Distill Structural Knowledge by Weighting Samples for Visual Place Recognition
Supplementary Material

Yanqing Shen Sanping Zhou Jingwen Fu Ruotong Wang Shitao Chen Nanning Zheng
National Key Laboratory of Human-Machine Hybrid Augmented Intelligence,
National Engineering Research Center for Visual Information and Applications,
Institute of Artificial Intelligence and Robotics, Xi’an Jiaotong University

This supplementary material is structured as follows. In Section 1, we explain the reasons for choosing the datasets in the main paper and present the detailed dataset configurations. In Section 2, we show some additional quantitative results and analysis on datasets, as well as results on the computation time. This section also contains several qualitative results. Section 3 provides the experimental details of methods, including detailed training settings of our method, ablation studies and the implementation of SOTA methods. Finally, Section 4 contains some additional ablation studies.

1. Detailed Dataset Configuration

To facilitate an informed assessment of the results, we further detail datasets and the usage, which were briefly mentioned in Section 4.2 of the main paper. We evaluate our method on 4 key benchmark datasets.

**Mapillary Street Level Sequences (MSLS).** MSLS [15] is introduced to promote lifelong place-recognition research, and contains over 1.6 million images recorded in urban and suburban areas over 7 years. Compared to other datasets, it covers the most comprehensive variation (dynamic objects, season, light, viewpoint, and weather), and we only evaluate the image-to-image task. GPS coordinates and compass angles are provided for each image, and the ground truth corresponding to a query is the reference images located within 25m and 40° from the query. The dataset is divided into a training set, a public validation set and a withheld test set (MSLS challenge). In training, we define a distance $d_{qp}$ to represent the FOV overlap between query $q$ and positive $p$:

$$d_{qp} = \|x_q - x_p\|_2 / 25 + (\theta_q - \theta_p) / 40 < 1$$  \hspace{1cm} (1)$$

where $x$ is the GPS coordinate and $\theta$ is the angle. Eq. (1) ensures an overlapping area between a query and a positive.

**Nordland.** The Nordland dataset [13] contains four timestamp-aligned image sequences recorded in four seasons, and hence it contains challenging appearance changes and different weather conditions while few viewpoint variations. We use the partitioned dataset [9] containing 3450 images per sequence, with summer as reference and winter as query. Same as [6, 9], we also remove all black tunnels and times when the train is stopped. Ground truth tolerance is set to 2 frames, that means that one query image corresponds to 5 reference images.

**Pittsburgh.** The Pittsburgh dataset [14] contains 250k images derived from Google Street View panoramas. The data is generated by 24 perspective images (two pitch and twelve yaw directions) at each place, which results in significant viewpoint variations, along with dynamic objects. As only GPS information is available, the ground truths for evaluation are defined as reference images within 25m from the query. In our experiments, we use the subset, Pitts30k, and the weakly supervised sample mining strategy [1] in training. It contains 30k database images and 24k queries, which are geographically divided into train/val./test sets.

2. Additional Results

**Complementarity of RGB and SEG.** In the main paper, we mentioned that there is a sample-level complemen-
Figure 2. **Qualitative Results.** In these examples, our method successfully retrieves the matching reference images. For other methods, red borders indicate false matches and green borders indicate correct matches.

Figure 3. **Comparison with state-of-the-art on MSLS val. set.** We show the comparison of Recall@N performance with other methods. Results w/o re-ranking are depicted in dotted line, while results with re-ranking are depicted in solid line. * indicates unofficially reproduced results, and details are in Section 3.2.

Table 1. Performance of selective distillation with different samples on Nordland dataset. None refers to the rgb-branch without distillation and All refers to non-selective distillation.

