
	fr1_desk	fr2_desk	fr3_long_office	fr3_teddy	Chess	Fire	Office	Pumpkin	Heads	01	07	10	13	14
[12], [78]	-	11	15	2	5	4	10	4	-	5	-	6	3	4
Ours	14	22	42	6	13	6	21	6	15	7	7	7	3	5
GT	16	23	45	7	16	6	27	6	18	8	7	7	3	6

results of the objects in 14 sequences of the three datasets, in which the objects are placed randomly and in different directions. As is shown, the proposed method achieves promising results with a monocular camera, which demonstrates the effectiveness of our pose estimation algorithm.

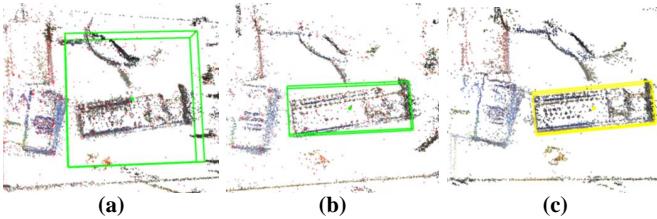


Fig. 11: Visualization of the pose estimation. (a): Initial object pose and size. (b): Object pose and size after iForest. (c): object pose and size after iForest and line alignment.



Fig. 12: Results of object pose estimation. Odd columns: original RGB images. Even column: estimated object poses.

D. Object-Oriented Map Building

Then, we build the object-oriented semantic maps based on the robust data association algorithm, the accurate object pose estimation algorithm and a semi-dense mapping system [81]. Fig. 13 shows three examples of TUM fr3_long_office and fr2_desk, where (d) and (e) show semi-dense semantic and object-oriented maps built by our object SLAM. Compared with the sparse map of ORB-SLAM2, our maps can express the environment much better. Moreover, the object-oriented map shows superior performance in environment understanding than the semi-dense map.

The mapping results of other sequences in TUM, Microsoft RGB-D, and Scenes V2 datasets are shown in Fig. 14. It can be seen that the system can process multiple classes of objects with different scales and orientations in complex environments. Inevitably, there are some inaccurate estimations. For instance, in the *fire* sequence, the chair is too large to be well observed by the fast-moving camera, thus yielding an inaccurate estimation. We also conduct the experiment in a

real scenario (Fig. 15). It can be seen that even if the objects are occluded, they can be accurately estimated, which further verifies the robustness and accuracy of our system.

E. Augmented Reality Experiment

Early augmented reality used QR codes, 2D manual features, or image temples to register virtual 3D models, resulting in a restricted range of motion and poor tracking. The sparse point cloud map created by SLAM enables large-scale tracking and high-robust registration for AR. Geometric SLAM-based AR, however, is only concerned with accuracy and robustness, not authenticity. Conversely, our object SLAM-based AR provides complete environment information, thus a more realistic immersive experience can be achieved.

In the case of the desk scene in Fig. 16(a), we use the method described above to construct an object map, as shown in Fig. 16(b), in which we model objects such as the book, keyboard, and bottles.

3D Registration: We present an object-triggered virtual model registration method, instead of 3D registration triggered by a plane or position humanly specified. As shown in Fig. 18(a), the top row represents three raw frames from the video stream, while the bottom row represents the corresponding real-virtual integration scene. Virtual models can be seen registered on the desk to replace real objects based on the object semantics, pose, and size encoded in the object map.

Occlusion and collision: Physical occlusion and collision between the actual scene and virtual models is the crucial reflection of augmented reality. The top row, as seen in Fig. 18(b), is the result of common augmented reality, where virtual models are registered on the top layer of the image, resulting in an unrealistic separation of real and virtual scenes. The bottom row of Fig. 18(b) shows the outcome of our object SLAM-based augmented reality, in which the foreground and background are distinguished, and the real object obscures a portion of the virtual model, where the virtual and physical worlds are fused together. Similarly, Fig. 18(c) depicts the collision effect. The virtual model in the top row falls on the desk without colliding with the bottle. Contrarily, the bottom row shows the outcome of our object SLAM-based augmented reality, in which the virtual model falls and collides with the real bottle, with the dropping propensity changing.

Semantic interaction: Interaction, cascading user command with the real scene and virtual models, plays a crucial role in augmented reality applications. As shown in Fig. 17, clicking different real-world objects produces different virtual interactive effects.

The above functions, object-triggered 3D registration, occlusion, collision, and interaction, rely on accurate object perception of the proposed object SLAM framework. The experimental results demonstrate that object SLAM-based

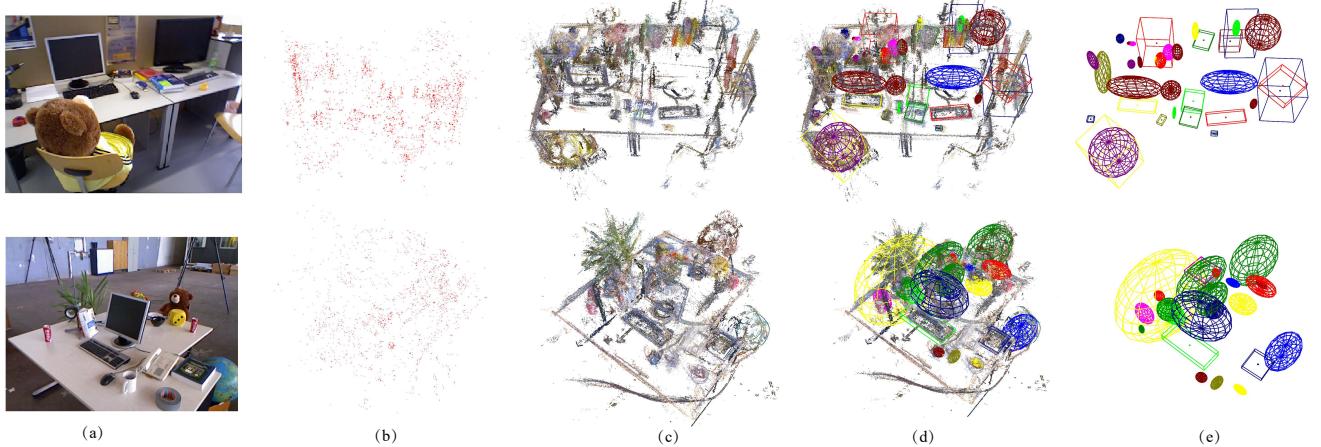


Fig. 13: Different map representations. (a) the RGB images. (b) the sparse map. (c) semi-dense map. (d) our semi-dense semantic map. (e) our lightweight and object-oriented map. (d) and (e) are build by the proposed method.

