Appendix

A. Generalization Performance with the RGB Based Cost Volume

As mentioned in Section 1 of the main text, RGB images have consistent representations across stereo views, and traditional stereo matching methods generalize well across different domains. To further validate the importance of stereo representation consistency to generalization, we design a network that concatenates the RGB image pair directly to construct the cost volume, which is served as the input to the network. Compared to the PSMNet [4] baseline, we simply replace the concatenated feature-based cost volume with RGB based cost volume, as shown in Figure 5. To match the input size requirement of the cost aggregation network ($D \times 64 \times \frac{H}{4} \times \frac{W}{4}$), a 3D convolution layer is applied to the RGB based cost volume ($D \times 6 \times H \times W$) that generates the cost volume as in PSMNet. We eliminate as many learnable parameters from the feature extractor as possible and demonstrate that once the stereo feature consistency is highly satisfied, the training of the cost aggregation network doesn't affect the generalization performance of the whole stereo network. We present the generalization performance on different datasets in Figure 6. The error rate above a given threshold is used as the error metric. Constructing the cost volume directly on RGB image pairs provides a significant performance improvement over the cost volume with inconsistent stereo features (given by the feature extractor of PSMNet) on all datasets. This verifies our motivation to encourage stereo feature consistency for better generalization ability.

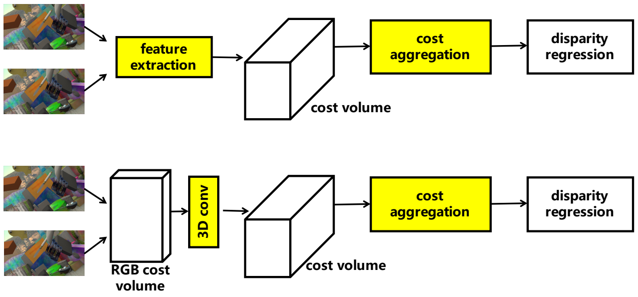

Figure 5. The architecture of the original PSMNet with the concatenated feature cost volume (above) and its variant with the RGB based cost volume (below). Structures that contain learnable parameters are highlighted.

B. Comparison with Pre-training method

The pre-training technique could obtain features that adapt to multiple domains and transfer to downstream tasks. For example, SAND [41] utilizes metric learning to enforce feature consistency between stereo views, and uses these pre-trained features to fine-tune the stereo matching network. This pre-training method treats consistent feature learning and stereo network training as a two-stage framework. Nevertheless, our method jointly trains the stereo network and enforces the stereo feature consistency. We compare the feature consistency and the disparity error of the pre-training method [41] and our joint training one in the whole training process. Our method produces more consistent features and lower disparity errors.

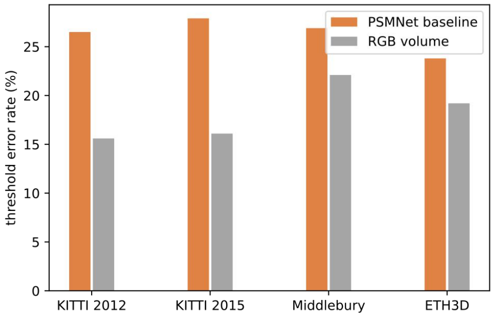

Figure 6. Generalization performance with cost volume construction using learned features and RGB images. Threshold error rates (%) are used (KITTI: 3.0, Middlebury: 2.0, ETH3D: 1.0).

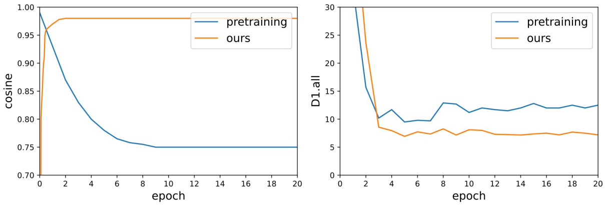

Figure 7. Comparison of enforcing feature consistency in a separate pre-training stage [41] and jointly with stereo network training (ours). We evaluate both the stereo feature consistency and the generalization performance on the KITTI 2015 dataset. The cosine similarity and the D1_all error rate are used to measure the stereo feature consistency and the disparity accuracy, respectively.

