Supplementary materials for Reversing the cycle: self-supervised deep stereo through enhanced monocular distillation

Filippo Aleotti*[1], Fabio Tosi*[1], Li Zhang**[2], Matteo Poggi[1], and Stefano Mattoccia[1]

[1] University of Bologna, Viale del Risorgimento 2, Bologna, Italy
[2] China Agricultural University, Beijing, China

1 Extended Quantitative Comparison

In this section, we report more quantitative comparisons with self-supervised state-of-the-art frameworks on both non-occluded (*Noc*) and all (*All*) regions on the KITTI 2015 training set, as well more detailed cross-validation on DrivingStereo. Both are reported in a more compact form in the main paper for the sake of space.

Table 1 is provided as a complement to Table 5 in the main paper to allow an exhaustive evaluation. It further confirms that stereo networks trained with our proxies notably outperform existing approaches. It is worth to notice how our models achieve a lower percentage of errors (D1) compared to strategies using raw LiDAR measurements or deploying stereo videos at training time [11].

Table 2 reports iResNet [9] and GWCNet [5] experiments across KITTI (K) and DrivingStereo (DS) datasets, as complement to Table 4 in the main paper. In general, GWCNet and iResNet confirm the trend observed with StereoDepth and PSMNet, with GWCNet resulting slightly more effective when transferred from K to DS and vice-versa.

2 Configuration details

Traditional Stereo Methods. Here, we report details concerning both SGM [6] and WILD [14] used as traditional stereo matchers. For the SGM [6] algorithm, we compute initial matching costs applying a 9×7 census transform and using the Hamming distance on pixel vectors. We set parameters $P1$ and $P2$ to 7 and 17, respectively. The cost aggregation step has been performed along 8 independent paths, while the disparity range is set to $[0, 192]$.

For WILD [14], we adopt the publicly available code provided by the authors. Differently from the original settings used in [14], we apply a few modifications. In particular, we select a different combination of traditional confidence measures from [10] to retain highly accurate points from the block matching algorithm.

* Joint first authorship
** Work done while at University of Bologna.

				Lower is better		Higher is better		
Method	Region	RMSE	RMSE log	D1	EPE	δ <1.25	$\delta < 1.25^2$	$\delta < 1.25^3$
Godard et al.[3] (stereo)	Noc	4.392	0.146	9.19	-	0.942	0.978	0.989
Yang et al.[16]	Noc	-	-	8.95	1.61	-	-	-
Tonioni et al. [12]	Noc	-	-	8.51	1.48	-	-	-
Zhou et al.[20]	Noc	-	-	8.35	1.44	-	-	
Lai et al.[7]	Noc	4.168	0.149	8.22	1.40	0.947	0.979	0.990
Yang et al.[16] ⋄	Noc	-	-	7.70	1.46	-	-	-
Li and Yuan [8] ⋄	Noc	-	-	6.65	1.73	-	-	-
Ours (Stereodepth) †	Noc	3.894	0.116	4.21	1.06	0.971	0.988	0.993
Ours (GWCNet [5])	Noc	3.623	0.111	3.78	1.02	0.974	**0.989**	0.993
Ours (PSMNet [1])	Noc	3.772	0.115	3.68	**0.99**	0.974	0.988	0.993
Ours (iResNet [9])	Noc	**3.472**	**0.107**	**3.64**	**0.99**	**0.975**	**0.989**	**0.994**
Smolyanskiy et al. [11] (LiDAR) †	All	-	-	15.00	-	-	-	-
Smolyanskiy et al. [11] (photo)	All	-	-	12.90	-	-	-	-
Godard et al.[3] (stereo)	All	5.742	0.202	10.80	-	0.928	0.966	0.980
Yang et al.[16]	All	-	-	10.03	1.89	-	-	-
Zhou et al.[20]	All	-	-	9.41	-	-	-	
Smolyanskiy et al. [11] (photo + LiDAR) †	All	-	-	8.80	-	-	-	-
Lai et al.[7]	All	4.186,	0.157	8.62	1.46	0.946	0.979	0.990
Yang et al.[16] ⋄	All	-	-	8.79	1.84	-	-	-
Tonioni et al. [12]	All	-	-	8.78	1.48	-	-	-
Li and Yuan [8] ⋄	All	-	-	8.21	1.73	-	-	-
Zhou et al.[20] †	All	-	-	7.29	-	-	-	
Wang et al.[15] (stereo only)	All	4.187	0.135	7.07	-	0.955	0.981	0.990
Wang et al.[15] (ego motion)	All	3.488	0.121	6.43	-	0.964	0.985	0.992
Zhong et al.[18]	All	4.857	0.165	6.42	-	0.956	0.976	0.985
Wang et al.[15] (stereo videos)	All	3.404	0.121	5.94	-	0.965	0.984	0.992
Zhong et al.[19] *	All	(**3.176**)	(0.125)	(5.14)	-	(0.967)	-	-
Ours (Stereodepth)	All	3.882	0.117	4.39	1.07	0.971	0.988	**0.993**
Ours (GWCNet [5])	All	3.614	0.111	3.93	1.04	0.974	**0.989**	**0.993**
Ours (iResNet [9])	All	3.464	**0.108**	3.88	1.02	**0.975**	0.988	**0.993**
Ours (PSMNet [1])	All	3.764	0.115	**3.85**	**1.01**	0.974	0.988	**0.993**

