Supplementary Material for Deep Color Guided Coarse-to-Fine Convolutional Network Cascade for Depth Image Super-Resolution

Yang Wen, Bin Sheng, Ping Li, Weiyao Lin, and David Dagan Feng, *Fellow, IEEE*

I. Percent of Error Comparison on Middlebury Datasets

Our proposed approach is illustrated visually in Fig. 1–Fig. 3 and compared with a great many popular super resolution methods. Among the figures, Fig. 1, Fig. 2, and Fig. 3 show the super resolution pixel errors of the Middlebury data. From the figures, we can see that our proposed methods also generate more visual appealing results than the previously reported approaches. Boundaries in our results are generally sharper and smoother along the edge direction.

II. Comparison with Network Based Methods

Fig. 4 presents a comparison with different guided SRCNN [8] based methods. the classical SRCNN [8] models released by the authors were trained on color patches, and we specifically retrained the models using depth patches. However, our experimental results in Fig. 4 show that depth patches based method is also not very suitable for DSR. It will blur sharp depth edges as discussed before and shown in error maps in Fig. 5(a) and Fig. 4. The reason is that depth images are normally quite different from natural color images. They are textureless while having very sharp but sparse edges, the traditional CNN tends to ignore sharp edges when trained with depth patches. The performance will be even lower than the original color-trained SRCNN. It also proves that our color guidance is more effective.

III. Visual Comparison with Methods that Require an External Depth Database

Fig. 5 presents visual comparison with methods that require an external depth database [7], [8] on *Book*, *Laundry* and *Reindeer* databases when the upsampling factor is 4. SRCNN [8] is mainly designed for color image super-resolution. Unlike disparity/depth images, sharp color edges are abnormal. As a result, SRCNN [8] tends to blur the disparity map a bit as can be seen from the error maps in Fig. 5(a), although these errors are almost invisible from the disparity maps. PB [7] successfully maintains the sharp edges as can be seen from Fig. 5(b). However, its performance is lower than SRCNN [8] around thin-structured objects. This is mainly because PB [7] mainly depends on the similarity between the training data and the input low-resolution disparity map (where the details of the thin-structured objects are gone) to find high-resolution patches from training data as output. The proposed method is indeed an extension of SRCNN [8] with the use of an additional registered color image to better preserve depth edges and thus outperforms [7], [8] on average.

