Guided Semantic Flow
- Supplementary Material -

Sangryul Jeon[1], Dongbo Min[2], Seungryong Kim[3], Jihwan Choe[4], Kwanghoon Sohn[1]

[1]Yonsei University [2]Ewha Womans University [3]Korea University [4]Samsung
{cheonjsr,khsohn}@yonsei.ac.kr, dbmin@ewha.ac.kr,
seungryong_kim@korea.ac.kr, jihwan.choe@samsung.com

Here we describe more details on the implementation of our system in Sec. 1, and the analysis of loss functions in Sec. 2. Also, more qualitative results are provided in Sec. 3 on PF-WILLOW dataset [4], PF-PASCAL dataset [5], Caltech-101 dataset [9], and Spair-71k benchmark [11].

1 Implementation Details

Network architecture Detailed architectures of pruning networks and matching networks are summarized in Table 1. For both networks, we employ an encoder-decoder style architecture which has been adopted in many pixel-level prediction tasks such as disparity estimation, optical flow, or semantic segmentation [3, 2, 10]. To deal with large geometric variations, both networks are designed to incorporate the full set of pairwise scores by setting the kernel size of 'conv1_p' and 'conv1_m' along channel dimension to $(h \times w)$.

Pruning networks ($\mathbf{W}_P$)			
Layer	Kernel size	Ch I/O	Stride
conv1_p	5×5	$h \times w/64$	-
conv1b_p	3×3	64/64	-
pool1_p	2×2	64/64	2
conv2_p	3×3	64/128	-
conv2b_p	3×3	128/128	-
pool2_p	2×2	128/128	2
conv3_p	3×3	128/256	-
conv3b_p	3×3	256/256	-
pool3_p	2×2	256/256	2
conv4_p	3×3	256/512	-
conv4b_p	3×3	512/512	-
upconv5_p	3×3	512/512	2
conv5_p	3×3	768/256	-
conv5b_p	3×3	256/256	-
upconv6_p	3×3	256/256	2
conv6_p	3×3	384/128	-
conv6b_p	3×3	128/128	-
upconv7_p	3×3	128/128	2
conv7_p	3×3	192/64	-
conv7b_p	3×3	64/64	-
conv8_p	5×5	$64/h \times w$	-
sigmoid	-	$h \times w/h \times w$	-

Matching networks ($\mathbf{W}_M$)			
Layer	Kernel size	Ch I/O	Stride
conv1_m	5×5	$h \times w/64$	-
conv1b_m	3×3	64/64	-
pool1_m	2×2	64/64	2
conv2_m	3×3	64/128	-
conv2b_m	3×3	128/128	-
pool2_m	2×2	128/128	2
conv3_m	3×3	128/256	-
conv3b_m	3×3	256/256	-
pool3_m	2×2	256/256	2
conv4_m	3×3	256/512	-
conv4b_m	3×3	512/512	-
upconv5_m	3×3	512/512	2
conv5_m	3×3	768/256	-
conv5b_m	3×3	256/256	-
upconv6_m	3×3	256/256	2
conv6_m	3×3	384/128	-
conv6b_m	3×3	128/128	-
upconv7_m	3×3	128/128	2
conv7_m	3×3	192/64	-
conv7b_m	3×3	64/64	-
conv8_m	5×5	64/2	-

Table 1: Network architecture of the pruning and matching networks.

For each linear layer in both networks, a batch normalization layer is followed, and ReLU is the default activation function. Following [8], 'sigmoid' layer is added at the end of the pruning networks to yield a value of $(0, 1)$.

Before starting training, the pruning networks are initialized to yield $\mathcal{F}(Q; \mathbf{W}_P) \approx \mathbf{1}_{20\times20\times400}$ by setting conv8_p to produce a value of 10, so that the following sigmoid layer yields about 0.99995. Similarly, the matching networks are initialized to yield $\mathcal{F}(C'; \mathbf{W}_M) = \mathbf{0}_{20\times20\times2}$ by setting conv8_m to produce a value of 0. The warping operator $\circ$ in Eq. (12) and (14) is implemented with a bilinear upsampler in [**?**].

Training Details To obtain warped input images paired with the original images, we used random global affine and local TPS transformation as in [13, 15]. Taking the normalized image height and width as 1,

- global affine transformation includes Gaussian random translation in ±0.15, rotation in $\pm10°$, and scaling in [0.8,1.2].
- local TPS transformation includes random translation in ±0.1 regarding the regular-grid control points.

