IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 10, 2020, accepted July 31, 2020, date of publication August 10, 2020, date of current version August 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3015209

Deep Convolutional Grid Warping Network for Joint Depth Map Upsampling

YOONMO YANG, (Graduate Student Member, IEEE), DONGSIN KIM, AND BYUNG TAE OH, (Member, IEEE)

School of Electronics and Information Engineering, Korea Aerospace University, Goyang 10540, South Korea

Corresponding author: Byung Tae Oh (byungoh@kau.ac.kr)

This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of ICT under Grant NRF-2019R1F1A1063229, and in part by the GRRC Program of Gyeonggi Province (Study on 3D point cloud processing and application technology) under Grant 2020-B02.

ABSTRACT Depth maps play an important role in the representation of 3D information. They are often simultaneously acquired with color images; however, their resolution is significantly lower than that of color images owing to hardware limitations. In this paper, we propose a novel approach to upsample depth maps by using geometric deformation instead of pixel value refinement, which is employed in a majority of existing methods. This approach, known as grid warping, displaces the position of blurred pixels around the edge towards the center of the edge. The displacement vector for warping is obtained from an analysis of the corresponding high-resolution color image. Furthermore, we propose an edge signal and displacement vector modeling for a more effective analysis. The experimental results show that the proposed method significantly improves the quantitative and visual performance, as compared to state-of-the-art methods. The source codes of the proposed method will be available at https://github.com/yym064/DeepGridWarp.

INDEX TERMS Depth map upsampling, joint upsampling, grid warping, deep learning, CNN.

I. INTRODUCTION

Owing to the developments in 3D technologies, considerable attempts have been made to apply 3D technologies to various types of applications, including robotics and advanced driver assistance systems [8], [35]. Depth information plays a critical role in these applications for internal as well as external processing.

Passive and active methods are popularly used to acquire depth maps [16], [33], [37], [39], [41]. In the passive method, depth information is obtained indirectly; a typical example of this method is stereo matching, wherein depth information is estimated from two scenes with a binocular parallax. On the contrary, in the active method, the depth map is acquired directly. In this method, depth information is captured via special devices such as laser range scanners or time-of-flight cameras. Microsoft Kinect and SoftKinect are examples of devices used to directly capture depth information [39], [41]. However, the resolution and quality of the acquired depth map is generally low; as compared to RGB color images, owing to the limitations in the hardware technology.

The associate editor coordinating the review of this manuscript and approving it for publication was Long Wang.

Even though insufficient resolution can be partially solved by many of the existing upsampling schemes, its quality is significantly lower than the quality of color images, especially when attempting to considerably increase the resolution. A popular approach to address this problem is joint filtering, i.e., the upsampling filter is derived using the depth map as well as its corresponding color image. In the concept of joint filtering, it is assumed that the edge structure of the depth map is highly correlated to its corresponding color image. Therefore, a filter is designed to transfer meaningful structural information in the color edge to the depth map.

These approaches commonly result in two problems during information transfer. First, they transfer unwanted information such as texture patterns. This is because the information transfer is realized via the kernel method; however, the kernel computation is sensitive to small pixel changes such as texture pattern. Second, these methods frequently cause under- or over-shooting artifacts around the edge boundary due to inaccurate kernel estimations.

To further investigate the proposed approach, we first explore the existing works on image upsampling technology. We classify the existing upsampling methods into the following three categories: model-based kernel filtering, optimization problem, and deep learning approaches.

A. MODEL-BASED KERNEL FILTERING

A majority of the methods falling under this category employ the variational form of the adaptive nonlinear filter, such as the bilateral filter (BF) [32] or guided filter [10]. In [19], a joint bilateral upsampling (JBU) filter is proposed to incorporate the corresponding high-resolution (HR) color image information. Kim *et al.* proposed a trilateral filter to reduce the blurring artifact caused by the misalignment of the depth edge and the color edge [18]. Jung proposed an adaptive joint trilateral filter, wherein the color and depth maps are simultaneously restored according to the classification of the depth edge [15]. In [24], an extension of the joint bilateral filter is proposed to avoid the textural copying artifact which occurs when depth structure information is transferred to the depth map. To prevent artifacts, it integrates local gradient information during filtering. Chan *et al.* proposed a noise-aware filter (NAF) for depth upsampling [3]. It acts as a multilateral filter by adjusting the influence of color similarity to prevent texture copying artifacts. In [27], Min *et al.* proposed a weighted mode filter (WMF) that generates filter coefficients based on local statistical information induced by a histogram. Hua *et al.* proposed an extended guided filter (EGF) by inserting an additional term by considering the local 2nd order gradients of the depth map [12]. This filter employs an onion peel-like filtering, which significantly improves the performance of the filter. As an extension of EGF, Yang *et al.* proposed a confidence-based joint guided filter (CJGF) by controlling the filtering order using the confidence map derived from the shape of the unreliable region, depth map, and color pixel values [40].

B. OPTIMIZATION PROBLEM

The methods falling under this category build the objective function by considering various factors and attempt to minimize it. In [4], a depth upsampling problem is formulated with the Markov random field (MRF), wherein the data term is determined using a given depth map, and the smoothness term is determined using estimated HR depth samples derived from the HR color images. Based on MRF framework, Park *et al.* [30] proposed to use an additional term known as non-local mean regularization, which is implemented using the anisotropic structure-aware filter. Similar to the non-local mean filter, this term enables the contribution from faraway pixels during processing. Another MRF formulation was suggested by Lu *et al.* [25], wherein the truncated absolute difference between the estimated and the input depth value is employed for depth map upsampling. Liu and Gong proposed to use anisotropic heat diffusion filtering (ADF) [23], where the known pixels of depth maps are set as heat sources, and depth enhancement is performed by diffusing depth value from sources to unknown pixels based on color similarity.

C. DEEP LEARNING

The recent popularity of deep learning has motivated active research in deep learning-based approaches. In [5], a single image super resolution network based on a convolutional neural network (CNN) was proposed. Lim *et al.* proposed a deeper and more complicated network structure that consists of several residual blocks to extract meaningful features from the input image [22]. Harris *et al.* proposed a different approach using deep back-propagation network (DBPN) [9], where feature maps are first extracted using convolution layer, and then upsampled and downsampled repeatedly to feedback error in each stage. When multiple or multimodal input data such as depth map upsampling with corresponding color images are available, different deep network architectures can be considered. Hui *et al.* proposed a network which gradually upsamples the depth map by using color images as a reference [13]. Li *et al.* proposed the deep joint filtering (DJF) by using a two-stream network, wherein one stream extracts feature maps from the color image and the other stream extracts features from the depth map [21]. Then, the extracted feature maps are combined using a shallow network called the fusion network. In [36], Su *et al.* proposed a pixel adaptive convolution (PAC), which mimics the bilateral filter. Adopting a different approach, Kim *et al.* [17] focused on the receptive field for depth map upsampling; the receptive fields were enlarged using deformable kernel convolution (DKN).