C. Channel Visualization of Feature Consistency

We visualize the feature vectors of some matching pixels and show that our method successfully improves the stereo feature consistency over the baseline method. Channel-wise values of left and right feature vectors are shown in Figures 8 to 11, covering the training set and several unseen datasets. The PSMNet [4] baseline extracts feature representations that are inconsistent between stereo viewpoints, while our method can maintain the stereo feature consistency across different domains.

D. More Qualitative Results

In this section, we provide more qualitative results of baseline models and models with our method. PSMNet [4] and GANet [57] are selected as our baseline models. All models are trained on the SceneFlow dataset and evaluated on four unseen domains. As shown in Figures 12 to 15, models with our method generalize better than their counterparts.

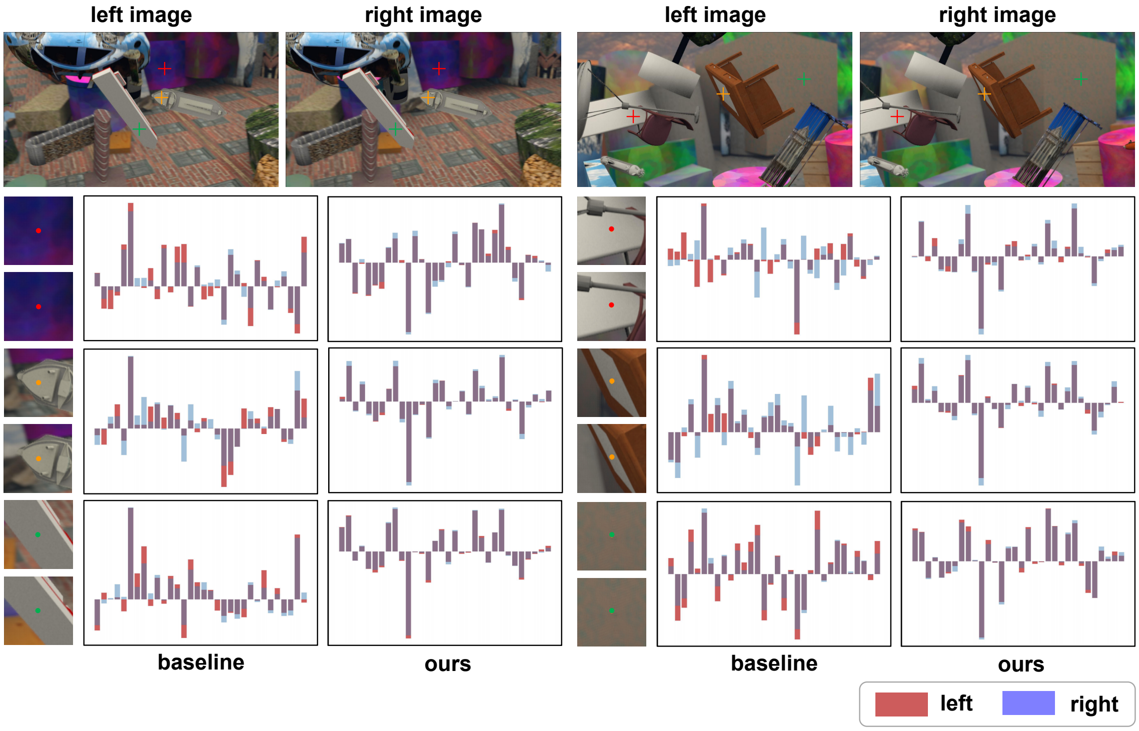

Figure 8. Channel-wise values of matching feature representations from the SceneFlow training set. Matching pixels are zoomed in to make them more visible.