Table 1. Quantitative results on the KITTI 2015 training set. Ours indicates networks trained using MCN-BM/W-ARC labels, † using LiDAR supervision and ⋄ pre-training on synthetic datasets. ∗ indicates models trained on the same KITTI 2015 data, therefore not directly comparable with other methods.

Specifically, we used Disparity Agreement (DA), Disparity Scattering (DS), Left-Right Consistency (LRC), Average Peak Ration Measure (APKR), Uniqueness Constraint (UC) and MED (Difference with Median) by setting $\delta_0 = 0$ and $\delta_1 = 0.4$, i.e. filtering out all wrong matches and keeping only very confident pixels. We found out that such configuration allows keeping a larger number of depth values while preserving accuracy. As for SGM, we use $[0, 192]$ as disparity range.

Monocular Completion Network. To train MCN, we follow the protocol described in [13], using Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate has been set to be 10^{-4} and halved after $180k$ and $240k$, respectively. We used the same parameters from [13] for the loss function which includes an image reconstruction loss based on stereo images, a disparity smoothness loss and, finally, a proxy supervision loss. We train with a batch size of 6 for $300k$ iterations, on 640×192 and 256×512 random crops respectively for KITTI and DrivingStereo. In our experiments, neither pre-training on other datasets nor post-processing procedures have been performed.

To generate multiple inferences during the consensus phase, we perform image augmentation and random sampling of input points, as described in the main

Backbone	Supervision	Source → Target					
		DS → DS		K → DS		DS → K	
		D1(%)	EPE	D1(%)	EPE	D1(%)	EPE
Stereodepth	MCN-BM/W-ARC	2.47	0.94	2.97	0.96	5.64	1.22
PSMNet	MCN-BM/W-ARC	**1.87**	**0.86**	2.32	0.88	5.16	1.17
GWCNet	MCN-BM/W-ARC	2.04	0.89	**2.15**	**0.82**	**4.94**	**1.14**
iResNet	MCN-BM/W-ARC	2.63	0.96	2.70	0.91	5.75	1.21

Table 2. Cross-validation analysis. We tested, on the target dataset, models trained on the source one. Notice that no fine-tuning on the target dataset is performed in case of cross-validation.

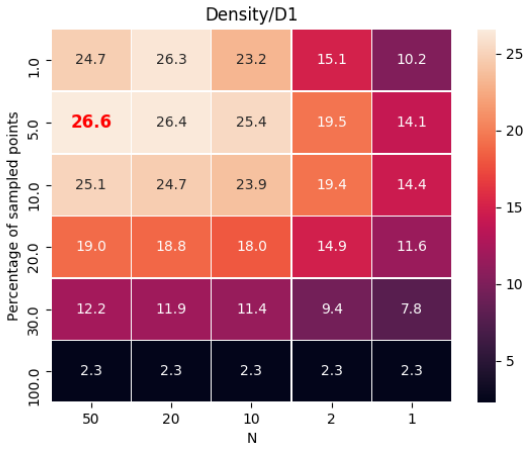

Fig. 1. Random Sampling Analysis. We asses the impact of sparse disparity points given as input to the MCN network during the distillation phase on the KITTI 2015 training set. Each cell contains the ratio between the density of valid points and D1 error. Brighter colors encode higher ratio values.

