

Visual Loop Closure Detection Through Deep Graph Consensus

Martin Büchner¹, Liza Dahiya², Simon Dorer¹, Vipul Ramtekkar²,
Kenji Nishimiya², Daniele Cattaneo¹, Abhinav Valada¹

Abstract—Visual loop closure detection traditionally relies on place recognition methods to retrieve candidate loops that are validated using computationally expensive RANSAC-based geometric verification. As false positive loop closures significantly degrade downstream pose graph estimates, verifying a large number of candidates in online simultaneous localization and mapping scenarios is constrained by limited time and compute resources. While most deep loop closure detection approaches only operate on pairs of keyframes, we relax this constraint by considering neighborhoods of multiple keyframes when detecting loops. In this work, we introduce LoopGNN, a graph neural network architecture that estimates loop closure consensus by leveraging cliques of visually similar keyframes retrieved through place recognition. By propagating deep feature encodings among nodes of the clique, our method yields high precision estimates while maintaining high recall. Extensive experimental evaluations on the TartanDrive 2.0 and NCLT datasets demonstrate that LoopGNN outperforms traditional baselines. Additionally, an ablation study across various keypoint extractors demonstrates that our method is robust, regardless of the type of deep feature encodings used, and exhibits higher computational efficiency compared to classical geometric verification baselines. We release our code, supplementary material, and keyframe data at <https://loopgnn.cs.uni-freiburg.de>.

I. INTRODUCTION

Simultaneous localization and mapping (SLAM) is a fundamental task in mobile robotics, serving as the foundation for various downstream components such as navigation [1], [2] and task planning [3]. While odometry methods provide local estimates of robot pose changes [4], [5], they are inherently subject to accumulating errors, leading to considerable drift and causing downstream tasks to fail due to map inconsistencies. Loop closure detection aims to address this problem by identifying previously visited places and estimating the relative transformation between the current image and the past traversal. This transformation serves as a *hard* constraint throughout pose graph optimization, thereby reducing drift and mitigating errors stemming from noisy odometry.

In practice, the majority of established loop closure detection approaches integrated into visual SLAM rely on image retrieval methods [7] that are either vocabulary-based, such as DBoW [8], or global descriptor-based, such as the family of VLAD feature approaches [6], [9]. Broadly, these approaches fall under the umbrella of visual place recognition

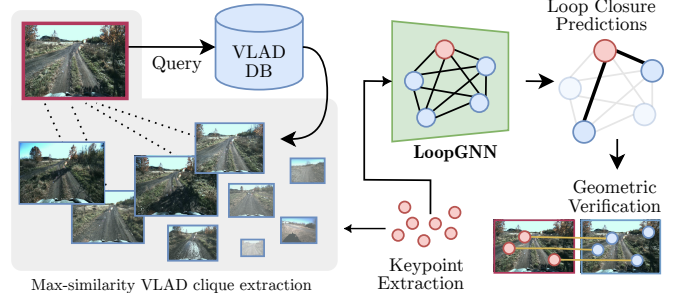


Fig. 1. Given a query keyframe, the set of closest keyframes is retrieved through classical place recognition methods such as VLAD [6]. In this work, we introduce LoopGNN, a novel graph neural network that produces precise loop closure estimates via deep graph consensus among the obtained closest set of keyframes. We find that our method scales across various types of deep keypoint encodings and outperforms baselines relying on only pairs of keyframes to identify loop closures.

(VPR). Given a query keyframe, those methods provide sets of similar camera images that may represent potential loop closures. As such, VPR approaches are evaluated in terms of high recalls. In order to construct actual loop closure constraints, those proposals require filtering to reduce false positives via compute-intense geometric verification. The most common approach for this task is the Random Sample Consensus (RANSAC) [10] algorithm under an epipolar constraint [11]. Due to its iterative formulation, RANSAC becomes computationally expensive when evaluating a significant number of loop closure candidates, effectively limiting its deployment in real-time applications [12]. Additionally, we observe that while deep learning has been successfully employed for place recognition, this is not the case for robust loop closure detection [13]. This is primarily due to visual loop closure detection being heavily affected by perceptual aliasing and being rather sensitive to varying camera poses. However, we argue that previous deep loop closure detection methods would benefit from not considering only pairs of frames [12], [14]–[16], but larger neighborhoods.

In this work, we present a novel deep loop closure detection approach that operates on sets of extracted keyframes given a query frame and estimates loop closures through deep graph consensus. By identifying maximum-similarity cliques given a query frame and its most-similar keyframes, we construct graphs connecting all retrieved frames among each other. We use per-frame keypoint features as node features in our graph and propose a novel graph neural network (GNN) architecture, termed LoopGNN, that performs neural message passing to allow information exchange among retrieved keyframes. By combining retrieved keyframes, we aim to learn a deep graph consensus under the assumption that similar locations are

¹ Department of Computer Science, University of Freiburg, Germany.

² Solution System Development Center in Honda R&D Co., Ltd., Japan.

This paper provides a supplementary material at <https://loopgnn.cs.uni-freiburg.de>. This work was supported by Honda R&D Co., Ltd., an academic grant from NVIDIA and the BrainLinks-BrainTools Center of the University of Freiburg. We sincerely thank Fabian Schmidt and Nick Heppert for discussions.

retrieved as groups of frames during place recognition. We demonstrate that this ultimately increases the precision of predicted loop closures and decreases the number of pairs that need to undergo geometric verification.

We demonstrate the effectiveness of our approach on a specifically curated split of the outdoor all-terrain vehicle driving dataset TartanDrive 2.0 [17] featuring largely unstructured environments that induce a significant degree of perceptual aliasing. Second, we evaluate on the NCLT dataset [18] and ablate the main design choices of our proposed framework. As part of this work, we make the following contributions:

- We propose graph similarity-based consensus estimates for precise loop closure identification through graph neural networks and couple this with geometric verification.
- We provide extensive comparisons against previous place recognition and learning-based loop closure detection approaches and delineate the computational benefits of our method over previous work.
- We establish a split of the TartanDrive 2.0 dataset that paves the way forward for research on visual-inertial SLAM under dynamic motion in unstructured environments.
- We make the reprocessed datasets, the learning framework, and code publicly available at <https://loopgnn.cs.uni-freiburg.de>.

