

Highly Compressive Visual Self-localization Using Sequential Semantic Scene Graph and Graph Convolutional Neural Network

Yoshida Mitsuki Yamamoto Ryogo Tanaka Kanji

Abstract—In this paper, we address the problem of image sequence-based self-localization from a new highly compressive scene representation called sequential semantic scene graph (S3G). Recent developments in deep graph convolutional neural networks (GCNs) have enabled a highly compressive visual place classifier (VPC) that can use a scene graph as the input modality. However, in such a highly compressive application, the amount of information lost in the image-to-graph mapping is significant and can damage the classification performance. To address this issue, we propose a pair of similarity-preserving mappings, image-to-nodes and image-to-edges, such that the nodes and edges act as absolute and relative features, respectively, that complement each other. Moreover, the proposed GCN-VPC is applied to a new task of viewpoint planning of the query image sequence, which contributes to further improvement in the VPC performance. Experiments using the public NCLT dataset validated the effectiveness of the proposed method.

I. INTRODUCTION

Image sequence -based self-localization has received considerable attention as a highly compressive and discriminative approach to long-term visual place recognition (VPR) across domains (e.g., weather, time of day, and season). Given a long-term image sequence $I_1^{map}, \dots, I_T^{map}$ covering the robot workspace in a past domain (“map”), the sequence -based self-localization aims to determine the most matched sub-sequence for a short-term query sequence $I_{t_1}^{query}, \dots, I_{t_N}^{query}$ ($|t_N - t_1| \ll T$) in a new unseen domain [1]. An advantage of this sequence -based self-localization approach is a good balance between the discriminative power and compactness. Even in highly compressive applications (e.g., 4-bit image descriptor [2]) where typical single-view self-localization approaches suffer, the sequence -based self-localization approach may maintain high performance owing to the multi-view information fusion. Moreover, such a map can be flexibly reorganized into a local map I_a, \dots, I_b with arbitrary start a and goal b viewpoints ($|b - a| < T$), which enables an efficient reuse of storage in applications such as multi-robot multi-session knowledge sharing [3].

In this study, we explore the sequence -based self-localization problem from a novel perspective of sequential semantic scene graph (S3G), as shown in Fig. 1. The motivation for this approach is threefold. First, semantic scene graph (S2G) $x^G = \langle x^N, x^E \rangle$, a graph whose nodes x^N represent semantic regions (e.g., object regions) and edges

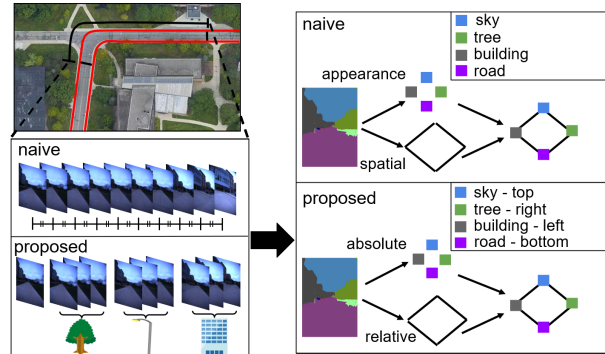


Fig. 1. Visual place recognition (VPR) from a novel highly compressive sequential semantic scene graph (S3G) is considered. To address the information lost in dimension reduction, the appearance/spatial image information is mapped to two different features, absolute (node) and relative (edge) features, which complement each other. Additionally, a new task of viewpoint planning of the query S3G is enabled by the trained VPR, to further improve the VPR performance.

x^E represent relationship between nodes, is one of most compact scene descriptors in computer vision [4]. Second, recently-developed deep graph convolutional neural networks (GCNs) can serve as a powerful visual-place-classifier (VPC) $y = f_{VPC}(x)$ that takes a scene graph x as query and predicts the place class y [5]. Third, such a well-trained VPC is noted to provide useful information for viewpoint planning of the query sequence $I_{t_1}^{query}, \dots, I_{t_N}^{query}$, to further improve the VPR performance. The remaining issue is how to design an effective similarity-preserving mapping $x^G = f_{2G}(I)$ from an input image I to a scene graph x^G .

