CricaVPR: Cross-image Correlation-aware Representation Learning for Visual Place Recognition

Supplementary Material

A. Overview

This supplementary material provides the following additional content about experimental results and analysis:

B. Visualizations of Place Features Using t-SNE

C. Tunable Parameters

D. Additional Results on Challenging Datasets

E. Additional Ablations on Cross-image Encoder

F. Effects of Batch Size

G. Additional Ablations on MulConvAdapter

H. Effects of Adaptation on the Used SPM Feature

I. Comparison to Other Methods with the Same Training Dataset

J. Datasets Details

K. Compared Methods Details

L. Additional Qualitative Results and Failure Cases

M. Limitations

Note that the experiments in this supplementary material are conducted as in the main paper. That is, PCA is used to reduce the descriptor dimensionality to 4096-dim when comparing our method with other methods. However, it is not used by default in ablation experiments.

B. Visualizations of Place Features using t-SNE

In this section, we use the t-SNE [17] method to map our place features to 2-dimensional space and visualize their distribution. We employ pre-trained DINOv2, adapted DINOv2 (with our MulConv adapter), and our entire network (with our MulConv adapter and cross-image encoder) to extract features of 432 images from 36 different places (12 images per place). There exist variations in viewpoints and conditions among the 12 images of the same place. Suppl. Fig. 1 illustrates the visualization results. It can be observed that some features of different places, which are extracted by pre-trained DINOv2, are not well separated. This demonstrates the limited discriminability of place features extracted by pre-trained DINOv2. However, after performing our adaptation, the adapted DINOv2 successfully distinguishes most of the places, while a few places are still not well distinguished. By applying both our adaptation and the cross-image encoder, our proposed model effectively clusters image features of the same place and separates features of different places, i.e., pulling the features of the same place closer together and pushing the features of different places farther apart. This clearly demonstrates the efficacy of our approach in addressing the challenge of perceptual aliasing.

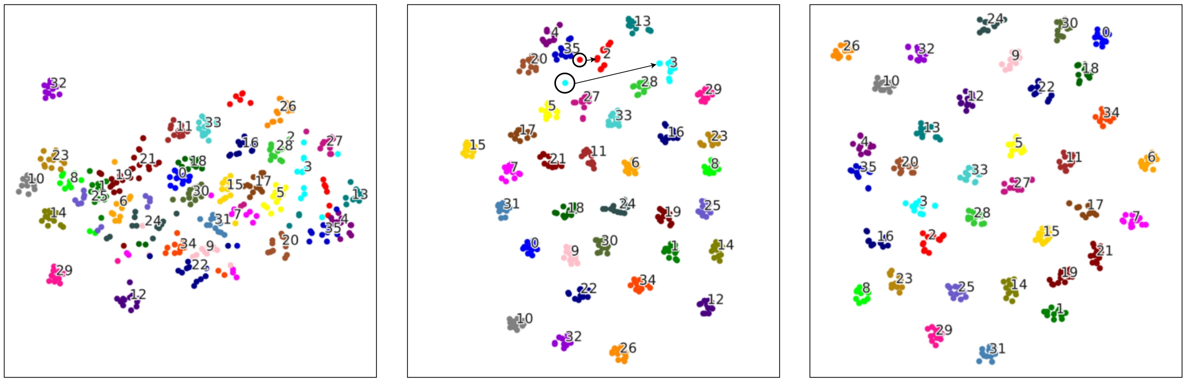
Method	Total	Backbone	Adapter	Others	**Tunable**
CosPlace-V	-	14.7	0	0.3	7.3+145.7
CosPlace-R	-	23.5	0	4.2	26.3+582.8
FullTuning	97.6	86.6	0	11.0	97.6
Ours	106.8	86.6+9.2	9.2	11.0	20.2

Supplementary Table 1. The number of parameters (M) in models. We mainly focus on tunable parameters (the last column). “CosPlace-V” and “CosPlace-R” represent the CosPlace methods using VGG16 and ResNet50 to produce 512-dim and 2048-dim features, respectively. Taking CosPlace-V as an example, since it adds multiple classifiers (for multiple groups of training data) after the model during training, the tunable parameters contain the parameters of the trainable part in the model (7.3M) and all classifiers (145.7M). “FullTuning” represents full fine-tuning of the DINOv2 backbone (including our cross-image encoder) without the adapter. The “Others” in the table are the aggregation module for CosPlace, the cross-image encoder for FullTuning and Ours (the parameters of GeM pooling are so few that they can be ignored).

It is worth mentioning that this visualization method commonly used in classification tasks has rarely been used in previous VPR works. We can use it thanks to the recently proposed GSV-Cities dataset [1] (and the SF-XL dataset [4]) that split place images into a finite number of categories.

C. Tunable Parameters

We provide detailed model parameters as shown in Suppl. Table 1 (using CosPlace as baseline). Since we use an adapter-based parameter-efficient fine-tuning method to train our model, the tunable part of our model only contains the adapter inserted into the backbone and the cross-image encoder after the backbone. The number of tunable parameters of our model is 20.2M, which is only about 1/5 of the full fine-tuning DINOv2 (with the cross-image encoder). This is also less than that of CosPlace using ResNet50 to produce 2048-dim features (including 26.3M tunable parameters in the model and 582.8M tunable parameters of classifiers).



