2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Neighborhood Consensus Guided Matching Based Place Recognition with Spatial-Channel Embedding

Kunmo Li[1], Yunzhou Zhang[1*], Jian Ning[1], Xinge Zhao[1], Guiyuan Wang[2], Wei Liu[2]

***Abstract*— As a crucial part of mobile robotics and autonomous driving, Visual Place Recognition (VPR) is usually addressed by recognizing its similar reference images from a pre-obtained database. However, VPR always suffers from environmental changes, such as weather, illumination, perceptual-aliasing and so on. To address this, we firstly introduce a robust and discriminative global descriptor aggregation technique that normalizes the spatial and channel dimensions of features. A Spatial-Channel Embedding (SCE) module is proposed to learn the spatial and scale information of features which make global features more discriminative. Meanwhile, the traditional re-ranking methods (e.g. RANSAC) for geometric consistency verification are time-consuming. Here we propose a Neighborhood Consensus Guided Matching (NCGM) module, which uses Neighborhood Consensus to filter the features from patch-level matching to achieve more accurate matching while reduces the time consumption. Through extensive experiments on multiple benchmarks, we demonstrate that our method outperforms several state-of-the-art methods while maintaining lower time consumption and storage requirements.**

I. Introduction

Visual Place Recognition (VPR) is a basic part of robot autonomous navigation, and it is also an important task of computer vision. Accurate place recognition helps the robot to recognize its own position, so as to complete the task well. In general, visual place recognition is usually regarded as an image retrieval problem [1][2], that is, given a query image, it needs to find the most similar image from the database for matching. According to the type of image descriptors, image descriptors can be divided into global features [2][3][4][5][6] and local features [7][8]. Traditional image descriptors such as SIFT [9] and SURF [10] are usually aggregated into a global feature vector to represent image information. With the advancement of deep learning technology, feature extraction methods based on neural networks have gradually demonstrated excellent performance, such as [11][12].

The current methods are roughly divided into two categories according to whether only global features are used, namely global retrieval and re-ranking. Global retrieval refers to the retrieval and matching of images using only global descriptors, usually using efficient K-NN operation. Representative works include NetVLAD [3] and its variants. Re-ranking, also known as the two-stage method, involves using global features to filter out the top-k images initially. Subsequently, the images are re-ranked using local feature characteristics, such as geometric verification [13], to complete the retrieval task. Representative works include [13] [14]. Although the re-ranking method is generally better than the global retrieval based method in terms of retrieval and matching, it also leads to a certain time delay due to the re-ranking operations such as spatial verification.

The VPR system faces various challenges in practical scenarios, such as dynamic objects, perceptual aliasing, seasonal changes, and viewpoint changes. It is important to extract relatively unchanged features from changing scenarios, while simultaneously maintaining low time consumption and memory usage to adapt to real-world scenarios. In this paper, the two stages of visual place recognition method are optimized. For the global retrieval, we focus on improving the discrimination and robustness of global descriptors, so as to obtain better coarse search results. Re-ranking of traditional methods is often time-consuming, which is less friendly for real-world scenarios. To address this, we focus on improving the accuracy of the matching while paying more attention to the design of the network to reduce the consumption of time and memory.

The contributions of this paper are as follows:

- We propose a Spatial-Channel Embedding (SCE) module to learn and normalize the spatial and channel dimensions of the feature to make the global descriptor more discriminative.
- We propose a Neighborhood Consensus Guided Matching (NCGM) module which uses Neighborhood Consensus to filter the features while maintaining lower time consumption and memory footprint.
- Extensive and thorough experiment results show that our method can outperform several state-of-the-art methods on several benchmarks.