II. RELATED WORK

In this section, we review related work on visual correspondence, visual loop closure detection, and graph-based methods for loop closure detection.

Visual Correspondence: Traditional methods for visual correspondence rely on handcrafted feature descriptors such as ORB [19], SIFT [20], and SURF [21]. These approaches often struggle in environments with occlusions, illumination changes, and strong viewpoint variations. To address these challenges, deep learning-based feature descriptors have emerged as a powerful alternative. LIFT [22] was among the first frameworks to jointly optimize keypoint detection and descriptor extraction. Building on earlier deep learning approaches, SuperPoint [23] introduced a fully convolutional network for self-supervised feature extraction that is used by SuperGlue [24] to refine feature matching using graph neural networks. Recent methods such as XFeat [25] and RoMa [26] improve efficiency and robustness in challenging conditions. XFeat uses a lightweight CNN for fast keypoint extraction, while RoMa leverages DiNOv2 [27] for accurate dense feature matching in complex environments.

Visual Loop Closure Detection: Visual loop closure detection often relies on image retrieval techniques where a single query image is matched against a database of previously observed locations. Traditional methods for visual loop closure rely on BoW (Bag-of-Words) or DBoW2 [8], which approximate nearest neighbors in the image feature space and construct a visual vocabulary for place recognition. Another family of approaches utilizes global descriptor-based methods, where an entire image is represented by a compact vector for efficient matching. A well-known method in this family

is VLAD [6], which aggregates local feature descriptors into a fixed-length representation and tends to be more robust than BoW methods. NetVLAD [9] extends VLAD by incorporating a learnable feature aggregation mechanism using deep networks, improving performance in challenging situations. However, the aforementioned approaches usually require computationally heavy geometric verification through RANSAC [10].

Place recognition techniques are optimized for high recall values and thus retrieve a broad set of candidate matches to maximize coverage [9], [28], [29]. However, for robust localization, high precision is crucial to avoid incorrect loop closures that can introduce errors into the pose graph. Only few learning-based methods attempt to directly address the problem of loop closure detection by not depending on prior place recognition methods. For instance, Calc [15] employs a self-supervised autoencoding scheme where random projective transformations are applied to camera images, and a neural network is trained to predict Histogram of Oriented Gradient (HOG) descriptors of the original image given the augmented one. Calc2.0 [30] additionally includes a variational autoencoder and a triplet-loss embedding network to embed images and cluster same-place images together. Siamese-ResNet [31], on the other hand, employs a Siamese network to train a contrastive loss for robust image similarity learning. Finally, LoopCNN [14] extracts high-dimensional image descriptors using a pre-trained CNN and compares PCA-reduced features across image pairs.

Graph Estimation for Loop Closure Detection: Various methods have investigated graph structures to describe either the scene morphology or objects including their semantics depicted in images [32]–[36] in order to find feature-rich representations for matching. Another branch of work entirely focuses on modeling ambiguous observations directly rather than relying on post-hoc corrections by employing graph-based probabilistic reasoning. This enables the identification of perceptual aliases, thereby reducing false positive loop closures [37], [38]. In contrast to the aforementioned works, our approach focuses on employing deep graph structures at the intersection of visual place recognition and loop closure detection. In detail, we consider graphs of retrieved images given a query image in order to learn consensus estimates for loop closure detection.

III. LOOPGNN METHOD

In this section, we detail our proposed LoopGNN approach that performs loop closure detection by estimating deep graph consensus and predicting relative transformations. Our approach consists of a method for extracting similarity cliques given a query keyframe and a graph neural network for encoding keyframe features as detailed in Fig. 2.

A. Keyframe Extraction

In order to sample keyframes, we follow previous work [39] that extracts frames when a cumulative distance threshold of 0.5 m is exceeded. For each sequence, we obtain a posed graph of keyframes $\mathcal{G}_K = \{V_i\}$, where $i \in \{1, 2, \dots, N\}$,