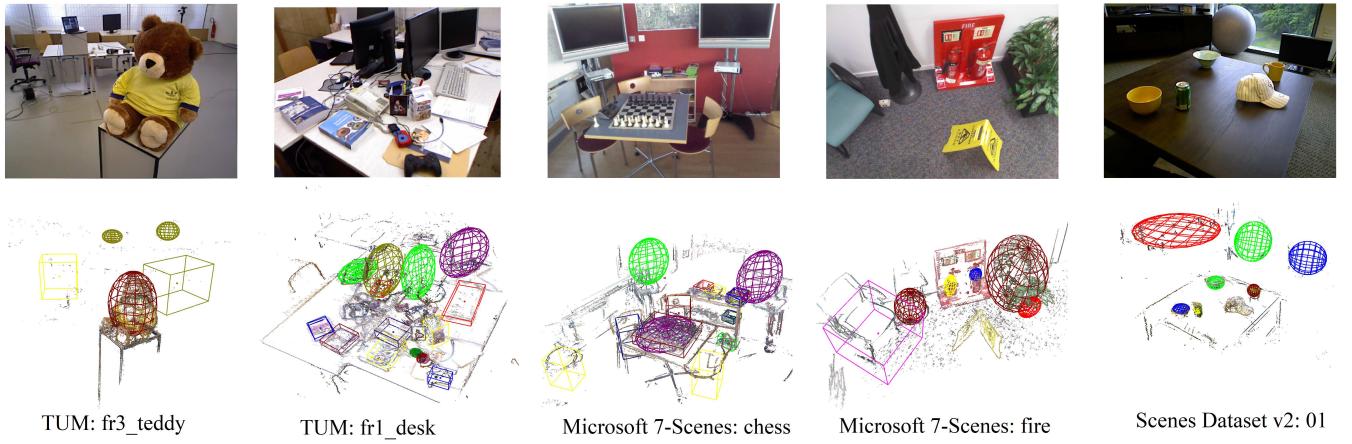


Fig. 14: Mapping results on the three datasets. Top: raw images. Bottom: semi-dense object-oriented map.

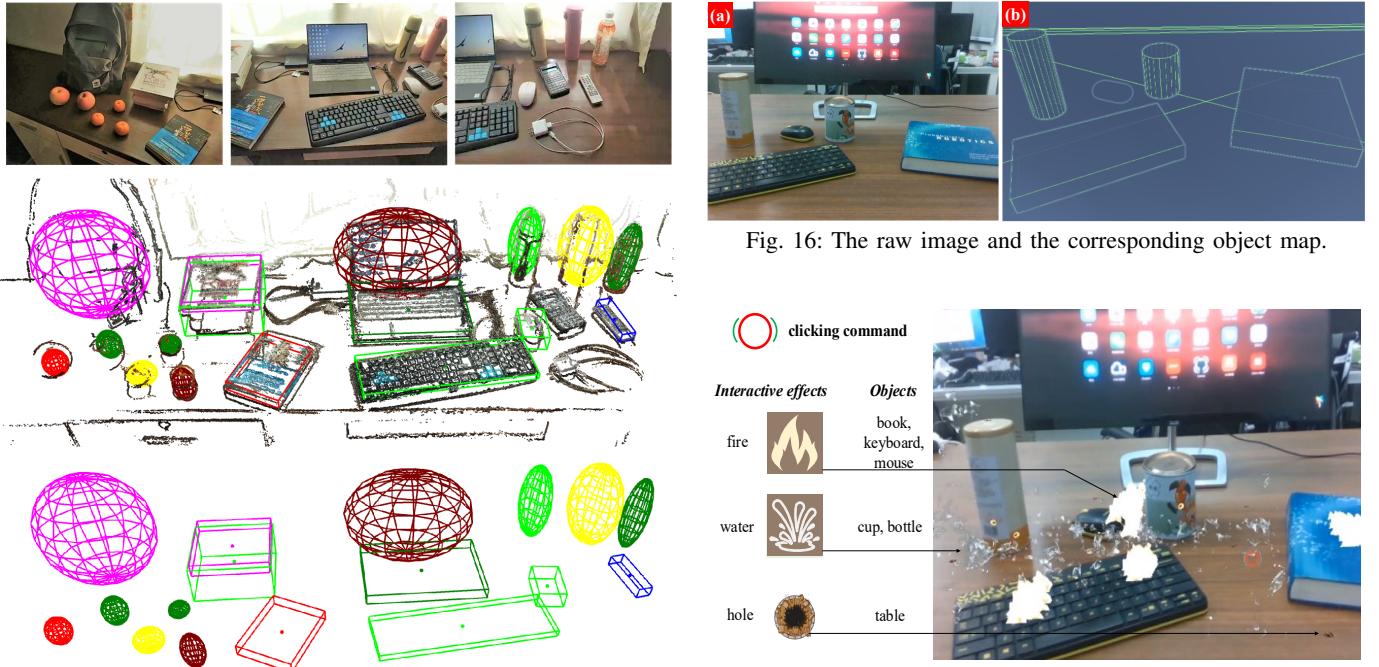


Fig. 15: Mapping results in a real scenario. Top: raw images. Middle: semi-dense object-oriented map. Bottom: lightweight and object-oriented map.

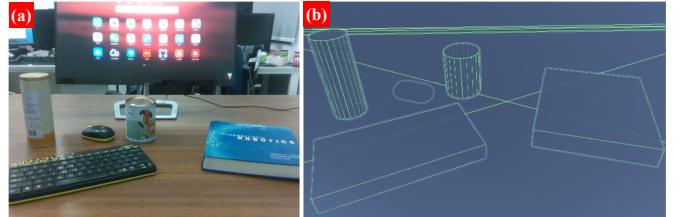


Fig. 16: The raw image and the corresponding object map.

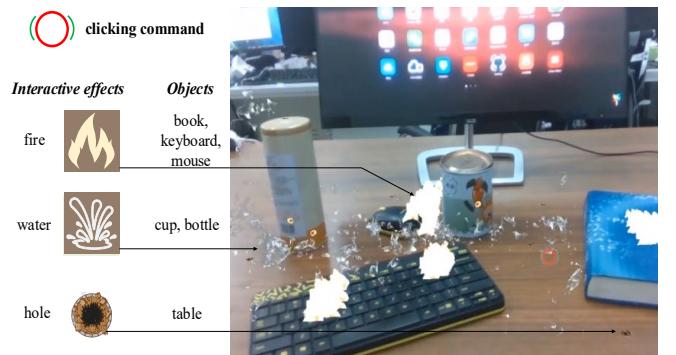


Fig. 17: The demonstration of interaction. Reactions are visualized as a series of augmented reality events.

augmented reality has a fascinating benefit in areas such as gaming, military training, and virtual decorating.

F. Object-based Scene Matching and Relocalization

Scene Matching. In this experiment, we evaluate the performance of the proposed object descriptor-based scene matching, which is crucial for multi-agent collaboration, scene reidentification, and multi-maps merging at different periods. We acquire two separate trajectories and their associated object maps in the same scene, then utilize the suggested method to figure out their relationship. Fig. 19 illustrates the map-matching results in three settings.

The results of the TUM and Microsoft sequences are shown in Fig. 19(a) and Fig. 19(b). The two maps with different scales and numbers of objects match accurately, and the translation between them is also resolved. The match is not based on point clouds or BoW (Bag of Words) of keyframes, but the semantic object descriptor constructed by the object topological map. Additionally, the scale inconsistency of the two maps is also eliminated. Fig. 19(c) shows a real-world example of the matched result. Apart from the previous features, what is worth noting is that the two maps were **recorded under different illumination**. With this scenario, the traditional appearance-based method is trends to fail, demonstrating the robustness of the proposed object descriptor with the semantic level invariance property.