In this paper, we propose a novel and distinct approach for a joint depth map upsampling algorithm with a deep network. Instead of directly inferring the HR depth map, or estimating the local-adaptive kernel, the proposed method reconstructs the low-resolution depth map by warping the pixel position without changing the depth intensity. The major contributions of the study are as follows:

- To the best of our knowledge, this is the first deep learning-based approach to upsample depth map via the image warping technique.
- We extract the displacement vector for grid warping from the corresponding color image and design the network for an efficient reconstruction of HR depth maps, using the estimated deformation information.
- We validated the proposed approach via mathematical edge modeling, which verifies the robustness of the proposed displacement vector estimation.

The remainder of this paper is organized as follows. In Section II, the warping method described in detail. The proposed system and its theoretical analysis are introduced in Section III. The implementation details and experimental results are provided in Section IV. Finally, the conclusions of the study are presented in Section V.

II. IMAGE RESTORATION BY GRID WARPING

For the purpose of image restoration, image warping methods were proposed in [1], [20], [28], especially aimed toward image deblurring. The basic assumption in these techniques is that the blurring process distorts the edge by shifting the pixels away from the true edge. Therefore, the remedy for deblurring should be the inverse process of pixel shift, i.e., shifting the pixels back toward the edge, as shown in Fig. 1(a).

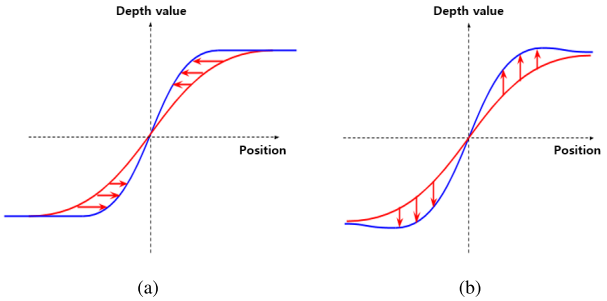

FIGURE 1. Comparison of signal reconstruction from blurred (red) to deblurred (blue): (a) pixel position shift (b) pixel value modification.

However, kernel based methods reconstruct the distorted pixel by shifting the pixel value, as shown in Fig. 1(b). For the convenience of explanation, we use the 1D edge profile of 2D images. Mathematically, in the 1D domain, restoring the blurred signal $I_b(x)$ to the reconstructed signal $\hat{I}(x)$ can be performed by determining the displacement function $d(x)$ as

$$\hat{I}(x) = I_b(x + d(x)) \tag{1}$$

Therefore, the core approach of a warping based restoration scheme is to determine the grid displacement function. Nasonova *et al.* [28] considered that the ideal step-edge, i.e., 1D edge, has the following profiles:

$$H(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{2}$$

If the blurring effect is modeled via Gaussian filtering, then the blurred edge can be modeled as

$$\begin{aligned} I_b(x) &= H(x) * G(x, \sigma) \\ &= \int_{-\infty}^{x} \delta(x) * G(t, \sigma)\, dt = \int_{-\infty}^{x} G(t, \sigma)\, dt \end{aligned} \tag{3}$$

where σ is the blurring parameter, $*$ indicates the convolution operation, and $\delta(t)$ is the delta function. This model shows that the edge profile modified due to blurring has the form of a cumulative Gaussian distribution function, as shown in Fig. 1.

In [28], the displacement function for (3) is obtained by the spring model as

$$d(x, \sigma) = \kappa \frac{d}{dx} G(x, \sigma) \tag{4}$$

where κ controls the sharpness of deblurring. The displacement vector has a positive value when $x < 0$, a negative value when $x > 0$, and a maximum value when $x = \pm\sigma$. As shown in (3) and (4), the overall performance of this scheme is significantly dependent on the determination of the true edge position (i.e., $x = 0$) and the optimal determination of σ.

III. PROPOSED METHOD

A. OVERALL SYSTEM STRUCTURE

The fundamental idea of the proposed scheme is to use the grid warping technique for depth map upsampling. As the conventional grid warping technique is designed to reconstruct a blurred image, the input image is resized to achieve the target resolution, and then the resized image is assumed as the blurred version of the ground-truth. However, unlike the conventional warping scheme stated in Section I, we consider the case that a pair of HR reference color image and low-resolution depth map is given, which enables us to infer displacement vector flows in a different manner.

As mentioned in the detailed literature review in Section II, the core step in deblurring via the warping scheme is the determination and localization of the displacement function $d(x)$. In the proposed scheme, we directly obtain the displacement vector from the given HR reference color image instead of directly estimating them.

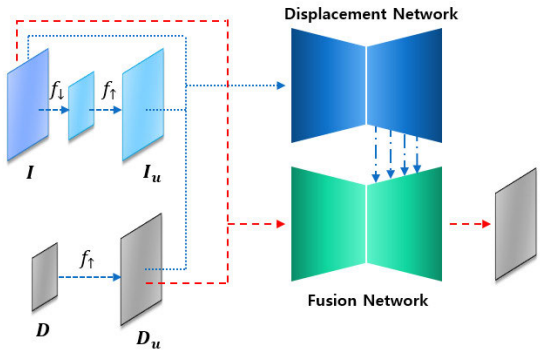

FIGURE 2. Overall system structure.

The overall process is depicted in Fig. 2. First, we downsample the HR color image $I(x)$ to have the same resolution of LR depth map $D(x)$, and upsample it to generate $I_u(x)$ via simple, pre-determined downsampling and upsampling methods, i.e.,

$$I_u(x) = f_\uparrow(f_\downarrow(I(x))) \tag{5}$$

where $f_\uparrow$ and $f_\downarrow$ indicate the upsampling and downsampling functions, respectively. $I_u(x)$ is assumed to be the blurred version of $I(x)$. Subsequently, the displacement vector can be extracted by analyzing the relationship between $I(x)$ and $I_u(x)$ as

$$\hat{d}(x) = \arg\min_{d(x)} \|I_u(x + d(x)) - I(x)\|^2 \tag{6}$$

Once $\hat{d}(x)$ are obtained, the target upsampled depth map $\hat{D}(x)$ is computed in the same manner:

$$\hat{D}(x) = D_u(x + \hat{d}(x)) \quad where \;\; D_u = f_\uparrow(D) \tag{7}$$

B. ANALYSIS

In this Section, we present the theoretical investigation and analysis of the proposed approach via signal modeling. As previously introduced in [28], the modeling for 1D edge profile is more focused on the analysis because edges mainly affect the overall upsampling performance, especially in depth maps.