Here, we propose to exploit a pair of similarity-preserving-mappings, image-to-nodes $x^N = f_{2N}(I)$ and image-to-edges $x^E = f_{2E}(I)$, which provide two forms (i.e., nodes x^N and edges x^E) of appearance/spatial image features. In earlier approaches, the appearance and spatial image features are separately mapped to nodes and edges, respectively (i.e., appearance-to-nodes, and spatial-to-edges) [6]. However, in our highly compressive application, the amount of information lost in the dimension-reduction is significant that recognition errors can be frequently yielded in a mapping. In the proposed approach, the nodes x^N and edges x^E act as *absolute* (e.g., size and brightness) and *relative* (e.g., larger and brighter) features [7], respectively, that complement each other, to further improve the overall VPR performance.

Our contributions were as follows. (1) We presented a highly compressive S3G -based VPR approach. S3G is an extension of the sequence -based self-localization framework toward the application of an S2G. Moreover, we addressed

Our work has been supported in part by JSPS KAKENHI Grant-in-Aid for Scientific Research (C) 17K00361 and 20K12008.

The authors are with Graduate School of Engineering, University of Fukui, Japan. mf210369@g.u-fukui.ac.jp, mf220362@g.u-fukui.ac.jp, tnknkj@u-fukui.ac.jp

the viewpoint-planning to control the viewpoints of the S3G to further improve the overall VPR performance. (2) We proposed a pair of similarity-preserving mappings, image-to-nodes and image-to-edges, such that the absolute features (i.e., nodes) and relative features (i.e., edges) complement each other. (3) We implemented a prototype system of the proposed framework for simultaneous VPR and viewpoint planning, and validated the effectiveness of the proposed method through experimental comparisons and ablation studies using the public NCLT dataset [8].

II. RELATED WORK

Highly compressive scene descriptors have been extensively studied in long-term large-scale VPR applications. In single-view VPR applications, vector quantization (e.g., bag-of-words) [9], and compact binary codes based on either local [9] or global features [10] have been studied. More recently, the local features of the image are packed into a compact binary code to further reduce bits per image [11]. Compared to these single-view approaches, the sequence-based self-localization approach requires fewer bits per image owing to the multi-view information fusion. Our approach can be viewed as a new approach that combines the advantages of compressive sequence-based self-localization approach and a compact graph-based scene descriptor.

The field of sequence-based self-localization was pioneered by the seminal work in [12] on the first cross-domain self-localization system named SeqSLAM. Several related works have studied on the issues of improving the descriptive power (e.g., [13]) and inference algorithms (e.g., [14]). Most existing works focused on improving the VPR accuracy, while suppressing the execution time. In contrast, we focus mainly on two aspects: graph neural networks and scene graph compression. Moreover, we also consider a new task of viewpoint-planning, which is particularly effective for autonomous navigation scenarios, where the robot must adapt its viewpoints to minimize the energy consumption in moving between viewpoints.

Graph-based scene representation has attracted recent research interest in self-localization [15]–[17]. For example, in [15], the method of decoupling a scene into subimages via semantic segmentation and connecting the segmented subimages via object-level edges experimentally demonstrated good VPR performance.

In earlier approaches, nodes are typically described by high-dimensional visual features. However, such an approach is expensive in our highly compressive application [18]. In our approach, a lightweight feature category ID is employed instead of the costly high-dimensional feature.

In several approaches, graph-based VPR is formulated as a graph matching task [19]. However, the cost of graph matching algorithms increases rapidly with the environment size. In our approach, graph-based VPR is formulated as a graph classification task that inherits the desirable properties of the classification task formulation, such as flexibility in defining place classes [20], compressed classifier model [21], and high classification speed [22].

Our active vision approach based on viewpoint-planning is inspired by the recent progress in active self-localization. In [23], an active self-localization task was addressed by extending the Markov localization framework for action planning. In [24], an appearance-based active observer for a micro-aerial vehicle was presented. In [25], a deep neural network-based extension of active self-localization was addressed using a learned policy model. In [26], the policy model and the perceptual and likelihood models were completely learned. In [27], a neural network-based active SLAM framework was investigated. However, these existing studies suppose in-domain scenarios, where the appearance changes of the scene between the training and testing domains was not large. In our approach, we address the issue from a novel perspective of domain-invariant VPR. To the best of our knowledge, no prior work exists on the use of GCN in the aforementioned context.

