A Decomposition Model for Stereo Matching Supplementary Material

Chengtang Yao, Yunde Jia, Huijun Di*, Pengxiang Li, Yuwei Wu
Beijing Laboratory of Intelligent Information Technology
School of Computer Science, Beijing Institute of Technology, Beijing, China
{yao.c.t,jiayunde,ajon,lipengxiang,wuyuwei}@bit.edu.cn

Contents

1. Method Details

1.1. Proof of Theorem 1

Theorem 1. *Supposing $s \in \{2, 3, \cdots\}$ is the size of upsampling ratio between adjacent levels, $1 < C \leq 8/7$ is a constant value and $\mathcal{O}(\cdot)$ represents the tight upper bound, then the complexity* O *of exhaustive search process is*

$$\mathrm{O} = \mathrm{W}_0\mathrm{H}_0\mathrm{D}_0\ \mathcal{O}(s^{3L}C). \tag{1}$$

Proof. As $\mathrm{W}_l = \mathrm{W}_0 s^l$ and so does H_l and D_l, we get $\mathrm{O}_l = \mathrm{W}_l\mathrm{H}_l\mathrm{D}_l = \mathrm{O}_0 s^{3l}$. We then rewrite $\mathrm{O} = \sum_{l=0}^{l=L} \mathrm{O}_l$ as $\mathrm{O} = \sum_{l=0}^{L} \mathrm{O}_0 s^{3l} = \mathrm{O}_0 \frac{s^{3(L+1)}-1}{s^3-1} < \mathrm{O}_0 s^{3L} \frac{s^3}{s^3-1}$. We use C to represent $\frac{s^3}{s^3-1}$ where $1 < \frac{s^3}{s^3-1} \leq \frac{8}{7}$ because s is at least 2.

1.2. Dense Matching

As shown in Figure 1, we build the full cost volume via cross-correlation after warping the right feature maps. We then incorporate eright 3D convolutions to rectify the cost volume. A softmax operation is also used to turn cost volume into a probability volume. The dense disparity map is finally obtained via the regression of probability volume and the sampled disparities.

*Corresponding author

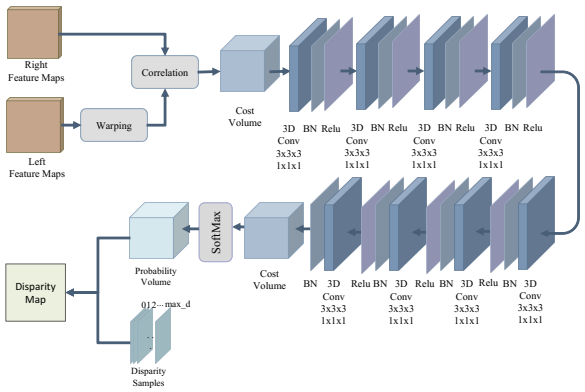

Figure 1: Architecture Of dense matching module. $3\times3\times3$ is the kernel size of 3D convolution, and $1 \times 1 \times 1$ is the stride size.

1.3. Backpropagation of Sparse Matching

In the main draft, sparse matching is formulated as follows:

$$C_l(h,w,d) =< \acute{F}_l(h,w), \grave{F}_l(h,w-d) >, \tag{2}$$

$$\begin{aligned} P_l(h,w,d) &= \frac{\mathrm{e}^{C_l(h,w,d)-C_l^{max}(h,w)}}{\sum_{d=0}\mathrm{e}^{C_l(h,w,d)-C_l^{\max}(h,w)}}, \\ C_l^{\max}(h,w) &= \max_d C_l(h,w,d), \end{aligned} \tag{3}$$

$$\hat{D}_l(h,w) = \sum_{d=0} P_l(h,w,d)\cdot d. \tag{4}$$

For the convenience of the derivation of the backpropagation, we rewrite the above equations as

$$\hat{D}_l(h,w) = \frac{\sum_{d=0}\mathrm{e}^{<\acute{F}_l(h,w),\grave{F}_l(h,w-d)> - C_l^{\max}(h,w)}\cdot d}{\sum_{d=0}\mathrm{e}^{<\acute{F}_l(h,w),\grave{F}_l(h,w-d)> - C_i^{\max}(h,w)}}. \tag{5}$$