(a) Result of pre-trained DINOv2 **(b) Result of adapted DINOv2** **(c) Result of our model**

Supplementary Figure 1. **Visualizations of place features in 2-dimensional space using t-SNE.** We use the features of 432 images from 36 different places (i.e. 36 categories) for visualization. (a), (b), and (c) are the results of pre-trained DINOv2, adapted DINOv2 (using our MulConv adapter), and our complete model (using MulConv adapter and cross-image encoder), respectively. Note that the positions of two points in (b) are improper, that is, the corresponding feature representation will suffer perceptual aliasing.

Method	Nordland			AmsterTime			SVOX-Night			SVOX-Rain			SVOX-Sun		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
SFRS [7]	16.0	24.1	28.7	29.7	48.5	55.6	28.6	40.6	46.4	69.7	81.5	84.6	54.8	68.3	74.1
CosPlace [4]	58.5	73.7	79.4	38.7	61.3	67.3	44.8	63.5	70.0	85.2	91.7	93.8	67.3	79.2	83.8
MixVPR [2]	76.2	86.9	90.3	40.2	59.1	64.6	64.4	79.2	83.1	91.5	97.2	98.1	84.8	93.2	94.7
EigenPlaces [5]	71.2	83.8	88.1	48.9	69.5	76.0	58.9	76.9	82.6	90.0	96.4	98.0	86.4	95.0	96.4
CricaVPR (ours)	**90.7**	**96.3**	**97.6**	**64.7**	**82.8**	**87.5**	**85.1**	**95.0**	**96.7**	**95.0**	**98.2**	**98.7**	**93.7**	**98.4**	**98.6**

Supplementary Table 2. Comparison to SOTA methods on challenging datasets. The best is highlighted in **bold** and the second is underlined. We employ PCA to reduce the descriptor dimension of our method to 4096-dim.

D. Additional Results on Challenging Datasets

The main paper has presented the R@1 results of our method compared to state-of-the-art (SOTA) methods on three challenging datasets, i.e., Nordland, AmsterTime, and SVOX (SVOX-Night, SVOX-Rain). Here, we provide the complete R@1/R@5/R@10 results as shown in Suppl. Table 2, complementing another challenging query subset (SVOX-Sun) of the SVOX dataset. Before our method was proposed, MixVPR and EigenPlaces had their own advantages on these challenging datasets, and no method completely outperformed the other methods. However, our proposed CricaVPR achieves better performance compared to all previous methods on these datasets, particularly outperforming other methods by a large margin on Nordland, AmsterTime, and SVOX-Night, which are quite difficult.

Moreover, we also provide the results of our CricaVPR on Pitts250k (97.5% R@1) in Section I.

E. Additional Ablations on Cross-image Encoder

In the main paper, we have combined the proposed cross-image correlation awareness implemented by our cross-image encoder with three different global representations to demonstrate its effectiveness. In this section, we further compare the performance of constructing the cross-image encoder using different numbers of transformer encoder layers, and the results are shown in Suppl. Table 3. Compared to not using the cross-image encoder (No encoder), incorporating the cross-image encoder constructed with any number of transformer encoder layers leads to significant performance improvements. However, when only one transformer encoder layer is used, there is still a noticeable performance gap compared to using multiple transformer encoder layers (on Pitts30k and Tokyo24/7), indicating that a single transformer encoder layer alone cannot sufficiently correlate images within a batch. The best performance is achieved when using two transformer encoder layers, which is the recommended configuration.

F. Effects of Batch Size

Since our method correlates all images within a batch and utilizes the cross-image variations (including images from the same place and images from different places) as a cue to guide the representation learning in VPR, the training batch size is also a factor that may have an impact on per-

Cross-image encoder	Pitts30k			Tokyo24/7			MSLS-val		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
No encoder	90.6	95.9	97.2	85.1	93.3	95.6	85.5	93.2	94.3
Transformer encoder layer ×1	92.9	96.6	97.5	92.7	95.2	96.5	89.6	**95.7**	**96.4**
Transformer encoder layer ×2	**94.8**	**97.4**	**98.1**	**93.0**	**97.1**	**97.8**	**89.9**	95.4	96.2
Transformer encoder layer ×3	94.5	**97.4**	**98.1**	**93.0**	96.2	**97.8**	88.8	94.7	96.1

Supplementary Table 3. The results of constructing the cross-image encoder using different numbers of transformer encoder layers.