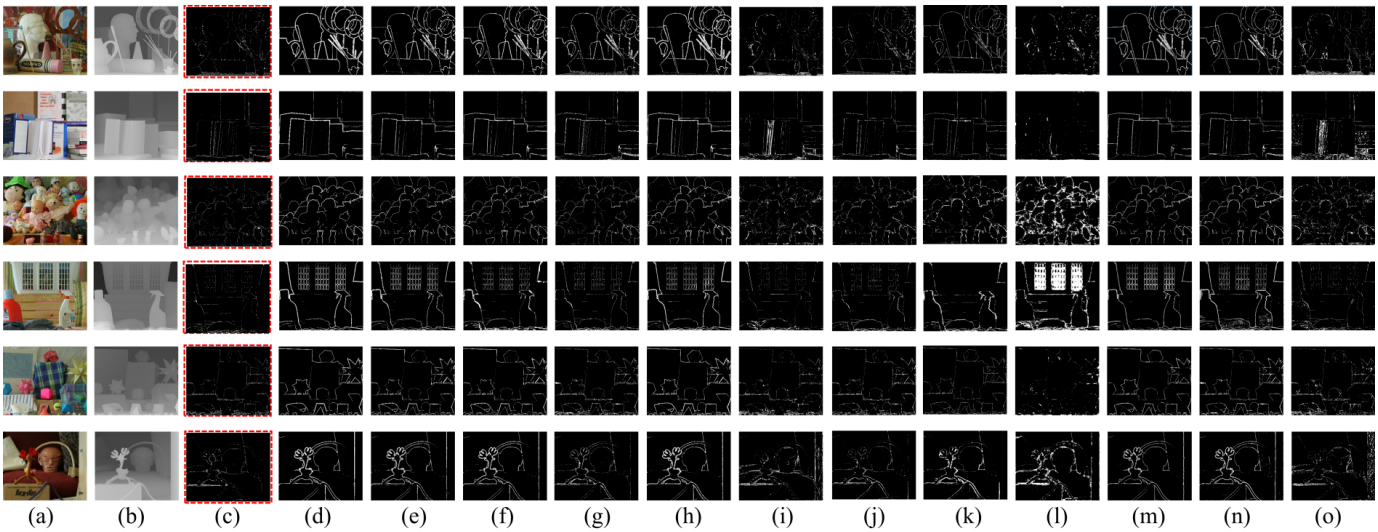

Fig. 1: Visual comparison of pixel errors on Middlebury database (scaling factor = 4). (a) Color image. (b) Ground truth. (c) Our Proposed. (d) AP [1]. (e) Bicubic. (f) CLMF0 [2]. (g) CLMF1 [2]. (h) Edge [3]. (i) Guided [4]. (j) JBFcv [5]. (k) JGF [6]. (l) PB [7]. (m) SRCNN [8]. (n) TGV [9]. (o) Tree [10].

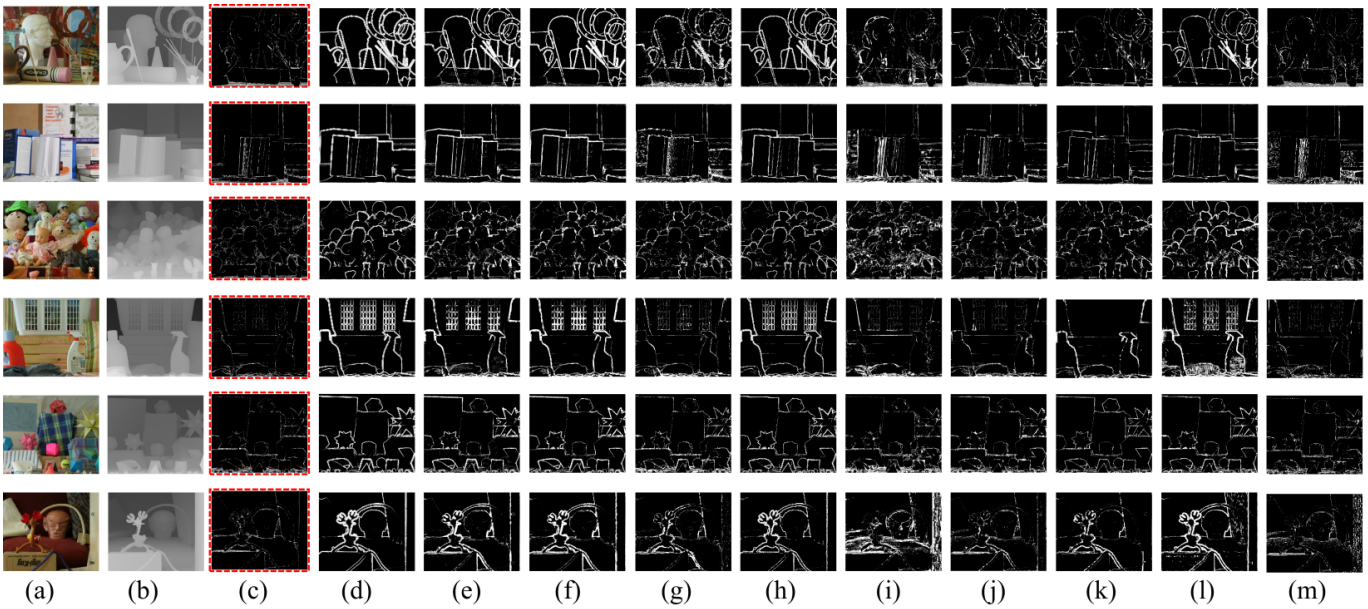

Fig. 2: Visual comparison of pixel errors on Middlebury database (scaling factor = 8). ((a) Color image. (b) Ground truth. (c) Our Proposed. (d) AP [1]. (e) Bicubic. (f) CLMF0 [2]. (g) CLMF1 [2]. (h) Edge [3]. (i) Guided [4]. (j) JBFcv [5]. (k) JGF [6]. (l) TGV [9]. (m) Tree [10].

