Supplementary Material for
Mapillary Street-Level Sequences: A Dataset for Lifelong Place Recognition

Frederik Warburg[†,*], Søren Hauberg[†], Manuel López-Antequera[‡], Pau Gargallo[‡],
Yubin Kuang[‡], and Javier Civera[§]

[†]Technical University of Denmark, [‡]Mapillary AB, [§]University of Zaragoza
[†]{frwa,sohau}@dtu.dk, [‡]{manuel, pau, yubin}@mapillary.com, [§]jcivera@unizar.es

1. Supplementary

In this supplementary material, we provide additional details on the validation of the Mapillary Street-Level Sequences dataset. The structure of the document is as follows: In Section 2 we provide a more elaborated explanation on the proposed sequence-to-image (seq2im) and image-to-sequence (im2seq) models. In Section 3 and 4 we show how hard negative mining works for image sequences, and we highlight the importance of choosing the positive samples wisely in the training procedure. In Section 5, we show that training on the MSLS dataset improves the generalization capabilities of deep models by an evaluation across several large place recognition datasets. We highlight the learned network attention at different layers in Section 6, and show the learned embedding space in Section 7. In Section 8, we show more image examples from the MSLS dataset. Lastly, we show in Section 9 that there exist a visual overlap between training and test set for the large Pittsburgh and Tokyo datasets, which potentially can bias the model evaluation on these datasets. This further advocates for the adoption of MSLS, which has a large geographical distance between training and test set. The MSLS dataset is available for academic research at www.mapillary.com/datasets/places

2. Seq2im and im2seq Overview

In this section, we extend our explanations on the two proposed architectures for the seq2im and im2seq tasks. The key idea is to treat each frame independently in the model, and exploit the sequential structure when retrieving the closest places.

In the seq2im case, we propose a majority voting across the sequence, i.e. select the image in the database that most images are nearest to in the query sequence.

Given a query sequence of N frames, we calculate the distance from each frame to each database image. We then look at the closest k distances for each of our N frames in our query sequence. This gives a total of $k \times N$ closest database images. We then select the most frequently occurring. The intuition is that if all the frames in a sequence are close to a database image, then we are more confident that this database image is indeed close to

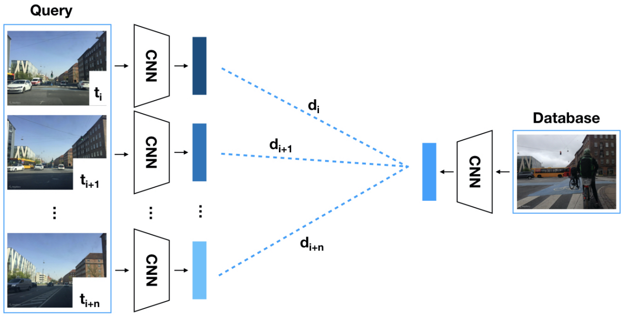


Figure 1: **seq2im:** For each query sequence, distances d_i,...,d_{i+n} are calculated for each single image in the database. We investigate both retrieving the image from the database with the minimal distance or the shortest distance to most frames in the sequence.

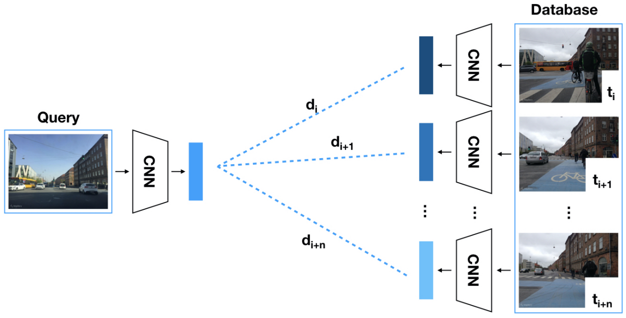


Figure 2: **im2seq:** We extract the sequence that contains the image with the shortest distance to the query image.

[*]The main part of this work was done while Frederik Warburg was an intern at Mapillary.

the query sequence. We also test the selection of the closest image in the database among all the images in the query sequence.

In the im2seq case, we select the sequence that contain the frame that is closest to query image. In practice, we calculate the distances from the query image to all the frames in the database sequences, and select the sequence that contains the nearest frame. This is visualized in Figure 2.