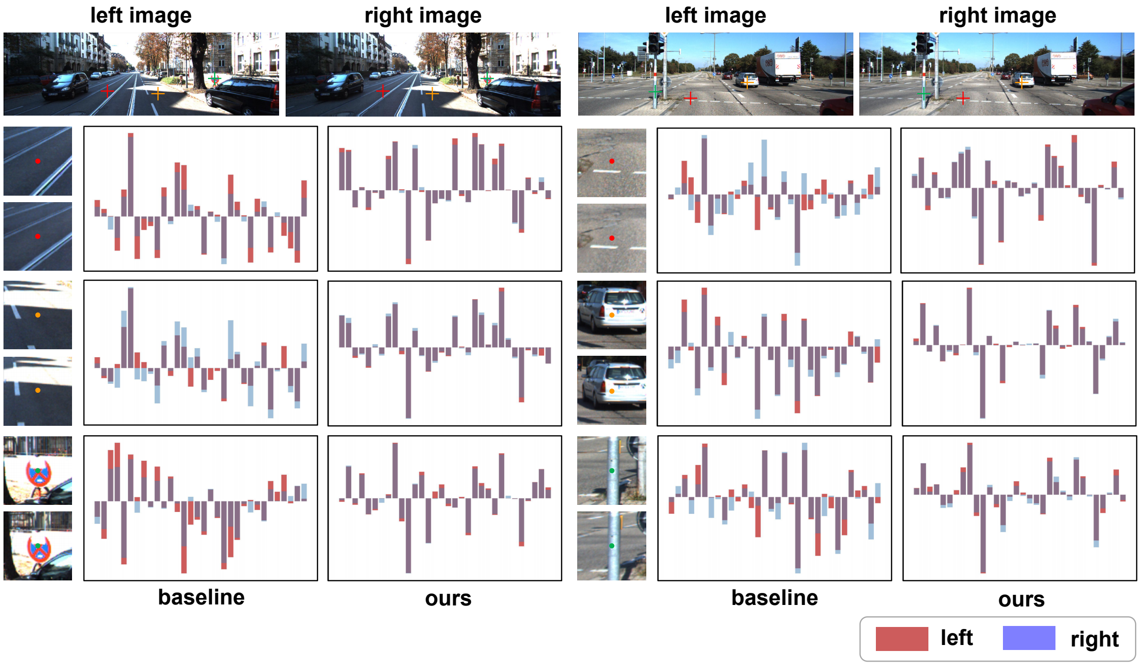

Figure 9. Channel-wise values of matching feature representations from the KITTI 2015 dataset. Matching pixels are zoomed in to make them more visible.

