Supplementary Material: Predicting Visual Overlap of Images Through Interpretable Non-Metric Box Embeddings

1 Datasets Discussion

Since our proposed world-space measure requires depth information during training, we conduct our experiments on the 7Scenes [5] and MegaDepth [3] datasets. Unfortunately, other datasets used for localization benchmarks such as Aachen [4], Cambridge Landmarks [2], CMU-Seasons [1], *etc.* do not provide the depth maps that our world-space measure requires. It may seem that InLoc [6] is a good indoor localization dataset for our method, as it provides RGBD frames captured with a laser scanner. However, InLoc often has a single panoramic 3D scan per room, which prohibits generation of pairs of images with meaningful visual surface overlaps for training. Further, the panoramic images are cropped at exactly 30° intervals which also introduces bias to the pairwise visual overlap values.

2 Learning Asymmetric Measures

Adding to the results in Table 1 of the main paper, we report the ability of vector embeddings to learn the asymmetric normalized surface overlap in Table 1. Both here and in the main paper we evaluate vector embeddings on $\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y})$, but while we trained on $\mathsf{NSO}^{\mathsf{sym}}(\mathbf{x}, \mathbf{y})$ in the main paper, we train vectors to learn $\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y})$ in this experiment. Because vector embeddings have to predict the same value for both $\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y})$ and $\mathsf{NSO}(\mathbf{y} \mapsto \mathbf{x})$ we expect them to converge to predicting the average of both measures, because — at least analytically — this would lead to the smallest training error.

[⋆] Project page: www.github.com/nianticlabs/image-box-overlap

	$\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y})$ with vectors		
	L_1-Norm	RMSE	Acc.$<$ 0.1
Notre-Dame	0.251	0.256	60.1%
Big Ben	0.440	0.364	25.2%
Venice	0.201	0.216	69.5%
Florence	0.163	0.174	72.2%

Table 1. Evaluation of vector embeddings, trained on normalized surface overlap. We measure the discrepancy between the predicted and ground-truth asymmetric overlaps on the test set.

The low accuracy on Big Ben in Table 1 could result form properties of the Big Ben scene. Most random pairs of images have have non-zero overlap (it should be noted that the underlying ground-truth SfM model assumes that the Big Ben tower only has two sides due to symmetry). This is not the case for the other scenes, where many image pairs have almost zero enclosure **and** concentration, such that prediction of a symmetric overlap does not cause high errors. Note also that we did not balance the training sets.

3 Interpretable Queries

Further to Section 4.2 in our main work, we demonstrate the interpretability of our method on additional *test* queries from the MegaDepth dataset. We show two different types of figures:

– Interpretability plots: For a query image from the test set what do the predicted enclosure and concentration of retrievals from the **training** set images tell us about their relationship to one-another?
– Generalization plots: Does our embedding generalize to the images from the **test** set and are pairwise relationship interpretations valid?

3.1 Interpretability Plots

These plots, Figures 5, 6, 7, 8, 9, 10, 11, 12 and 13, illustrate the interpretability that we gain when using box embeddings trained with normalized visual overlap as the world-space measure. The key takeaway is that we can qualitatively observe the relationship between the query and each of the retrieved images. The relationships can be grouped into four categories: Given a query image, a retrieval can be a:

– zoom-out
– crop-out or oblique-out
– close-up
– clone-like

We show three different examples for the scenes $Venice$ and $Florence$ and two examples for $BigBen$ and $NotreDame$ additionally to Figure 5 in the main work. For each figure a single query and up to 36 retrieved images from the training set are shown. They are placed into one of six buckets according to the **predicted** normalized box overlap $\mathsf{NBO}(\mathbf{b_x} \mapsto \mathbf{b_y})$ and $\mathsf{NBO}(\mathbf{b_y} \mapsto \mathbf{b_x})$. The vertical axis describes the enclosure, so $\mathsf{NBO}(\mathbf{b_x} \mapsto \mathbf{b_y})$, or "*how much surface from the query image is visible in the retrieved image*". The horizontal axis describes the concentration $\mathsf{NBO}(\mathbf{b_y} \mapsto \mathbf{b_x})$, in other words "*how much surface of the retrieved image is visible in the query image*".

The numbers below each image are enclosure and concentration estimated with box representations as well as ground-truth values estimated with semi-dense depth maps.

3.2 Generalization Plots

The interpretability plots retrieve images from the training set of our box embeddings. Next, we demonstrate qualitatively that the learned representations generalize to the images in the test set. Here the test set are images in MegaDepth that do not have dense depth information. Hence, we can only report qualitative results for this larger test set.

So, *we now retrieve images from the test set* using a random query image from the test set. Here, we plot retrieved images on a 2D grid, using enclosure and concentration of the retrieved image as 2D coordinate, where the x-coordinate denotes the concentration, and the y-coordinate denotes the enclosure.

These plots, Figures 14, 15, 16, 17 and 18, also provide interpretability, as one can observe different clusters of images as zoom-outs, crop-outs, close-ups and clones of the query in similar “quadrants” as Interpretability plots. We also show some result on 7Scenes in Figures 19 and 20.

4 Predicting Relative Scale

Further to Section 4.4 in the main paper we evaluate the ability of our method to predict *useful* relative scale. We show:

- Relative scale plots: Given pairs of images from the test set and their box representations, can we estimate their relative scale difference? Relative scale plots are further qualitative examples similar to Figure 1 and Figure 7 in the main paper
- the computational efficiency of pre-scaling.
- a comparison between rescaling based on our pipeline and a homography based approach

We illustrate that we can estimate geometric relationships between two images from the test set using our box embeddings. For two images from the test set we can estimate the relative scale of the first image in the second image.

Hence, we plot two test images and a rescaled version of the first image such that any geometric verification between the two images is now easier to do due to matching scale, Figures 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 and 32.

This relative scale estimate is relatively accurate if the images have zoom-in/zoom-out relationship. If the images are in crop-out/oblique-out to one-another, then the rescaling is not necessarily going to make matching easier. So, image pairs that seem to be failure cases in terms of estimated relative scale often have low enclosure value (<80%), which means that these image pairs can be detected and treated accordingly. To demonstrate that this filtering approach is effective, we show failure cases with an enclosure of at least 80%.

4.1 Implementation Details and Efficiency

To measure the efficiency of pre-scaling images according to their relative scale, we measure the processing times of two different pipelines that take a query and its retrieved nearest neighbor as inputs. We assume that the box overlap of two images needs to be computed for the retrieval of the nearest neighbor and is therefore known. We compare (i) using a PnP + RANSAC pose estimation pipeline based on SIFT features with 3 layers in each octave and (ii) computing the relative scale based on predicted box overlap, rescaling one image of a pair accordingly and using a PnP + RANSAC pose estimation pipeline based on SIFT features with only 1 layer in each octave. Both pipelines include the detection and computation of SIFT features with OpenCV in Python using `cv2.xfeatures2d.SIFT_create(contrastThreshold=0.03,sigma=1.2)` and `cv2.solvePnPRansac(flags=cv2.SOLVEPNP_P3P)` to solve for pose. Averaged over 100 runs the pipeline with three layers per octave needs 2.4 seconds, while the same pipeline with one layer per octave takes 1.2 seconds. The computation of the relative scale and resizing with OpenCV takes another 0.008 seconds on average. This means, that per image pair our pipeline saves more than a second, or 49% compared to a strictly feature-based pipeline. We used an Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz CPU on Ubuntu.

4.2 Homography-based Scale Estimation

Lastly, we compare our NSO-based scale estimate to a homography based scale estimation in Figure 33. For the images from Figure 7 in the main paper we estimate the relative scale with an OpenCV pipeline. We extract SIFT features with `cv2.xfeatures2d.SIFT_create()` and estimate a homography with `cv2.findHomography(method=cv2.RANSAC,ransacReprojThreshold=5.0)`.

5 Further Localization Evaluations

In this section we provide more detail on our localization pipeline as well as further evaluations of different embeddings for the localization task expanding on Section 4.3 in the main paper.

The pipeline used for localization consists of the following steps using OpenCV in Python:

- Retrieve nearest neighbor according to retrieval function (see Tables 2, 3)
- Extract SIFT features with `cv2.xfeatures2d.SIFT_create` and default parameters except `contrastThreshold=0.03` and `sigma=1.2`
- Use Lowe Ratio Test with threshold 0.9
- Infer 3D points from 2D matches in database image using corresponding depth maps
- Get pose between 2D matches in query and 3D matches in retrieval using `cv2.solvePnPRansac` with `flags=cv2.SOLVEPNP_P3P` and guess an initial pose as the pose of the retrieved nearest neighbor with `useExtrinsicGuess=1`
- If matching fails use pose of nearest neighbor as prediction

5.1 7-Scenes

Figure 1 shows sorted rotation errors for each scene, similarly to Figure 6 in the main paper, for different embeddings. Again, rotation error is computed per query, when matched against 10-th and 30-th nearest neighbor from the training set (gallery).

When solving for the pose, retrieving 10-th nearest neighbor for matching seems to be sufficient to estimate good pose for most of the scenes. However, even in this setting we can see that Frustum overlap is under-performing compared to embeddings trained with surface overlap. Only in Pumpkin scene with 10-th retrieved image, the frustum overlap is marginally better. Between surface overlap-based embeddings the performance is quite comparable. There are no systematic improvements nor deteriorations with relative scale correction as captured scenes are all rooms with limited scale variation.

5.2 Megadepth

Figure 3 shows evaluation results for Venice and Florence scenes to complement Figure 6 of the main paper.

In Table 2 we report median error translation and rotation error of estimated pose for 100 images of the test set (that have corresponding depth maps) similar to Table 2 in the main paper. The Megadepth scenes are not metric, so the scale factor of translation errors is not known. Furthermore, all these errors are relatively low, corresponding to accurate pose, so it is difficult to draw conclusions from these results.

Hence, we also evaluated different embeddings with a larger test set which consists of images in Megadepth that do not have depth maps. This results in 3165, 1931, 2255 and 1159 test set images for Big Ben, Notre-Dame, Venice and Florence, respectively. Table 3 show median errors for the larger test set. As can

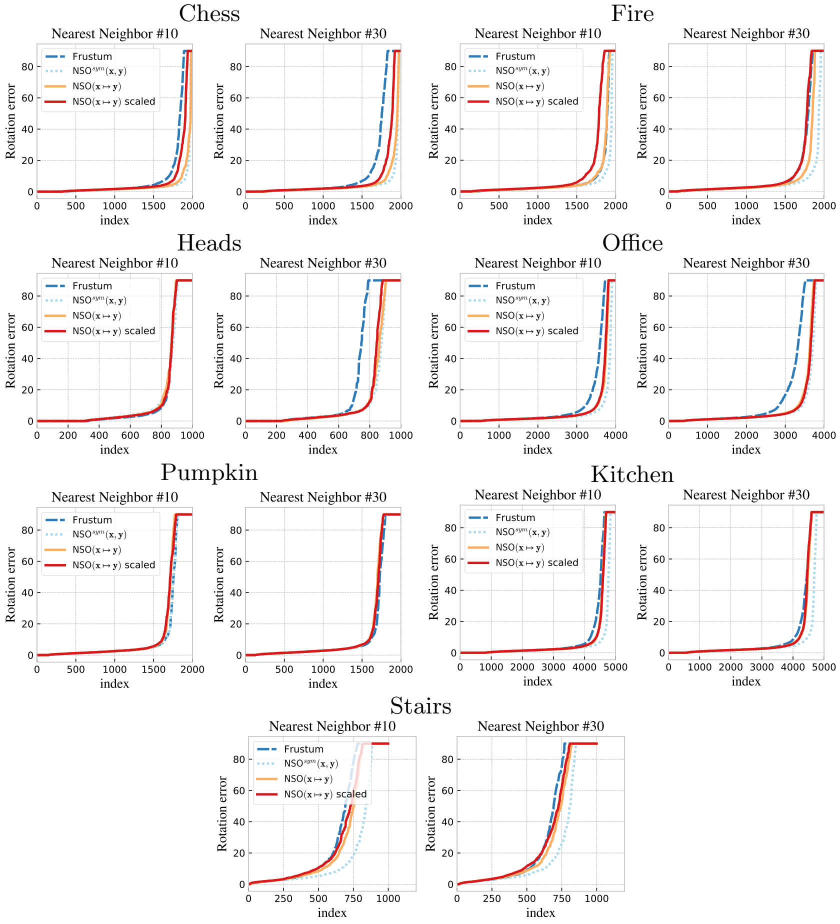


Fig. 1. 7-Scenes pose rotation error. Each plot shows (sorted) rotation error (capped at 90°) when each test image is matched against 10-th and 30-th closest retrieved image for pose estimation. As we can see, normalized surface overlap methods are superior to Frustum overlap.

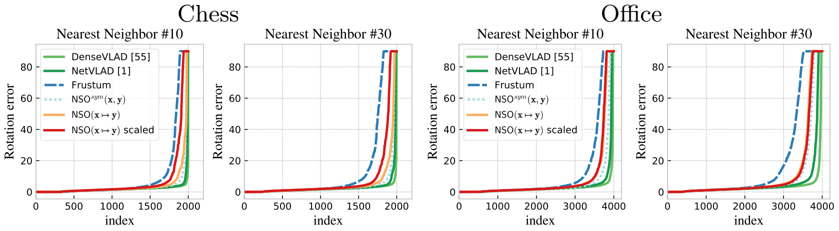


Fig. 2. 7-Scenes pose rotation error, comparison with DenseVLAD and NetVLAD Each plot shows (sorted) rotation error (capped at 90°) when each test image is matched against 10-th and 30-th closest retrieved image for pose estimation.

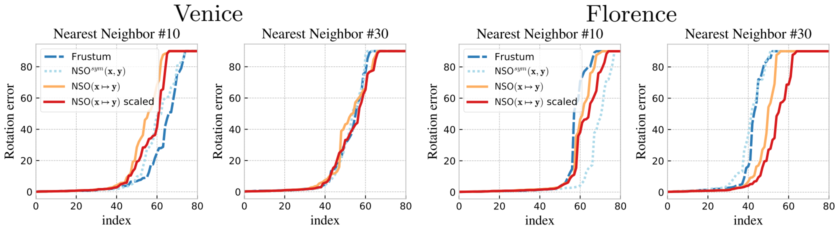


Fig. 3. Additional results to Fig. 6 on MegaDepth. Each plot shows (sorted) rotation error (capped at 90°) when each test image is matched against 10-th and 30-th closest retrieved image for pose estimation. As we can see, box embeddings with surface overlap measure tend to outperform alternatives, especially when rescaling images according to estimated relative scale and for the 30-th neighbor. Results evaluated for 100 images from the test set.

