— Supplementary Material —
Breaking the Frame:
Visual Place Recognition by Overlap Prediction

Tong Wei[1], Philipp Lindenberger[2], Jiří Matas[1], Daniel Barath[2,3]
[1] Visual Recognition Group, FEE, Czech Technical University in Prague
[2] Computer Vision and Geometry Group, ETH Zurich, [3] HUN-REN SZTAKI
{weitong, matas}@fel.cvut.cz, {philipp.lindenberger, danielbela.barath}@inf.ethz.ch

1. Ablations

In this section, we will analyze certain components and parameters of the proposed method.

Radius Search Threshold. In the main paper, similar patches are selected by running radius search in the embedding space. We compute the median similarity as the radius search threshold over 100 random samples from the query and database images. Here, to understand how sensitive VOP is to the setting of this threshold, we show tuning results on the validation set of MegaDepth with different thresholds (horizontal axis) in Fig. 1. AUC scores and median errors are shown on the top-10 retrieved image pairs.

Dropout Layer. Table 1 shows the relative pose estimation performance on the top-k images retrieved by the model with or w/o global prefilter ([CLS] tokens), and dropout layer. Testing sets of MegaDepth are used, as in the main paper. The [CLS] prefiltering improves the AUC scores. However, it increases the median pose errors marginally at the same time. As shown in the fifth row, data augmentation is essential in robustly learning the embeddings. Also, the last two rows in Table 1 show that the dropout layer improves the performance on MegaDepth. As shown in the main paper, VOP generalizes well for pose estimation on other data and indoor localization.

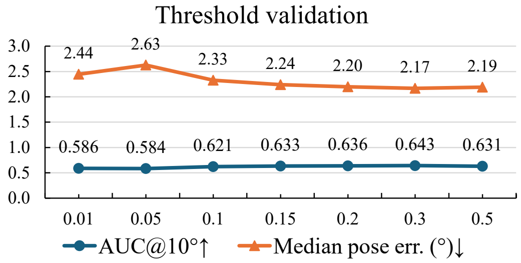


Figure 1. Ablations on the threshold used in radius search. AUC@10° and median pose errors on the validation scene shown.

Prefilter	Augment	Dropout	AUC@10° ↑	Med. pose error (°) ↓	inliers ↑
✓	×	×	65.1	2.19	**272.5**
×	✓	×	66.3	2.18	222.0
✓	✓	×	66.7	2.18	263.0
✓	✓	✓	**67.6**	**2.02**	246.5

Table 1. Relative pose estimation on the MegaDepth dataset [29] on the top 5 retrieved images using different configurations, with the best results in bold. Prefilter indicates if the [CLS] token was employed to shortlist the potential candidates before overlap prediction. Augment refers to whether data augmentation was used.

2. Qualitative Results

Most VPR methods prioritize retrieving similar images, typically resulting in short baselines that are not suitable for reconstruction. These goals conflict: the most similar images often produce short baselines, making pose estimation unstable. We aim to move beyond traditional similarity metrics and design retrieval methods tailored for geometric challenges, such as selecting images suitable for pose estimation. We visualize three query examples in Fig. 2 with their top-1 retrieved images using different methods. VOP results in low pose errors as we find images with reasonable baselines for stable pose estimation.

3. Discussions

Training. To better understand the training process, Fig. 3 illustrates the training and validation losses on patch-level contrastive loss and the average similarity changing among different epochs on MegaDepth. It shows the contrastive loss helps to learn the embeddings of negative patches less similar and closer to positive ones, and it converges fast. The similarities shown are averaged over all positive/negative patches of the validation set indicated by the GT labels built from 3D reconstructions.