II. Related Works

A. Global Retrieval

Traditional visual place recognition methods can be divided into two types based on local feature descriptors and global feature descriptors. Traditional methods such as SIFT [9] and SURF [10] are usually used to extract local features from images. BoW [15] uses K-means clustering for local features, then constructs a dictionary according to the cluster center. Place recognition methods based on deep learning are gradually being widely used because of their excellent

*The corresponding author of this paper
[1]Kunmo Li, Yunzhou Zhang, Jian Ning and Xinge Zhao are with College of Information Science and Engineering, Northeastern University, Shenyang 110819, China. zhangyunzhou@mail.neu.edu.cn
[2]Guiyuan Wang and Wei Liu are with Jiangsu Shuguang Optoelectronics Co., Ltd., Yangzhou 225009, China.
This work was supported by National Natural Science Foundation of China (No. 61973066) and Major Science and Technology Projects of Liaoning Province(No. 2021JH1/10400049).

performance. NetVLAD [3] trains global image descriptors in an end-to-end manner, which treats the output of CNNs as descriptor vectors. Subsequently, a series of variants of NetVLAD appear. For example, CRN [16] increases the performance of the network by introducing the idea of feature weighting. SFRS [17] proposes self-enhanced image region similarity labels to handle noisy GPS labels and achieves excellent performance on several standard benchmark datasets. Extracting relatively invariant features in the scene is of great significance to the accuracy of place recognition. DISP [18] is a method that integrates image domain adaptation into a self-supervised training framework for robust training. Image domain conversion is used as a preprocessing step before feature extraction, and finally achieves the effect of pixel-level domain alignment. A recent work, GCL [19], introduces a generalized contrastive loss (GCL) that exploits graded similarity of image pairs to learn effective descriptors for VPR. This approach is an efficient global feature based place recognition algorithm that does not require re-ranking.

B. Re-ranking

In recent years, some works are not limited to global descriptors, but adopt a two-stage method to first use global descriptors to obtain the top-k candidates, and then use local descriptors to re-ranking [20][21][22][23]. Patch-NetVLAD [13] combines the advantages of local and global descriptor methods by using NetVLAD [3] residuals to obtain patch-level features, which can effectively deal with the impact of environment and viewpoint changes on VPR. TransVPR [14] uses Transformer to automatically find task-relevant features in the image. The attentions extracted by Transformer's shallow, middle and deep layers respectively represent image features at different semantic levels. And due to Transformer's self-attention mechanism, these attentions automatically focus on the important region of image. SAMLoc [24] proposes a self-supervised hierarchical localization framework, based on multi-task distillation, by using structure-aware constraints for long-term visual localization. Other two-stage methods from related tasks, e.g. SuperGlue [21] and DELG [22], are also evaluated for VPR. However, they show a much slower inference speed. In general, the re-ranking of current VPR methods mostly relies on time-consuming RANSAC [25].

III. Method

The overall framework of our proposed model is shown in Fig. 1. The whole process is divided into four parts: triplet generation, feature extraction, global retrieval and re-ranking. Firstly, the query images and database images in the dataset are mined by triples to form several image triples. Each triplet consists of one query corresponding to a positive sample and a negative sample. These image triples are divided into small batches and then sent to the unified feature extraction network for feature extraction, and the global feature and local feature are obtained. The global features of the images are optimized by the SCE module for global retrieval to obtain top-100 images. The local features pass through the Mixer-Attention Layer and are then sent to the NCGM module. At last, the top-100 images are re-ranked to obtain the final retrieval results.

A. Spatial-Channel Embedding (SCE)

This section introduces the architecture of our proposed Spatial-Channel Embedding module, as shown in Fig. 2. Inspired by MLP-Mixer [26], auming that the input of the model is $X \in \mathbb{R}^{n\times d}$, where n is the number of feature descriptors, d is the dimension of feature descriptors, and Y is the output of the module. The model can be represented as follows:

$$Z = \delta(XP), \quad Z_1 = S(Z), \quad Y = Z_1 Q \tag{1}$$

where δ represents the GELU activation function, P and Q represent the linear projection along the feature channel size, which is also a learnable matrix. Z_1 indicates the output of SCE, and $S(.)$ indicates the mapping of SCE. In order to better realize the feature information interaction of spatial dimensions, for $S(.)$, we define a linear map:

$$f_{W,b}(Z) = WZ + b \tag{2}$$

where $W \in \mathbb{R}^{n\times d}$ represents the mapping parameter, W changes as Z changes, further we can get the following expression:

$$Z_1 = S(Z) = Z \odot f_{W,b}(Z) = Z \odot (WZ + b) \tag{3}$$

where $\odot$ represents element-wise multiplication, we define W and b to be close to 0 and 1, respectively, when the network is initialized. At this time, the value of A is approximately 1, can be expressed as $f_{W,b} \approx 1$.

