A visual place recognition approach using learnable feature map filtering and graph attention networks

Qin, C., Zhang, Y., Liu, Y., Coleman, S., Du, H., & Kerr, D. (2021). A visual place recognition approach using learnable feature map filtering and graph attention networks. *Neurocomputing*, 457, 277-292.
<https://doi.org/10.1016/j.neucom.2021.06.038>

[Link to publication record in Ulster University Research Portal](#)

Published in:
Neurocomputing

Publication Status:
Published (in print/issue): 07/10/2021

DOI:
[10.1016/j.neucom.2021.06.038](https://doi.org/10.1016/j.neucom.2021.06.038)

Document Version
Author Accepted version

General rights
Copyright for the publications made accessible via Ulster University's Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Ulster University's institutional repository that provides access to Ulster's research outputs. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact pure-support@ulster.ac.uk.

A Visual Place Recognition Approach using Learnable Feature Map Filtering and Graph Attention Networks

Cao Qin · Yunzhou Zhang* · Yingda Liu · Sonya Coleman · Huijie Du · Dermot Kerr

Received: date / Accepted: date

Abstract Visual place recognition (VPR) in environments subject to extreme appearance variation due to changing weather, illumination or seasons is a challenging task. Recent works have shown that features learned from CNNs can achieve promising performance. However, most of the existing methods concentrate so much on the image itself that they neglect the architecture of the network, especially different filters that may carry more meaningful information. In this paper, we develop a learnable feature map filtering (FMF) module constrained by triplet loss to re-calibrate the weight of the individual feature map. In this way, specific feature maps that encode invariant characteristics of location are extracted. Moreover, to make full use of the rich global mutual information that resides in the sample set, we propose an influence-based graph attention network (IB-GAT) with a verification subnet to better incorporate the relations among samples during the training process. Different from conventional GAT approaches, IB-GAT enables feature nodes to attend over the influence of other nodes instead of the original feature. Thus refined features with more discriminative power could be generated. Extensive experiments have been conducted on six public VPR datasets with varying appearances. Ablation analysis verifies the potential

efficacy of the FMF module and the IB-GAT components. The experimental results also demonstrate that the proposed methods can achieve better performance than the current state of the art.

Keywords Visual place recognition, Convolutional Neural Networks (CNNs), Feature Map Filtering, Graph Attention Networks

1 Introduction

Visual place recognition (VPR) [23] aims to localize the place of the query image by associating it to one or several similar images stored in a pre-built environmental map. It has attracted a lot of attention for its wide applications in autonomous driving [24], simultaneous localization and mapping (SLAM) [9] and augmented reality (AR) [26]. Though great progress has been made in VPR, it remains extremely difficult under changing environmental conditions.

Traditional VPR methods that design handcrafted features to learn representation for place images, often are unable to derive valid features when encountering significant changes [27] in the appearance of places such as illumination, weather conditions, as well as some dynamic objects. Successfully deep learned CNNs have motivated many studies to extract features from CNNs for visual place recognition and have shown promising results [39, 38, 29, 30, 5]. There are two general methods to achieve VPR with the help of CNNs. The first one tackles the problem using pre-trained models, where the whole image [38] or regions-of-interest (ROIs) [39] are fed into the model and a certain intermediate feature map is extracted. The second one proposes to train or fine-tune networks for place recognition rather than using pre-trained models [30, 5]. However, most of the ex-

* Corresponding author.

C. Qin, Y. Zhang, Y. Liu

College of Information Science and Engineering, Northeastern University, Shenyang, China

E-mail: zhangyunzhou@mail.neu.edu.cn(✉)

H. Du

Science and Technology on Space System Simulation Laboratory, Beijing, China

S. Coleman, D. Kerr

Intelligent Systems Research Centre, University of Ulster, Derry, UK

isting approaches have put so much emphasis on the image itself that they have ignored the architecture design of the network, where different filters may carry more meaningful information. Although several works like [18, 16] revealed that some activations in intermediate layers are associated with environmental changes and tried to only extract the most useful feature maps for place recognition, prior off-line statistical processing was required. In this paper, we focus on filtering feature maps in an end-to-end manner with the filters learned by back-propagation. We aim to find which feature map corresponds to information that can be regarded as interference for VPR applications against varying environmental conditions. Such feature maps may have learned to encode differentiation due to a winter or spring day, a cloudy or sunny day, etc. By suppressing the appearance-relevant feature maps and retaining appearance-irrelevant features, we can obtain a more robust feature representation.

Considering that place recognition is a classification problem, some works [30, 5] cope with the problem using classification networks, where the network is trained with a standard cross-entropy loss for landmarks classification. Unlike the classification networks which learn to classify places, metric learning methods were proposed [43, 22, 1, 6] to learn relatively small feature distances between images of the same location and large feature distances between those of different locations. Such approaches can also produce visual representations at an efficient size, which are essential for fast place recognition. Nevertheless, most metric learning approaches only consider the pairwise similarity while ignoring the internal relationship among the images of the entire set. As a result, some hard negative or hard positive pairs fail to get proper similarity scores since only inadequate information among images is used for similarity evaluation.

To overcome this limitation, we need to mine and make full use of the valuable internal relationship among different samples. The burgeoning graph attention networks (GATs) [42], which can exploit the mutual information and propagate messages among different nodes in an arbitrary graph structure, have drawn increasing attention and successfully been applied in many computer vision studies [35, 47, 10, 45]. Moreover, they integrate graph computation with neural network learning, which makes the training end-to-end and benefits from learning the feature representation. Thus in this work, we follow the graph attention networks to utilize the rich mutual information between features, and in the meantime learn discriminative features for visual place recognition.

Fig. 1 Simplified principle scheme of the proposed framework. Our method re-calibrates the weights of the original feature maps and generates the filtered feature maps. Then in the feature graph, each feature node will attend over information (orange lines) from others to produce the optimized features with more discriminative power.

Fig. 1 shows the simplified principle scheme of the proposed framework which consists of two main parts. The first part is based on a feature embedding network with triplet loss but critically has the ability to learn features that we desire. We mainly attempt to recalibrate the weight of feature maps using a feature map filtering module in an end-to-end manner. Also, a verification subnet is introduced to force two images of the same location to be mapped to nearby points in feature space. We consider that verification models only use weak labels (same/different) and do not take all of the images into consideration. Hence in the second part, we reform the features filtered by the FMF module as a complete graph and employ the mechanism of GAT to propagate mutual information among nodes. With this kind of message propagation mechanism, our network can fully exploit the valuable information to produce discriminative features of the image for VPR.

To summarize, the main contributions of this paper are as follows:

- We propose a novel approach of achieving feature map filtering on convolutional layers in an end-to-end manner, while extracting robust features for place recognition under extreme appearance changes.
- To determine which feature map is more important, we develop a learnable Feature Map Filtering (FMF) module with triplet loss to auto-select the weights of different feature maps.
- An influence based graph attention networks (IB-GAT) is introduced to verification subnet to incorporate the relationships among samples during the training process. As a result, more discriminative and accurate features of places will be generated.

