Deep Edge Map Guided Depth Super Resolution

Zhongyu Jiang[a], Huanjing Yue[a,*], Yu-Kun Lai[b], Jingyu Yang[a], Yonghong Hou[a], Chunping Hou[a]

[a]*the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China*
[b]*the School of Computer Science and Informatics, Cardiff University, UK*

Abstract

Accurate edge reconstruction is critical for depth map super resolution (SR). Therefore, many traditional SR methods utilize edge maps to guide depth SR. However, it is difficult to predict accurate edge maps from low resolution (L-R) depth maps. In this paper, we propose a deep edge map guided depth SR method, which includes an edge prediction subnetwork and an SR subnetwork. The edge prediction subnetwork takes advantage of the hierarchical representation of color and depth images to produce accurate edge maps, which promote the performance of SR subnetwork. The SR subnetwork is a disentangling cascaded network to progressively upsample SR result, where every level is made up of a weight sharing module and an adaptive module. The weight sharing module extracts the general features in different levels, while the adaptive module transfers the general features to the specific features to adapt to different degraded inputs. Quantitative and qualitative evaluations on various datasets with different magnification factors demonstrate the effectiveness and promising performance of the proposed method. In addition, we construct a benchmark dataset captured by Kinect-v2 to facilitate research on real-world depth map SR.

Keywords: super resolution, depth map, edge prediction, disentangling

*Corresponding author
Email addresses: jiang_zhongyu@yeah.net (Zhongyu Jiang), dayueer@tju.edu.cn (Huanjing Yue), Yukun.Lai@cs.cardiff.ac.uk (Yu-Kun Lai), yjy@tju.edu.cn (Jingyu Yang), houroy@tju.edu.cn (Yonghong Hou), hcp@tju.edu.cn (Chunping Hou)

1. Introduction

Depth images/videos are widely used in modern applications, such as 3DTV[1, 2], recognition[3, 4], action analysis[5] and automotive driver assistance. It is a typical method to use existing depth sensors (often based on Time-of-Flight (ToF) or structured light) to obtain depth maps. However, depth images captured by depth sensors have many degradations, which limit their applications. Therefore, advanced methods are urgently needed to improve the quality of depth images, especially for improving the spatial resolution.

Color-guided depth image SR methods become prevailing due to two reasons. First, existing depth sensors are usually accompanied by color sensors, which enable capturing the color and depth image pairs (RGB-D pairs) simultaneously. Second, the RGB-D pairs usually have consistent structures in the edge regions since they are different descriptions of the same scene. Despite promising results [6, 7], this type of methods tend to bring texture copy and blurring artifacts [8, 9] due to the neglect of inconsistent areas between RGB-D pairs.

To tackle this problem, there are many attempts [10, 11] to mitigate negative effects of the color image, such as designing elaborate weighting factors [12, 13, 14], adopting joint guidance [15, 16], learning complementary information of RGB-D pairs [17, 18], and explicit inconsistency measurement[8, 10]. Among these attempts, a simple and intuitive solution to avoid texture copy artifacts is utilizing mutual edges in an RGB-D pair as guidance [19, 20]. Accurate reconstruction of edges is critical to image SR [21] but LR depth map alone is not enough to produce an accurate edge map. It will be tougher for a real-world LR depth map since it has structure missing along edges, as shown in Fig. 1. Therefore, we propose to learn accurate edge maps with the guidance of color images to help super-resolve depth maps, which can avoid introducing the texture details of the color images to the depth maps.

How to synthesize satisfactory edge maps with RGB-D pairs is another problem. The extracted edges tend to be inaccurate or discontinuous when adopting simple edge detection methods [22, 23, 24] or when the external dataset is inad-

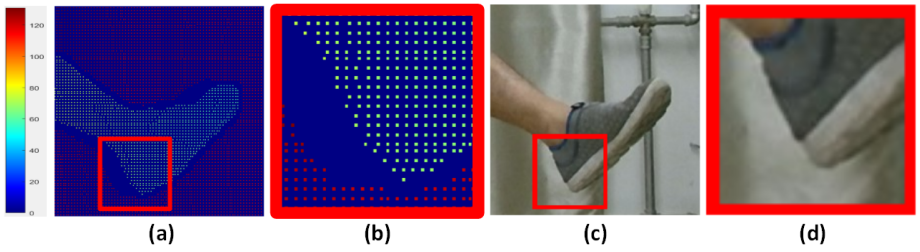

Figure 1: Visualization of the depth missing artifacts in the real captured depth map. (a) is a depth map captured by Kinect v2 which is warped to the color view. (b) is the zoomed patch in (a), where the blue pixels do not have valuable depth values. (c) is the corresponding color image. (d) is the zoomed patch in (c).