III. APPROACH

The system includes the (offline) training module and (on-line) test module. In addition, a scene graph descriptor submodule is employed by both the training and test modules. These modules are detailed in the following subsections.

A. Edge: Relative Feature

We considered several possible approaches to constructing graph edges from an input query/map image sequence. The first approach is to view each image frame in the sequence as a graph node and connect neighboring image nodes by edges [5]. The second approach is to apply pre-defined partitioning, such as grid-based partitioning, to obtain spatio-temporal node regions. The third approach is to employ semantic video segmentation techniques to segment an input image sequence (video clip) into spatio-temporal region nodes. The fourth approach employs image segmentation techniques to segment images into image region nodes.

In this study, we chose the image scene graph with image region nodes (i.e., the fourth approach) for the following reasons. The first approach is simple and effective. Indeed, in our earlier system on GCN-based VPR [5], each image frame is partitioned into predefined regions (e.g., “center,” “left,” “right”). However, such a graph structure does not reflect the scene layout, i.e., provides no information. In experiments, we used this approach as the baseline method. The second and third approaches are prohibitive in our application; hence, in these approaches, the start/goal endpoints of individual sequences must be defined during the training stage, and cannot be controlled on the fly during the testing stage. In contrast, the fourth approach is flexible and informative. Unlike the second and third approaches, it allows a map-user robot to flexibly control the endpoints on the fly during the testing stage. Moreover, unlike in the first approach, the graph edges reflect the scene layout, thus yielding informative scene graphs.

The procedure for graph construction is as follows. First, semantic labels are assigned to pixels using DeepLab v3+ [28], pretrained on Cityscapes dataset. Then, regions smaller

than 100 pixels are regarded as not characterizing the input scene, and removed. Subsequently, connected regions with the same semantic labels are identified using a flood-fill algorithm [29], and each is assigned a unique region ID. Next, each region is connected to each of its neighboring regions by an edge. Finally, an image scene graph with image region nodes is obtained.

B. Node: Absolute Feature

Image region descriptors (i.e., node descriptor in our case) have been intensively studied. Existing descriptors can be generally categorized into two groups: local features [10] and global features [30]. Most are in the form of high-dimensional feature vectors [30], or an unordered collection of vector quantized features called “bag-of-words” [9]. However, their space costs increase in proportion to the number and dimensionality of the original feature vectors, and typically require a hundreds or thousands of bytes per scene. This is expensive in our highly compressive application.

In the field of image retrieval, semantic labels (e.g., object category IDs) have been used as extremely compact descriptors that can be used for indexing and retrieving images [31].

A key difference of our robotics application from these image retrieval applications is that in robotics, the semantics of surrounding objects and their spatial information such as bearing (“B”) and range (“R”) play important roles in VPR. Our approach is closely related to the special interest groups of SLAM researches focusing on different types of spatial information, namely, range-bearing SLAM [32], range-only SLAM [33], and bearing-only SLAM [34].

Based on the aforementioned consideration, our VPR task was formulated as a bearing-range-semantic (BRS) measurements-based VPR.

Specifically, in our approach, the semantic labels output by a semantic segmentation network [35] were re-categorized into seven different semantic category IDs: “sky,” “tree,” “building,” “pole,” “road,” “traffic sign,” and “the others” which respectively correspond to the labels {“sky”}, {“vegetation”}, {“building”}, {“pole”}, {“road,” “sidewalk”}, {“traffic-light,” “traffic-sign”}, and {“person,” “rider,” “car,” “truck,” “bus,” “train,” “motorcycle,” “bicycle,” “wall,” “fence,” “terrain”} in the original label space. The location of the region center was quantized by a 3×3 regular grid into nine “bearing” category IDs. The region size was quantized into three “range” category IDs: “short distance (larger than 150 K pixels),” “medium distance (50 K-150 K pixels),” and “long distance (smaller than 50 K)” for 616×808 image. Finally, these semantic, bearing and range category IDs are combined to obtain a 189-dim 1-hot vector as the node descriptor.