Table III analyzes the performance time of the algorithm. The matching duration is found to be the primary cost, and the time is positively related to the number of objects. The average total duration is approximately 1.23ms, which is both practical and economical in various robot applications.

TABLE III: TIME ANALYSIS OF SCENE MATCHING (ms)

Scene	Object Num.	Descriptor Generate		Match	Pose Resolve	Total
		Map1	Map2			
1	6+4	0.203	0.167	0.458	0.286	0.744
2	10+8	0.213	0.184	0.673	0.383	1.056
3	14+14	0.483	0.437	1.001	0.891	1.892
Ave	10+8.7	0.300	0.263	0.711	0.520	1.231

Relocalization. We perform relocalization experiments with parallax to demonstrate the robustness of the proposed matching method to viewpoint changes. As illustrated in the figure in Tab. IV, we first construct a prior map with a set of prior keyframes and then utilize query keyframes for relocalization, which do not overlap the trajectories of prior keyframes and have parallax. We conduct several repeated experiments under different parallax conditions and compare the success rate of relocalization with ORB-SLAM3 [82]. When the parallax is less than 20°, as shown in Tab. IV, ORB-SLAM3 achieves a relocalization success rate of 32.5%; however, the rate drops sharply to 0 when the parallax is greater than 20°, which demonstrates that the appearance-based descriptor represented by ORB-SLAM3 is extremely sensitive to parallax. Conversely, our method is robust to parallax and achieves a success rate of over 12% even under challenging large parallax.

However, the accuracy of 14.9% is still unsatisfactory. We found that the primary reason is that the observations of the two sets of keyframes are incomplete, thus resulting in inaccurate object modeling. To prove our hypothesis, we manually

TABLE IV: RELOCALIZATION SUCCESS RATE UNDER DIFFERENT PARALLAX

Prior keyframes	Parallax	Query keyframes	Total times	Succ. times	Parallax	Success rate(%)
			329	49	<20	14.9 32.5
			648	96	20-50	14.8 0
			900	109	50-100	12.1 0

generate a scene with objects and divide it into a prior map and query map, assuming that prior and query keyframes generate them, respectively, and that the poses of the objects are obtained from the ground truth. As depicted in the figure in Tab. V, we adjust the proportion of shared objects across the two maps and measure the success rate of relocalization. As demonstrated in Tab. V, we obtain a 100% success rate with a public object ratio of over 50% and retain over 80% accuracy with a ratio of 33%. The result demonstrates the effectiveness of our proposed object descriptor and matching algorithm. It also illustrates its sensitivity to object pose and suggests that more accurate object modeling methods can improve its performance.

TABLE V: RELOCALIZATION SUCCESS RATE UNDER DIFFERENT PUBLIC OBJECT RATIO

Prior map	Query map	Public obj. (30%)	Total times	Succ. times	Public obj. ratio (%)	Success rate (%)
			500	500	60	100
			500	500	55	100
			500	500	50	100
			500	499	44	99.8
			500	467	38	93.4
			500	406	33	81.2

G. Evaluation of Active Mapping

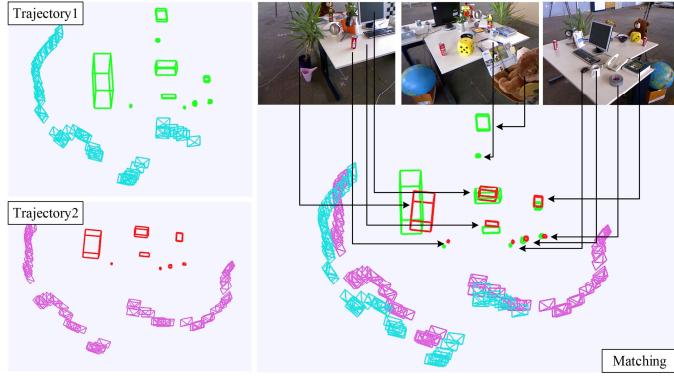
To validate the effectiveness of the active map building and the viability of robot manipulation led by the map, we conduct extensive evaluations in both simulation and real-world environments. The simulated robotic manipulation scene is set in Sapien [83], shown in Fig. 20, where the number of objects and the scene complexities vary in different scenes.

The accurate position estimate is critical for successful robotic manipulation operations such as grasping, placing, arranging, and planning. However, precision is difficult to ensure when the robot estimates autonomously. To quantify the effect of active exploration on object pose estimation, like previous studies [74], [84], we compare our object-driven method with two typically used baseline strategies, *i.e.*, randomized exploration (Random.) and coverage exploration (Cover.). As indicated in Fig. 20, for randomized exploration, the camera pose is randomly sampled from the reachable set relative to the manipulator, while for coverage exploration, a coverage trajectory based on Boustrophedon decomposition [85] is leveraged to scan the scene. At the beginning of all the explorations, an initialization step (Init.), in which the camera is sequentially placed over the four desk corners from a top view, is applied to start the object mapping process. The simulator provides the ground truth of object position, orientation, and size. Correspondingly, the accuracy of pose estimation is evaluated by the Center Distance Error (CDE, cm), the Yaw Angle Error (YAE, degree), and the IoU (including 2D IoU from the top view and 3D IoU) between the ground truth and our estimated results.

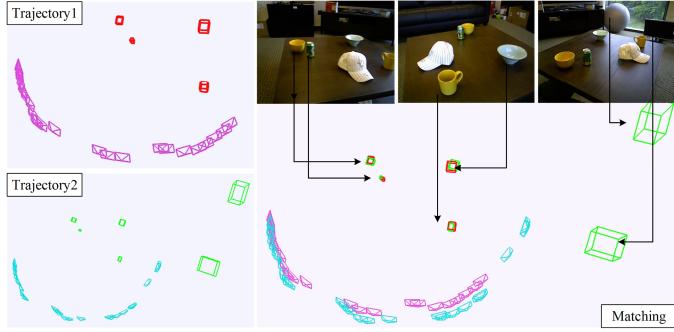


(a) Object-triggered 3D registration. Top: raw images of the scene. Bottom: augmented reality scene with the standard AR without occlusion. (b) The demonstration of occlusion. Top: the standard augmented reality. Bottom: our object-SLAM-based augmented reality with the awareness of collision. (c) The demonstration of collision. Top: the standard augmented reality. Bottom: our object-SLAM-based augmented reality with the awareness of collision.