equate [22, 20, 25] for sparse representation based and example based methods. Moreover, it is difficult to design handcrafted rules [19, 25, 26] to generate accurate HR edge maps, since the LR depth edges are smooth meanwhile the HR color edges are sharp but are not consistent with the depth maps in texture areas. Therefore, we propose to utilize the deep neural networks to fuse the complementary information of the LR depth edges and the HR color edges.

Effective edge guidance alone is not enough for the depth SR task. What kind of SR subnetwork architecture to choose is also important. We adopt a progressive strategy in this paper, which has been proved to be effective in image SR [27, 13]. Despite the same network architecture for each cascade level, previous works generally assign different network parameters for different levels. Different from them, we propose to share most of parameters among all levels to reduce the number of parameters of the original cascaded network. Meanwhile, we further propose an adaptive module to increase the capacity in dealing with inputs with different down-sampling ratios.

Furthermore, we observe that the depth image SR performance will be heavily reduced if directly applied to real-world captured depth maps. The main reason is the lack of real-world dataset. This motivates us to construct a real-world degraded RGB-D dataset to facilitate more research on real-world depth map SR.

The main contributions of this study are summarized as follows.

- We propose to predict the depth edges via fusing the deep features extracted from two kinds of images , i.e. the color and depth maps, in different scales. Instead of directly concatenating the depth maps and color images together to predict edge maps, we propose to predict their multi-scale features separately since they have different properties and resolutions. Then the multi-scale based fusion strategy enables us to reconstruct sharp and accurate edge maps.

- We propose a disentangling cascaded SR network, which consists of a weight sharing module and an adaptive module in each level. The weight sharing module extracts the general features in different levels to reduce the number of parameters, while the adaptive module transfers the general features to the specific features to adapt to different degraded inputs.

- We construct a benchmark dataset [1] of 75 RGB-D images captured by Kinect-v2 for real-world scenes. We first warp the LR depth images to the view and size of the color images. Then we utilize the method in [13] to reconstruct the warped depth images and then refine them via manual adjustment to synthesize pseudo ground truth (GT). In summary, our dataset contains LR depth images, their corresponding high resolution (HR) color images, warped LR depth images, and the pseudo GT. The constructed dataset will facilitate more research on the enhancement of LR depth images.

- Extensive experiments on various datasets with different measurements demonstrate that our method has better performance than state-of-the-art SR methods.

The rest of this paper is organized as follows. Sec. 2 presents related work. We present our SR method by introducing the edge prediction subnetwork and

[1] We will release this dataset after the paper acceptance.

SR subnetwork in Sec. 3. Sec. 4 evaluates the proposed model with experiments on both synthetic and real-world data followed by conclusion in Sec. 5.

2. Related Work

Depth image SR methods can be divided into two categories: traditional depth image SR and deep depth image SR. Traditional SR methods are more flexible, while deep SR methods are good at learning the complex mapping functions from a large scale dataset. Real-world depth image SR is a challenging problem but only very limited works have related research, which needs more attention. We will describe above categories in detail below.

2.1. Traditional Depth Image Super Resolution

Traditional depth SR methods can be classified into three sub-categories: learning-based methods, filtering-based methods, and regularization-based methods.

The key of learning-based methods [6] is to learn a sparse representation of the depth image by carefully designing dictionaries. The dictionaries are generally learned from a external dataset [28, 22]. Among them, Ferstl *et al.* [22] estimated edge maps with a learned dictionary, which are used in variational depth SR as an anisotropic guidance. Since global dictionaries cannot adapt to local characteristics of depth signals well, the works in [25, 24] constructed local sub-dictionaries, and edge-aware constraints were used to preserve significant edges in the depth map.