- Experimental evaluation is carried out on publicly available datasets, which shows that the proposed method has achieved competitive results.

2 Related Work

2.1 Visual Place Recognition with CNN features

Triggered by the recent successful application of deep learning over many computer vision techniques, a variety of studies [39, 38, 29, 11, 7, 2] have been introduced that address the VPR problem through CNN-based descriptors. Sünderhauf et al. [38] comprehensively analyzed the ability of CNN features for visual place recognition (appearance and viewpoint invariance). The authors extracted the feature maps from several layers of AlexNet [19] and compared their performance over four challenging datasets. Instead of utilizing features from one single layer, Arroyo et al. [2] concatenated and treated features from several CNN layers as a whole description vector. To improve the robustness under varied conditions and viewpoints, Chen et al. [7] proposed to extract visual landmarks directly based on activations of the convolutional layers. Although the above studies have made great improvements, the employed models pre-trained on tasks that are different from VPR decrease the representational ability of the generated visual features.

Recent research has focused on training or fine-tuning networks for place recognition. Chen et al. [5] interprets place recognition as a classification problem and trains a classification network using a massive landmark dataset. Deep Local Feature (DELF) [30] proposed CNN-derived local feature with attention, which is learned only using place annotations. As such classification networks were trained to identify landmarks, they produce better features related to specific places than those models trained for other tasks. However, the classification networks require huge landmark datasets, where each place needs a great number of sample images captured at different times. Some works [1, 22, 43] adopted metric learning methods to directly learn visual representations in an end-to-end manner. NetVLAD [1] took advantage of triplet loss [34] to learn the optimal VLAD encoding to match places across both conditions and viewpoint changes. Lifted structured loss [43] was used to train an Omnidirectional CNN to handle severe camera pose variation. Such metric learning methods has the ability to map the appearance difference to the spatial metric distance between different images. Motivated by these works, we also train our network with triplets of images selected from datasets, but the difference is that we learn an FMF module simultaneously

to recalibrate the weight of the feature maps to obtain the invariant features.

2.2 Feature Map Filtering in Deep Networks

While most CNN-based methods focused on either the choice of layers to use or the choice of spatial regions across the feature map stack, feature maps within the stack should also be considered. Several works first performed network pruning to compress CNNs. Han et al. [15] omitted the non-essential connections whose weights were below a threshold from the network, then applied weight sharing and Huffman coding to reduce the size of AlexNet. Another alternate method [50] is to perform Linear Discriminant Analysis (LDA) to calculate discriminability values of feature maps. They successfully omit a lot of redundant feature maps without significant loss in the original accuracy. Many works have also proved that individual feature map carries specific information that is tied closely to the outputs of the classifier. Polyak et al. [32] utilized the variance in the activated channel to estimate the significance of feature maps. Guo et al. [14] presented a method that removes the parts of the network which are related to the unimportant classes in an online manner. Zhou et al. [48] evaluated the performance of individual feature maps on various pixel-wise semantic segmentation tasks and found that a single feature map sensitive to different objects, textures, materials, and scenes. Besides, similar phenomena were also shown in the literature [18, 16] we have mentioned before.

In contrast to the above works which intended to eliminate the specific convolutional weights of filters, a 'soft' filtering method has been used to enhance performance of CNN-based tasks. Li et al. [21] proposed a feedback mechanism to re-weight feature maps for the purpose of boosting the classification performance. Attention-oriented approaches such as SENet [17] and convolutional block attention module (CBAM) [46] both performed channel-wise recalibration in order to focus on important feature maps and suppressing unnecessary ones. More close to our work, Lee et al. [20] introduced a style-based recalibration module (SRM) that incorporates the interaction between channels. In their SRM block, they use a style pooling approach composed of the average and standard deviation to compute channel-wise attention. Inspired by this work, in our task, images taken at different times in the same place can be also considered to be different in style. Therefore, the designed FMF module applies similar pooling operation to capture the style information of images. Compared to [20] which focused on the problem of image generation with the help of style pooling,

we concentrate on extracting invariant features for visual place recognition.

2.3 Graph Neural Network

The topic of Graph Neural Networks (GNNs) has received growing attention in recent years. A GNN model was presented in [33] as an extension for recursive neural networks that can process the data represented in graph domains. Bruna et al. [3] and Duvenaud et al. [8] proposed two constructions of deep convolutional networks on graphs (GCN) based on the spectrum of graph Laplacian and non-spectral manners respectively. Yan et al. [47] proposed a spatial-temporal graph convolutional network to handle the problem of human action recognition. Veličković et al. [42] further proposed an attention-based architecture called Graph attention networks (GATs) to perform node classification on graph-structured data. This model utilizing attention was able to extend over neighborhood features and successfully achieved state-of-the-art performance across node classification benchmarks.

The majority of these methods fit within the framework of message propagation [12]. In the message propagation framework, a GNN is regarded as message-passing operation where nodes are iteratively updated through aggregating the features of their neighbor nodes. Based on this consideration, GNN is employed in this paper to mine mutual information from images (nodes) and generate refined features with more discriminative power. Another important work was Similarity-Guided Graph Neural Network (SGGNN) [36], where the authors developed a novel deep learning framework to utilize graph convolutional operator to learn probe-gallery relationships for person re-identification. Although our method adopts a similar way of passing deeply learned messages, our target is to optimize features rather than similarities. More importantly, our proposed IB-GAT is not a post-processing step like re-ranking in SGGNN, but to enhance the mapping capability of the feature embedding network through feedback loss in an end-to-end manner.

3 Proposed Approach

This section details the proposed approaches. We first describe the overall network architecture and then elaborate on the design of the FMF module and the GAT-based verification subnet. Finally, we present the similarity measurement process.

3.1 Overview

We propose a deep neural network architecture that formulates the problem of visual place recognition as binary classification. Given an input pair of images, the task is to determine whether or not the two images denote the same location. Fig. 2 illustrates the pipeline which consists of two main parts. In the first part, we first extract a set of features X from a mini-batch of images through CNN, where each feature uniquely represents the visual information of the corresponding image. The FMF module produces channel-wise recalibrated weights and we can obtain refined feature maps X' which are constrained by triplet loss. In the second part, we consider the set of filtered features from the same place as a node and build a complete graph. In the graph, each edge encodes the similarity between connected nodes (including self-joins). Then generated deeply learned messages are propagated among nodes to update the features associated with each node. The features $\tilde{X}$ are supervised by binary classification loss through a verification subnet. A feedback loss is also introduced to constrain the difference between the original features and the output features. It forces the feature embedding network to directly produce optimized features without applying graph attention networks.

3.2 Feature Map Filtering Module

Our network is integrated with a base network for dense feature extraction. The convolutional neural network takes an image as input and generates a 3-D representation of size $W \times H \times C$, where H and W are spatial dimensions and C is the channel dimension, otherwise termed the number of feature maps.