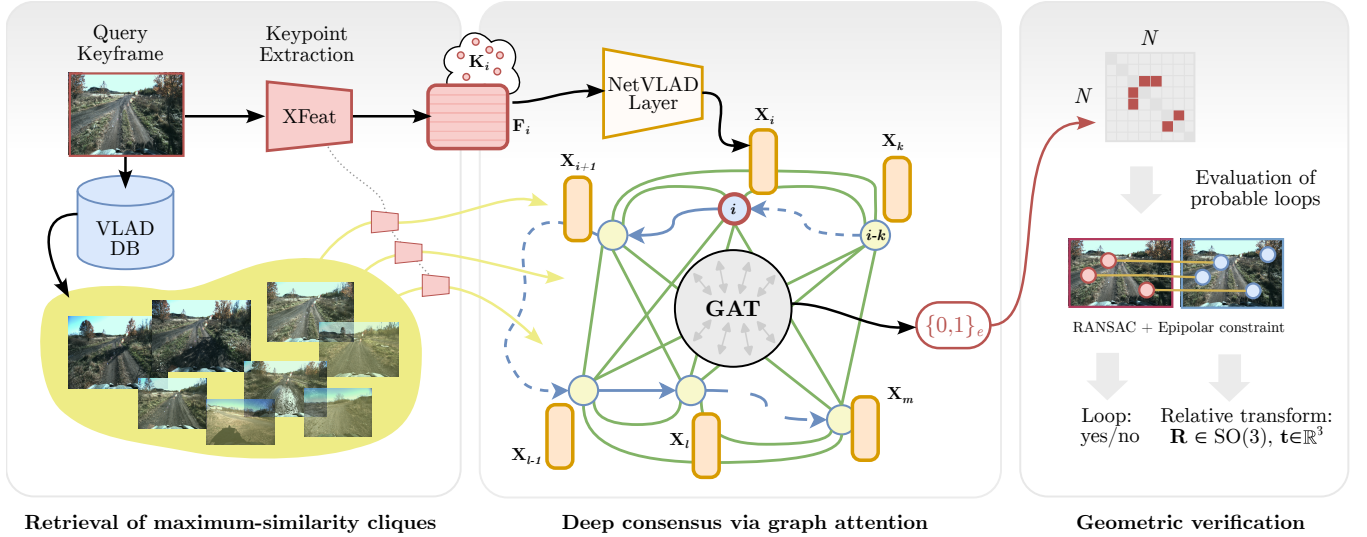


Fig. 2. Overview of our LoopGNN approach: We create keyframes from robot trajectories and utilize a deep keypoint extractor such as XFeat [25] to obtain keypoints for each image. Next, we fit a VLAD-based place recognition model allowing robust and fast retrieval of similar frames given a query frame (left). In the following, given a query frame, we independently encode the keypoint descriptors of all frames (query and retrieved ones) using a NetVLAD layer and construct a neighborhood graph. We feed this attributed graph into a graph attention network in order to produce a deep consensus regarding loop closures among keyframes of the neighborhood (middle). Finally, we extract the set of highest-scoring edge-wise predictions of the network and validate pairs of frames using RANSAC-based geometric verification (right).

and N is the total number of keyframes in the sequence. For each of those keyframes V_i we assume access to its camera image $\mathbf{I}_i \in \mathbb{R}^{H \times W \times C}$. Next, we feed $\mathbf{I}_i$ through any kind of lightweight keypoint feature extractor $f_{KP}(\mathbf{I}_i) = \{\mathbf{K}_i, \mathbf{F}_i\}$, which provides us with a set of keypoints $\mathbf{K}_i \in \mathbb{R}^{N_K \times 2}$ and their associated keypoint descriptors $\mathbf{F}_i = [\mathbf{f}_{i,1}, \mathbf{f}_{i,2}, \dots, \mathbf{f}_{i,N_K}] \in \mathbb{R}^{N_K \times N_{KD}}$, where N_K may vary per image.

B. Global Feature Aggregation

While the obtained set of keypoint descriptors represents a reliable basis for re-identification, it is generally too high-dimensional to be used for online similarity search. In order to facilitate fast similarity search over $\mathcal{G}_K$, we employ the Vector-of-Linearly-Aggregated-Descriptors (VLAD) technique. By clustering the feature descriptors $\mathbf{F}_i$ of the generated key-points using N_C cluster prototypes we construct a vocabulary that allows the generation of image-level features. Since the keypoints may stem from non-Euclidean manifolds, we employ either a cosine similarity-based or Euclidean distance-based variant. We obtain image-level VLAD features by computing the sum of residuals of the keypoint descriptors of an image to each of the cluster centers $\mathbf{c}_j$, with $j \in \{1, 2, \dots, N_C\}$, as:

$$\mathbf{D}_{i,j} = \sum_{k=1}^{N_K} \beta_j(\mathbf{f}_{i,k}) \cdot \cos(\mathbf{f}_{i,k}, \mathbf{c}_j), \quad (1)$$

where in the case of hard-assignment, that we employ, β_j equals 1 if $\mathbf{f}_{i,k}$ is assigned to $\mathbf{c}_j$ and 0 otherwise. Finally, we craft image-level VLAD descriptors $\mathbf{D}_i$ via concatenation of cluster-aggregated descriptors and normalization as detailed in prior work [6]. Thus, each of the keyframes $V_i \in \mathcal{G}_K$ is

parametrized as follows:

$$V_i = \{\mathbf{I}_i, \mathbf{F}_i, \mathbf{K}_i, \mathbf{D}_i\}. \quad (2)$$

C. Graph Construction

In the following, we define *maximum-similarity cliques* by performing a dense similarity search across all global keyframes descriptors $\mathbf{D}_i$. This provides us with a set of maximum-similar keyframes given a query keyframe q . At inference time, we query the existing keyframes' VLAD descriptors based on the query descriptor using cosine similarities. Next, we identify the closest k -% of keyframes to the query keyframe and construct an interconnected, undirected graph of keyframes $\mathcal{G}(i = q) = (V_q, E_q)$ that constitutes a maximum-similarity clique. Given the query keyframe, we partition all created edges given their incidence to the query keyframe and define *query edges* as depicted in Fig. 3. In the following, we detail how to obtain a deep consensus estimate for improved loop closure detection by passing messages within the clique.

D. Graph Attention Network

As part of this work, we propose a novel approach that provides a graph consensus by learning expressive node features from keypoint descriptors. To do so, we require an order-invariant operation that aggregates descriptors. Given the successful use of the VLAD clustering mentioned above, we employ its learned counterpart, namely NetVLAD embeddings [9], to aggregate keypoint descriptors $\mathbf{F}_i$ into preliminary node features, see Fig. 2. In the next step, we downproject the NetVLAD features to form node features $\mathbf{X} = [\mathbf{X}_1, \dots, \mathbf{X}_N]^T \in \mathbb{R}^{N \times 256}$ and perform a series of message passing steps. Throughout our experiments, we found that the graph attention mechanism (GAT) [40] performs