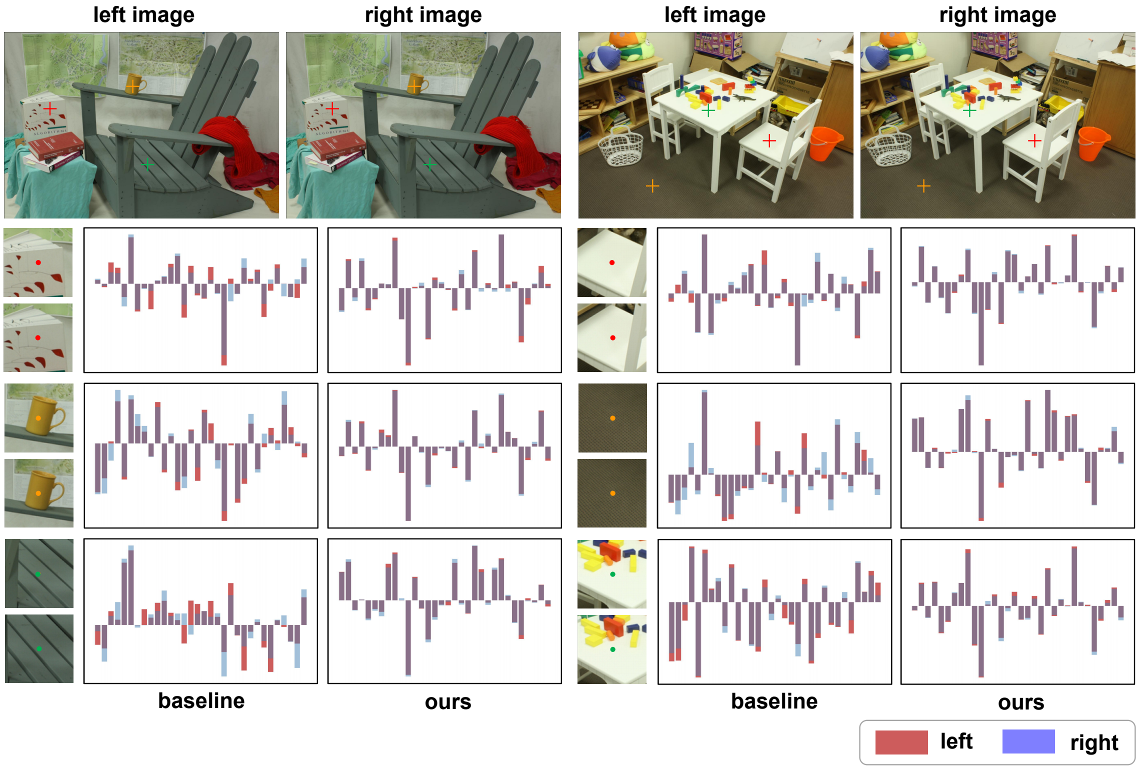

Figure 10. Channel-wise values of matching feature representations from the Middlebury dataset. Matching pixels are zoomed in to make them more visible.

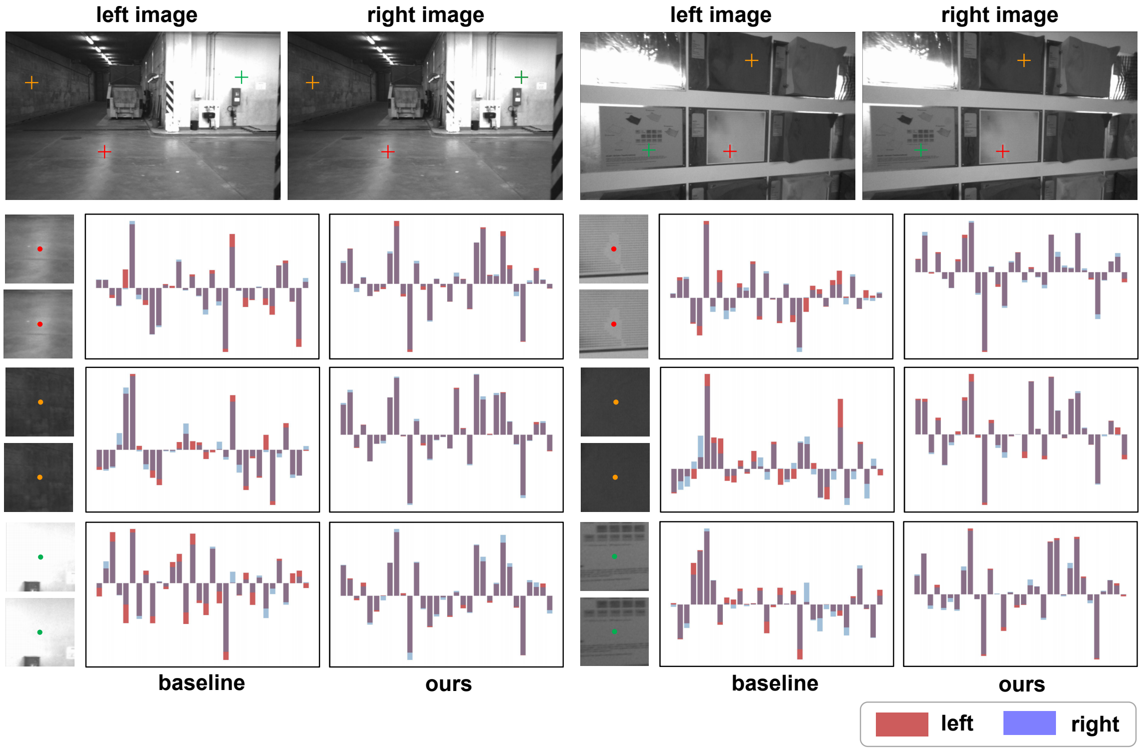

Figure 11. Channel-wise values of matching feature representations from the ETH3D dataset. Matching pixels are zoomed in to make them more visible.