Batch Size	with cross-	Pitts30k			Tokyo24/7			MSLS-val		
(Number of Places)	image encoder	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
$NP = 16$	×	89.5	95.3	96.8	75.6	89.2	91.4	80.9	90.4	92.7
$NP = 32$	×	89.9	95.2	96.7	81.3	91.1	93.0	83.0	93.1	93.8
$NP = 64$	×	90.7	95.9	97.5	84.4	94.3	96.5	84.1	92.3	94.2
$NP = 72$	×	90.6	95.9	97.2	85.1	93.3	95.6	85.5	93.2	94.3
$NP = 16$	✓	94.6	97.0	97.7	87.9	94.9	96.2	84.1	92.6	94.2
$NP = 32$	✓	94.8	97.4	98.0	91.1	94.9	96.8	85.0	93.1	95.1
$NP = 64$	✓	**94.9**	**97.5**	**98.1**	92.4	95.6	97.1	88.1	95.0	95.1
$NP = 72$	✓	94.8	97.4	**98.1**	**93.0**	**97.1**	**97.8**	**89.9**	**95.4**	**96.2**

Supplementary Table 4. Results of different training batch sizes, i.e., different numbers of places (4 images per place). NP is the abbreviation of "Number of Places". We provide the results with or without the cross-image encoder.

formance. The training dataset GSV-Cities [1] provides 4 images per place by default, and we use different batch sizes, i.e., one batch contains different numbers of places, to train our models. It should be noted that we use the multi-similarity (MS) loss to train the models (same as MixVPR), which inherently leads to a result that a larger batch size is more conducive to providing hard sample pairs to train a robust model. Therefore, we also provide the results obtained at different batch sizes without using the cross-image encoder as a reference. The results are shown in Suppl. Table 4. Regardless of whether the cross-image encoder is used, the performance degradation caused by the smaller batch size is not obvious on Pitts30k, but is significant on more difficult Tokyo24/7 and MSLS. When using the cross-image encoder, the absolute R@1 drops caused by using the smallest batch size ($NP = 16$) compared to the largest batch size ($NP = 72$) on Pitts30k, Tokyo24/7, and MSLS-val are 0.2%, 5.1%, and 5.8% respectively. When the cross-image encoder is not used, the absolute R@1 drops caused by that are 1.1%, 9.5%, and 4.6% respectively. This indicates that: 1) More challenging (test) datasets require larger batch size to train a more robust model. 2) Our proposed cross-image encoder, to some extent, reduces the demand for a larger batch size when using the MS loss for training.

Batch	Pitts30k			Tokyo24/7			MSLS-val		
Size	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
1	91.6	95.7	96.9	89.5	94.6	96.2	88.5	95.1	95.7
4	93.9	97.2	97.7	87.3	93.7	94.6	88.0	**95.5**	**96.5**
8	**94.8**	**97.4**	**98.1**	91.7	96.2	97.5	89.1	95.1	95.9
16	93.7	97.0	**98.1**	**93.0**	**97.1**	**97.8**	**89.9**	95.4	96.2
32	93.0	96.9	97.9	92.7	96.2	97.5	88.9	**95.5**	96.2

Supplementary Table 5. Results of different inference batch size.

In addition, we also conduct experiments to study the impact of different batch sizes during inference, i.e., inference batch size. The results are as shown in Suppl. Table 5. Since our method learns cross-image correlation-aware representation during training, setting the batch size to 1 during testing makes our cross-image encoder ineffective, further leading to the gap between training and testing, i.e., performance in this case will be reduced. Besides, an inference batch size that is too small (e.g., 4) will lead to unstable results (even worse than when it equals 1). Although the inference batch size that achieves the best performance on different datasets does not appear to be fixed (too small or too large will reduce performance), setting it to 16 can achieve excellent results on all datasets. So we set it to 16 (except on Pitts30k/Pitts250k we set it to 8 for better results).

G. Additional Ablations on MulConvAdapter

We have verified the effectiveness of the proposed multi-scale convolution adapter (MulConvAdapter) by comparing it with the vanilla adapter and ConvAdapter (i.e., Convpass [9]). To further demonstrate the advantages of MulConvA-

Conv Size	Pitts30k			Tokyo24/7			MSLS-val		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
1×1	94.5	97.2	97.8	91.7	95.9	97.1	88.2	95.3	95.5
3×3	94.3	97.1	97.9	91.7	95.2	96.8	87.6	94.3	95.8
5×5	94.7	97.3	97.8	90.2	94.6	96.8	87.6	95.1	**96.4**
MulConv	**94.8**	**97.4**	**98.1**	**93.0**	**97.1**	**97.8**	**89.9**	**95.4**	96.2

Supplementary Table 6. The results of convolution-based adapters. "MulConv" is our MulConvAdapter. Note that the single convolution kernel adapter here has one more skip connection than the 3×3 convolution adapter (ConvAdapter) in the main paper.