Training $\mathcal{L}$	Box $\mathsf{NSO}(\mathbf{x}\mapsto\mathbf{y})$	Box Scaled $\mathsf{NSO}(\mathbf{x}\mapsto\mathbf{y})$	Vector $\mathsf{NSO}^{sym}(\mathbf{x},\mathbf{y})$	Vector Frustum
Retrieval func.	$\mathsf{NBO}(\mathbf{b_x}\mapsto\mathbf{b_y})+$ $\mathsf{NBO}(\mathbf{b_y}\mapsto\mathbf{b_x})$	$\mathsf{NBO}(\mathbf{b_x}\mapsto\mathbf{b_y})+$ $\mathsf{NBO}(\mathbf{b_y}\mapsto\mathbf{b_x})$	Eucl. dist.	Eucl. dist.
Notre-Dame	.038, 0.79°	.038, 0.87°	.048, 1.02°	.047, 1.05°
Big Ben	.067, 0.87°	.067, 0.87°	.070, 0.87°	.096, 0.83°
Venice	.096, 1.01°	.098, 1.23°	.102, 0.87°	.085, 0.91°
Florence	.081, 1.08°	.079, 1.10°	.072, 1.03°	.048, 0.68°

Table 2. Comparison of rotation and translation errors on the MegaDepth dataset, where boxes learn surface overlap asymmetrically while vectors are trained symmetrically. The first entry of each cell denotes the translation error up to scale, the second entry is the rotation error in degrees. Results evaluated for 100 images from the test set.

be seen, correcting for scale using Box embeddings is superior to alternatives on 3 scenes.

Similarly, (sorted) rotation error evaluated for the larger test set could seen in Figure 4. Here the error is computed against 1st nearest neighbor retrieval. These plots indicate a similar conclusion. The surface overlap based embeddings are outperformed by Frustum overlap embedding for Florence scene. Florence scene has images that capture a large area with complex narrow streets, however the training set consists of only 1471 images. We suspect that our CNNs need more training data to learn generalizable surface overlaps.

	Box	Box Scaled	Vector	Vector
Training $\mathcal{L}$	$\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y})$	$\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y})$	$\mathsf{NSO}^{sym}(\mathbf{x}, \mathbf{y})$	Frustum
Retrieval func.	$\mathsf{NBO}(\mathbf{b_x} \mapsto \mathbf{b_y})+$ $\mathsf{NBO}(\mathbf{b_y} \mapsto \mathbf{b_x})$	$\mathsf{NBO}(\mathbf{b_x} \mapsto \mathbf{b_y})+$ $\mathsf{NBO}(\mathbf{b_y} \mapsto \mathbf{b_x})$	Eucl. dist.	Eucl. dist.
Notre-Dame	0.84, 15.1°	0.81, 13.8°	**0.27**, **5.3°**	0.98, 25.2°
Big Ben	2.91, 58.0°	**2.85**, **53.9°**	3.21, 62.4°	3.30, 69.8°
Venice	3.20, 68.9°	3.24, 58.6°	**1.83**,**33.8°**	2.70, 65.6°
Florence	1.44, 35.0°	1.33, 31.6°	**0.37**, **5.3°**	0.75, 10.6°

Table 3. Comparison of rotation and translation errors on the MegaDepth dataset for test set without depth images, where boxes learn surface overlap asymmetrically while vectors are trained symmetrically. The first entry of each cell denotes the translation error up to scale, the second entry is the rotation error in degrees. Total number of images in these test sets (in Figure order): 3165, 1931, 2255 and 1159.

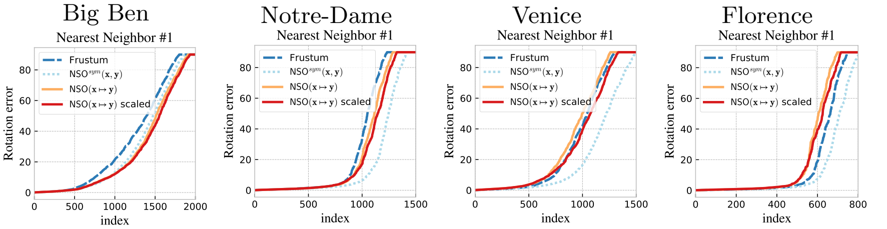


Fig. 4. Results on images without depth (MegaDepth) Each plot shows (sorted) rotation error (capped at 90°) when each test image is matched against the closest retrieved image for pose estimation. As we can see, box embeddings with surface overlap measure tend to outperform alternatives, especially when rescaling images according to estimated relative scale. Total number of images in these test sets (in Figure order): 3165, 1931, 2255 and 1159.

Please note that we had to make an assumption that could not be verified with the authors of MegaDepth at the time of submission. Though the camera intrinsics are provided for the original sized images, they must be adjusted when working with the resized depth maps. The authors note that the image size of the depths is slightly different (a few pixels) due to their SfM pipeline. However, it is not clear if the depths are rescaled or cropped or both, and our observations did not provide a definite answer to this question. In our experiments we assume that the resized images in the depth dataset of MegaDepth are rescaled versions of the original images in the SfM dataset. When estimating pose between query and retrieval with our Pnp pipeline we adjust the camera intrinsics according to this assumption.

6 Generalizability to Different Datasets

To investigate the ability of our method to generalize to new scenes we provide some qualitative evaluations on the Cambridge Landmarks dataset. We use our model trained on the Notre Dame scene of MegaDepth and retrieve nearest neighbors of queries from King's College in Cambridge. Both queries and data base images have never been seen during training. Figure 34 shows some randomly picked examples. Note that we report qualitative results only, as the Cambridge dataset does not provide per image 3D points limiting the applicability of our pipeline requiring 2D-3D point correspondences between query and retrieved image for PnP pose estimation. Nonetheless, it is apparent that our method is able to generalize to an unseen dataset retrieving useful nearest neighbors.

7 Box Dimensionality Ablation Study

We report the impact of the box dimensionality on the ability to predict normalized surface overlap. We state the root mean squared error (RMSE) on MegaDepth's Big Ben scene in Table 4.

	8D	16D	32D	64D
After 20 epochs	0.11/0.13/83%	0.11/0.14/83%	0.11/0.13/82%*	0.13/0.14/81%
After 31 epochs	0.11/0.14/83%	0.11/0.14/83%	0.11/**0.12**/83%	0.12/0.14/82%

* For this ablation study we trained all models using the same random seed including the 32D model, which is why this result is slightly different from the result reported in the main paper.

Table 4. L1-Norm/RMSE/Accuracy<0.1 between predicted and ground truth $\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y})$ on 1,000 random image pairs from the test set for different box dimensions.

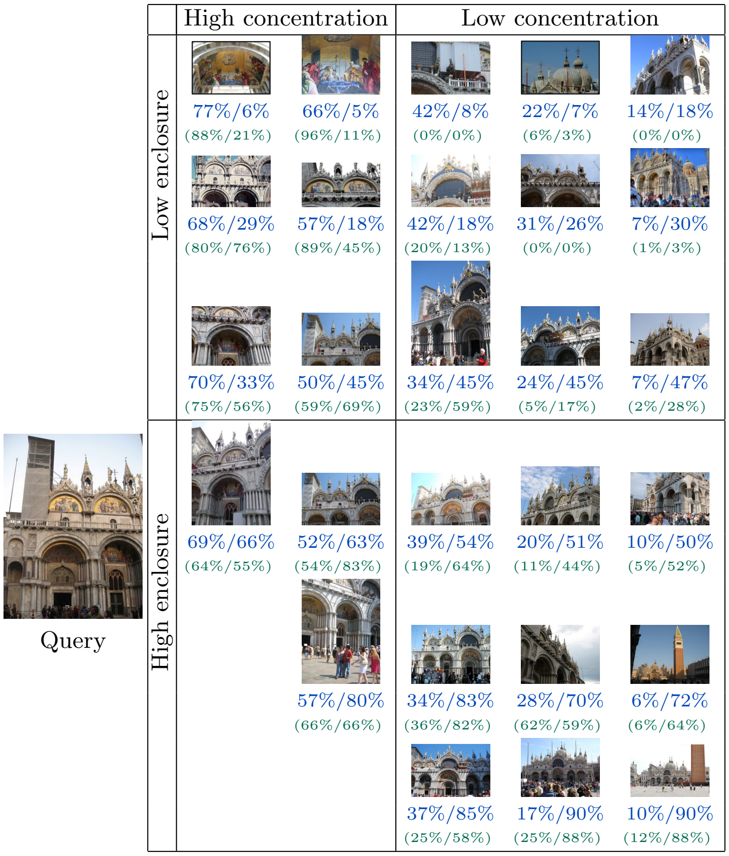


Fig. 5. Interpretability plot: Venice No. 1. Results of predicted and ground-truth enclosure and concentration relative to the query image on the left. The numbers below each image indicate the predicted and ground-truth concentration/enclosure. It can be observed that the images in the upper left quadrant are close-ups of the query. The images in the lower left quadrant are clones of the query. The retrieved clones preserve the normals of the surfaces in the query. Images in the lower right quadrant are zoom-outs of the scene in the query. The images in the upper right corner are mostly oblique-outs and show the scene in the query from different angles. *This caption applies to all Interpretability plots unless otherwise stated.*

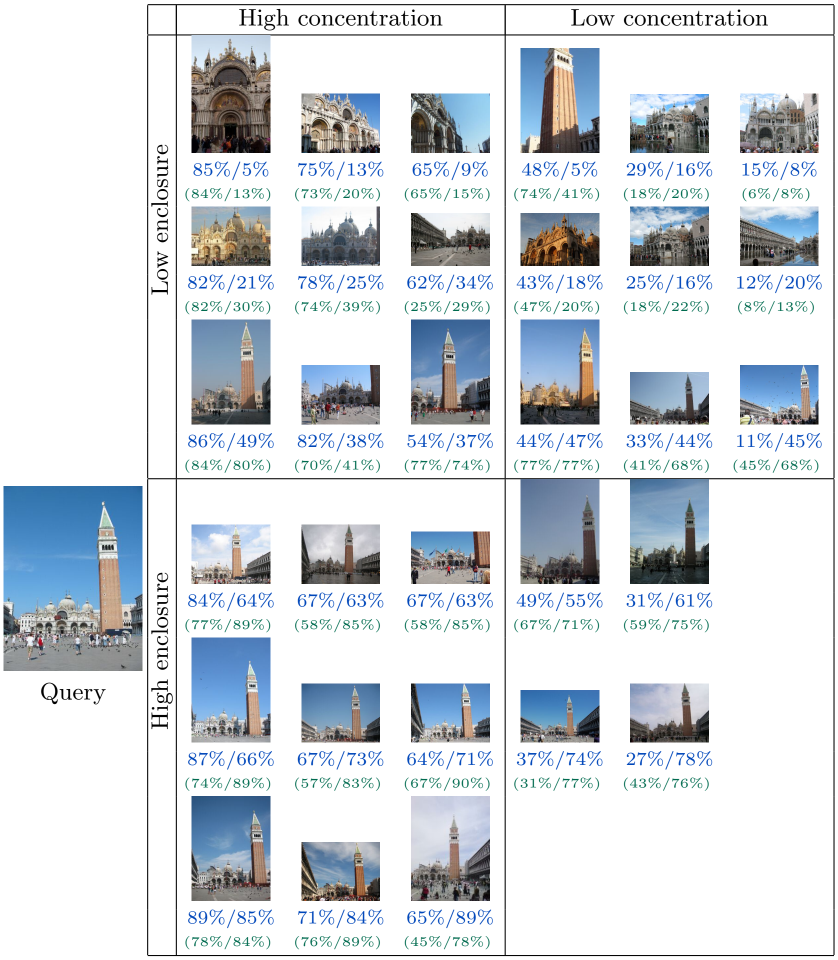


Fig. 6. Interpretability plot: Venice No. 2.

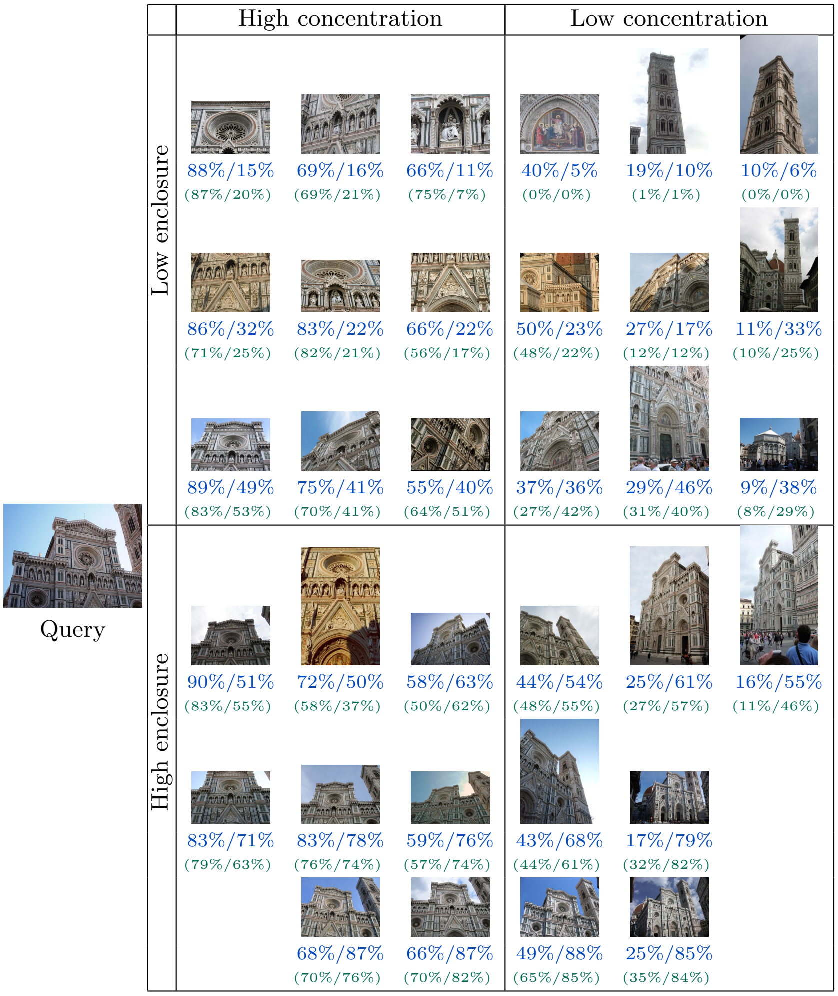


Fig. 7. Interpretability plot: Florence No. 1.

