2026 IEEE/SICE International Symposium on System Integration (SII)
January 11-14, 2026. Cancún, México

Loop Closure using AnyLoc Visual Place Recognition in DPV-SLAM

Wenzheng Zhang*, Kazuki Adachi*, Yoshitaka Hara†, Sousuke Nakamura‡

* Graduate School of Science and Engineering, Hosei University, Tokyo, Japan

† Future Robotics Technology Center (fuRo), Chiba Institute of Technology, Chiba, Japan

‡ Faculty of Science and Engineering, Hosei University, Tokyo, Japan

***Abstract*— Loop closure is crucial for maintaining the accuracy and consistency of visual SLAM. We propose a method to improve loop closure performance in DPV-SLAM. Our approach integrates AnyLoc, a learning-based visual place recognition technique, as a replacement for the classical Bag of Visual Words (BoVW) loop detection method. In contrast to BoVW, which relies on handcrafted features, AnyLoc utilizes deep feature representations, enabling more robust image retrieval across diverse viewpoints and lighting conditions. Furthermore, we propose an adaptive mechanism that dynamically adjusts similarity threshold based on environmental conditions, removing the need for manual tuning. Experiments on both indoor and outdoor datasets demonstrate that our method significantly outperforms the original DPV-SLAM in terms of loop closure accuracy and robustness. The proposed method offers a practical and scalable solution for enhancing loop closure performance in modern SLAM systems.**

I. Introduction

Simultaneous Localization and Mapping (SLAM) is a fundamental technology that enables autonomous navigation in mobile robotic systems. Among the various SLAM modalities, visual SLAM achieves both robot localization and map building solely based on visual input, typically from monocular or stereo cameras.

In monocular visual SLAM, accumulated drift and scale inconsistency inevitably emerge over time, potentially compromising the global consistency of the constructed map. Loop closure is a crucial process for mitigating these issues by recognizing previously visited locations and subsequently refining the map.

In real-world environments, mobile robots often face challenging conditions such as illumination changes, dynamic occlusions, and substantial viewpoint variations. These factors impose strict demands on the robustness of visual loop detection. Conventional Bag of Visual Words (BoVW) approaches are particularly vulnerable under such circumstances, frequently resulting in false matches or missed detections, and thereby degrading both the accuracy and consistency of SLAM systems.

Recent advances in learning-based visual feature representations have enabled novel loop detection methods that surpass classical approaches such as BoVW. In particular, AnyLoc [1], a visual place recognition technique, achieves high generalization and semantic discrimination through extensive pretraining, demonstrating superior robustness to environmental variations compared to conventional methods.

Unlike classical BoVW approaches that depend on handcrafted local features such as ORB [2] or SIFT [3] and manually constructed vocabularies, AnyLoc utilizes deep neural networks to extract robust and semantically meaningful image representations. Its pretraining strategy facilitates strong generalization across diverse environments and temporal changes, making it highly suitable for loop detection in real-world environments.

In our previous work [4], we evaluated several monocular visual SLAM systems and found that Deep Patch Visual SLAM (DPV-SLAM) [5] demonstrated outstanding performance. DPV-SLAM is a monocular visual SLAM method extended from Deep Patch Visual Odometry (DPVO) [6], which efficiently leverages deep feature representations to achieve real-time performance with low memory consumption. While DPV-SLAM leverages deep learning in its visual odometry module, it still relies on the classical BoVW-based approach for loop detection.

In this paper, we propose to enhance the loop closure module of DPV-SLAM by replacing its BoVW-based loop detection with AnyLoc. We further introduce a complete loop closure pipeline that combines adaptive similarity thresholding with geometric verification. The proposed method is evaluated using real-world camera data to validate its effectiveness.

Our contributions are as follows:

- We integrate AnyLoc, a learning-based visual place recognition method, into loop closure of DPV-SLAM.
- We introduce an automatic threshold adjustment mechanism that adapts the similarity threshold to different environments.
- Experiments show that our method outperforms the original DPV-SLAM in terms of loop closure accuracy and robustness.

II. Related Work

Research related to the proposed method can be broadly categorized into three groups: loop detection methods based on Bag of Visual Words (BoVW), loop detection methods utilizing learning-based visual features, and loop detection methods leveraging topological and semantic information.

