PUMP: Pyramidal and Uniqueness Matching Priors for Unsupervised Learning of Local Descriptors Supplementary Material

Jérome Revaud Vincent Leroy Philippe Weinzaepfel Boris Chidlovskii
NAVER LABS Europe
firstname.lastname@naverlabs.com

In this supplementary material, we first provide the proof for the unique matching prior (Section A). We then give more details about the experimental setting (Section B) and provide text support for the complementary video file (Section C). In Section D, we report the impact of different backbone architectures on the method performance. Section C presents some qualitative results of quasi-dense matching. Finally, Section E proposes a visualization of feature matching via a heatmap.

A. Unique matching prior: proof

In Section 3 of the main paper, we stated the uniqueness of matching prior $\bar{C}$ in Equation 2 (here we drop the index L for the sake of simplicity). Given that a $\bar{C}$ is ℓ_1-normalized, the uniqueness loss concretely encourages all values in $\bar{C}$ to be close to 0 except one per row (*i.e.*, one per high-level patch) that will be close to 1.

To prove this, we remind (Equation 1 of the main paper) that $\forall \boldsymbol{p}$, $\sum_{\boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q}} = 1$. We use the basic summation formula $(\sum_i x_i)^2 = \sum_i x_i^2 + \sum_{i \neq j} x_i x_j$ to get the following equality for any $\boldsymbol{p}$:

$$\Big(\sum_{\boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q}}\Big)^2 = \sum_{\boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q}}^2 + \sum_{\boldsymbol{q}} \sum_{\boldsymbol{q'} \neq \boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q'}} = 1. \quad \text{(A)}$$

This can be rewritten as:

$$\sum_{\boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q}}^2 = 1 - \sum_{\boldsymbol{q}} \sum_{\boldsymbol{q'} \neq \boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q}} \bar{C}_{\boldsymbol{p},\boldsymbol{q'}}. \quad \text{(B)}$$

Given than $\bar{C}$ has only non-negative values, this term reaches its maximum if $\forall \boldsymbol{q} \neq \boldsymbol{q'}$, either $\bar{C}_{\boldsymbol{p},\boldsymbol{q'}} = 0$ holds or $\bar{C}_{\boldsymbol{p},\boldsymbol{q}} = 0$. Given that the sum over all $\boldsymbol{q}$ is equal to 1, thanks to the ℓ_1-normalization, the optimal solution is that all values in $\bar{C}$ for a given $\boldsymbol{p}$ are equal to 0, except one that is equal to 1.

B. Details on the experimental setting

In Section 4.3 of the main paper, we present experimental results on several keypoint-based matching tasks and benchmarks. We have implemented the extraction process of local descriptors according to the available code[1] from R2D2 [3]. Namely, we extract descriptors by rescaling the input image at multiple scales (powers of $2^{1/4}$), starting from the original resolution and downscaling the image until it becomes smaller than 30% of its initial size. For the sake of fairness, when computing MMA (Tables 3 and 4, Figure 5 of the main paper) we select a matching threshold that corresponds to the same number of matches obtained by default with the employed keypoint detector (SIFT, SuperPoint or R2D2). To that aim, we reject all matches between descriptors $\mathbf{d_1}, \mathbf{d_2}$ such that $\mathbf{d_1}^\top \mathbf{d_2} < \tau$, where τ is a matching threshold tuned to reach the target number of matches. We believe this is fair, since what matters ultimately for downstream tasks is having as many correct matches as possible, and not just the proportion of correct matches. We plot in Figure A the evolution of the MMA@5 score as a function of the number of matches per image, in comparison to the R2D2 descriptor, when using R2D2 keypoint detection. Our method obtains consistently higher MMA scores for higher number of matches. In other words, regardless of the matching threshold, PUMP performs significantly better than R2D2 in the sense that it can output much more correct matches per image. Note that it can also achieve a much better MMA as well, regardless of how the matching threshold for R2D2 is tuned. Similar conclusions hold for all considered keypoint detectors and descriptors.