Ablated versions	Pitts30k		Tokyo24/7		MSLS-val	
	R@1	R@5	R@1	R@5	R@1	R@5
FrozenDINOv2-GeM	79.2	90.1	65.4	83.8	40.8	51.5
FrozenDINOv2-SPM	74.8	90.1	49.8	67.0	45.4	60.7
Adapt-GeM	87.1	94.0	70.2	85.4	78.4	87.8
Adapt-SPM	90.6	95.9	85.1	93.3	85.5	93.2

Supplementary Table 7. The results of the GeM and SPM representation using a frozen DINOv2 or adapted DINOv2 backbone. All results here have been provided in Table 4 and Table 5 of our main paper.

dapter over adapters using only a single-size convolution kernel, we compare MulConvAdapter with three adapter variants employing three different convolution kernel sizes (1×1, 3×3, and 5×5). To be fair, the three adapters based on a single convolution kernel use skip connection like our MulConvAdapter (the ConvAdapter in the main paper does not), that is, our MulConvAdapter differs from these three adapters only in the convolution kernel. The results are presented in Suppl. Table 6. Except for our MulConvAdapter, the adapters based on 1×1, 3×3, and 5×5 convolution kernels have advantages in different datasets (and metrics), indicating that it is difficult for an adapter with a single-size convolution kernel to perform well for all place images on the VPR task. In contrast, our MulConvAdapter integrates these three convolution kernels to consistently provide proper local information, thus achieving the best performance.

H. Effects of Adaptation on the Used SPM Feature

In our method, we mainly use the spatial pyramid model (SPM) representation that combines the class token and the GeM feature. An interesting phenomenon is that when using the frozen DINOv2 as the backbone, the SPM feature (FrozenDINOv2-SPM) performs worse than GeM (FrozenDINOv2-GeM) on Pitts30k and Tokyo24/7 (see Suppl. Table 7). However, after using our adaptation, Adapt-SPM performs much better than Adapt-GeM. This shows that our adaptation makes the combined class token and GeM features in the SPM representation more compatible.

Method	Training set	Pitts250k			MSLS-val		
		R@1	R@5	R@10	R@1	R@5	R@10
CosPlace‡	SF-XL	92.3	97.4	98.4	87.4	94.1	94.9
NetVLAD†	GSV-Cities	90.5	96.2	97.4	82.6	89.6	92.0
CosPlace†	GSV-Cities	91.5	96.9	97.9	84.5	90.1	91.8
CricaVPR	GSV-Cities	**97.5**	**99.4**	**99.7**	**90.0**	**95.4**	**96.4**

Supplementary Table 8. The results of methods trained on GSV-Cities. The suffix †/‡ means that the method is different from the main paper on the backbone and/or training set. Since SF-XL is built for CosPlace (or it is part of CosPlace), CosPlace‡ trained on SF-XL is better than CosPlace† trained on GSV-Cities.

Method	Training set	Pitts30k			Pitts250k		
		R@1	R@5	R@10	R@1	R@5	R@10
SFRS	Pitts30k	89.4	94.7	95.9	90.7	96.4	97.6
MixVPR	GSV-Cities	91.5	95.5	96.3	94.1	98.2	98.9
EigenPlaces	SF-XL	92.5	96.8	97.6	94.1	97.9	98.7
CricaVPR*	Pitts30k	**93.0**	**96.9**	**97.9**	**95.9**	**99.0**	**99.5**

Supplementary Table 9. Results of CricaVPR* trained on Pitts30k.

I. Comparison to Other Methods with the Same Training Dataset

Most methods (except MixVPR) use different training datasets than our method. The GSV-Cities dataset used in our method has been shown to achieve better results than the datasets with weak supervision (e.g., Pitts30k and MSLS) [1]. Training different methods with the same dataset can promote fair comparisons. However, completely achieving it is hard as some methods are designed based on the characteristics of a certain (type of) dataset, and training on others may make some components of them meaningless. To minimize the impact of the training dataset on results, we use the results (reported in the MixVPR paper) of NetVLAD and CosPlace (both based on ResNet50) trained on GSV-Cities for a more fair comparison. The results are shown in Suppl. Table 8 and our method still significantly outperforms others. Note that in this section we have added the results on Pitts250k (larger but easier than Pitts30k). Besides, we also provide the results of training our model on the smallest/weakest Pitts30k dataset in Suppl. Table 9. Our model trained on Pitts30k still gets SOTA results (better than EigenPlaces trained on SF-XL and much better than SFRS also trained on Pitts30k).

J. Datasets Details

Pitts30k [15] is derived from Google Street View panoramas with GPS labels. It consists of images from 24 different viewpoints for each place in urban scenes, exhibiting significant viewpoint variations, moderate condition variations, and a small number of dynamic objects. Pitts30k is a subset of Pitts250k (but harder than Pitts250k for most methods). In our experiments, we mainly use the Pitts30k test set.

Tokyo24/7 [16] comprises a total of 75,984 database images and 315 query images from urban environments. The query images are selected from a pool of 1,125 images captured from 125 places, each involving 3 different viewpoints and 3 different times of the day. This dataset shows viewpoint variations and significant condition changes, particularly day-night changes.

MSLS (Mapillary Street-Level Sequences) [19] is a large-scale VPR dataset that encompasses more than 1.6 million images captured in urban, suburban, and natural environments across 30 cities spanning six continents. This dataset provides GPS coordinates and compass angles for each image, and shows various changes caused by illumination, weather, season, viewpoint, dynamic objects, and so on. It is divided into three sets: training, public validation (MSLS-val), and withheld test (MSLS-challenge). To ensure comprehensive evaluation, we assess the model on both the MSLS-val and MSLS-challenge sets, as done in previous works [8, 11, 12].