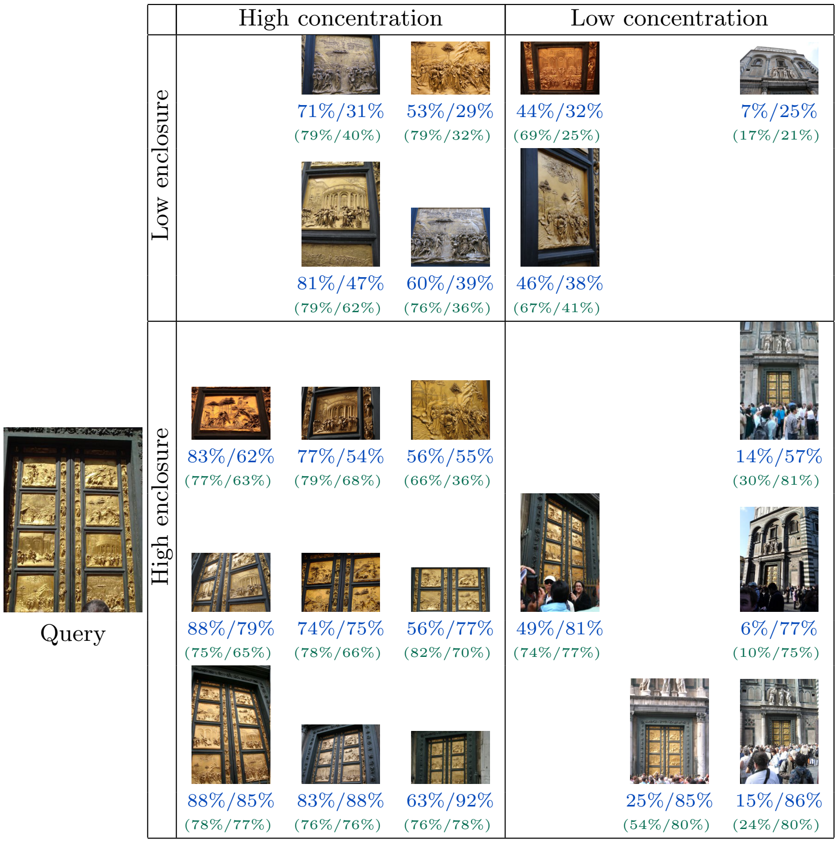


Fig. 8. Interpretability plot: Florence No. 2.

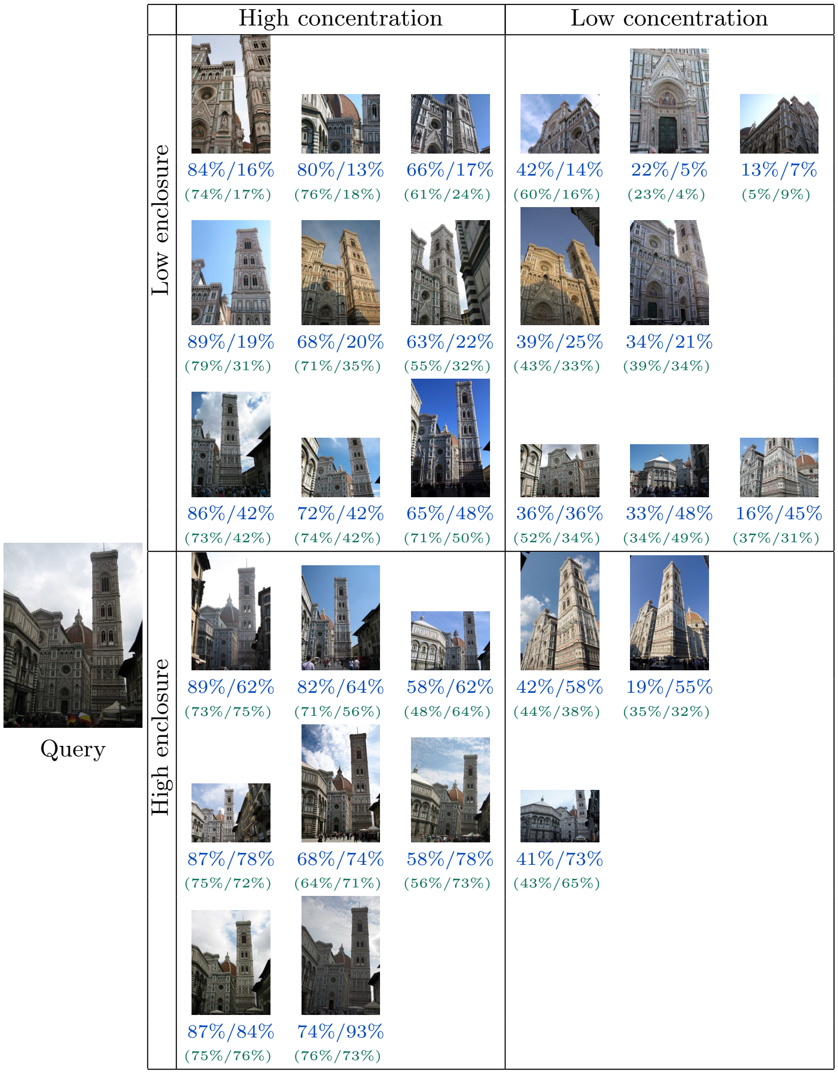


Fig. 9. Interpretability plot: Florence No. 3.

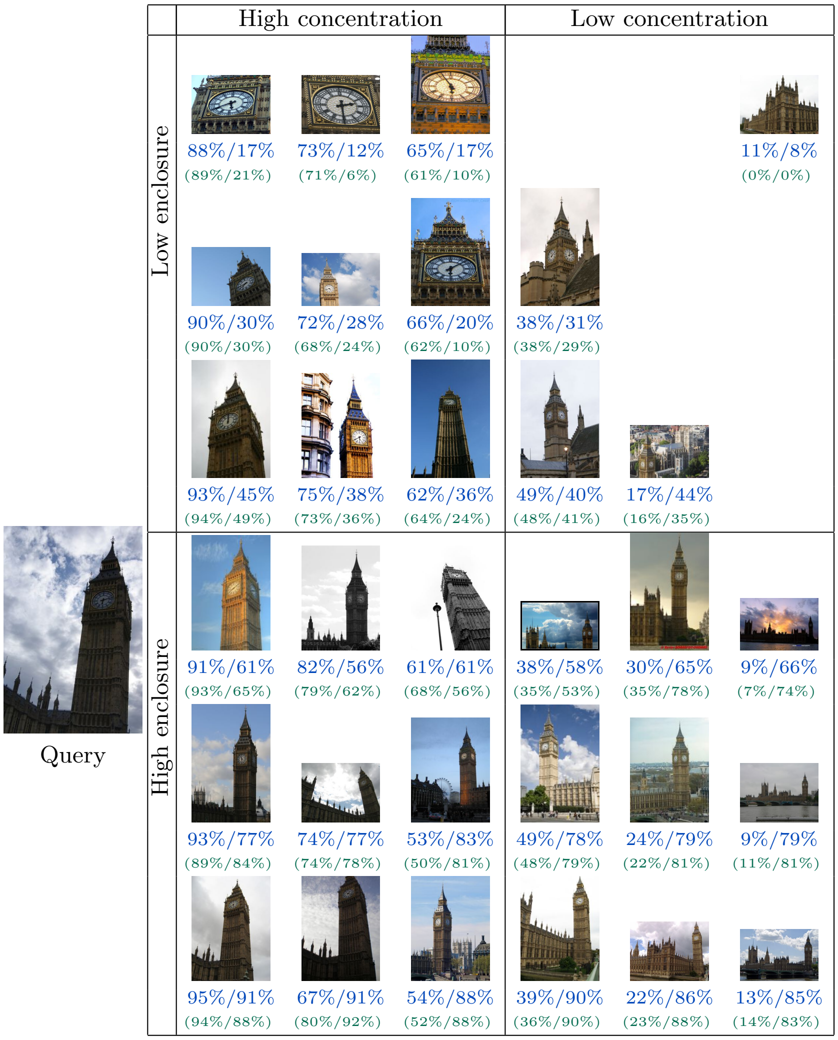


Fig. 10. Interpretability plot: Big Ben No. 1.

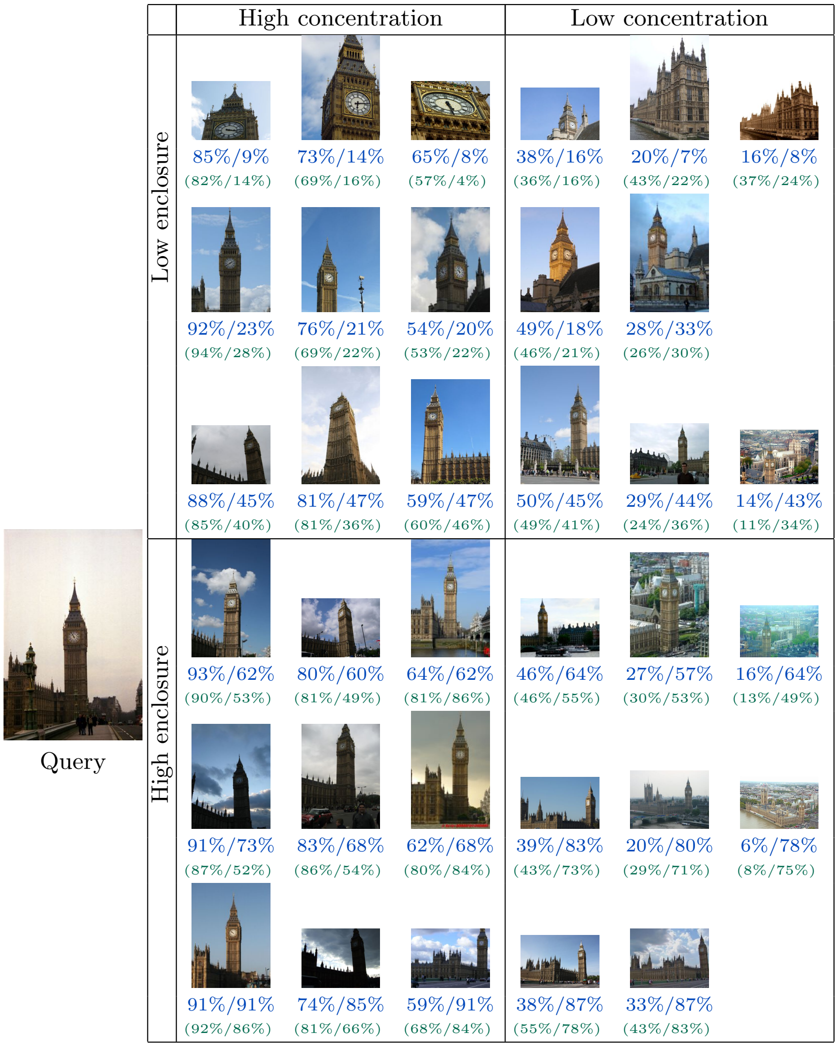


Fig. 11. Interpretability plot: Big Ben No. 2.

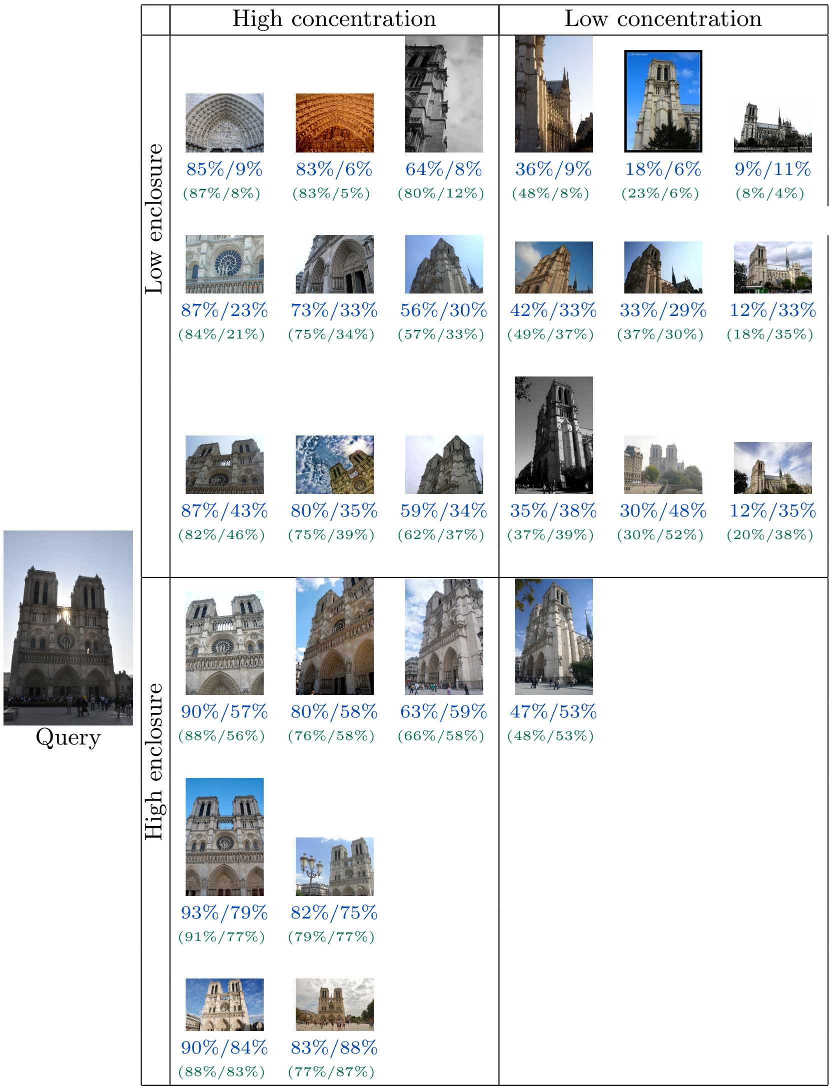


Fig. 12. Interpretability plot: Notre Dame No. 1.