Unlike the off-line statistical methods used in [18, 16], we follow SRM [20] to re-evaluate the weight of the channel and select the important feature maps in an end-to-end manner. In the FMF module as shown in Fig. 3, we firstly squeeze global spatial information into a channel descriptor. This is performed by utilizing global average pooling to obtain channel-wise statistics. The standard deviation (Std Pooling) of each feature map is also computed to generate another channel descriptor. Formally, given the input feature maps $X \in \mathbb{R}^{N \times W \times H \times C}$, the two-channel descriptors $\mathbf{S} \in \mathbb{R}^{N \times C \times 2}$ are calculated by:

$$\mu_{nc}(x) = \frac{1}{HW} \sum_{h=1}^H \sum_{w=1}^W x_{nwhc} \quad (1)$$

Fig. 2 The overall depiction of the pipeline for our model. In this architecture, the CNNs take as input images and return as output a series of feature maps X . Following, the FMF module generates the filtered feature X' map which is then optimized by IB-GAT to $\tilde{X}$. The features $\tilde{X}$ are supervised by binary classification loss through verification subnet. The feedback loss enables the feature embedding network to directly produce refined features without applying graph attention networks.

Fig. 3 Diagram of the FMF Module. As illustrated, the FMF Module utilizes both average pooling and standard deviation pooling which is followed by fully connected layers and activation functions.

$$\sigma_{nc}(x) = \sqrt{\frac{1}{HW} \sum_{h=1}^H \sum_{w=1}^W (x_{nwhc} - \mu_{nc}(x))^2} \quad (2)$$

$$\mathbf{s}_{nc} = [\mu_{nc}, \sigma_{nc}] \quad (3)$$

where $\mathbf{s}_{nc}$ is the single element in $\mathbf{S}$. In the following, descriptor $\mathbf{S}$ is forwarded to a network to produce the channel attention map. The network is composed of two fully connected (FC) layers and two activation functions. More non-linear information is utilized than that of only using a fully connected layer directly, which can better fit the complex correlation among channels. Considering that we do not want to eliminate some layers but to suppress the effect of some unimportant feature maps through weights, thus the sigmoid layer is utilized to obtain the normalized weight between 0 and 1.

In short, the channel weight is computed as:

$$\mathbf{m} = \sigma_1(\mathbf{W}_2 \sigma_2(\mathbf{W}_1 \mathbf{S})) \quad (4)$$

where σ_1 and σ_2 denote the Sigmoid function and the ReLU function, respectively. $\mathbf{W}_1 \in \mathbb{R}^{\frac{C}{r} \times C}$ and $\mathbf{W}_2 \in \mathbb{R}^{C \times \frac{C}{r}}$ are the parameters of FC layers and r is a reduction ratio which reduces the computation by controlling the number of neurons in the first fully connected layer.

Finally, the original input X is rescaled by the weights $\mathbf{m} \in \mathbb{R}^{N \times C}$ and followed by pooling operation p :

$$\mathbf{x}'_c = p(\mathbf{m}_c \cdot x) \quad (5)$$

where $X' = [\mathbf{x}'_1, \mathbf{x}'_2, \dots, \mathbf{x}'_C]$, $\mathbf{m}_c$ is the element in $\mathbf{m}$, and x represents a single feature map in X .

Then the network is trained through a triplet embedding scheme, where three features $X'(I_a)$, $X'(I_p)$ and $X'(I_n)$ are generated from the same network. $\{I_a, I_p\}$ is the positive pair includes two images captured

from the same location but own different characteristics. $\{I_a, I_n\}$ forms the negative pair which taken from varying locations but have similar characteristics. The generated features X' is constrained by the following triplet loss:

$$L_t = \sum_i^N \max(\|X'(I_a) - X'(I_p)\|_2 - \|X'(I_a) - X'(I_n)\|_2 + \alpha, 0) \quad (6)$$

which is zero when the distance of the negative pair is larger than the distance of the positive pair by at least a margin α . N is the number of all triplets in a batch. Actually, there are some other improved loss functions based on triplet loss, such as Lifted Struct [31], Proxy-NCA [28] and Ranked List Loss [44], are all applicable in this network. In the process of training, in order to reduce the loss value, the network will select the channel weights of feature maps through the FMF module, which will pay more attention to the features that are not related to appearance changes and suppress the features related to appearance changes. In this way, the distance between the positive sample pairs will be smaller, and the distance between the negative sample pairs will be larger, which makes it easier to achieve the optimization goal.

3.3 Influence Based Graph Attention Network

IB-GAT is designed to address the VPR problem where lots of valuable mutual information between features is neglected. Like the attentional structure in [42], IB-GAT employs similar the attention mechanism but propagates the different message. We start with introducing the graph formulation of IB-GAT and then concentrate on building the message generation mechanism.

3.3.1 Graph Formulation

The input of IB-GAT is a set of filtered features that is processed by the FMF module and pooling operation such that $X' = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N\}$, $\vec{x}_i \in \mathbb{R}^{d'}$, where N is the size of the mini-batch, and d' is the dimension of single feature. The proposed IB-GAT generates a new set of refined features $\tilde{X} = \{\vec{\tilde{x}}_1, \vec{\tilde{x}}_2, \dots, \vec{\tilde{x}}_N\}$, $\vec{\tilde{x}}_i \in \mathbb{R}^{\tilde{d}}$ as output. Usually, the dimension of the input is the same as that of the output, which means $d' = \tilde{d}$.

IB-GAT mainly operates on the graph structure. Given the set of input features X' , we construct an undirected complete graph $G(V, E)$:

$$V = \{v_1, v_2, \dots, v_N\} \quad (7)$$

Fig. 4 The network architecture of the message generation.

where V denotes the set of nodes. Each node represents a single feature in X' . Edges E are composed of the distances between any two nodes. In this work, we measure the edge e_{ij} between two nodes $\vec{x}_i$ and $\vec{x}_j$ using the Euclidean distance:

$$e_{ij} = \|\vec{x}_i - \vec{x}_j\|_2 \quad (8)$$

The attention mechanism is generally taken advantage of to depict the relative importance between two features. In this formulation, since the relation between two nodes is determined by the edge, we can simply construct a function $f: \mathbb{R} \rightarrow \mathbb{R}$ to map edge to attention. Specifically, it is defined as:

$$\alpha_{ij} = \exp(-e_{ij}^2/\gamma) \quad (9)$$

where α_{ij} means the relevance between the i -th node and j -th node, and γ is a hyper-parameter. As we can see, the shorter the distance between nodes, the higher the attention is gained. Since the complete graph we constructed includes only a mini-batch of filtered features, we can compute the attention for every node with all other nodes. To make the attention belonging to different nodes comparable, L1 normalization is performed:

$$\alpha_{ij} = \frac{\exp(-e_{ij}^2/\gamma)}{\sum_{k \in N} \exp(-e_{ik}^2/\gamma)} \quad (10)$$

In this way, we can obtain a normalized attention matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$. Note that the diagonal elements $\mathbf{A}_{ii}$ are set to 0 to avoid self-enhancing.

