Appendix

In a first step, we review the multi-crop strategy [5]. The pseudo-code for the global-crop algorithm and the multi-crop algorithm can be respectively found in Algorithm 1 and Algorithm 2. We then show visualizations of the matchings which are enforced at training time, visualizations of the collapse of the similarity matching function and visualizations of the correspondence mapping between two different images based on the similarity of the local-representations.

A. Details on Multi-Crop

Details on the multi-crop strategy [5] are left out in the main paper for simplicity. For completeness, we give a review of the multi-crop strategy and explicitly explain how it affects our loss terms.

A.1. Review of Multi-Crop

In Section 3.2 we explain the data-augmentation pipeline and how multiple augmentations of a single input image are generated. To obtain an augmented image $\tilde{\boldsymbol{x}}$ from an input image $\boldsymbol{x}$, we sample an augmentation vector $\boldsymbol{w} = [\boldsymbol{w}_{geo}, \boldsymbol{w}_{pho}]$ from some distribution $\mathcal{D}_{aug}$. This augmentation vector parametrizes both the geometric- and photometric transforms that are applied to $\boldsymbol{x}$. Spatial transforms include `CROP`, `RESIZE` and `HORIZONTAL_FLIP` while photometric transforms include `COLOR_JITTER`, `SOLARIZE`, `GAUSSIAN_BLUR` and `GRAYSCALE`. We denote the composition of all geometric transforms by $\mathbf{G}$ and the composition of all photometric transforms by $\mathbf{P}$. In Section 3.2, we assumed that all augmentation vectors $\boldsymbol{w}$ are sampled from the same distribution $\mathcal{D}_{aug}$. The multi-crop strategy [5] removes this assumption. Instead, we segregate augmentations into two categories, global- and local-crops. Local-crops are taken from smaller regions of the input image while global-crops are taken from larger ones. Local-crops are also resized to 96×96 pixels while global-crops are resized to 224×224 pixels. This is illustrated in Figure 6.

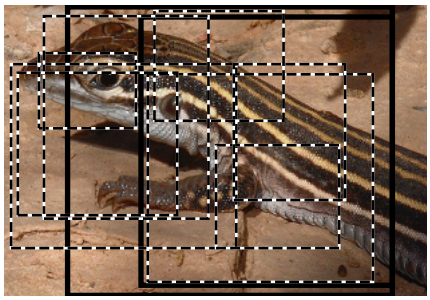

Figure 6. **Example of the sampled locations using the multi-crop strategy.** 2 global- and 8 local-crops are respectively shown in thick full lines and thin dashed.

For every single original input image $\boldsymbol{x}$, 2 global-crops and $N_L = 8$ local-crops are generated. All augmentation vectors for local-crops are sampled from the same distribution $\mathcal{D}^L_{aug}$ while the augmentation vectors for global-crops are individually sampled from two different distributions: $\mathcal{D}^G_{aug_1}$ and $\mathcal{D}^G_{aug_2}$. Given an input image $\boldsymbol{x}$, the set of augmentations $\mathcal{V} = \{\tilde{\boldsymbol{x}}^1, \tilde{\boldsymbol{x}}^2, \cdots \tilde{\boldsymbol{x}}^{N_L+2}\}$ is generated following Algorithm 2. Lines 4-7 from Algorithm 1 can be replaced with all lines from Algorithm 2 to make it implement the multi-crop strategy. $\tilde{\boldsymbol{x}}^1$ and $\tilde{\boldsymbol{x}}^2$ correspond to the global-crops while $\tilde{\boldsymbol{x}}^3, \tilde{\boldsymbol{x}}^4, \cdots, \tilde{\boldsymbol{x}}^{2+N_L}$ correspond to the local-crops. More details can be found in the code.