Models	Res	time (s)	time/MP (s)	time/GD (s)	NonOcc						All					
					bad 2.0	bad 4.0	avgerr	rms	A90	A99	bad 2.0	bad 4.0	avgerr	rms	A90	A99
PSMNet [1]	Q	0.64	2.62	32.2	42.1	23.5	6.68	19.4	17.0	84.5	47.2	29.2	8.78	23.3	22.8	106
DeepPruner [2]	**Q**	**0.13**	0.41	4.38	30.1	15.9	4.80	14.7	10.4	67.7	36.4	21.9	6.56	18.0	17.9	83.7
GANet [9]	H	8.53	6.33	16.4	**18.9**	11.2	12.2	35.4	40.0	84.5	**24.9**	**16.3**	15.8	42.0	50.9	194
AANet [8]	H	4.56	4.17	11.0	25.2	19.6	8.88	26.2	24.2	131	31.8	25.8	12.8	32.8	41.4	142
ours	**F**	**0.51**	**0.10**	**0.23**	**20.2**	**11.2**	**3.72**	**12.5**	**10.1**	**46.8**	**27.0**	**17.0**	**5.37**	**15.9**	**15.0**	**72.2**

Table 1: The comparison of algorithms on Middlebury-v3 dataset (Q: quadratic resolution, H: half resolution, F: full resolution).

We then compute the backpropagation over $\acute{F}_l(h,w)$ as

$$\frac{\partial \hat{D}_l(h,w)}{\partial \hat{F}_l(h,w,c)} = \sum_{d=0}(\hat{F}_l(h,w-d,c)(d-\hat{D}_l(h,w) \\ e^{<\acute{F}_l(h,w),\grave{F}_l(h,w-d)>-C_i^{\max}(h,w)}) \\ / \sum_{d=0} e^{<\acute{F}_l(h,w),\grave{F}_l(h,w-d)>-C_i^{\max}(h,w)}, \tag{6}$$

$$\frac{\partial \mathcal{L}}{\partial \dot{F}_l(h,w,c)} = \frac{\partial \mathcal{L}}{\partial \hat{D}_l(h,w)} \frac{\partial \hat{D}_l(h,w)}{\partial \dot{F}_l(h,w,c)}. \tag{7}$$

As for $\grave{F}_l(h,w)$, we compute its backpropagation as

$$\frac{\partial \hat{D}_l(h,w+d')}{\partial \grave{F}_l(h,w,c)} = (\acute{F}_l(h,w+d',c)(d'-\hat{D}_l(h,w+d') \\ e^{<\acute{F}_l(h,w+d'),\grave{F}_l(h,w)>-C_i^{\max}(h,w+d')}) \\ / \sum_{d=0} e^{<\acute{F}_l(h,w+d'),\grave{F}_l(h,w+d'-d)>-C_i^{\max}(h,w+d')}). \tag{8}$$

$$\frac{\partial \mathcal{L}}{\partial \grave{F}_l(h,w,c)} = \sum_{d'=0} \frac{\partial \mathcal{L}}{\partial \hat{D}_l(h,w+d')} \frac{\partial \hat{D}_l(h,w+d')}{\partial \grave{F}_l(h,w,c)} \tag{9}$$

1.4. Loss

In addition to the unsupervised loss $\mathcal{L}_l^{\mathrm{DLD}}$ for detail loss detection, we also design a supervised loss for disparity estimation. As there is only ground truth GT of disparity map at the highest level, we downsample the ground truth to each level GT_l. At the lowest level, we use smooth L1 between the predicted dense disparity map and the downsampled ground truth:

$$\mathcal{L}_0 = smooth\mathrm{H}_{L_1}(\mathrm{D}_0 - GT_0), \\ smooth\mathrm{H}_{L_1}(\epsilon) = \begin{cases} 0.5\epsilon^2, & if \mid \epsilon \mid < 1 \\ \mid \epsilon \mid -0.5, & otherwise \end{cases}. \tag{10}$$

