

Simple Yet Effective Way for Improving the Performance of Depth Map Super-Resolution

Yoon-Jae Yeo , Min-Cheol Sagong, Yong-Goo Shin , Seung-Won Jung , Senior Member, IEEE, and Sung-Jea Ko , Fellow, IEEE

Abstract—In depth map super-resolution (SR), a high-resolution color image plays an important role as guidance for preventing blurry depth boundaries. However, excessive/deficient use of the color image features often causes performance degradation such as texture-copying/edge-smoothing in flat/boundary areas. To alleviate these problems, this letter presents a simple yet effective method for enhancing the performance of the SR without requiring significant modifications to the original SR network. To this end, we present a self-selective concatenation (SSC), which is a substitute for the conventional feature concatenation. In the upsampling layers of the SR network, the SSC extracts spatial and channel attention from both color and depth features such that color features can be selectively used for depth SR. Specifically, the SSC learns to use sufficient color features for rendering sharp depth boundaries, whereas their effects are reduced in smooth regions to prevent texture-copying. The proposed SSC can be included in any existing SR networks that have the encoder-decoder structure. The experimental results show that the proposed method can further improve the performances of existing SR networks in terms of the root mean squared error and peak signal-to-noise ratio.

Index Terms—Depth map super-resolution (SR), convolutional neural network, deep learning.

I. INTRODUCTION

DEPTH information is essential for many computer vision applications, such as autonomous navigation, virtual reality, and 3D reconstruction. Currently, many low-cost but low-resolution (LR) depth sensors are widely used, and they are generally embodied with high-resolution (HR) color sensors [21]–[23]. Therefore, many studies have examined the super-resolution (SR) of depth images using HR color images as guidance [1]–[9], [16]–[19].

As in the case with other computer vision tasks, convolutional neural networks (CNNs) are currently dominating depth SR

[1]–[8]. Motivated by classical color-guided depth SR methods [16]–[19], CNN-based methods attempt to use color features such that the resultant HR depth image can inherit the HR details of the color image. The main difference in the existing CNN-based methods involves the use of color features for depth SR.

Hui *et al.* [1] achieved promising SR results by using a multi-scale guided network (MSG-Net) that used the HR intensity image as guidance to complement depth features. Considering that the edges play the most important role in depth SR, Wen *et al.* [4] developed a handcrafted preprocessing method that refined the initially interpolated depth map using the corresponding color pixels to preserve details. On the other hand, some approaches inferred the HR depth edge map using the HR color image and the LR depth map, and used the HR edge map as the supplementary information to refine the boundaries of the depth map [5], [6]. To further improve the performance, Guo *et al.* [7] employed rich hierarchical features from the HR intensity image through the residual U-Net [24] architecture. Moreover, with the HR intensity image, Zuo *et al.* [8] progressively upscaled the LR depth map with global and local residual learning. In addition, Zhao *et al.* [9] simultaneously increased the resolution of the color image and the depth map using the generative adversarial network [20]. Most of the aforementioned methods [1]–[3], [6]–[9] used the guide features simply by concatenating them with the depth features. Therefore, the depth SR network needs to learn how to use two types of features properly, which remains a challenge. Indeed, the sharpness of the depth boundaries can be lost without sufficient usage of the guide features. Otherwise, if the guide features are used excessively in flat areas, unnecessary high frequency color details can be transferred from the guide image to the depth map.

This letter presents a novel feature concatenation method that significantly enhances the performance of depth map SR. Specifically, we propose the self-selective concatenation (SSC) to effectively combine the guide features with the depth features. Unlike the ordinary concatenation that simply combines two types of features in a fixed manner, the SSC selectively adjusts the usage of the guide features depending on the spatial locations and channels. The proposed SSC module requires a marginal number of parameters, and it can be readily applied to existing SR networks by changing the concatenation method. Through extensive experiments on the MPI Sintel depth dataset [10] and the Middlebury dataset [11]–[13] with/without Gaussian noise, we show that the proposed method can further improve the performance of the existing SR networks [1], [7], [8] in terms of the root mean squared error (RMSE) and peak signal-to-noise ratio (PSNR).