Nordland [14] captures images from a fixed viewpoint in the front of a train in four seasons. This dataset exhibits significant variations in conditions such as season and lighting, without viewpoint changes. Its images primarily depict suburban and natural environments, and the ground truth information is provided through frame-level correspondence. Following previous works [5], we extract images at 1FPS, and use the winter images as queries and the summer images for reference (i.e. database).

AmsterTime [20] contains more than a thousand query-reference image pairs captured from Amsterdam. Each pair consists of a grayscale historical image as the query and a contemporary image from the same place (identified by human experts) as the reference. The dataset involves very long-term time spans, and diverse domain variations in viewpoints, modalities (RGB vs grayscale), etc., which makes it quite difficult for VPR.

SVOX [6] is a cross-domain VPR dataset collected in a variety of weather and lighting conditions. It includes a large-scale database sourced from Google Street View images spanning the city of Oxford. The queries are extracted from the Oxford RobotCar dataset [13] and divided into multiple subsets for different weather and lighting conditions. We evaluate the model performance using the three most challenging query subsets: SVOX-Night, SVOX-Rain, and SVOX-Sun.

K. Compared Methods Details

NetVLAD [3] is a well-known VPR approach with a differentiable VLAD layer, which can be integrated into common neural networks. In our experiments, we use its PyTorch implementation[1] with the released VGG16 model trained on Pitts30k for comparison.

SFRS [7] utilizes self-supervised image-to-region similarities to mine hard positive samples for training a more robust NetVLAD model. In the comparison experiments, we follow its official implementation[2] with the model trained on the Pitts30k dataset.

Patch-NetVLAD [8] is a two-stage method that utilizes NetVLAD-based multi-scale patch-level features to re-rank the candidate images retrieved using NetVLAD global features. The official implementation[3] with the performance-focused configuration is used in our experiments. Following the original paper, the model trained on the Pitts30k dataset is tested on Pitts30k and Tokyo24/7, while the model trained on the MSLS dataset is evaluated on MSLS (-val and -challenge).

TransVPR [18] is a two-stage VPR method that leverages attentions from three levels of Transformer to produce global features for candidates retrieval, and employs an attention mask to filter feature maps to yield key-patch descriptors for re-ranking candidates. The official implementation[4] is used for comparison experiments. The model trained on the Pitts30k dataset is evaluated on Pitts30k and Tokyo24/7, and the model trained on the MSLS dataset is assessed on MSLS.

CosPlace [4] treats VPR model training as a classification problem and trains the model on the individually constructed San Francisco eXtra Large (SF-XL) datasets with the Large Margin Cosine Loss (i.e., cosFace) to achieve remarkable results. We follow its official implementation[5] with the VGG16 backbone (producing 512-dim global features) for testing.

GCL [10] uses an automatic annotation strategy producing graded similarity labels for image pairs to re-label VPR datasets, and a novel generalized contrastive loss to utilize such labels to train contrastive networks. we use the results (yield by the version using ResNet152-GeM with PCA) from the original paper for comparison.

MixVPR [2] introduces a novel holistic feature aggregation approach for global-retrieval-based VPR. It utilizes feature maps yielded by a pre-trained backbone as initial feature representations, and employs a sequence of Feature-Mixer modules to incorporate global relationships into each

[1] https://github.com/Nanne/pytorch-NetVlad

[2] https://github.com/yxgeee/OpenIBL

[3] https://github.com/QVPR/Patch-NetVLAD

[4] https://github.com/RuotongWANG/TransVPR-model-implementation

[5] https://github.com/gmberton/CosPlace

feature map to produce final global features. We follow the official implementation[6] and its best configuration, i.e., using the ResNet50 backbone producing 4096-dim global features, for comparison experiments.

EigenPlaces [5] can be seen as an improvement work on CosPlace. This work trains the networks on images from different viewpoints (of the same place), thus improving the viewpoint robustness of learned global representations. It is the most recent work and achieves the best performance on most VPR datasets. We follow its official implementation[7] and the configuration using ResNet50 as the backbone to yield 2048-dim features.

Besides, the results on the three challenging datasets (Nordland, AmsterTime, and SVOX) are directly referenced from the EigenPlaces paper [5]. These results are basically consistent with what we have reproduced.

L. Additional Qualitative Results and Failure Cases

In the main paper, we have presented a small number of qualitative results to show the robustness of our approach in challenging scenarios. In this section, we add more examples to vividly demonstrate the performance of VPR methods. Suppl. Fig. 2, Suppl. Fig. 3, and Suppl. Fig. 4 show examples on Pitts30k, Tokyo24/7, and MSLS-val, respectively. These examples demonstrate that our method is more robust against variations in conditions and viewpoints, as well as perceptual aliasing, than previous methods. Suppl. Fig. 5 and Suppl. Fig. 6 show examples on AmsterTime and Nordland, which demonstrate that our method can correctly recognize place images over long time spans and under extreme environments in general. However, our method also produces erroneous results in a few cases (the last examples of these two figures) when images from different places are very similar, especially when lacking discriminative landmarks.