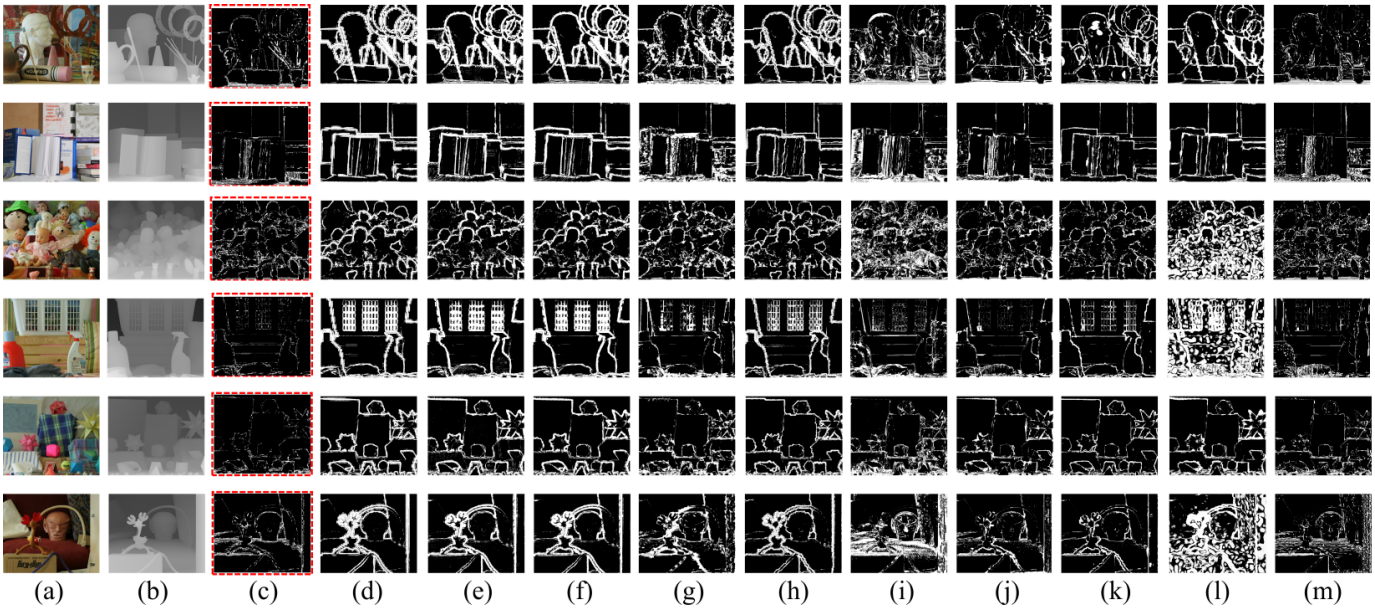

Fig. 3: Visual comparison of pixel errors on Middlebury database (scaling factor = 8). (a) Color image. (b) Ground truth. (c) Our Proposed. (d) AP [1]. (e) Bicubic. (f) CLMF0 [2]. (g) CLMF1 [2]. (h) Edge [3]. (i) Guided [4]. (j) JBFcv [5]. (k) JGF [6]. (l) TGV [9]. (m) Tree [10].