Supervision. As discussed in the main paper, we build the supervision based on the ground-truth depth provided by Megadepth for training. However, some datasets do not pro-

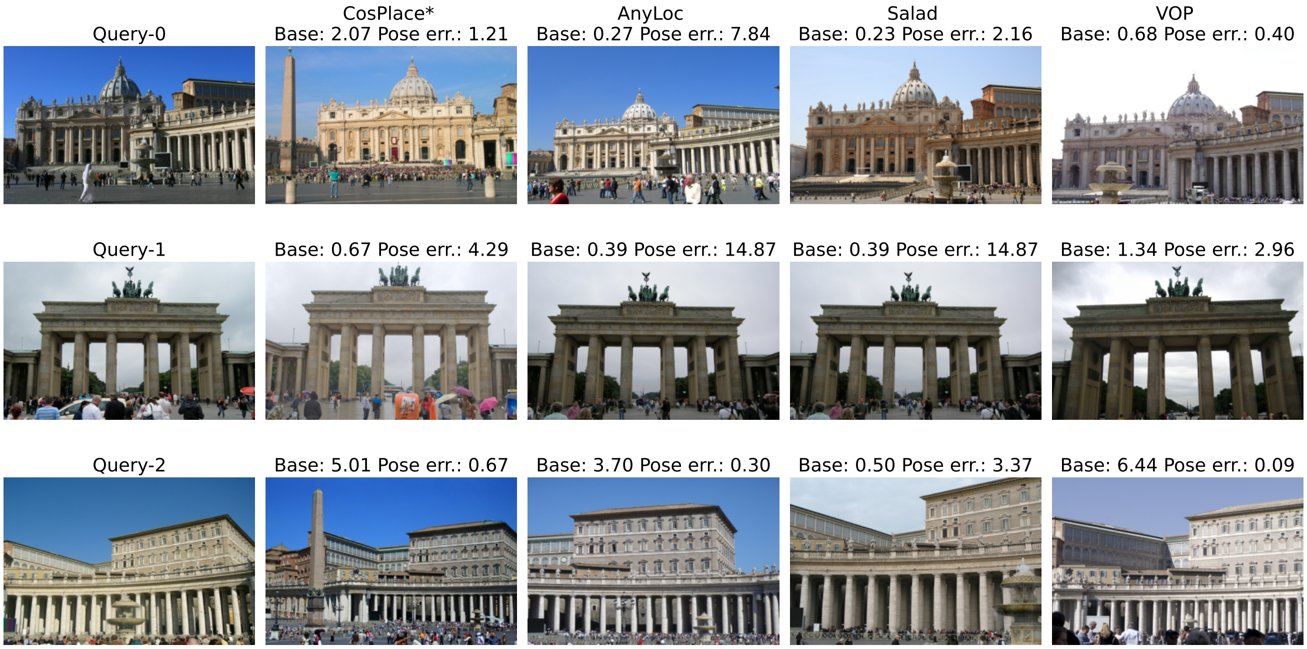

Query-0
CosPlace*
Base: 2.07 Pose err.: 1.21
AnyLoc
Base: 0.27 Pose err.: 7.84
Salad
Base: 0.23 Pose err.: 2.16
VOP
Base: 0.68 Pose err.: 0.40
Query-1
Base: 0.67 Pose err.: 4.29
Base: 0.39 Pose err.: 14.87
Base: 0.39 Pose err.: 14.87
Base: 1.34 Pose err.: 2.96
Query-2
Base: 5.01 Pose err.: 0.67
Base: 3.70 Pose err.: 0.30
Base: 0.50 Pose err.: 3.37
Base: 6.44 Pose err.: 0.09

Figure 2. Baselines and pose errors between the retrieved images and queries using different methods shown.

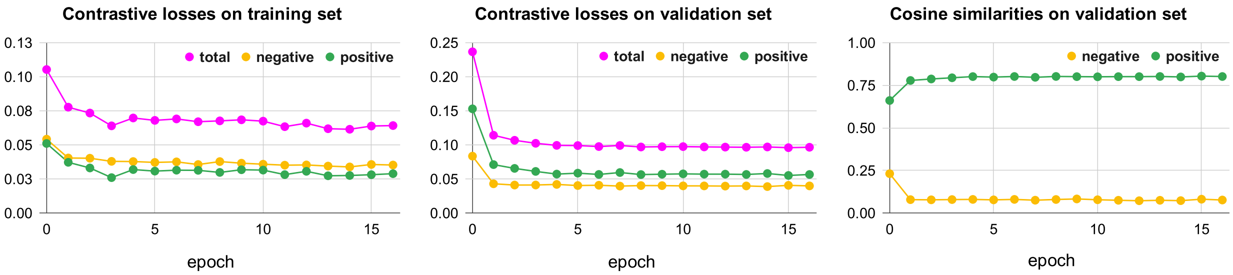


Figure 3. Contrastive losses on the training set (*left*) and validation (*middle*) over different epochs shown for negative or positive patch pairs. The *right* plot shows the average cosine similarities over different patch samples.

