

Learn to Memorize and to Forget: A Continual Learning Perspective of Dynamic SLAM

Supplementary Material

Baicheng Li¹, Zike Yan^{2†}, Dong Wu¹, Hanqing Jiang³, and Hongbin Zha^{1†}

¹ National Key Lab of GAI, School of IST
PKU-SenseTime Machine Vision Joint Lab
Peking University

² AIR, Tsinghua University

³ SenseTime Research

1 Supplementary Results

1.1 Segmentation Pruning

Segmentation with FastSAM [47] leads to varying granularities. A person may be separated into different parts as arms, legs, body, and head. In such a case, we wish to retain the instance-level segmentation instead of the part-level decomposition, thereby reducing the number of classifier updates.

Specifically, as illustrated in Fig. 1, for any two segments R_1 and R_2 , we consider R_1 to be a part of R_2 and delete it if the portion of overlapped areas between R_1 and R_2 is larger than T_R as:

$$S_{R_1 \cap R_2} > T_R \times S_{R_1} \quad (1)$$

where T_R is set to 0.9 in our experiments.

1.2 Mesh Evaluation

We follow the dense dynamic SLAM system of ReFusion [25] to evaluate the mesh results quantitatively. As shown in Fig. 2 and 3, our method achieves comparable results with ReFusion [25] and outperforms StaticFusion [32]. Note that NeRF-based methods mainly focus on realistic rendering (as presented in our main paper and supp. video) instead of surface reconstruction, the quantitative results sufficiently verify our map quality in dynamic environments.

1.3 Ablation Study on the Replay Training of the Classifier

We compare the number of updates for the classifier based on whether replay training is performed. The experiment is conducted with different image encoders across two high-dynamic sequences: 'person_tracking' and 'balloon'. The results are illustrated in Fig. 4. It is clearly observable that the classifier undergoing replay training requires fewer updates compared to the one without replay, regardless of the image encoder used. Detailed analysis can be found in the "continual learning of the classifier" parts of Sec. 4.3 and Sec. 5.3.

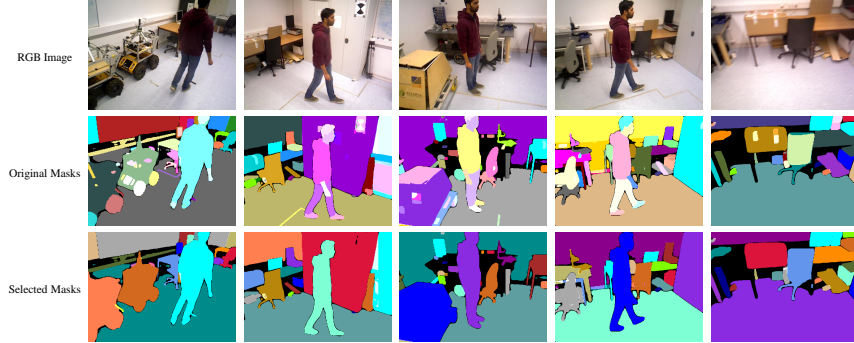


Fig. 1: The mask selection strategy helps to reduce a significant number of unnecessary masks, thereby lessening interference to the system.

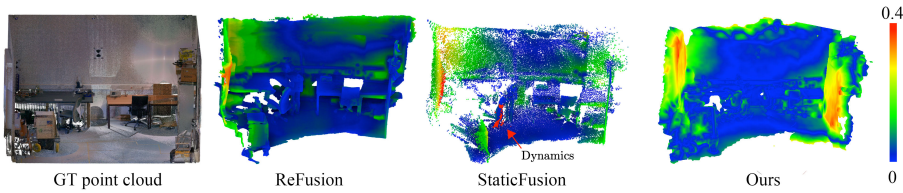


Fig. 2: Additional comparison results of mapping and tracking.

1.4 Run-time

We also tested the average run-time of each component of the system in the two sequences mentioned above. In high-dynamic environments, we effectively balanced the system’s speed and accuracy. Our method can run at a frame rate of around 1 fps. In contrast, the frame rates of Co-SLAM [43], iMap [34], and NICE-SLAM [49] are approximately 3 fps, 2 fps, and less than 0.1 fps, respectively.

Table 1: Average run-time of Bonn dataset.