At first glance, the use of edges as additional features might look unnecessary, because both the appearance and spatial features are already included in the node features. However, its objectives are quite different. That is, the edge descriptor is suitable for describing the relative feature (e.g., position-relationship) rather than the absolute feature (e.g.,

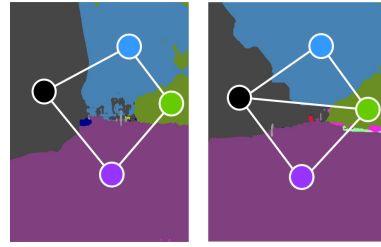


Fig. 2. Importance of edges. In highly compressive applications, a naive strategy of using only absolute features (i.e., nodes) suffer from information loss during dimension reduction. To address this issue, we exploit the edges as relative features that complement the absolute features.

position). The nodes and edges can act as error detection codes [36] that complement each other. Figure 2 illustrates an example showing how the edge features help to discriminate a nearly duplicate S2G pair, which the node descriptor alone could not discriminate.

C. Region Merging

Most existing techniques for semantic segmentation trade domain-invariance for accuracy. Their objective function and performance index place the highest priority on pixel-level precision/recall accuracy, i.e., detecting semantically coherent regions as accurately as possible. Hence, their graph topology is often affected by the other factors, such as viewpoint drift and occlusions. This is problematic for our VPR application as the graph topology is a primary cue that dominates the GCN performance.

To address this issue, we adopted a region merging technique, inspired by a recent work in [37]. Specifically, we remove small regions whose areas are smaller than 1,000 pixels (for 616×808 image). We noted this simple technique to be very effective in improving VPR performance, as illustrated through ablation studies in the experimental section (IV).

D. Self-localization from S2G

Self-localization from S2G is directly addressed by GCN-based VPC. It takes a single-view image and predicts the place class. The GCN is the most popular approach to graph neural networks. It has been successfully used in various applications, including web-scale recommender systems [38], and chemical reactivity [39]. In our recent study [5], a GCN is trained as an image-sequence classifier and achieved the state-of-the-art VPR performance.

For the definition of place classes, we follow the grid-based partitioning in [40]. In the experimental environment, this yields 10×10 grid cells and 100 place classes in total.

In this study, we trained a GCN using the S2Gs as the training data. The graph convolution operation takes node v_i in the graph and processes it in the following manner. First, it receives messages from nodes connected by the edge. The collected messages are then summed via the SUM function. The result is passed through a single-layer fully connected neural network followed by a nonlinear transformation for

conversion into a new feature vector. In this study, we used the rectified linear unit (ReLU) operation as the nonlinear transformation. The process was applied to all the nodes in the graph in each iteration, yielding a new graph that had the same shape as the original graph but updated node features. The iterative process was repeated L times, where L represents the ID of the last GCN layer. After the graph node information obtained in this manner were averaged, the probability value vector of the prediction for the graph was obtained by applying the fully connected layer and the softmax function. For implementation, we used the deep graph library [41] on the Pytorch backend.

E. Self-localization from S3G

The sequence -based self-localization system aims to estimate the robot location from a sequence of S2Gs (i.e., S3G). It consists of two modules: information fusion and viewpoint planning.

The information fusion module employs a particle filter to incorporate the history of perceptual-action measurements $I_1^{query}, \dots, I_n^{query}$ at each viewpoint t_n of the query image sequence into the current belief of the robot location, as in [42]. An action corresponds to a forward movement along the viewpoint trajectory, and a perception corresponds to a class-specific probability density vector (PDV) output by the GCN. Our framework differs from standard particle filter-based self-localization in that a perceptual measurement and the belief are provided in the form of class-specific reciprocal rank vector, where the i -th element is the class-specific reciprocal rank value of the i -th class. Specifically, the elements of the PDV are sorted in the descending order of the probability values, and then the reciprocal of the rank value of the i -th class in the sorted list is computed, as in [5]. In the measurement update, each particle is categorized into a place class, and its weight is updated by adding the reciprocal rank value of the class to which the particle’s hypothesized location belongs.