First, it is preferable to use the ideal step edge model in (1), but we generalize it by convolving the Gaussian filter as

$$I(x, \sigma_c) = \int_{-\infty}^{x} G(t, \sigma_c)\, dt \tag{8}$$

Thus, the image edge varies smoothly rather than changing abruptly, and the varying speed is controlled by a small value σ_c (the step edge is the case of $\sigma_c = 0$). When the given image edge is blurred by upsampling, it can be modeled by convolution with another Gaussian filter as

$$I_u(x,\sigma_1) = I(x,\sigma_c) * G(x,\sigma) = \int_{-\infty}^{x} G\left(t, \sigma_1 = \sqrt{\sigma_c^2+\sigma^2}\right) dt \quad (9)$$

Similarly, the 1D depth map profile can be obtained as

$$D_u(x,\sigma_2) = I(x,\sigma_d) * G(x,\sigma) = \int_{-\infty}^{x} G\left(t, \sigma_2 = \sqrt{\sigma_d^2+\sigma^2}\right) dt \quad (10)$$

Based on the 1D modeling, it is concluded that the proposed approach can appropriately upsample the depth map by the following two properties.

Property I: The displacement vector is independent of the edge signal scale.

Proof I: In (3), when the signal is scaled by h, i.e., the edge signal is formed by a scaled step edge function given as

$$I'(x) = \int hH(t) * G(t)dt = h \cdot I(x),$$

its blurred signal will be

$$I'_u(x) = (I' * G)(x) = ((hI) * G)(x) = (h * (I * G))(x)$$

by the linear property of convolution operation. Therefore,

$$\arg\min_{d(x)} \|I'_u(x+d(x)) - I'(x)\|^2 = \arg\min_{d(x)} \|h\,(I_u(x+d(x)) - I(x))\,\|^2 = \hat{d}(x) \quad (11)$$

Property II: The error caused by replacing the displacement vector in the depth map with that of the color image is approximately proportional to σ_c/σ. Thus, the error reduces when the blurring artifacts are dominant ($\sigma_c \ll \sigma$).

Proof II: As in (9), the edge parameter σ_c of the original color signal is changed to $\sqrt{\sigma_c^2+\sigma^2}$ by blurring. Let the edge parameter for depth map $\sigma_d = s\sigma_c$. Then, the edge parameter in the blurred depth signal will have $\sqrt{s^2\sigma_c^2+\sigma^2}$. To determine the extent of the influence of the change of edge parameter on the disparity vector, we consider the x-directional variation by σ variation. We need to find $dx/d\sigma = dI^{-1}/d\sigma$ in (8); however, its close-form solution cannot be derived. Instead, we consider the sigmoid function for (8) as in [42]

$$I(x,\sigma) \approx \frac{1}{1+e^{-x/p\sigma}} \quad (12)$$

where $p^{-1} = 0.9\sqrt{\pi}$ is a constant. From this equation, we can obtain its inverse function as

$$x = -p\,\sigma\,\log\left(1-\frac{1}{y}\right) \quad (13)$$

Therefore,

$$\frac{dx}{d\sigma} = -p\log\left(1-\frac{1}{y}\right) \quad (14)$$

From (14), the x-directional variation according to σ is only a function of y. Therefore, we can infer that the optimal value for the displacement vector is proportional to the difference of two edge parameters, i.e.,

$$\begin{aligned} d_c(x) &= k\left(\sqrt{\sigma_c^2+\sigma^2}-\sigma_c\right) \\ d_d(x) &= k\left(\sqrt{s^2\sigma_c^2+\sigma^2}-s\sigma_c\right) \end{aligned} \quad (15)$$

where k is a proportional parameter. Intuitively, $s = 1$ (equivalently $\sigma_c = \sigma_d$) will provide an error-free result. When $s \neq 1$, the relative error, E_r, can be computed as

$$E_r = \frac{d_c(x)-d_d(x)}{d_d(x)} = \frac{\sqrt{1+C^2}-1}{\sqrt{s^2+C^2}-s} - 1 \approx \frac{C-1}{C-s} - 1 = \frac{s-1}{C-s} \quad (16)$$

where $C = \sigma/\sigma_c \gg 1$ (i.e., the blurring parameter for upsampling is significantly larger than the edge parameter) is used for the approximation. Furthermore, $0 \leq s < 1$ (i.e., the depth map has a more rapidly varying edge than the color image), and $C \gg s$ in most cases. As a result, we can conclude that $|E_r| \ll 1$. Additionally, this derivation provides a numerical model of the estimated relative error.

C. NETWORK ARCHITECTURE

The function blocks of the proposed system are presented in Fig. 2 and implemented in the deep networks. The left network, called displacement network, seeks the displacement vectors from the HR color image. The right network, called fusion network, attempts to reconstruct the depth map using the transferred displacement vector. Both networks are concatenated and trained end-to-end. Additional details on the network design are presented in below.

1) DISPLACEMENT NETWORK

The displacement network is designed to estimate the displacement vector at each pixel position. As stated in Section I, signal blurring is assumed to be the outward shift of pixels from the true edge. Obtaining displacement vectors between two views is similar to obtaining an optical flow. Therefore, we adopt a state-of-the-art optical flow FlowNetS structure, as shown in Fig. 3(a) [6].

As stated in Section II, the previous approaches in [20], [28] attempted to localize the true edge before applying grid warping. This is because it determines the image warping direction around the true edge and significantly affects overall performance. However, the proposed approach does not involve this constraint because the corresponding color image is used as a reference, thereby sufficiently guiding deformation direction. Here, we assume the color and depth maps to be perfectly aligned. However, slight misalignments can also be managed due to its multiresolution architecture.

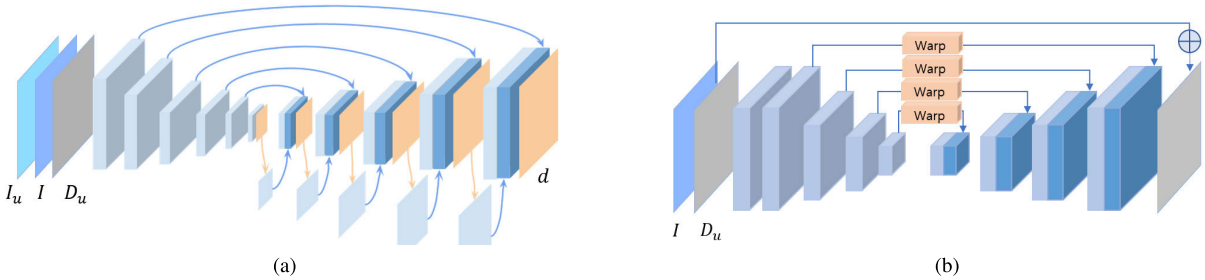

FIGURE 3. (a) Displacement network, (b) fusion network.