The purpose of designing SCE is to pay extra attention to the spatial and scale information of global features, because these parts of the information are more discriminative. As shown in Fig. 2, the SCE module performs normalization on the channel and spatial dimensions of the features. This process yields descriptors C_c and C_s, respectively, enabling the network to learn scale, space, and other information. In Spatial branch we use the convolution layer and L2 normalization, and in Channel-branch we use the full connected layer. The resulting feature descriptor inherits the advantages of spatial dimension and channel dimension. Through parameter learning and normalization of different dimensions, the spatial and scale information of global features can be paid more attention and learned, and the robustness of global features can be improved.

Supposing that the feature map $G \in R^{H\times W}$, the final descriptor can be expressed as $K = \{k^c\}$, $c = 1, 2, ...C$, then

$$k^c = \langle f^c, G \rangle_F \tag{4}$$

where $f^c \in R^{H\times W}$ represents the feature map of G at the c-th channel, $\langle a, b \rangle$ indicates the Frobenius inner product and k^c represents the feature activation of K at the c-th channel.

We design the Spatial-Channel Embedding, which differs from the conventional L2 normalization by encoding features from both the channel and spatial branches. The spatial and scale information within the feature is enhanced, resulting in the acquisition of a globally discriminative descriptor.

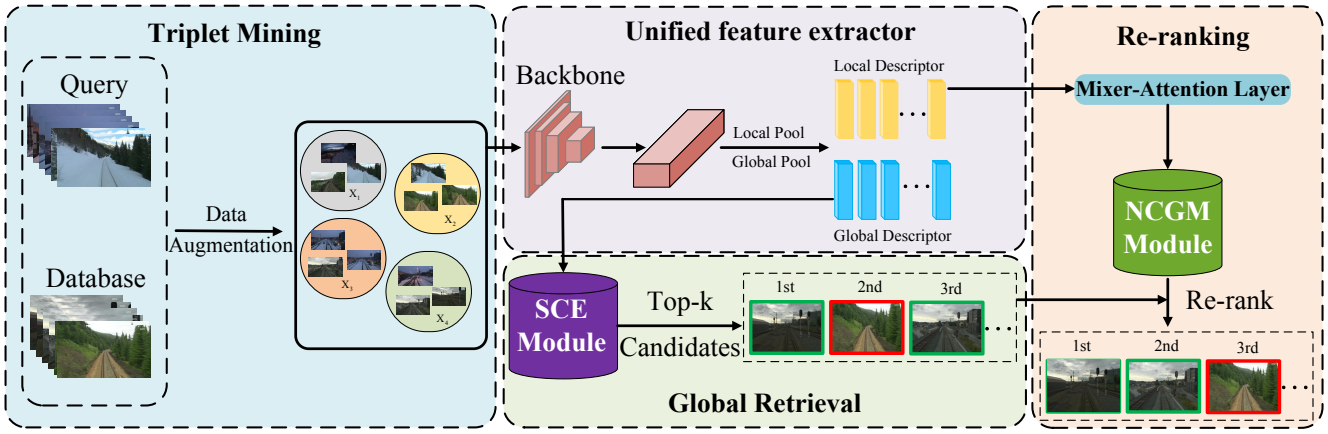

Fig. 1. **An overview of the proposed framework.** It is mainly divided into two steps: global retrieval and re-ranking. After global retrieval, top-100 images are obtained from the database, and then they are re-ranked to get the final retrieval results.