paper. In particular, we apply the same color augmentation of [13] on RGB images, random horizontal flip with a probability of 0.5 (notice that in case of flipping the final prediction has to be flipped again to obtain a disparity map aligned with the reference image) and random point selection. Concerning sampling at testing time, we intuitively expect the density of sampled points to impact on performance: using sparser inputs at test time w.r.t. the training phase would degrade the accuracy, while a higher density of pixels results in lower randomness, thus thwarting the consensus mechanism.

We empirically found out that the best choice consists of a broader set of points at test time w.r.t. during training, but yet small if compared to the total amount of available pixels. Figure 1 reports a thorough analysis concerning the impact of points at inference time for MCN-BM/W-ARC. In particular, we evaluate both the density (i.e. the number of valid pixels) and the D1 error metric of the disparities output of the consensus mechanism, by varying the number N

Network	Dataset	Batch	Decay Ep.	Total Ep.
Stereodepth	K	12	90	100
PSMNet	K	2	8	11
iResNet	K	4	21	24
GWCNet	K	2	6	8
Stereodepth	DS	12	40	50
PSMNet	DS	2	3	4
iResNet	DS	4	5	8
GWCNet	DS	2	2	3

Table 3. Stereo network training configuration.

of inferences and the percentage of sampled pixels. Each entry in the figure represents the ratio between the density and D1. This metric assumes a higher value in case of high density and low D1, so it is suited for selecting the best configuration among those considered. It can be noticed that, as expected, the consensus mechanism fails in case of configurations with low randomness (i.e. few multiple estimates or a large set of pixels in input). Given these considerations, we choose for our experiments the configuration $N = 50$ and 0.05 of sampled pixels. A similar analysis has also been conducted for the SGM/L counterpart.

Stereo networks. To train all the stereo architectures, we define a loss $\mathcal{L}$ obtained as the mean of the multi-scale losses $\mathcal{L}_s$, where each term is weighted by a factor of γ set to $0.2, 0.6, 1.0$ respectively for the $\frac{1}{4}$, $\frac{1}{2}$ and full-resolution predictions (in case of iResNet, we keep these values also for the layers in the refinement module). At each scale s, the differences between predictions $\mathcal{D}^S$ and proxy values $\mathcal{D}^{\mathbb{P}}$ is computed using the smooth L1 loss H. Notice that only valid points (i.e. those preserved by the filtering procedure) in $\mathcal{D}^{\mathbb{P}}$ are taken into account when calculating the loss. We refer to this set of valid points as V. We set the maximum disparity as 192. Table 2 reports the batch size, the total number of epochs and the decaying epoch (i.e., the epoch in which the learning rate has been halved) for each network both on K and DS. Proxy-supervised models have been trained with 640×192 and 256×512 random crops respectively on K and DS.

In case of *PHOTO* configuration (i.e. training exploiting only RGB images), we leverage only image reconstruction: we rely on the framework released by [4], where the original monocular network has been replaced by Stereodepth or PSMNet. We train for 100 and 7 epochs Stereodepth and PSMNet, respectively, halving the learning rate at 90 and 6. Input images are resized to 640×192. Finally, all the Stereodepth models, as well as all the other architectures, have been trained from scratch, i.e. not starting from ImageNet pretraining [2] nor synthetic datasets.

$$\mathcal{L} = \frac{1}{S}\sum_{s}^{S}\gamma_s\mathcal{L}_s \qquad \mathcal{L}s(\mathcal{D}^S, \mathcal{D}^{\mathbb{P}}) = \frac{1}{V}\sum_{V} H(\mathcal{D}^S - \mathcal{D}^{\mathbb{P}}) \qquad H(x) = \begin{cases} |x| - 0.5, & |x| >= 1 \\ \frac{x^2}{2}, & |x| < 1 \end{cases}$$

3 Qualitative results

We report more qualitative results, depicting both proxy labels sourced by MCN as well as disparity estimates by stereo networks trained on such annotations.