<table>
<thead>
<tr>
<th>Group for distillation</th>
<th>Sample ratio</th>
<th>Nordland</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R@1 R@5 R@10 R@15 R@20</td>
<td></td>
</tr>
<tr>
<td>None</td>
<td>0% 48.3 69.2 76.1 80.6 83.0</td>
<td></td>
</tr>
<tr>
<td>All</td>
<td>100% 55.4 71.9 76.9 79.4 80.7</td>
<td></td>
</tr>
<tr>
<td>D1</td>
<td>4.11% 51.2 69.1 75.9 79.8 82.5</td>
<td></td>
</tr>
<tr>
<td>D2</td>
<td>50.67% 52.5 72.1 79.8 83.5 85.8</td>
<td></td>
</tr>
<tr>
<td>S1</td>
<td>69.55% 54.8 74.0 81.1 83.2 84.8</td>
<td></td>
</tr>
<tr>
<td>S2(D1)</td>
<td>30.45% 33.0 54.4 64.2 69.5 72.8</td>
<td></td>
</tr>
<tr>
<td>R1</td>
<td>73.24% 55.2 73.7 79.6 82.6 84.0</td>
<td></td>
</tr>
<tr>
<td>R2</td>
<td>26.76% 51.0 67.4 73.0 75.4 77.8</td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>69.55% 56.1 75.5 82.9 86.2 88.3</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Performance of seg-branch with different number of clustered classes. Note that † indicates another weighted encoding.

<table>
<thead>
<tr>
<th>Clustered classes</th>
<th>Nordland</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R@1 R@5 R@10 R@15 R@20</td>
</tr>
<tr>
<td>3</td>
<td>40.4 63.6 73.6 78.0 81.1</td>
</tr>
<tr>
<td>6†</td>
<td>41.1 63.8 73.2 78.6 81.6</td>
</tr>
<tr>
<td>150</td>
<td>25.1 41.4 50.4 55.3 58.9</td>
</tr>
<tr>
<td>6</td>
<td>34.5 53.0 61.3 65.9 69.1</td>
</tr>
<tr>
<td>(Ours)</td>
<td>56.1 75.5 82.9 86.2 88.3</td>
</tr>
</tbody>
</table>

matches with challenging conditions, such as viewpoint changes, occlusions caused by dynamic objects, and seasons. In these examples, other methods show a tendency to retrieve images with similar appearance as the query. Especially in the case of MSLS, the color tone and the vehicle ahead of retrievals of Patch-NetVLAD and DELG are consistent with query. Ours can successfully retrieve images based on structural information, paying more attention to the spatial information of static objects in the background.

**Additional recall plots.** Table 1 in the main pa...
per shows the Recall@N performance on the benchmark datasets. More intuitively, Figure 3 shows the detailed Recall@N performance for the MSLS validation dataset.

**Efficiency-accuracy.** Table 2 in the main paper shows the feature extraction, feature matching time and storage required to process each query. In Figure 4, we show the the accumulated time of feature extraction and matching as well as their performances on MSLS validation set. Ours achieves the best trade-off between accuracy and efficiency.

**Ablations on other datasets.** In Table 1 and Table 2, we provide more results on Nordland dataset for ablation experiments discussed in the main paper.

### 3. More Implementation Details

#### 3.1. Training Details

**Ours.** To obtain segmentation images offline, we use ADE20K [19] and PSPNet [17] and the open-source codebase with configuration file of “ade20k-resnet50dilated-ppm_deepsup”. MobileNetV2 is initialized with the pre-trained weights on ImageNet. MobileNet-L in seg-branch refers to the implementation of depth-stream in MobileSal.

In the first learning stage, rgb-branch and seg-branch are fine-tuned with the whole backbone using initial lr=0.001, where rgb-branch starts with a pre-trained model on ImageNet and seg-branch starts with random parameters. In the second learning stage, backbone also starts with a pre-trained model on ImageNet and is fine-tuned as a whole with initial lr=0.0001. We also attempted to continue the second stage on the basis of pre-trained network in the first training stage, but the difference is not significant.

Models are all optimized by AdamW optimizer [18] with 0.0001 weight decay and cosine learning rate decay schedule, and in VPR loss is 0.1. The network which yields the best recall@5 on the val. set is used for testing.

For running SuperGlue network with SuperPoint based on our global retrieval, the details are shown in Section 3.2.

**Concat-input.** We concatenate RGB image and encoded segmentation label map in channel $C$ as input, where the $C$ of the first layer changes from 3 to 9 compared with rgb-branch. The model is fine-tuned with the whole backbone and initial lr=5e-5. The dim of global features is 448.