We attempted to utilize the training set of Spair-71k [11] to learn our networks, but the large intra-class geometric differences in SPair-71k [11], including viewpoint, occlusion, and truncation, often make our weakly-supervised learning objectives fragile. Note that similar experimental results are reported in [11], where the performances of weakly-supervised methods [14, 13, 15] are dropped when trained on Spair-71k. The masks of the training image pairs of PF-PASCAL [5] are obtained from PASCAL VOC 2012 segmentation dataset [1]. We used a grid search to set the weighting parameters. We determined $\lambda = 3$ with interval of 1 during the first stage of our training, and $\lambda_{\text{M}} = 5, \lambda_{\text{sm}} = 0.1$ with the intervals of 2 and 0.1 respectively during the second stage. The parameters are selected to yield the best performance on the validation set of PF-PASCAL dataset [5]. We trained our networks using the VLFeat MatConvNet toolbox [12] on an Intel Core i7-3770 CPU with an NVIDIA GeForce GTX TITAN X GPU.

c_P \ c_M	0.1	2	5	7
0.1	9.8	16.4	20.3	18.0
2	18.7	25.1	29.8	26.9
5	22.2	31.8	**33.5**	30.1
7	20.8	24.4	27.1	25.5

(a) c_P, c_M

$\mathcal{N}$ \ $\mathcal{M}$	3×3	5×5	7×7	11×11
3×3	27.7	30.9	29.8	29.3
5×5	30.5	**33.5**	32.8	32.0
7×7	28.3	30.0	29.1	29.8
11×11	25.1	27.7	28.1	28.3

(b) $\mathcal{N}$, $\mathcal{M}$

Table 2: Hyper-parameter study on the testing pairs of SPair-71k benchmark [11] for (a) different variances c_P, c_M and (b) different neighbourhoods $\mathcal{N}$, $\mathcal{M}$. The performances are evaluated by fixing the other hyper-parameters.

2 Discussion

Hyper-parameters Similar to the works of stereo confidence estimation [7, 8], we set ρ to 0.9 at which the density of confident matches used in the loss computation becomes 75% on average. We used the grid search to set the parameters of variances c_P, c_M and neighbourhoods $\mathcal{N}$, $\mathcal{M}$ that produce the best result on the validation split of PF-PASCAL dataset [5]. Table 2 summarizes the performances of our model "Ours w/ResNet" with different variances c_P, c_M and neighbourhoods $\mathcal{N}$, $\mathcal{M}$.

As shown in Table 2 (a), when the variance c_M is set close to 0, this would provide overly dampened correlation volume C' making the matching module too sensitive. The appropriately set variance c_P would impose the spatial smoothness regularization on the interpolated guidance displacements G' so that invalid pixels of G can have correct values. In Table 2 (b), by enlarging the neighbourhood size of $\mathcal{N}$ and $\mathcal{M}$, the matching accuracy improves until 5×5 with longer training and testing times, but larger window sizes reduce matching accuracy due to greater matching ambiguity, as similarly reported in [6].

Loss functions By dividing our training into two stages, the pruning networks converged more stably since the freezed parameters $\mathbf{W}_F$ provide the fixed values of Q and F in Eq.(11) and (12). Nevertheless, our networks still can be converged only with the second stage; the gradients computed from the matching loss $\mathcal{L}_{\text{M}}$ are provided to all pixels i and j of the parameters $\mathbf{W}_F$ (*i.e.* independent to the set $\mathcal{S}$) and prevent the parameters $\mathbf{W}_F$ from being trapped in local minima, such as becoming zero. Note that the matching loss $\mathcal{L}_{\text{M}}$ considers negative samples for reconstructing feature maps which would provide more robustness when learning the parameters $\mathbf{W}_F$. Also note that learning our networks with synthetic transformations at first avoids the negative samples within $\mathcal{M}$ from being poorly defined during training.

In Eq. (11) and (12), for a pixel i and its possible matching candidate j, we consider the matches whose confidence scores are higher than the threshold ρ, such that $Q(i,j) > \rho$. Among them (*i.e.* $\{i,j\} \in \mathcal{S}^*$), the loss $\mathcal{L}_{\text{P}}$ enourages the pruning networks to retain only the matches that satisfy both mutual and geometry consistency constraint. To be specific, $\mathcal{L}_{\text{sil}}$ encourages Q'_{ij} to be close to Q_{ij}, and, at the same time, $\mathcal{L}_{\text{geo}}$ retains only geometrically consistent matches among Q'_{ij}. It is worth to note that $\mathcal{L}_{\text{geo}}$ plays a role of a reconstruction loss by enforcing the displacement map G' to be well-aligned with source and target features. On the other hand, $\mathcal{L}_{\text{sil}}$ acts as a regularization loss by ensuring that the refined confidence volume Q' is not too different from the initial confidence volume Q. By combining two losses, the pruning network can effectively generate the refined confidence volume Q'.