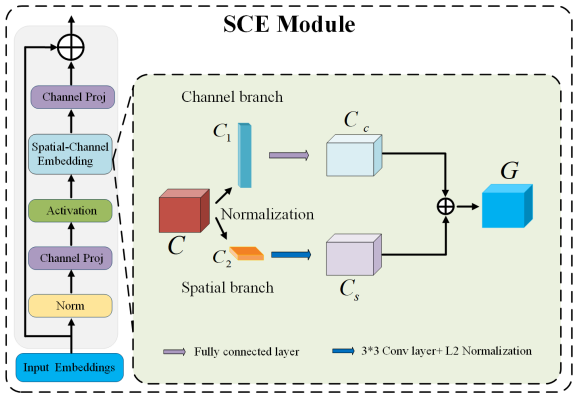

Fig. 2. **The proposed Spatial-Channel Embedding Module.** By optimizing the spatial and channel dimensions of the feature to make the global descriptor more discriminative.

B. Neighborhood Consensus Guided Matching (NCGM)

In this part, we introduce the architecture of the Neighborhood Consensus Guided Matching module. As shown in Fig. 3, the main function of this module is to achieve accurate matching. Firstly, the aggregated feature descriptors undergo spatial positional encoding and adaptive average pooling. Subsequently, they are transformed into independent feature representations through flattening. The output features are input into the Multi-Scale Attention Module and subsequently matched at the patch level using the Transformer Layer. In the patch-level matching process, it is inevitable that some incorrect matches will occur. We use the Neighborhood Consensus to filter the features to get more accurate matching. Before the NCGM, there is the Mixer-Attention Layer, which comprises four components: two fully-connected layers, a GELU nonlinearity, and an Attention Module.

We incorporate the concept of clustering into the Attention Module of the Mixer-Attention Layer, where the attention score is used to weight the residual of the descriptor relative to the cluster center, thereby improving the discriminative capability of the resulting aggregated feature descriptor.

Assuming that the dimension of the feature map before entering the Attention Module is $H \times W \times D$. The proposed Attention Module consists of a 1×1 convolution module layer and a softplus activation function. The convolutional layer produces an attention map of size $H \times W$, which serves as the weight $\{w_i\}$ for each image descriptor $\{x_i\}$. After clustering, these descriptors are assigned the corresponding weights derived from the attention map. The relationship can be expressed as follows:

$$V(j,k) = \sum_{i=1}^{N} w_i a_k(x_i)(x_i(j) - c_k(j)) \tag{5}$$

The module first aggregates the descriptors $\{x_i\}$ mapped by the GELU module into K clusters $\{c_k\}$, and then calculates the residual $(x_i(j) - c_k(j))$. And we use this to assign the weight a_k of each descriptor x_i relative to the cluster c_k.

The Multi-Scale Attention Module is designed to integrate information from features of different scales. To achieve this effectively, the Multi-Scale Attention Module utilizes three convolutional kernels with varying sizes: $128 \times 3 \times 3$, $128 \times 5 \times 5$ and $128 \times 7 \times 7$, respectively. Assuming that the flattened features are expressed as $X \in R^{S \times C}$, where S represents the number of feature vectors and C indicates the dimensionality of each feature vector. In the Transformer Layer, matrix transposition and information fusion of the feature spaces is realized by sending each row of the matrix to *MLP*. Then the matrix generated in the previous step is transposed again and then sent to another *MLP* row by row to achieve information fusion of feature channels. Layer Normalization and Skip-connection are used to further optimize the generated features. The input-output relationship of the Transformer Layer can be expressed as:

$$U_{*,i} = X_{*,i} + W_2 \sigma(W_1 LayerNorm(X)_{*,i}) \tag{6}$$

$$Y_{j,*} = U_{j,*} + W_4 \sigma(W_3 LayerNorm(U)_{j,*}) \tag{7}$$

where i=1,2,...,C; j=1,2,...,S. σ is an element-wise nonlinearity. W_1, W_2, W_3 and W_4 represent the learnable weight parameters.