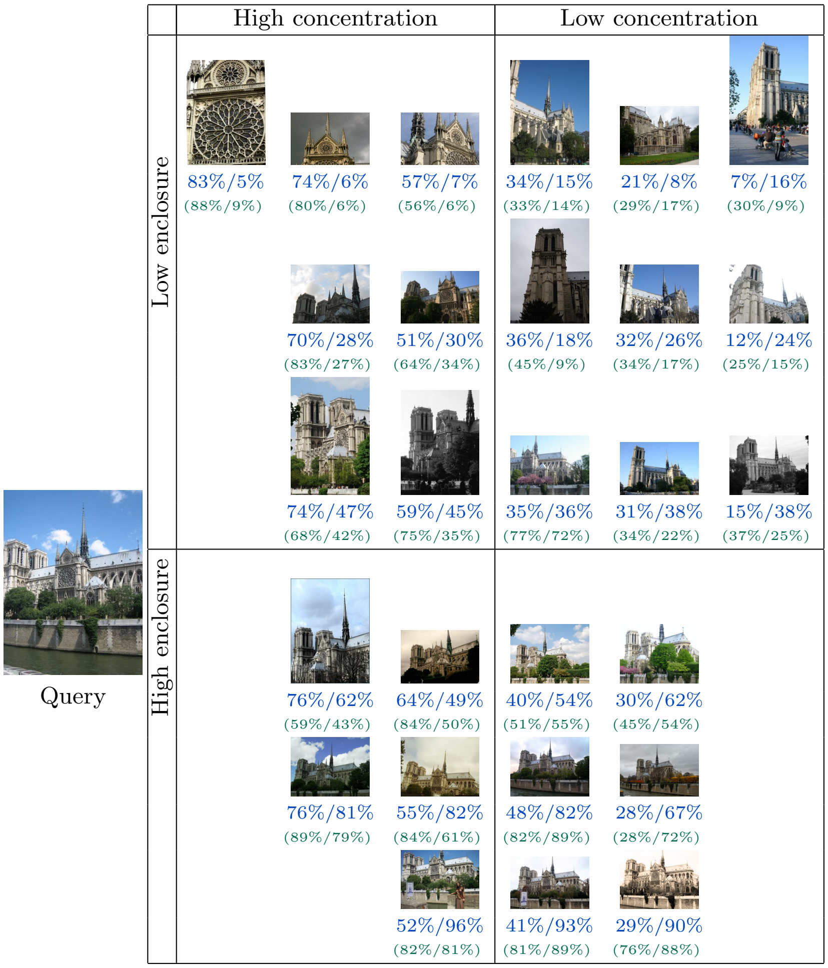


Fig. 13. Interpretability plot: Notre Dame No. 2.

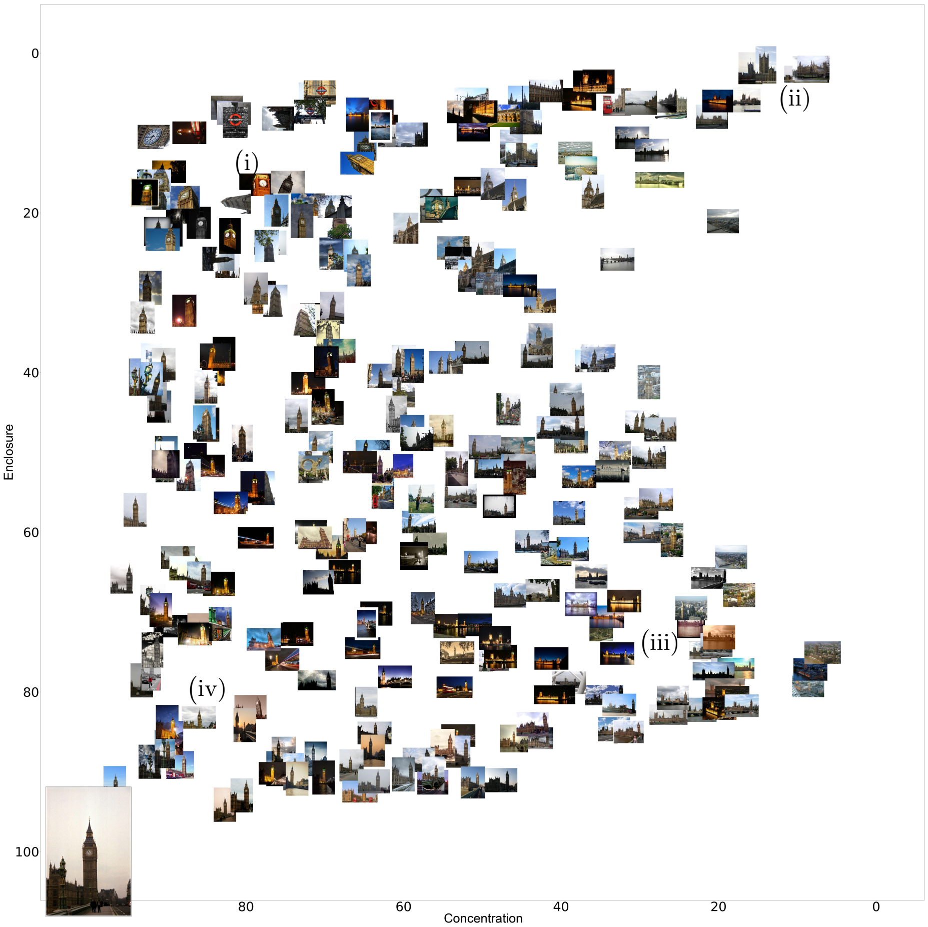


Fig. 14. Generalization plot: Big Ben. In this caption we expand on details and explanations of Fig.5 (Right) from the main paper. We show a query image from the test set (lower left corner) and the concentration and enclosure between randomly sampled test images from the MegaDepth SfM data set for which no depth maps are provided. The query image shows Big Ben from the view of the Westminster Bridge. (i) It can be observed that close-ups on the tower clock are clustered around the coordinates (80,15) which is consistent with our terminology of retrievals with high concentration and low enclosure. (ii) The images in the upper right corner show the waterfront side of Westminster Palace. These are crop-outs of the query image. In fact, the tower in the lower left corner of the query is one of the two towers that mark the corners of the water-front side of the palace. The retrievals in the upper right quadrant of the cluster therefore extends the view of the query. (iii) The images in the lower right area of the cluster clearly show zoom outs, with the pointy bell tower visible in all images. (iv) Lastly, one can observe that the images in the $clone - like$ category are in fact similar views on Big Ben. Note that some of the retrievals are rotated images and sometimes cause outlier predictions. *This caption applies to all generalization plots unless otherwise stated.*

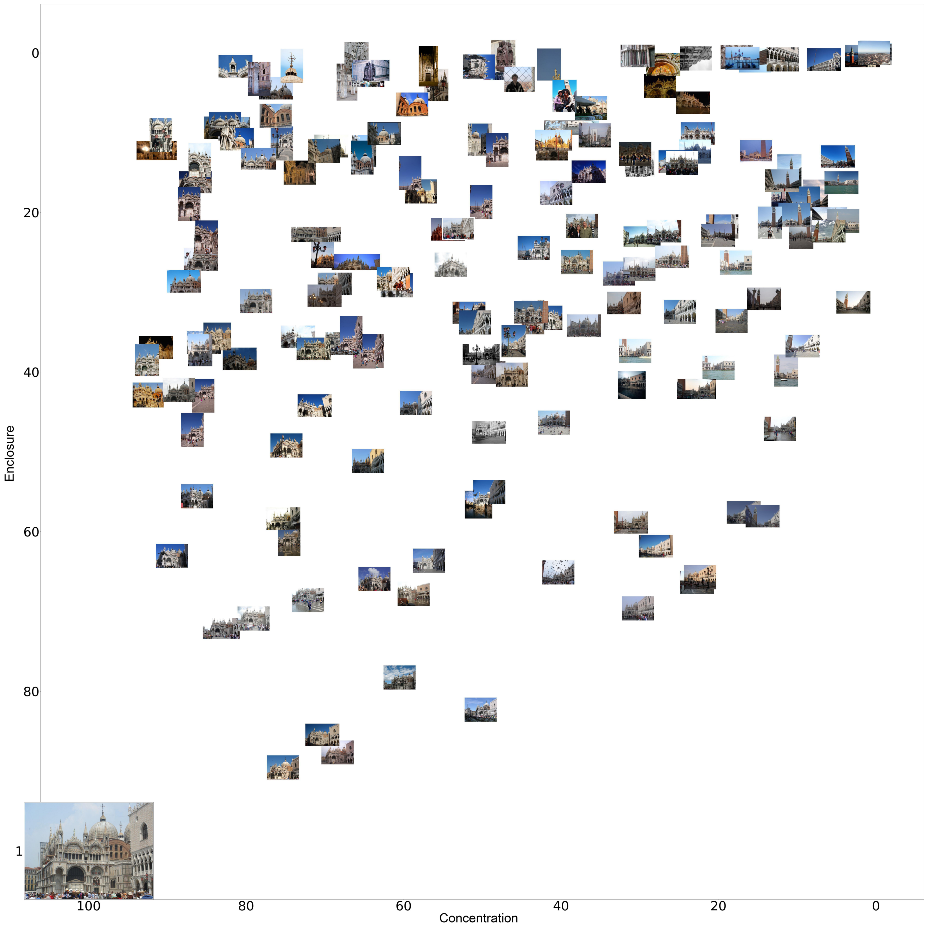


Fig. 15. Generalization plot: Venice. We show a query image from the test set (lower left corner) and the concentration and enclosure between randomly sampled test images from the MegaDepth SfM dataset for which no depth maps are provided. The query image shows the side view on Saint Mark's Basilica. One can observe the front of the Basilica from a very oblique angle in the left-most fifth of the image.
The images in the left upper corner show images with high concentration and low enclosure. According to our classification these are close-ups. Especially around the coordinates (80, 10) one can clearly observe zoomed in views on the side of the Basilica. The right upper quarter of the cluster consist of the crop-outs and oblique-outs. Note the images around the coordinate (10,25). These are mostly front views on the Basilica, and correspond to the left-most part of the query image–from a very different angle. Lastly, observe that images in the left lower corner are similar to the query, and images around (20,60) are zoom-outs.

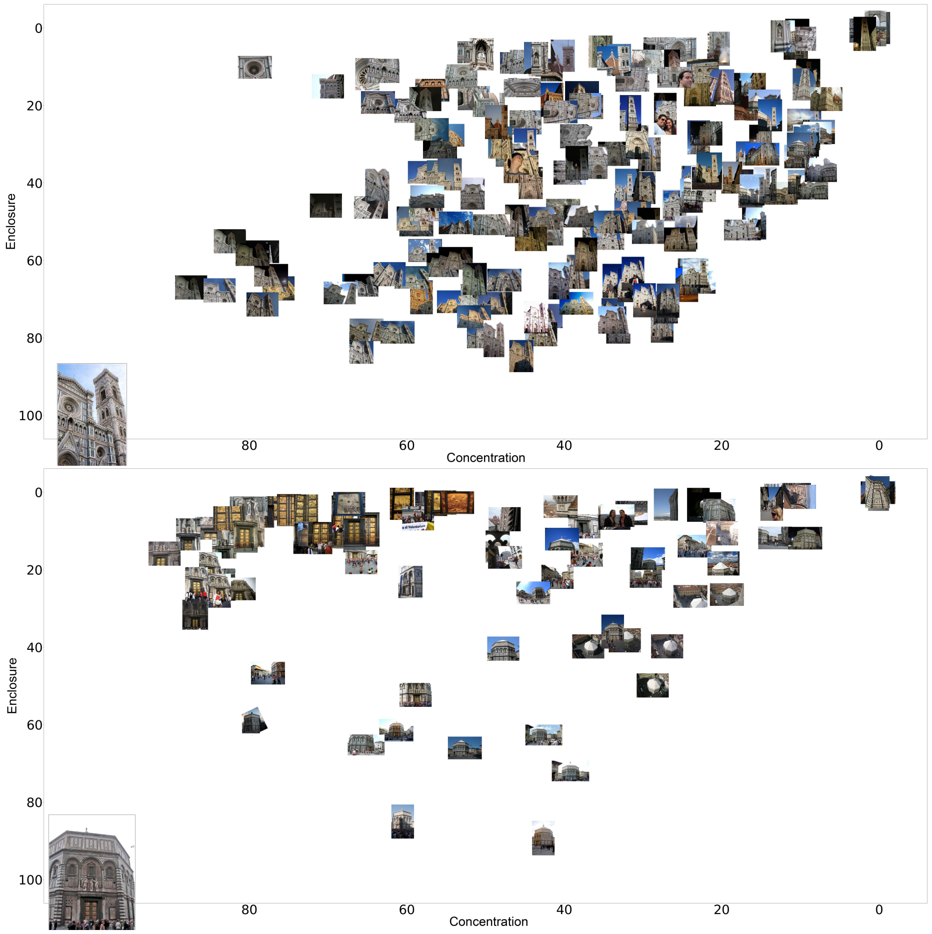


Fig. 16. Generalization plot: Florence. Because the scene is very complex, streets are narrow, and there are not many images of the same view we show two scenes from Florence.

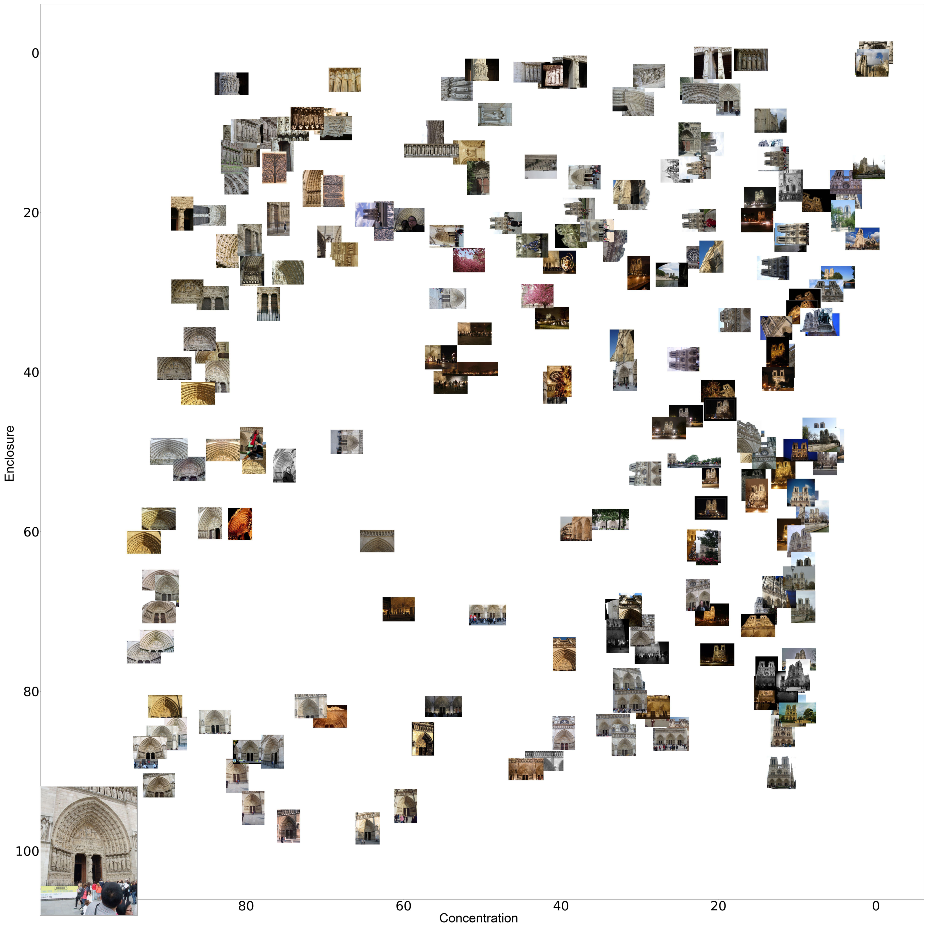


Fig. 17. Generalization plot: Notre Dame.