C. Video with additional qualitative results

In the supplementary video file, we complement the qualitative results of quasi-dense matching presented in Section 4.2 and Figure 4 of the main paper. For pairs of source-target images, the video shows the result of warping a target image to a source image according to the interpolated PUMP matches. Examples include ETH3D 'Lakeside' scene (Figure 4) as well as challenging 'in the wild'

[1] https://github.com/naver/r2d2

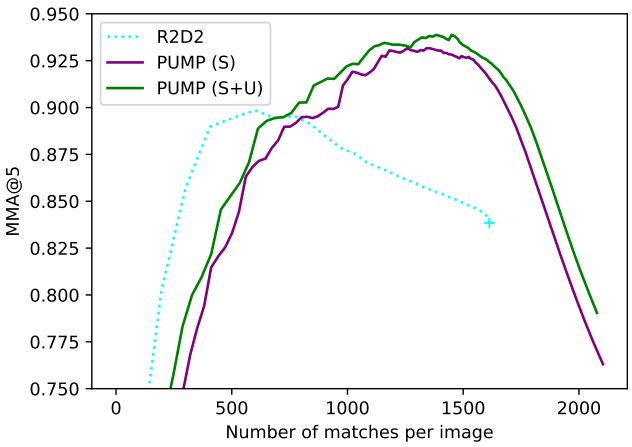

Figure A. Evolution of MMA@5 score on HPatches as a function of the matching threshold (all sequences combined, *i.e.*, viewpoint + illumination). Exactly the same keypoints (here, R2D2 keypoints) are used for all methods, only keypoint descriptors change.

image pairs, with high-frequency details, a wide baseline, scale variation, and illumination changes. We find these examples interesting as they showcase the robustness and generalization capability of PUMP. In fact, we used the model trained only on SfM-120k [2] in the S+U setting, *i.e.*, self-supervision in the form of synthetic deformations in addition to our unsupervised loss. This model is able to consistently handle complex textures such as moss, rocks, trees and snow. In Figure C, we show an enlarged version of the wide baseline matching examples from Figure 4 of the main paper. Large errors only appear around motion boundaries.

Figure B. An image pair from MegaDepth [1] depicting the *same* scene, yet without any shared keypoints according to COLMAP.

D. Impact of the backbone architecture

In this section, we verify if the gain of PUMP can be attributed to using a ConvMixer backbone or from the unsupervised loss. To this end, we report in Table A the results using the backbone from R2D2 instead of ConvMixer. It confirms our initial findings: even though R2D2 backbone performs sightly worse on HPatches, the relative improvement yielded by our unsupervised loss ($S \rightarrow S + U$) stays nearly identical to the one observed with ConvMixer on both datasets.

Backbone	Loss	HPatches			Aachen Day-Night		
		MMA@1	MMA@3	MMA@5	0.25m, 2°	0.5m, 5°	5m, 10°
ConvMixer	S	37.46	83.38	91.46	69.63	84.82	96.86
ConvMixer	S+U	**37.83**	**84.16**	**92.42**	73.30	86.91	98.43
R2D2	S	36.76	80.73	88.71	70.68	85.86	96.34
R2D2	S+U	37.20	81.39	89.41	**73.82**	**87.43**	**98.95**

Table A. Performance comparison for different backbone architectures.

E. Visualization of feature matching heat maps

Using our unsupervised loss could allow to increase the amount of data source for training, especially because the SfM pipeline often fails, *e.g.* in case of heavy occlusions, changing illumination, lack of surface texture, or missing camera intrinsics. Figure B shows an example image pair from MegaDepth [1] for which there is no shared SIFT keypoints according to COLMAP's output, despite depicting the exact same building.

In order to give better insights about how the method works internally, we show in Figures D and E the correlation maps at different levels in the pyramid. The leftmost column shows the first image, with the query pixel $\mathbf{p}$ and its receptive field (*red*), that doubles at each pyramid level. The middle column depicts, at each pyramid level, the positions of the 3 most correlated patches in the second image according to the correlation map $C^{\ell}_{\boldsymbol{p},\cdot}$ for $\mathbf{p}$, respectively in red, green and blue. The rightmost column shows the raw correlation map $C^{\ell}_{\boldsymbol{p},\cdot}$. While initial correlations are extremely noisy due to the very challenging nature of the scene and the inherent lack of specificity for small patches on repetitive textures, it gradually improves when more and more correlations are aggregated into higher-level patches. Finally, the high level parent patch at level $\ell = 4$ resolves the correct correspondence with little to no ambiguity, illustrating how higher-level correlations are consolidated compared to low-level ones.