At higher levels, there are four intermediate results at each level, including the upsampled dense disparity map from previous level D_l', the sparse disparity map $\hat{D}_l$, the fused disparity map $\bar{D}_l$ and the refined disparity map D_l. To this end, we use a weighted combination of smooth L1 loss over them:

$$\begin{aligned} \mathcal{L}_l = \ & \gamma_1 * smooth\mathrm{H}_{L_1}(\mathrm{D}_l - GT_l) \\ & + \gamma_2 * smooth\mathrm{H}_{L_1}(\bar{D}_l - GT_l) \\ & + \gamma_3 * smooth\mathrm{H}_{L_1}(\hat{D}_l - GT_l \ M_{F\acute{A}_l}) \\ & + \gamma_4 * smooth\mathrm{H}_{L_1}(D_l' - GT_l). \end{aligned} \tag{11}$$

Finally, we train our model using end-to-end learning with following loss function:

$$\mathcal{L} = \mathcal{L}_0 \cdot \mathrm{W}_0 + \sum_{l=1}^{l=L}(\mathcal{L}_l \cdot \mathrm{W}_l + \mathcal{L}_l^{\mathrm{DLD}} w_l'), \tag{12}$$

where W_l and w_l' are the loss weight.

2. More Details on Experiment

We set $\gamma_1 = 0.5$, $\gamma_2 = 0.2$, $\gamma_3 = 0.2$, $\gamma_4 = 0.1$, and $w_0 = 0.037$, $w_1 = 0.11$, $w_2 = 0.33$, $w_3 = 1$, $w_1' = 0.01$.

2.1. Middlebury-v3

We present the comparison of results on the Middleburyv3 dataset [7]. We first give a brief description of the metric. time/MP: time normalized by the number of pixels (sec/megapixels). time/GD: time normalized by the number of disparity hypotheses (sec/(gigapixels*ndisp)). bad xx: percentage of bad pixels whose error is greater than xx. avgerr: average absolute error in pixels. rms: root mean-square disparity error in pixels. Axx: xx-percent error quantile in pixels. As shown in Table 1, our model achieves the best speed on time/MP and time/GD. our model also obtains almost the best results on most metrics about accuracy.

2.2. KITTI 2015

Despite the comparison with state-of-the-art methods in the main draft, we also give a visualization of our results on the KITTI 2015 dataset [4, 5, 6]. As shown in Figure 2, our model achieves competitive estimations in various scenarios.