3. Triplet Mining of Sequences

In Figure 3 we show examples of triplets mined for sequence to sequence training. These triplets are found with the same procedure as for single-view. Also, for the sequence triplets, we see that challenging triplets are mined during training.

4. Positive Mining Strategy

It is important to note that the triplet loss is a local approximation of the recall, and that we are not guaranteed to update the embedding space everywhere when using this local approximation. We found that in the MSLS dataset, the images taken at night and images that are sideways facing are underrepresented compared to forward-looking images taken during daytime. This meant that our model performed poorly on these underrepresented image classes. Through empirical studies, we found that using a weighted sampling of underrepresented classes during training significantly improves the convergence speed and generalization across datasets. We use the curated image tags to weight these underrepresented classes by the inverse of their occurrence when sampling examples to the sub-cache. This results in a sub-cache with an equal amount of sideways/front-facing and day/night images.

5. Improved Generalization Capabilities

The increased size and diversity of our dataset can improve models' generalization capabilities. In order to show that, we compared four NetVLAD models. The first two are, a version trained on the Pittsburgh250k dataset (NetVLAD (Pitts250k)) and a version trained on the MSLS dataset (NetVLAD (MSLS)). These are trained with the same training procedure. We further include results from a model trained on MSLS with our proposed positive mining strategy (NetVLAD† (MSLS)), and lastly a model trained on MSLS with our mining strategy followed by fine-tuning on Pittsburgh250k (NetVLAD† (MSLS + Pitts250k)). We justify the improved generalization by evaluating on several popular datasets (see Table 1). Observe the smoother cross-dataset performance of the models trained on Mapillary SLS.

From Table 1, we see that the model trained on Pittsburgh seems to overfit to the Pittsburgh environment as it achieves a high recall for the Pittsburgh datasets but relatively low recall for the MSLS, RobotCar and Tokyo 24/7 datasets. As shown in Section 9, there is a visual overlap between the training and test set of the Pittsburgh and Tokyo datasets. We created a test set that we denoted as revisited-Pittsburgh250k and revisited-Tokyo 24/7, where images with visual overlap in the test set are excluded. However, the evaluation on these revisited datasets result in the same ordering of the models. This might suggest that the model overfits to the image collection procedure, image artifacts or very specific features from the Pittsburgh environment rather than the specific places near the border of the original test set.

	NetVLAD (Pitts250k)	NetVLAD (MSLS)	NetVLAD† (MSLS)	NetVLAD† (MSLS + Pitts250k)
Tokyo TM	**0.98**	**0.98**	**0.98**	**0.98**
Tokyo 24/7	0.72	0.75	0.76	**0.81**
Pitts250k	**0.91**	0.87	0.87	0.90
Pitts30k	**0.91**	0.89	0.90	**0.91**
R-Tokyo 24/7	0.71	0.74	0.75	**0.81**
R-Pitts250k	**0.91**	0.87	0.88	0.90
R-Pitts30k	**0.93**	0.91	0.92	**0.93**
RobotCar	0.74	**0.81**	**0.81**	**0.81**
MSLS	0.35	0.44	**0.47**	**0.47**

Table 1: Evaluation on test sets of popular datasets. We compare the top-5 recall of NetVLAD trained on Pittsburgh250k, NetVLAD trained on MSLS, and NetVLAD trained on MSLS and fine-tuned on Pittsburgh250k. † indicates that we apply our proposed positive mining strategy during training to sample more day/night and sideways-facing triplets.

Table 1 also shows that all the models trained on the MSLS dataset perform better on the MSLS, Tokyo 24/7 and RobotCar datasets than the model trained on Pittsburgh250k. From Table 1, we show that fine-tuning the model on the Pittsburgh250k dataset after training on the MSLS dataset results in the most general features. This model has the highest or second highest recall across all datasets. This shows that the increased size and diversity of the Mapillary SLS dataset helps the model learn a better feature representation for place recognition and improve its generalization capabilities. We note that our proposed positive mining strategy seems to improve model performance slightly across all datasets. We found experimentally that the positive mining strategy was important for good fine-tuning performance.

Lastly, note that top-5 recall is significantly lower on the MSLS test set, which is a measurement of its higher degree of difficulty. This shows that state-of-the-art place recognition models are still limited, and that the diversity and challenge of existing datasets does not sufficiently reflect real-world use-cases.

We achieved state-of-the-art performance by training our model on the MSLS dataset with our proposed positive mining strategy and fine-tuning on the Pittsburgh dataset. We found that this model had the best performance across several popular datasets for place recognition.