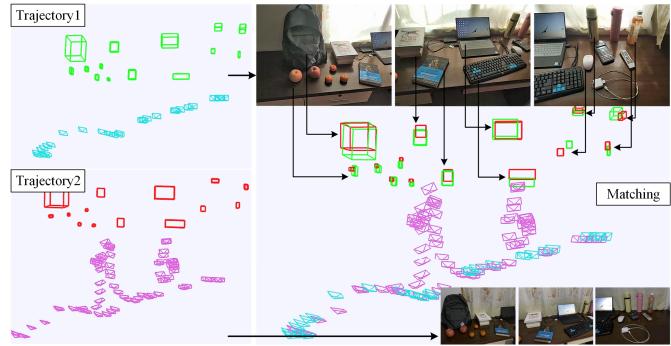
Fig. 18: 3D registration, occlusion, and collision in object SLAM-based augmented reality.



(a) The matching result in TUM RGB-D fr2_desk sequence.



(b) The matching result in RGB-D Scenes v2 scene_01 sequence.



(c) The matching result under different lighting conditions in the real world.

Fig. 19: The quantitative analysis of scene matching.

Table VI shows the evaluation results in seven scenes (Fig. 20). We can see our proposed object-driven exploration strategy achieves a 3D IoU of 45.3%, which is 15.53%, 8.85%, and 13.3% higher than that of the randomized exploration, the coverage exploration, and the initialization, respectively. For 2D IoU, our method achieves an accuracy of 64.83%, which is

TABLE VI: ACCURACY OF OBJECT POSE ESTIMATION

Scene	Metrics	Ours	Random.	Cover.	Init.
1	3D IoU	0.427	0.3056	0.3329	0.3586
	2D IoU	0.6225	0.4571	0.5221	0.5212
	CDE	1.5272	2.2699	1.7876	2.2022
	YAE	3.5	4.8	3.8	2.8
2	3D IoU	0.4307	0.3017	0.4224	0.3400
	2D IoU	0.8679	0.6422	0.7730	0.6480
	CDE	1.4646	2.1096	1.5931	1.9822
	YAE	2.4	1.8	2.7	2.4
3	3D IoU	0.4132	0.3125	0.3685	0.2617
	2D IoU	0.6225	0.4915	0.5517	0.3909
	CDE	1.5503	2.0672	1.4841	2.7489
	YAE	3.9	3.7	3.8	4.9
4	3D IoU	0.4790	0.3824	0.3664	0.3007
	2D IoU	0.6536	0.5886	0.4788	0.4869
	CDE	1.3335	1.3514	1.7508	1.927
	YAE	2.9	2.8	2.1	2.1
5	3D IoU	0.5177	0.2696	0.2884	0.3720
	2D IoU	0.6263	0.4297	0.4326	0.6142
	CDE	1.3704	2.5077	2.1753	2.0084
	YAE	3.9	2.1	3.9	2.1
6	3D IoU	0.4411	0.3000	0.3597	0.2916
	2D IoU	0.5437	0.4850	0.4783	0.5042
	CDE	2.5998	3.4278	2.8411	3.4965
	YAE	2.3	2.7	3	2.7
7	3D IoU	0.4626	0.2118	0.4133	0.3153
	2D IoU	0.6017	0.3839	0.5569	0.4541
	CDE	1.49928	2.3822	1.4832	2.0467
	YAE	2.1	4.5	2.5	2.5
Mean	3D IoU	0.453	0.2977	0.3645	0.3200
	2D IoU	0.6483	0.4969	0.5419	0.5171
	CDE	1.6207	2.3022	1.8736	2.3446
	YAE	3	3.2	3.1	2.8

15.14%, 10.64%, and 13.12% higher than baseline methods. In terms of CDE, our method reaches 1.62cm, significantly less than other methods. For YAE, all exploration strategies achieve an error of approximately 3°, which verifies the robustness of our line-alignment-based yaw angle optimization method. The level of above precision attained is sufficient for robotic manipulation [86]. Moreover, we also find that randomized exploration sometimes performs worse than the initialization result (rows 2, 5, and 7), which indicates that increasing observations do not necessarily result in more accurate pose estimation, and purposeful exploration is necessary.

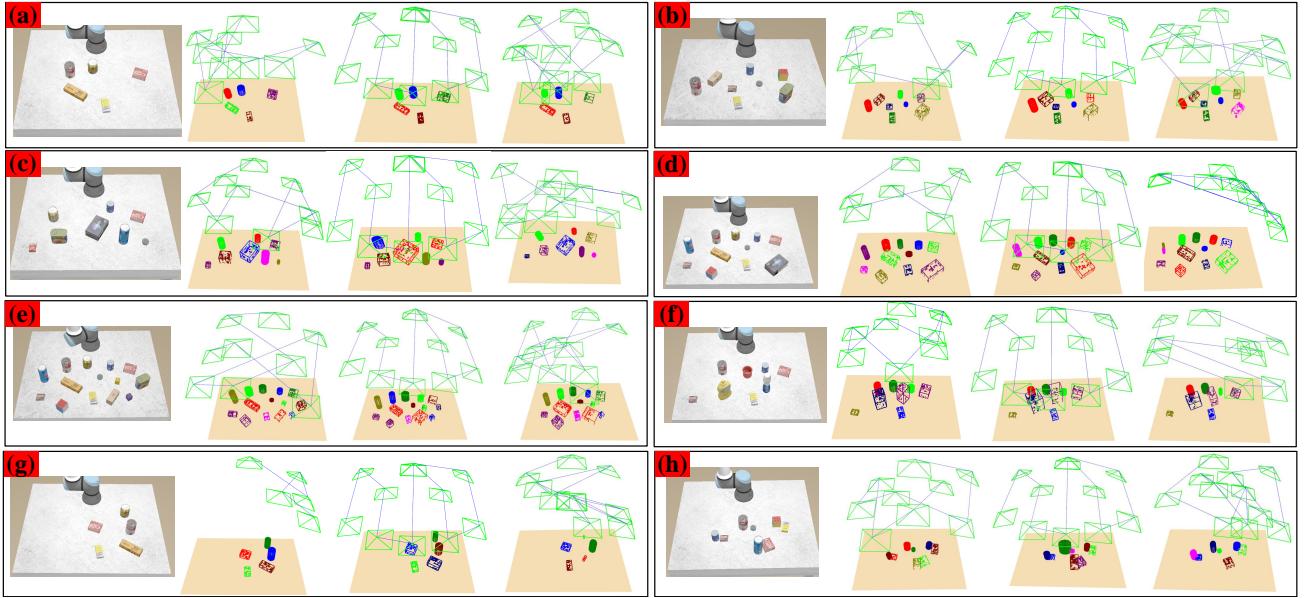


Fig. 20: Comparison of mapping results. The first column in the sub-picture: the scene image; the second column: the result of our object-driven exploration; the third column: the result of the coverage exploration; the fourth column: the result of the randomized exploration.

The mapping results are shown in Fig. 20. The cubes and cylinders are used to model the objects, including poses and scales (analyzed above), based on their semantic categories. The following characteristics are present: **1)** The system can accurately model various objects as the number of objects increases, as shown in Fig. 20(a)-(e), demonstrating its robustness. **2)** Among objects of various sizes, our method focuses more on large objects with lower observation completeness (see Fig. 20(f)). **3)** When objects are distributed unevenly, our proposed strategy can swiftly concentrate the camera on object regions, thus avoiding unnecessary and time-consuming exploration (see Fig. 20(g)). **4)** For scenes with objects close to each other, our method can focus more on regions with fewer occlusions (see Fig. 20(h)). These behaviors verify the effectiveness of our exploration strategy. Additionally, our method has a shorter exploration path yet produces a more precise object posture.