In Eq. (13), the loss is computed for every pixel i. $\mathcal{L}_{\text{M}}$ is formulated based on the intuition that the matching score between the source feature at each pixel i, F^s_i , and the warped target feature $[\tau \cdot F^t]_i$ should be maximized when the correct correspondence field τ is given. Assuming that the point i is a positive

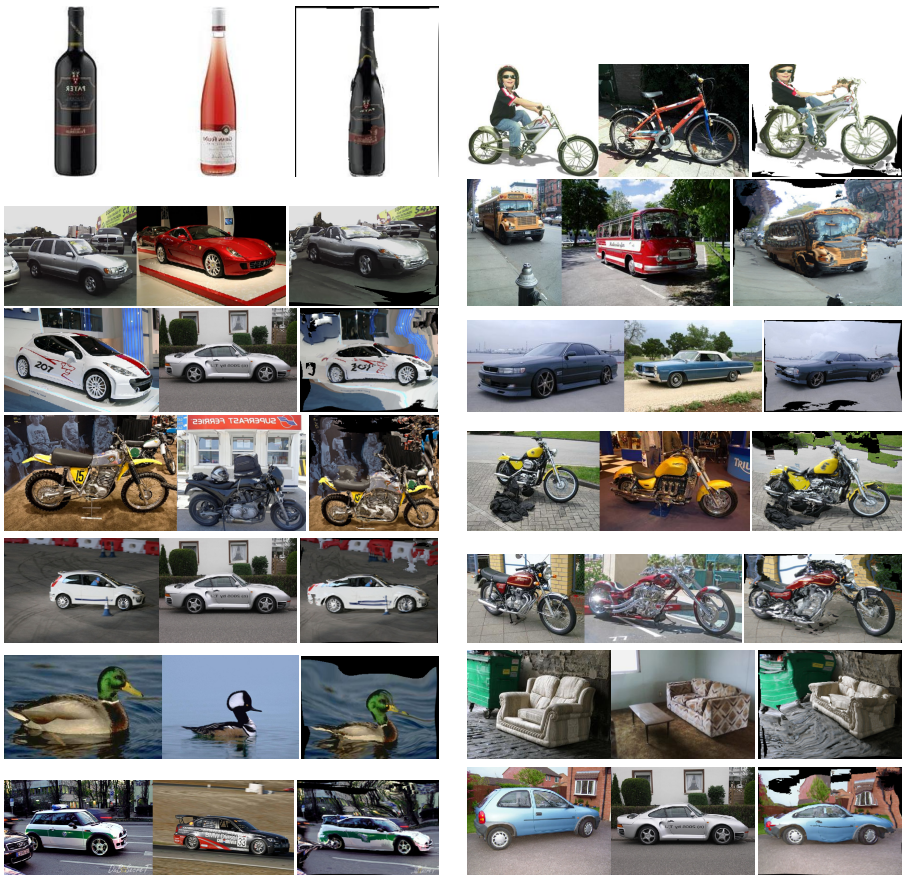
sample with correct transformation and the other points within the local window $\mathcal{M}_i$ centered at the pixel i are negative samples with wrong transformation candidates, this can be treated as a classification problem in that the network can learn a correspondence field τ as a hidden variable for maximizing the scores of positive samples while keeping the scores of negative samples low through the softmax normalization of Eq. (14).

3 More Results

In this section, we show more qualitative results compared to the state-of-the-art approaches on the PF-WILLOW dataset [4], PF-PASCAL dataset [5] in Fig. 1, and Caltech-101 dataset [9], Spair-71k dataset [11] in Fig. 2. The source images are warped to the target images using dense correspondence fields obtained from our model.