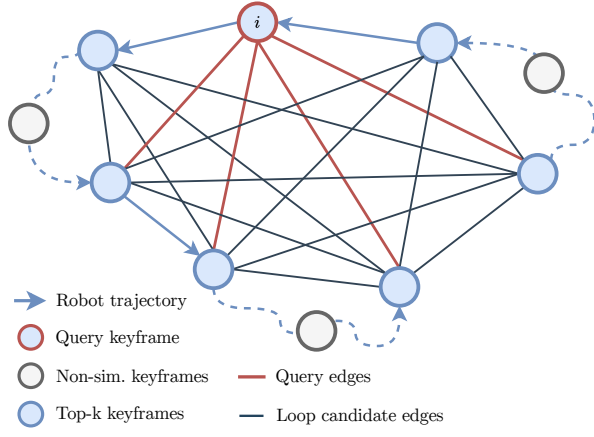


Fig. 3. Maximum-similarity clique construction: Given robot trajectory, we extract keyframes at fixed distance intervals. Given a query keyframe, we retrieve the set of top-k closest keyframes using VLAD-based place recognition. We impose interconnected edges and take the obtained graph $\mathcal{G}(q) = (V_q, E_q)$ as input to our LoopGNN pipeline. At the output, we only consider the edge scores of the query edges denoted here.

best in learning descriptive node features through graph interactions, ultimately providing more reliable loop closure predictions. Within each GAT layer, learned attention scores are estimated to produce weighted transformations of the original node features as follows:

$$\mathbf{X}_m^{(l+1)} = \sigma \left(\sum_{n \in \mathcal{N}(m)} \alpha_{mn} \mathbf{W}^{(l)} \mathbf{X}_n^{(l)} \right), \quad (3)$$

where $\mathbf{X}_m^{(l+1)}$ is the updated node feature, $\mathbf{W}^{(l)}$ is the learnable weight matrix of the involved layer l , $\mathcal{N}(m)$ represents the 1-hop neighborhood of node m , and α_{mn} is the attention weight between node m and n . We chain multiple GAT layers to iteratively refine node features. After propagating messages for L iterations, we concatenate the node features $\mathbf{X}^{(L)}$ of nodes incident to each edge in order to form edge features that allow regressing binary scores that constitute loop closure predictions. We supervise the network by employing a binary cross-entropy loss, where the ground truth is one if the relative pose between the query image and the considered loop candidate is lower than a threshold of 4 m and 30° . We additionally incorporate loop closure candidates across sequences to allow for improved robustness. Furthermore, we found that the accuracy of predictions is particularly high on the query edges as visualized in Fig. 3. Given that finding, we only consider the scores of those edges during inference. The remaining set of edge scores is disregarded and thus only serves the consensus estimate of the query edges.

E. Geometric Verification

In the following, we employ geometric verification to first, increase robustness of the predicted loop scores, and second, obtain relative camera transforms. We analyze the score distribution of the GNN outputs and choose a threshold that preserves high recalls across multiple sequences. Next, we validate keyframe pairs (i, j) whose score is higher than

the chosen threshold, and retrieve the associated keypoints $\mathbf{K}_i, \mathbf{K}_j$, along with their corresponding descriptors $\mathbf{F}_i, \mathbf{F}_j$. Next, we compute mutual keypoint correspondences among the two images. In order to verify whether each candidate loop is geometrically valid, we compute the underlying fundamental matrix $\mathcal{F}$ that satisfies the epipolar constraint:

$$\hat{\mathbf{K}}_j^T \mathcal{F} \hat{\mathbf{K}}_i = 0, \quad (4)$$

where $\hat{\mathbf{K}}_i$ and $\hat{\mathbf{K}}_j$ denotes the set of matched keypoints in image $\mathbf{I}_i$ and $\mathbf{I}_j$, respectively. We repeat this optimization iteratively using the robust RANSAC estimator [10] to obtain the transformation with the highest percentage of inlier keypoints. Given that a particular projection threshold of 50% is exceeded, we further obtain the associated relative camera transformation, i.e., the relative rotation $\mathbf{R} \in SO(3)$ and translation $\mathbf{t} \in \mathbb{R}^3$ by computing the corresponding essential matrix:

$$\mathcal{E} = \mathcal{K}_C^T \mathcal{F} \mathcal{K}_C, \quad (5)$$

where $\mathcal{K}_C \in \mathbb{R}^{3 \times 3}$ denotes the camera intrinsic matrix. Finally, we perform Singular Value Decomposition (SVD) of $\mathcal{E}$ and obtain the corresponding rotational and translational component $(\mathbf{R}, \mathbf{t})$ up to scale.

IV. EXPERIMENTAL EVALUATION

In this section, we demonstrate the capabilities of our proposed LoopGNN framework on two distinct datasets, *TartanDrive 2.0* [17] and *NCLT* [18], and compare it against previous VPR, deep loop closure detection, and classical feature matching and geometric verification methods.

A. Datasets

In the following, we provide details on the datasets that we employ for evaluation:

TartanDrive 2.0: The TartanDrive 2.0 dataset [17] (TD2) is a large-scale off-road terrain driving dataset tailored for self-supervised learning of dynamics models. It consists of unstructured and texture-less sequences. Different from its original intended use case, in this work, we propose a dataset split that facilitates research in the domain of visual-inertial SLAM. Specifically, we select sequences that contain sufficient IMU initialization phases at their beginnings where the vehicle does not move and that contain enough loop closures. At the same time, we believe that the amount of perceptual aliasing and high speed driving behavior represents a highly challenging test bed for visual odometry and mapping research. In this work, we have constructed a train/test-split that consists of 7 training and 3 test scenes, respectively containing 16625 and 23433 keyframes. As described in Sec. III-A, we sample keyframes following the procedure introduced in VINS-Fusion [39]. We list all considered sequences in Tab. I.

NCLT: The NCLT dataset [18] is a large-scale collection of 27 LiDAR-camera sequences recorded with a mobile robot across indoor and outdoor environments. It contains omnidirectional imagery, 3D LiDAR, IMU, and GNSS measurements, and provides global ground-truth robot trajectories

TABLE I

OVERVIEW OF THE SELECTED TARTANDRIVE 2.0 DATASET SEQUENCES

Split	Sequence	# Frames
Train	2023-11-14-14-26-22_gupta	1272
	2023-11-14-14-34-53_gupta	1632
	2023-11-14-14-45-20_gupta	823
	2023-11-14-14-52-22_gupta	2066
	2023-11-14-15-02-21_figure_8	3782
	gupta_skydio_2023-09-14-11-03-09	853
	turnpike_2023-09-12-12-53-32	6197
Test	figure_8_2023-09-13-17-24-26	4429
	figure_8_morning_2023-09-12-10-37-17	11065
	figure_8_morning_slow_2023-09-12-11-06-32	7939

All selected TD2 sequences contain at least one loop and are captured under varying weather conditions. All train sequences cover distinct regions while the test sequences show partial overlap.