3.3.2 Influence Generation and Node Updating

Instead of integrating original features, we propagate the generated influence among all connecting nodes. This kind of propagation mechanism is able to avoid the invalidity of node information to some extent. The features of nodes are then updated using weighted fusion which combines all input influence messages and the node's original features. Before message propagating begins, every node will first encode an influence message for sending to other connected nodes. We adopt the 2-FC layer message network proposed in [36]. The

Fig. 5 A comparison of the node updating process between the conventional GATs and the proposed IB-GAT. (a) Node updating of conventional GATs. (b) Node updating of IB-GAT.

feature of each node is fed into a network with two FC layers, BN layers, and ReLU functions to generate an influence message z_i , which is illustrated in Fig. 4. Such a design is equivalent to a transformation of feature space, which can extract and integrate the useful information of node features. The fully connected layer and activation function are to increase the nonlinear expression ability of the model. And the BN layer plays the role of regularization.

This network is able to learn more meaningful information for updating the node's feature:

$$z_i = I(\vec{x}_i) \quad \text{for } i = 1, 2, \dots, N \quad (11)$$

where $I(\cdot)$ represents the influence message generation network.

While updating the features, the conventional GATs only use the attention matrix $\mathbf{A}$ to achieve feature fusion by linear combination. As we have also obtained the attention matrix $\mathbf{A}$ and the influence message z_i generated by each node, the update for a single node $\vec{x}_i$ is:

$$\vec{x}_i = (1 - \beta)\vec{x}_i + \beta \sum_{k \in N} \alpha_{ik} z_k \quad (12)$$

where $\vec{x}_i$ denotes the i -th refined node feature, β is the weighting parameter that balances the original feature and the received influence message.

The node updating procedure of IB-GAT compared with the conventional GATs is illustrated by Fig. 5. Note that the convolutional GATs utilize attention values to compute a linear combination of the node features corresponding to them to serve as the final output node features. As a comparison, the proposed IB-GAT applies different messages learned from a deep network. After such an updating operation, each node is influenced by the information passed from the surrounding nodes. Therefore, it is expected to generate improved features and result in significant improvement in visual place recognition.

3.3.3 Feedback Loss

Instead of optimizing similarities between images like [36], the proposed method aims at learning optimized features. It is not always applicable to process the image set as graphs while testing the performance. Thus, we intend to directly generate optimized features using feedback loss. We utilize the mean squared error (MSE) loss to constrain the difference between the original feature and the optimized features:

$$\mathcal{L}_{FL} = \sum_{i \in N} \left\| \vec{x}_i - \vec{x}_i' \right\|_2^2 \quad (13)$$

where feedback loss is just an auxiliary module to avoid the time-consuming operation of building graphical models. In this way, we can extract features from X' without any extra pre-processing method in the testing procedure.

3.4 Verification Subnet

The verification subnet accepts optimized feature pairs generated by the previous component as input. It calculates the distance between two features followed by a fully connected layer with two softmax outputs. Specifically, the difference features are obtained by element-wise subtraction and square operations. We denote the input features as $\vec{x}_i$ and $\vec{x}_j$, the difference D is computed as:

$$D = [\vec{x}_i - \vec{x}_j] \cdot [\vec{x}_i - \vec{x}_j] \quad (14)$$

where $\cdot$ represents the element-wise multiplication. Since our task is to determine the similarity among nodes, a naive method would be simply feeding the difference features into a binary classifier. The output indicates the probability of "same location" (1) or "different locations" (0). For each image pair in the mini-batch training, a binary cross-entropy loss function is employed:

Table 1 Tolerances for true positives matches.

Dataset	Location Tolerances
Nordland	10 frames
Alderley	2 frames
Gardens Point Walking	3 frames
St Lucia	30 meters
Pittsburgh250k	25 meters
Tokyo 24/7	25 meters

$$L_{ce} = - \sum_{i=1}^N y_i \log(f(D_i)) + (1 - y_i) \log(1 - f(D_i)) \quad (15)$$

where y_i is the ground-truth label of the fed image pair with 1 or 0.

3.5 Similarity Measurement for VPR

In order to evaluate the performance of the proposed algorithm, we use the single-frame matching method and measure the similarity between features. The final feature vector $\hat{F}$ can be obtained from X' . The query feature $\hat{F}_q$ of the query location l_q and the database feature vector $\hat{F}_{db}$ are compared using the cosine distance as in equation (16)

$$s(\hat{F}_q, \hat{F}_{db}) = \frac{\hat{F}_q \cdot \hat{F}_{db}}{\|\hat{F}_q\| \|\hat{F}_{db}\|} \quad (16)$$

The location l_s with the minimum distance to the query location l_q is regarded as a true positive match if it is from the same location as l_q (within dataset tolerances—see Table 1 for a summary of tolerances).

4 Experimental Evaluation

4.1 Implementation Details

We built the networks based on AlexNet [19] pre-trained on ImageNet. We generate features from layers Conv2 to Conv5. By experimenting on different layers, the layers where the FMF module provides better performance could be determined. We also perform data augmentation with random horizontal flipping and resize all input images to 224×224 pixels.

The training process consists of two important stages. In the first stage, we train the feature embedding network with the FMF module using triplet loss. In the second stage, we boost the filtered features using IB-GAT. To construct each mini-batch, we randomly sample images according to their locations and characters

such as day/night, winter/summer and sunny/cloudy. A mini-batch includes B (usually equals 8) randomly selected locations and K images for each location, K is the sum of different characteristics. In this experiment, the positive pair includes two images captured from the same location but with their own different characteristics. The negative pairs are images taken from varying locations but have similar characteristics. Besides, we adopt the sampling method of hard triplets selection for each mini-batch and finally get a total of $B \times K$ triplets. Note that we use the same mini-batch setting for the aforementioned two stages. In the testing stage, we directly extract optimized features which are enhanced by IB-GAT with feedback loss.

Hyper-parameters: We use the stochastic gradient descent algorithm with an initial learning rate of 0.01 on all datasets and reduce the learning rate gradually by a factor of 10 after 2 epochs. The margin α in triplet loss is set to 0.1. The balancing weight β is set to 0.9, γ is set to 200. It is useful to add dropout to improve the robustness of the model. The dropout rate is set as 0.5.

4.2 Datasets

To evaluate the proposed approach, datasets are required to traverse the path in different environments but without too much view-point change. Moreover, ground truth information, such as the corresponding scenes should be contained in the datasets.

Nordland Dataset: The Nordland dataset [37] is one of the most challenging place recognition datasets due to the changing landscape and weather, as Fig. 6(a) illustrates. It includes four simultaneous video streams of different seasons. Each 9-hour video corresponds to a season, and they were manually aligned so that frames with the same numeral are from the same location. In addition to the extreme changes in appearance produced by the season, these images also include extreme blurring because of the train's excessive speed. We extract the image from video at a rate of a frame per second removing all frames where the train was in a tunnel or stationary. Then the sequence is divided into two parts, one for training (27000 images), and the other for testing (1000 images). We use a ground truth tolerance of 10 frames.