MCN and distillation. Figure 2 reports examples from the KITTI 2015 training set, showing from left to right the reference image, provided ground truth map and proxy labels obtained by MCN-BM/W-ARC and MCN-SGM/L-ARC. Figure 3 collects few examples of distilled annotations from the DrivingStereo dataset, showing respectively from left to right the reference image, provided ground truth and distilled labels using MCN-BM/W-ARC.

Stereo networks results. Figure 4 shows results with stereo architectures trained on our proxy labels, on two stereo pairs from KITTI 2015. On top, we report the reference image and provided ground truth map, followed by four disparity maps estimated by the four networks considered in our experiments. Figure 5 shows the same for a stereo pair from the DrivingStereo dataset.

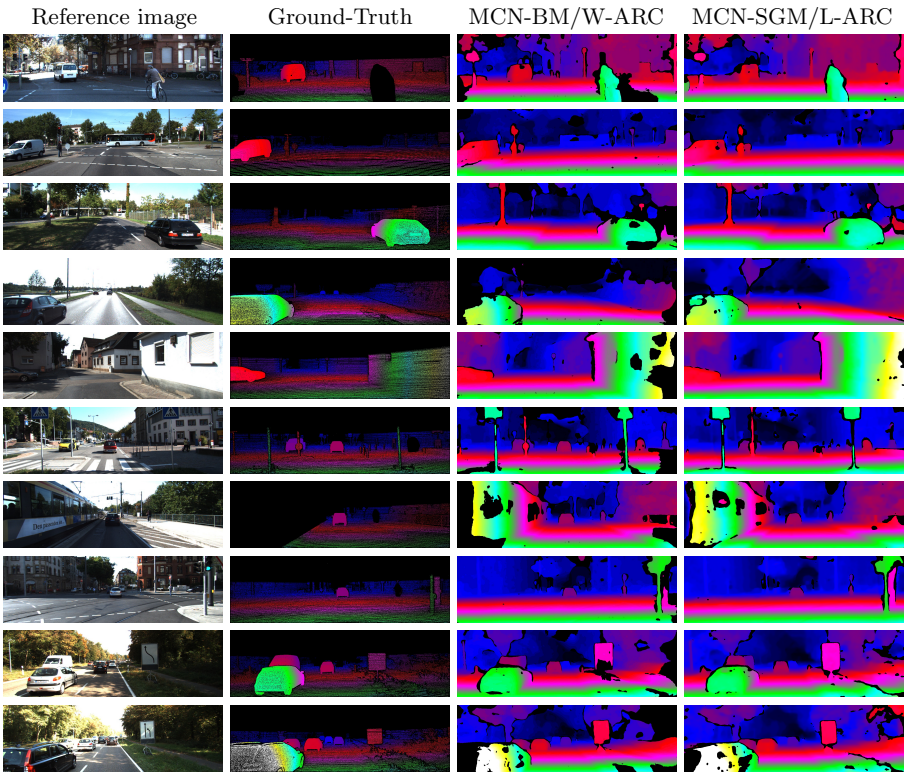

Fig. 2. Distilled proxies on the KITTI 2015 training set. From left to right, the reference image, the ground-truth and our proxies distilled using the consensus mechanism starting from BM/W and SGM/L.