References

[1] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 2

[2] Filip Radenovic, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *CVPR*, 2018. 2

[3] Jerome Revaud, Philippe Weinzaepfel, César Roberto de Souza, and Martin Humenberger. R2D2: repeatable and reliable detector and descriptor. In *NeurIPS*, 2019. 1

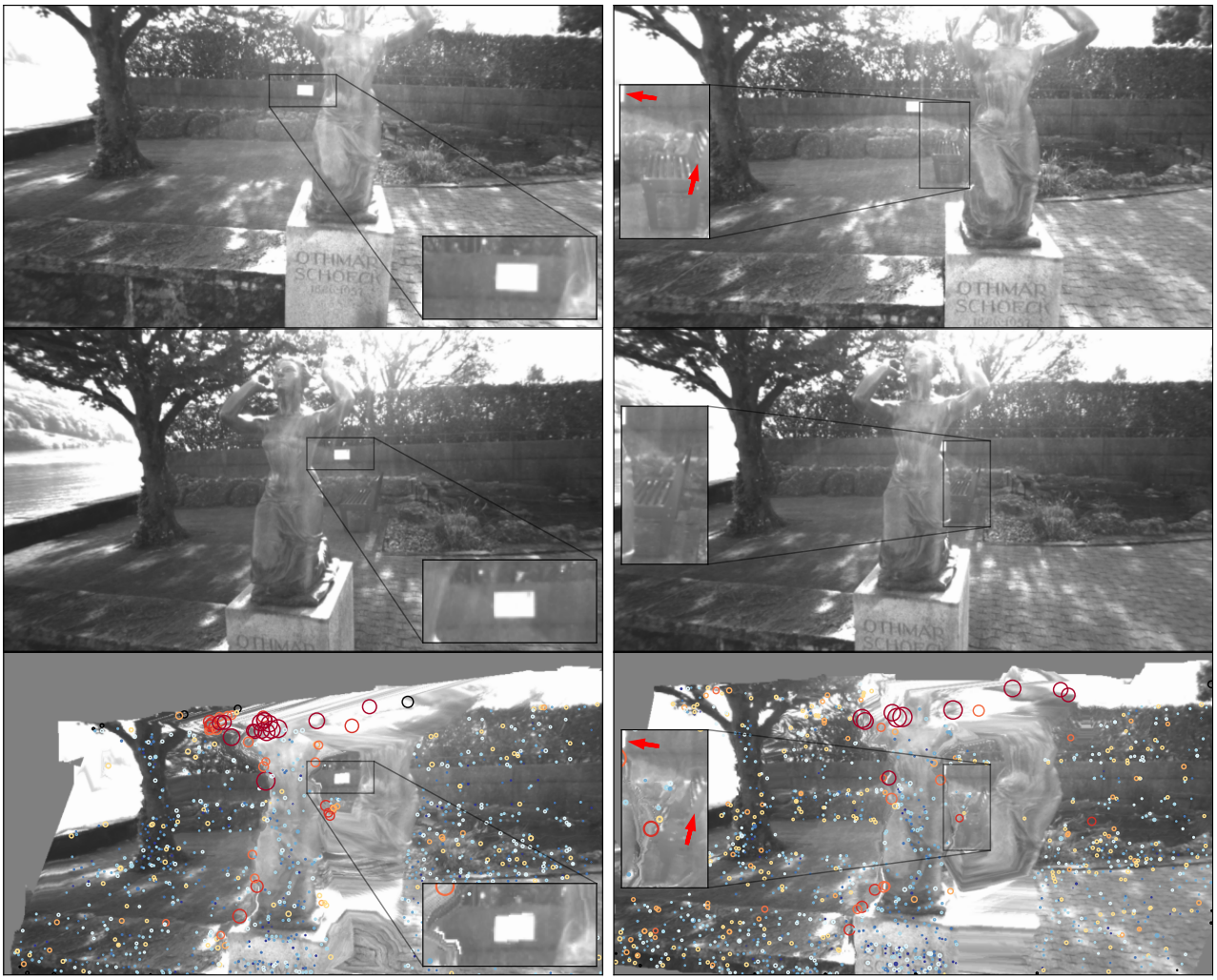

Figure C. Enlarged version of the wide baseline matching examples from Figure 4 of the main paper. Each column shows a different pair from the most challenging 'lakeside' sequence in the ETH3D dataset. The first two rows show pairs of images to match. The third row shows the first image warped to the second one according to the dense matching predicted by our model. Errors on the ground-truth control points are represented as circles whose area is proportional to the error, using KITTI error color-code. Zoom insets highlight challenging areas enclosed between motion boundaries. Overall, large errors mostly appear around motion boundaries.