The viewpoint planning is formulated as a reinforcement-learning (RL) problem, in which a learning agent interacts with a stochastic environment. The interaction is modeled as a discrete-time discounted Markov decision process (MDP). A discounted MDP is a quintuple (S, A, P, R, γ) , where S and A are the set of states and actions, respectively. P denotes the state transition distribution, R denotes the reward function, and $\gamma \in (0, 1)$ denotes a discount factor ($\gamma = 0.9$). The learning rate was set to $\alpha = 0.1$. We denoted $P(\cdot|s, a)$ and $r(s, a)$ as the probability distribution over the next state and the immediate reward of performing action a for state s , respectively. State s is defined as the class-specific reciprocal rank vector, as in [43]. Action a is defined as a forward movement along the route. In the experiments, the action candidate set was $A = \{1, 2, \dots, 10\}$ (m). A training/test episode was a 4 sequential action. We addressed the issue of the curse of dimensionality in RL by employing the recently developed nearest neighbor approximation of Q-learning (NNQL) [44] with $k = 4$. The immediate reward

was provided at the final viewpoint of each training episode, as the reciprocal rank value of the ground truth viewpoint.

IV. EXPERIMENTS

The proposed method was evaluated in a cross-season self-localization scenario. The goal of the evaluation was to validate whether the GCN-based VPR method could boost the performance in both passive VPR and active VPR scenario.

The public NCLT dataset [8] was used in the experiments (Fig. 3). The NCLT dataset is a dataset for long-term autonomy obtained using a Segway vehicle at the University of Michigan North Campus. While the vehicle travels seamlessly indoors and outdoors, the vehicle encountered various geometric changes (e.g., object placement changes, pedestrians, car parking/stopping) and photometric changes (e.g., lighting conditions, shadows, and occlusions).

In particular, we assumed challenging cross-season self-localization scenarios, in which the self-localization system is trained and tested in different seasons. To this end, four pairings of (training, test) seasons were created from four different seasons’ datasets “2012/1/22 (WI),” “2012/3/31 (SP),” “2012/8/4 (SU),” and “2012/11/17 (AU)”: (WI, SP), (SP, SU), (SU, AU), and (AU, WI). Additionally, an extra season “2012/5/11 (EX)” was used to train VPR. That is, VPR was trained only once in the season EX prior to the self-localization tasks, and then the learned VPR parameters were commonly used for all the training-testing season pairs.

Figure 4 demonstrates scene graphs obtained using the proposed scene graph construction procedure. Notably, the domain-invariant parts (e.g., buildings and roads) of the input scenes tended to be selected as the image region nodes.

Three different VPR methods, GCN, naive Bayes nearest neighbor (NBNN), and k-nearest neighbor (kNN) were evaluated. All took an image sequence represented as an S3G, as input. The GCN is the proposed method described in Section III. Because no prior study exists on the highly compressive S3G applications, our investigation mainly focused on developing and evaluating as various possible ablations of this method as possible. NBNN and kNN methods are based on measuring dissimilarities in the node feature set between a query-map pair of interest. NBNN [45] is a best known method to measure dissimilarities between such a feature set pair, and has shown high performance in our earlier studies (e.g., [46]). In that, the L2 distance from the nearest-neighbor map feature to each query feature is computed, and then it is averaged over all the query features, which yields the NBNN dissimilarity value. kNN is a traditional non-parametric classification method based on the nearest-neighbor training sample in the feature space, in which the class labels most often assigned to the training samples of the kNN (i.e., minimum L2 norm) are returned as classification results. In that, an image is described by a 189-dim histogram vector by aggregating all the node features that belong to the image.

VPR performance was evaluated in terms of top-1 accuracy. The evaluation procedure was as follows. First, VPR

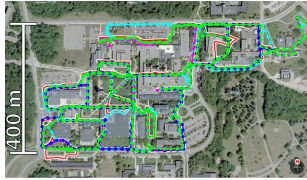


Fig. 3. Experimental environments. The trajectories of the four datasets, “2012/1/22,” “2012/3/31,” “2012/8/4,” and “2012/11/17,” used in our experiments are visualized in green, purple, blue, and light-blue curves, respectively.