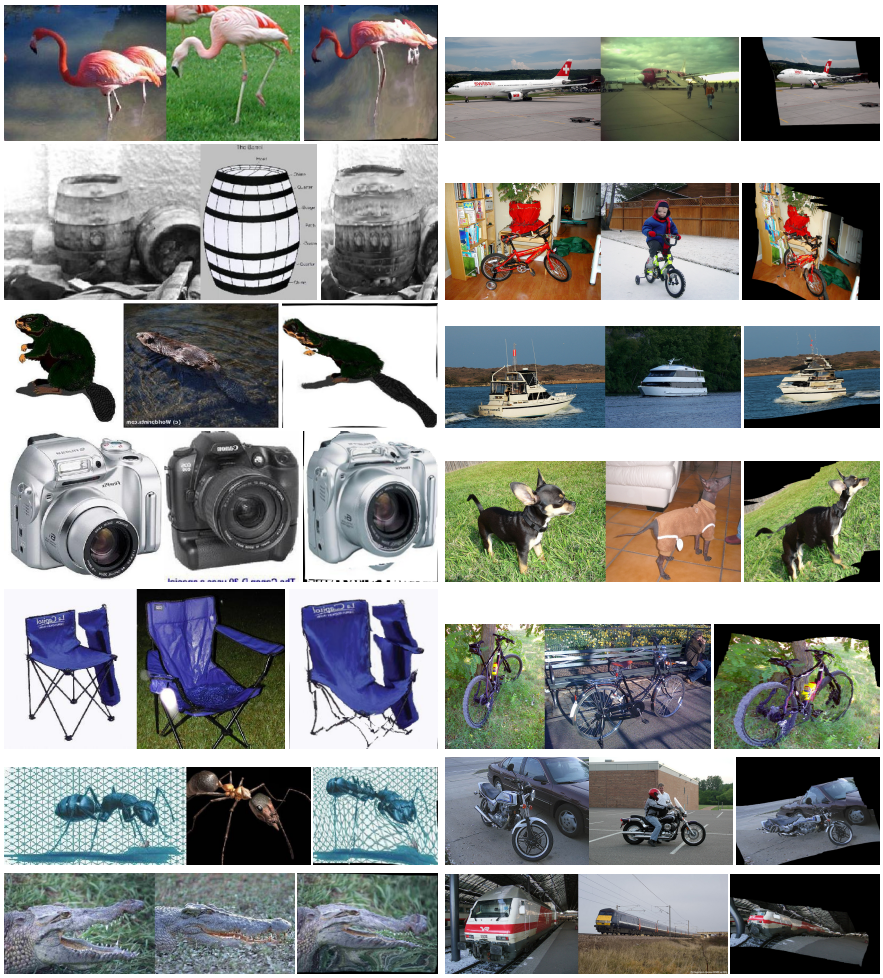
References

1. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision **88**(2), 303–338 (2010)
2. Fischer, P., Dosovitskiy, A., Ilg, E., Häusser, P., Hazırbaş, C., Golkov, V., Van der Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. In: ICCV (2015)
3. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: CVPR (2017)
4. Ham, B., Cho, M., Schmid, C., Ponce, J.: Proposal flow. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3475–3484 (2016)
5. Ham, B., Cho, M., Schmid, C., Ponce, J.: Proposal flow: Semantic correspondences from object proposals. IEEE Trans. PAMI (2017)
6. Kim, S., Lin, S., Jeon, S., Min, D., Sohn, K.: Recurrent transformer networks for semantic correspondence. In: Advances in Neural Information Processing Systems (2018)
7. Kim, S., Kim, S., Min, D., Sohn, K.: Laf-net: Locally adaptive fusion networks for stereo confidence estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 205–214 (2019)
8. Kim, S., Min, D., Kim, S., Sohn, K.: Unified confidence estimation networks for robust stereo matching. IEEE Transactions on Image Processing **28**(3), 1299–1313 (2019)
9. Li, F.F., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Trans. PAMI **28**(4), 594–611 (2006)
10. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
11. Min, J., Lee, J., Ponce, J., Cho, M.: Hyperpixel flow: Semantic correspondence with multi-layer neural features. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
12. Online.: http://www.vlfeat.org/matconvnet/
13. Rocco, I., Arandjelović, R., Sivic, J.: Convolutional neural network architecture for geometric matching. In: CVPR (2017)
14. Rocco, I., Cimpoi, M., Arandjelović, R., Torii, A., Pajdla, T., Sivic, J.: Neighbourhood consensus networks. In: Advances in Neural Information Processing Systems. pp. 1658–1669 (2018)
15. Seo, P.H., Lee, J., Jung, D., Han, B., Cho, M.: Attentive semantic alignment with offset-aware correlation kernels. In: ECCV (2018)



(a) Our result on PF-WILLOW dataset [4] (b) Our result on PF-PASCAL dataset [5]

Fig. 1: Qualitative results of the semantic alignment on the testing pair of PF-WILLOW dataset [4] and PF-PASCAL dataset [5].



(a) Our result on Caltech-101 dataset [9] (b) Our result on Spair-71k dataset [11]

Fig. 2: Qualitative results of the semantic alignment on the testing pair of Caltech-101 dataset [9] and Spair-71k dataset [11].