Algorithm 1 Dual global-local self-distillation framework

Input: $\mathcal{X}$: an unlabeled dataset, N: the number of augmentations per input image, $\mathbf{P}$: a photometric-augmentation function, $\mathbf{G}$: a geometric-augmentation function, $\mathcal{D}_{aug}$: an augmentation-vector distribution, f_s: a backbone student, f_t: a backbone teacher, `OPTIMIZER`: an optimizer
Output: Trained weights $\boldsymbol{\theta}_t$

1: $\boldsymbol{\theta}_s = \boldsymbol{\theta}_t = \boldsymbol{\theta}_{init}$
2: **for** epoch = $1 \cdots$ NB_EPOCHS **do**
3: **for** $\boldsymbol{x} \in \mathcal{X}$ **do**
4: **for** $n \in [N]$ **do**
5: Sample $\boldsymbol{w}^n = [\boldsymbol{w}^n_{geo}, \boldsymbol{w}^n_{pho}]$ from $\mathcal{D}_{aug}$
6: $\tilde{\boldsymbol{x}}^n = \mathbf{P}\left(\mathbf{G}(\boldsymbol{x}, \boldsymbol{w}^n_{geo}), \boldsymbol{w}^n_{pho}\right)$
7: **end for**
8: $\mathcal{W}_{geo} = \{\boldsymbol{w}^1_{geo}, \boldsymbol{w}^2_{geo}, \boldsymbol{w}^3_{geo}, \cdots\}$
9: Infer $\mathcal{E} = \{\boldsymbol{e}^1, \boldsymbol{e}^2, \boldsymbol{e}^3, \cdots\}$ from $\mathcal{W}_{geo}$
10: $\mathcal{V} = \{\tilde{\boldsymbol{x}}^1, \tilde{\boldsymbol{x}}^2, \tilde{\boldsymbol{x}}^3, \cdots\}$
11: $\bar{\mathcal{Z}}_s = \{\bar{f}_s(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in \mathcal{V}\}$
12: $\bar{\mathcal{Z}}_t = \{\bar{f}_t(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in \mathcal{V}\}$
13: $\mathcal{Z}_s = \{f_s(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in \mathcal{V}\}$
14: $\mathcal{Z}_t = \{f_t(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in \mathcal{V}\}$
15: $\mathcal{L} = \mathcal{L}_G + \mathcal{L}_L^{sim/geo}$ (Eq. (2), Eq. (7)/Eq. (5))
16: $\boldsymbol{\theta}_s \leftarrow$ `OPTIMIZER`$(\boldsymbol{\theta}_s, \nabla_{\boldsymbol{\theta}_s}\mathcal{L})$
17: **end for**
18: $\boldsymbol{\theta}_t \leftarrow \lambda\boldsymbol{\theta}_t + (1-\lambda)\boldsymbol{\theta}_s$
19: **end for**
20: **return** $\boldsymbol{\theta}_t$

Algorithm 2 Algorithm 1 edit for multi-crop

4: Sample $\boldsymbol{w}^1 = [\boldsymbol{w}^1_{geo}, \boldsymbol{w}^1_{pho}]$ from $\mathcal{D}^G_{aug_1}$
5: $\tilde{\boldsymbol{x}}^1 = \mathbf{P}\left(\mathbf{S}(\boldsymbol{x}, \boldsymbol{w}^1_{geo}), \boldsymbol{w}^1_{pho}\right)$
6: Sample $\boldsymbol{w}^2 = [\boldsymbol{w}^2_{geo}, \boldsymbol{w}^2_{pho}]$ from $\mathcal{D}^G_{aug_2}$
7: $\tilde{\boldsymbol{x}}^2 = \mathbf{P}\left(\mathbf{S}(\boldsymbol{x}, \boldsymbol{w}^2_{geo}), \boldsymbol{w}^2_{pho}\right)$
8: **for** $n \in \{3, 4, \cdots, 2+N_L\}$ **do**
9: Sample $\boldsymbol{w}^n = [\boldsymbol{w}^n_{geo}, \boldsymbol{w}^n_{pho}]$ from $\mathcal{D}^L_{aug}$
10: $\tilde{\boldsymbol{x}}^n = \mathbf{P}\left(\mathbf{S}(\boldsymbol{x}, \boldsymbol{w}^n_{geo}), \boldsymbol{w}^n_{pho}\right)$
11: **end for**