2) FUSION NETWORK

The fusion network consists of three steps: feature extraction, feature warping, and reconstruction, as presented in Fig. 3(b).

In the feature extraction part, we adopt an architecture similar to the encoder of the autoencoder. It consists of five convolutional layers. The first two convolutional layers use the common convolution, and the remaining three layers employ the stride convolution to extend a receptive field with small network parameters. The input of this network is the upsampled depth map D_u, and 64 feature maps are extracted in each resolution. During the feature warping step, the extracted feature maps are shifted by the displacement vector obtained from the displacement network. In this process, the spatial transformer network is adopted for the realization of the shifting operation, because it is eligible to represent various operations including scaling, cropping, rotation, and non-rigid deformations [14]. It can be also trained with a standard backpropagation method, which makes the proposed system end-to-end trainable. Using the spatial transformer network, feature maps are shifted to align the center edge position in each resolution level.

During the reconstruction step, the network fuses the warped feature maps to reconstruct the HR depth map image. This network has an architecture similar to the decoder of the autoencoder and contains five convolutional layers as feature extraction. From the lowest to the highest resolutions, the warped feature maps are upsampled via bilinear interpolation and concatenated with the feature map having the next resolution; they are then sequentially convolved in each convolutional layer. In the conventional autoencoder network, skip-connection and feature maps concatenation techniques are commonly used for fast training and to avoid gradient vanishing, especially when the network structure is considerably deep. However, note that the proposed method does not employ these techniques in the middle of the reconstruction network, because the extracted feature maps in the encoder are deformed by warping; therefore; skip-connection would degrade the performance.

3) LOSS FUNCTION

For the network training, the L1 norm is used. For a given network output D_p and the ground truth depth map D_{gt}, the loss function can be formulated as

$$L = \frac{1}{N}\sum_{i=1}^{N}\sum_{j}\left\| D_p^i(j) - D_{gt}^i(j) \right\|_1 \tag{17}$$

where N and j are the number of training samples and the pixel position, respectively.

IV. EXPERIMENT

In this Section, the implementation details and test datasets are introduced. Subsequently, the proposed method is compared with various state-of-the-art depth map upsampling methods, quantitatively and visually. Furthermore, we conducted extensive experiments to further analyze the proposed method in various situations.

A. IMPLEMENTATION DETAILS

Similar to [17], we trained the network to upsample depth maps for scale factors of 2, 4, 8, and 16, with random initializations, respectively. We used the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate starts at $1e^{-4}$ and is divided by 5 at every 5 epochs. The batch size is 1. Our experiments were performed on an Ubuntu operating system. We trained a network by using a GTX 1080Ti GPU card. The network is trained in an end-to-end manner using PyTorch [31].

B. DATASETS

For a fair comparison, we collected four popular datasets. The first two datasets were divided into training and evaluation sets. The other two were only used for evaluation. The details on the datasets are given below:

1) Sintel dataset [2]. This is a computer graphic video with fine textures. It provides color and depth map pair video sequences. Each sequence consists of either 50 or 40 frames. The resolution of the sequences is 1024×438. The color image and the depth map are well aligned. A total of 1000 color-depth image pairs are used as the training dataset, and a total of 300 color-depth image pairs as the testing dataset.
2) NYU v2 dataset [34]. This dataset consists of RGB/D image pairs captured with the Microsoft Kinect. The resolution of the image pairs is 640×480. We split the image pairs into 800 training dataset and 600 testing datasets.
3) Lu dataset [26]. Six RGB/D image pairs were provided. The resolution of the LU dataset is 640×480. They were acquired using the ASUS Xtion Pro camera. This dataset was only used for evaluation.

TABLE 1. Quantitative performance comparisons in terms of RMSE.

Dataset	Lu				NYU v2				Sintel				Middlebury			
Methods	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
BF [32]	2.953	3.423	5.398	22.013	1.666	2.378	6.970	21.209	3.460	4.347	5.877	18.692	2.173	2.978	4.368	17.161
JBU [19]	2.247	3.838	5.829	8.495	1.368	2.623	4.975	8.252	3.261	4.415	5.951	8.388	1.908	3.104	4.796	7.630
ADF [23]	1.605	3.500	8.960	9.562	1.141	2.000	6.836	8.394	4.485	9.725	20.150	17.884	1.668	2.694	4.201	7.089
NAF [3]	2.228	3.798	5.741	8.370	1.268	2.411	4.472	7.129	3.239	4.407	5.861	8.230	1.768	2.862	4.404	6.950
WMF [27]	3.752	3.742	6.056	22.213	1.942	2.648	7.286	21.258	3.755	4.685	6.253	18.798	2.568	3.286	4.826	17.240
EGF [12]	2.063	3.576	6.073	10.846	1.222	2.501	4.512	7.294	3.665	6.249	9.333	12.859	1.759	3.280	6.157	9.236
CJGF [40]	2.327	2.263	3.625	7.444	1.663	2.014	3.244	5.441	3.537	4.207	5.422	7.594	1.936	2.255	3.597	6.302
DJF [21]	1.161	2.393	3.938	6.784	1.181	1.728	2.595	4.859	2.710	3.619	4.879	7.914	1.115	1.822	3.323	6.197
PAC [36]	1.022	1.386	3.417	6.012	0.869	1.520	2.746	4.317	2.554	3.722	5.419	7.500	1.009	1.609	3.543	5.742
DBPN [9]	1.036	1.509	3.320	6.230	0.729	1.385	2.952	4.778	1.383	3.524	5.351	7.668	0.985	1.446	3.017	5.564
DKN [17]	0.734	1.135	2.419	6.100	0.660	1.080	1.911	3.522	1.367	2.916	4.546	5.770	0.984	1.317	2.349	4.492
Ours	**0.465**	**1.036**	**2.241**	**5.089**	**0.498**	**0.912**	**1.602**	**3.179**	**0.986**	**2.664**	**3.806**	**4.920**	**0.818**	**1.270**	**2.073**	**4.063**

TABLE 2. Quantitative performance comparisons in terms of MAE.