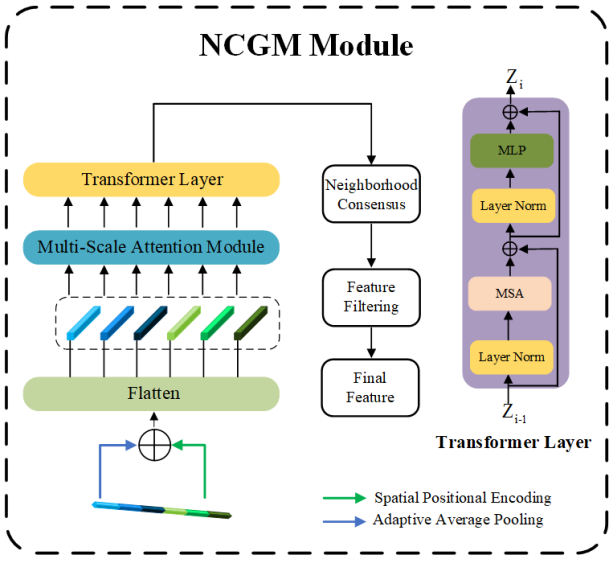

Fig. 3. **The proposed Neighborhood Consensus Guided Matching Module.** The Neighborhood Consensus is employed to filter the features, which have been previously optimized by the Multi-Scale Attention Module and Transformer Layer, in order to achieve more accurate matching.

TABLE I
SUMMARY OF THE DATASETS. "+" MEANS THAT THE DATASET CONTAINS SUCH CHANGES, WHILE "-" IS THE OPPOSITE.

Dataset	Environment			Variation				
	Urban	Suburban	Natural	Viewpoint	Day/night	Weather	Seasonal	Dynamic
MSLS [27]	✓	✓	✓	+	+	+	+	+
Pitts30k [28]	✓			+	−	−	−	+
Tokyo 24/7 [29]	✓			+	+	−	−	+
RobotCar-S2 [30][31]	✓			+	+	+	+	+
Extended-CS [32][33]	✓	✓	✓	+	+	+	+	+

IV. EXPERIMENT

A. Datasets and Evaluation Metrics

We evaluate all methods on several public place recognition benchmark datasets: MSLS [27], Pitts30k [28], Tokyo 24/7 [29], RobotCar Seasons v2 (RobotCar-S2) [30][31] and Extended CMU Seasons (Extended-CS) [32][33]. Tab. I summarizes the qualitative nature of them.

We follow TransVPR [14], that is, for MSLS, Pitts30k and Tokyo 24/7 datasets, we use the widely used Recall@N evaluation metric. For these datasets, we define that a query image is correctly localized if at least one of the top N reference images is within the standard ground truth tolerance, i,e. 25m translational and 40° orientation error for MSLS, 25m translational error for Pitts30k and Tokyo 24/7.

For the RobotCar Seasons v2 and Extended CMU Seasons datasets, instead of calculating the exact 6-DOF pose of the query image, we directly use the pose of the best matching reference image. We use the default translation and rotation thresholds given in the datasets and get the corresponding scores by weighted average.

B. Implementation Details

Model Settings. Our model is implemented in PyTorch framework. We use CCT [34] backbone pre-trained on ImageNet [35] to extract features, and then the obtained features are sent to the SCE module to achieve global retrieval. The proposed Neighborhood Consensus Guided Matching module is designed to achieve more accurate matching. For the fairness of the comparison, we follow the model settings and parameter configurations specified by the author in the original paper.

Training. We train our model on two datasets: Pitts30k for urban imagery (Pitts30k and Tokyo 24/7 datasets), and MSLS for all other conditions. All images are resized to 640 by 480 pixels before training. Adam optimizer is adopted to optimize our model. For the global retrieval, the momentum is set to 0.9 and the weight decay is set to 0.001. The loss function is triplet loss [36]. The initial learning rate is set to 0.05 and multiplied by 0.3 every 5 epochs. For the re-ranking, the Multi-Scale Attention Module and Transformer Layer are initialized by training for 5 epoches on MSLS training set with 0.0005 initial learning rate. The training process is stopped if the performance no longer improves within 3 epochs.