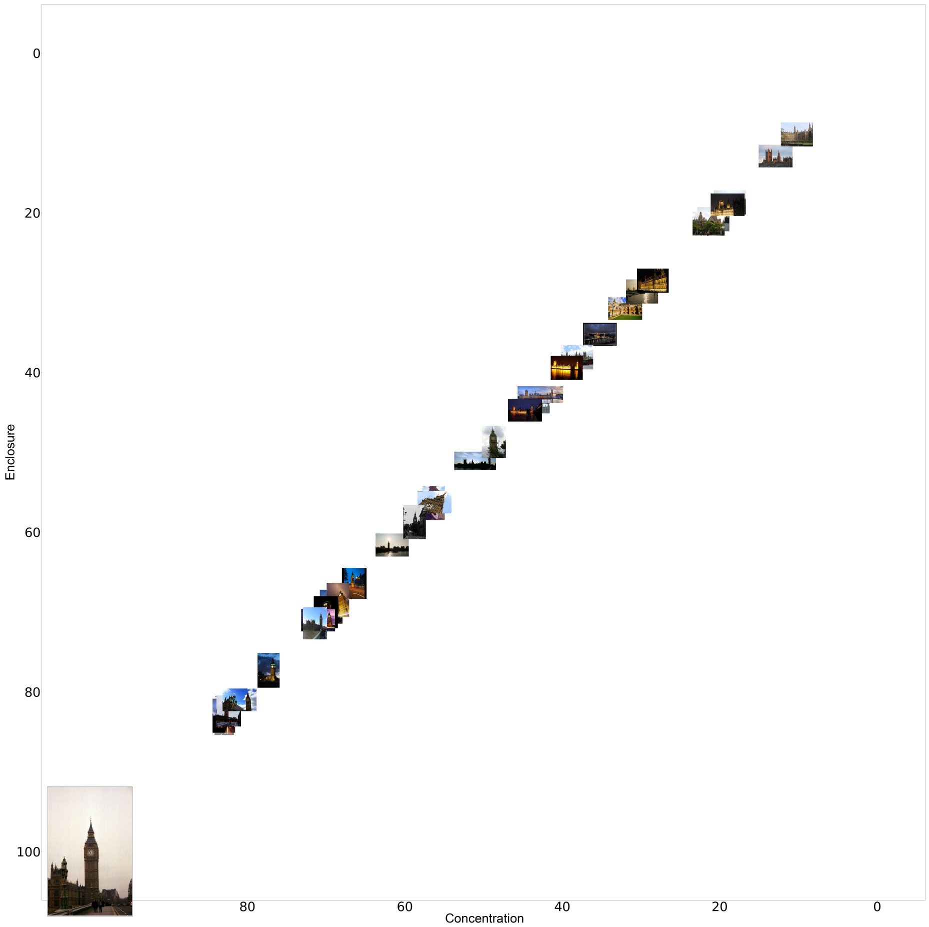


Fig. 18. Generalization plot: Big Ben NSO^{sym}. We show a query from the test set and report the predicted symmetric normalized surface overlap on a subset of test images. Because the embedding space measure is symmetric concentration and enclosure are equal for a given image. It can be observed that nearby images show similar views on the scene. However, the distance between the retrievals is not interpretable.

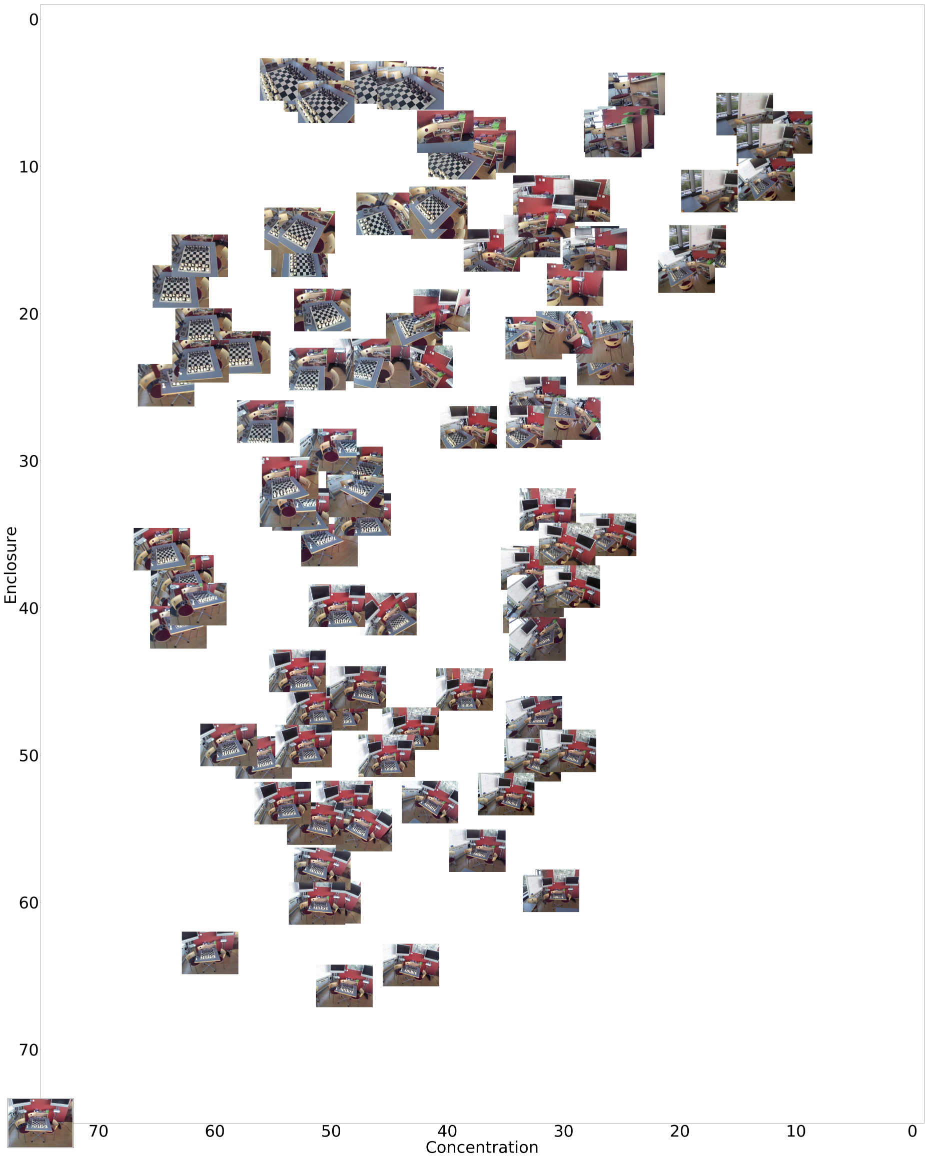


Fig. 19. Generalization plot: 7 Scenes/Chess. Examples from a different dataset. Note how images in the lower right quadrant show the chess table zoomed out, while the images in the upper right corner show the table from a different angle. All images are from the test set.

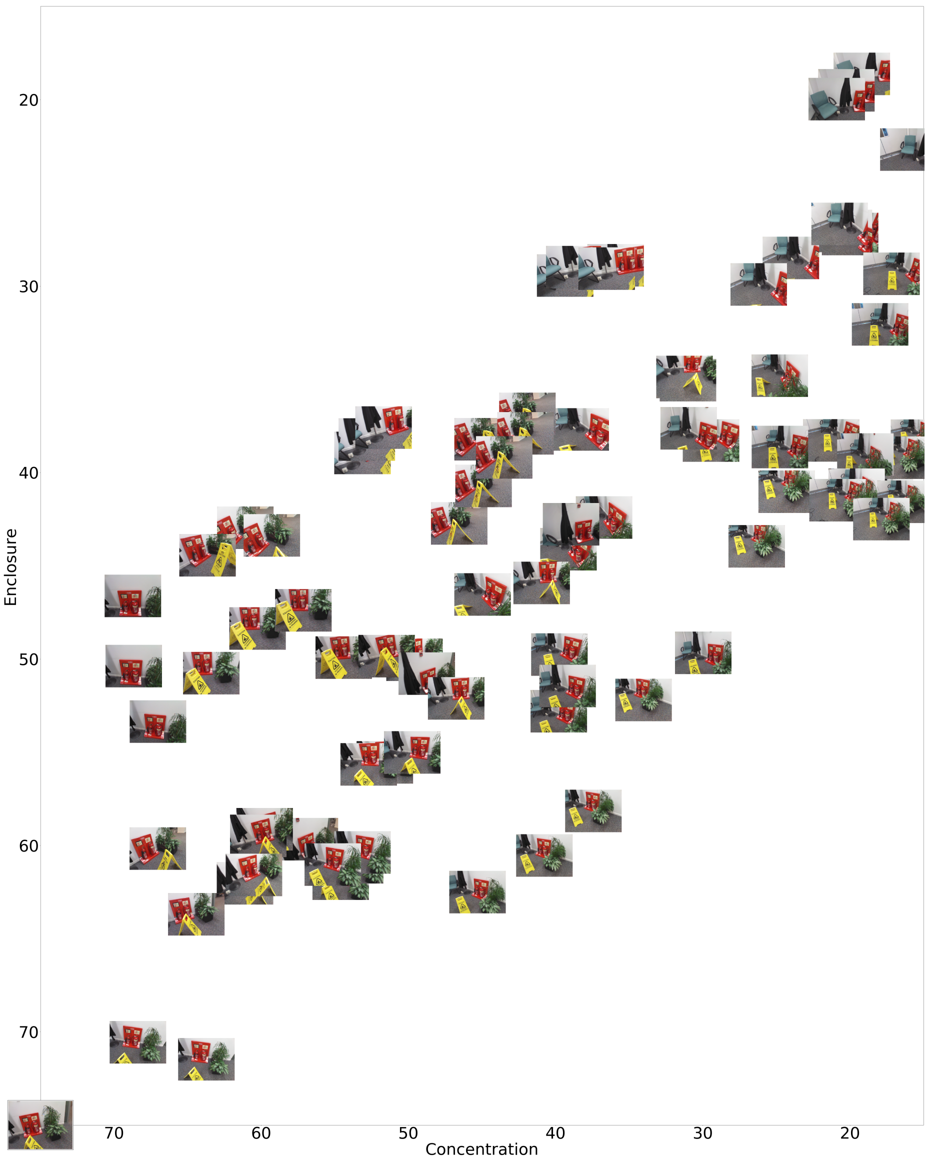
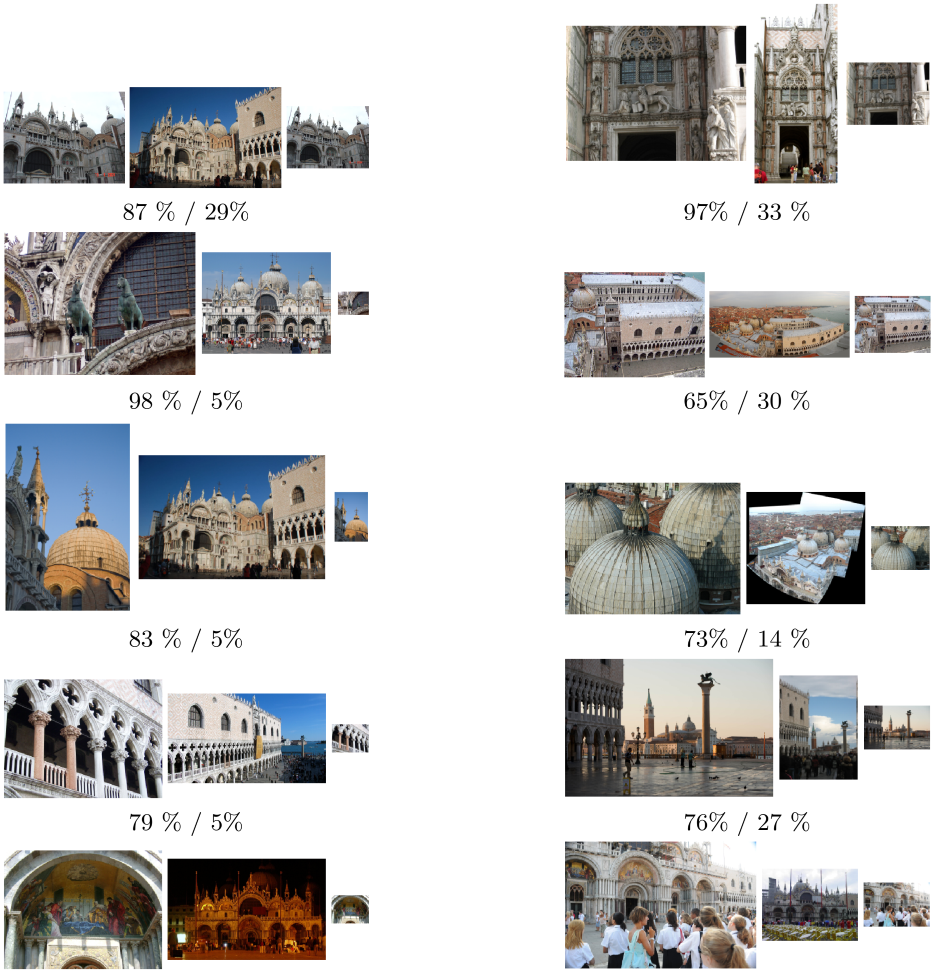