TABLE II

OVERVIEW OF THE NCLT DATASET SEQUENCES

Split	Sequence	# Frames	Daytime	Weather
Train	2012-01-22	5321	afternoon	cloudy
	2012-02-02	5932	afternoon	sunny
	2012-02-18	5383	evening	sunny
	2012-05-11	5635	midday	sunny
Test	2012-02-12	5522	midday	sunny
	2012-02-19	5531	midday	partly cloudy
	2012-03-31	5661	midday	cloudy
	2012-05-26	5632	evening	sunny

The selected NCLT sequences covering diverse environmental conditions, including variations in daytime, sky, foliage, and snow. The sequences are split following LiSA [41], with the upper four used for training and the lower four for validation.

optimized across all sequences. Each sequence covers the same area traversed at different times across all four seasons and features dynamic obstacles, viewpoint variations, and changing lighting conditions. In our experiments, we follow the train/test split of LiSA [41] with four sequences used for training and four sequences used for validation. Since the contained sequences are very long, we limit the number of keyframes via position-based cropping to a subset of the scene, which yields between 4000 to 6000 keyframes per sequence under the same keyframe generation scheme as used for TD2. All NLCT sequences used are listed in Tab. II.

B. Metrics

Compared to the task of VPR, which is typically evaluated only in terms of recall, we compare our method against a number of baselines both in terms of precision and recall. In more detail, we compute the Maximum Recall (MR), which constitutes the highest possible recall under a precision of 100%, i.e., without incurring any false positive predictions. In addition, we also report the Average Precision (AP) to quantify the integral under the precision-recall curve, as done in previous loop closure detection methods [12], [15]. In addition, we measure the predicted pose accuracy in terms of Absolute Translation Error (ATE) and Relative Pose Error (RPE). Since we do not regress the scale of the explored scene, the ATE measurements are reported up-to-scale, i.e., we



Fig. 4. A selection of various keyframes of the TD2 sequence figure_8_2023-09-13-17-24-26 underlining the degree of perceptual aliases contained.

normalize translation vectors to unit length and compare their L2 distance. The RPE is computed as the relative geodesic distance on $SO(3)$:

$$\theta = \cos^{-1} \left(\frac{\text{tr}(\mathbf{R}_{gt}^T \hat{\mathbf{R}}) - 1}{2} \right), \quad (6)$$

where $\mathbf{R}_{gt}$ represents the relative ground truth rotation and $\hat{\mathbf{R}}$ denotes the predicted relative camera rotation.

C. Implementation Details

Keypoint Extraction and VLAD Clustering: Since our proposed LoopGNN architecture is agnostic to the choice of keypoint extractor, we evaluate its performance gain across four different traditional and learning-based extractors, namely SIFT [20], SuperPoint [23], ALIKED [42], and XFeat [25]. Across all keypoint types, we fit VLAD vocabularies using 64 cluster prototypes and up to 2048 keypoints per image. We use cosine-similarities for clustering in the case of XFeat, ALIKED, and SuperPoint given their deep feature embedding characteristic and Euclidean distances for SIFT. For all deep keypoint extractors, we find retrieval rates of at least 94.5% within the top 1% of keyframes. Nonetheless, we note that this does not imply that the true entirety of actual loops is retrieved as just 1% of keyframes is evaluated.

Network Architecture: We train the proposed network for 10 epochs using a batch size of 2, i.e., two retrieved neighborhoods, a learning rate of $1e-4$, and 6 message passing steps. During training, we apply a dropout probability of 0.2 and thus sample various node neighborhoods stochastically to increase robustness. To prevent overfitting on the training dataset, we employ early stopping based on the per-batch AP and MR. During training, we consider both intra- and inter-sequence loops to increase robustness to lighting and viewpoint variations. Given the high amount of potential keyframe pairs, we evaluate only intra-sequence loops on the test set for both *NCLT* and *TartanDrive 2.0*.

D. Baselines

We compare our proposed method LoopGNN with a set of deep loop closure detection baselines, namely Calc [15], LoopCNN [14], and our own constructed baseline Pair-NetVLAD. In addition, we examine LoopGNN in comparison to two prominent visual place recognition methods, VLAD [6] and DBow2 [8]. Finally, we assess the performance when employing robust keypoint matching and RANSAC for geometric verification.

Calc: Calc [15] uses an unsupervised deep learning framework for loop closure detection as detailed in Sec. II. We fine-tune the base Calc model on the TartanDrive 2.0 and NCLT training splits, respectively.

LoopCNN: LoopCNN [14] utilizes a pre-trained convolutional neural network model that is not fine-tuned on target datasets, which we make similar use of.

Pair-NetVLAD: We present a naive baseline method called Pair-NetVLAD by eliminating the message passing stage of our proposed LoopGNN network, see Sec. III-D. Under a GNN depth of zero, the node features do not benefit from feature propagation within neighborhoods, thus disabling the proposed graph consensus mechanism. Instead, the Pair-NetVLAD baseline produces keyframe embeddings that are stacked for each pair of keyframes and regressed using an MLP to yield a binary score per edge.

XFeat + RANSAC: In addition to the deep loop closure detection baselines, we provide experimental results using a classical keypoint matcher relying on XFeat [25] and RANSAC under an epipolar constraint for robust geometric verification, as utilized in previous studies [12]. Since we employ XFeat’s deep keypoint descriptors in LoopGNN, we believe that this comparison is representative.