C. Comparisons with State-of-the-Art Methods

Visual Place Recognition Benchmarks. We compare our method with several state-of-the-art algorithms on place recognition datasets, including methods that use only global retrieval: NetVLAD [3], SFRS [17] and GCL [19]; and methods that use both global retrieval and re-ranking: SP-SuperGlue [7][21], DELG [22], Patch-NetVLAD [13], and TransVPR [14]. For SP-SuperGlue, we firstly use NetVLAD to filter candidate images by global retrieval, then extract SuperPoint patch descriptors and finally SuperGlue is applied to identify matches and to re-rank the candidate images. For all two-stage methods, the first step is to conduct global retrieval to obtain top-100 images, followed by re-ranking.

Tab. II shows the quantitative results of our method compared with state-of-the-art methods on benchmark datasets. When employing both global retrieval and re-ranking, the results indicate that our method outperforms all other methods on the majority of benchmark datasets. It is worth noting that our method also achieves competitive results using only global retrieval. Our method outperforms GCL on the MSLS validation, MSLS challenge, Pitts30k test, and Tokyo 24/7 datasets, achieving an absolute increase on Recall@1 of 6.8%, 3.2%, 10.3% and 38.1% respectively. On the challenging RobotCar Seasons v2 and Extended CMU Season datasets, it can be seen that our method greatly surpasses all algorithms on both global retrieval and re-ranking.

Latency and Memory. In the real-world, VPR systems must consider time consumption and memory usage as important factors. Tab. III shows the feature extraction latency, matching latency, and memory usage required by various methods for processing a single query image. We conducted these experiments on the MSLS val dataset. In terms of

TABLE II

COMPARISON WITH STATE-OF-THE-ART METHODS ON SIX BENCHMARK DATASETS.

Method	MSLS val			MSLS challenge			Pitts30k test			Tokyo 24/7			RobotCar Seasons v2			Extended CMU Seasons		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	.25m/2°	.5m/5°	5.0m/10°	.25m/2°	.5m/5°	5.0m/10°
NetVLAD [3]	53.1	66.5	71.1	35.1	47.4	51.7	81.9	91.2	93.7	64.4	78.4	81.6	5.6	20.7	71.8	5.9	18.0	76.9
SFRS [17]	69.2	80.3	83.1	41.5	52.0	56.3	**89.4**	**94.7**	**95.9**	**85.4**	91.1	93.3	8.0	27.3	80.4	6.7	21.9	92.6
GCL [19]	80.9	90.7	92.6	**62.3**	**76.2**	**81.1**	79.2	90.4	93.2	58.1	74.3	78.1	4.7	21.0	74.7	6.1	18.2	74.9
Ours(w/o re-ranking)	**83.5**	**91.0**	**92.8**	61.3	72.9	76.4	86.1	93.6	95.5	80.7	**91.4**	**94.0**	**9.1**	**32.9**	**86.7**	**8.9**	**27.6**	**95.5**
SP-SuperGlue [7][21]	78.1	81.9	84.3	50.6	56.9	58.3	87.2	94.8	96.4	88.2	90.2	90.2	9.5	35.4	85.4	9.5	30.7	96.7
DELG [22]	83.2	90.0	91.1	52.2	61.9	65.4	**89.9**	95.4	96.7	95.9	96.8	97.1	2.2	8.4	76.8	5.7	21.1	93.6
Patch-NetVLAD [13]	79.5	86.2	87.7	48.1	57.6	60.5	88.7	94.5	95.9	86.0	88.6	90.5	9.6	35.3	90.9	11.8	36.2	96.2
TransVPR [14]	86.8	91.2	92.4	63.9	74.0	77.5	89.0	94.9	96.2	-	-	-	9.8	34.7	80.0	-	-	-
Ours	**87.7**	**93.0**	**95.4**	**65.5**	**77.3**	**81.0**	89.5	**95.7**	**96.8**	**96.2**	**97.1**	**97.5**	**12.1**	**42.9**	**91.3**	**14.7**	**43.8**	**98.4**

TABLE III

FEATURE EXTRACTION TIME, DESCRIPTOR MATCHING TIME, AND MEMORY USAGE OF ALL METHODS. THE FOLLOWING DATA IS MEASURED ON NVIDIA GEFORCE RTX 3090 GPU AND INTEL XEON GOLD 6148 CPU.