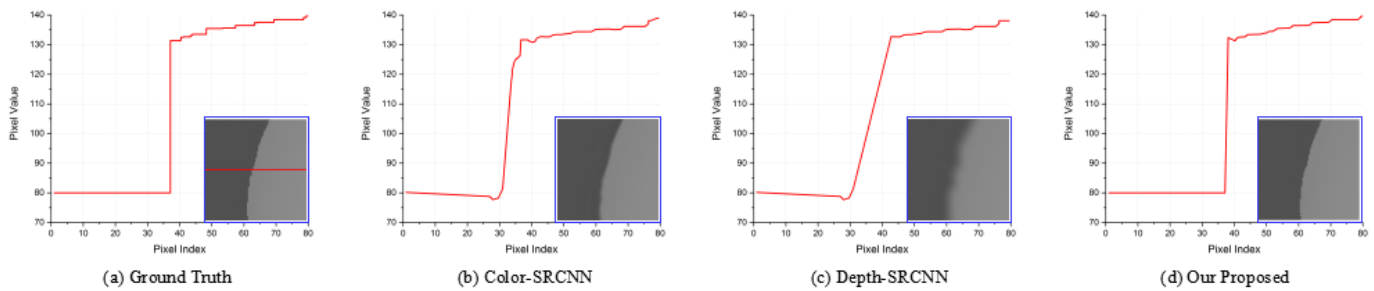

Fig. 4: Comparison with both color and depth trained SRCNN [8]. First row: (a) is the ground-truth disparity map and (b)-(d) are the disparity maps upsampled from the original color-trained SRCNN, depth-trained SRCNN and the proposed networks. Bottom row: disparity values along the red line in the first row of (a). Note that the proposed networks can better preserve the sharp depth edges in (a).

TABLE I: Synthesis performance comparison on the all databases [9] using MAD metric with 4× upsampling factor.

		KSVD [11]	CDLLC [12]	Xie [13]	PB [7]	SRCNN [8]	ATGV-Net [14]	Song [15]	Wang [16]	MSG-Net [17]	Our_CS	Our_FS
Average		3.30	3.05	2.30	4.08	4.62	3.48	2.23	4.39	2.18	2.17	**2.10**
Variance		2.08	1.89	0.64	5.53	4.33	2.14	0.61	4.00	0.57	**0.55**	0.57
95% confidence interval	Lower Bound	2.47	2.25	1.84	2.62	3.42	2.63	1.78	3.23	1.74	1.74	**1.67**
	Upper Bound	4.14	3.84	2.76	6.33	5.83	4.32	2.68	5.54	2.62	2.59	**2.54**