H. Object Grasping and Placement

This experiment uses the incrementally generated object map to perform object grasping. Fig. 21(a) and Fig. 21(b) illustrate the grasping process in simulated and real-world environments, with the object map included. After extensive testing, we obtained a grasping success rate of approximately 86% in the simulator and 81% in the real world, which may be affected by environmental or manipulator noises. It is found that the center and direction of the objects have a significant influence on grasping performance. The proposed method performs well regarding these two metrics, thus ensuring high-quality grasping. Overall, our object SLAM-based pose estimation results can satisfy the requirements of grasping.

We argued that the proposed object map level perception outperforms object pose-only perception and provides information for more intelligent robotics decision-making tasks in addition to grasping. Such include avoiding collisions with other objects, updating the map after grasping, object

arrangement and placement based on object properties, and object delivery requested by the user. We design the object placement experiments to verify the global perception capabilities introduced by object mapping. As shown in Fig. 22, the robot is required to manipulate the original scene (see Fig. 22(a)) to the target scene (see Fig. 22(c)) according to object sizes and classes encoded in the object map.

The global object map is shown in Fig. 22(b), which contains the semantic labels, size, and pose of the objects. The two little blocks are picked up and placed in the large cup (Fig. 22(d)), while the cups are ordered by volume (Fig. 22(e)) and the bottles by height (Fig. 22(f)). This task is challenging for the conventional grasping approach since lacking global perception such as object's height on the map, its surroundings, and could interact with which objects.

IX. DISCUSSION AND ANALYZE

In this section, we analyze the limitations and implementation details of our method and provide potential solutions for the object SLAM community.

1) Data association. Experiments revealed that the two primary reasons for the failure of data association could be summed up as follows: **i)** Long-tailed distribution. In some cases, object centroids are located in the tail of the distribution, which violates the Gaussian distribution assumption and causes the association to fail. Although our multiple association strategy alleviates this to some extent. **ii)** Detected semantic label mistakes. Even if the IoU-based or distribution-based method determines an association between two objects, the association will fail if the labels are inconsistent. Error in label recognition is one of the most common issues with detectors. More generalized, accurate detectors or fine-tuning on specific datasets are potential alternatives.

The running time of data association is shown in the first two columns of Tab. VII. Distribution-based methods include non-parametric and t-tests, while IoU-based methods include

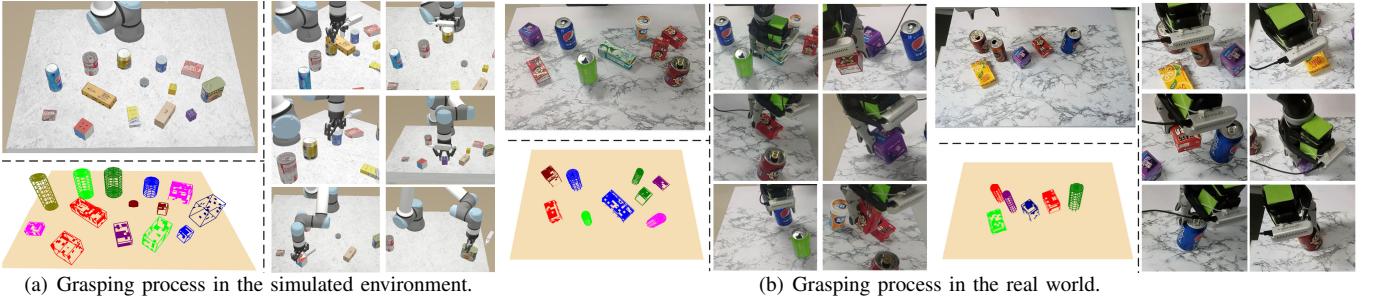


Fig. 21: The demonstration of grasping process.

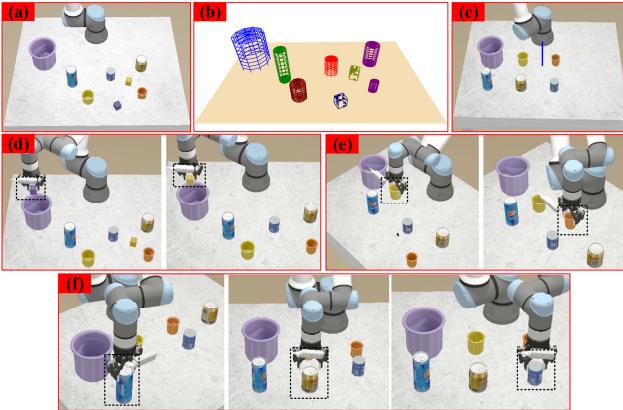


Fig. 22: Object placement according to the global object map.

motion IoU and project IoU, where projecting 3D points to 2D images takes most of the time. Note that the total duration of data association is less than the sum in the table, approximately 8 ms per frame, because sometimes some strategies can be skipped. We perform data association on every frame, which would be more time-efficient if simply performed on keyframes, as CubeSLAM does.

TABLE VII: TIME ANALYSIS OF DATA ASSOCIATION AND OBJECT PARAMETERIZATION (ms/frame)

Data association Distribution-based	Object Parameterization i-Forest	Object Parameterization Line alignment	
		IoU-based	
4.92	7.83	3.86	2.71

2) Object Parameterization. Two factors typically cause failure situations of object pose estimation: **i)** Object surfaces lack texture, or objects are only partially seen due to occlusion or camera viewpoint. In this case, few object point clouds are collected, significantly reducing the pose estimation performance. **ii)** Too many outliers lead the object to be estimated too large, or the modeled object is extremely small since the i-Forest algorithm falls into a local optimum. In the alternatives, the 3D detector [87] based on the complete point cloud may not be optimal due to the incremental characteristic of SLAM; image-based 6-DOF pose estimation [19], [88] is limited by the scale of the training data, resulting in poor generalization [51]. Conversely, incremental detection/segmentation [89] and joint point cloud-image multimodal RGB-D 3D object detection [90], [91] are potentially feasible.

The runtime of object parameterization is shown in the last two columns in Tab. VII, which takes around 6.5 ms per frame on average. The full SLAM system (for camera tracking and

semantic mapping) runs at about 10 fps.

3) Augmented Reality. Augmented reality performance depends on object modeling, camera localization accuracy, and the rendering effect. Here we provide detailed engineering implementations for SLAM developers to migrate their algorithms to augmented reality applications. The AR system comprises three modules: **i)** The Localization and Semantic Mapping (LSM) modules provide camera position, point cloud, and object parameters. Sections IV and V introduce the techniques. **ii)** ROS [92] data transfer module: send images captured by the camera to the LSM module and then publish the estimated camera pose and map elements. **iii)** Virtual-real rendering module: Use the Unity3D engine to subscribe to topics published by ROS, construct a virtual 3D scene, and render it to a 2D image plane. Tab. VIII details the duration of each module, which is executed in parallel.