Fig. 20. Generalization plot: 7 Scenes/Fire



87 % / 29%

97% / 33 %

98 % / 5%

65% / 30 %

83 % / 5%

73% / 14 %

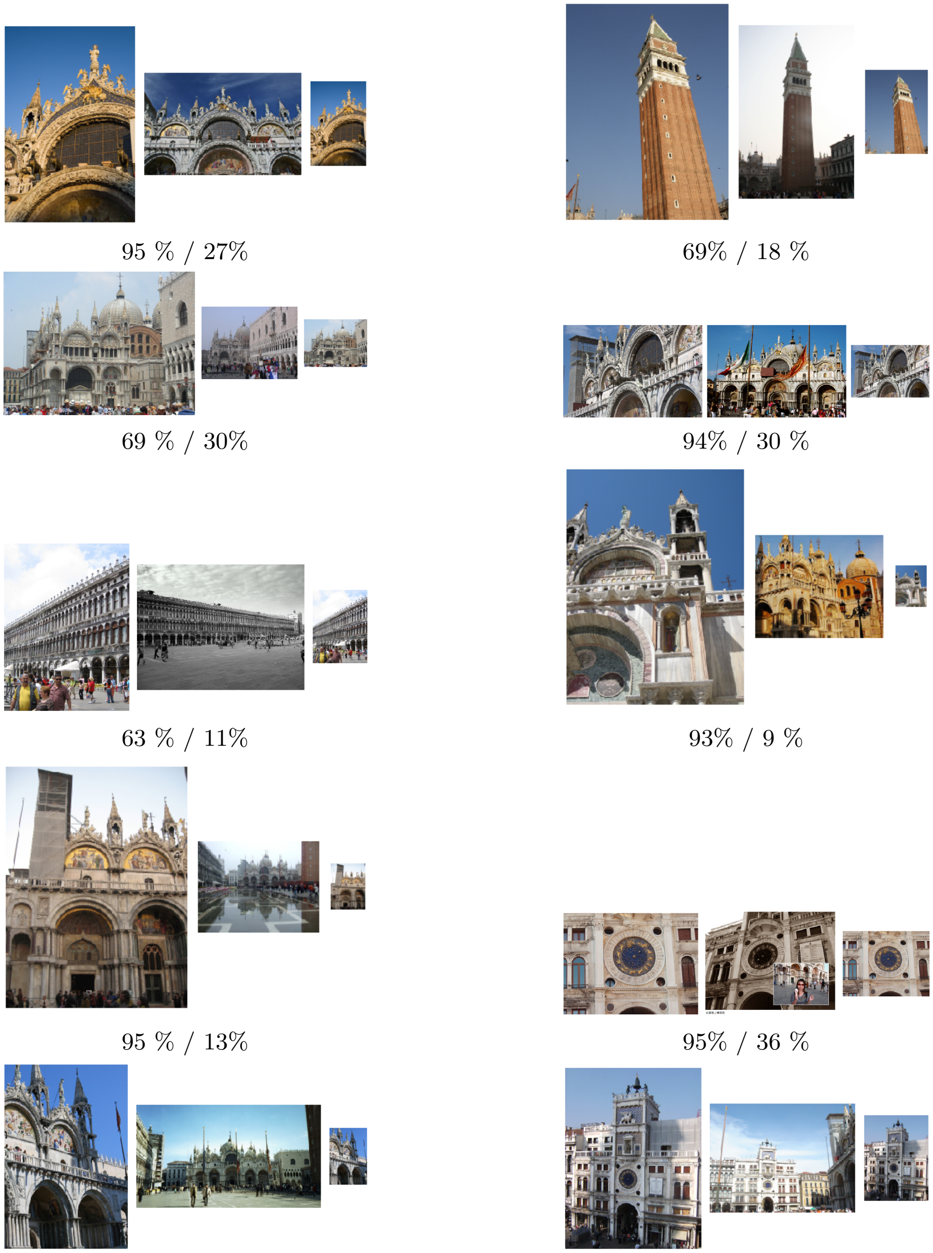
79 % / 5%

76% / 27 %

82 % / 5%

93% / 39 %

Fig. 21. Relative scale plot: Venice No. 1. Illustrated are several examples of how our method can estimate geometric relationships between images. For each pair the enclosure and concentration are calculated from which the relative estimated scaled can be derived. Based on that scale, the first image is resized and shown in the third position. The resized images match the scale of the scene in the first image to the scale in the second image. Note, that the resized images are sometimes very small, and the reader is encouraged to zoom into the images. The two numbers below each image pair show the estimated enclosure and concentration. Note that although some scale estimates are inaccurate, overwhelmingly the rescaling does not increase the scale difference between the two images, but only reduces it. *This caption applies to all Relative scale plots.*



95 % / 27%

69% / 18 %

69 % / 30%

94% / 30 %

63 % / 11%

93% / 9 %

95 % / 13%

95% / 36 %

88 % / 9%

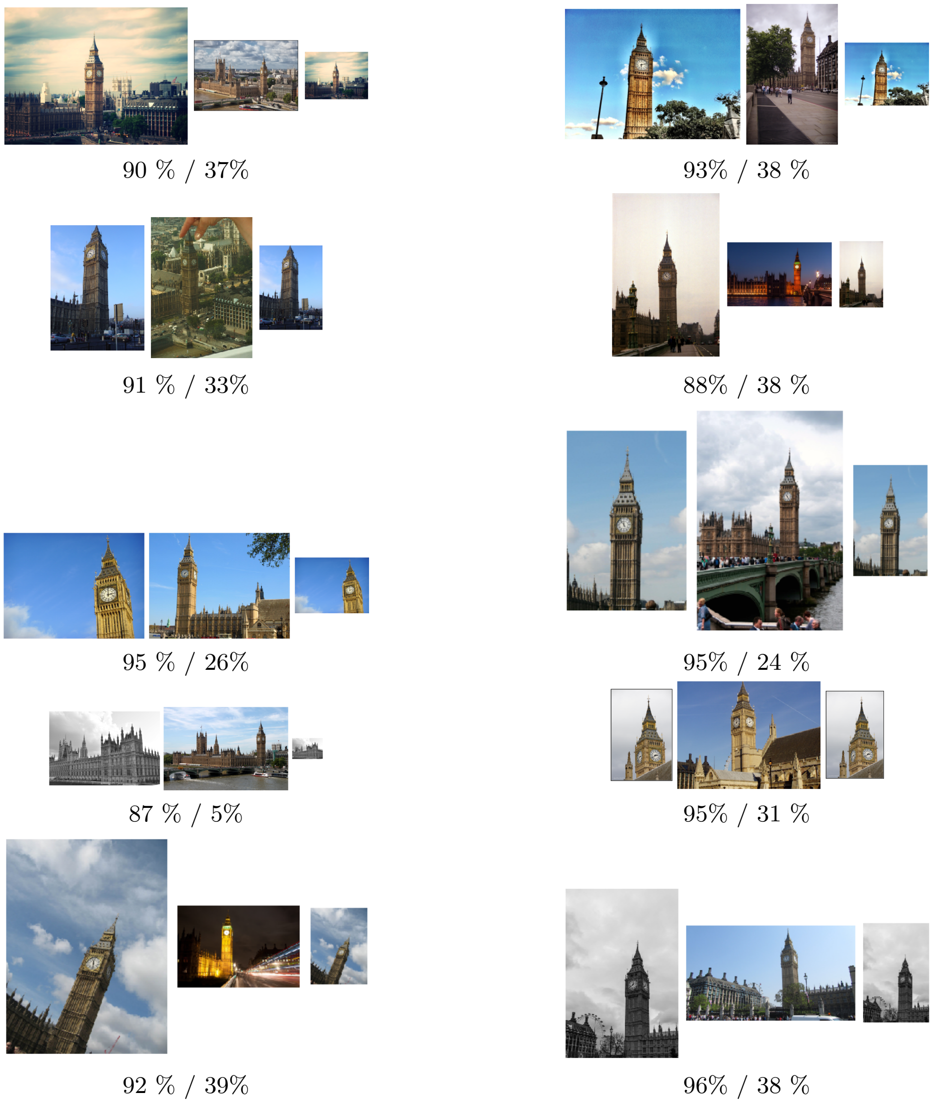
85% / 29 %

Fig. 22. Relative scale plot: Venice No 2.

84% / 10%

80% / 7%

Fig. 23. Relative scale plot: Unsuccessful cases for Venice scene (test image pairs here were found by querying the database for images that had enclosure > 0.6 and $0.05 <$ concentration < 0.4).



90 % / 37%

93% / 38 %

91 % / 33%

88% / 38 %

95 % / 26%

95% / 24 %

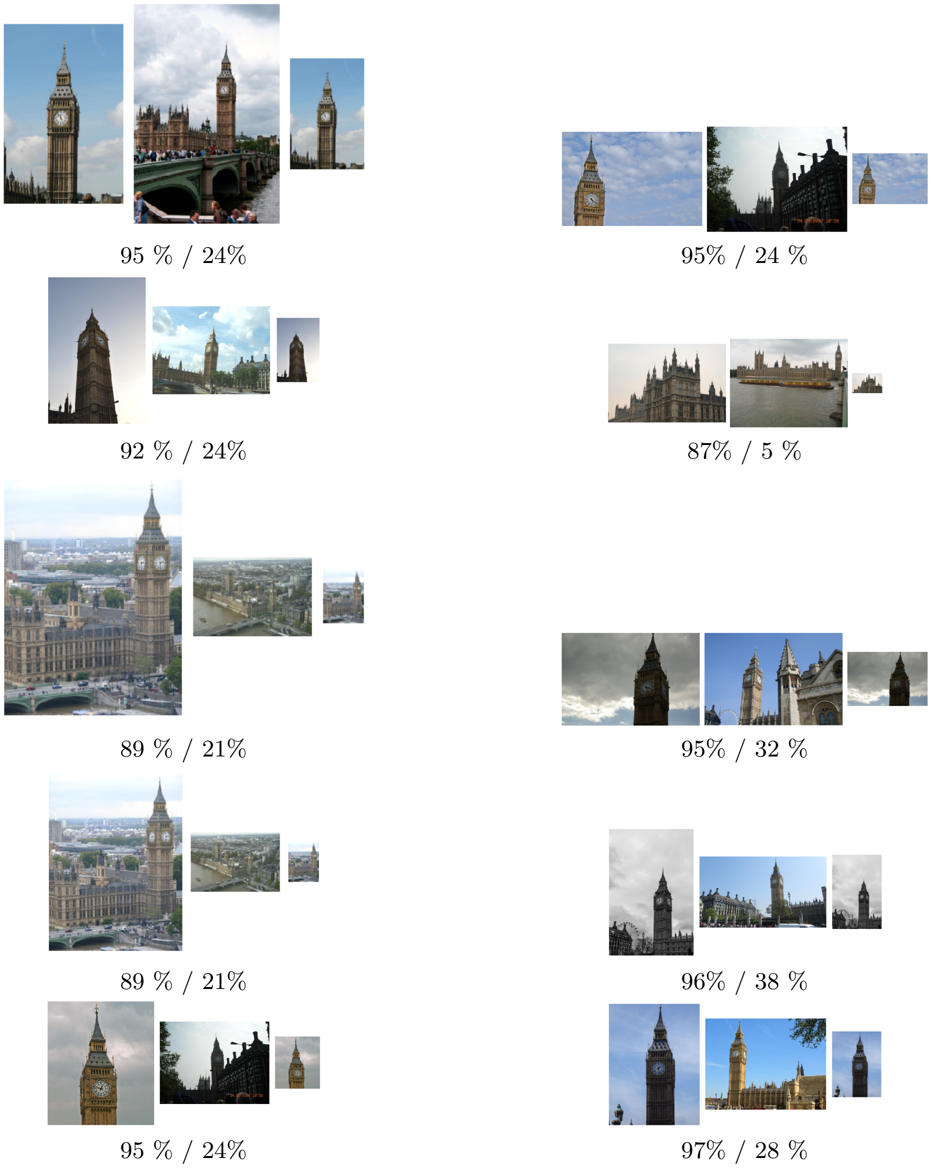
87 % / 5%

95% / 31 %

92 % / 39%

96% / 38 %

Fig. 24. Relative scale plot: Big Ben No. 1.



95 % / 24%

95% / 24 %

92 % / 24%

87% / 5 %

89 % / 21%

95% / 32 %

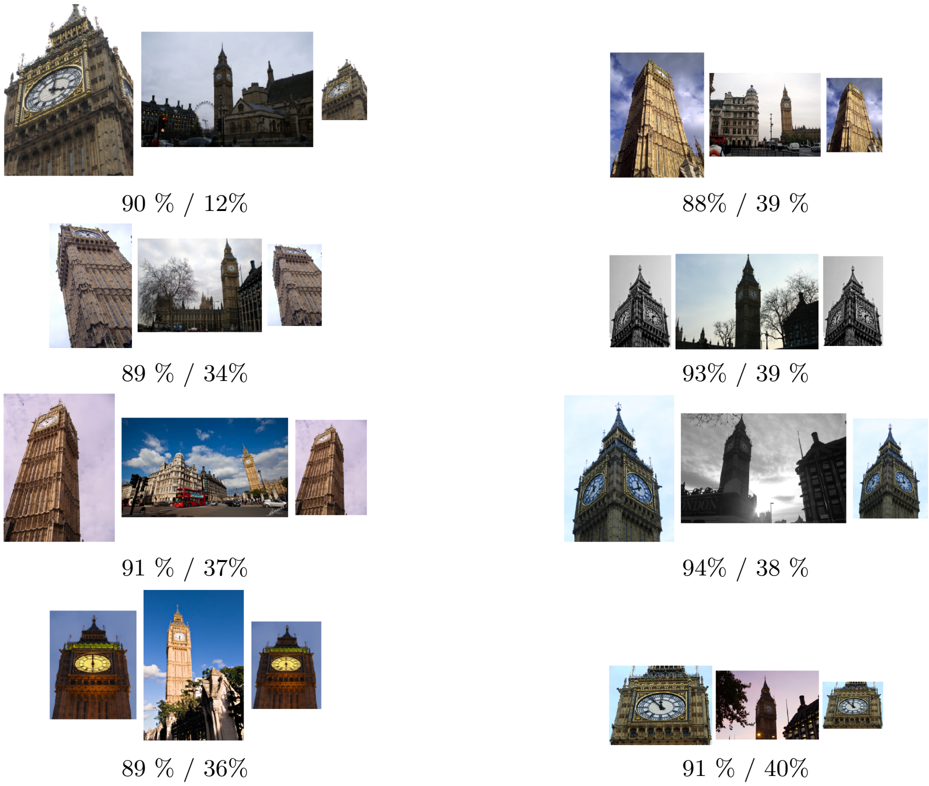
89 % / 21%

96% / 38 %

95 % / 24%

97% / 28 %

Fig. 25. Relative scale plot: Big Ben No. 2.