Dataset	Lu				NYU v2				Sintel				Middlebury			
Methods	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
BF [32]	0.804	0.866	1.271	5.485	0.556	0.698	1.510	4.885	0.416	0.488	0.678	3.114	0.856	1.056	1.512	4.927
JBU [19]	0.597	1.043	1.852	3.537	0.425	0.817	1.748	3.411	0.327	0.563	0.997	1.922	0.734	1.139	1.828	3.235
ADF [23]	0.376	0.943	2.067	4.453	0.292	0.648	2.039	3.304	0.434	1.573	6.205	5.143	0.562	1.013	1.777	3.486
NAF [3]	0.573	1.005	1.796	3.463	0.413	0.778	1.578	2.975	0.330	0.561	0.951	1.704	0.693	1.058	1.687	2.955
WMF [27]	0.557	0.636	1.347	5.732	0.413	0.613	1.622	4.965	0.216	0.339	0.631	3.111	0.771	1.013	1.593	4.978
EGF [12]	0.252	0.596	1.314	3.321	0.266	0.631	1.309	2.704	0.313	0.627	1.231	2.251	0.533	1.009	2.014	3.605
CJGF [40]	0.279	0..406	0.925	2.305	0.307	0.542	1.096	2.258	0.248	0.446	0.766	1.540	0.574	0.802	1.390	2.760
DJF [21]	0.768	0.829	1.250	2.617	0.727	0.851	1.176	2.362	0.834	1.023	1.189	2.539	0.700	0.990	1.600	3.202
PAC [36]	0.322	0.536	1.445	2.896	0.318	0.630	1.289	2.356	0.396	0.722	1.502	2.621	0.514	0.813	1.804	3.241
DBPN [9]	0.229	0.543	1.233	2.524	0.206	0.550	1.243	2.296	0.075	0.476	1.032	2.097	0.438	0.686	1.297	2.741
DKN [17]	0.103	0.240	**0.599**	1.971	0.166	0.368	0.748	1.673	0.083	0.309	0.781	1.340	0.415	0.587	0.968	2.117
Ours	**0.070**	**0.231**	0.601	**1.616**	**0.134**	**0.329**	**0.633**	**1.484**	**0.060**	**0.283**	**0.531**	**0.975**	**0.383**	**0.583**	**0.924**	**1.926**

4) Middlebury dataset [11]. It consists of 30 RGB/D image pairs from the 2001-2006 datasets. It is only used for evaluation.

From the sampled image, we generate image patches of size 150 × 150 for training models. To enrich constructed image patches more, we adopt data augmentation techniques, including vertical/horizontal flip, 90° rotation, and random crop. For a fair comparison, we retrain the reference algorithm using our training dataset.

C. QUANTITATIVE EVALUATION

For the evaluation of the proposed method, a few superior upsampling schemes are selectively compared, including model-based filtering, optimization, and deep learning-based methods. As mentioned in the previous Section, we evaluate our model using four different evaluation datasets, which have different resolutions and different color-depth alignment quality.

Tables 1 and 2 exhibit the average root mean square error (RMSE) and mean absolute difference error (MAE) value for each scheme, respectively. The lowest RMSE and MAE values are presented in bold red, and the second lowest are presented in blue. It is observed that the proposed method achieves the best performance for almost all test cases in terms of the both RMSE and MAE. The only exemption shows the second-best performance with negligible difference. For the upsampling factor of four, DKN is comparable with the proposed method, whereas the proposed method outperforms for all other upsampling factors.

For the computational complexity, we measure the runtime for the deep learning-based schemes of DJF, PAC, DKN, and the proposed method on the same machine. The proposed scheme took an average of 15 ms for the Middlebury datasets, which was slower than DJF and PAC, but significantly faster than DKN.

Finally, we investigated how much the proposed grid-warping-based scheme could improve the upsampling performance. To evaluate the potential of the proposed approach, we replaced the input of the displacement network I and I_u by D_u and D_{gt}, and the network was trained with the same

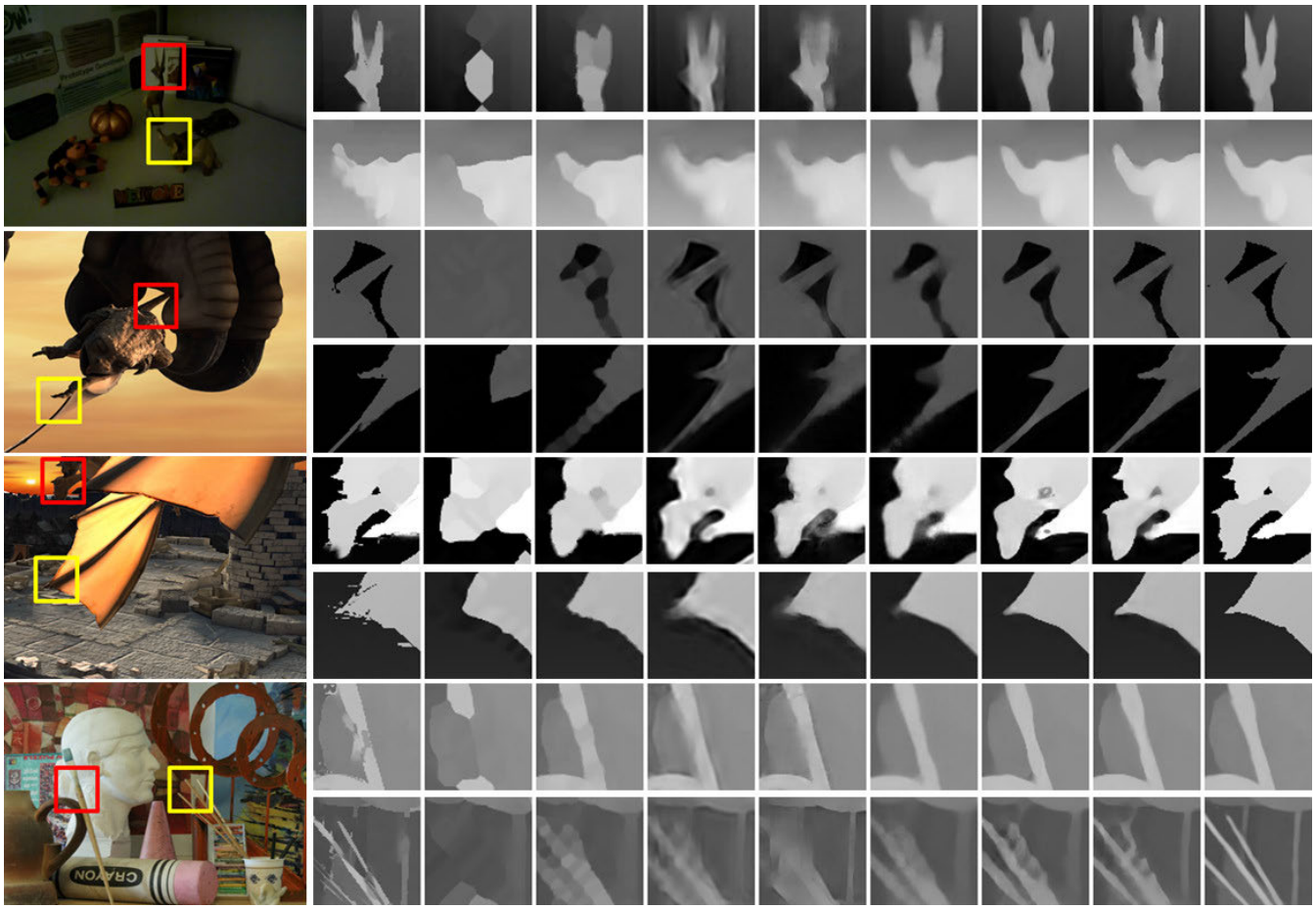

FIGURE 4. Visual comparisons for the 8× upsampled *Lu*, *Temple 3*, *Temple 2*, and *Art* depth maps by various methods: BF [32], EGF [12], CJGF [40], DJF [21], PAC [36], DBPN [9], DKN [17], Ours, and ground-truth (from left to right).