TABLE VIII: TIME ANALYSIS OF AUGMENTED REALITY

Module	Time (ms/frame)
Localization and semantic mapping	81.84
ROS data transfer	10.96
Virtural-real rendering	25.00

4) Scene Matching. The principal causes for the failure of scene matching and relocalization are: **i)** There are few common objects between the two maps, resulting in a significant difference in the descriptors of the same object in the two maps, leading to matching fails. **ii)** The parallax of the trajectories of the two maps is excessively large, and the observation is insufficient, which affects the accuracy of object modeling and the construction of descriptors. Regarding the first issue, other non-object-level landmarks, such as planes and structural components, can be considered for descriptor construction. For the second challenge, more accurate object modeling techniques can improve the performance of matching and relocalization, as demonstrated by our experiments.

5) Object Grasping. There are two limitations to the object grasping task: **i)** Textured objects and tabletops are required for point-based SLAM tracking to succeed. **ii)** Objects are all regular cube and cylinder shapes in our experiments. Complex irregular objects may necessitate more detailed shape reconstruction and grasp point detection. Nonetheless, we demonstrate the potential of object SLAM for grasping tasks without object priors. Model-free and unseen object grasping will be the future trend. In terms of running time, the speed is even faster than 10fps because, in this setting, the data association is more straightforward, and more time is spent on the active mapping analysis process.

X. CONCLUSION

We presented an object mapping framework that aims to create an object-oriented map using general models that parameterize the object's position, orientation, and size. First, we investigated related fundamental techniques for object mapping, including multi-view data association and object pose estimation. We then center on the object map and validate its potential in multiple high-level tasks such as augmented reality, scene matching, and object grasping. Finally, we analyzed the limitations and failure instances of our method and gave possible alternatives to inspire the development of related fields. The following points will be given significant consideration in future work: 1) Dynamic objects data association, tracking, and trajectory prediction; 2) Irregular and unseen object modeling and tightly coupled optimization with SLAM; 3) Object-level relocalization and loop closure; 4) Omnidirectional perception with multi-sensor and multiple semantic networks to realize spatial AI.

REFERENCES

- [1] A. J. Davison, "Futuremapping: The computational structure of spatial ai systems," *arXiv preprint arXiv:1803.11288*, 2018.
- [2] Q. Wang, Z. Yan, J. Wang, F. Xue, W. Ma, and H. Zha, "Line flow based simultaneous localization and mapping," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1416–1432, 2021.
- [3] Y. Zhou, H. Li, and L. Kneip, "Canny-vo: Visual odometry with rgb-d cameras based on geometric 3-d-2-d edge alignment," *IEEE Transactions on Robotics*, vol. 35, no. 1, pp. 184–199, 2018.
- [4] R. Yunus, Y. Li, and F. Tombari, "Manhattanslam: Robust planar tracking and mapping leveraging mixture of manhattan frames," in *Proceedings of 2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6687–6693.
- [5] S. Zhao, P. Wang, H. Zhang, Z. Fang, and S. Scherer, "Tp-tio: A robust thermal-inertial odometry with deep thermalpoint," in *Proceedings of 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4505–4512.
- [6] S. Cao, X. Lu, and S. Shen, "Gvins: Tightly coupled gnss–visual–inertial fusion for smooth and consistent state estimation," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2004–2021, 2022.
- [7] T.-M. Nguyen, S. Yuan, M. Cao, T. H. Nguyen, and L. Xie, "Virtual slam: Tightly coupled camera-imu-uwb-lidar slam," *arXiv preprint arXiv:2105.03296*, 2021.
- [8] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfusion: Dense 3d semantic mapping with convolutional neural networks," in *Proceedings of 2017 IEEE International Conference on Robotics and automation (ICRA)*. IEEE, 2017, pp. 4628–4635.
- [9] S. Yang, Y. Huang, and S. Scherer, "Semantic 3d occupancy mapping through efficient high order crfs," in *Proceedings of 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 590–597.
- [10] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *Proceedings of 2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1689–1696.
- [11] D. Frost, V. Prisacariu, and D. Murray, "Recovering stable scale in monocular slam using object-supplemented bundle adjustment," *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 736–747, 2018.
- [12] A. Iqbal and N. R. Gans, "Localization of classified objects in slam using nonparametric statistics and clustering," in *Proceedings of 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 161–168.
- [13] B. Mu, S.-Y. Liu, L. Paull, J. Leonard, and J. P. How, "Slam with objects using a nonparametric pose graph," in *Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4602–4609.
- [14] N. Sünderhauf, T. T. Pham, Y. Latif, M. Milford, and I. Reid, "Meaningful maps with object-oriented semantic mapping," in *Proceedings of 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 5079–5085.
- [15] 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*, vol. 4, no. 3, pp. 3037–3044, 2019.
- [16] A. Sharma, W. Dong, and M. Kaess, "Compositional and scalable object slam," in *Proceedings of 2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 11626–11632.
- [17] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. Kelly, and A. J. Davison, "Slam++: Simultaneous localisation and mapping at the level of objects," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 1352–1359.
- [18] S. Choudhary, L. Carlone, C. Nieto, J. Rogers, Z. Liu, H. I. Christensen, and F. Dellaert, "Multi robot object-based slam," in *Proceedings of International Symposium on Experimental Robotics*. Springer, 2016, pp. 729–741.
- [19] Y. Labb  , J. Carpentier, M. Aubry, and J. Sivic, "Cosypose: Consistent multi-view multi-object 6d pose estimation," in *Proceedings of European Conference on Computer Vision*. Springer, 2020, pp. 574–591.
- [20] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, "Probabilistic data association for semantic slam," in *Proceedings of 2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 1722–1729.
- [21] P. Parkhiya, R. Khawad, J. K. Murthy, B. Bhowmick, and K. M. Krishna, "Constructing category-specific models for monocular object-slam," in *Proceedings of 2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4517–4524.
- [22] N. Joshi, Y. Sharma, P. Parkhiya, R. Khawad, K. M. Krishna, and B. Bhowmick, "Integrating objects into monocular slam: Line based category specific models," in *Proceedings of the 11th Indian Conference on Computer Vision, Graphics and Image Processing*, 2018, pp. 1–9.
- [23] E. Sucar, K. Wada, and A. Davison, "Nodeslam: Neural object descriptors for multi-view shape reconstruction," in *2020 International Conference on 3D Vision (3DV)*. IEEE, 2020, pp. 949–958.
- [24] S. Yang and S. Scherer, "Cubeslam: Monocular 3-d object slam," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 925–938, 2019.
- [25] L. Nicholson, M. Milford, and N. Sünderhauf, "Quadricslam: Dual quadrics from object detections as landmarks in object-oriented slam," *IEEE Robotics and Automation Letters*, vol. 4, no. 1, pp. 1–8, 2018.
- [26] Y. Wu, Y. Zhang, D. Zhu, Y. Feng, S. Coleman, and D. Kerr, "Eaoslam: Monocular semi-dense object slam based on ensemble data association," in *Proceedings of 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 4966–4973.
- [27] Y. Wu, Y. Zhang, D. Zhu, X. Chen, S. Coleman, W. Sun, X. Hu, and Z. Deng, "Object slam-based active mapping and robotic grasping," in *Proceedings of 2021 International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 1372–1381.
- [28] K. Chen, J. Liu, Q. Chen, Z. Wang, and J. Zhang, "Accurate object association and pose updating for semantic slam," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 25169–25179, 2022.
- [29] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "Fairmot: On the fairness of detection and re-identification in multiple object tracking," *International Journal of Computer Vision*, vol. 129, no. 11, pp. 3069–3087, 2021.
- [30] J. Li, D. Meger, and G. Dudek, "Semantic mapping for view-invariant relocalization," in *Proceedings of 2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7108–7115.
- [31] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger, "Fusion++: Volumetric object-level slam," in *Proceedings of 2018 international conference on 3D vision (3DV)*. IEEE, 2018, pp. 32–41.
- [32] J. Wang, M. R  n, and L. Agapito, "Dsp-slam: Object oriented slam with deep shape priors," in *Proceedings of 2021 International Conference on 3D Vision (3DV)*. IEEE, 2021, pp. 1362–1371.
- [33] B. Xu, A. J. Davison, and S. Leutenegger, "Learning to complete object shapes for object-level mapping in dynamic scenes," in *Proceedings of 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 2257–2264.
- [34] Y. Liu, Y. Petillot, D. Lane, and S. Wang, "Global localization with object-level semantics and topology," in *Proceedings of 2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4909–4915.
- [35] K. Ok, K. Liu, K. Frey, J. P. How, and N. Roy, "Robust object-based slam for high-speed autonomous navigation," in *Proceedings of 2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 669–675.