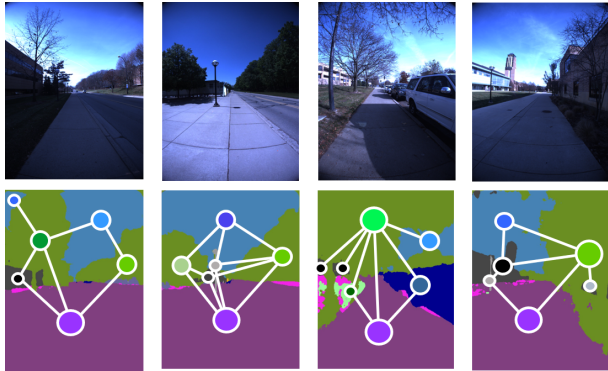


Fig. 4. S2G examples. Top: The input image. Bottom: S2G overlaid on the semantic label image.

performance at all viewpoints of the query sequence, and not just the final viewpoint, were computed. Then, top-1 accuracy at each viewpoint was computed from the latest particle filter output based on whether the class with highest belief value matches the ground-truth.

Figure 5 shows examples of views before and after planned NBV actions. Intuitively convincing behavior of the robot was observed. Before the move, the scene was a non-salient one consisting only of the sky, the road, and the trees (Fig. 5 a,b,c), or the field of view was very narrow due to occlusions (Fig. 5d). After the move, landmark objects came into view (Fig. 5 a,c) or additional landmarks appeared (Fig. 5 b,d). Such behaviors are intuitively appropriate and effective for seeking and tracking landmarks when a human becomes lost (humans look for familiar landmarks). Our approach enables the robot to learn such appropriate step sizes from available visual experience.

An ablation study was conducted to observe the effects of individual components, including the relative edge feature (Section III-A) and region merging technique (Section III-C). Table I lists the results of all combinations of the proposed and baseline methods with and without viewpoint-planning (VP). Notably, the proposed method yielded a superior performance compared to the other methods. The technique of region merging (Section III-C) contributed to reduce the number of nodes while retaining the VPR performance. The number of nodes was reduced from 19.8 to 7.2 per S2G on average, which results in the reduction of computation time for VPR from 0.82 ms to 0.16 ms. Particularly, the use of edge feature and the NBV module often significantly boosted



(a) (b) (c) (d)

Fig. 5. NBV planning results. In each figure, the bottom and top panels show the view image before and after planned movements, respectively.

TABLE I
PERFORMANCE RESULTS.

		w/ region merging		w/o region merging	
		S	BRS	S	BRS
VPR	GCN	11.7	18.9	12.0	19.1
	KNN	5.8	15.6	6.3	12.9
	NBNN	1.3	3.4	1.4	3.5
VPR+VP	GCN	-	30.6	-	-
	KNN	-	-	-	-
	NBNN	-	-	-	-

the VPR performance.

Finally, we investigated space costs. The number of nodes per S2G was 7.2 on average. The node descriptor consumed 8-bit per node. The space cost for nodes and edges were 57.8-bit and 12.5-bit per S2G, respectively, on average. This is significantly lower cost than typical methods based on high-dimensional vectorial features, and than compact variants such as bag-of-words. Notably, the current descriptors were not compressed, i.e., they may be further compressed.

V. CONCLUSIONS

In this study, a novel highly compressive VPR framework for sequence-based self-localization using S3Gs was proposed. For such a highly compressive application, the amount of information lost in the dimension reduction is significant and can damage the VPR performance. To address this issue, we proposed the exploitation of a pair of similarity-preserving mappings, image-to-nodes and image-to-edges, such that the nodes and edges complement each other. Moreover, the S3G framework was applied to the viewpoint-planning task to control the viewpoints of an S3G to further improve the performance. Experiments using the public NCLT dataset with performance comparisons and ablation studies validated the effectiveness of the proposed method.

REFERENCES

- [1] M. J. Milford and G. F. Wyeth, “Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights,” in *2012 IEEE Int. Conf. Robotics and Automation*, 2012, pp. 1643–1649.
- [2] F. Yan, O. Vysotska, and C. Stachniss, “Global localization on openstreetmap using 4-bit semantic descriptors,” in *2019 European Conf. Mobile Robots (ECMR)*, 2019, pp. 1–7.