VLAD & DBow2: We also assess the performance of two prominent VPR methods, VLAD [6] and DBow2 [8].

E. Visual Loop Closure Detection

In the following, we compare the performance of LoopGNN against the set of baselines described in Sec. IV-D on the TD2 and NCLT datasets (Sec. IV-A). Across all experiments, we first retrieve the top-1% neighborhoods by comparing global VLAD descriptors as stated in Sec. III-B and report the predictive performance only among the retrieved neighboring frames, not against all possible loops. This is due to the sheer number of possible combinations to be evaluated reaching 10^8 keyframe pairs for the longest TD2 test sequence.

In general, we regard TD2 to be the vastly more challenging dataset as it exhibits less structural features, especially on the test set, as shown in Fig. 4. As detailed in Tab. III, we observe that the two VPR methods, DBow2 using ORB keypoints [19] and VLAD utilizing XFeat features [25] generally yield relatively high average precision given that they are optimized towards high recalls. However, we note that both methods produce subpar maximum recalls at precision equal to 100%. As expected, this indicates that VPR methods are not robust to perceptual aliases and require downstream geometric verification to produce accurate loop closures.

TABLE III
LOOP CLOSURE DETECTION PERFORMANCE ON TARTANDRIVE 2.0

	Model	AP [%]	MR [%]	RPE [deg]	ATE [1]
VPR	DBow2 [8]	40.84	00.34	-	-
	VLAD [6]	70.85	00.00	-	-
LCD	Calc [15]	57.56	02.79	-	-
	Pair-NetVLAD	43.38	06.39	-	-
	LoopCNN [14]	66.26	01.55	-	-
	LoopGNN (ours)	84.63	45.52	-	-
GV	XFeat + RANSAC	85.94	28.91	9.566	2.459
	LoopGNN + RANSAC	94.39	79.10	10.42	2.461

The average precision (AP) and maximum recall (MR) are reported as average of the sequences. We compute the average translation error (ATE) as well as rotation error (RPE) under all retrieved pairs constituting a ground truth loop closure. VPR, LCD, and GV denote visual place recognition methods, deep loop closure detection methods, and geometric verification-based methods.

Next, we compare various deep loop closure detection methods in Tab. III. While our baseline PairNet-VLAD underperforms compared to both Calc and LoopCNN in terms of average precision, it yields a slightly higher maximum recall. We attribute this to the direct binary supervision the network benefits from, whereas the two other methods directly maximize feature similarity given an external learning signal. In contrast, LoopGNN operates on VLAD-retrieved neighborhoods of frames, which drastically improves both the average precision and recall as detailed in Tab. III, thus being an order of magnitude more robust towards false positive detection of perceptual aliases.

Lastly, we compare LoopGNN against a keypoint matcher utilizing XFeat’s keypoints followed by a geometric verification scheme. In addition, we also report the accuracy of the corresponding relative camera transformations in Tab. III using the RPE and ATE metrics, see Sec. IV-B. We observe that XFeat + RANSAC outperforms the standard LoopGNN in terms of AP. Nonetheless, this comes at the cost of a drastic decrease in maximum recall at 100% precision. This indicates that the XFeat + RANSAC pipeline is not robust to large amounts of perceptual aliases as depicted in Fig. 4. Nevertheless, it’s MR of 28.91% is substantially higher than the remaining deep loop closure detection and VPR baselines. In contrast, when combining LoopGNN with RANSAC, by geometrically verifying the top-0.5% of scores predicted by LoopGNN using RANSAC, we observe an AP increase of around 10% while reaching maximum recalls of nearly 80%. This ultimately demonstrates the robustness of LoopGNN in identifying perceptual aliases through deep graph consensus while introducing a minimal computational overhead. In addition to the overall quantitative results on TD2, we visualize the predictions of XFeat + RANSAC and LoopGNN in Fig. 5. Although LoopGNN yields robust estimates, we observe that the top-1% VLAD neighborhoods retrieve only a fraction of the true loop closures under a high degree of perceptual aliases, thus representing a significant bottleneck in loop closure detection.

Additionally, we train LoopGNN on the NCLT dataset

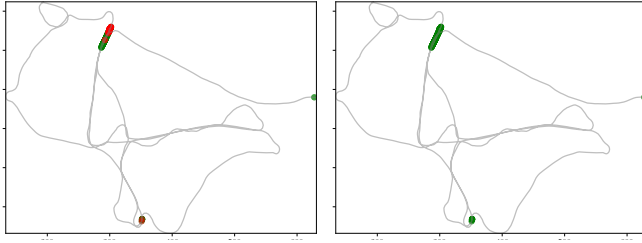


Fig. 5. Qualitative results comparing the loop closure predictions of XFeat [25] + RANSAC (left) against LoopGNN (right) under similar VLAD neighborhood retrieval of 1% at maximum recall at 100% precision. Red dots represent false negative loop closures, while green dots represent true positive loop closures. The underlying scene is figure_8_2023-09-13-17-24-26 of the TartanDrive 2.0 dataset.

TABLE IV

LOOP CLOSURE DETECTION PERFORMANCE ON THE NCLT DATASET

Model	AP [%]	MR [%]
Calc [15]	38.71	00.32
Pairwise-VLAD	36.90	04.29
LoopCNN [14]	50.05	00.92
LoopGNN (ours)	71.69	20.82

2% retrieval ratio The average precision (AP) and maximum recall (MR) are evaluated per-sequence. We compute the average translation error (ATE) as well as rotation error (RPE) using given ground-truth loop closures or all true positive predicted loop closures.

and compare it against Calc [15], LoopCNN [14], and our Pair-NetVLAD baseline. Different to the TD2 dataset, we opt for a larger neighborhood size of 2% given the shorter sequence length of the reprocessed NCLT dataset. As detailed in Tab. IV, LoopGNN outperforms both Calc and LoopCNN in terms of AP and MR, thus showing a similar trend as on the TD2 data. Compared to its Pair-NetVLAD counterpart, LoopGNN is more robust.