Alderley Dataset: The Alderley dataset was first introduced in SeqSLAM [27]. It consists of two videos, one on rainy nights and the other on sunny days. Fig. 6(b) shows an example that contains severe changes in illumination and weather conditions in a given location. It is difficult to identify the same place from

Fig. 6 Randomly selected sample images from the dataset. (a) Images in Nordland Dataset. (b) Images in Alderley Dataset. (c) Images in Gardens Point Walking Dataset. (d) Images in St.Lucia Dataset. (e) Images in Pittsburgh Dataset. (f) Images in Tokyo 24/7 Dataset.

these two images even for humans. Frame correspondences are provided in the dataset for place recognition as ground-truth. We used the first 2000 frames of the sequence for the test set, and the rest 12607 frames for the training of the network. The ground truth tolerance is set to 2 frames.

Gardens Point Walking: This dataset was recorded on the Gardens Point Campus of QUT university campus. There are three traverses along the same route, one during the night and two during the day as shown in Fig. 6(c). Each traverse contains 200 images. The night traverse is recorded from the right side of the pathway (night_right). The two day traverses are viewed from the left and right side of the pathway, respectively (day_left, day_right). This dataset has provided frame-level correspondences. We train our network on the comparison between the night_right sequence and day_right sequence. We then select 140 images from each traverse for training and the rest 60 images for testing. The ground truth tolerance is set to 3 frames.

St Lucia: The St Lucia dataset [13] consists of several car traverses through the suburb of St Lucia, Queensland. The videos are captured with a forward-facing camera placed on the roof of a car across five different times of day. We train the network and test on the early morning sequence (time:190809_0845) and the late afternoon sequence (time:180809_1545) which contains significant appearance changing. We use the provided GPS information and set ground-truth tolerance to 30 meters. The images are extracted from the 15 FPS videos. The first 3500 images are used for training and the next 500 are for evaluation.

Pittsburgh250k: The Pittsburgh250k dataset [41] includes 250k perspective images generated from Google Street View panoramas of the Pittsburgh area, which are equally divided into training, validation and test-

ing sets. The query images were obtained in a different session that depict the same places from different seasons, illumination conditions and viewpoints. For fast training, we adopt the smaller subset Pittsburgh30k (Pitts30k), which consists of 30k database images. The training set contains 10k images and the test set is composed of 1k images. We use the provided GPS information and the ground-truth tolerance is set to 25m.

Tokyo 24/7: The Tokyo 24/7 dataset [40] contains 76k database images and 315 query images captured by iPhone5s and SonyXperia smartphones. At each location, images are captured at 3 different viewing directions and at 3 different times of day. The ground truth GPS coordinates at each location were recorded, and the ground-truth tolerance is set to 25m.

4.3 Comparison Methods

To evaluate the performance of the proposed method, we compare it with several different state-of-the-art methods:

- (a) **Conv3** [38] is a learning-based method which extracts features from the Conv3 layer of the AlexNet. It is one of the first methods to use the deep convolutional feature to deal with the problem of visual place recognition. The version in Python is implemented and the Gaussian random projection is applied to compress the original 64896-dimensional vector to a smaller size.
- (b) **NetVLAD** [1] achieved weakly supervised training for place recognition using a CNN architecture that embeds a traditional VLAD layer. We use the Pytorch implementation¹ of NetVLAD with the hard-triplet Loss.

¹ <https://github.com/lyakaap/NetVLAD-pytorch>

- (c) **CALC** [25] proposed an unsupervised convolutional autoencoder network architecture for robust and fast loop closure in visual SLAM. We utilize the open-source implementation ² and fine-tune the model using our data.
- (d) **Landmarks** [39] presented a ConvNet landmark-based approach for visual place recognition. Landmarks are extracted using Edge Boxes [49] and the number of landmark proposals per image for these experiments is set to 50.
- (e) **SeqSLAM** [27] was a visual place recognition approach utilizing sequence images. It was designed to deal with visual navigation under changing conditions. We implement it using the `openseqslam` ³ in Matlab .

4.4 Evaluation Metric

The designed methodology for testing performance is principally based on precision-recall curves, which are calculated from the similarity matrix obtained in each test set. A threshold is set and used in the matching process between the similarity matrix and the ground-truth matrix. In this way, the occurrence times of TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative) on the dataset are obtained. The values of precision (P) and recall (R) are calculated as follows:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN} \quad (17)$$

By scanning the threshold of similarity score in a uniform distribution between 0 and 1, we can obtain the precision-recall curve.

Maximum recall at 100% precision: the proportion of correct matches that can be achieved with no false positives. This can be observed visually in any precision-recall curve, as it will be the recall rate where the precision first dips down from 1.0 and a higher value is desired.

Maximum F1 score: F1 score is a balance between precision and recall, and we intend to obtain the maximum value. F1 score is calculated by the harmonic mean of precision and recall:

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \quad (18)$$

4.5 Ablation Analysis

We also investigate the effectiveness of each component in the proposed method by performing several exper-

iments on the Nordland and Gardens Point datasets. Table 2 shows the maximum F1 score and maximum recall at 100% precision across different components.

FMF module. As one of the main ideas of the proposed method is to eliminate or suppress the influence of the appearance changing, we now investigate how greatly it is achieved. We use AlexNet as the baseline and extract the original features from the Conv3 layer. We first verify the benefit of the adopted pooling method compared to different other pooling operations. Fig. 7 shows the results of varying pooling operations in the FMF module with the training of triplet loss. While each single pooling component of the FMF module brings meaningful promotion of performance, the integration of them further improves the performance. Compared with the combination of MaxPool and AvgPool presented in CBAM [46], the proposed FMF module achieves about 4% improvement on Max F1 score and Recall value on both datasets (0.55569 $\rightarrow$ 0.57684, 0.19576 $\rightarrow$ 0.20285, 0.63579 $\rightarrow$ 0.66308, 0.37272 $\rightarrow$ 0.38654). These results validate the effectiveness of the adopted AvgPool and StdPool in the experiment.

The reduction ratio r is also an important hyperparameter which affects the computation of FMF module. To get the best value, we conduct experiments on the Nordland dataset for a range of different r . The comparison is shown in Fig. 8 and Table 3, which reveal the performance does not improve monotonically with increased r . In particular, we found that setting $r = 8$ achieved a good tradeoff between computation and performance, thus we used this value for all experiments.

Furthermore, we show the percentage of feature maps with the weight of 0 in Table 5. From the results we can infer that the lower the network layer, the more feature maps need to be filtered out. We also display the recalibrated weights of the feature maps in Fig. 9 according to the images sampled from Nordland Dataset. It is noted that the regions with a value of 0 are basically the same even if the features are not from the same place or the same season. And only a few weights of the feature maps are different. This supports our claim that FMF module learns to suppress the contribution of confusing feature maps, which helps the network to concentrate more on meaningful features.

IB-GAT. The original GATs [33] were proposed to handle the problem of classifying nodes on a graph. We take advantage of it and improve it by propagating influence from nodes instead of the original features. In order to examine the effectiveness of IB-GAT and understand its role more intuitively, we randomly selected several places from Nordland dataset with their corresponding images and showed the similarity matrix after the IB-GAT operation. We compare the result with

² <https://github.com/rpng/calcul>

³ <https://openslam-org.github.io/openseqslam.html>

Table 2 Maximum F1 score and recall value at maximum precision across different components on the Nordland dataset and GardensPoint dataset.