- [3] M. T. Lazaro, L. M. Paz, P. Pinies, J. A. Castellanos, and G. Grisetti, "Multi-robot slam using condensed measurements," in *2013 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2013, pp. 1069–1076.
- [4] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph r-cnn for scene graph generation," in *Proc. European Conf. computer vision (ECCV)*, 2018, pp. 670–685.
- [5] K. Takeda and K. Tanaka, "Dark reciprocal-rank: Teacher-to-student knowledge transfer from self-localization model to graph-convolutional neural network," in *2021 Int. Conf. Robotics and Automation (ICRA)*, 2021, pp. 4348–4355.
- [6] Y. Zhu, Y. Ma, L. Chen, C. Liu, M. Ye, and L. Li, "Gosmatch: Graph-of-semantics matching for detecting loop closures in 3d lidar data," in *2020 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2020, pp. 5151–5157.
- [7] Y. Soury, E. Noury, and E. Adeli, "Deep relative attributes," in *Asian Conf. computer vision*, 2016, pp. 118–133.
- [8] 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.
- [9] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, IEEE Int. Conf.*, vol. 3, 2003, pp. 1470–1470.
- [10] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.
- [11] H. Jégou, M. Douze, and C. Schmid, "Packing bag-of-features," in *2009 IEEE 12th Int. Conf. Computer Vision*, 2009, pp. 2357–2364.
- [12] N. Stünderhauf, P. Neubert, and P. Protzel, "Are we there yet? challenging seqslam on a 3000 km journey across all four seasons," in *Proc. of Workshop on Long-Term Autonomy, IEEE Int. Conf. Robotics and Automation (ICRA)*, 2013.
- [13] B. Dongdong, W. Chaogun, B. Zhang, Y. Xiaodong, and Y. Xuejun, "Cnn feature boosted seqslam for real-time loop closure detection," *Chinese Journal of Electronics*, vol. 27, no. 3, pp. 488–499, 2018.
- [14] F. Zeng, A. Jacobson, D. Smith, N. Boswell, T. Peynot, and M. Milford, "I2-s2: Intra-image-seqslam for more accurate vision-based localisation in underground mines," in *Proc. Australasian Conf. Robotics and Automation (ACRA) 2018*, 2018, pp. 1–10.
- [15] 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.
- [16] C. Zhang, I. Budvytis, S. Liwicki, and R. Cipolla, "Lifted semantic graph embedding for omnidirectional place recognition," in *2021 International Conference on 3D Vision (3DV)*, 2021, pp. 1401–1410.
- [17] E. Stumm, C. Mei, S. Lacroix, J. Nieto, M. Hutter, and R. Siegwart, "Robust visual place recognition with graph kernels," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2016, pp. 4535–4544.
- [18] S. Schubert, P. Neubert, and P. Protzel, "Fast and memory efficient graph optimization via icm for visual place recognition," in *Proc. of Robotics: Science and Systems (RSS)*, 2021.
- [19] X. Kong, X. Yang, G. Zhai, X. Zhao, X. Zeng, M. Wang, Y. Liu, W. Li, and F. Wen, "Semantic graph based place recognition for 3d point clouds," in *2020 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2020, pp. 8216–8223.
- [20] P. Beeson, N. K. Jong, and B. Kuipers, "Towards autonomous topological place detection using the extended voronoi graph," in *Proc. 2005 IEEE Int. Conf. Robotics and Automation*, 2005, pp. 4373–4379.
- [21] Y. Li, S. Lin, B. Zhang, J. Liu, D. Doermann, Y. Wu, F. Huang, and R. Ji, "Exploiting kernel sparsity and entropy for interpretable cnn compression," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2019, pp. 2800–2809.
- [22] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.
- [23] W. Burgard, D. Fox, and S. Thrun, "Active mobile robot localization," in *IJCAI*, 1997, pp. 1346–1352.
- [24] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Adaptive mobile robot navigation and mapping," *The International Journal of Robotics Research*, vol. 18, no. 7, pp. 650–668, 1999.
- [25] D. S. Chaplot, E. Parisotto, and R. Salakhutdinov, "Active neural localization," *arXiv preprint arXiv:1801.08214*, 2018.