90 % / 12%

88% / 39 %

89 % / 34%

93% / 39 %

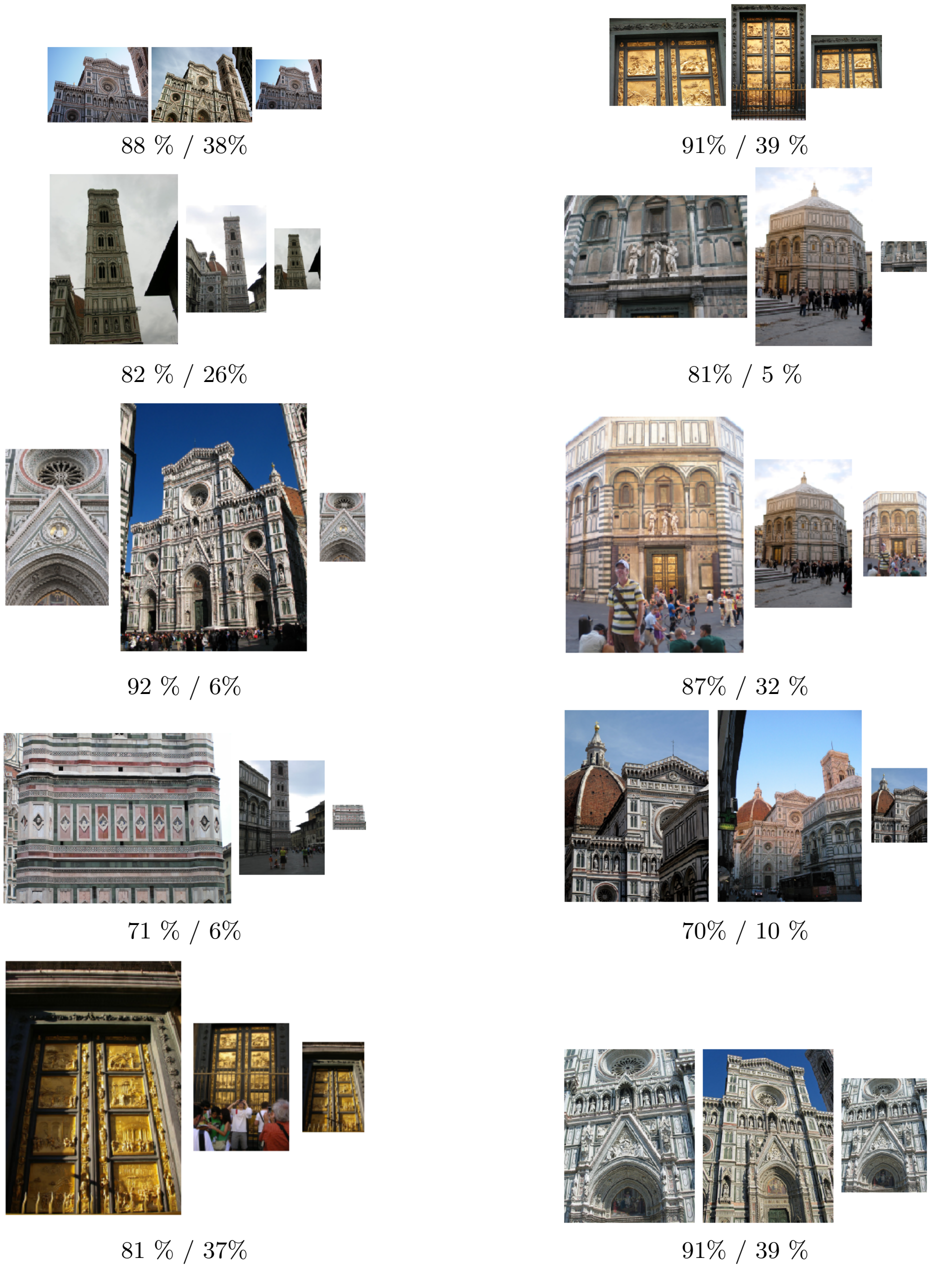
91 % / 37%

94% / 38 %

89 % / 36%

91 % / 40%

Fig. 26. Relative scale plot: Less successful cases Big Ben (18 out of 95 pairs shown in the document).



88 % / 38%

91% / 39 %

82 % / 26%

81% / 5 %

92 % / 6%

87% / 32 %

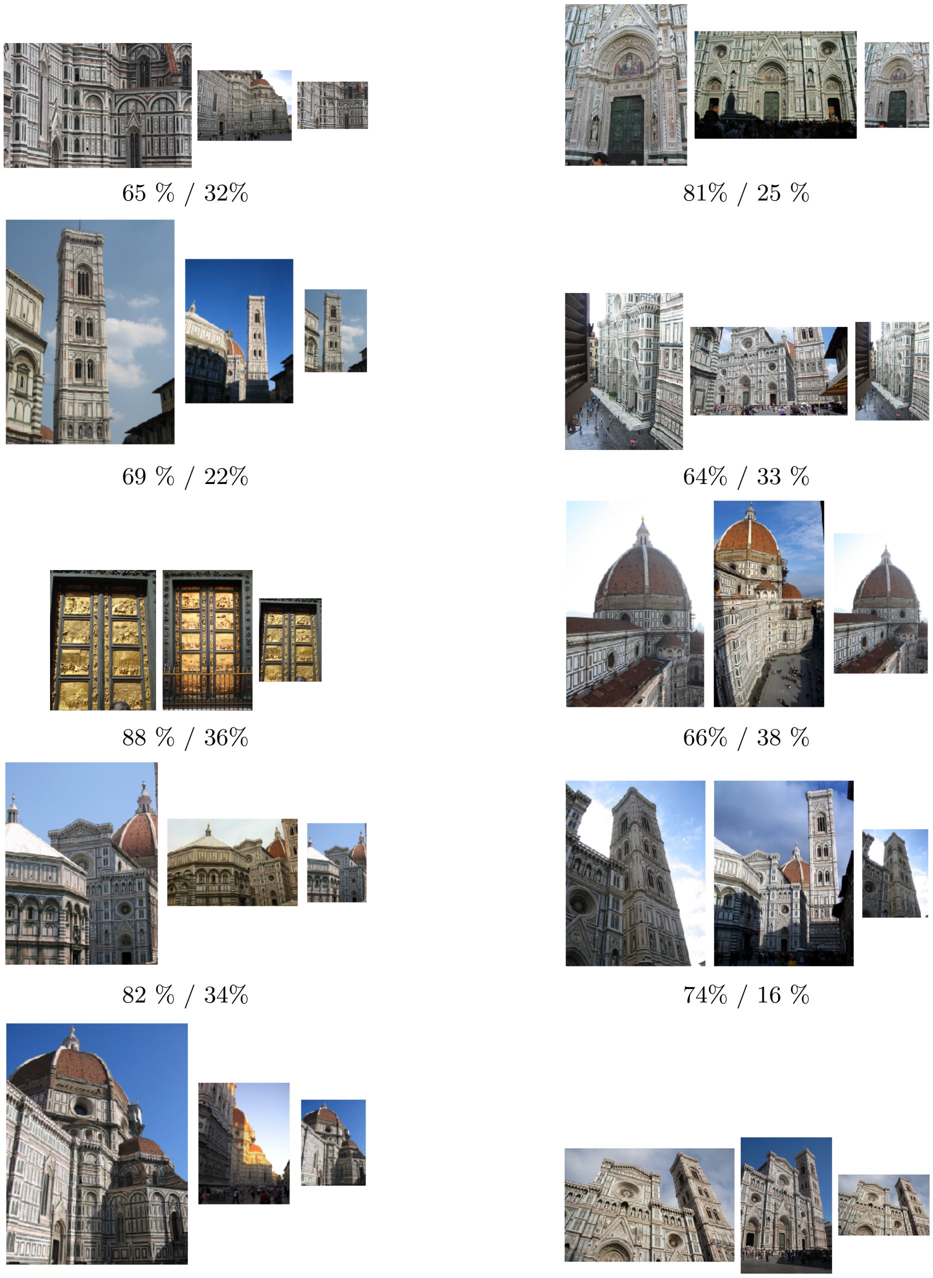
71 % / 6%

70% / 10 %

81 % / 37%

91% / 39 %

Fig. 27. Relative scale plot: Florence No. 1.



65 % / 32%

81% / 25 %

69 % / 22%

64% / 33 %

88 % / 36%

66% / 38 %

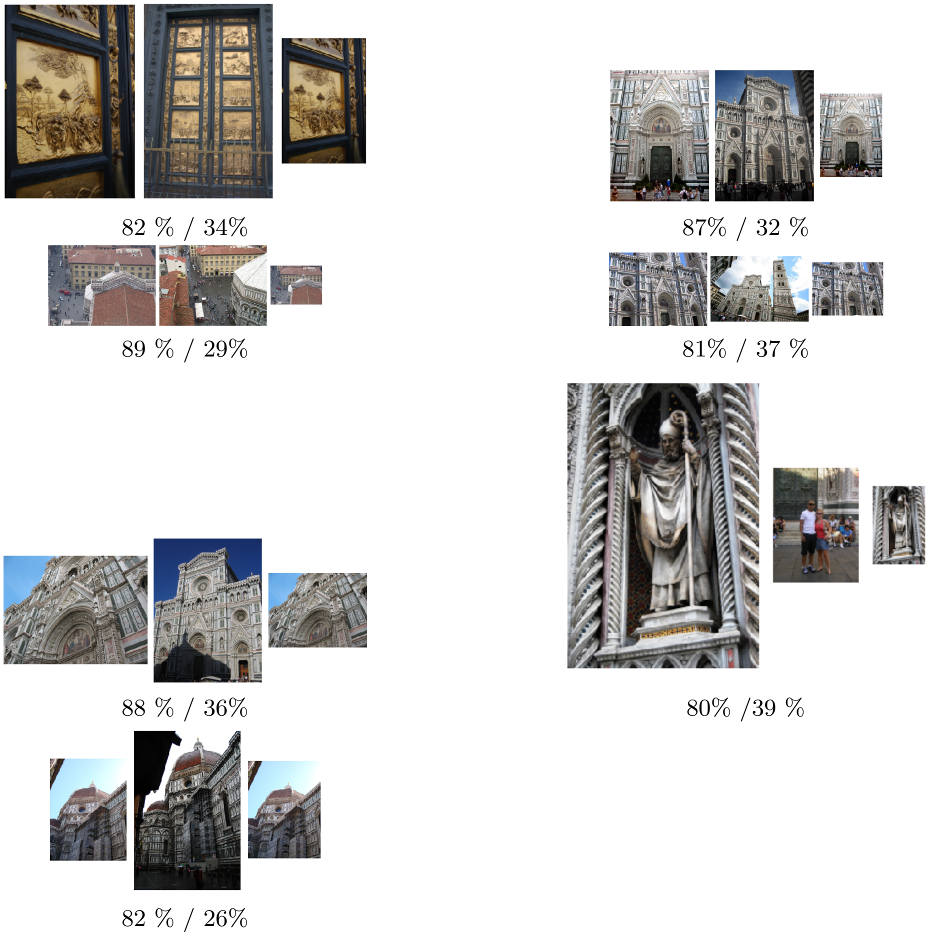
82 % / 34%

74% / 16 %

74 % / 37%

83% / 37 %

Fig. 28. Relative scale plot: Florence No. 2.



82 % / 34%

89 % / 29%

88 % / 36%

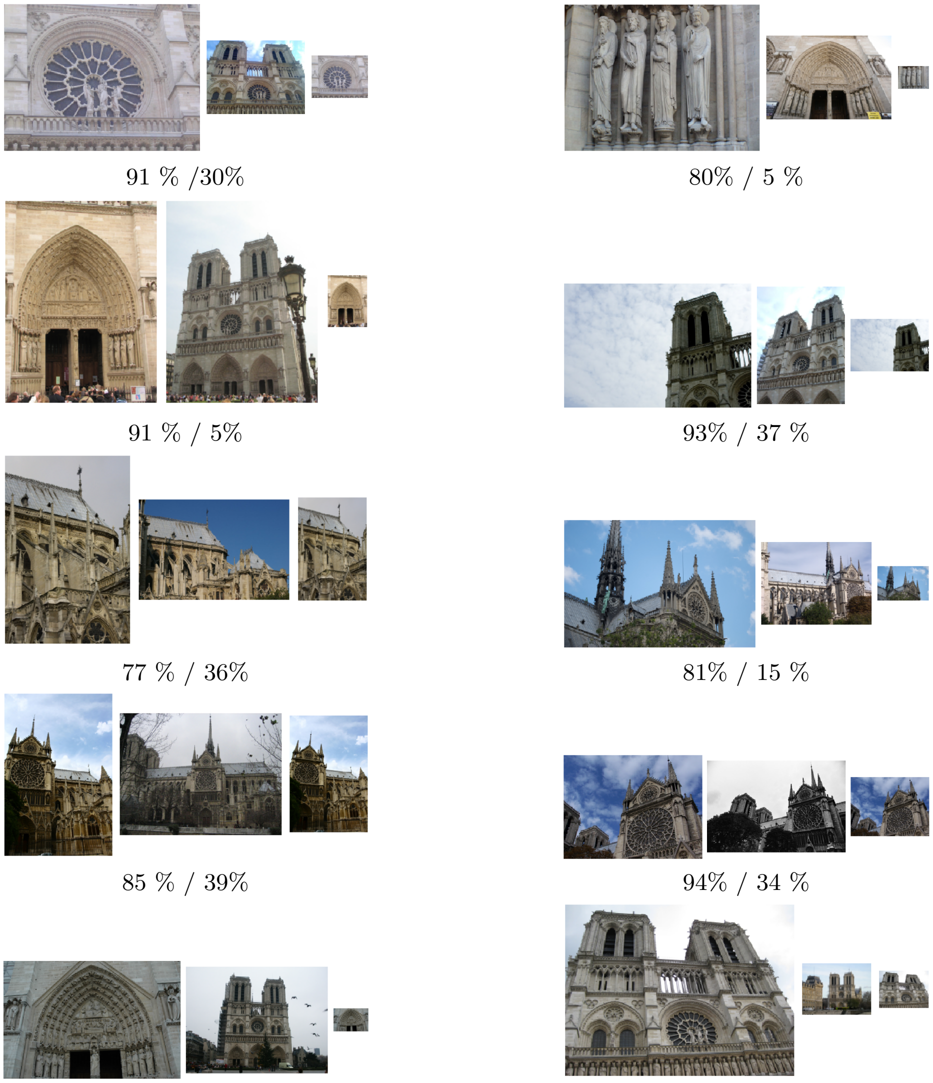
82 % / 26%

87% / 32 %

81% / 37 %

80% /39 %

Fig. 29. Relative scale plot: Less successful cases Florence (7 out of 57 pairs shown in the document).