- [36] Y. Xiang and D. Fox, "Da-rnn: Semantic mapping with data associated recurrent neural networks," in *Proceedings of Robotics: Science and Systems (RSS)*, 2017.
- [37] K. Li, D. DeTone, Y. F. S. Chen, M. Vo, I. Reid, H. Rezatofighi, C. Sweeney, J. Straub, and R. Newcombe, "Odam: Object detection, association, and mapping using posed rgb video," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5998–6008.
- [38] N. Merrill, Y. Guo, X. Zuo, X. Huang, S. Leutenegger, X. Peng, L. Ren, and G. Huang, "Symmetry and uncertainty-aware object slam for 6dof object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14901–14910.
- [39] C. Xing, X. Sun, A. Cramariuc, S. Gull, J. J. Chung, C. Cadena, R. Siegwart, and F. Tschopp, "Descriptellation: Deep learned constellation descriptors for slam," *arXiv preprint arXiv:2203.00567*, 2022.
- [40] M. Strecke and J. Stuckler, "Em-fusion: Dynamic object-level slam with probabilistic data association," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 5865–5874.
- [41] S. Yang, Z.-F. Kuang, Y.-P. Cao, Y.-K. Lai, and S.-M. Hu, "Probabilistic projective association and semantic guided relocalization for dense reconstruction," in *Proceedings of 2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7130–7136.
- [42] J. Zhang, M. Gui, Q. Wang, R. Liu, J. Xu, and S. Chen, "Hierarchical topic model based object association for semantic slam," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 11, pp. 3052–3062, 2019.
- [43] T. Ran, L. Yuan, J. Zhang, L. He, R. Huang, and J. Mei, "Not only look but infer: Multiple hypothesis clustering of data association inference for semantic slam," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.
- [44] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 165–174.
- [45] M. Han, Z. Zhang, Z. Jiao, X. Xie, Y. Zhu, S.-C. Zhu, and H. Liu, "Reconstructing interactive 3d scenes by panoptic mapping and cad model alignments," in *Proceedings of 2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 12199–12206.
- [46] Z. Cao, Y. Zhang, R. Tian, R. Ma, X. Hu, S. Coleman, and D. Kerr, "Object-aware slam based on efficient quadric initialization and joint data association," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9802–9809, 2022.
- [47] M. Runz, M. Buffier, and L. Agapito, "Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects," in *Proceedings of 2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2018, pp. 10–20.
- [48] S. Lin, J. Wang, M. Xu, H. Zhao, and Z. Chen, "Topology aware object-level semantic mapping towards more robust loop closure," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7041–7048, 2021.
- [49] J. Lu, B. Tian, H. Shen, and X. Zhang, "Real-time instance-aware segmentation and semantic mapping on edge devices," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–9, 2022.
- [50] J. Li, K. Koreitem, D. Meger, and G. Dudek, "View-invariant loop closure with oriented semantic landmarks," in *Proceedings of 2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 7943–7949.
- [51] Y. Ming, X. Yang, and A. Calway, "Object-augmented rgb-d slam for wide-disparity relocalisation," in *Proceedings of 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 2203–2209.
- [52] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
- [53] P. Schmuck and M. Chli, "Ccm-slam: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams," *Journal of Field Robotics*, vol. 36, no. 4, pp. 763–781, 2019.
- [54] A. Gawel, C. Del Don, R. Siegwart, J. Nieto, and C. Cadena, "X-view: Graph-based semantic multi-view localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1687–1694, 2018.
- [55] X. Guo, J. Hu, J. Chen, F. Deng, and T. L. Lam, "Semantic histogram based graph matching for real-time multi-robot global localization in large scale environment," *IEEE Robotics and Automation Letters*, 2021.
- [56] C. Qin, Y. Zhang, Y. Liu, and G. Lv, "Semantic loop closure detection based on graph matching in multi-objects scenes," *Journal of Visual Communication and Image Representation*, vol. 76, p. 103072, 2021.
- [57] Z. Zhang and D. Scaramuzza, "Beyond point clouds: Fisher information field for active visual localization," in *Proceedings of 2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5986–5992.
- [58] Z. Zeng, A. Röfer, and O. C. Jenkins, "Semantic linking maps for active visual object search," in *Proceedings of 2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1984–1990.
- [59] B. Charrow, G. Kahn, S. Patil, S. Liu, K. Goldberg, P. Abbeel, N. Michael, and V. Kumar, "Information-theoretic planning with trajectory optimization for dense 3d mapping," in *Proceedings of Robotics: Science and Systems*, vol. 11. Rome, 2015, pp. 3–12.
- [60] C. Wang, D. Zhu, T. Li, M. Q. Meng, and C. W. de Silva, "Efficient autonomous robotic exploration with semantic road map in indoor environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2989–2996, July 2019.
- [61] S. Kriegel, C. Rink, T. Bodenmüller, and M. Suppa, "Efficient next-best-scan planning for autonomous 3d surface reconstruction of unknown objects," *Journal of Real-Time Image Processing*, vol. 10, no. 4, pp. 611–631, 2015.
- [62] K. Wada, E. Sucar, S. James, D. Lenton, and A. J. Davison, "Morefusion: Multi-object reasoning for 6d pose estimation from volumetric fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14540–14549.
- [63] D. Almeida, E. Ataer-Cansizoglu, and R. Corcodel, "Detection, tracking and 3d modeling of objects with sparse rgb-d slam and interactive perception," in *Proceedings of 2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2019, pp. 1–8.
- [64] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
- [65] F. Wilcoxon, "Individual comparisons by ranking methods," in *Breakthroughs in statistics*. Springer, 1992, pp. 196–202.
- [66] S. Sidney, "Nonparametric statistics for the behavioral sciences," *The Journal of Nervous and Mental Disease*, vol. 125, no. 3, p. 497, 1957.
- [67] E. L. Lehmann and H. J. D'Abrera, *Nonparametrics: statistical methods based on ranks*. Holden-day, 1975.
- [68] S. Yang and S. Scherer, "Monocular object and plane slam in structured environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3145–3152, 2019.
- [69] T. Pire, J. Corti, and G. Grinblat, "Online object detection and localization on stereo visual slam system," *Journal of Intelligent & Robotic Systems*, vol. 98, no. 2, pp. 377–386, 2020.
- [70] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 6, no. 1, pp. 1–39, 2012.
- [71] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "Lsd: A line segment detector," *Image Processing On Line*, vol. 2, pp. 35–55, 2012.
- [72] S. A. Cook, "The complexity of theorem-proving procedures," in *Proceedings of the third annual ACM symposium on Theory of computing*, 1971, pp. 151–158.
- [73] G. Kahn, P. Sujan, S. Patil, S. Bopardikar, J. Ryde, K. Goldberg, and P. Abbeel, "Active exploration using trajectory optimization for robotic grasping in the presence of occlusions," in *Proceedings of 2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4783–4790.
- [74] E. Arruda, J. Wyatt, and M. Kopicki, "Active vision for dexterous grasping of novel objects," in *Proceedings of 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 2881–2888.
- [75] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.
- [76] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.
- [77] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proceedings of 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 573–580.
- [78] A. Iqbal and N. R. Gans, "Data association and localization of classified objects in visual slam," *Journal of Intelligent & Robotic Systems*, vol. 100, pp. 113–130, 2020.
- [79] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2930–2937.