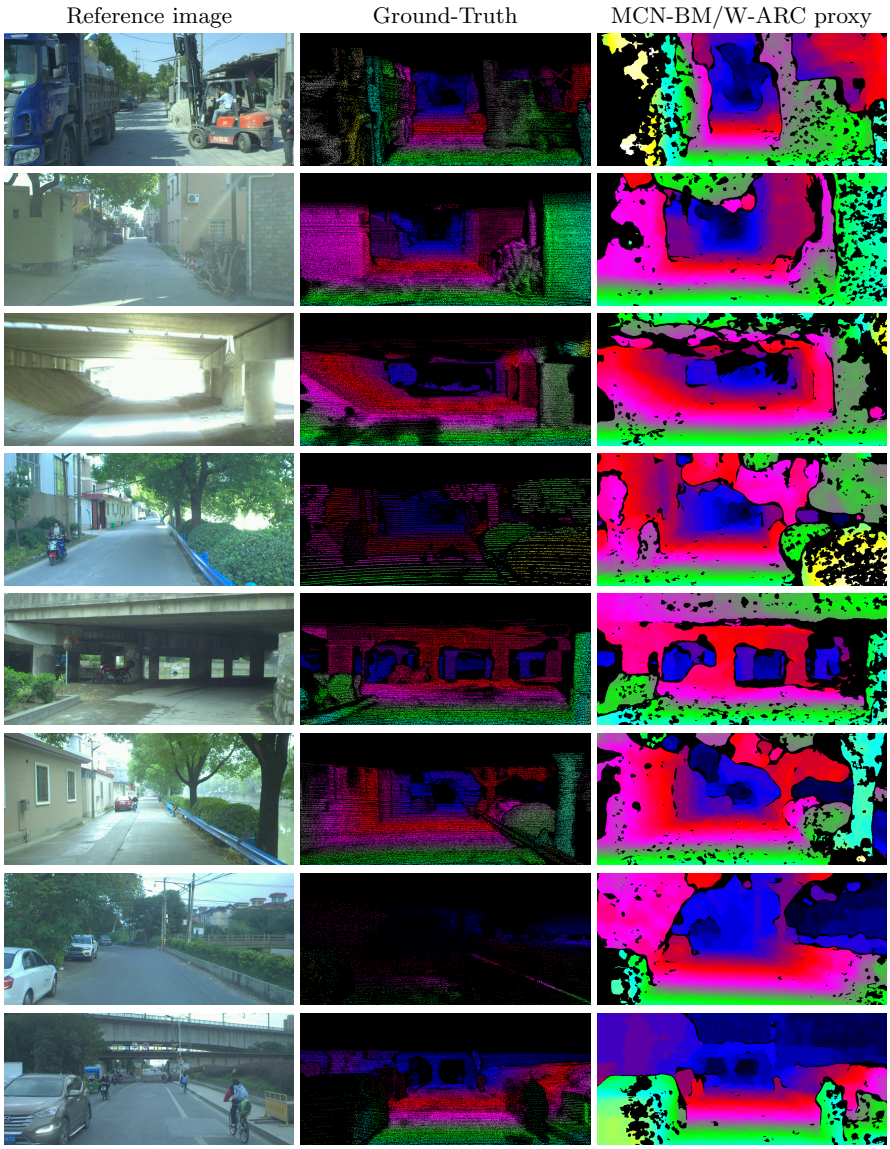

Fig. 3. Distilled proxies on DrivingStereo. From left to right, the reference image, the ground-truth and our proxies distilled using the consensus mechanism starting from BM/W.

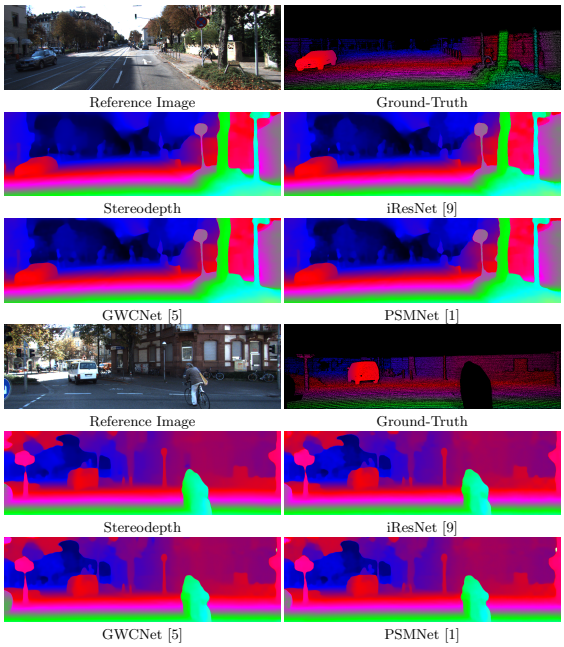

Fig. 4. Qualitative results on KITTI 2015. On the top row, reference image and ground truth map. Then, disparity maps estimated by four different architectures, trained on our proxies.