91 % /30%

80% / 5 %

91 % / 5%

93% / 37 %

77 % / 36%

81% / 15 %

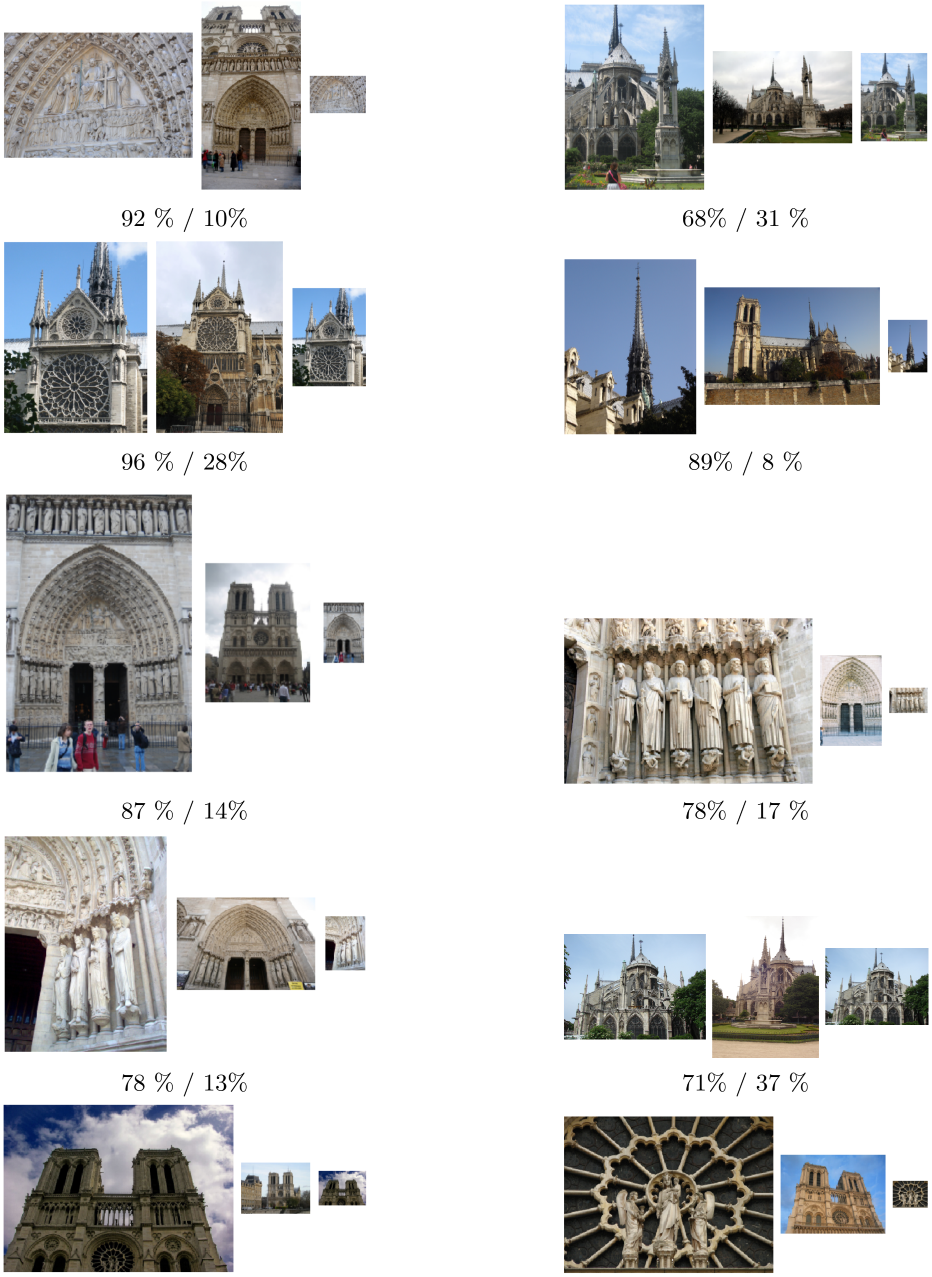
85 % / 39%

94% / 34 %

94 % / 5%

77% / 39 %

Fig. 30. Relative scale plot: Notre Dame No. 1.



92 % / 10%

68% / 31 %

96 % / 28%

89% / 8 %

87 % / 14%

78% / 17 %

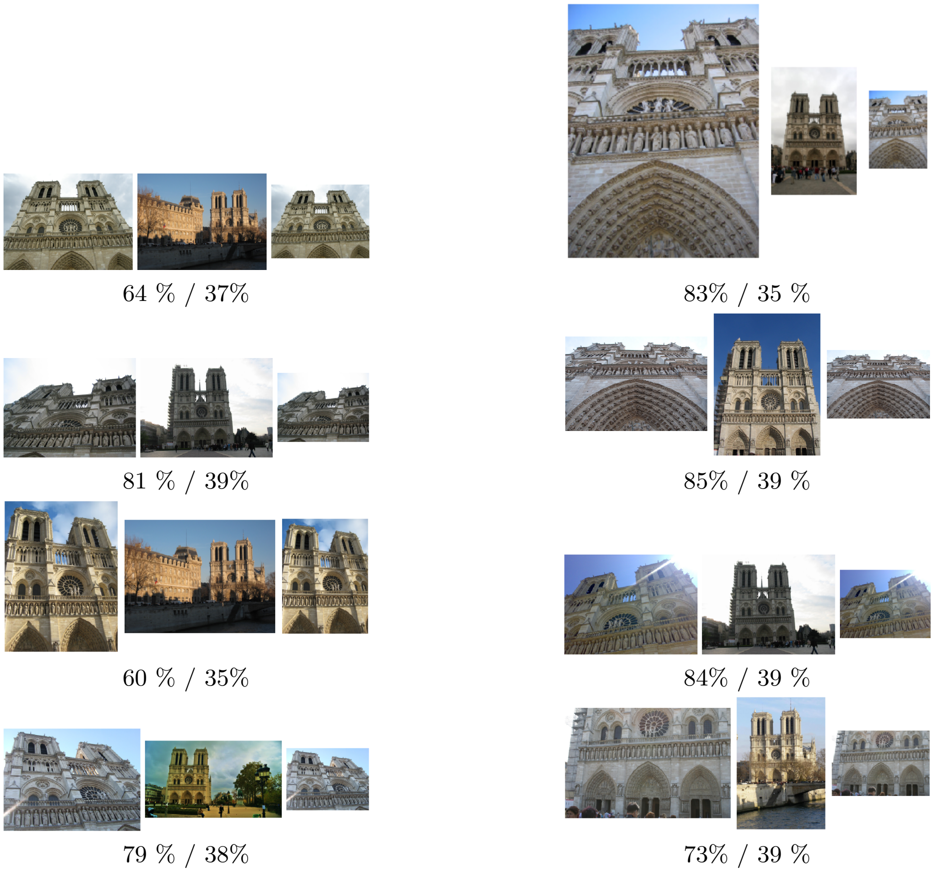
78 % / 13%

71% / 37 %

83 % / 38%

87% / 9 %

Fig. 31. Relative scale plot: Notre Dame No. 2.



64 % / 37%

83% / 35 %

81 % / 39%

85% / 39 %

60 % / 35%

84% / 39 %

79 % / 38%

73% / 39 %

Fig. 32. Relative scale plot: Less successful cases Notre Dame (11 out of 61 pairs shown in the document).

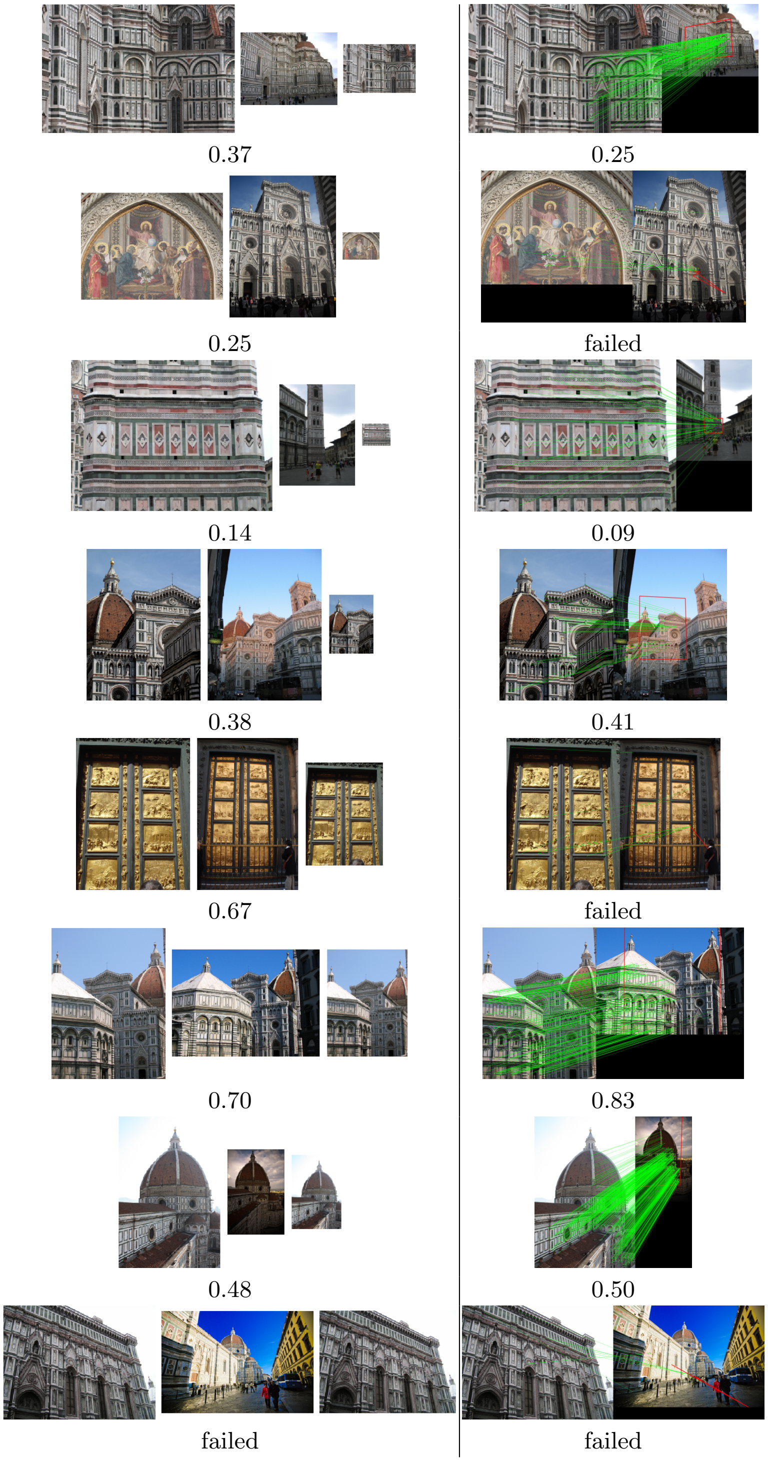


Fig. 33. Comparison of NSO-based scale estimation (left) to a local feature based pipeline (right). Compare the area of the resized image on the left hand size to the area in the red polygon on the right. A scale of 0.37 means that the resized height and width are 0.37 times the size of the original height and width. Note that differences in the two approaches are to be expected. For instance, Normalized Surface Overlap is trained on pixels with depth, so pixels that show sky are not considered in the ground truth. Further, homography estimation is limited to planar surfaces, as opposed to NSO from which the overall scale difference of the scene is inferred.

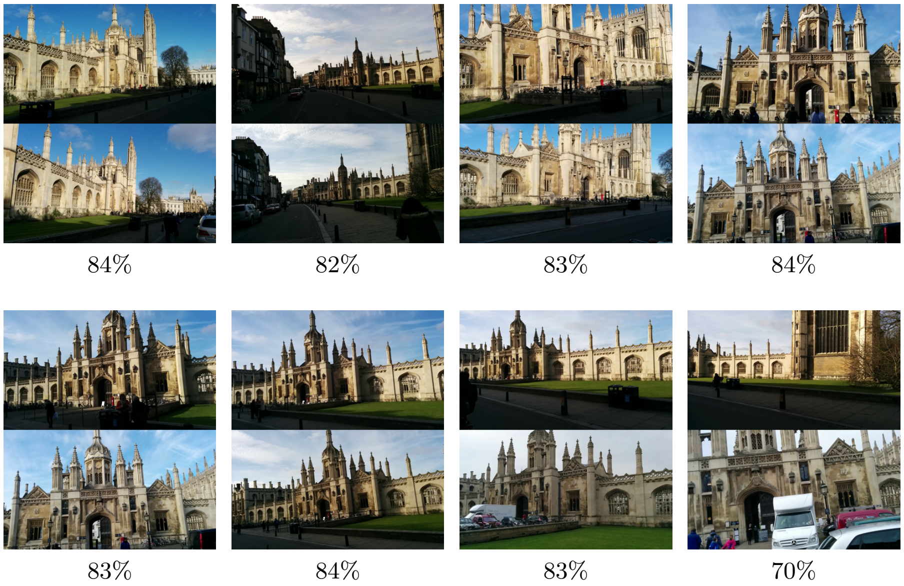


Fig. 34. Generalizability to a new dataset: A model trained on MegaDepth's Notre Dame is used to retrieve nearest neighbors from Cambridge's King's College. The top image of each example is the query and the bottom image is the retrieved image. Below each query-retrieval pair is the predicted symmetric normalized surface overlap $\mathsf{NSO}^{sym}(\mathbf{x}, \mathbf{y}) = \frac{1}{2}(\mathsf{NSO}(\mathbf{x} \mapsto \mathbf{y}) + \mathsf{NSO}(\mathbf{y} \mapsto \mathbf{x}))$. Queries are from the provided test split, and retrievals from the train split. The last example shows a failure case.

References

1. Badino, H., Huber, D., Kanade, T.: The CMU Visual Localization Data Set. http://3dvis.ri.cmu.edu/data-sets/localization (2011) 1
2. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE international conference on computer vision. pp. 2938–2946 (2015) 1
3. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: Computer Vision and Pattern Recognition (CVPR) (2018) 1
4. Sattler, T., Weyand, T., Leibe, B., Kobbelt, L.: Image retrieval for image-based localization revisited. In: BMVC (2012) 1
5. Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in rgb-d images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2930–2937 (2013) 1
6. Wijmans, E., Furukawa, Y.: Exploiting 2D floorplan for building-scale panorama RGBD alignment. In: CVPR (2017) 1