Instance feature ex- traction	Tracking	Bundle adjustment	Classifier updating
231ms	149ms	161ms	397ms

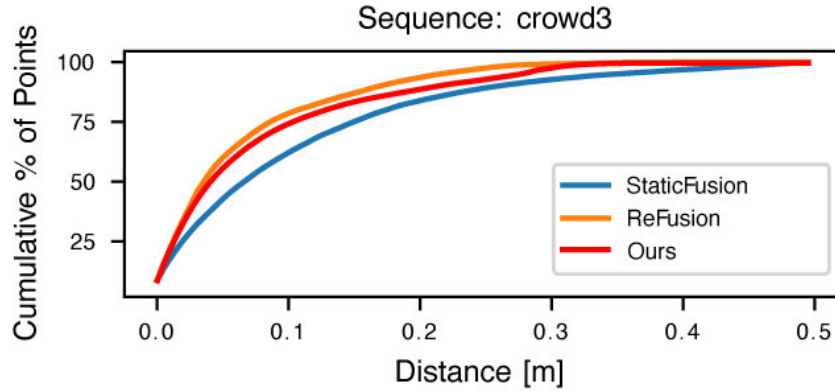


Fig. 3: Additional comparison results of mapping and tracking.

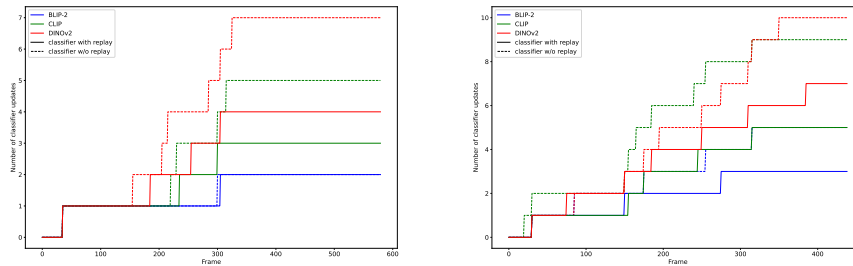


Fig. 4: Classifier update times: with and without replay training.

1.5 Visualization of Tracking Results

In Table 1 of the main paper, we present the quantitative results of camera tracking, showing that our method achieves higher accuracy compared to other dynamic SLAM approaches. Here, we also provide a qualitative demonstration of these results. Fig. 5 shows a comparison of our trajectories with the ground truth on three sequences.

1.6 Ablation Study on the Visual Encoders and Prior Knowledge

We tested the impact of different experimental settings on the accuracy of camera tracking. Table 2 and Table 3 present the ablation study results on Bonn RGB-D dataset. As shown in Table 2, using different visual encoders causes slight variations in the tracking results, with BLIP-2 [17] performing the best and DINOv2 [22] performing the worst. Table 3 reflects that the presence of prior knowledge has minimal impact on tracking accuracy. In most cases, our

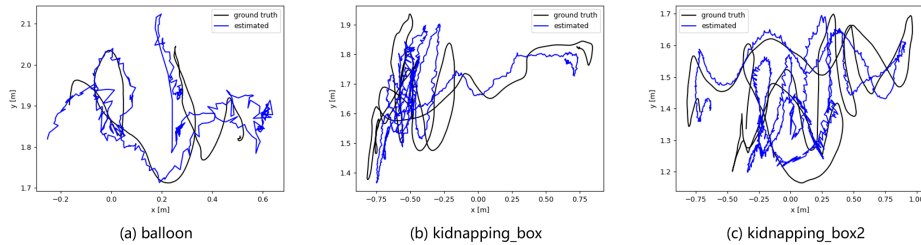


Fig. 5: Trajectory plots of three sequences in Bonn RGB-D dataset.

Table 2: ATE (RMS) with different visual encoders. **Table 3:** ATE (RMS) with and without prior knowledge.

Sequence	BLIP-2	CLIP	DINOv2	Sequence	w/o prior	with prior
balloon	0.206	0.211	0.228	balloon	0.206	0.212
synchronous	0.130	0.139	0.146	synchronous	0.130	0.134
person_tracking	0.274	0.271	0.278	person_tracking	0.274	0.259

method achieves precise camera tracking without prior knowledge. Only in handling extremely challenging scenarios (such as Fig. 9 in the main paper), do we need to incorporate prior knowledge.

2 The Forgetting Issue of Implicit Neural Representations

The global representation of iMap [34] results in severe catastrophic forgetting if keyframe-based replay is not deployed, as the distribution shift constantly occurs during the sequential data capturing. On the other hand, subsequent NeRF-based SLAMs like Co-SLAM [43] and Point-SLAM [29] introduce the local neural representations. They store local features of the scene on grids or points, which to some extent alleviates this negative impact. However, they also employ a global decoder to interpret local features, leading to catastrophic forgetting of the global decoder if keyframe replay is not performed, thereby affecting the operation of the entire system. Therefore, the keyframe buffer lays the foundation of the neural SLAM systems for distilling all past knowledge jointly. We control the replayed buffer through a continually learned classifier so effects from dynamic objects over the past observations can be instantly eliminated, thereby leading to the forgetting of these dynamic objects and maintaining an invariant map to be updated. The methodology is applicable not only to global neural representations but also to representations with discretely-stored local features.