Manuscript received September 29, 2020; revised November 6, 2020; accepted November 15, 2020. Date of publication November 19, 2020; date of current version December 9, 2020. This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT Future Planning, under Grant NRF-2020R1F1A1069009. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Dezhong Peng. (Corresponding author: Seung-Won Jung.)

Yoon-Jae Yeo, Min-Cheol Sagong, Seung-Won Jung, and Sung-Jea Ko are with the Department of Electrical Engineering, Korea University, Seongbuk-gu, Seoul 02841, South Korea (e-mail: yjyeo@dal.korea.ac.kr; mcsagong@dal.korea.ac.kr; swjung83@korea.ac.kr; sjko@korea.ac.kr).

Yong-Goo Shin is with the Division of Smart Interdisciplinary Engineering, Hannam University, Daedeok-gu, Daejeon 34430, South Korea (e-mail: ygshin@hnu.kr).

Digital Object Identifier 10.1109/LSP.2020.3039429

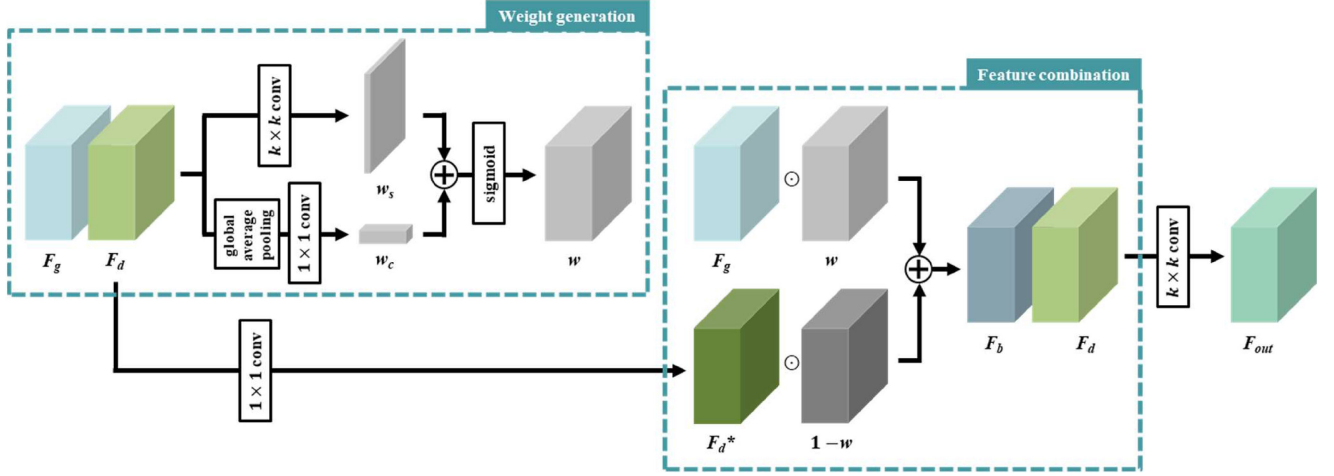


Fig. 1. Combination method for the guide and depth features in the proposed SSC. The rectified linear unit (ReLU) or parametric ReLU activation functions are applied after all convolutional layers.

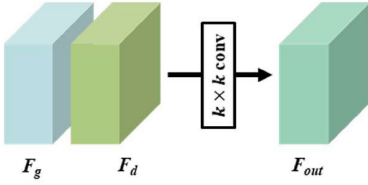


Fig. 2. Simple concatenation of the guide and depth features commonly used in conventional depth SR networks.