In many conventional visual SLAM systems, loop detection commonly relies on BoVW approaches such as DBoW2 [7]. These methods quantize local features like ORB or SIFT into visual words and enable fast image retrieval based

978-1-6654-5784-2/26/$31.00 ©2026 IEEE

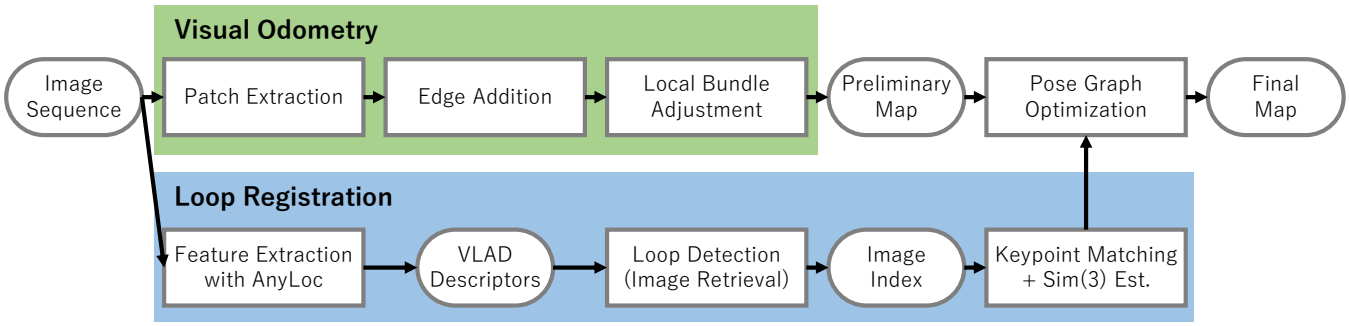

Fig. 1. DPV-SLAM pipeline with AnyLoc-based visual place recognition.

on word frequency, achieving high computational efficiency and processing speed. Such approaches have been adopted in systems like ORB-SLAM [8] and DPV-SLAM. However, BoVW-based methods often suffer from performance degradation caused by illumination changes and viewpoint variations.

To address these challenges, loop detection methods using deep learning have attracted attention. Techniques like NetVLAD [9] and Patch-NetVLAD [10] aggregate local descriptors into global image descriptors, enabling robust visual place recognition. Among these, AnyLoc is trained on large-scale datasets covering scenes across multiple cities, achieving high generalization capability and semantically meaningful feature representations in both structured and unstructured environments.

Furthermore, topological loop detection methods that exploit spatial structure and temporal context have also been proposed. For instance, Topomap [11] captures the spatial structure of environments using graph representations, contributing to loop detection and map consistency. Similarly, the integration of semantic information as demonstrated in SemanticFusion [12] has proven effective in stabilizing recognition under complex environmental conditions.

In summary, existing SLAM systems have gradually evolved from classical BoVW approaches toward learning-based methods for loop detection.

Motivated by this trend, this paper proposes the integration of the deep visual place recognition method AnyLoc into the original DPV-SLAM framework, replacing its BoVW module to enhance the robustness and accuracy of loop detection. Furthermore, an adaptive thresholding mechanism is introduced, enabling the system to dynamically adjust image matching threshold in response to environmental changes. This design allows the system to maintain stable performance across diverse and challenging scenarios.

III. Loop Closure with AnyLoc

A. Overview

Fig. 1 illustrates the pipeline of the proposed method. In the Visual Odometry module, the input image undergoes feature extraction and pose estimation, followed by bundle adjustment to construct a map prior to loop closure.

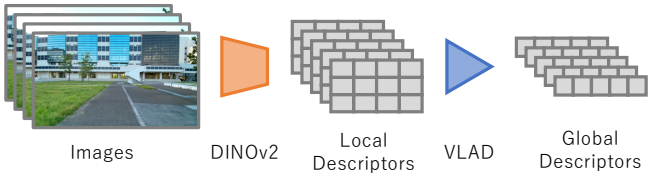

Fig. 2. Extraction of global descriptors using AnyLoc-VLAD-DINOv2.