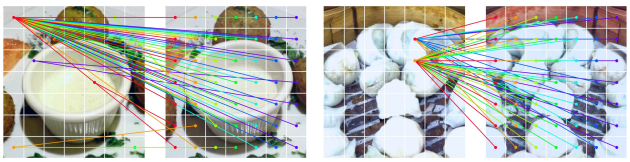

Figure 7. **Visualization of the collapse of the similarity matching function.** This example shows the matchings of the `Similarity` setting trained on Food-101.

A.2. Loss expression with Multi-Crop

The loss expression for $\mathcal{L}_G$ and $\mathcal{L}_L^{sim/geo}$ are slightly affected due to the multi-crop strategy. Only the 2 global-crops are fed to the teacher while the student is fed all crops in $\mathcal{V}$. The definition of the global- and local-representation set are thus slightly changed compared to Section 3.4. Given a student backbone f_s and teacher backbone f_t as well as a set $\mathcal{V} = \{\tilde{\boldsymbol{x}}^1, \tilde{\boldsymbol{x}}^2, \cdots \tilde{\boldsymbol{x}}^{2+N_L}\}$ containing $2 + N_L = |\mathcal{V}|$ augmented views of the same input image, a forward pass of $\tilde{\boldsymbol{x}}^1$ and $\tilde{\boldsymbol{x}}^2$ in the teacher network and a forward pass of all augmentations $\tilde{\boldsymbol{x}}^n$ in the student network results in:

1. two sets of global[4]-representations $\bar{\mathcal{Z}}_s = \{\bar{f}_s(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in \mathcal{V}\}$ and $\bar{\mathcal{Z}}_t = \{\bar{f}_t(\tilde{\boldsymbol{x}}^1), \bar{f}_t(\tilde{\boldsymbol{x}}^2)\}$

2. two sets of local-representations $\mathcal{Z}_s = \{f_s(\tilde{\boldsymbol{x}}) : \tilde{\boldsymbol{x}} \in \mathcal{V}\}$ and $\mathcal{Z}_t = \{f_t(\tilde{\boldsymbol{x}}^1), f_t(\tilde{\boldsymbol{x}}^2)\}$

Given the new definition of the above sets, the only changes in Eq. (2), Eq. (5) and Eq. (7) are the normalization constants. Equation (2) becomes

$$\mathcal{L}_G = \frac{1}{2(N_L+1)} \sum_{\bar{\boldsymbol{z}} \in \bar{\mathcal{Z}}_t} \sum_{\substack{\bar{\boldsymbol{z}}' \in \bar{\mathcal{Z}}_s \\ \tilde{\boldsymbol{x}} \neq \tilde{\boldsymbol{x}}'}} H\left(\bar{h}(\bar{\boldsymbol{z}}), \bar{h}(\bar{\boldsymbol{z}}')\right) \tag{8}$$

Equation (5) becomes

$$\mathcal{L}_L^{sim} = \frac{1}{2(N_L+1)} \sum_{\boldsymbol{z} \in \mathcal{Z}_t} \sum_{\substack{\boldsymbol{z}' \in \mathcal{Z}_s \\ \tilde{\boldsymbol{x}} \neq \tilde{\boldsymbol{x}}'}} L_L^{sim}(\boldsymbol{z}, \boldsymbol{z}') \tag{9}$$

and Equation (7) becomes

$$\mathcal{L}_L^{geo} = \frac{1}{2(N_L+1)} \sum_{\boldsymbol{z} \in \mathcal{Z}_t} \sum_{\substack{\boldsymbol{z}' \in \mathcal{Z}_s \\ \tilde{\boldsymbol{x}} \neq \tilde{\boldsymbol{x}}'}} L_L^{geo}(\boldsymbol{z}, \boldsymbol{z}') \tag{10}$$