**Concat-feat.** Two separate networks, same as the two branches in the first stage, are used to extract features separately. Then the two features are concatenated, followed by a $L_2$ normalization step, as final global features to build loss function. The two models are fine-tuned with the whole backbone and initial lr=0.00005. The final loss is the direct sum of the two losses. The dim of global features is 928.

#### 3.2. Implementation Details of Baselines

**NetVLAD** [1]. We use the pytorch implementation and its released model trained on Pitts30k training set with VGG-16 backbone. Note that this method does not resize the image.

**SFRS** [5]. This work proposes a self-supervised method with image-to-region similarities to fully explore the potential of difficult positive images alongside their sub-regions. We use the official implementation and the released model trained on Pitts30k training set.

**SP-SuperGlue** [4, 12]. SuperGlue trains a neural network that matches two sets of local features by jointly finding correspondences and rejecting non-matchable points. The implementation of SP-SuperGlue in our main paper includes: using NetVLAD for global retrieval, then extracting SuperPoint local features, and applying SuperGlue to identify matches and to re-rank candidates. We use the official implementation and choose the pre-trained outdoor weights on MegaDepth dataset [8].

**Patch-NetVLAD** [6]. This work derives patch-level features from NetVLAD residuals. We use the official implementation for speed-focused and performance-focused configurations in our main paper. Following the original paper, the model trained on Pitts30k is used for urban imagery (Pittsburgh). The model trained on MSLS is used for all other conditions.

**DELG** [2]. This work unifies global and local features into a single deep model. We refer to the pytorch implementation of two models and change the extraction of global features (dim=2048) to the way in the original paper: For global features, we use 3 scales; $L_2$ normalization is applied for each scale independently, then the three global features are average-pooled, followed by another $L_2$ normalization step. For local features, all the reproduced re-

---

1. The largest open-source dataset for semantic segmentation and scene parsing, similar to the distribution of VPR datasets.
4. https://download.pytorch.org/models/mobilenet_v2-b0353104.pth
5. https://github.com/yuhuan-wu/MobileSal
6. https://github.com/milesial/Pytorch-UNet
8. https://github.com/yxgeee/OpenVBL
12. The results in Table 2 in the main paper are the best of the two models.
results are from the provided model, in which dim is 512 and is different from the original paper (dim=128).

**TransVPR.** This work proposes a holistic model based on vision Transformers, which can aggregate task-relevant features. We use the official implementation\(^\text{13}\). Same to Patch-NetVLAD, it finetuned the model on MSLS training set and Pitts30k training set.

4. Additional Ablation Studies and Analysis

**Prior weights for encoding.** Limited to the length of the article and the research content, in Section 4.5 of the main paper, we only manually set three different weighting cases and choose the best of the three, without finding the optimal one. Figure 6 indicates that the value of static buildings should be larger than that of dynamic objects, which is accorded with our intuition. This weighted attempt provides the possibility for follow-up research, and further advancements can be done, such as using grid search or learning methods to obtain the weights through iterative optimization.

**Weighting function.** For the weight function Eq. (4) given in the main paper, we make the following supplementary explanations. In main paper, we attempted to apply different constant weights to different groups according to the performance of separate group in Table 4.

In fact, we also tried other constant weights, as shown in Table 3. The results show that GP-D(8-4-1-0) and GP-D(4-2-1-0) has a more reasonable weight distribution than others, which proves our prediction of the importance of different groups: the greater the performance improvement when participating in distillation alone, the higher the weight. Moreover, the difference between GP-D(8-4-1-0) and GP-D(4-2-1-0) is small, and this experiment is mainly for providing a numerical reference for the design of our weight function. Therefore, we finally choose GP-D(8-4-1-0) as the reference.

Based on reference constant weights, the specific function design includes the following consideration:

- \(\varphi\) cannot be negative;
- The non-zero part of \(\varphi\) should be proportional to \(y - x\) and inversely proportional to \(x\);
- The value of \(\varphi\) cannot be too large;
- The partial derivatives should be different for 3 groups.

The prototype of the function can be denoted as \(\frac{f(y-x)}{g(x)}\), where \(f(\cdot)\) and \(g(\cdot)\) are monotonically increasing functions. Considering \(x\) should play an important role in weights than \(y - x\), we choose liner for \(f(\cdot)\) and natural logarithm for \(g(\cdot)\) (see Figure 5).