vide depth information. Thus, we provide another option to build the supervision, *i.e.*, patch-level positive and negative labels by matching the images using the SOTA dense feature matching method, RoMA [3]. The patch pairs containing more than 5 correspondences are set as positive, and vice versa. As shown in Table 2, RoMA-based supervision works comparable to the model trained with depth supervision. Thus, we recommend using RoMA dense correspondences as a labeling option when fine-tuning on new data.

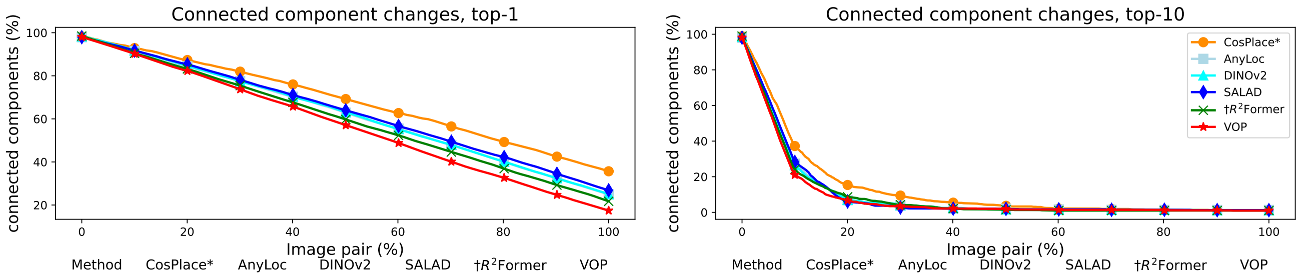
Supervision	depth-based	RoMA-based
Avg. accuracy (%) ↑	**74.8**	74.1
Avg. med. error (°) ↓	**1.6**	1.7

Table 2. Average accuracy (%) and median pose error (°) on the retrieved top-5 images using different supervision on the test sets as the main paper. InLoc is excluded from the median error.

Method	Backbone	train data	pre-filter
NetVLAD [2]	ResNet-18 [4]	Pitts30k [5]	-
CosPlace [6]	ResNet-101 [4]	SF-XL [6]	-
CosPlace* [6]	ResNet-101	MegaDepth	-
DINOv2 [36]	ViT-G14 [2]	LVD-142M [36]	-
AnyLoc [26]	DINOv2	-	-
SALAD [25]	DINOv2	GSV-Cities [1]	-
P-NetVLAD [20]	NetVLAD	Pitts30k& MSLS [6]	NetVLAD
†P-NetVLAD [20]			DINOv2-CLS
R^2Former [62]	ViT-S [2]	MSLS [6]	R^2Former
†R^2Former [62]			DINOv2-CLS
VOP	DINOv2	MegaDepth	DINOv2-CLS

Table 3. Backbones, training data, and the perfilter methods used for reranking shortlist generation are listed.

Fairness Comparison. As shown in Table 3, we show the backbones, training data and prefiltering methods (for reranking methods). We include fine-tuned results for CosPlace and results of the reranking methods tested with the same prefilter as ours.