Running concurrently with this process is the Loop Registration module, which determines the occurrence of loop closures. This module extracts global descriptors using AnyLoc and retrieves multiple candidate images with high similarity scores. Based on a threshold, it decides whether a loop closure has occurred and performs geometric verification. Upon successful verification, the map is optimized through pose graph optimization, generating the final consistent map.

B. Global Descriptor Extraction

Fig. 2 illustrates the extraction of global descriptors. Global descriptors are extracted from keyframe images using AnyLoc-VLAD-DINOv2. In this process, DINOv2 functions as the backbone for visual feature extraction, producing local descriptors from the input images. These local descriptors are then aggregated into highly discriminative global descriptors using the VLAD feature aggregation method. The extracted global descriptors are stored in a database for subsequent retrieval.

C. Adaptive Similarity Threshold Adjustment

In conventional methods such as BoVW-based AnyLoc, the similarity threshold for loop closure decision must be manually set, which affects the system's generalization performance. To address this, we propose an adaptive method for similarity threshold adjustment.

Specifically, during the initial phase, a small set of valid loop closure candidate similarity scores (default: 5) are collected, and their median value is calculated and used as the current threshold (loop_thresh). This threshold reflects the score distribution according to the environment.

Nevertheless, owing to the initial instability of the environment and the influence of subsequent environmental variations, once a new threshold is detected, the system records the maximum threshold observed at that moment

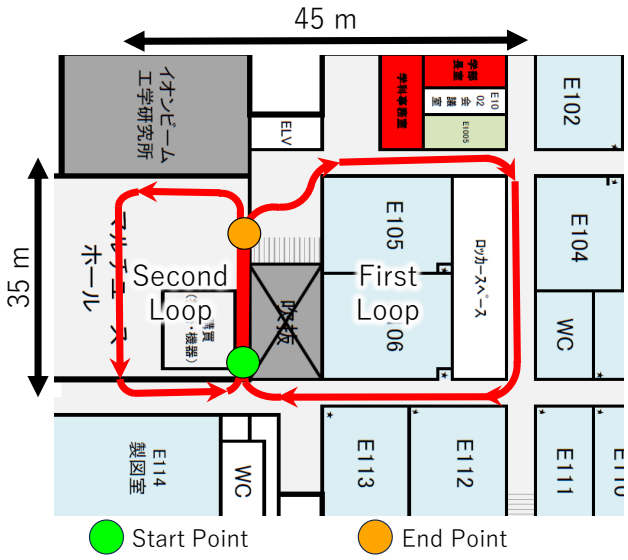

Fig. 3. Corridor environment.

and updates the previous value accordingly. This updating mechanism improves the adaptability and robustness of the loop detection process.

D. Loop Detection and Sim(3) Pose Registration

After the threshold is determined, the similarity between the global descriptor of each keyframe and the query is computed, and keyframes exceeding the loop_thresh are considered as loop candidates. Temporally close frames (e.g., within 50 frames) are excluded, and the candidate with the highest similarity is selected.

Following loop detection, DISK [13] keypoints are extracted and matched using LightGlue [14]. Then, a Sim(3) transformation is estimated via RANSAC [15] combined with the Umeyama algorithm [16]. Geometric verification is considered successful if the number of inliers exceeds a predefined threshold (30 points), and the estimated pose is incorporated into the pose graph as a loop constraint.

E. Pose Graph Optimization

After successful loop detection and geometric verification, the estimated Sim(3) transformation is added into the pose graph as a loop closure constraint. We utilize the CUDA-accelerated block-sparse optimization backend from DPV-SLAM to perform nonlinear pose graph optimization (PGO), achieving efficient and real-time global pose refinement. The optimization runs asynchronously in a separate thread, and upon completion, a callback function updates the entire trajectory, enabling continuous online map optimization.