Filtering-based methods enhance depth maps via local filters, which usually depend on the guidance images. The benchmark work is the joint bilateral filter in [29], which calculates the filter parameters in depth SR using the RGB-D pairs. Lu *et al.* [30] further improved the performance via introducing shape-adaptive local support representation and integration technique. Since joint filtering usually introduces texture copy artifacts, more works are proposed to remedy the artifacts. The works in [16, 15] utilized recovered depth maps to

dynamically correct SR results. The works in [20, 19] utilized edge maps to solve this problem more effectively. However, the edge quality in work [20] went down dramatically when there were not enough training examples in the external dataset. Edges in [19] were not very accurate, since they extracted edges from the bicubic version of LR depth map and precision of edges only reached 2×2 pixels.

The regularization-based methods utilize regularization terms to make the ill-posed depth SR problem well constrained. Common regularization includes nonlocal regularization [31, 32], smoothness regularization [12, 33], total variation (TV) regularization [34, 13] and graph Laplacian regularization [35]. These regularizers greatly improve the depth SR performance. To handle texture-copy artifacts, works in [10, 11, 8] embedded the inconsistency between the depth and color images into the weights of regularization. The works in [33, 36] utilized a nonconvex penalty function to improve robustness of the smoothness regularization, which reduced the texture copy artifacts. Liu *et al.* [23] proposed a gradient consistency regularizer to remedy the structure discrepancy problem, which can be viewed as a special form of edge image.

2.2. Deep Depth Image Super Resolution

The applications of convolutional neural networks (CNNs) greatly improve depth SR performance benefiting from advanced network architectures [37, 38], effective loss functions [39, 40] and massive data. In CNN-based methods, color image is a useful extra information to increase the reconstruction accuracy [41]. However, the fusion method of RGB-D pairs need more attention to avoid texture-copy artifacts. To selectively transfer only consistent information of RGB-D pairs, the works in [17, 18] utilized halfway concatenation to fuse RGB-D features to super-solve depth maps. In contrast, post-fusion layers and an advanced training strategy were adopted to fuse RGB-D features in the multi-scale cascade network [42]. Ye *et al.* [9] utilized a separate color branch as prior knowledge to promote depth SR, which remedied texture copy artifacts by introducing color information only in earlier blocks. The above methods only

rely on the location change of fusion to reduce texture copy artifacts, which are difficult for CNNs to distinguish between edges and textures well [43]. Further constraints are needed to reduce texture copy artifacts.

To address texture-copy artifacts, Zuo *et al.* [44] proposed a local affine transformation to filter out the unrelated intensity features explicitly by Hadamard product operations. Deng *et al.* [45] split the common information from different modalities by designing extraction modules for unique feature and common feature respectively. In this paper, we propose to explicitly predict depth edge maps from RGB-D pairs via CNNs to avoid texture copy artifacts.

2.3. Real World Depth Image Super Resolution

How to tackle real-world degradation is important but only has limited works. Ferstl *et al.* [34] released three scene images captured by ToF and intensity cameras simultaneously, which enabled traditional depth SR methods to use real-world data for the first time. The works in [31, 13] hereafter evaluated their methods on several real-world depth maps captured by Kinect camera. To promote more CNN-based methods to improve performance of real-world depth image SR, Song *et al.* [46] improved the generation methods of synthetic LR depth maps to simulate real-world degradation. However, the gap between their new generation method and real-world degradation process needs to be explored. Moreover, they only evaluated their method on synthetic dataset. Gu *et al.* [47] proposed a domain transfer method between synthetic depth maps and real-world depth maps to improve the real-world depth map SR performance. This work makes a step forward in the processing of real-world degradation depth images using CNN-based methods. However, there is still no real dataset captured by kinect-v2 camera, which is the most popular depth sensor. To promote more research for CNN-based methods, we therefore construct a real-world dataset captured by Kinect-v2 camera in this paper.