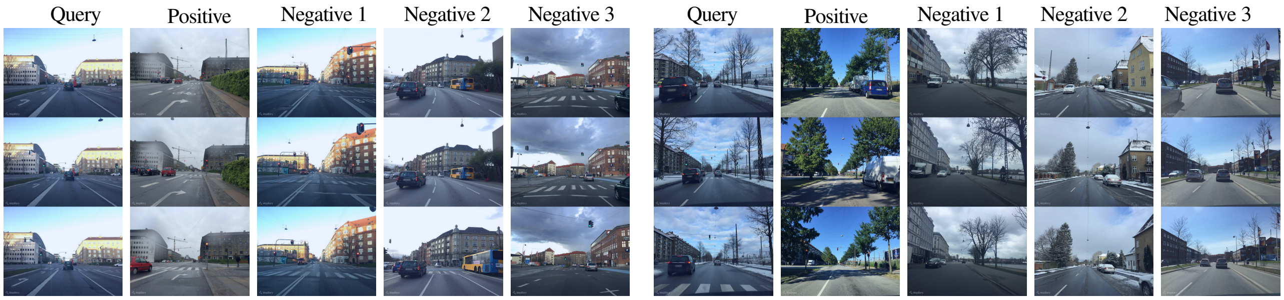
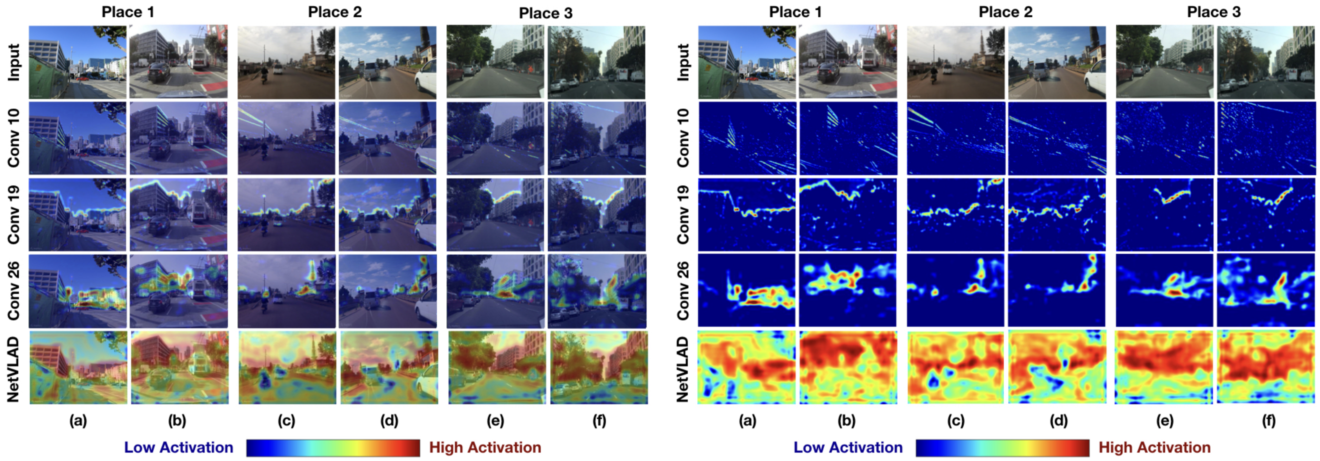


Figure 3: Sequence triplets with multiple negatives mined with the same training procedure as the in the image to image case. Note the drastic seasonal and weather changes between the query and positive images.