IV. Experiments

A. Overview

We conducted experiments in several real-world environments and compared our approach with previous work. Specifically, this study aims to answer the following questions:

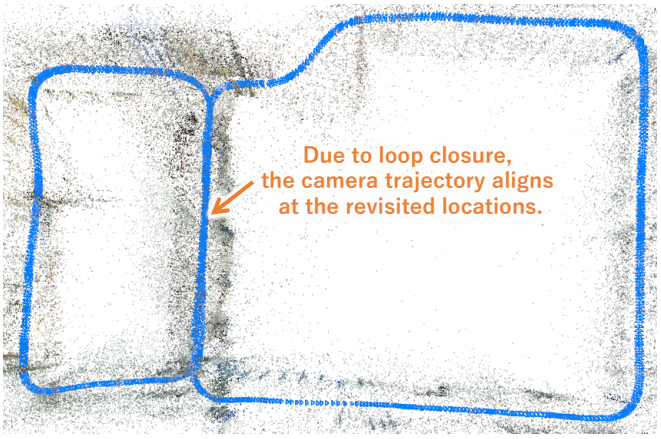

Fig. 4. 3D point cloud map and camera trajectory in the corridor environment (proposed method).

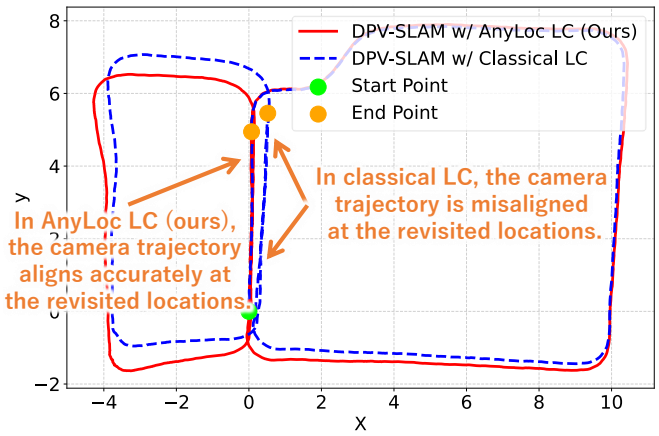

Fig. 5. Comparison of camera trajectories in the corridor environment.

Q1: Does the proposed method improve the accuracy of loop detection compared to the baseline, thereby contributing to a more accurate map?
Q2: Does the proposed method exhibit better generalization capability than previous approach across diverse environments?

B. Environments

Experiments were conducted in both indoor and outdoor environments at the Koganei Campus of Hosei University.

Fig. 3 shows the indoor environment and its trajectory. The indoor environment in Fig. 3 is relatively dark and features multiple occurrences of pedestrian crossings.

Fig. 6 shows the outdoor environment (courtyard) and its trajectory. The outdoor courtyard environment includes multiple sections along the path with similar textures such as grass and concrete.

Additionally, Fig. 9 shows the outdoor environment (between buildings) and its trajectory. This environment also contains several instances of pedestrian crossings.

C. Data Collection and Processing

A Logitech MX BRIO 700 (C1100) camera was used to capture images at a frame rate of 30 Hz. The camera

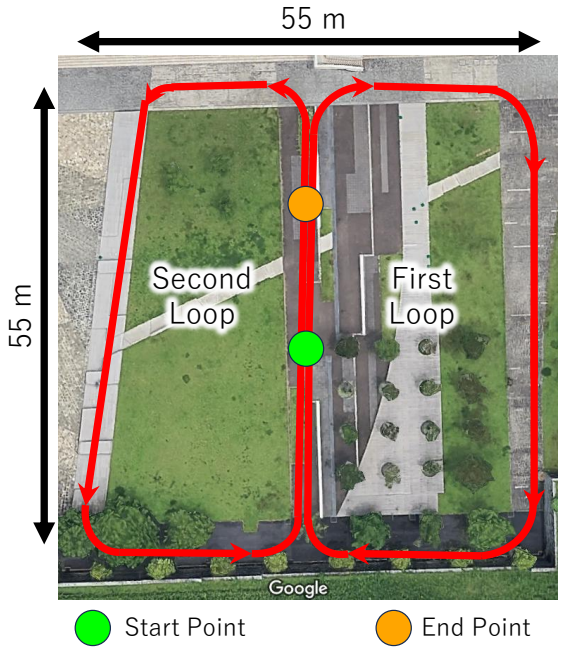

Fig. 6. Outdoor environment (courtyard).