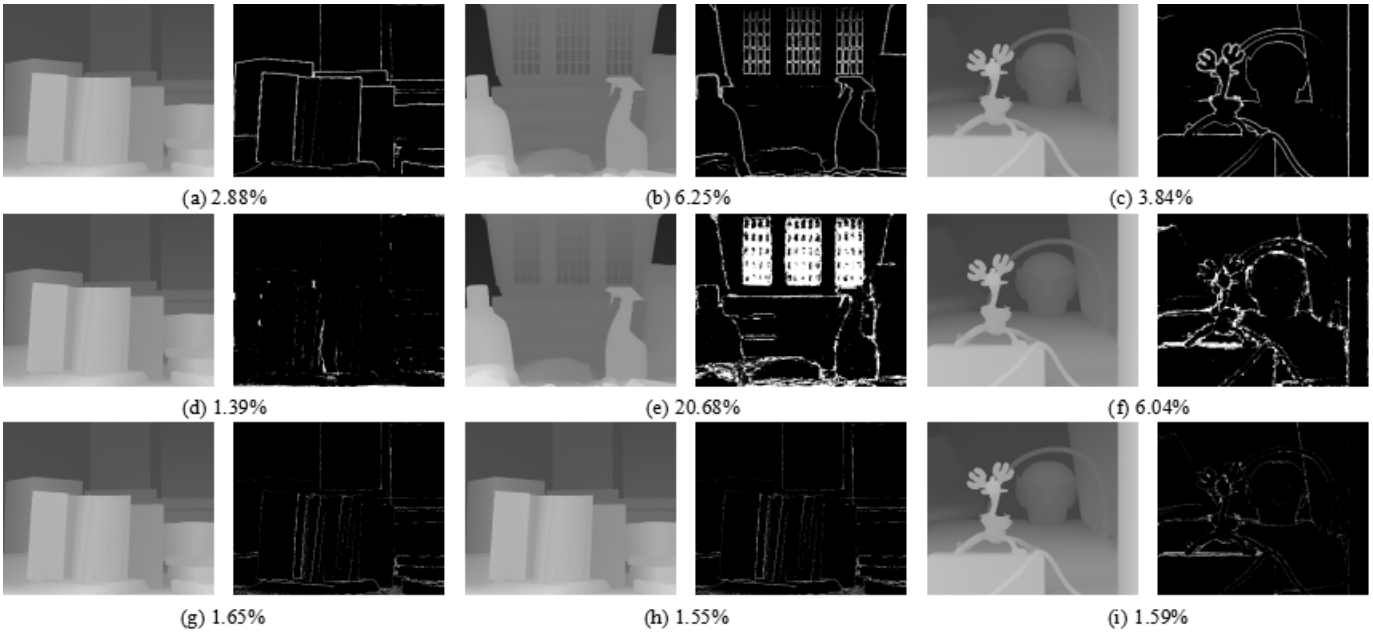

Fig. 5: Visual comparison with methods that require an external depth database [7], [8] on three Middlebury databases (*Book*, *Laundry* and *Reindeer*) when the spatial resolution is enhanced $16\times$ (4×4). (a)-(c) are the upsampled disparity and error maps of [8], [7] and the proposed method respectively. The numbers under the disparity and error maps are the corresponding percentage of error pixels and the top performers are marked in bold. SRCNN [8] is mainly designed for color image super-resolution. Unlike disparity/depth images, sharp color edges are abnormal. As a result, SRCNN [8] tends to blur the disparity map a bit as can be seen from the error maps in (a), although these errors are almost invisible from the disparity maps. PB [7] successfully maintains the sharp edges as can be seen from (b). However, its performance is lower than SRCNN [8] around thin-structured objects. This is mainly because PB [7] mainly depends on the similarity between the training data and the input low-resolution disparity map (where the details of the thin-structured objects are gone) to find high-resolution patches from training data as output. The proposed method is indeed an extension of SRCNN [8] with the use of an additional registered color image to better preserve depth edges and thus outperforms [7], [8] on average.

IV. Visual Comparison when the Spatial Resolution is Enhanced

Fig. 6 presents a comprehensive visual comparison with three other algorithms on the *Laundry* and *Dolls* databases. As can be seen from the close-ups of the *Laundry* database in Fig. 6(b)–Fig. 6(c), AP [1] is more likely to blur the depth edges while JGF [6] may produce false depth edges (due to color textures). The depth edges obtained from Edge [3] is better. However, similar to AP [1] and JGF [6], it may remove the depth details between pixels with similar colors as can be seen from the close-up of the *Dolls* database (in last row of Fig. 6(d)). As demonstrated in Fig. 6(e), the proposed method is more robust to inconsistent color and depth edges with the learned data-driven filter kernel.

To verify the superiority of our algorithm, we provide a 95 percent confidence interval performance comparison for the evaluated measures. From Table I, we can see that our algorithm has the lowest average and variance values on the all databases using MAD metric with 4 upsampling factor. That is to say, our method performs better on the entire database than any other methods.

Fig. 7 reveals the potential texture-copying artifact in the previous edge-preserving filtering based DSR methods using the *Dolls* dataset. Note that [5] and [10] have obvious texture-copying artifacts when the upsampling factor is high. For instance, around the eyes and ears of the donkey. The proposed method can better suppress this artifact as can be seen in Fig. 7(g). And it can be seen our approach can solve the texture-copying problem very well.