II. PROPOSED METHOD

Figs. 1 and 2 show the proposed and conventional feature concatenation methods, respectively. The conventional depth SR networks [1]–[3], [6]–[9] simply concatenate the depth features F_d and the guide features F_g , then extract the output features F_{out} using the convolutional layer as follows:

$$F_{out} = f^{k \times k} ([F_d, F_g]) \quad (1)$$

where $f^{k \times k}$ represents the convolutional layer with the filter size $k \times k$, and $[\cdot]$ is concatenation in the channel axis. It is challenging for the depth SR network to adjust importance of guide and depth features by this simple concatenation.

By contrast, the proposed SSC combines the two types of features at different rates depending on the spatial locations and channels. Fig. 1 illustrates how the SSC combines the color and depth features. First, the weight generation module produces the spatial and channel weight maps, which are given as

$$w_s = f^{k \times k} ([F_d, F_g]) \quad (2)$$

$$w_c = f^{1 \times 1} (AvgPool [F_d, F_g]) \quad (3)$$

$$w = \sigma(w_s + w_c) \quad (4)$$

where w_s and w_c indicate the attentive weights for the spatial locations and channels, respectively. To obtain w_s , a $k \times k$ convolution is applied to extract spatial contextual information and to reduce the number of channels to 1. Furthermore, to estimate the attention across the channels, w_c is extracted from global average pooling and 1×1 convolution. Inspired by the

well-known bottleneck attention module (BAM) [14], the final weight map w is obtained through the summation of w_s and w_c followed by the sigmoid function σ . Note that this summation includes tensor broadcasting to enable element-wise summation. The weight map w determines the proportion of the guide features to be used for depth SR.

Next, the feature combination module in Fig. 1 combines the guide and depth features as follows:

$$SSC(F_d, F_g) = [F_d, F_b] \quad (5)$$

where

$$F_b = w \odot F_g + (1 - w) \odot F_d^* \quad (6)$$

$$F_d^* = f^{1 \times 1} (F_d) \quad (7)$$

Here, $\odot$ represents element-wise multiplication and F_b is the blended features in which the two types of features are adaptively combined. Note that the blending ratio between the guide and depth features is determined using w . For example, if w has all ones (or all zeros), F_b is constructed from only the guide information, i.e., F_g (or depth information, i.e., F_d^*). Ultimately, instead of F_g , F_b is concatenated with F_d as guidance to support the upscaling layer of the SR network. In this manner, the SSC induces the network to appropriately regulate the influence of the guide information. To increase the diversity of the depth features, we blend F_d^* , which is acquired by a 1×1 convolutional layer, instead of F_d . Consequently, the network is driven to effectively learn to give appropriate contributions to the two types of features based on spatial and channel attention. The output features F_{out} are finally obtained as follows:

$$F_{out} = f^{k \times k} (SSC(F_d, F_g)) \quad (8)$$

The proposed method can be applied to the existing depth SR networks that have encoder-decoder structures without significantly changing their structures. Moreover, if the number of channels of F_g and F_d is chosen as 32, the SSC only consumes 2.8k network parameters. Therefore, the SSC does not impose a large computational burden on the original network.

TABLE I
EXPERIMENTAL RESULTS IN TERMS OF RMSE/PSNR OBTAINED BY APPLYING THE PROPOSED METHOD TO THE BASELINE MODELS [1], [7], [8]