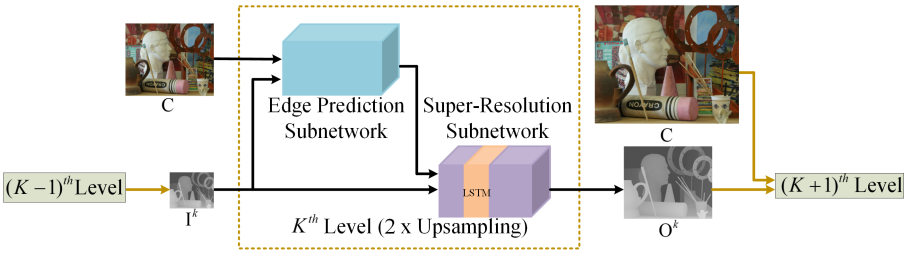

Figure 2: Flowchart of the proposed cascaded network. For $2^m\times$ upsampling, there are m levels, where each level implements a $2\times$ upsampling. There are an edge prediction subnetwork and a SR subnetwork in each level.

3. The Proposed Depth Map Super-Resolution Method

As shown in Fig. 2, out network can be unfolded to K levels for $2^K\times$ upsampling. Each level contains two subnetworks: edge prediction subnetwork (EPN) and super-resolution subnetwork (SRN).

For the k-th level, denote the input LR depth map, corresponding HR color image and output depth map as $\mathbf{I}^k$, $\mathbf{C}$ and $\mathbf{O}^k$ respectively. Then the operation of the k-th level can be presented by:

$$\mathbf{O}^k = \mathcal{F}_{SRN}(\mathbf{I}^k, \mathcal{F}_{EPN}(\mathbf{I}^k, \mathbf{C})), \tag{1}$$

where $\mathcal{F}_{EPN}(\cdot)$ and $\mathcal{F}_{SRN}(\cdot)$ denotes the *Edge Prediction Subnetwork* and *Super-Resolution Subnetwork* respectively. The input $\mathbf{I}^0$ of the first level is the original LR depth map $\mathbf{D}^L$.

3.1. Edge Prediction Subnetwork

We propose to predict depth edges via fusing the deep features in different scales extracted from the color and depth maps. The network structure is presented in Fig. 3. The LR depth map $\mathbf{I}^k$ and HR color map $\mathbf{C}$ go through the multi-scale based feature extraction block independently, which produces fine to coarse scale features. Then we utilize the feature fusion block to fuse the two kinds of features together to generate the edge maps in different scales. Take the first scale in the feature fusion block as an example, we first utilize 1x1

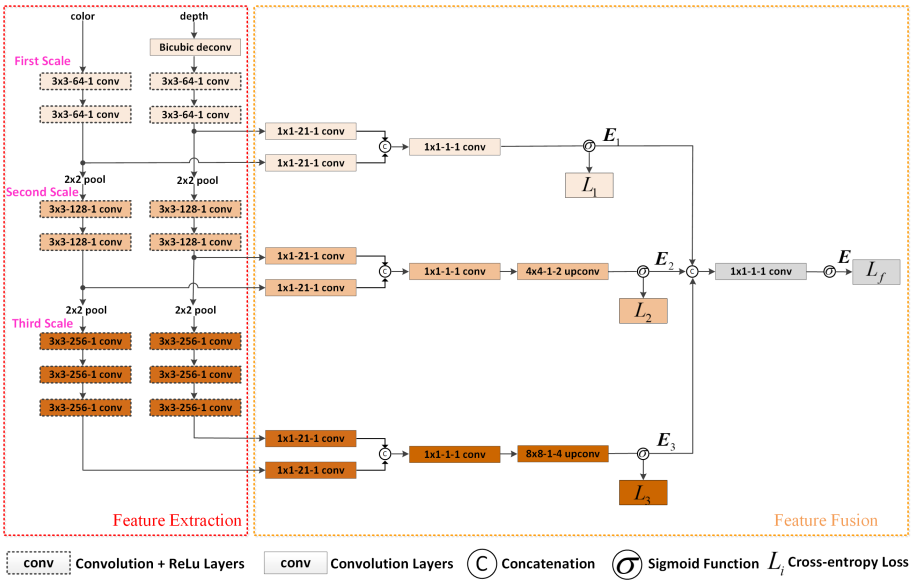

Figure 3: The proposed edge prediction subnetwork. The input is the HR color image and LR depth map, and the output is the edge map $\mathbf{E}$. For $k \times k - c - s$ listed in the rectangle box, k is the kernel size , c is the channel number after the corresponding conv (upconv) operations, and s is the stride.