TABLE 3. Run-time comparisons for joint deep learning-based approaches.

Methods	DJF [21]	PAC [36]	DKN [17]	Ours
Run-times (ms)	2.2	3.0	152.7	15.2

approach. This experiment assumed that ground-truth displacement vectors could be derived by D_u and D_{gt}. As shown in Table 4, it is reported that the warping-based approach was able to recover almost error-free upsampled depth maps. It shows there is still room for improvement, especially for large-scale factors.

D. VISUAL COMPARISON

We examined the effectiveness of the proposed warping based scheme in handled challenging cases in joint depth upsampling. As shown in Fig. 4, we chose a few complicated regions from various images and compared the performance in the zoomed images. In most cases, our scheme was capable of maintaining a sharper shape of the depth map edge as compared to the other schemes. For example, as shown in the first row in Fig.4, the proposed method achieved the sharpest depth edge while well preserving the overall shape of the object. In the case of multiple overlapped objects in the fifth row, each object is clearly separated in our proposed scheme. We also test our scheme in more challenging situation such as low-resolution images. In general, upsampling a low-resolution image is more challenging than upsampling a high-resolution image, because low-resolution image has more complicated patterns than high-resolution image in the same size of region. Although, we did not train our model using low-resolution datasets, the proposed scheme achieved relatively sharper depth edge compared to any other schemes as shown in second and third row of Fig 5. However, when small objects disappeared during downsampling, as shown in the last row of 4, similar to the other schemes, the proposed method was unable to sufficiently restore the depth map. Another weakness of our scheme is color image dependency. When the proposed method restores edge of depth map, it tends to mimic the shape of color image, not the shape of depth map. For example, Although the visual results in Fig.4 are predicted by one model, sharpness of each visual result is different. This phenomenon is more clearly when comparing the real image with the computer graphic image. as shown in Fig.4, the second row result is much blurred than the third row result. It is because the sharpness of third row color image is much sharp than the second row.

FIGURE 5. Visual comparisons for the 8× upsampled *Middlebury* and *NYU v2* depth maps by various methods: BF [32], EGF [12], CJGF [40], DJF [21], PAC [36], DBPN [9], DKN [17], Ours, and ground-truth (from left to right).

TABLE 4. RMSE of the proposed system using the displacement vector derived from the ground-truth depth map.

Dataset	Lu			NYU v2			Sintel			Middlebury		
Methods	2×	4×	8×	2×	4×	8×	2×	4×	8×	2×	4×	8×
Ours	0.465	1.036	2.241	0.498	0.912	1.602	0.986	2.664	3.806	0.818	1.270	2.073
Ours with D_{gt}	0.068	0.254	0.405	0.064	0.196	0.422	0.343	0.451	0.811	0.117	0.259	0.655

E. DISPLACEMENT VECTOR ANALYSIS

In this Section, we visualized the displacement vector field to verify the assumption of grid warping. The displacement vector field was displayed in the same manner as the conventional optical flow visualization in Fig. 6. Overall, displacement vectors were formed around the edge at orthogonal and opposite directions with respect to the edge. These two observations are consistent with our hypothesis. However, some unexpected chessboard patterns were observed. We analyzed these types of patterns that would be generated by the transpose convolution operation that was used to increase the resolution of feature maps. It was reported that the transposed convolution often caused chessboard patterns in the output image owing to uneven overlaps [29]. To further confirm the effect of the transpose convolution, we replaced the transpose convolution layer with the bilinear interpolation. As shown in Fig. 6, the modified model with bilinear interpolation yielded the pattern-free displacement vector fields. However, we want to maintain use of the transpose convolution owing to its higher performance.

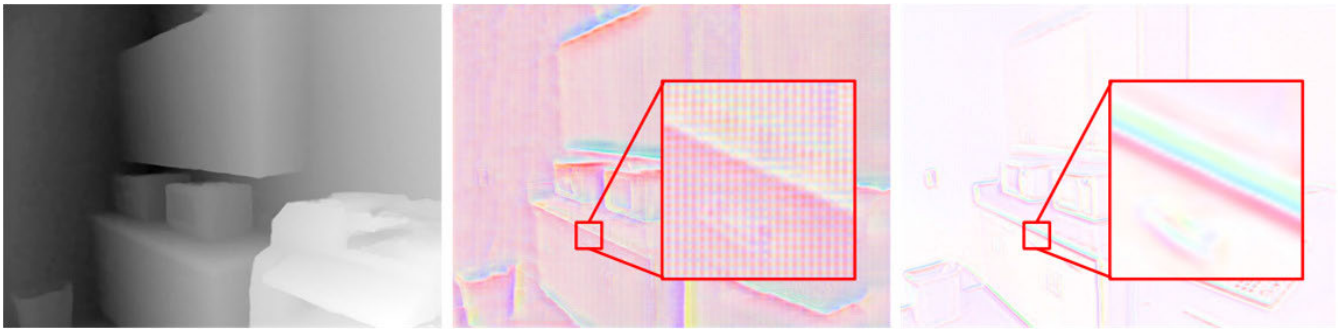

FIGURE 6. Displacement vector field visualization (scale 8×): ground-truth depth map, displacement vector field with transpose convolution, and its pattern-free vector field without transpose convolution (from left to right).

TABLE 5. Complexity comparisons for joint deep learning-based approaches.

Methods	Run-time (ms)	# of parameters (M)	FLOPs (G)
DJF [21]	2.2	0.30	19.53
PAC [36]	3.0	0.22	6.15
DKN [17]	152.7	1.16	111.89
Ours	15.2	34.23	71.51

TABLE 6. Complexity analysis of each network in the proposed scheme.