Scaling factor	Baseline model	SSC	Art	Books	Dolls	Laundry	Moebius	Reindeer	Average ±standard deviation
2 ×	Zuo [8]		0.853/49.51	0.782/50.27	0.800/50.07	0.794/50.14	0.781/50.28	0.816/49.90	0.804±0.025/50.03±0.26
		✓	0.785/50.23	0.721/50.97	0.749/50.64	0.720/50.98	0.752/50.61	0.747/50.66	0.746±0.022/50.68±0.25
	Guo [7]		0.766/49.44	0.759/50.52	0.751/50.62	0.790/50.18	0.764/50.47	0.747/50.67	0.753±0.014/50.48±0.16
		✓	0.758/50.54	0.724/50.94	0.734/50.81	0.723/50.94	0.728/50.88	0.743/50.71	0.735±0.012/50.80±0.14
	Hui [1]		0.826/49.79	0.737/50.78	0.764/50.47	0.759/50.52	0.757/50.54	0.775/50.35	0.770±0.028/50.41±0.30
		✓	0.796/50.11	0.724/50.93	0.750/50.64	0.730/50.86	0.734/50.82	0.760/50.52	0.749±0.024/50.65±0.28
4 ×	Zuo [8]		1.456/44.86	0.866/49.38	0.968/48.41	0.998/48.15	0.904/49.00	1.171/46.76	1.060±0.201/47.76±1.54
		✓	1.395/45.24	0.773/50.37	0.917/48.88	0.932/48.74	0.848/49.56	1.109/47.23	0.996±0.201/48.34±1.68
	Guo [7]		1.323/45.69	0.807/50.00	0.921/48.84	0.974/48.36	0.860/49.42	1.078/47.48	0.994±0.170/48.30±1.41
		✓	1.282/45.97	0.741/50.74	0.879/49.26	0.874/49.30	0.781/50.28	1.050/47.71	0.934±0.183/48.88±1.61
	Hui [1]		1.474/44.76	0.822/49.83	0.968/48.41	1.003/48.11	0.893/49.11	1.107/46.57	1.060±0.212/47.80±1.69
		✓	1.416/45.11	0.769/50.41	0.941/48.66	0.939/48.68	0.839/49.65	1.125/47.10	1.005±0.214/48.27±1.74
8 ×	Zuo [8]		2.350/40.71	1.092/47.37	1.235/46.30	1.433/45.00	1.086/47.41	1.752/43.26	1.491±0.447/45.01±2.40
		✓	2.281/40.97	1.020/47.96	1.199/46.55	1.359/45.46	1.032/47.86	1.720/43.42	1.435±0.446/45.37±2.50
	Guo [7]		2.403/40.52	1.051/47.70	1.178/46.71	1.394/45.24	1.087/47.40	1.755/43.25	1.478±0.477/45.14±2.55
		✓	2.344/40.73	0.994/48.18	1.171/46.76	1.369/45.40	1.037/47.81	1.691/43.57	1.434±0.469/45.41±2.60
	Hui [1]		2.422/40.45	1.111/47.22	1.220/46.40	1.533/44.31	1.107/47.25	1.746/43.29	1.526±0.464/44.82±2.44
		✓	2.354/40.70	1.047/47.73	1.201/46.54	1.465/44.81	1.100/47.31	1.706/43.49	1.479±0.452/45.10±2.45
16 ×	Zuo [8]		3.746/36.66	1.729/43.38	1.634/43.84	2.294/40.92	1.657/43.74	2.624/39.75	2.281±0.750/41.38±2.61
		✓	3.563/37.10	1.535/44.41	1.558/44.28	2.236/41.14	1.520/44.50	2.487/40.22	1.150±0.735/41.94±2.74
	Guo [7]		3.982/36.13	1.651/43.77	1.574/44.18	2.412/40.48	1.618/43.95	2.731/39.40	2.328±0.860/41.32±2.96
		✓	3.956/36.19	1.621/43.94	1.546/44.35	2.358/40.68	1.571/44.21	2.706/39.48	2.293±0.863/41.47±3.01
	Hui [1]		3.854/36.41	1.810/42.98	1.665/43.70	2.622/39.77	1.729/43.37	2.704/39.49	2.397±0.774/40.95±2.64
		✓	3.782/36.58	1.666/43.70	1.600/44.05	2.320/40.82	1.597/44.07	2.699/39.51	2.277±0.789/41.45±2.78

III. EXPERIMENTAL RESULTS

In this section, we present qualitative and quantitative performance comparison results to demonstrate the superiority of the proposed method. To evaluate the effectiveness of the proposed method, we adopted the state-of-the-art depth map SR methods [1], [7], [8] as the baseline models.