B. Training matchings visualized

To build a better intuition of the coherence that the local-representation loss enforces, we illustrate a few pairs of augmentations $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{x}}'$ and show how the local-representations are matched during the training phase both for the `Similarity` (Fig. 8 and Fig. 9) and `Geometric` setting (Fig. 10 and Fig. 11). This is done both for a pair of 2 global-crops as well as a pair of 1 global-crop and 1 local-crop. Using a Swin-T/W=7 [31] backbone results in dense-representations which are downscaled with a factor of 32 compared to the augmentations. Global-crops of size 224×224 result in a dense-representation of spatial dimension 7×7 while local-crops of size 96×96 result in a dense-representation of spatial dimension 3×3.

As mentioned in the paper, the training matchings of the `Similarity` setting depend on the state of the local-representations. The visualization here use a network trained until the last epoch (300) on ImageNet-1k. The images shown are from the validation set of ImageNet-1k. The matchings from Figure 10 (`Geometric` setting) seem to be more well behaved than the matchings from Figure 8 (`Similarity` setting). Moreover, we can observe cases of collapse of the similarity based matching function in Figure 8 even though the network is trained on a large scale dataset (ImageNet-1k). This happens when the photometric transforms applied to both crops are highly different from each other (e.g. dog in 5th row of Figure 8).

C. Collapse of the similarity matching function visualization

The collapse of the similarity matching function when trained on Food-101 [4] can be visualized in Figure 7. Similar effects (though less strong) can be observed in Figure 8 (trained on ImageNet-1k [14]).

D. Correspondence mapping between 2 different images

Although not the goal of our work, we show qualitative results of the correspondence mapping between two different images. We use 2 pairs of 2 images with similar semantics and show a visual alignment between the two. Local-representations from each image are linked to the most similar local-representation in the other image based on their cosine similarity. The 15 assignments with the highest similarities are shown in Figure 12. This is done with Swin-T backbones trained on all 3 different settings (`Vanilla`, `Similarity` and `Geometric`). Overall, all settings seem to show decent alignments of the 2 images though qualitative gains (on the correspondence mapping) can be observed with the additional local cues, especially in the `Geometric` setting.

[4]The local/global terminology used here should not be confused with the local/global terminology of the multi-crop strategy. We refer the reader to Section 3.1 for more information on global- and local-representations.