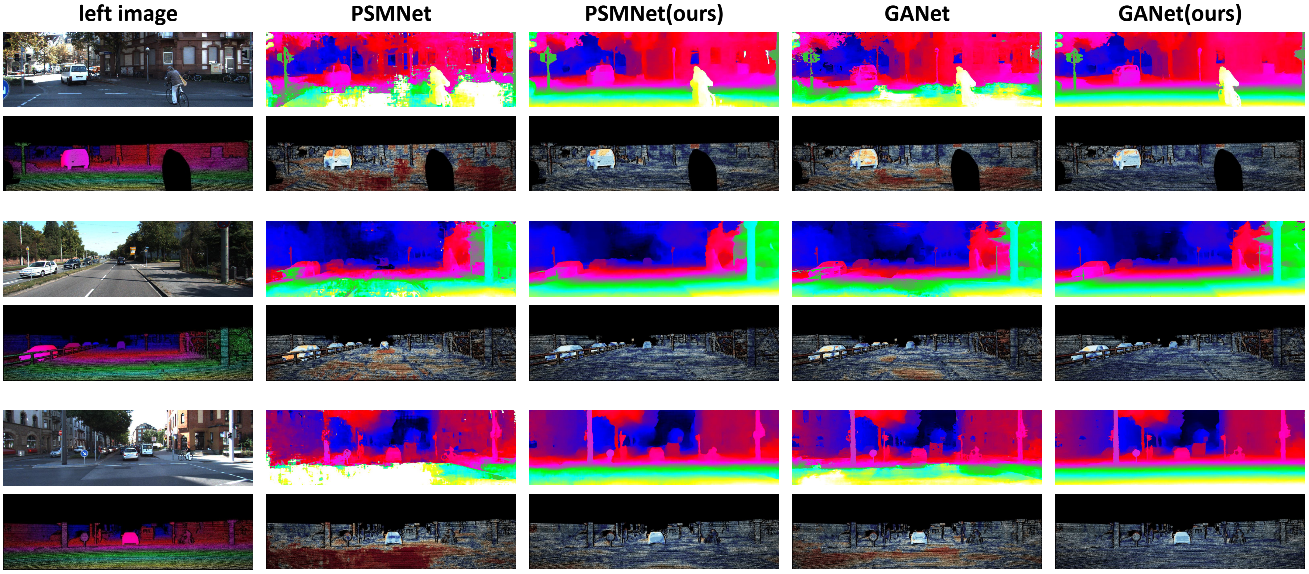

Figure 12. Qualitative results on the KITTI2015 training set. The left panel shows the left input image of the stereo pair and the ground truth disparity. And for each example, the first row shows the colorized disparity estimation and the second row shows the error map.