A. Implementation Details

We trained the entire SR network for 500k iterations using a batch size of 32 in an end-to-end manner. We employed the Adam optimizer [15], which is a stochastic optimization method that involves the adaptive estimation of moments. The hyper-parameters of the Adam optimizer β_1 and β_2 were set to 0.9 and 0.999, respectively. We set the initial learning rate to 0.0001 and reduced the learning rate by multiplying it by 0.8 every 10 steps. Our experiments were conducted on CPU Intel(R) Xeon(R) CPU E3-1245 v5 and GPU TITAN X (Pascal), and implemented in TensorFlow v1.8.

In our experiments, we trained the networks using 58 RGB-D image pairs from the MPI Sintel dataset and 34 RGB-D image pairs (6, 10, and 18 image pairs collected from the 2001, 2006, and 2014 datasets, respectively) from the Middlebury dataset. We cropped 128×128 patches with stride 64 while excluding the patches without depth information, and augmented them with rotation and flipping. The depth images were downsampled by bicubic interpolation with various scaling factors (*i.e.*, 2, 4, 8,

TABLE II
PROCESSING TIME (S) OF THE BASELINE MODEL [8] W/ OR W/O THE SSC

SSC	Scaling factor				Average ±standard deviation
	2 ×	4 ×	8 ×	16 ×	
	0.2044	0.2310	0.2361	0.2380	0.2274±0.0135
✓	0.2380	0.2694	0.2826	0.2840	0.2685±0.0185

and 16) to obtain LR depth patches. During the test procedure, keeping consistent with recent methods [1]–[8], we employed the hole-filled Middlebury dataset for the given scaling factors.

B. Quantitative Comparison

We evaluated the effectiveness of the proposed SCC by measuring the RMSE and PSNR. We conducted extensive experiments while adjusting the scaling factors from 2 to 16. To confirm the generalizability of the proposed method, we applied the proposed method to diverse baseline SR networks on various scaling factors. The results are listed in Table I, where ✓ indicates that our SSC has been applied to the baseline model. The proposed method meaningfully improved the performances of all baseline models. Furthermore, we measured the average processing time of the baseline model [8] with and without SSC. As shown in Table II, the average processing time was increased from 0.23s to 0.27s by the adoption of the SSC.

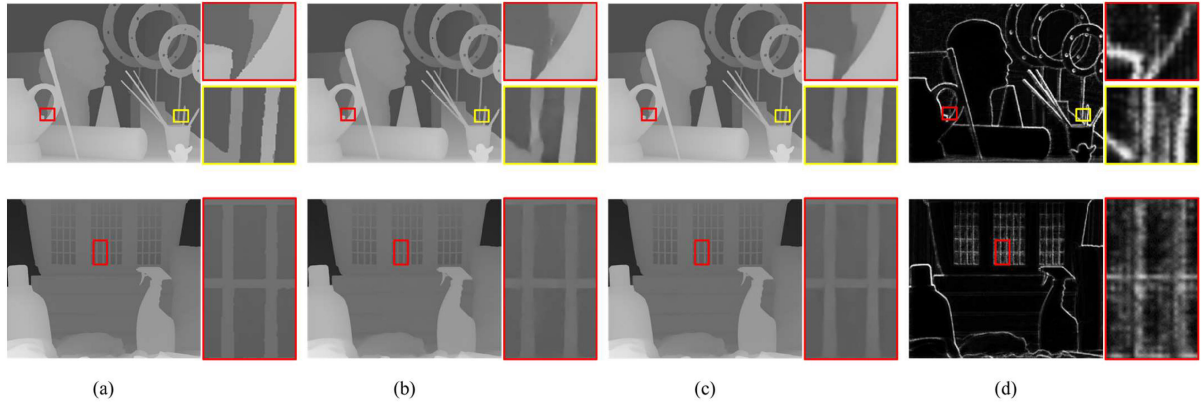


Fig. 3. Visual quality comparison using the Middlebury dataset (16 $\times$): (a) Original images, (b) resultant images of the baseline model [8], (c) resultant images of the proposed method, and (d) spatial weight maps in the SSC.