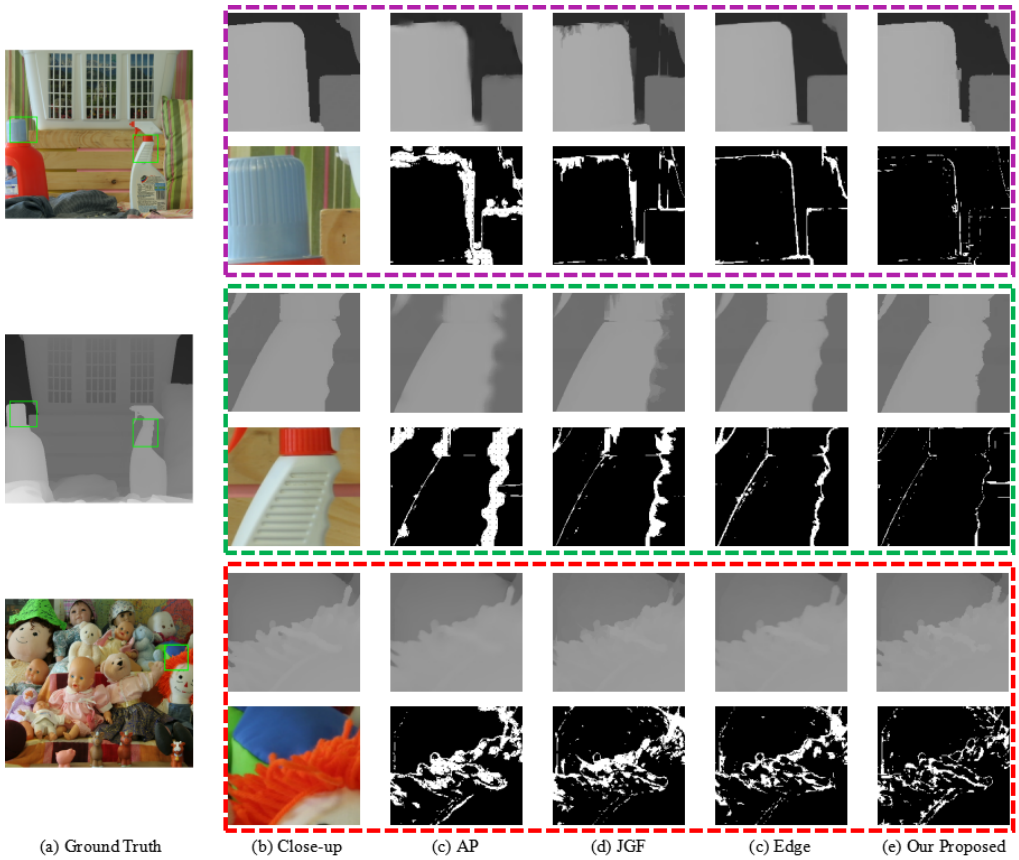

Fig. 6: Visual comparison when the spatial resolution is enhanced $64\times$ (8×8). As seen from the close-ups of the *Laundry* database in (b)-(c), AP [1] is more likely to blur the depth edges while JGF [6] may produce false depth edges (due to color textures). The depth edges obtained from Edge [3] is better. However, similar to AP [1] and JGF [6], it may remove the depth details between pixels with similar colors as can be seen from the close-up of the *Dolls* database (in last row of (d)). As can be seen from (e), the proposed method is more robust to inconsistent color and depth edges with the learned data-driven filter kernel.