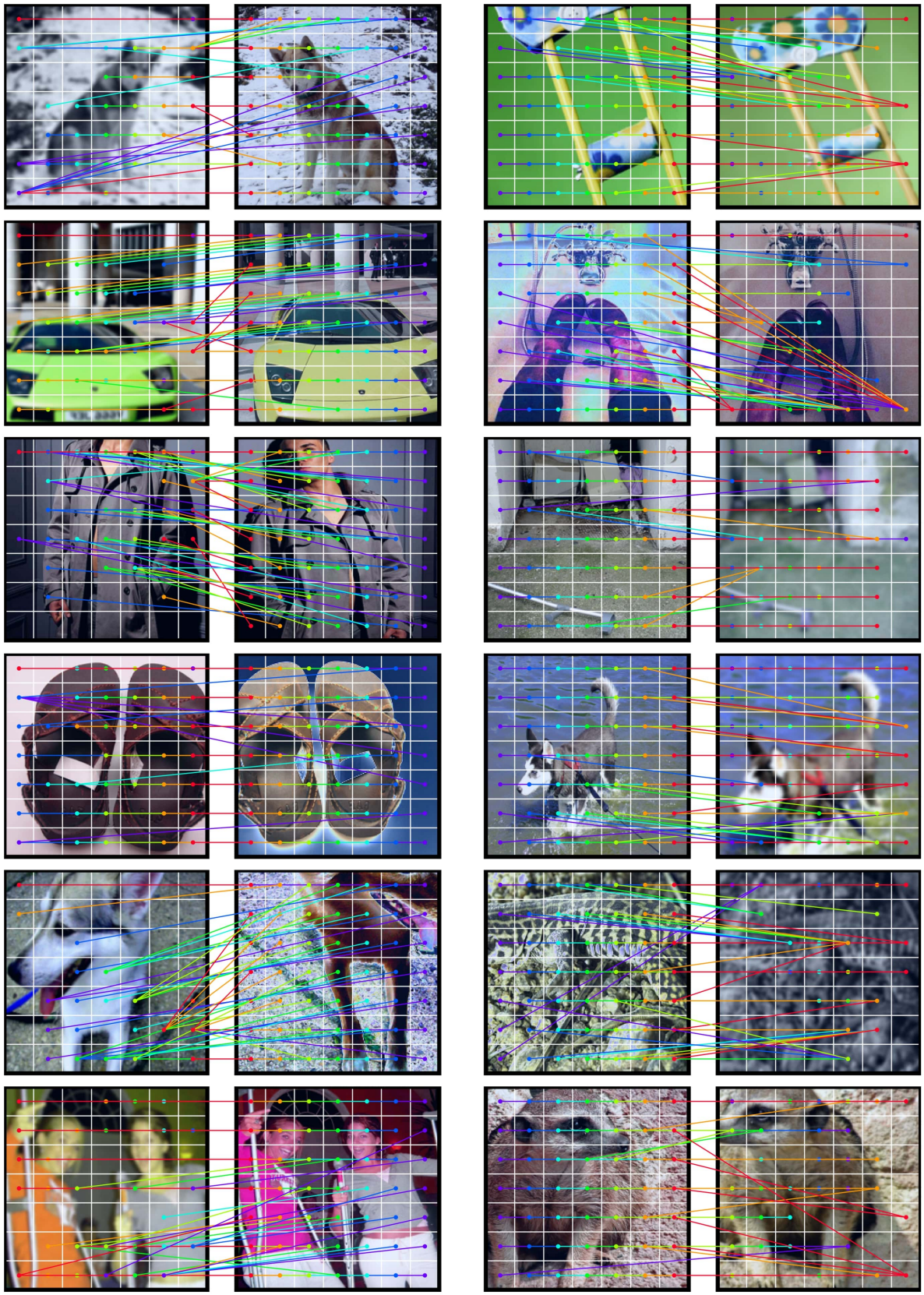

Figure 8. **Visualization of the training matchings between 2 augmentations (both are global-crops) of an input image with the `Similarity` setting.** Each location on a coarse grained grid (corresponding to the area of an output token) on the right view is linked to the best matching location on the left view based on their similarity. Colors are used only to better differentiate different matchings. (extended version of the left part of Fig. 3 in the main paper)

Figure 9. **Visualization of the training matchings between 2 augmentations (1 global- and 1 local-crop) of an input image with the `Similarity` setting.** Each location on a coarse grained grid (corresponding to the area of an output token) on the right view is linked to the best matching location on the left view based on their similarity. Colors are used only to better differentiate different matchings.

Figure 10. **Visualization of the training matchings between 2 augmentations (both are global-crops) of an input image with the `Geometric` setting.** Each location on a coarse grained grid (corresponding to the area of an output token) on the right view is linked to the best matching location on the left view based on their similarity. Colors are used only to better differentiate different matchings. (extended version of the right part of Fig. 3 in the main paper)