Method	Extraction latency (ms)	Matching latency (s)	Memory (MB)
SP-SuperGlue [7][21]	163	7.6	1.93
DELG [22]	192	35.3	0.9
Patch-NetVLAD [13]	1330	7.5	44.14
TransVPR [14]	45	3.2	1.17
Ours	20	2.1	0.95

TABLE IV

ABLATION STUDY ON MODEL COMPONENTS.

SCE	MAL	NCGM	Tokyo24/7		
			R@1	R@5	R@10
×	×	×	70.2	75.9	80.6
×	×	✓	74.2	79.5	84.2
×	✓	×	72.6	77.5	83.6
×	✓	✓	77.9	84.1	87.1
✓	×	×	80.7	91.4	94.0
✓	×	✓	90.4	92.6	94.6
✓	✓	×	88.4	91.7	93.3
✓	✓	✓	**96.2**	**97.1**	**97.5**

the feature extraction, our method is 2.25 times faster than TransVPR and 66.5 times faster than Patch-NetVLAD. Our method is 1.5 times and 16.8 times faster than TransVPR and DELG in terms of the matching latency. As for memory footprint, ours is slightly inferior to DELG. Overall, our method achieves a better balance of efficiency and memory, making it more practical for practical scenarios.

D. Ablation Studies

Components Ablation. To investigate the effectiveness of our proposed SCE module, Mixer-Attention Layer (MAL) and NCGM module, we conduct ablation studies on the Tokyo 24/7 dataset. The experimental results are shown in Tab. IV. It can be seen that only when SCE module, Mixer-Attention Layer and NCGM module are applied at the same time can our method achieve the best performance.

TABLE V

COMPARISON WITH DIFFERENT RE-RANKING METHODS (TOP- 100 CANDIDATES RERANKED) ON THE TOKYO 24/7 DATASET.

Method	R@1	R@5	R@10
No Reranking	80.7	91.4	94.0
RANSAC [25]	84.2	93.5	95.2
SuperGlue [21]	86.8	95.7	96.1
RRT [37]	82.9	91.9	93.5
LoFTR [38]	85.9	92.1	95.7
CVNet [39]	75.6	87.1	91.2
Ours	**96.2**	**97.1**	**97.5**

And it also demonstrates that each of the proposed modules makes a contribution to the improvement of the network's performance.

Re-ranking Methods. Additionally, we investigate the performance of different re-ranking methods. We compare our re-ranking method with five re-ranking methods, including RANSAC [25], SuperGlue [21], RRT [37], LoFTR [38] and CVNet [39]. The global retrieval phase uses our backbone network and the re-ranking uses the respective methods. The experimental results are shown in Tab. V. As can be seen from the results, our re-ranking method outperforms all other methods.

V. CONCLUSION

In this work, we propose a novel approach of feature aggregation and implement a more efficient re-ranking method. Our proposed Spatial-Channel Embedding (SCE) module, which normalizes the channel and spatial dimensions of features to stimulate better descriptor learning. The spatial and scale information in the feature is enhanced and the global descriptor with discrimination is obtained. In addition, Neighborhood Consensus Guided Matching (NCGM) module is proposed to achieve more accurate matching while needing less time and storage requirements. We conduct extensive experiments and ablation studies on place recognition benchmarks. The results show that the proposed method not only outperforms several state-of-the-art methods on several benchmark datasets, but also performs well in terms of time and memory consumption.

REFERENCES

[1] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *ieee transactions on robotics*, vol. 32, no. 1, pp. 1–19, 2015.

[2] Z. Chen, A. Jacobson, N. Sünderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford, "Deep learning features at scale for visual place recognition," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3223–3230.