TABLE III
EXPERIMENTAL RESULTS OF THE ABLATION STUDIES ABOUT THE SSC IN
TERMS OF THE AVERAGE PSNR

Baseline model	SSC			Scaling factor			
	s	c	1×1 conv	$2 \times$	$4 \times$	$8 \times$	$16 \times$
Zuo [8]	✓	✓	✓	50.68	48.34	45.37	41.94
		✓	✓	50.35	48.15	45.08	41.67
	✓		✓	49.70	47.84	45.33	41.56
	✓	✓		50.50	47.79	44.97	41.56
Guo [7]	✓	✓	✓	50.80	48.88	45.41	41.47
		✓	✓	50.68	48.34	45.11	41.43
	✓		✓	50.32	48.54	45.24	41.33
	✓	✓		50.55	48.43	45.03	41.40

To analyze the effectiveness of the proposed method in detail, we also conducted ablation studies by removing each component of the SSC. As depicted in Table III, the proposed SSC trained without spatial attention or channel attention exhibits considerably poorer performance than the full model. This indicates that using both attention weights encourages the network to learn to effectively fuse two types of features. Furthermore, the SSC without the 1×1 convolutional layer shows a decrease in performance. These results demonstrate that the additional convolutional layer helps the depth features to be utilized in a diverse manner.

To assess the robustness of the proposed SSC, likewise with the existing methods [6]–[8], we conducted several extra-experiments that involved adding Gaussian noise with a standard deviation of 5 to the guide image or the depth map in the test datasets. Specifically, we added Gaussian noise with random standard deviation varying from 3 to 7 to the guide image or the depth map in the training datasets and retrained the networks. As presented in Table IV, the proposed method further improved the performance of the baseline model [8]. This indicates that the SSC is insensitive to noise in both the guide image and the depth map.

C. Qualitative Comparison

Fig. 3 compares the visual quality of the upscaled depth maps obtained by the baseline model [8] and the proposed method. As

TABLE IV
QUANTITATIVE COMPARISON ON THE NOISY DATASET FOR THE BASELINE
MODEL [8] IN TERMS OF THE AVERAGE PSNR

Noisy input	SSC	Scaling factor			
		$2 \times$	$4 \times$	$8 \times$	$16 \times$
Depth map		48.38	45.95	44.06	40.93
	✓	48.56	46.55	44.48	41.18
Guide image		49.92	48.16	45.09	41.21
	✓	50.92	48.57	45.19	41.60

depicted in Fig. 3(b), the baseline model yielded depth maps with blurry boundaries as can be noticed from the magnified regions of the sticks and basket frames. By contrast, the proposed method reconstructed sharp boundaries in these regions as shown in Fig. 3(c). Fig. 3(d) illustrates the spatial weight map w_s . Notably, depth boundaries received high attention, indicating that the depth SR network can effectively use F_g to render sharp depth boundaries. This demonstrates that the SSC played its intended role, *i.e.*, selective control of color and depth features for depth SR.

IV. CONCLUSION

In this letter, we introduced a novel feature concatenation method for CNN-based depth map SR. The proposed SSC effectively combines the depth and guide features depending on the spatial locations and channels. The SSC consistently contributed to the performance improvement of the existing CNN-based depth map SR methods without imposing training overhead or needing to significantly modify the original network architectures. In our future work, we will apply the SSC to other vision problems, such as image denoising and stereo matching, where the guide information can be utilized.