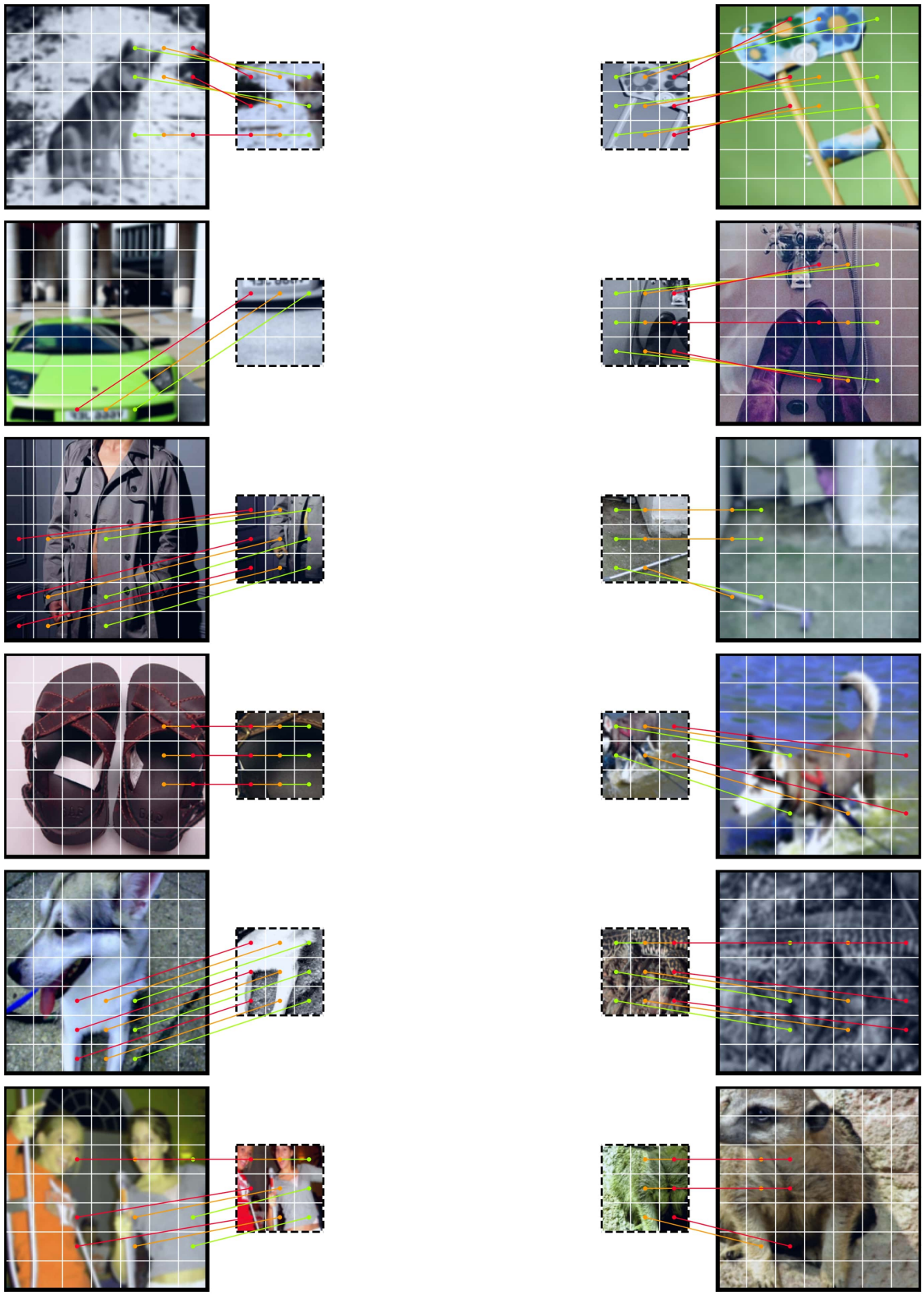

Figure 11. **Visualization of the training matchings between 2 augmentations (1 global- and 1 local-crop) of an input image with the `Geometric` setting.** Each location on a coarse grained grid (corresponding to the area of an output token) on the right view is linked to the best matching location on the left view based on their similarity. Colors are used only to better differentiate different matchings.