[3] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[4] A. Gordo, J. Almazan, J. Revaud, and D. Larlus, "End-to-end learning of deep visual representations for image retrieval," *International Journal of Computer Vision*, vol. 124, no. 2, pp. 237–254, 2017.

[5] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 9, pp. 1704–1716, 2011.

[6] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning cnn image retrieval with no human annotation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 7, pp. 1655–1668, 2018.

[7] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 224–236.

[8] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler, "D2-net: A trainable cnn for joint description and detection of local features," in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 2019, pp. 8092–8101.

[9] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.

[10] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[11] F. Lateef and Y. Ruichek, "Survey on semantic segmentation using deep learning techniques," *Neurocomputing*, vol. 338, pp. 321–348, 2019.

[12] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of convnet features for place recognition," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015, pp. 4297–4304.

[13] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 141–14 152.

[14] R. Wang, Y. Shen, W. Zuo, S. Zhou, and N. Zheng, "Transvpr: Transformer-based place recognition with multi-level attention aggregation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 648–13 657.

[15] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, IEEE International Conference on*, vol. 3. IEEE Computer Society, 2003, pp. 1470–1470.

[16] H. Jin Kim, E. Dunn, and J.-M. Frahm, "Learned contextual feature reweighting for image geo-localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2136–2145.

[17] Y. Ge, H. Wang, F. Zhu, R. Zhao, and H. Li, "Self-supervising fine-grained region similarities for large-scale image localization," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*. Springer, 2020, pp. 369–386.

[18] M. Venator, Y. El Himer, S. Aklanoglu, E. Bruns, and A. Maier, "Self-supervised learning of domain-invariant local features for robust visual localization under challenging conditions," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2753–2760, 2021.

[19] M. Leyva-Vallina, N. Strisciuglio, and N. Petkov, "Data-efficient large scale place recognition with graded similarity supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 23 487–23 496.

[20] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 716–12 725.

[21] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4938–4947.

[22] B. Cao, A. Araujo, and J. Sim, "Unifying deep local and global features for image search," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*. Springer, 2020, pp. 726–743.

[23] C. Wang, R. Xu, S. Xu, W. Meng, and X. Zhang, "Cndesc: Cross normalization for local descriptors learning," *IEEE Transactions on Multimedia*, 2022.

[24] J. Ning, Y. Zhang, X. Zhao, S. Coleman, K. Li, and D. Kerr, "Samloc: Structure-aware constraints with multi-task distillation for long-term visual localization," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 719–11 725.

[25] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[26] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit *et al.*, "Mlp-mixer: An all-mlp architecture for vision," *Advances in neural information processing systems*, vol. 34, pp. 24 261–24 272, 2021.

[27] F. Warburg, S. Hauberg, M. Lopez-Antequera, P. Gargallo, Y. Kuang, and J. Civera, "Mapillary street-level sequences: A dataset for lifelong place recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2626–2635.

[28] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual place recognition with repetitive structures," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 883–890.

[29] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla, "24/7 place recognition by view synthesis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1808–1817.

[30] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[31] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic *et al.*, "Benchmarking 6dof outdoor visual localization in changing conditions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8601–8610.

[32] H. Badino, D. Huber, and T. Kanade, "Visual topometric localization," in *2011 IEEE Intelligent vehicles symposium (IV)*. IEEE, 2011, pp. 794–799.

[33] C. Toft, W. Maddern, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, T. Pajdla *et al.*, "Long-term visual localization revisited," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 4, pp. 2074–2088, 2020.

[34] A. Hassani, S. Walton, N. Shah, A. Abuduweili, J. Li, and H. Shi, "Escaping the big data paradigm with compact transformers," *arXiv preprint arXiv:2104.05704*, 2021.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[36] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823.

[37] F. Tan, J. Yuan, and V. Ordonez, "Instance-level image retrieval using reranking transformers," in *proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 105–12 115.

[38] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "Loftr: Detector-free local feature matching with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8922–8931.

[39] S. Lee, H. Seong, S. Lee, and E. Kim, "Correlation verification for image retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5374–5384.