Methods	Displacement network	Fusion network	Total
# of Parameters (M)	33.18	1.05	34.23
FLOPs	28.73	42.77	71.51

F. COMPLEXITY COMPARISON

In this Section, more intensive complexity analysis and comparisons are conducted. We measure the run-time, the number of parameters, and the floating-point operations per second (FLOPs), where FLOPs are measured with upsampled 256×256 input depth map. As exhibited in Table 5, the proposed scheme has largest number of parameters compared to other schemes, especially approximately 30 times more than DKN. On the other hand, the proposed scheme has about 60% FLOPs compared to DKN. It is because the proposed scheme deals with feature maps in lower resolution, and it results in fewer operations. For the further analysis of the proposed scheme, the complexity of each network is measured. As reported in Table 6, the displacement network accounts for 97% of the total number of parameters, whereas the fusion network performs more FLOPs. It is analysed that the larger resolution feature maps are more processed in the fusion network.

G. NOISY ENVIRONMENT EVALUATION

In this Section, we consider more challenging test scenarios for practical applications. First, we tested the case when the edge of color image is more blurred. In the proposed system, the computation of displacement vector highly relies on the quality of HR color image and could degrade the performance when its quality is low. We consider that the given image is distorted by blurring, and it was simply modeled via Gaussian blurring with σ_b. As shown in Table 7, the blurred color image slightly decreases the performance. For the weakly blurred case (e.g., $\sigma_b \leq 2$), the performance degradation of the proposed system was small compared to other deep learning-based joint upsampling methods. However, the performance worsens when σ_b is large. It is well matched with the proposed mathematical modeling in Section III.B, i.e., the overall performance degradation is still small for $\sigma \ll \sigma_b$, whereas performance worsens for larger σ_b. It is also noteworthy that the proposed method is slightly more robust toward the blurry reference image than other joint upsampling methods are more sensitive towards the blurry reference image.

TABLE 7. Performance variation for Gaussian blurred (with σ_b) reference color image (Middlebury, 16× scaling, RMSE).

Methods	$\sigma_b = 0$	$\sigma_b = 1$	$\sigma_b = 2$	$\sigma_b = 3$	$\sigma_b = 4$
DJF [21]	6.079	6.079	6.191	6.320	6.433
DKN [17]	4.492	4.542	4.674	4.818	4.926
Ours	4.063	4.017	4.148	4.316	4.442

Second, we test the case of a color-depth misalignment situation. Most of the joint depth map schemes assume that color and depth map are well aligned. However, this is not true for some practical situations. To generate the misalignment, we shifted the color images in the test dataset to the n pixel in the horizontal and vertical directions. As shown in Table 8, the proposed model does not degrade much for small pixel shift, but the performance drop becomes larger for large misalignment as all other joint upsampling methods do. It is analyzed that slight misalignment can be managed by multiresolution structure, while misuse of the displacement vector by large misalignment is unavoidable, which results in large performance degradation.

H. JOINT SALIENCY MAP UPSAMPLING

In this Section, we explore how effectively the proposed scheme can be applied to saliency map upsampling [38]. To measure the accuracy of upsampled saliency map, we use RMSE and structure-measure(S-measure) metric [7]. Here, S-measure evaluates region-aware and object-aware structural similarity between ground truth and predicted

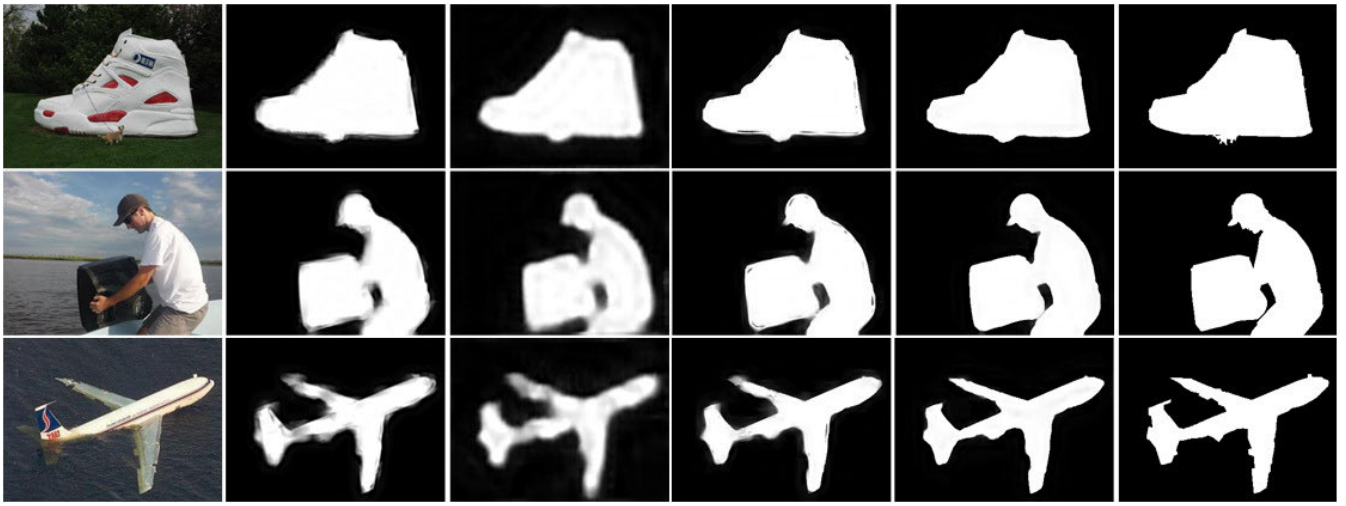

FIGURE 7. Visual comparisons of 16× upsampled saliency map results: Color image, DJF [21], PAC [36], DKN [17], Ours, and ground-truth (from left to right).

TABLE 8. Performance variation by misalignment with pixel shift (with n) of color image (Middlebury, 16× scaling, RMSE).

Methods	$n = 0$	$n = 1$	$n = 2$	$n = 3$
DJF [21]	6.079	6.140	6.341	6.568
DKN [17]	4.492	4.611	4.992	5.292
Ours	4.063	4.189	4.696	5.037

TABLE 9. Quantitative comparisons on saliency map upsampling.

Methods	DJF [21]	PAC [36]	DKN [17]	Ours
RMSE	21.996	23.466	19.719	18.000
S-measure [7]	0.948	0.930	0.957	0.963

upsampled saliency map. DUT-OMRON dataset is used for the evaluation. As reported in Table 9, the proposed scheme shows the best performance among the joint deep learning-based approaches in terms of RMSE (lower is better) and S-measure (higher is better). Fig. 7 shows the visual comparisons. We observe that fine-details and its structures are well preserved compared to any other schemes.