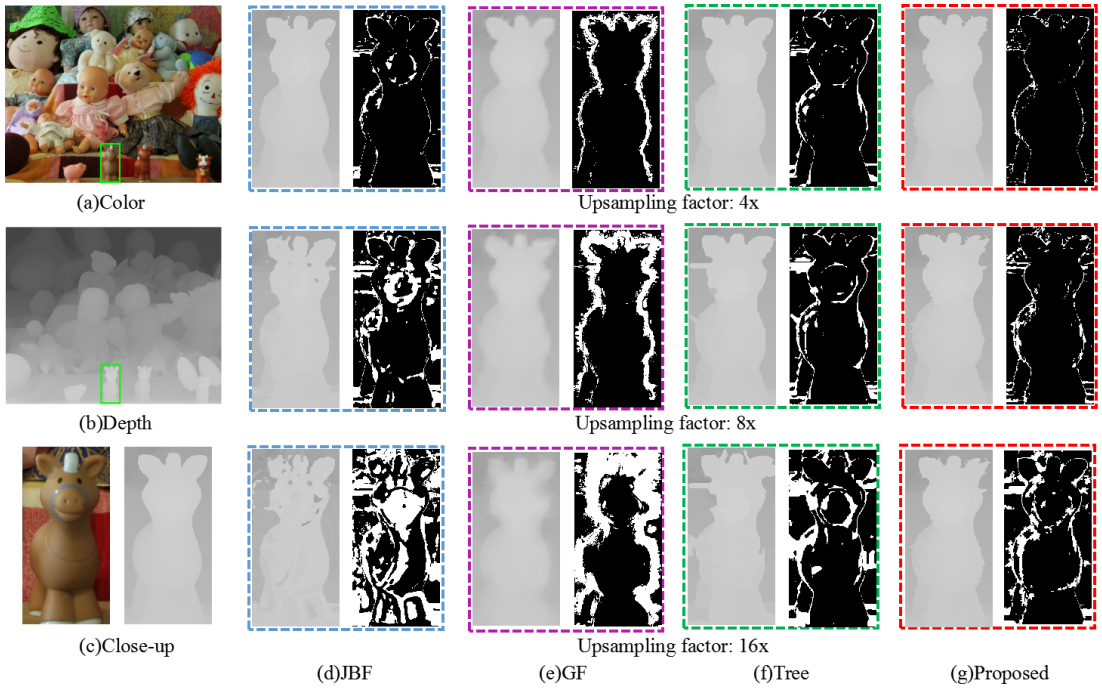

Fig. 7: Texture-copying artifact. (a) presents the high-resolution color image and the ground-truth disparity map. (b) and (c) are close-ups from (a). (d)-(g) are the close-ups of the disparity maps upsampled using different methods and the corresponding disparity error maps (obtained with error threshold 1). Note that [5] and [10] have obvious texture-copying artifacts when the upsampling factor is high. For instance, around the eyes and ears of the donkey. The proposed method can better suppress this artifact as can be seen in (g).

REFERENCES

[1] J. Yang, X. Ye, K. Li, C. Hou, and Y. Wang, "Color-guided depth recovery from RGB-D data using an adaptive autoregressive model," *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3443–3458, 2014.

[2] J. Lu, K. Shi, D. Min, L. Lin, and M. N. Do, "Cross-based local multipoint filtering," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 430–437.

[3] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon, "High quality depth map upsampling for 3D-TOF cameras," in *International Conference on Computer Vision*, 2011, pp. 1623–1630.

[4] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013.

[5] Q. Yang, R. Yang, J. Davis, and D. Nistér, "Spatial-depth super resolution for range images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[6] M.-Y. Liu, O. Tuzel, and Y. Taguchi, "Joint geodesic upsampling of depth images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 169–176.

[7] O. Mac Aodha, N. D. F. Campbell, A. Nair, and G. J. Brostow, "Patch based synthesis for single depth image super-resolution," in *European Conference on Computer Vision*, 2012, pp. 71–84.

[8] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, 2014, pp. 184–199.

[9] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rüther, and H. Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *IEEE International Conference on Computer Vision*, 2013, pp. 993–1000.

[10] Q. Yang, "Stereo matching using tree filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 4, pp. 834–846, 2015.

[11] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International Conference on Curves and Surfaces*, 2012, pp. 711–730.

[12] J. Xie, C.-C. Chou, R. Feris, and M.-T. Sun, "Single depth image super resolution and denoising via coupled dictionary learning with local constraints and shock filtering," in *IEEE International Conference on Multimedia and Expo*, 2014, pp. 1–6.

[13] J. Xie, R. S. Feris, and M.-T. Sun, "Edge-guided single depth image super resolution," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 428–438, 2016.

[14] G. Riegler, M. Rüther, and H. Bischof, "ATGV-Net: Accurate depth super-resolution," in *European Conference on Computer Vision*, 2016, pp. 268–284.

[15] X. Song, Y. Dai, and X. Qin, "Deep depth super-resolution: Learning depth super-resolution using deep convolutional neural network," in *Asian Conference on Computer Vision*, 2016, pp. 360–376.

[16] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *IEEE International Conference on Computer Vision*, 2015, pp. 370–378.

[17] T.-W. Hui, C. C. Loy, and X. Tang, "Depth map super-resolution by deep multi-scale guidance," in *European Conference on Computer Vision*, 2016, pp. 353–369.