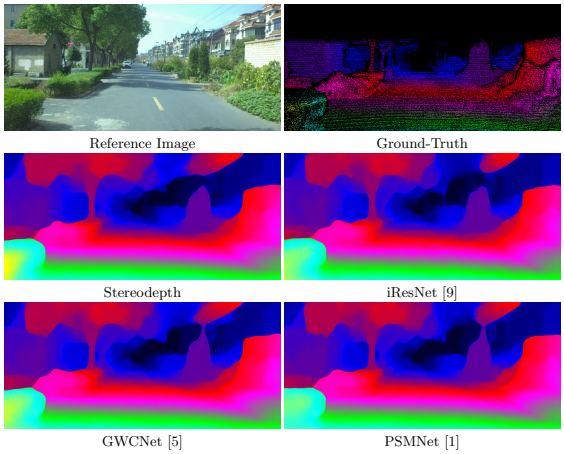

Fig. 5. Qualitative results on DrivingStereo. On the top row, reference image and ground truth map. Then, disparity maps estimated by four different architectures, trained on our proxies.

Generalization. We also show additional qualitative results concerning generalization capability. Figures 6 and 7 collects two examples each, respectively from Middlebury v3 and ETH3D datasets. From left to right, we report the reference image and ground truth map. Then, we show disparity maps estimated by state-of-the-art **self-supervised** frameworks, i.e. Wang et al. [15] and Lai et al. [7], followed by ours. As last, we also show estimation by [17] as state-of-the-art **supervised** network. In both cases, networks have been trained or fine-tuned on KITTI, but never on Middlebury nor ETH3D. It can be perceived how our results are much more detailed than those by other self-supervised techniques and, sometimes, even better than those produced by a supervised network.

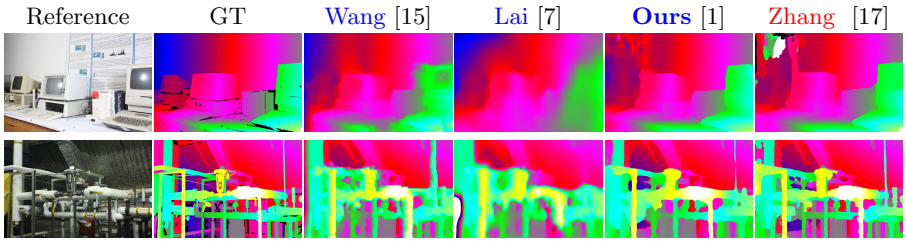

Fig. 6. Qualitative results on Middlebury v3. Methods in **blue** are self-supervised, while in **red** are supervised with ground-truth. We test networks trained on the KITTI dataset to estimate disparity on Middlebury v3, framing completely different environments.

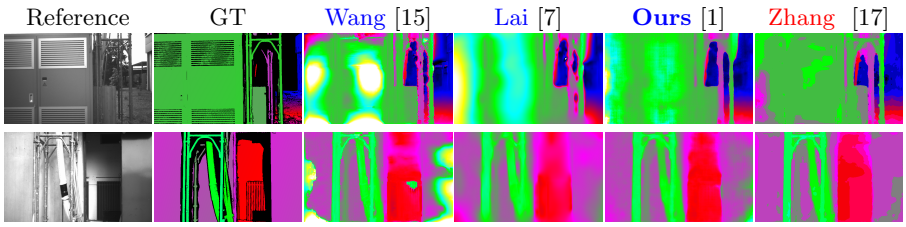

Fig. 7. Qualitative results on ETH3D. Methods in **blue** are self-supervised, while in **red** are supervised with ground-truth. We test networks trained on the KITTI dataset to estimate disparity on ETH3D, framing completely different environments.

4 Augmentation strategy for occlusions

Neither traditional stereo algorithms nor image reconstruction losses provide guidance on occlusions, i.e. where pixels do not have matches on the other view. Both stereo and monocular networks, when self-supervised from stereo pairs, fail to explain such regions because of the lack of supervision there, predicting inconsistent values for occluded pixels (e.g. treating the occlusion as part of the foreground object).

In this section, we detail our augmentation strategy tailored to handle occlusions, coupling with monocular networks only. Occlusions occur near depth boundaries, opposite in position w.r.t. the other view (i.e. on the left image, they occur behind foreground objects, on their left), as shown on top of Figure 8. By flipping the image, occluded regions are mirrored and thus appear on the right of foreground elements, as shown in the second row of Figure 8. By randomly feeding the network at training time with a flipped image we can provide supervision for regions that, otherwise, would never receive it (e.g. left border of the car in figure). Thus, we force the network to handle both object boundaries in order to minimize the loss function. This strategy alleviates the occlusion artefacts at testing time since the network has learned to explain such critical regions with plausible values.