TABLE V

ABLATION STUDY OVER VARIOUS KEYPOINT EXTRACTORS

Feature Extractor	Model Variant	AP [%]	MR [%]
SIFT [20]	Pair-NetVLAD	00.00	00.00
	LoopGNN	00.00	00.00
SuperPoint [23]	Pair-NetVLAD	57.18	07.14
	LoopGNN	84.49	21.00
ALIKED [26]	Pair-NetVLAD	57.67	12.85
	LoopGNN	78.83	30.74
XFeat [25]	Pair-NetVLAD	43.38	06.39
	LoopGNN	84.63	45.52

Across all experiments, we retrieve the 1% of closest keyframes given a query keyframe using fitted VLAD clusters under the respective keypoint type mentioned. We demonstrate that our method shows significant performance gains in both AP and MR when employing any kind of deep keypoint feature extractor (SuperPoint, ALIKED, XFeat). On the contrary, we observe that within our framework traditional keypoint methods such as SIFT are not producing valuable estimates when used in VLAD clustering as well as loop closure detection.

TABLE VI
ABLATION STUDY OF NEIGHBORHOOD SIZES

Neighborhood Size	AP [%]	MR [%]
0.5%	80.03	17.34
1.0%	84.63	45.52
1.5%	<u>80.39</u>	16.44

We report the average precision and maximum recall at precision equal to one (MR) across various retrieved neighborhood sizes fed to the LoopGNN network.

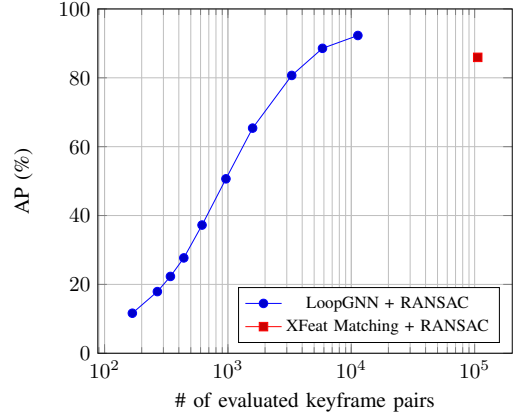


Fig. 6. Efficiency comparison of LoopGNN employing XFeat keypoint descriptors against classical XFeat descriptor matching on TartanDrive 2.0. The matches of both methods are geometrically verified under the epipolar model to increase robustness. Both methods draw keyframe pairs from the top-1% VLAD neighborhoods on TD2 scene figure_8_2023-09-13-17-24-26.

F. Ablation Study

In the following, we ablate on two key design parameters of our proposed framework. First, we demonstrate that LoopGNN’s capabilities are orthogonal to the underlying deep keypoint descriptor as detailed in Tab. V. Across all three deep keypoint methods (SuperPoint [23], ALIKED [42], XFeat [25]), we observe a similar relative performance improvement over the Pair-NetVLAD baseline. However, under traditional keypoint descriptors such as SIFT [20] our method does not produce viable outputs. We attribute this to significantly lower recalls during the VLAD retrieval stage and lower fidelity of SIFT features.

Furthermore, we demonstrate the effect of different neighborhood sizes on the performance of LoopGNN in Tab. VI. In general, we notice small decreases in terms of AP while the robustness to false positives measured as MR drops sharply. Thus, adequately-sized VLAD neighborhoods have a substantial influence on prediction robustness.

G. Computational Considerations

We demonstrate the computational benefits of running LoopGNN in comparison to utilizing a classical deep image matching + RANSAC pipeline as evaluated in Tab. III. In Fig. 6, we present the computational efficiency of LoopGNN over a classical matching pipeline. We observe a consistently rising AP when gradually increasing the number of considered top-scoring candidate edges from 10^2 to 10^4 for a scene of 4400 keyframes. Thus, LoopGNN allows for a considerable

reduction in the number of keyframe pairs requiring geometric verification. While it is generally not feasible to verify large numbers of candidates in an online scenario, LoopGNN provides precise scoring to ultimately decrease the number of candidates that need to undergo verification. In addition, GPU-based inference while retrieving top-1% neighborhoods allows for inference speeds of up to 20Hz on a NVIDIA A6000 GPU. Given these findings, we believe that integrating LoopGNN into an online visual SLAM method allows for a considerable reduction in map drift due to higher precision and recall of detected loop closures.

V. CONCLUSION

In this work, we introduced LoopGNN, a novel deep graph consensus approach to estimate robust loop closures. We found that conducting deep information exchange among retrieved initial candidates allows for a significant reduction in the actual number of keyframes that need to undergo geometric verification as demonstrated on two distinct datasets. In future work, we intend to evaluate the feasibility of larger neighborhoods and aim to integrate a differentiable RANSAC module into the LoopGNN architecture to perform batched geometric verification within a single forward pass. To foster future research in this domain, we make code and reprocessed keyframe data publicly available.