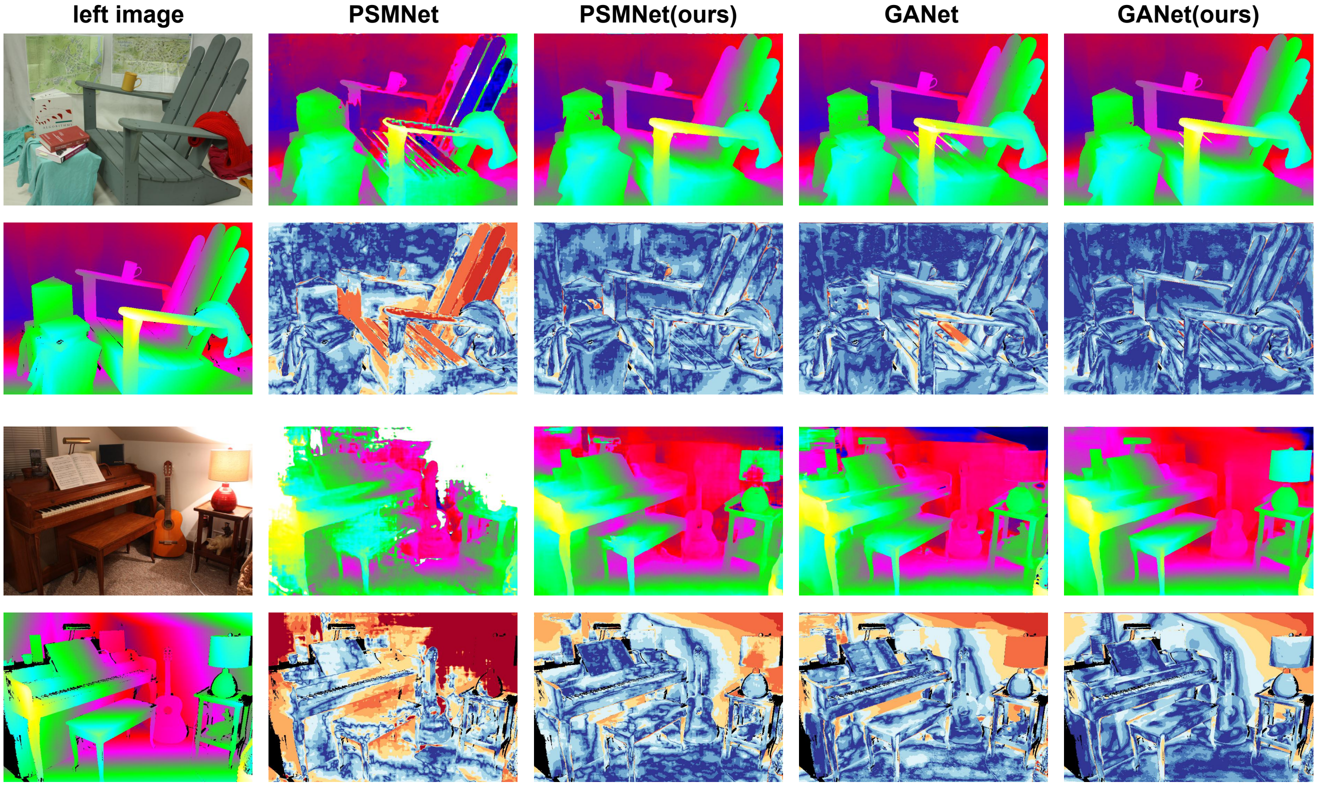

Figure 13. Qualitative results on the Middlebury training set. The left panel shows the left input image of the stereo pair and the ground truth disparity. And for each example, the first row shows the colorized disparity estimation and the second row shows the error map.