	Nordland		GardensPoint	
	Max F1 score	Recall at 100% Precision	Max F1 score	Recall at 100% Precision
Conv3	0.52352	0.01939	0.5974	0.30732
Conv3_trip	0.54898	0.05816	0.60111	0.33224
Conv3_trip_AvgPool	0.55751	0.12693	0.6497	0.36312
Conv3_trip_StdPool	0.57141	0.14143	0.64228	0.3513
Conv3_trip_MaxPool	0.55227	0.15313	0.63343	0.33972
Conv3_trip_AvgPool_MaxPool	0.55569	0.19576	0.63579	0.37272
Conv3_trip_AvgPool_StdPool(FMF)	0.57684	0.20285	0.66308	0.38654
Conv3_trip_GAT	0.55548	0.10033	0.59507	0.33344
Conv3_trip_IB-GAT	0.55658	0.14427	0.61517	0.32376
Conv3_trip_FMF_GAT	0.63333	0.17651	0.65716	0.38428
Conv3_trip_FMF_IB-GAT	0.63565	0.23148	0.67758	0.40964
Conv2_trip	0.4788	0.04728	0.53374	0.18957
Conv2_trip_FMF	0.55667	0.13375	0.60683	0.24417
Conv3_trip	0.54898	0.05816	0.60111	0.33224
Conv3_trip_FMF	0.57684	0.20285	0.66308	0.38654
Conv4_trip	0.49541	0.05405	0.5845	0.31392
Conv4_trip_FMF	0.55997	0.18023	0.62389	0.37503
Conv5_trip	0.48576	0.03898	0.56771	0.29557
Conv5_trip_FMF	0.50751	0.06411	0.63017	0.38368

Table 4 Parameter sizes and the performance for different reduction ratio values on the Nordland dataset.

Reduction Ratior	Million Parameters	Max F1-score	Recall at 100% precision
2	1.38	0.594	0.181
4	1.30	0.581	0.199
8	1.26	0.583	0.205
16	1.24	0.580	0.197

the situation without using IB-GAT. It can be clearly seen in Fig. 10 that similarity scores between samples belonging to the same location are larger, while the similarity scores between samples belonging to different locations are decreased visibly. Then, we conduct experiments to verify that the proposed IB-GAT is better suited than GAT for VPR tasks. The PR-curves can be found in Fig. 12. IB-GAT has always achieved better performance than GAT whether the FMF module is adopted or not, except for the case that the recall at 100% precision on GardensPoint dataset.

Moreover, we adopt the classical Calinski-Harabaz Index (CHI) [4] to evaluate the clustering characteristics of features. For each dataset, we randomly selected 50 locations with a certain number of images according to their appearance changes. The results of CHI are shown in Fig. 11 respectively. We find that the proposed IB-GAT improves the clustering characteristics of the features slightly compared with the original GAT.

Layers. We provide a thorough investigation of the utility of different layers in the AlexNet from Conv2 to Conv5 and evaluate their gain under the use of the FMF module. To ensure a fair comparison, the triplet loss method is employed during training. For AlexNet

on Nordland, the FMF module does not always improve the localization performance across all layers (Fig. 13(a)). Except for Conv5, other layers find a significant improvement using the FMF module. When filtering is applied to Gardens Point (Fig. 13(b)), it is expected that all tested layers obtain great gains. Among all layers, Conv2 always keeps the highest improvement rate on Max F1 score. But for the actual performance, Conv3 always performs best in both datasets.

Component. We add each component to the baseline and present the results in Fig. 14. In overview, the FMF module and IB-GAT operation both play a significant role. On Nordland datasets, the FMF module get 0.02786 and 0.14469 improvement comparing with the case of Conv3 with triplet loss for the F1 Score and Recall value, respectively. Correspondingly, the performance of only utilizing IB-GAT operation increase 0.0076 and 0.08611. The FMF module seems to have a bigger impact on the performance than IB-GAT and similar results have also been found in Gardens-Point dataset. What’s more, the maximum benefit is achieved when the two components are combined. We infer that the FMF module successfully suppresses the influence of the appearance changing. However, FMF module only affects the individual features of a single location and ignores the correlation of multiple locations. Then, the introduction of IB-GAT just integrates the influence from other locations and further improve the discriminative ability of final features.

Table 5 The number of feature maps pre-filtering and post-filtering on the Nordland dataset and Gardens Point Walking Dataset.

Layers	Total Map Count	Nordland		GardensPoint	
		Filtered Map Count	%	Filtered Map Count	%
Conv_2	256	76	29.7	60	23.4
Conv_3	384	88	22.9	77	20.1
Conv_4	384	93	24.2	74	19.3
Conv_5	256	66	25.8	52	20.3

Fig. 7 Comparison of different pooling methods in FMF module. (a) PR-curves on Nordland Dataset. (b) PR-curves on Gardens Point Walking Dataset.

4.6 Comparison with State-of-the-art Methods

This subsection reports the PR-curves of all compared methods across six datasets. We also display some matched exemplar images in Fig. 24. While some examples have been retrieved correctly by most approaches, a few images still get incorrectly matched results.

Results on Nordland Dataset. Fig. 15 plots PR-curves of all methods on the Nordland spring-winter

Fig. 8 Comparison of different reduction ratio values of FMF module on the Nordland dataset.

sequence. The proposed approach significantly outperforms all other methods except for the SeqSLAM. In this dataset, the velocity in query sequence is equal to the velocity in the reference traverse. This gives the sequence-based approaches more advantages than other methods. Surprisingly, the Landmarks method underperformed with a big margin. This may be because there are not enough obvious objects or targets in this dataset. From the Fig. 22 and Fig. 23, we observe that our method achieves a precision of 23% at 100% recall which is greater than other methods, and gets 23% promotion over the original Conv3 in terms of the maximum F1 Score. It mimics that the employment of FMF module and IB-GAT is very productive under strong conditional changes.

Results on Alderley Dataset. The PR curves plotted in Fig. 16 show an acceptable accuracy for the proposed method in this challenging case. Interestingly, the NetVLAD and the CALC method both perform well here, but SeqSLAM achieves bad results due to losing the advantage of equal velocity. We can see that the proposed method maintains a higher value (0.091) on maximum recall at 100% precision. Although the benefit of using the FMF module and IB-GAT further increases, it still can't provide the best performance in terms of F1-score which is 0.03 lower than NetVLAD.

Fig. 9 Re-calibrated weights of the feature maps. (a) Weights with the same location across different seasons. (b) Weights with the same season across different locations.

Fig. 10 Visualization of similarity matrix. (a) Similarity matrix without using IB-GAT. (b) Similarity matrix after using IB-GAT.