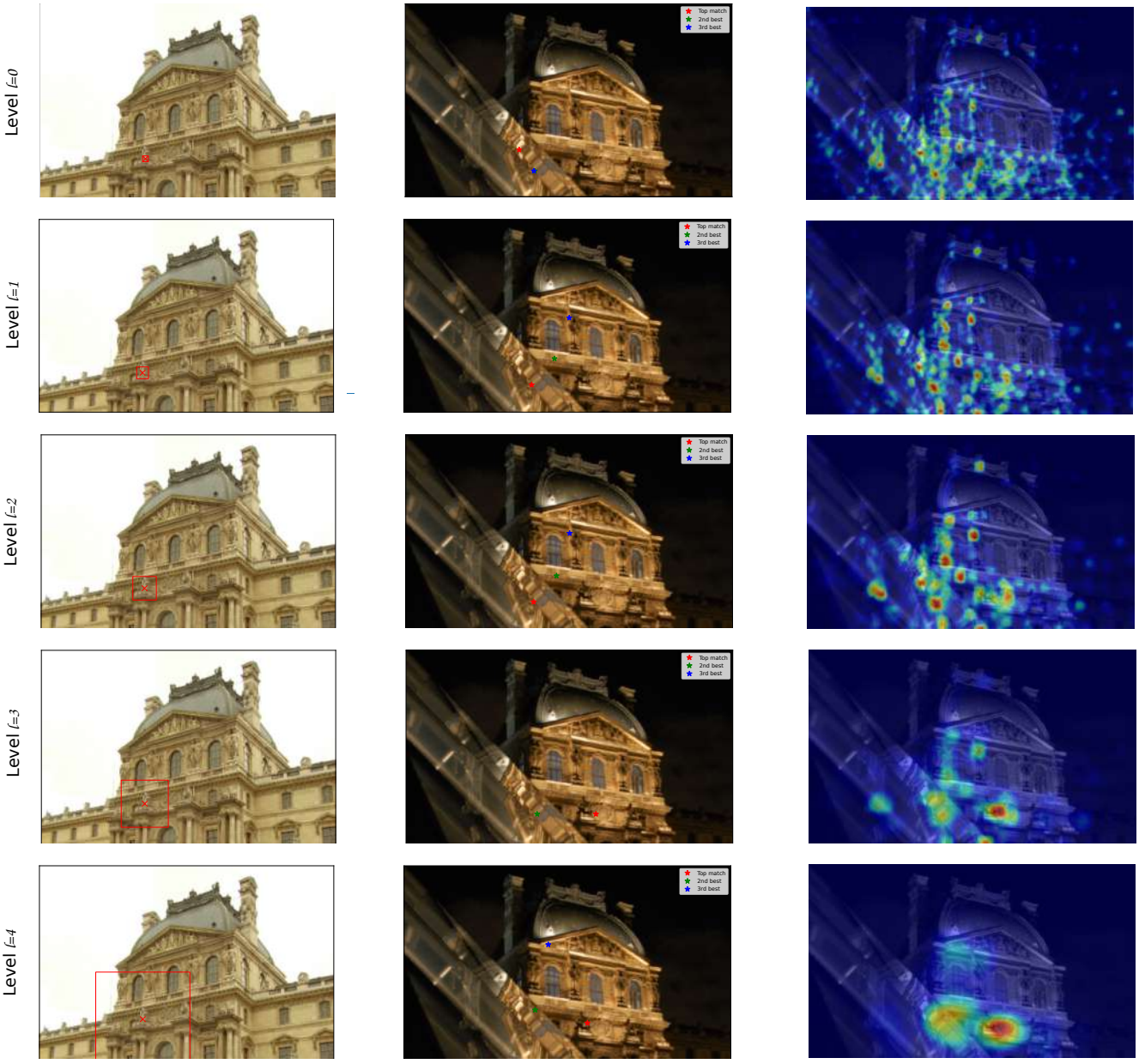

Figure D. Consolidated correlation maps $C^{\ell}_{\boldsymbol{p},\cdot}$ and top-3 matches at different pyramid level $\ell = \{1, 2, 3, 4\}$ for a particular pixel $\boldsymbol{p}$ in the first image. See text for details.