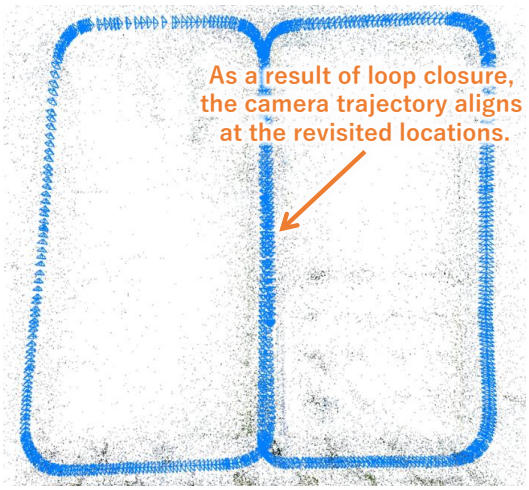

Fig. 7. 3D point cloud map and camera trajectory in the outdoor courtyard (proposed method).

was handheld during walking, and data was recorded using rosbag.

After data collection, experiments were conducted offline. The computer used for processing was a desktop running Ubuntu 22.04, equipped with an Intel Core i7-13700 CPU, 48 GB of RAM, and an NVIDIA GeForce RTX 4070 Ti SUPER GPU with 16 GB of VRAM.

D. Evaluation Methodology

To evaluate the effectiveness of the proposed method, we conducted a qualitative assessment. The main objective of the evaluation is to compare our approach with DPV-SLAM in terms of loop detection accuracy, map reconstruction quality, and overall system performance.

During the evaluation, we adopted an offline experimental

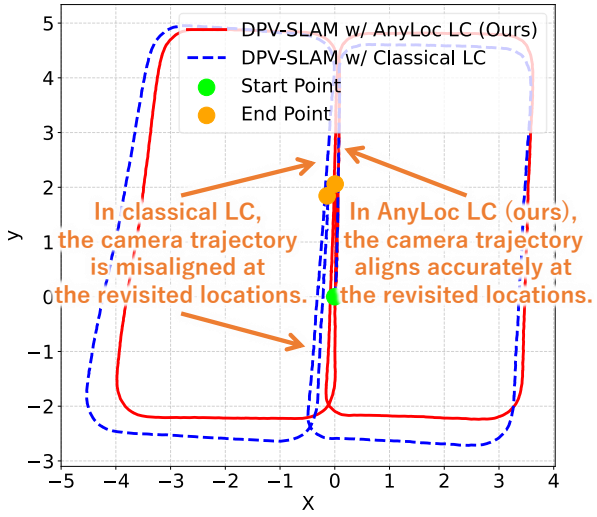

Fig. 8. Comparison of camera trajectories in the outdoor environment (courtyard).

approach. Data were collected using a unified sensor setup and were separately processed by both the proposed method and the original DPV-SLAM system. By comparing the loop detection performance and map quality on the same dataset, we aimed to further validate the advantages of our method in practical scenarios. The mapping results were evaluated based on alignment errors with respect to the ground-truth environment map, and visual comparisons were also conducted to examine the completeness and consistency of the reconstructed maps.

Furthermore, we conducted a comparative analysis of image retrieval results produced by different loop detection methods in the three experimental environments, thereby revealing the differences in loop detection accuracy among the methods.

E. Experimental Results

Fig. 4 shows the 3D point cloud map and camera trajectory of the proposed method in the indoor environment. Fig. 5 illustrates a comparison of camera trajectories in the same environment.

Next, Fig. 7 shows the 3D point cloud map and camera trajectory of the proposed method in the outdoor environment (courtyard), while Fig. 8 illustrates a comparison of camera trajectories.

The previous method failed to detect valid loops, resulting in significant misalignment of camera trajectories at revisited locations along the loop. In contrast, the proposed method successfully detected loops while maintaining high positional accuracy, causing the camera trajectories at revisited locations to overlap after loop closure. These results demonstrate that our method effectively improves the accuracy of loop detection, thereby leading to more precise map building. This addresses **Q1**.

Then, Fig. 10 shows the 3D point cloud map and camera trajectory in the outdoor environment (between buildings), while Fig. 11 illustrates a comparison of camera trajectories.