Method	CosPlace*	AnyLoc	DINOv2	SALAD	†R^2Former	VOP
max cc_{size}	14.5	18.5	19.7	16.5	45.9	**53.2**
# cc	35.7	24.9	25.0	26.8	21.7	**17.4**
# skipped	17.1	21.6	19.1	**23.1**	4.2	13.9
# success	64.3	75.1	75.0	73.2	78.3	**82.6**
# failure	18.6	**3.3**	5.9	3.8	17.5	3.6

Method	CosPlace*	AnyLoc	DINOv2	SALAD	†R^2Former	VOP
max cc_{size}	98.2	97.0	98.4	97.8	**100.0**	**100.0**
idx_{last}	46.0	39.4	35.4	35.0	**26.4**	35.4
# skipped	84.1	**88.9**	88.7	**88.9**	86.4	88.4
# success	**9.8**	**9.8**	**9.8**	**9.8**	**9.8**	**9.8**
# failure	6.1	**1.3**	1.5	**1.3**	3.8	1.8

Figure 4. The number of connected components (vertical axis, cc) is plotted against the index of the image pair (horizontal) on which RANSAC-based relative pose estimation runs. The *left* plot shows results for the top-1 database (DB) images paired with $0.4K$ query images, where the number of pairs equals the number of queries. The *right* plot shows results for the top-10 DB images with a termination criterion applied when all images are in a single component. Row "max cc_{size}" is the number of elements in the largest cc. Row "# cc" is the final number of connected components, while "idx_{last}" shows the index when the termination criterion was triggered in the *right* plot. Row "# skipped", "# success", and "# failure" show the numbers of skipped, succeeded, and failed RANSACs. All are in percentages.

Pose Graph construction. Similar to Fig. 5 shown in the main paper, here, we replace 1K random orders of queries by their predicted similarity/overlap scores. As shown in Fig. 4, VOP built a larger connected component on top-1 of the queries than the competitors, also with the most number of successful RANSACs. On the right plot (*top-10*), VOP iterates more queries than R^2Former when ranked by the scores. However, the run-time is comparable as VOP skipped more times of RANSAC runs.

Image Patching. We investigated the number of patches to be used to split the images. The experiments were conducted on the testing scenes of MegaDepth, with all the images resized to 224^2. From the trained VOP model with a patch size of 14^2, we can extract 256 patch descriptors. Then, average pooling is applied to aggregate the patch descriptors to different patch sizes, such as 28^2, 56^2, 112^2, and 224^2. For example, patch size = 224^2 will lead to a single patch of an image. The retrieval is done on the same prefiltered image list, similarly as in the main paper. As shown in Table 4, the aggregated patches *e.g.*, patch size=224^2, perform worse than 14^2 on MegaDepth pose estimation and could not generalize well on Inloc localization. This demonstrates that patch-level features can potentially improve estimated pose and other geometric problems.

Storage & Query Speed. As we reduce the dimensionality of the DINOv2 features from 1024 to 256, the embeddings of all patches of each image need a total of 512 kB, while, for AnyLoc, the storage per image is 384 kB. While we require slightly more storage than AnyLoc [26], the difference is small. Compared to storing the local features in the reranking-based methods, VOP costs less. In addition, we compare the time of querying an image from the database of different sizes using AnyLoc or VOP in seconds. Prefiltering top 20 images by DINOv2 [CLS] token and running VOP for reranking to get top-1 out of 500 images cost 0.009 seconds, while 0.003 for AnyLoc. Querying top-1 from 5K images (prefiltered to 100), our method costs 0.03 seconds, while AnyLoc runs in 0.02s. Note that VOP needs much more time for radius search without prefiltering. We recommend using VOP as a reranking method in retrieval.

Patch size	AUC@10° ↑	med. pose err. ↓	recall@5°, 0.5m	↑
	MegaDepth		DUC1	DUC2
224^2	67.0	2.09	30.8	24.4
112^2	67.0	2.09	38.4	38.9
56^2	66.5	2.17	48.5	57.3
28^2	65.9	2.29	59.1	72.5
14^2	**67.6**	**2.03**	**72.2**	**77.1**

Table 4. Ablations on different patch sizes used in inference time. We show the AUC@10° scores and median pose errors of the top-5 retrieved images on MegaDepth [29], and the recall@°, 0.5m on the top-40 of the localization data, Inloc [51] (DUC1, DUC2).

References

[1] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguere. Gsv-cities: Toward appropriate supervised visual place recognition. *Neurocomputing*, 2022. 2

[2] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[3] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *CVPR*, 2024. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[5] Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *CVPR*, 2013. 2

[6] 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 *CVPR*, 2020. 2