(a) Feature activation overlaid image

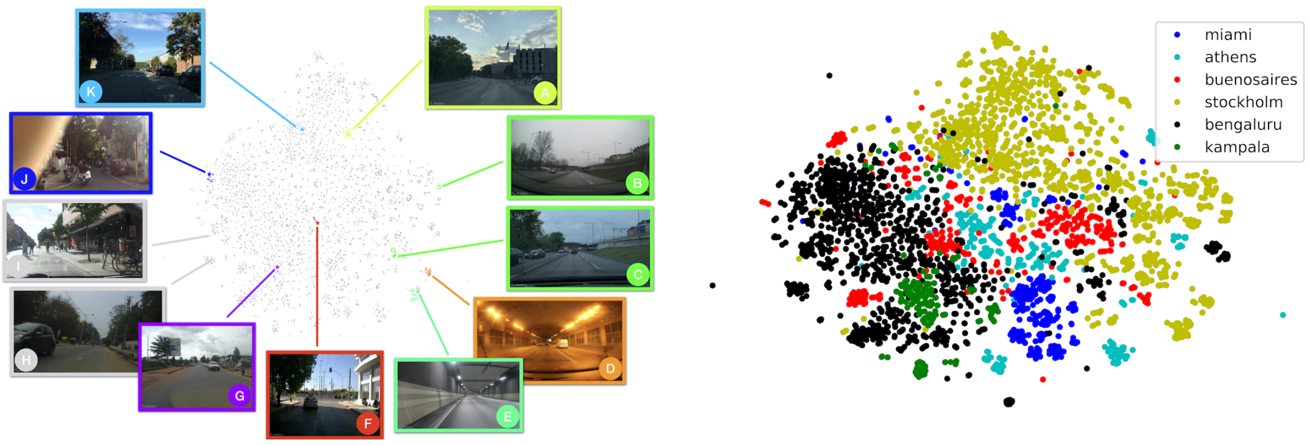
(b) Feature activation

Figure 4: The top row shows the input images. The remaining rows shows the feature activation for filters from convolutional layers 10, 19, 26 and the NetVLAD layer. In (a) these are visualized as heatmaps overlaid on top of the input image and in (b) we show the original feature maps. Blue indicates low activation and red indicates high feature activation.

6. Network Attention

To gain a better understanding on what the networks have learned, we visualize the layer activations at different depths of the NetVLAD model with VGG16 base. We visualize the activation across different input images to explain the networks' attention. We expect the model to have higher activation around stationary objects in the scene such as buildings and lower activation around changing objects such as vehicles, vegetation and roadwork. In Figure 4, we show the activation for several filters at different depths in the model.

From Figure 4 we see that the filter from convolutional layer 10 fires at longer diagonal lines, such as those found at the power cables in (c,d,f), the building facades in (a,b) and on the road paint in (a,b,e,f). These line features are simple, low-level features, which coincides with the fact that low level features are typically found in the early convolutional layers [3]. We see that the filter from convolutional layer 19 fires around the horizontal contour of the buildings. This more general feature is important for place recognition as the contour of the building is typically a non-changing feature across seasons and weather. Note that when there are small viewpoint changes such as between (e) and (f) this feature is effective for recognizing a place. Even more general features are found in convolutional layer 26. This filter activation fires on buildings. Notice that this activation seem to be invariant to viewpoint and scale changes as it fires in the same image regions for each of the image pairs. Examples are the church tower in (c,d) and the building facades in (a,b) and (e,f). Surprisingly, this filter also fires at the scooter driver's head in (c). This is a undesirable feature as we would expect the model to ignore dynamic objects. This might also suggest that using a segmentation model to filter out dynamic object might improve performance. Finally, we find the most general feature activation in the NetVLAD layer. This layer also fires on buildings and has lower activation around seasonal objects (such as trees) and dynamic objects (such as scooters, cars and buses), which is in



(a) Qualitative Examples

(b) Colored by city

Figure 5: (a) Visualization of the learned embedding space. 5,000 images from the test set are projected into a 2-D space using T-SNE. 8 randomly selected query images are highlighted with a small star and their positives are plotted in the same color as them. We highlight 11 image examples A,...,K from the embedding space. (b) Visualization of the learned embedding space color-coded by city. We see that the cities tend to cluster together and cities we would expect to be more similar, such as Kampala and Bengaluru, are also close in the embedding space.

line with the behavior that we expect the model to have learned.