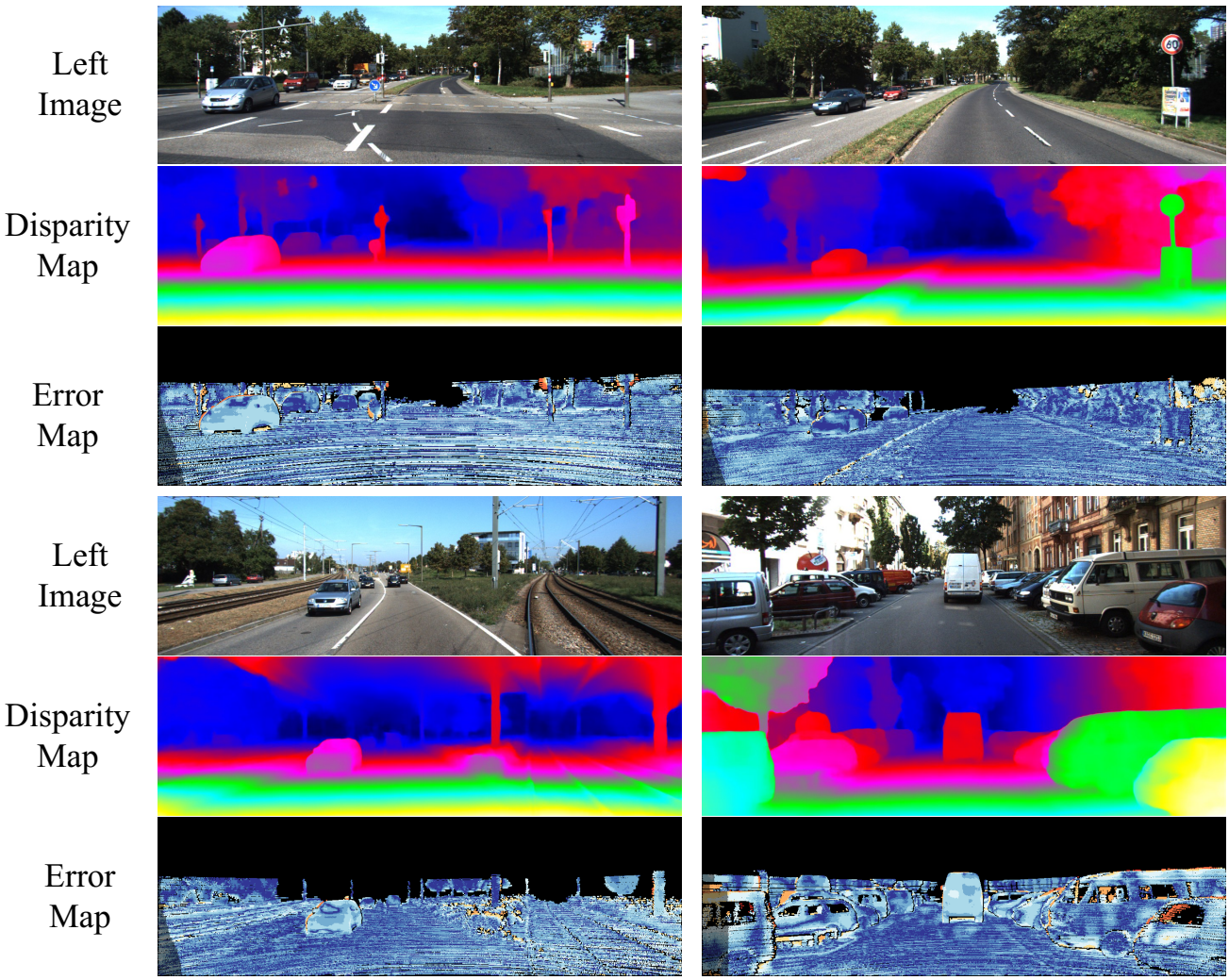

Figure 2: Visualization of results on KITTI2015 dataset.

2.3. SceneFlow

We give a visualization of our results on the Scene Flow dataset [3]. As shown in Figure 3, our model achieves great results in different areas, like thin or small objects and large texture-less areas.

References

[1] Chang. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5410–5418, 2018. 2

[2] Duggal. Deeppruner: Learning efficient stereo matching via differentiable patchmatch. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4384–4393, 2019. 2

[3] Mayer. A large dataset to train convolutional networks for disparity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4040–4048, 2016. 3

[4] Menze. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3061–3070, 2015. 2

[5] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. In *Isprs Workshop on Image Sequence Analysis (ISA)*, 2015. 2

[6] Moritz Menze, Christian Heipke, and Andreas Geiger. Object scene flow. *Isprs Journal of Photogrammetry and Remote Sensing (JPRS)*, 2018. 2

[7] Scharstein. High-resolution stereo datasets with subpixel-accurate ground truth. In *Proceedings of the German Conference on Pattern Recognition (GCPR)*, pages 31–42, 2014. 2

[8] Xu. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE Conference on Com-*