convolution to shrink the channel numbers of extracted features from depth and color images respectively. Hereafter, we concatenate the two kinds of features together, which further goes through another 1x1 convolution to generate the edge map $\mathbf{E}_1$ in the first scale. The second and third scale feature fusion process are similar to that in the first scale except that we utilize the upconvolution block to make the predicted edge maps ($\mathbf{E}_2$ and $\mathbf{E}_3$) have the same size as $\mathbf{E}_1$. After generating $\mathbf{E}_1$, $\mathbf{E}_2$, and $\mathbf{E}_3$, we further concatenate them together to generate the final edge map $\mathbf{E}$. In this way, we can take advantage of the complementary information from $\mathbf{E}_1$, $\mathbf{E}_2$, and $\mathbf{E}_3$ to generate $\mathbf{E}$.

The edge prediction task can be formulated as a classification problem [48, 49], i.e. whether the pixel is on an edge or not. Therefore, we utilize cross-entropy loss [49] to train edge maps at three scales and the final fused edge

map:

$$\mathcal{L}_{edge} = \sum\nolimits_t \alpha_t \mathcal{L}_t + \mathcal{L}_f, \tag{2}$$

where $\mathcal{L}_t$ represents the side-output loss in scale $t \in [1, 3]$, and $\mathcal{L}_f$ represents the loss in fusion stage, α_t is the weighting parameters (we set $\alpha_t = 1$ here).

$\mathcal{L}_t$ and $\mathcal{L}_f$ aim to make the predicted edge (non-edge) distribution in the predicted edge map become close to the real edge (non-edge) distribution in the label edge map, and can be formulated as

$$\begin{aligned} \mathcal{L} &= -\sum\nolimits_j [\beta \log(P_j > 0.5) + (1-\beta)\log(P_j \leq 0.5)] \\ &= -\sum\nolimits_{j \in E_+} \beta \log(P_j > 0.5) - \sum\nolimits_{j \in E_-} (1-\beta)\log(P_j \leq 0.5), \end{aligned} \tag{3}$$

where $P_j \in [0, 1]$ represents the predicted probability of pixel j belongs to an edge, which is computed using sigmoid function. The non-edge pixels whose value is 0 in the edge label map belong to the set E_-, and edge pixels whose value is 1 in the edge label map belong to the set E_+. $|E_+|$ and $|E_-|$ are the pixel numbers of set E_+ and E_-, $\beta = |E_+|/(|E_+| + |E_-|)$. The label maps are the binarization results of the edges generated by method [49] with GT depth maps as inputs.

We take the dot product of the binarization of the learned edge map and the color image as the final edge map guidance, where the color image can help accurately localize the edge positions, since the learned edge map often includes thick edges. Then the edge guidance enters into the SR subnetwork to help super resolve LR depth images.

Fig. 4 presents the estimated edge maps in different scales. From the first to the third scale, edge maps become coarser, emphasizing more significant edges along depth discontinuous while ignoring fine-grained details in texture regions. These multi-scale based hierarchical representation is a critical strategy for edge prediction [48, 49].

We would like to point out that compared with directly predicting edge map from an LR depth map, the proposed method can generate much better results (as shown in Table 1). The main reason is that the LR depth maps

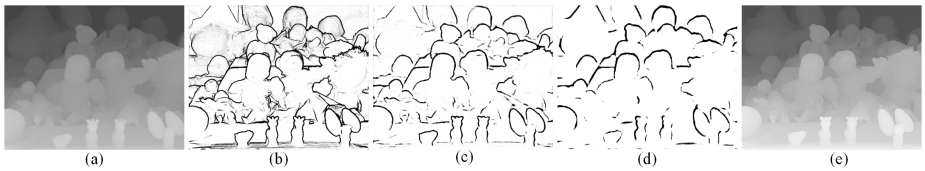

Figure 4: Edge maps predicted from fine to coarse scales. From left to right: (a) the LR depth, the edges extracted from the (b) first, (c) second, and (d) third scale respectively, and (e) the ground truth depth map. Please zoom in the figure for better observation.

are very smooth and real-world captured depth maps usually have structure missing along edges (as shown in Fig. 1), which makes it tougher to estimate accurate edge maps from LR depth maps alone. In addition, compared with directly generating edge map from the concatenating of color and depth images, the proposed method can generate better results (as shown in Table 1) since the depth and color images have different properties. Two separate branches in the feature extraction block can flexibly learn their key features contributing to the edge maps. Therefore, compared with work in [49], our two separate feature extraction branches for LR depth maps and HR color images respectively are elaborately designed for depth map SR. Our edge prediction subnetwork is also much simpler than that in [49] but works well in predicting HR depth edges.