REFERENCES

- [1] A. Werby, C. Huang, M. Büchner, A. Valada, and W. Burgard, "Hierarchical Open-Vocabulary 3D Scene Graphs for Language-Grounded Robot Navigation," in *Robotics: Science and Systems*, 2024.
- [2] D. Honerkamp, T. Welschhold, and A. Valada, "N² m²: Learning navigation for arbitrary mobile manipulation motions in unseen and dynamic environments," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3601–3619, 2023.
- [3] D. Honerkamp, M. Büchner, F. Despinoy, T. Welschhold, and A. Valada, "Language-grounded dynamic scene graphs for interactive object search with mobile manipulation," *IEEE Robotics and Automation Letters*, 2024.
- [4] W. Wang, Y. Hu, and S. Scherer, "Tartanvo: A generalizable learning-based vo," in *Proc. of the Conf. on Rob. Learning*, 2021, pp. 1761–1772.
- [5] N. Vödisch, D. Cattaneo, W. Burgard, and A. Valada, "Covio: Online continual learning for visual-inertial odometry," in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2023, pp. 2464–2473.
- [6] R. Arandjelovic and A. Zisserman, "All about vlad," in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2013, pp. 1578–1585.
- [7] F. Schmidt, C. Blessing, M. Enzweiler, and A. Valada, "Visual-inertial slam for agricultural robotics: Benchmarking the benefits and computational costs of loop closing," *arXiv preprint arXiv:2408.01716*, 2024.
- [8] D. Gálvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.
- [9] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2016, pp. 5297–5307.
- [10] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, p. 381–395, 1981.
- [11] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [12] J. Yu, H. Ye, J. Jiao, P. Tan, and H. Zhang, "Gv-bench: Benchmarking local feature matching for geometric verification of long-term loop closure detection," in *Int. Conf. on Intelligent Robots and Systems*, 2024, pp. 7922–7928.
- [13] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "The revisiting problem in simultaneous localization and mapping: A survey on visual loop closure detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 19929–19953, 2022.
- [14] X. Zhang, Y. Su, and X. Zhu, "Loop closure detection for visual slam systems using convolutional neural network," in *International Conference on Automation and Computing (ICAC)*, 2017, pp. 1–6.
- [15] N. Merrill and G. Huang, "Lightweight unsupervised deep loop closure," *Robotics: Science and Systems*, 2018.
- [16] M. Büchner and A. Valada, "3d multi-object tracking using graph neural networks with cross-edge modality attention," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9707–9714, 2022.
- [17] M. Sivaprakasam, P. Maheshwari, M. G. Castro, S. Triest, M. Nye, S. Willits, A. Saba, W. Wang, and S. Scherer, "Tartandrive 2.0: More modalities and better infrastructure to further self-supervised learning research in off-road driving tasks," in *Int. Conf. on Robotics & Automation*, 2024, pp. 12606–12606.
- [18] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of michigan north campus long-term vision and lidar dataset," *The International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2016.
- [19] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Int. Conf. on Computer Vision*, 2011, pp. 2564–2571.
- [20] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.
- [21] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. of the Europ. Conf. on Computer Vision*, 2006, pp. 404–417.
- [22] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "Lift: Learned invariant feature transform," in *Proc. of the Europ. Conf. on Computer Vision*, 2016, pp. 467–483.
- [23] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *IEEE/CVF Conf. on Comp. Vision and Pattern Recognition Workshops*, 2018, pp. 224–236.
- [24] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2020, pp. 4938–4947.
- [25] G. Potje, F. Cadar, A. Araujo, R. Martins, and E. R. Nascimento, "Xfeat: Accelerated features for lightweight image matching," in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2024, pp. 2682–2691.
- [26] J. Edstedt, Q. Sun, G. Bökman, M. Wadenbäck, and M. Felsberg, "RoMa: Robust Dense Feature Matching," *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2024.
- [27] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby et al., "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.
- [28] G. Berton, C. Masone, and B. Caputo, "Rethinking visual geo-localization for large-scale applications," in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, 2022, pp. 4878–4888.
- [29] 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*, 2023.
- [30] N. Merrill and G. Huang, "Calc2.0: Combining appearance, semantic and geometric information for robust and efficient visual loop closure," in *Int. Conf. on Intelligent Robots and Systems*, 2019, pp. 4554–4561.
- [31] K. Qiu, Y. Ai, B. Tian, B. Wang, and D. Cao, "Siamese-resnet: implementing loop closure detection based on siamese network," in *IEEE Intelligent Vehicles Symposium*, 2018, pp. 716–721.
- [32] J. Yu and S. Shen, "Semanticloop: Loop closure with 3d semantic graph matching," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 568–575, 2022.
- [33] S. An, G. Che, F. Zhou, X. Liu, X. Ma, and Y. Chen, "Fast and incremental loop closure detection using proximity graphs," in *Int. Conf. on Intelligent Robots and Systems*, 2019, pp. 378–385.
- [34] Z. Qian, J. Fu, and J. Xiao, "Towards accurate loop closure detection in semantic slam with 3d semantic covisibility graphs," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2455–2462, 2022.
- [35] R. Duan, Y. Feng, and C.-Y. Wen, "Deep pose graph-matching-based loop closure detection for semantic visual slam," *Sustainability*, vol. 14, no. 19, p. 11864, 2022.

- [36] J. J. Kim, M. Urschler, P. J. Riddle, and J. S. Wicker, "Closing the loop: Graph networks to unify semantic objects and visual features for multi-object scenes," in *Int. Conf. on Intelligent Robots and Systems*, 2022, pp. 4352–4358.
- [37] N. Sünderhauf and P. Protzel, "Switchable constraints for robust pose graph slam," in *Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 1879–1884.
- [38] P.-Y. Lajoie, S. Hu, G. Beltrame, and L. Carlone, "Modeling perceptual aliasing in slam via discrete–continuous graphical models," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1232–1239, 2019.
- [39] T. Qin, S. Cao, J. Pan, and S. Shen, "A general optimization-based framework for global pose estimation with multiple sensors," *arXiv preprint arXiv:1901.03642*, 2019.
- [40] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks," *Int. Conf. on Learn. Repr.*, 2018.
- [41] B. Yang, Z. Li, W. Li, Z. Cai, C. Wen, Y. Zang, M. Muller, and C. Wang, "Lisa: Lidar localization with semantic awareness," in *Proc. of the IEEE Conf. on Comp. Vision and Pattern Recognition*, June 2024, pp. 15 271–15 280.
- [42] X. Zhao, X. Wu, W. Chen, P. C. Y. Chen, Q. Xu, and Z. Li, "Aliked: A lighter keypoint and descriptor extraction network via deformable transformation," *IEEE Transactions on Instrumentation & Measurement*, vol. 72, pp. 1–16, 2023.