We point out that this strategy is effective with monocular networks only. Indeed, by flipping the images, we also invert the relative order between reference and target image (i.e. left to right before, right to left after flipping). In order to preserve the usual direction of matching of the stereo network (that is, the correspondent pixel, if it exists, is placed further left along the epipolar line) the right and the left images have to be switched, making the right the new reference image. However, this moves the occlusion on the left again, as depicted at the bottom of Figure 8.

Fig. 8. Augmentation for occlusion. Given a stereo pair (first row), the occluded area appears on the left side. If horizontal flip without switching is applied (second row) both to RGB and supervision, the occlusion moves on the right side. Instead, by flipping and switching the occlusion returns on the left (third row).

References

1. Chang, J.R., Chen, Y.S.: Pyramid stereo matching network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (June 2018)
2. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
3. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2017)
4. Godard, C., Mac Aodha, O., Brostow, G.J.: Digging into self-supervised monocular depth estimation. In: IEEE international conference on computer vision (ICCV). IEEE (2019)
5. Guo, X., Yang, K., Yang, W., Wang, X., Li, H.: Group-wise correlation stereo network. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3273–3282. IEEE (2019)
6. Hirschmuller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. vol. 2, pp. 807–814. IEEE, IEEE (2005)
7. Lai, H.Y., Tsai, Y.H., Chiu, W.C.: Bridging stereo matching and optical flow via spatiotemporal correspondence. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2019)
8. Li, A., Yuan, Z.: Occlusion aware stereo matching via cooperative unsupervised learning. In: ACCV. Springer (2018)
9. Liang, Z., Feng, Y., Guo, Y., Liu, H., Chen, W., Qiao, L., Zhou, L., Zhang, J.: Learning for disparity estimation through feature constancy. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (June 2018)
10. Poggi, M., Tosi, F., Mattoccia, S.: Quantitative evaluation of confidence measures in a machine learning world. In: IEEE international conference on computer vision (ICCV). pp. 5228–5237 (2017)
11. Smolyanskiy, N., Kamenev, A., Birchfield, S.: On the importance of stereo for accurate depth estimation: an efficient semi-supervised deep neural network approach. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. IEEE (2018)
12. Tonioni, A., Poggi, M., Mattoccia, S., Di Stefano, L.: Unsupervised adaptation for deep stereo. In: The IEEE International Conference on Computer Vision (ICCV). IEEE (Oct 2017)
13. Tosi, F., Aleotti, F., Poggi, M., Mattoccia, S.: Learning monocular depth estimation infusing traditional stereo knowledge. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2019)
14. Tosi, F., Poggi, M., Tonioni, A., Di Stefano, L., Mattoccia, S.: Learning confidence measures in the wild. In: BMVC. BMVA (Sept 2017)
15. Wang, Y., Wang, P., Yang, Z., Luo, C., Yang, Y., Xu, W.: Unos: Unified unsupervised optical-flow and stereo-depth estimation by watching videos. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 8071–8081. IEEE (2019)
16. Yang, G., Zhao, H., Shi, J., Deng, Z., Jia, J.: Segstereo: Exploiting semantic information for disparity estimation. In: 15th European Conference on Computer Vision (ECCV). Springer (2018)

17. Zhang, F., Prisacariu, V., Yang, R., Torr, P.H.: Ga-net: Guided aggregation net for end-to-end stereo matching. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 185–194. IEEE (2019)
18. Zhong, Y., Li, H., Dai, Y.: Self-supervised learning for stereo matching with self-improving ability. arXiv preprint arXiv:1709.00930 (2017)
19. Zhong, Y., Li, H., Dai, Y.: Open-world stereo video matching with deep rnn. In: ECCV. Springer (2018)
20. Zhou, C., Zhang, H., Shen, X., Jia, J.: Unsupervised learning of stereo matching. In: The IEEE International Conference on Computer Vision (ICCV). IEEE (October 2017)