Throughout the design process, we did not perform rigorous tuning of the parameters, but simply chose representative design to demonstrate our insight and motivation. In Table 4, we show more ablation experiments by replacing \(\frac{y-x}{\ln(x+1)}\) with \(\frac{y-x}{x}\) in (4). Combined with the performance of distillation with fixed weights in Table 3, it can be seen that the function performs better than the discrete fixed weights and prototype function, showing the advantages of (4).

**Sensitivity to hyper-parameters.** In Section 4.4 of the main paper, we have performed ablation experiments on the most important hyper-parameters. We further evaluate the sensitivity of our model to changes in the other two hyper-parameters: \(N_1\) and \(N_m\) in our group partition strategy.

Here we perform unweighted selective distillation with the experimental setup of GP-S, that is, on samples belong to \(S_1\). The results are shown in Table 5 and Figure 6. It can be seen that within the appropriate range of 5-15, the performance is relatively close and we select 10 in the main paper.

After \(N_1\) is set as 10, \(N_m\) is mainly used to limit the weight range.

**Sensitivity to Segmentation Models.** In order to use accurate semantic information, some previous works [7, 10]...
Figure 6. Ablation experiments on the recall performance of StructVPR with different $N_t$.

Table 5. The sample ratio corresponding to different $N_t$.

<table>
<thead>
<tr>
<th>$N_t$</th>
<th>3</th>
<th>5</th>
<th>10</th>
<th>15</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{S}_1$</td>
<td>60.75%</td>
<td>64.76%</td>
<td>69.55%</td>
<td>72.11%</td>
<td>73.82%</td>
</tr>
<tr>
<td>$\mathcal{S}_2$</td>
<td>39.25%</td>
<td>35.24%</td>
<td>30.45%</td>
<td>27.89%</td>
<td>26.18%</td>
</tr>
</tbody>
</table>

Figure 7. Examples of three semantic segmentation models with 6 classes and 150 classes.

Table 6. Performance of seg-branch and StructVPR with different segmentation models. PSP is for ResNet50-PSPNet (main paper), UPer is for ResNet50-UPerNet, and HR is for HRNetV2-C1.

<table>
<thead>
<tr>
<th>Method</th>
<th>MSLS val</th>
<th>MSLS test</th>
<th>Nordland</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R@1</td>
<td>R@5</td>
<td>R@10</td>
</tr>
<tr>
<td>SEG-UPer</td>
<td>66.2</td>
<td>80.7</td>
<td>83.8</td>
</tr>
<tr>
<td>SEG-HR</td>
<td>65.2</td>
<td>80.4</td>
<td>84.1</td>
</tr>
<tr>
<td>SEG-PSP</td>
<td>67.7</td>
<td>80.0</td>
<td>83.1</td>
</tr>
<tr>
<td>StructVPR-UPer</td>
<td>82.4</td>
<td>90.3</td>
<td>92.8</td>
</tr>
<tr>
<td>StructVPR-HR</td>
<td>81.62</td>
<td>90.41</td>
<td>91.9</td>
</tr>
<tr>
<td>StructVPR-PSP</td>
<td>83.0</td>
<td>91.0</td>
<td>92.6</td>
</tr>
</tbody>
</table>

StructVPR can achieve excellent performance without relying on semantic annotations ground truth. This is expected since the structural information we extract does not rely on completely accurate pixel-level segmentation, but more on spatial relative positional relationships. Moreover, the clustering operation in SLME also makes StructVPR less sensitive to segmentation results.

Analysis. StructVPR achieves better performance and maintains a low inference cost without re-ranking. Compared to rgb-branch, StructVPR does have more costs in training due to the large amount of training set. Nevertheless, compared to model training, annotation and group partition are not expensive and mostly one-time efforts. At last, considering the robustness of StructVPR to segmentation label map after SLME, we can seek smaller models or reduce the resolution to reduce costs of computing SEG images.

References

scale fusion of locally-global descriptors for place recognition. In CVPR, pages 14141–14152, 2021. 1, 3