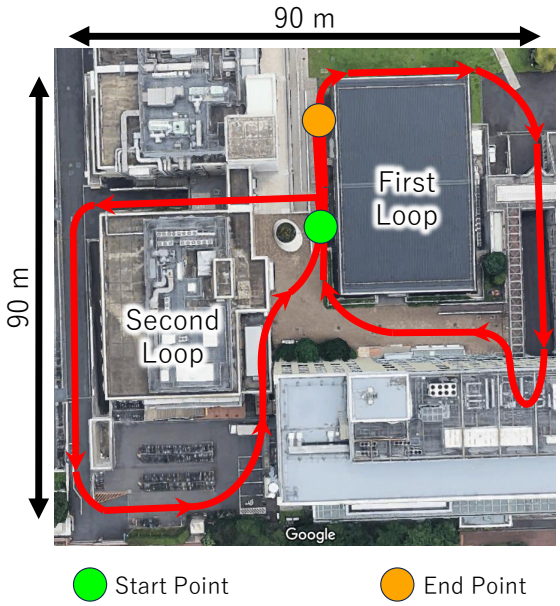

Fig. 9. Outdoor environment (between buildings).

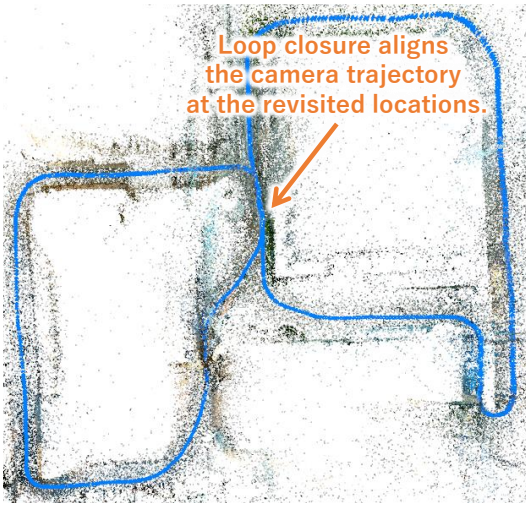

Fig. 10. 3D point cloud map and camera trajectory in the outdoor area between buildings (proposed method).

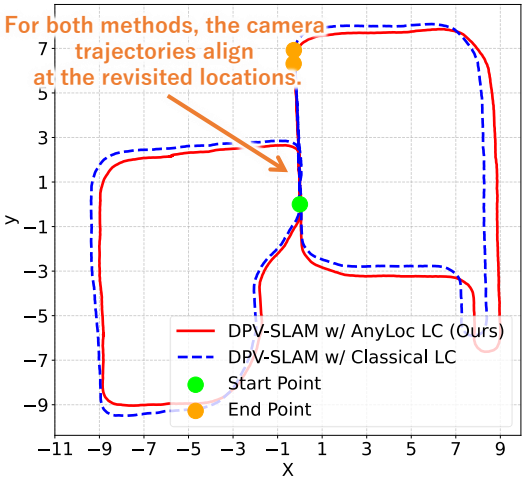

Fig. 11. Comparison of camera trajectories in the outdoor environment (between buildings).

(a) Query image (b) Retrieved result

Fig. 12. Loop closure retrieval performance (AnyLoc LC). A large number of correspondences are detected between the two images, and a subset of them is highlighted. These matched keypoints are subsequently used for registration and pose graph optimization.

(a) Query image (b) Retrieved result

Fig. 13. Loop closure retrieval performance (Classical LC). The figure displays the query image and its matches, with similar feature regions highlighted. Although the most similar images are obtained by adjusting the threshold, the number of matched keypoints is insufficient and fails geometric verification.

In this environment, no significant differences were observed between the conventional method and the proposed method. Both methods correctly detected loops, and loop closure resulted in overlapping camera trajectories at revisited locations.

Furthermore, as shown in Fig.12, the proposed method consistently retrieves highly similar images that successfully pass geometric verification, enabling reliable loop closure. In contrast, as shown in Fig.13, although the classical method detects the loop closure, the retrieved image cannot pass the final geometric verification and thus the loop closure cannot be achieved. These results highlight the robustness and effectiveness of our method in challenging and complex environments.