- [26] S. K. Gottipati, K. Seo, D. Bhatt, V. Mai, K. Murthy, and L. Paull, "Deep active localization," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4394–4401, 2019.
- [27] D. S. Chaplot, D. Gandhi, S. Gupta, A. Gupta, and R. Salakhutdinov, "Learning to explore using active neural slam," *arXiv preprint arXiv:2004.05155*, 2020.
- [28] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. European Conf. computer vision (ECCV)*, 2018, pp. 801–818.
- [29] Y. He, T. Hu, and D. Zeng, "Scan-flood fill (scaff): An efficient automatic precise region filling algorithm for complicated regions," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.
- [30] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [31] P. Hiremath and J. Pujari, "Content based image retrieval using color, texture and shape features," in *15th Int. Conf. Advanced Computing and Communications (ADCOM 2007)*, 2007, pp. 780–784.
- [32] M. Ramezani, G. Tinchev, E. Iuganov, and M. Fallon, "Online lidar-slam for legged robots with robust registration and deep-learned loop closure," in *2020 IEEE Int. Conf. Robotics and Automation (ICRA)*, 2020, pp. 4158–4164.
- [33] Y. Song, M. Guan, W. P. Tay, C. L. Law, and C. Wen, "Uwb/lidar fusion for cooperative range-only slam," in *2019 international Conf. robotics and automation (ICRA)*, 2019, pp. 6568–6574.
- [34] E. Bj, T. A. Johansen, et al., "Redesign and analysis of globally asymptotically stable bearing only slam," in *2017 20th Int. Conf. Information Fusion (Fusion)*, 2017, pp. 1–8.
- [35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation. arxiv 2015," *arXiv preprint arXiv:1505.04597*, 2019.
- [36] A. D. Córcoles, E. Magesan, S. J. Srinivasan, A. W. Cross, M. Steffen, J. M. Gambetta, and J. M. Chow, "Demonstration of a quantum error detection code using a square lattice of four superconducting qubits," *Nature communications*, vol. 6, no. 1, pp. 1–10, 2015.
- [37] B. Matejek, D. Haehn, H. Zhu, D. Wei, T. Parag, and H. Pfister, "Biologically-constrained graphs for global connectomics reconstruction," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2019, pp. 2089–2098.
- [38] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proc. 24th ACM SIGKDD Int. Conf. Knowledge Discovery & Data Mining*, 2018, pp. 974–983.
- [39] C. W. Coley, W. Jin, L. Rogers, T. F. Jamison, T. S. Jaakkola, W. H. Green, R. Barzilay, and K. F. Jensen, "A graph-convolutional neural network model for the prediction of chemical reactivity," *Chemical science*, vol. 10, no. 2, pp. 370–377, 2019.
- [40] G. Kim, B. Park, and A. Kim, "1-day learning, 1-year localization: Long-term lidar localization using scan context image," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1948–1955, 2019.
- [41] M. Wang, L. Yu, D. Zheng, Q. Gan, Y. Gai, Z. Ye, M. Li, J. Zhou, Q. Huang, C. Ma, Z. Huang, Q. Guo, H. Zhang, H. Lin, J. Zhao, J. Li, A. J. Smola, and Z. Zhang, "Deep graph library: Towards efficient and scalable deep learning on graphs," *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.
- [42] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *1999 Int. Conf. Robotics and Automation (ICRA)*, vol. 2, 1999, pp. 1322–1328.
- [43] K. Tanaka, "Active map-matching: Teacher-to-student knowledge transfer from visual-place-recognition model to next-best-view planner for active cross-domain self-localization," in *2021 IEEE Int. Conf. Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, 2021, pp. 1–6.
- [44] D. Shah and Q. Xie, "Q-learning with nearest neighbors," *arXiv preprint arXiv:1802.03900*, 2018.
- [45] T. Tommasi and B. Caputo, "Frustratingly easy nbnn domain adaptation," in *Proc. IEEE Int. Conf. Computer Vision*, 2013, pp. 897–904.
- [46] K. Tanaka, "Cross-season place recognition using nbnn scene descriptor," in *2015 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2015, pp. 729–735.