REFERENCES

- [1] T. W. Hui, C. C. Loy, and X. Tang, “Depth map super-resolution by deep multi-scale guidance,” in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 353–369.
- [2] W. Zhou, X. Li, and D. Reynolds, “Guided deep network for depth map super-resolution: How much can color help?” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, New Orleans, LA, USA, 2017, pp. 1457–1461.
- [3] G. Riegler, D. Ferstl, M. R  ther, and H. Bischof, “A deep primal-dual network for guided depth super-resolution,” in *Proc. BMVC*, 2016, pp. 7.1–7.14.

- [4] Y. Wen, B. Sheng, P. Li, W. Lin, and D. D. Feng, "Deep color guided coarse-to-fine convolutional network cascade for depth image super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 994–1006, Feb. 2019.
- [5] M. Ni, J. Lei, R. Cong, K. Zheng, B. Peng, and X. Fan, "Color-guided depth map super resolution using convolutional neural network," *IEEE Access*, vol. 5, pp. 26666–26672, 2017.
- [6] Z. Wang, X. Ye, B. Sun, J. Yang, R. Xu, and H. Li, "Depth upsampling based on deep edge-aware learning," *Pattern Recognit.*, vol. 103, 2020, Art. no. 107274.
- [7] C. Guo, C. Li, J. Guo, R. Cong, H. Fu, and P. Han, "Hierarchical features driven residual learning for depth map super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2545–2557, May 2019.
- [8] Y. Zuo, Q. Wu, Y. Fang, P. An, L. Huang, and Z. Chen, "Multi-scale frequency reconstruction for guided depth map super-resolution via deep residual network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 2, pp. 297–306, Feb. 2020.
- [9] L. Zhao, H. Bai, J. Liang, B. Zeng, A. Wang, and Y. Zhao, "Simultaneous color-depth super-resolution with conditional generative adversarial networks," *Pattern Recognit.*, vol. 88, pp. 356–369, 2019.
- [10] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 611–625.
- [11] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, nos. 1–3, pp. 7–42, 2002.
- [12] D. Scharstein and C. Pal, "Learning conditional random fields for stereo," in *Proc. CVPR*, 2007, pp. 1–8.
- [13] D. Scharstein *et al.*, "High-resolution stereo datasets with subpixel-accurate ground truth," in *Proc. GCPR*, 2014, pp. 31–42.
- [14] J. Park, S. Woo, J. Y. Lee, and I. S. Kweon, "BAM: Bottleneck attention module," in *Proc. BMVC*, 2018, pp. 1–14.
- [15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015, pp. 1–13.
- [16] O. Choi and S.-W. Jung, "A consensus-driven approach for structure and texture aware depth map upsampling," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3321–3335, Aug. 2014.
- [17] Y. Zuo, Q. Wu, J. Zhang, and P. An, "Explicit edge inconsistency evaluation model for color-guided depth map enhancement," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 2, pp. 439–453, Feb. 2018.
- [18] K. H. Lo, Y. C. F. Wang, and K. L. Hua, "Edge-preserving depth map upsampling by joint trilateral filter," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 371–384, Jan. 2018.
- [19] W. Liu, S. Jia, P. Li, X. Chen, J. Yang, and Q. Wu, "An MRF-based depth upsampling: Upsample the depth map with its own property," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1708–1712, Oct. 2015.
- [20] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, vol. 2, 2014, pp. 2672–2680.
- [21] Z. Zhang, "Microsoft Kinect sensor and its effect," *IEEE Multimedia*, vol. 19, no. 2, pp. 4–10, Feb. 2012.
- [22] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with Microsoft Kinect sensor: A review," *IEEE Trans Cybern.*, vol. 43, no. 5, pp. 1318–1334, Oct. 2013.
- [23] L. Keselman, J. Iselin Woodfill, A. Grunnet-Jepsen, and A. Bhowmik, "Intel RealSense stereoscopic depth cameras," in *Proc. CVPR Workshops*, 2017, pp. 1–10.
- [24] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, 2015, pp. 234–241.