7. Visualization of Learned Embedding

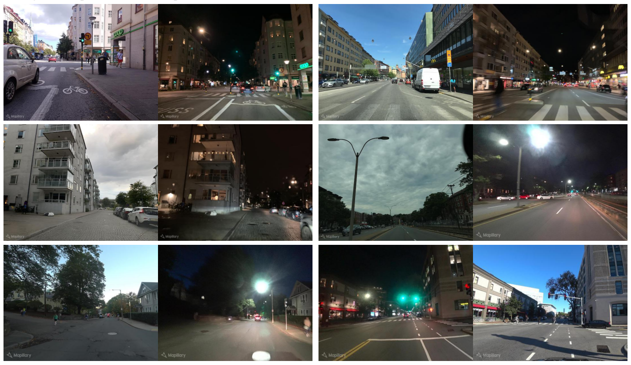
We investigate the learned embedding space to get a better understanding of what the model has learned. We randomly select 5,000 images from the test set, calculate their position in the 32,768-dimensional embedding space, and then use T-SNE to project these points into a two dimensional plane. In Figure 5a we show that the model has learned that the query examples are close to their positives. The only exception is the green cluster - highlighted with the two images B and C. We see that this scene has significant seasonal, viewpoint and weather changes - making it a challenging scenario. Furthermore, the reflection in the front window makes the example even more challenging. Below these images, we highlight two query images, D and E, captured inside a tunnel. These query images are close to their respective positives, but also close to each other in the embedding space. This intuitively makes sense as we hope that images that appear similar are close in the embedding space. Images G, H and J are captured from Bengaluru and Kampala. It seems that the model cluster these cities in the lower left corner. This is further confirmed in Figure 5b. We see that image I, which is taken in Stockholm, is also found in the lower left corner of the embedding space. This image is very blurry - suggesting that the model might have picked up the artifact that the imagery from Bengaluru and Kampala are more often in lower resolution and more blurry than the imagery from more developed cities. Also, this image contains many bicycles in the middle of the road, which the model might confuse with the many scooters often spotted in Bengaluru and Kampala.

In Figure 5b we highlight the models ability to cluster cities. The figure shows the same embedding space as visualized in Figure 5a, but color-coded by city. We see that cities we would expect to be more similar, such as Kampala and Bengaluru, are also closer in the embedding space. This clustering by cities hints that a hierarchical approach, where a classifier is trained to classify city or region followed by a place descriptor might enable a better place descriptor and a reduction of the number of model parameters. We will in the following subsection, further investigate the model performance across different parts of the world to map out the models' geographical biases.

8. More Image Examples

Place Recognition is a challenging problem due to the change in appearance that a place might experience over time. In this supplementary material, we illustrate some of the many appearance changes that are covered in the Mapillary Street-Level Sequence dataset.

Day/Night changes



Seasonal Changes & Changing Weather

Urban Environment & Dynamic Objects

Suburban Environment & Varying Viewpoints

Rural Environment

9. Visual Overlap in Pittsburgh and Tokyo 24/7 datasets

Two of the largest and most diverse datasets currently available for place recognition are the Pittsburgh250k [2] and the Tokyo [1] datasets. These datasets are collected with the same procedure from Google Street View panorama images. In this section, we highlight the undesirable visual overlap between the train/validation and test set for both of these dataset. We propose to separate the train/validation and test set by 100 meters to ensure a more fair evaluation on the test set.

In Figure 12, we show the train/validation/test division of the Pittsburgh250k and Tokyo datasets. We see that both the Pittsburgh250k and the Tokyo test sets share borders with their respective training and validation sets. It is common practice to train on both the training and the validation set once the hyper-parameters are chosen. Therefore, to obtain a fair evaluation on the test sets, we do not wish that the test sets share visual content with neither the training nor the validation set.

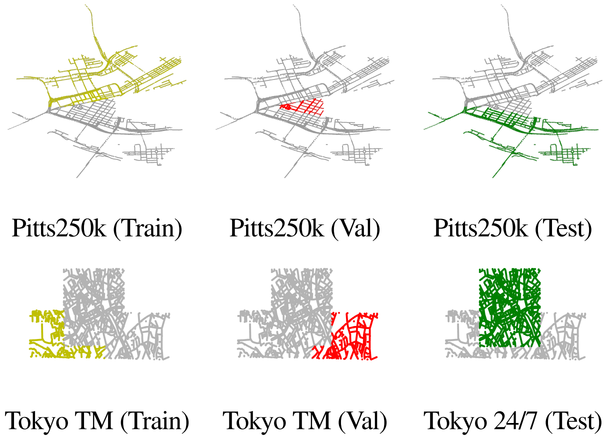


Figure 12: Shows the train/validation/test division of the Pittsburgh250K and Tokyo datasets. Note that both the training and validation sets share geographical borders with the test set for both datasets.