3.2. Super-Resolution Subnetwork

To effectively handle different magnification factors, we propose a cascaded SR subnetwork, where each cascaded level implements a $2\times$ upsampling, as shown in Fig. 2. Different from previous cascaded SR networks [42, 27], we propose a disentangling strategy, namely that the SR network in each level contains a weight sharing module and a weight adaptive module. The weight sharing module, which shares weights among all levels, aims at processing general features in different levels, and the weight adaptive module is designed to process specific features in each level. Compared with changing all the parameters in each level, the proposed disentangling strategy greatly reduces the number of parameters. The adaptive module transfers the general features to

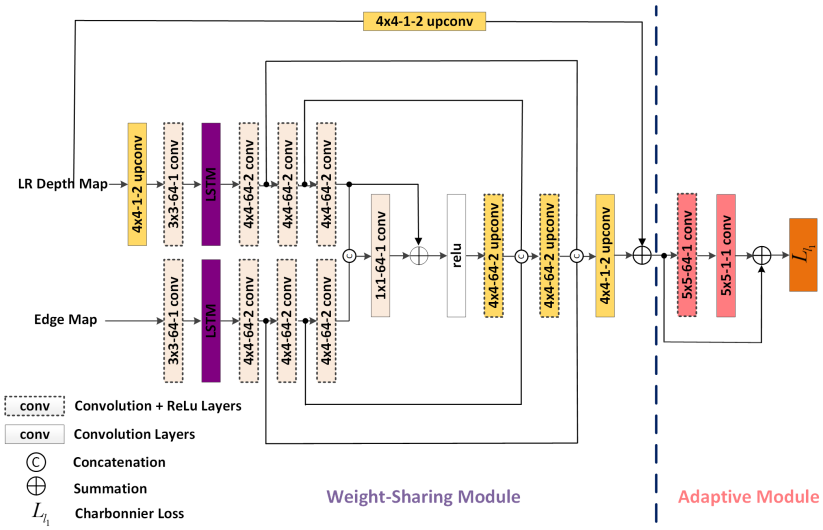

Figure 5: Network architecture at one level of the proposed SR subnetwork, which consists of a weight sharing module and an adaptive module. For $k \times k - c - s$ in each box, k represents the kernel size, c is the channel number generated by the corresponding conv/upconv operation, and s is the stride.

the specific features to adapt to different degraded inputs in order to achieve satisfied performance for different magnification ratios.

Fig. 5 presents the SR network architecture at one level. The weight sharing module is a UNet [50] with skip connections. It has two branches dealing with the predicted edge map and depth map respectively. We further introduce Convolutional Long Short-Term Memory (LSTM) layers [51] after the first convolution layer to fully capture the inter-level dependencies. The output features of the weight sharing module are the input of weight sharing module in the next level, and the weight adaptive module of the current level. The adaptive module is a residual unit consisting of two convolution layers and a ReLu activation layer.

Let $\mathbf{O}^k$ denote the recovered depth map at the k-th level. We use the Charbonnier loss function [27] to train the SR subnetwork to better handle outliers:

$$\mathcal{L}_{l1} = \frac{1}{N} \sum_k \sum_j \Re(\mathbf{D}_j^k - \mathbf{O}_j^k), \tag{4}$$

where $\Re(y) = \sqrt{y^2 + \varepsilon_c^2}$. ε_c is a small constant to make $\Re(y)$ differentiable. N is the number of training samples in each batch, $\mathbf{D}^k$ is the GT depth map at the k-th level, $\mathbf{O}_j^k$ is the depth value at position j in depth map $\mathbf{O}^k$.