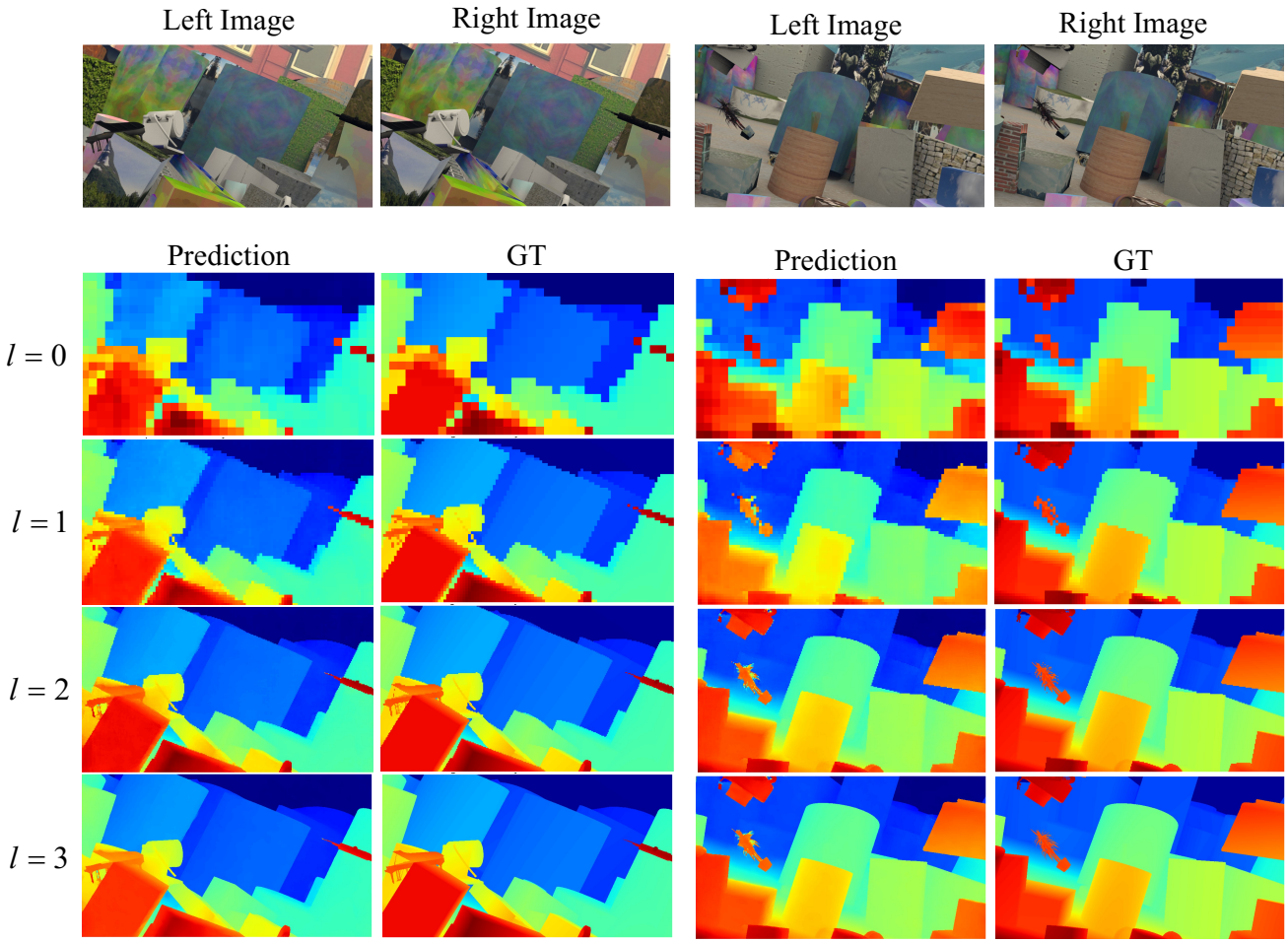

Figure 3: Visualization of results on Scene Flow dataset.

puter Vision and Pattern Recognition (CVPR), pages 1959–1968, 2020. 2

[9] Zhang. Ga-net: Guided aggregation net for end-to-end stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 185–194, 2019. 2