M. Limitations

In addition to the failure case mentioned in the previous section, our approach has two limitations. First, although our approach achieves excellent results with the 512-dim compact feature on Pitts30k, it does not perform well on datasets with severe condition changes (e.g., Tokyo247, MSLS) when the descriptor dimension is reduced to very low. Secondly, setting the inference batch size to 1 will render our cross-image encoder ineffective, resulting in a gap between training and testing, and thus not achieving optimal performance. These are the focal points for future improvement of our method.

[6] https://github.com/amaralibey/MixVPR

[7] https://github.com/gmberton/EigenPlaces

References

[1] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Gsv-cities: Toward appropriate supervised visual place recognition. *Neurocomputing*, 513:194–203, 2022. 1, 3, 4

[2] Amar Ali-Bey, Brahim Chaib-Draa, and Philippe Giguere. Mixvpr: Feature mixing for visual place recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2998–3007, 2023. 2, 5

[3] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016. 5

[4] Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking visual geo-localization for large-scale applications. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4878–4888, 2022. 1, 2, 5

[5] Gabriele Berton, Gabriele Trivigno, Barbara Caputo, and Carlo Masone. Eigenplaces: Training viewpoint robust models for visual place recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11080–11090, 2023. 2, 5, 6

[6] Gabriele Moreno Berton, Valerio Paolicelli, Carlo Masone, and Barbara Caputo. Adaptive-attentive geolocalization from few queries: A hybrid approach. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2918–2927, 2021. 5

[7] Yixiao Ge, Haibo Wang, Feng Zhu, Rui Zhao, and Hongsheng Li. Self-supervising fine-grained region similarities for large-scale image localization. In *European conference on computer vision*, pages 369–386. Springer, 2020. 2, 5

[8] Stephen Hausler, Sourav Garg, Ming Xu, Michael Milford, and Tobias 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*, pages 14141–14152, 2021. 5

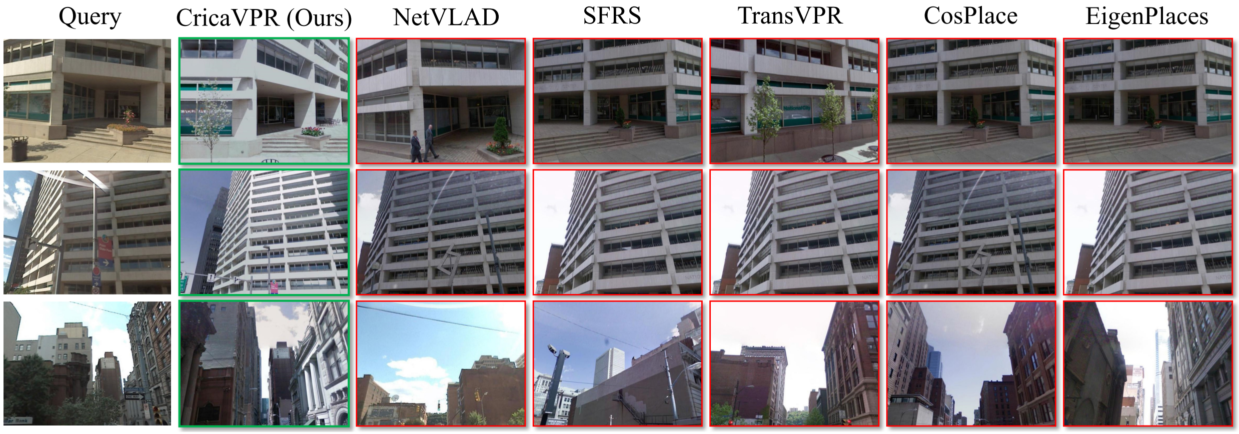
[9] Shibo Jie and Zhi-Hong Deng. Convolutional bypasses are better vision transformer adapters. *arXiv preprint arXiv:2207.07039*, 2022. 3

[10] María Leyva-Vallina, Nicola Strisciuglio, and Nicolai Petkov. Data-efficient large scale place recognition with graded similarity supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23487–23496, 2023. 5