Results on Gardens Point Walking Dataset. Though this dataset exhibits strong illumination, the majority of the tested VPR approaches perform relatively well as displayed in Fig. 17. This is because the distinctive objects or structures are captured in both the traverses. The proposed method achieves superior results both in the maximum F1-score (0.68) and recall values at max precision (0.41), while Landmarks and NetVLAD also perform nearly well on this dataset. The Landmarks method is intended to find regions that are likely to contain an object in an image, which just fits the characteristics of this dataset. One of the reasons for the good performance of NetVLAD could be its training on a huge urban place-based dataset (Pittsburgh

Fig. 11 CHI of using IB-GAT and GAT to refine the features on different datasets.

250K [41]), which includes confusing and dynamic objects along with strong illumination changes.

Results on St Lucia Dataset. The results in Fig. 18 illustrate the precision-recall performance on the St Lucia Dataset. As expected, our method achieves the best performance in terms of the maximum F1-score (0.63) and recall values at max precision (0.11). There is a significant gap between our method and the original Conv3, with an improvement in F1 score of 0.1. NetVLAD, CALC, and original Conv3 have shown similar performance.

Results on Pittsburgh250k Dataset. The PR curves plotted in Fig. 19 show competitive accuracy for the proposed method in this challenging case. Our method achieves the second-best performance in terms of Max F1-score (0.58) followed by NetVLAD (0.59). Landmarks and Conv3 have shown similar performance on this dataset. CALC obtain slightly better results than SeqSLAM. With respect to the recall value at max precision, all methods perform not well. We believe this is because this dataset contains more obvious changes in viewpoints and varied objects.

Results on Tokyo 24/7 Dataset. The PR performance on the Tokyo 24/7 dataset is shown in Fig. 20. It is notable that NetVLAD has achieved better results (0.82 Max F1-score) than all other methods, while the proposed method follow-ups with relatively poor performance (0.73 Max F1-score). To be mentioned, our method gets a better recall at maximum precision than NetVLAD. Conv3 and Landmarks suffer noticeable performance degradation, with respect to CALC. This experiment shows that on such large-scale datasets, the proposed method is still effective and maintain relatively stable performance.

The Number of Mismatched Pairs on Different Datasets. For each method, we count the number of mismatched appearance varied image pairs on different datasets. In particular, to make a fair comparison, we conducted experiments on the similarity

Fig. 12 Comparison of different GATs operation. (a) PR-curves on Nordland Dataset. (b) PR-curves on Gardens Point Walking Dataset.

Fig. 13 Comparison of different layers in AlexNet. (a) PR-curves on Nordland Dataset. (b) PR-curves on Gardens Point Walking Dataset.

threshold when each method obtains Maximum F1-score. The results are illustrated in Fig. 21. The number of image pairs involved in the test is different for each dataset, so the total number of mismatches varies greatly. NetVLAD is still very stable and Landmarks and CALC are relatively mediocre. On the Tokyo and Pittsburgh datasets, our method obtains more mismatches than NetVLAD. This may be because there are more viewpoints changes in these two datasets, which brings challenges to our method. But on other datasets, the proposed approach performs competitively.

4.7 Computational Performance

Deep learning methods are computationally intensive, which make it essential to test the run-time performance in order to realize the VPR application for robots. We report the computational cost in terms of the running time for (1) feature extraction by the networks, (2) feature matching between reference and a single query image. Table 6 shows evaluation results on the Nordland Dataset. We run experiments on 2300 images and record the average runtime. The SeqSLAM utilizing CPU processing with MATLAB keeps the fastest because it is not necessary to extract features. As expected, CNNs-based approaches always take more time to encode an image. The Landmarks are inefficient owing to the repeated use of EdgeBoxes algorithm. For

Fig. 14 Comparison of different components in the whole architecture. (a) PR-curves on Nordland Dataset. (b) PR-curves on Gardens Point Walking Dataset.

Fig. 15 Results on Nordland dataset.

Fig. 17 Results on Gardens Point Walking Dataset.

Fig. 16 Results on Alderley Dataset.

Fig. 18 Results on St Lucia Dataset.

Fig. 19 Results on Pittsburgh Dataset.

Fig. 20 Results on Tokyo 24/7 Dataset.

Fig. 21 The number of mismatched pairs for each method on different datasets.

a single image, the feature encoding of the proposed method takes approximately 29ms using Tensorflow on NVIDIA TITANX GPU with 12GB GPU memory. Feature matching between two images using the Python implementation costs approximately 0.33ms.

Fig. 22 Maximum F1 score of different methods on six datasets.

Fig. 23 Recall at maximum precision of different methods on six datasets.

Table 6 Processing Time Comparison on The Nordland Dataset

VPR system	Feature Extraction (ms)	Feature Matching (ms)
NetVLAD	980	0.038
CALC	39	0.31
Landmarks	2035	510
SeqSLAM	0	2.97
Conv3	136	0.34
Conv3_trip_FMF_IB-GAT	29	0.33

5 Conclusion and future works

The contribution of this paper has been to demonstrate how the FMF module and IB-GAT can be combined to augment visual place recognition capability of the middle convolutional feature map. In the first part of proposed architecture, we use a learnable FMF module to suppress confusing feature maps. In the second part, the use of IB-GAT enables filtered features by FMF to extend over the influence of other feature nodes. In this way, the ignored global mutual information is exploited to produce robust features with more discriminative power. Moreover, a feedback loss is introduced to make it possible to combine the two parts for end-to-end training. To evaluate the performance, we tested all important components of the proposed system and conduct experiments based upon six public datasets. The promising experimental results show that the proposed

Fig. 24 Samples of matched/mismatched images by various methods. Each column represents a query and matched images of different methods. Images with green frames are correct matches, while the ones with red frames are incorrect matches.

method is effective and outperforms other state-of-the-art methods in most cases.

The work discussed here could be improved by further optimization of the hyper-parameters or using deeper network structure. In the future, it would be interesting to investigate the response of regional features to the changes in appearance. Further work will also explore other types of graph neural networks for visual place recognition, such as graph embedding and graph convolutional network.

Acknowledgements Supported by Open Foundation of Zhi-jiang Laboratory(No. 2020NB0AB01), National Natural Science Foundation of China (No. 61973066, 61471110), Fundamental Research Funds for the Central Universities(N172608-005, N182608004), Equipment Pre-research Foundation(6140-3120111) and the Fundation of Key Laboratory of Aerospace System Simulation(6142002301).