V. CONCLUSION

In this paper, we proposed a novel depth map upsampling technique by the image warping approach. The displacement vector for the image deformation was computed by the corresponding HR color information, which is the major contribution of the study. Furthermore, we also provided the theoretical edge signal modeling to verify the robustness of the proposed approach. As a result, the proposed scheme outperformed model-based approaches and exhibited the best performance, as compared to other state-of-the-art deep learning-based schemes, in terms of the RMSE and MAE. The visual results also validate the superiority of the proposed scheme. Furthermore, more intensive experiments are provided to analyze the proposed method with various situations. However, the performance of the proposed method relies on the similarities of the color images and depth maps. This limitation will be addressed in a future work.

ACKNOWLEDGMENT

(Yoonmo Yang and Dongsin Kim contribute equally to this work.)

REFERENCES

[1] N. Arad and C. Gotsman, "Enhancement by image-dependent warping," *IEEE Trans. Image Process.*, vol. 8, no. 8, pp. 1063–1074, Aug. 1999.

[2] 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.* Cham, Switzerland: Springer, 2012, pp. 611–625.

[3] D. Chan, H. Buisman, C. Theobalt, and S. Thrun, "A noise-aware filter for real-time depth upsampling," in *Proc. Workshop Multi-Camera Multi-Modal Sensor Fusion Algorithms Appl.*, Oct. 2008.

[4] J. Diebel and S. Thrun, "An application of Markov random fields to range sensing," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2006, pp. 291–298.

[5] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.

[6] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. V. D. Smagt, D. Cremers, and T. Brox, "FlowNet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2758–2766.

[7] D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li, and A. Borji, "Structure-measure: A new way to evaluate foreground maps," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4548–4557.

[8] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.

[9] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1664–1673.

[10] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.

[11] H. Hirschmuller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[12] K.-L. Hua, K.-H. Lo, and Y.-C.-F. Frank Wang, "Extended guided filtering for depth map upsampling," *IEEE MultimediaMag.*, vol. 23, no. 2, pp. 72–83, Apr. 2016.

[13] T. W. Hui, C. C. Loy, and X. Tang, "Depth map super-resolution by deep multi-scale guidance," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 353–369.

[14] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.
[15] S.-W. Jung, "Enhancement of image and depth map using adaptive joint trilateral filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 2, pp. 258–269, Feb. 2013.
[16] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 9, pp. 920–932, Sep. 1994.
[17] B. Kim, J. Ponce, and B. Ham, "Deformable kernel networks for guided depth map upsampling," 2019, *arXiv:1903.11286*. [Online]. Available: http://arxiv.org/abs/1903.11286
[18] J. Kim, J. Lee, S. Han, D. Kim, J. Min, and C. Kim, "Trilateral filter construction for depth map upsampling," in *Proc. IVMSP*, Jun. 2013, pp. 1–4.
[19] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," *ACM Trans. Graph.*, vol. 26, no. 3, p. 96, Jul. 2007.
[20] A. Krylov, A. Nasonov, and Y. Pchelintsev, "Single parameter post-processing method for image deblurring," in *Proc. 7th Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Nov. 2017, pp. 1–6.
[21] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Joint image filtering with deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1909–1923, Aug. 2019.
[22] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 136–144.
[23] J. Liu and X. Gong, "Guided depth enhancement via anisotropic diffusion," in *Proc. Pacific-Rim Conf. Multimedia*. Cham, Switzerland: Springer, 2011., pp. 408–417.
[24] K.-H. Lo, Y.-C. Wang, and K.-L. Hua, "Joint trilateral filtering for depth map super-resolution," in *Proc. Vis. Commun. Image Process. (VCIP)*, Nov. 2013, pp. 1–6.
[25] J. Lu, D. Min, R. S. Pahwa, and M. N. Do, "A revisit to MRF-based depth map super-resolution and enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 985–988.
[26] S. Lu, X. Ren, and F. Liu, "Depth enhancement via low-rank matrix completion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3390–3397.
[27] D. Min, J. Lu, and M. N. Do, "Depth video enhancement based on weighted mode filtering," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1176–1190, Mar. 2012.
[28] A. Nasonova and A. Krylov, "Deblurred images post-processing by Poisson warping," *IEEE Signal Process. Lett.*, vol. 22, no. 4, pp. 417–420, Apr. 2015.
[29] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, vol. 1, no. 10, p. e3, Oct. 2016.
[30] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon, "High quality depth map upsampling for 3D-TOF cameras," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1623–1630.
[31] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.
[32] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664–672, Aug. 2004.
[33] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2003, pp. 195–202.
[34] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2012, pp. 746–760.
[35] G. P. Stein, Y. Gdalyahu, and A. Shashua, "Stereo-assist: Top-down stereo for driver assistance systems," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2010, pp. 723–730.
[36] H. Su, V. Jampani, D. Sun, O. Gallo, E. Learned-Miller, and J. Kautz, "Pixel-adaptive convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11166–11175.
[37] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 7, pp. 787–800, Jul. 2003.
[38] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3166–3173.
[39] L. Yang, L. Zhang, H. Dong, A. Alelaiwi, and A. El Saddik, "Evaluating and improving the depth accuracy of Kinect for windows v2," *IEEE Sensors J.*, vol. 15, no. 8, pp. 4275–4285, Aug. 2015.
[40] Y. Yang, H. S. Lee, and B. T. Oh, "Depth map upsampling with a confidence-based joint guided filter," *Signal Process., Image Commun.*, vol. 77, pp. 40–48, Sep. 2019.
[41] Z. Zhang, "Microsoft Kinect sensor and its effect," *IEEE MultimediaMag.*, vol. 19, no. 2, pp. 4–10, Feb. 2012.
[42] B. Zogheib and M. Hlynka, *Approximations of the Standard Normal Distribution*. Windsor, ON, Canada: Univ. of Windsor, Department of Mathematics and Statistics 2009.

YOONMO YANG (Graduate Student Member, IEEE) received the B.S. and M.S. degrees from the School of Electronics and Information Engineering, Korea Aerospace University, Goyang, South Korea, in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree. His research interest includes image/video restoration and compression.

DONGSIN KIM received the B.S. degree from the School of Electronics and Information Engineering, Korea Aerospace University, Goyang, South Korea, in 2020, where he is currently pursuing the M.S. degree. His research interest includes image/video restoration and compression.

BYUNG TAE OH (Member, IEEE) received the B.S. degree from Yonsei University, Seoul, South Korea, in 2003, and the M.S. and Ph.D. degrees from the University of Southern California (USC), Los Angeles, CA, USA, in 2007 and 2009, respectively, all in electrical engineering. From 2009 to 2013, he was a Research Staff with the Samsung Advanced Institute of Technology (SAIT), Samsung Electronics, South Korea. Since 2013, he has been with the School of Electronics and Information Engineering, Korea Aerospace University (KAU), where he is currently an Associate Professor. His research interests include image/video restoration and compression, and image/video forensics.