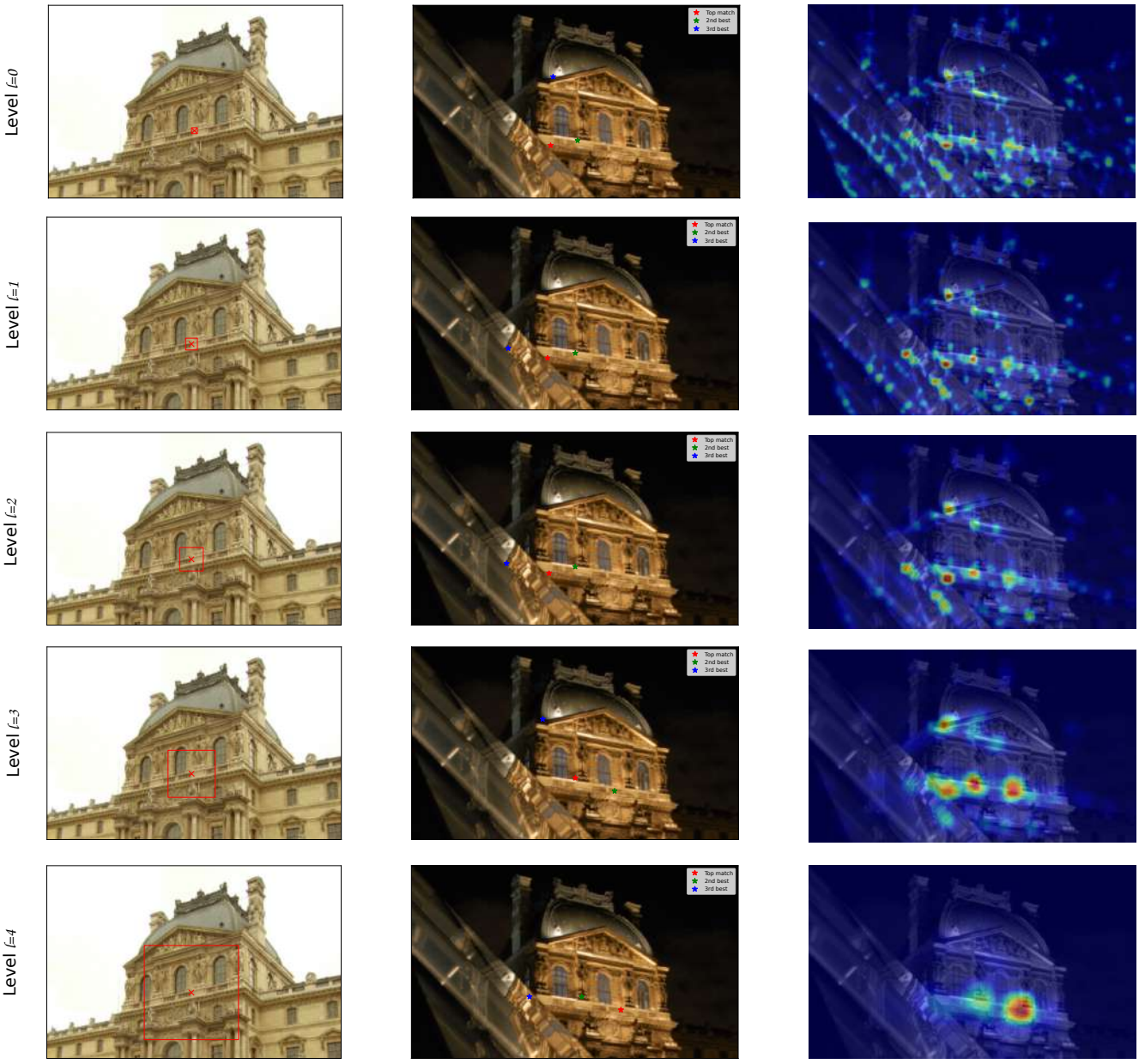

Figure E. Consolidated correlation maps $C^{\ell}_{\mathbf{p},\cdot}$ and top-3 matches at different pyramid level $\ell = \{1, 2, 3, 4\}$ for a particular pixel $\mathbf{p}$ in the first image. See text for details.