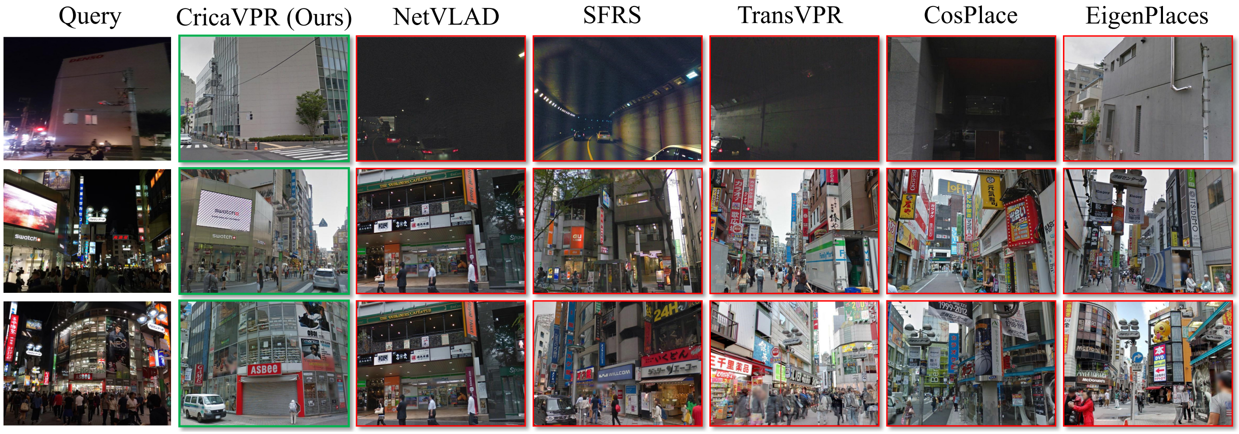
[11] Feng Lu, Shuting Dong, Lijun Zhang, Bingxi Liu, Xiangyuan Lan, Dongmei Jiang, and Chun Yuan. Deep homography estimation for visual place recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10341–10349, 2024. 5

[12] Feng Lu, Lijun Zhang, Xiangyuan Lan, Shuting Dong, Yaowei Wang, and Chun Yuan. Towards seamless adaptation of pre-trained models for visual place recognition. In *The Twelfth International Conference on Learning Representations*, 2024. 5

[13] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset.



Supplementary Figure 2. **Qualitative results on Pitts30k.** The proposed CricaVPR returns the correct database images, while other methods produce wrong results. In these examples, most of the other methods suffer from perceptual aliasing. In the first two examples, all other methods return highly similar but wrong places. In the third example, the buildings on the right of the images returned by TransVPR, CosPlace, and EigenPlaces are highly similar to the building on the right of the query image, indicating that the appearance of this building is not distinguishable enough, making these methods suffer from perceptual aliasing.



Supplementary Figure 3. **Qualitative results on Tokyo24/7.** The proposed CricaVPR returns the correct database images, while other methods produce wrong results. In these examples, the main challenges are the variations in lighting conditions across day and night, as well as perceptual aliasing. In the first example, as the query image is a nighttime image, all methods except for ours and EigenPlaces return nighttime but incorrect images. EigenPlaces returns a similar but incorrect image.

The International Journal of Robotics Research, 36(1):3–15, 2017. 5

[14] Niko Sünderhauf, Peer Neubert, and Peter Protzel. Are we there yet? challenging seqslam on a 3000 km journey across all four seasons. page 2013, 2013. 5

[15] Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 883–890, 2013. 4

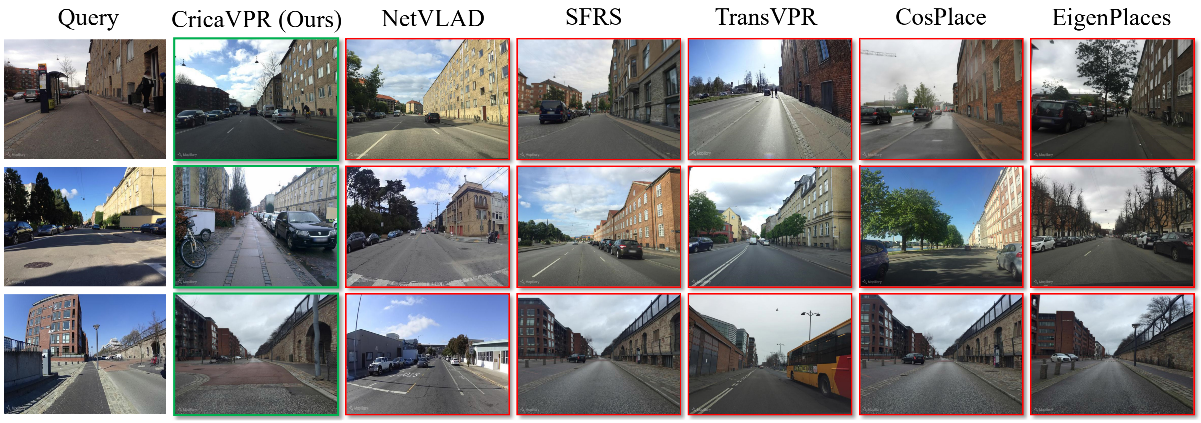
[16] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *IEEE conference on computer vision and pattern recognition*, pages 1808–1817, 2015. 5