We would like to point out that, although the work in [52] also has adaptive module, our work is different from [52] in three aspects. First, the adaptive module in [52] is designed to make the model generalized to unseen data while our adaptive module is designed to make the module adapt to different degraded inputs while reducing the number of parameters. Second, the work in [52] introduces an adaptive layer after each convolution layer, namely layer-level adaptation. In contrast, our work is network-level adaption which introduces an adaptive network after the whole weight sharing network. Third, the work in [52] only fine-tunes those adaptive layers when dealing with a new degradation type, while we update the whole SR network for all inputs with different down-sampling ratios.

4. Experimental Results and Analysis

In this section, we evaluate our method on both synthetic and real-world data. We compare the proposed method with nine state-of-the-art super-resolution methods, including filtering-based methods, i.e., guided filtering (GF) [53], static and dynamic guided filtering (SDF) [15], regularization-based methods, i.e., color-guided autoregressive (AR) [31], edge-guided method (EG) [20], joint local structural and nonlocal low-rank regularization (LN) [12], and deep image SR methods, i.e. multi-scale guidance network (MSG) [42], deep edge guided recurrent network (DEGR)[54], Laplacian pyramid SR network (LapSRN) [27] and pixel to pixel transformation network (GP2P) [43]. GF, AR, LN, MSG and GP2P directly utilize color images as guidance, while SDF utilizes both color image and depth image as guidances. EG and DEGR utilize edge guidance as our method. LapSRN and MSG have the pyramid strategy as ours.

All results are generated by the authors' codes and the same LR input. For comparison methods, we utilize parameters in their papers or make necessary

modifications. Specifically, we change the patch sizes of DEGR and LapSRN to make them the same as ours. We reduce the batchsize of LapSRN at large upsampling ratios due to memory limitation. We reduce the edge loss weight of DEGR to get better results. All deep image SR methods are retrained using our dataset, except the MSG which only released test code.

We use the same training and validation datasets with MSG for synthetic experiments. There are 82 images in our training set and 10 images in our validation set, which consist of 58 HR RGB-D pairs from MPI Sintel depth dataset [55] and 34 HR RGB-D pairs from Middlebury dataset [56]. We test all methods on three synthetic datasets : Middlebury dataset, LFD dataset [57] and ICL dataset [58]. For Middlebury dataset, we use eight images as the test set. For LFD dataset, we use all additional images (16 images) as the test set since the GT depth maps are provided for these images. ICL dataset has vidoes for two scenes. We randomly select six frames respectively for scene 'living room' and scene 'office room'. We have checked our training set to make sure that there is no overlap between training images and testing images. For real experiment, we retrain all compared methods on our constructed real-world dataset.

4.1. Parameter Setting

The training and validation images are cropped into small square sub-images with size $= \{64, 128, 256\}$ for the upsampling ratio $= \{2, 4, 8\}$. We use flipping (up-down and left-right) and clockwise rotations ($90°, 180°$ and $270°$) for data augmentation. The training and testing RGB-D images are normalized to the range [0, 1.0]. We train our model on the Caffe platform [59]. Adam optimizer is adopted with momentum $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We utilize the model parameters of [49] to initialize our edge prediction network. The learning rate for the first two scales and the third scale in the edge prediction network are set to 3e-7 and 3e-6 respectively. Since the convolution parameters of the third scale need more changes to capture more significant edges along depth discontinues rather than fine-detailed details as that done in [49], we set a higher learning

rate to the third scale. We train the SR subnetwok from scratch with initial learning rate 1e-3. The learning rates are dropped by half when the validation losses are no longer reduced.

4.2. Ablation

In this subsection, we test the effectiveness of each part of our method on Middlebury dataset [56] at 4× upsampling in Table 1, which are evaluated by RMSE (root mean square error) metric. The weight sharing module here does not include the LSTM layer. The method b is better than method a which demonstrates the effectiveness of LSTM layers to capture dependency among levels. Compared with method a, the superiority of method c shows that adaptive modules can improve the adaption to specific inputs. The method d is the proposed method, which achieves the best result. In contrast, the absence of edge guidance leads to the obvious drop of performance as shown in the result of method e. The performance is heavily degraded when directly taking the concatenation of RGB-D pairs as the input of edge prediction subnetwork as shown in method f, since RGB-D pairs are different kinds of images and this concatenation method can bring texture copy problem in predicted edge maps. The method g is only slightly worse than the method d, but the gap becomes obvious for large SR ratios. For example, the average RMSE result of method g at 8× upsampling is 1.83 while the result of the proposed method at 8× upsampling is 1.47. This indicates that it is more difficult to predict edge maps only from LR depth maps with the increasing of sampling ratios.