- [80] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3d scene labeling," in *Proceedings of 2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3050–3057.
- [81] S. He, X. Qin, Z. Zhang, and M. Jagersand, "Incremental 3d line segment extraction from semi-dense slam," in *Proceedings of 2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 1658–1663.
- [82] 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.
- [83] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang *et al.*, "Sapien: A simulated part-based interactive environment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 097–11 107.
- [84] D. Morrison, P. Corke, and J. Leitner, "Multi-view picking: Next-best-view reaching for improved grasping in clutter," in *Proceedings of 2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8762–8768.
- [85] D. Kaljaca, B. Vroegindeweij, and E. van Henten, "Coverage trajectory planning for a bush trimming robot arm," *Journal of Field Robotics*, vol. 37, no. 2, pp. 283–308, 2020.
- [86] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3343–3352.
- [87] Z. Liu, Z. Zhang, Y. Cao, H. Hu, and X. Tong, "Group-free 3d object detection via transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2949–2958.
- [88] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized object coordinate space for category-level 6d object pose and size estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2642–2651.
- [89] S.-C. Wu, J. Wald, K. Tateno, N. Navab, and F. Tombari, "Scenegraph-fusion: Incremental 3d scene graph prediction from rgbd sequences," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7515–7525.
- [90] Y. Wang, X. Chen, L. Cao, W. Huang, F. Sun, and Y. Wang, "Multimodal token fusion for vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 186–12 195.
- [91] H. Yang, C. Shi, Y. Chen, and L. Wang, "Boosting 3d object detection via object-focused image fusion," *arXiv preprint arXiv:2207.10589*, 2022.
- [92] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng *et al.*, "Ros: an open-source robot operating system," in *Proceedings of ICRA workshop on open source software*. Kobe, Japan, 2009.



Yanmin Wu received the B.S. degree in electronic information engineering from Shenyang Normal University, Shenyang, Liaoning, China, in 2018, and the M.S. degree in robot science and engineering from Northeastern University, Shenyang, Liaoning, China, in 2021.

He is currently pursuing a Ph.D. degree in computer applied technology at the School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School, Shenzhen, China. His research interests include visual SLAM, 3D reconstruction, and scene understanding.



Yunzhou Zhang received the B.S. and M.S. degrees in mechanical and electronic engineering from the National University of Defense Technology, Changsha, China, in 1997 and 2000, respectively, and the Ph.D. degree in pattern recognition and intelligent system from Northeastern University, Shenyang, China, in 2009.

He is currently a Professor with the Faculty of Robot Science and Engineering, Northeastern University. He leads the Cloud Robotics and Visual Perception Research Group. His research has been supported by funding from various sources, such as the National Natural Science Foundation of China, the Ministry of Education of China, and some famous high-tech companies. He has published many journal articles and conference papers on intelligent robots, computer vision, and wireless sensor networks. His research interests include intelligent robots, computer vision, and sensor networks.



Delong Zhu received the B.S. degree in computer science and technology from Northeastern University, Shenyang, Liaoning, China, in 2015, and the Ph.D. degree in electronic engineering from the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, in 2020.

He spent nine months at the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, as a Visiting Scholar. His research interests include motion planning in dynamic environments and deep reinforcement learning.



Zhiqiang Deng received the B.S. degree in automation from Shenyang Jianzhu University, Shenyang, China, in 2020. He is currently pursuing the master's degree in pattern recognition and intelligent systems from the College of Information Science and Engineering, Northeastern University, Shenyang, China. His research interests include visual simultaneous localization and mapping (SLAM) and augmented reality.



Wenkai Sun received the B.S. degree in automation from Tiangong University, Tianjin, China, in 2020. He is currently pursuing the master's degree in pattern recognition and intelligent systems from the College of Information Science and Engineering, Northeastern University, Shenyang, China. His research interests include semantic SLAM and augmented reality.



Xin Chen received the B.S. degree in Mechanical and Electronic Engineering from Harbin University of Science and Technology, in 2019, and the M.S. degree in robot science and engineering from Northeastern University, Shenyang, Liaoning, China, in 2022. His research interests include object pose estimation and robot manipulation.



Jian Zhang (M'14) received the B.S. degree from the Department of Mathematics, Harbin Institute of Technology (HIT), Harbin, China, in 2007, and received his M.Eng. and Ph.D. degrees from the School of Computer Science and Technology, HIT, in 2009 and 2014, respectively. From 2014 to 2018, he worked as a postdoctoral researcher at Peking University (PKU), Hong Kong University of Science and Technology (HKUST), and King Abdullah University of Science and Technology (KAUST).

Currently, he is an Assistant Professor with the School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School, Shenzhen, China. His research interests include intelligent multimedia processing, deep learning and optimization. He has published over 90 technical articles in refereed international journals and proceedings. He received the Best Paper Award at the 2011 IEEE Visual Communications and Image Processing (VCIP) and was a co-recipient of the Best Paper Award of 2018 IEEE MultiMedia.