[17] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 1

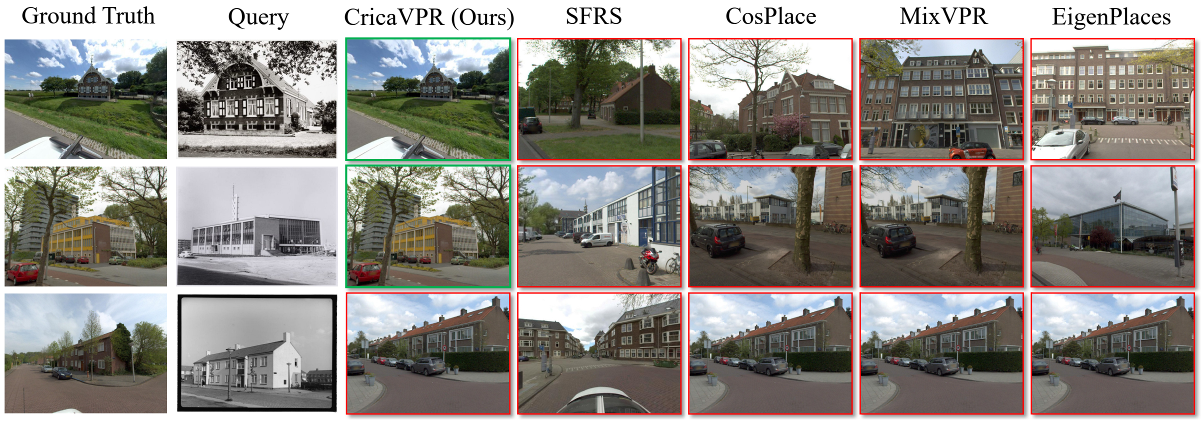
[18] Ruotong Wang, Yanqing Shen, Weiliang Zuo, Sanping Zhou, and Nanning Zheng. Transvpr: Transformer-based place recognition with multi-level attention aggregation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13648–13657, 2022. 5

[19] Frederik Warburg, Soren Hauberg, Manuel Lopez-Antequera, Pau Gargallo, Yubin Kuang, and Javier Civera. Mapillary street-level sequences: A dataset for lifelong place recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2626–2635, 2020. 5

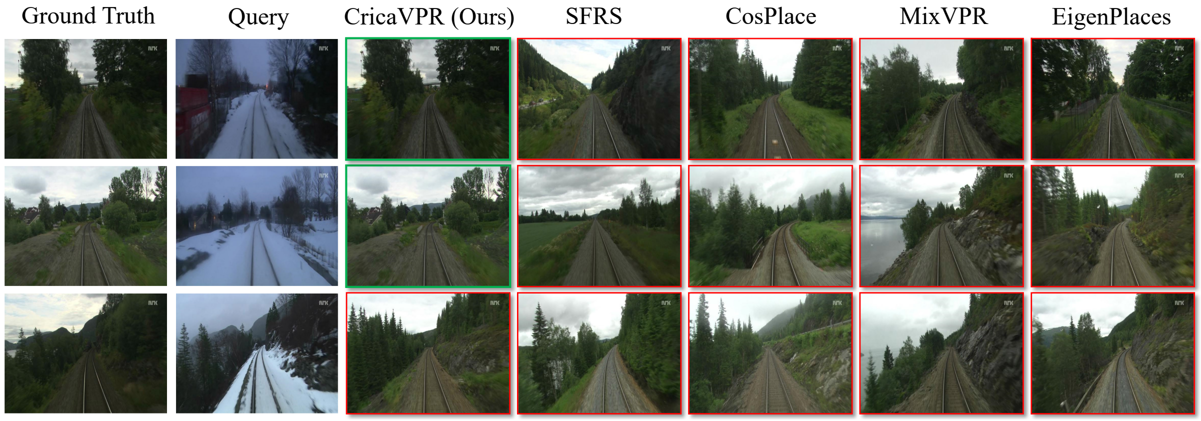
[20] Burak Yildiz, Seyran Khademi, Ronald Maria Siebes, and Jan Van Gemert. Amstertime: A visual place recognition benchmark dataset for severe domain shift. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 2749–2755. IEEE, 2022. 5



Supplementary Figure 4. **Qualitative results on MSLS-val.** The proposed CricaVPR returns the correct database images, while other methods produce wrong results. In the first example, NetVLAD returns a highly similar but wrong image. In the second example, the building on the left of the query image is occluded by trees, causing NetVLAD, TransVPR, and CosPlace to return incorrect results with obvious trees on the left side. In the third example, SFRS, CosPlace, and EigenPlaces return database images that are geographically close to the query image but exceed the set threshold (i.e. still wrong).



Supplementary Figure 5. **Qualitative results on AmsterTime.** There is a very long time span between the query (grayscale) image and the reference (RGB) image in this dataset. In the first two examples, the proposed CricaVPR returns the right database images, while other methods produce wrong results. In the first example, the discriminative buildings only occupy a small region of the reference image. In the second example, a new building appears in the reference image, and the original building has undergone some modifications. These cause other methods to return incorrect results. In the last example, there are images from different places in the database that are highly similar to the query image, causing none of the methods to retrieve the correct result.



Supplementary Figure 6. **Qualitative results on Nordland.** These examples show drastic variations in conditions (season, weather, and lighting). Meanwhile, there are almost no discriminative buildings in the images. These challenges are difficult to address for previous VPR methods, resulting in incorrect results being returned by all of them. Our method gets the right result in the first two examples but fails in the last one.