In Figure 13, we show the original division of the Pittsburgh250k test and train/validation set. We highlight examples along the border of the test and train/validation set that share substantial visual information. We found the closest points to be less than 1 meter apart. Note that the Pittsburgh30K dataset is a subset of the Pittsburgh250K dataset, so the findings also apply for this dataset division. Based on these findings, we propose to discard all the images in the test set that are closer than 100 m to the border. This proposed train/validation/test division is showed in Figure 14 for the Pittsburgh dataset. Note that the images at the border do not share visual information with the proposed data division. For fair visualizations, these images are chosen as the closest images that have the same view-direction similar to in Figure 13. By discarding images close to the border, we reduce

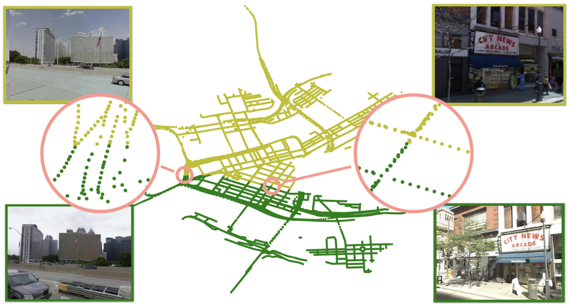


Figure 13: Shows a clear visual overlap between Pittsburgh test and train/validation dataset. The two zooms highlight that there is not a geographical overlap, but a clear visual overlap between the test and train/validation sets.

the Pittsburgh250k test set by 6.792 (7.4%) images and the Pittsburgh30k test set by 2.760 (16.4%) images, however, we ensure a fair evaluation of models trained on the Pittsburgh dataset.

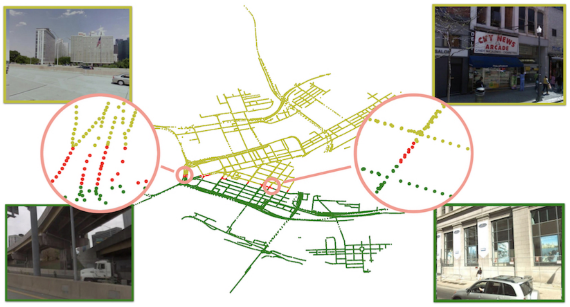


Figure 14: Shows the revisited Pittsburgh dataset. The yellow points are the original train/validation set, the red points are the images discarded from the test set, and the green points are the remaining images in the test set. The two zooms highlight that there is not a visual overlap between the revisited test and train/validation sets.

We implemented the same procedure for the Tokyo dataset. Figure 15 highlights the new train/validation/test division for this dataset as well as images for before the proposed split. The revisited-Tokyo 24/7 test is reduced by 9.342 (12.2%) images.

The proposed separation between the test and train/validation sets result in a more fair evaluation on the test set. However, as these datasets are collected from a rather small geographical region it is still unsure how models generalize to other regions, cities, suburban environments or more rural areas. Furthermore, these datasets are curated with image-to-image methods in mind, making it difficult to exploit the sequential information in image sequences.

The sequential structure of the Mapillary Street-Level Dataset, its large geographical coverage and temporal coverage makes it a valuable addition to the existing data corpus for Place Recognition.

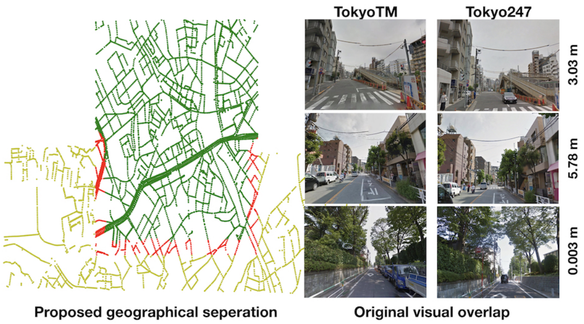


Figure 15: Shows the proposed train/validation test division of the Tokyo dataset and three image pairs that shows the significant visual overlap between the train/validation set (Tokyo TM) and the test set (Tokyo 24/7) in the original Tokyo dataset. Note that the highlighted places are all less than 6 meters apart and that the same place is usually defined within a 25 meters radius. Thus, the same places are part of both train/validation and test set.

References

[1] 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*, pages 1808–1817, 2015. 5

[2] A. Torii, J. Sivic, M. Okutomi, and T. Pajdla. Visual place recognition with repetitive structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(11):2346–2359, Nov 2015. doi: 10.1109/TPAMI.2015.2409868. 5

[3] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. *CoRR*, abs/1311.2901, 2013. URL http://arxiv.org/abs/1311.2901. 3