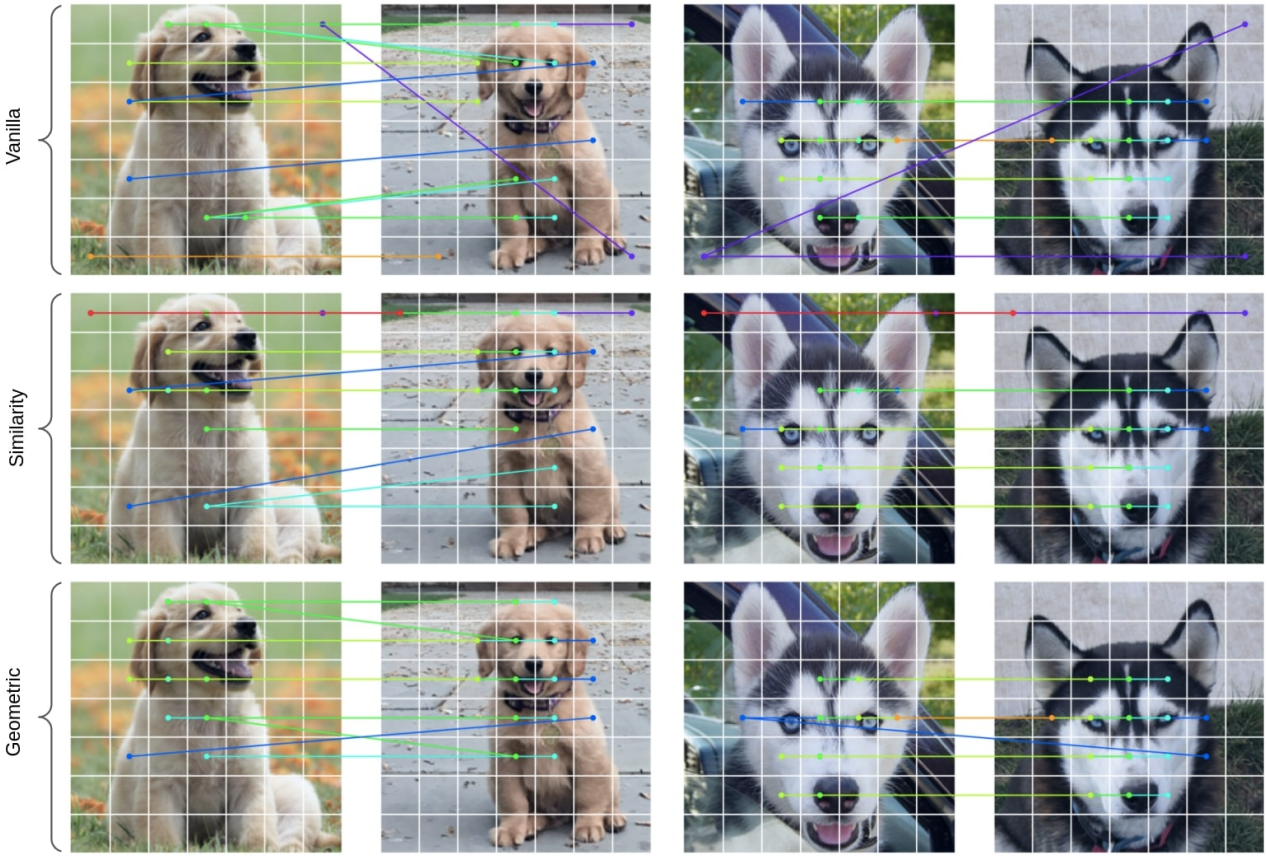

Figure 12. **Visualization of the correspondence mapping between 2 different images.** The top 15 matchings are shown for all 3 settings. The matchings are obtained based on the similarity of the learned local-representations. Colors are used only to better differentiate different matchings.