As illustrated in Fig. 14(a)–(c), our method consistently identifies a greater number of loop closures than the conventional approach across all three environments. Although adjusting the detection threshold of the classical method can moderately increase its detection count, its performance remains inferior to ours. These observations indicate that the proposed method substantially enhances the robustness and overall capability of loop detection, thereby improving the quality and accuracy of the generated maps.

Through experiments in three different environments, the proposed method outperformed conventional DPV-SLAM in both loop detection accuracy and trajectory precision. Even under challenging conditions such as low lighting, repetitive textures, and dynamic interference, our method demonstrated higher robustness and stability, indicating strong potential for real-world deployment. This addresses **Q2**.

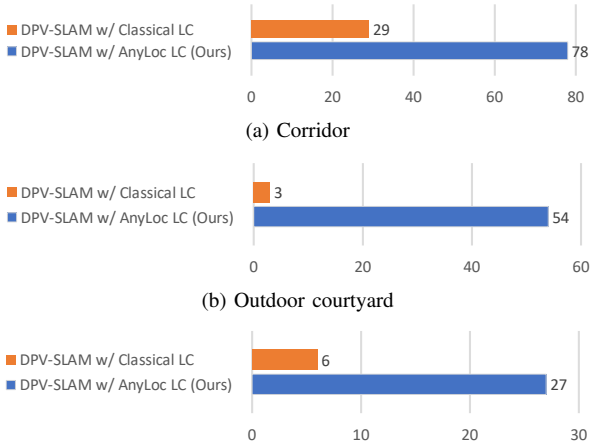

Fig. 14. Comparison of the number of detected loops in different environments.

V. Conclusion

In this paper, we propose integrating AnyLoc, a visual place recognition method, into DPV-SLAM to replace the conventional Bag of Visual Words (BoVW) based loop detection. Furthermore, we develop a loop closure pipeline that combines adaptive similarity threshold adjustment with geometric verification, and evaluate its effectiveness through real-world experiments. Experimental results in real environments demonstrate a significant improvement in loop detection accuracy when combining AnyLoc with DPV-SLAM.

One limitation of the proposed method is its substantial requirement for GPU computational resources. Although the system can operate on CPUs, the processing speed decreases considerably.

In future work, we plan to improve loop detection speed by excluding unnecessary keyframe global descriptors using positional information. Additionally, we aim to evaluate our method on standard benchmark datasets.

References

[1] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, "AnyLoc: Towards Universal Visual Place Recognition," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1286–1293, 2023.

[2] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An Efficient Alternative to SIFT or SURF," in *Proc. of IEEE Int. Conf. on Computer Vision (ICCV)*, 2011.

[3] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int. J. of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] K. Adachi, Y. Hara, and S. Nakamura, "Simulation Evaluation of Monocular Visual SLAM: ORB-SLAM3, DROID-SLAM, DPVO, and DPV-SLAM," in *Proc. of IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2025.

[5] L. Lipson, Z. Teed, and J. Deng, "Deep Patch Visual SLAM," in *Proc. of European Conf. on Computer Vision (ECCV)*, 2024.

[6] Z. Teed, L. Lipson, and J. Deng, "Deep Patch Visual Odometry," in *Proc. of Annual Conf. on Neural Information Processing Systems (NeurIPS)*, 2023.

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

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

[9] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN Architecture for Weakly Supervised Place Recognition," in *Proc. of IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-NetVLAD: Multi-Scale Fusion of Locally-Global Descriptors for Place Recognition," in *Proc. of IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[11] F. Blöchliger, M. Fehr, M. Dymczyk, T. Schneider, and R. Siegwart, "Topomap: Topological Mapping and Navigation Based on Visual SLAM Maps," in *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2018.

[12] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "SemanticFusion: Dense 3D Semantic Mapping with Convolutional Neural Networks," in *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017.

[13] M. Tyszkiewicz, P. Fua, and E. Trulls, "Disk: Learning Local Features with Policy Gradient," in *Proc. of Annual Conf. on Neural Information Processing Systems (NeurIPS)*, 2020.

[14] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "Lightglue: Local Feature Matching at Light Speed," in *Proc. of IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.

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

[16] S. Umeyama, "Least-Squares Estimation of Transformation Parameters between Two Point Patterns," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, 1991.