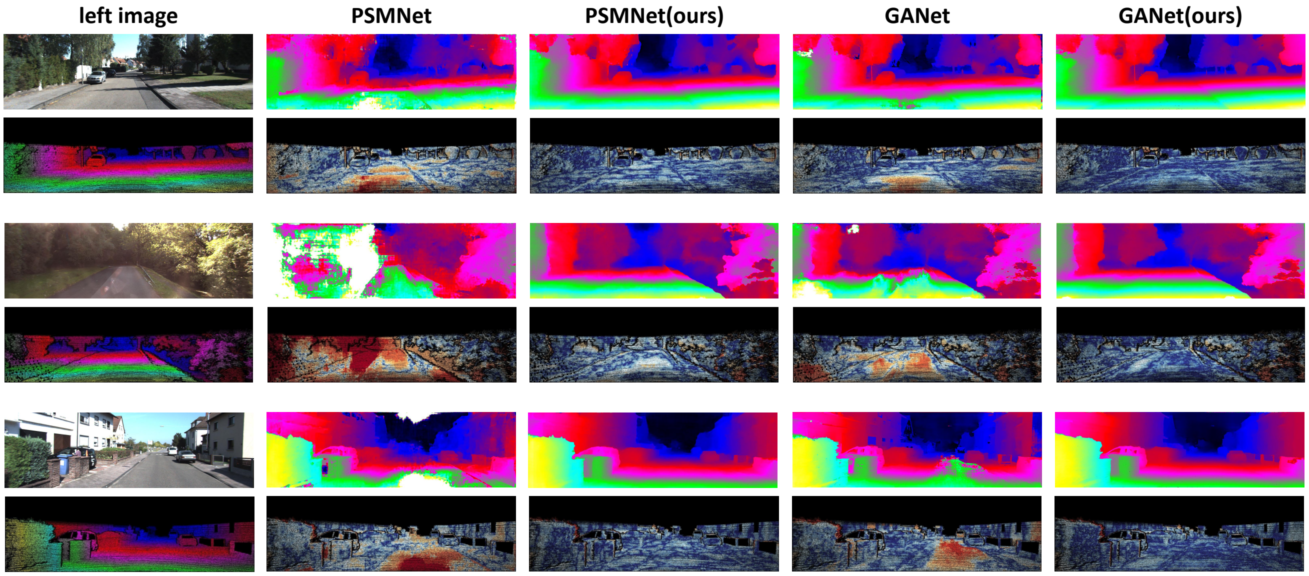

Figure 14. Qualitative results on the KITTI2012 training set. The left panel shows the left input image of the stereo pair and the ground truth disparity. And for each example, the first row shows the colorized disparity estimation and the second row shows the error map.

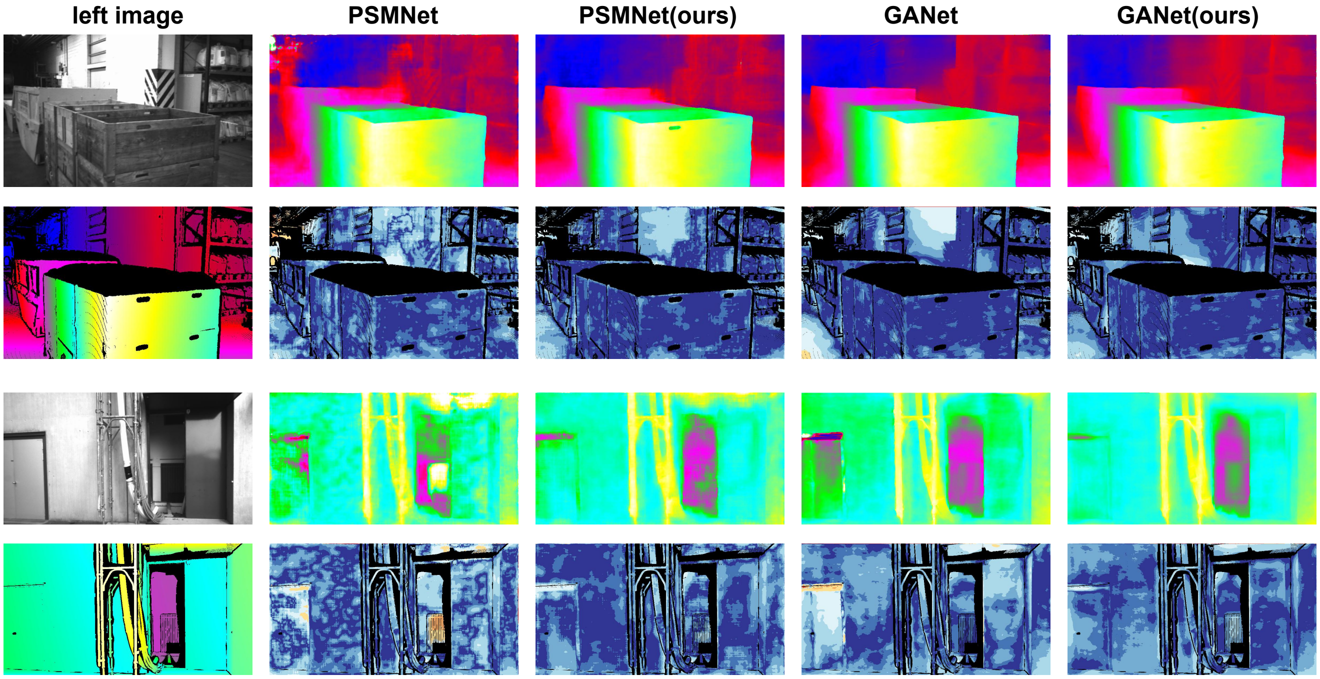

Figure 15. Qualitative results on the ETH3D training set. The left panel shows the left input image of the stereo pair and the ground truth disparity. And for each example, the first row shows the colorized disparity estimation and the second row shows the error map.