References

1. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: Netvlad: Cnn architecture for weakly supervised place recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5297–5307 (2016)
2. Arroyo, R., Alcantarilla, P.F., Bergasa, L.M., Romera, E.: Fusion and binarization of cnn features for robust topological localization across seasons. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4656–4663. IEEE (2016)
3. Bruna, J., Zaremba, W., Szlam, A., LeCun, Y.: Spectral networks and locally connected networks on graphs. arXiv preprint arXiv:1312.6203 (2013)
4. Caliński, T., Harabasz, J.: A dendrite method for cluster analysis. Communications in Statistics-theory and Methods **3**(1), 1–27 (1974)
5. Chen, Z., Jacobson, A., Sünderhauf, N., Upcroft, B., Liu, L., Shen, C., Reid, I., Milford, M.: Deep learning features at scale for visual place recognition. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 3223–3230. IEEE (2017)
6. Chen, Z., Lowry, S., Jacobson, A., Ge, Z., Milford, M.: Distance metric learning for feature-agnostic place recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2556–2563. IEEE (2015)
7. Chen, Z., Maffra, F., Sa, I., Chli, M.: Only look once, mining distinctive landmarks from convnet for visual place recognition. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9–16. IEEE (2017)
8. Duvenaud, D.K., Maclaurin, D., Iparraguirre, J., Bombarell, R., Hirzel, T., Aspuru-Guzik, A., Adams, R.P.: Convolutional networks on graphs for learning molecular fingerprints. In: Advances in neural information processing systems, pp. 2224–2232 (2015)
9. Gálvez-López, D., Tardos, J.D.: Bags of binary words for fast place recognition in image sequences. IEEE Transactions on Robotics **28**(5), 1188–1197 (2012)
10. Gao, J., Zhang, T., Xu, C.: Graph convolutional tracking. In: CVPR (2019)

11. Garg, S., Suenderhauf, N., Milford, M.: Don't look back: Robustifying place categorization for viewpoint- and condition-invariant place recognition. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 3645–3652. IEEE (2018)
12. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp. 1263–1272. JMLR. org (2017)
13. Glover, A., Maddern, W., Milford, M., Wyeth, G.: FAB-MAP + RatSLAM: Appearance-based SLAM for Multiple Times of Day. In: ICRA. Anchorage, USA (2010)
14. Guo, J., Potkonjak, M.: Pruning convnets online for efficient specialist models. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 113–120 (2017)
15. Han, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149 (2015)
16. Hausler, S., Jacobson, A., Milford, M.: Filter early, match late: Improving network-based visual place recognition. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3268–3275 (2019). DOI 10.1109/IROS40897.2019.8967783
17. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141 (2018)
18. Kenshimov, C., Bampis, L., Amirgaliyev, B., Arslanov, M., Gasteratos, A.: Deep learning features exception for cross-season visual place recognition. Pattern Recognition Letters **100**, 124–130 (2017)
19. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105 (2012)
20. Lee, H., Kim, H.E., Nam, H.: Srm: A style-based recalibration module for convolutional neural networks. arXiv preprint arXiv:1903.10829 (2019)
21. Li, X., Jie, Z., Feng, J., Liu, C., Yan, S.: Learning with rethinking: Recurrently improving convolutional neural networks through feedback. Pattern Recognition **79**, 183–194 (2018)
22. Lopez-Antequera, M., Gomez-Ojeda, R., Petkov, N., Gonzalez-Jimenez, J.: Appearance-invariant place recognition by discriminatively training a convolutional neural network. Pattern Recognition Letters **92**, 89–95 (2017)
23. Lowry, S., Sünderhauf, N., Newman, P., Leonard, J.J., Cox, D., Corke, P., Milford, M.J.: Visual place recognition: A survey. IEEE Transactions on Robotics **32**(1), 1–19 (2015)
24. McManus, C., Churchill, W., Maddern, W., Stewart, A.D., Newman, P.: Shady dealings: Robust, long-term visual localisation using illumination invariance. In: 2014 IEEE international conference on robotics and automation (ICRA), pp. 901–906. IEEE (2014)
25. Merrill, N., Huang, G.: Lightweight unsupervised deep loop closure. arXiv preprint arXiv:1805.07703 (2018)
26. Middelberg, S., Sattler, T., Untzelmann, O., Kobbelt, L.: Scalable 6-dof localization on mobile devices. In: European conference on computer vision, pp. 268–283. Springer (2014)
27. Milford, M.J., Wyeth, G.F.: Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In: 2012 IEEE International Conference on Robotics and Automation, pp. 1643–1649. IEEE (2012)
28. Movshovitz-Attias, Y., Toshev, A., Leung, T.K., Ioffe, S., Singh, S.: No fuss distance metric learning using proxies. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 360–368 (2017)
29. Naseer, T., Oliveira, G.L., Brox, T., Burgard, W.: Semantics-aware visual localization under challenging perceptual conditions. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 2614–2620. IEEE (2017)
30. Noh, H., Araujo, A., Sim, J., Weyand, T., Han, B.: Large-scale image retrieval with attentive deep local features. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3456–3465 (2017)
31. Oh Song, H., Xiang, Y., Jegelka, S., Savarese, S.: Deep metric learning via lifted structured feature embedding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4004–4012 (2016)
32. Polyak, A., Wolf, L.: Channel-level acceleration of deep face representations. IEEE Access **3**, 2163–2175 (2015)
33. Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. IEEE Transactions on Neural Networks **20**(1), 61–80 (2008)
34. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 815–823 (2015)
35. Shen, Y., Li, H., Yi, S., Chen, D., Wang, X.: Person re-identification with deep similarity-guided graph neural network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 486–504 (2018)
36. Shen, Y., Li, H., Yi, S., Chen, D., Wang, X.: Person re-identification with deep similarity-guided graph neural network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 486–504 (2018)
37. Sünderhauf, N., Neubert, P., Protzel, P.: Are we there yet? challenging seqslam on a 3000 km journey across all four seasons. In: Proc. of Workshop on Long-Term Autonomy, IEEE International Conference on Robotics and Automation (ICRA), p. 2013 (2013)
38. Sünderhauf, N., Shirazi, S., Dayoub, F., Upcroft, B., Milford, M.: On the performance of convnet features for place recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4297–4304. IEEE
39. Sünderhauf, N., Shirazi, S., Jacobson, A., Dayoub, F., Pepperell, E., Upcroft, B., Milford, M.: Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free. Proceedings of Robotics: Science and Systems XII (2015)
40. Torii, A., Arandjelovic, R., Sivic, J., Okutomi, M., Pajdla, T.: 24/7 place recognition by view synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1808–1817 (2015)
41. Torii, A., Sivic, J., Pajdla, T., Okutomi, M.: Visual place recognition with repetitive structures. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 883–890 (2013)
42. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. International Conference on Learning Representations (2018). URL <https://openreview.net/forum?id=rJXMpikCZ>. Accepted as poster
43. Wang, T.H., Huang, H.J., Lin, J.T., Hu, C.W., Zeng, K.H., Sun, M.: Omnidirectional cnn for visual place recognition and navigation. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 2341–2348. IEEE (2018)

44. Wang, X., Hua, Y., Kodirov, E., Hu, G., Garnier, R., Robertson, N.M.: Ranked list loss for deep metric learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5207–5216 (2019)
45. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)* **38**(5), 146 (2019)
46. Woo, S., Park, J., Lee, J.Y., So Kweon, I.: Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19 (2018)
47. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
48. Zhou, B., Bau, D., Oliva, A., Torralba, A.: Interpreting deep visual representations via network dissection. *IEEE transactions on pattern analysis and machine intelligence* (2018)
49. Zitnick, C.L., Dollár, P.: Edge boxes: Locating object proposals from edges. In: European conference on computer vision, pp. 391–405. Springer (2014)
50. Zou, J., Rui, T., Zhou, Y., Yang, C., Zhang, S.: Convolutional neural network simplification via feature map pruning. *Computers & Electrical Engineering* **70**, 950–958 (2018)