Table 1: Ablation study on Middlebury dataset at 4× upsampling evaluated by RMSE values.

Methods	Weight Sharing Module	LSTM	Adaptive Module	Edge Maps as Guidance	Edge Predicting Methods	Average RMSE
a	✓	×	×	✓	the proposed method	1.37
b	✓	✓	×	✓	the proposed method	1.21
c	✓	×	✓	✓	the proposed method	1.17
d	✓	✓	✓	✓	the proposed method	1.01
e	✓	✓	✓	×	no edge subnetwork	1.35
f	✓	✓	✓	✓	directly concatenating RGB-D pairs	1.35
g	✓	✓	✓	✓	only using LR depth map	1.08

Table 2: Quantitative comparison results on three datasets in terms of RMSE, IFC, and SSIM measurements. The best results are highlighted and the second best results are underlined.

Methods	Scale	Middlebury [56]	LFD [57]	ICL [58]
GF [53]	2×	1.81/2.56/0.99909	4.49/4.31/0.99307	2.65/10.63/0.99747
AR [31]	2×	1.74/1.87/0.99846	5.21/2.53/0.99387	3.01/2.02/0.99632
SDF [15]	2×	2.23/2.15/0.99778	5.41/2.91/0.99054	3.94/2.40/0.98731
EG [20]	2×	1.83/2.02/0.99851	5.17/2.14/0.99166	3.02/2.12/0.99671
LN [12]	2×	2.06/1.38/0.99711	5.18/1.96/0.99011	3.34/1.75/0.99362
MSG [42]	2×	0.51/6.46/0.99994	1.62/14.24/0.99962	0.81/16.08/0.99988
DEGR [54]	2×	2.02/4.26/0.99957	5.28/11.57/0.9957	3.22/13.02/0.99868
LapSRN [27]	2×	1.35/4.99/0.99983	3.85/11.35/0.99902	2.11/9.79/0.99964
GP2P [43]	2×	3.16/0.58/0.98636	6.43/1.27/0.97515	6.86/1.25/0.98465
Ours	2×	**0.46/8.82/0.99995**	**1.28/19.55/0.99980**	**0.68/18.86/0.99992**
GF [53]	4×	2.36/1.78/0.99686	6.27/3.24/0.97706	3.55/3.85/0.99269
AR [31]	4×	2.64/0.82/0.99364	7.66/1.20/0.96784	4.98/1.04/0.98439
SDF [15]	4×	3.26/0.85/0.98960	8.62/1.34/0.95505	6.19/1.38/0.97552
EG [20]	4×	2.49/1.21/0.99712	7.17/1.44/0.97376	4.22/1.28/0.98967
LN [12]	4×	3.02/0.82/0.99114	8.20/1.02/0.95991	4.87/0.96/0.98305
MSG [42]	4×	1.04/2.83/0.99957	3.92/4.81/0.99117	1.85/5.21/0.99809
DEGR [54]	4×	2.92/1.78/0.99613	8.22/4.20/0.96993	4.67/3.26/0.99048
LapSRN [27]	4×	1.26/2.38/0.99954	4.50/2.92/0.99218	2.32/2.48/0.99795
GP2P [43]	4×	3.20/0.56/0.98509	7.96/1.21/0.96978	10.37/1.17/0.97564
Ours	4×	**1.01/3.16/0.99965**	**3.43/7.49/0.99370**	**1.57/5.84/0.99881**
GF [53]	8×	3.39/0.92/0.98496	9.15/1.83/0.92750	5.31/2.13/0.97608
AR [31]	8×	3.85/0.51/0.97998	10.69/0.75/0.92564	7.52/0.75/0.96549
SDF [15]	8×	4.90/0.45/0.96783	12.63/0.79/0.90288	9.10/0.96/0.95445
EG [20]	8×	4.39/0.39/0.97802	11.71/0.59/0.89834	7.98/0.56/0.94900
LN [12]	8×	4.09/0.46/0.97786	11.48/0.69/0.91927	7.40/0.67/0.96389
MSG [42]	8×	1.76/1.23/0.99732	6.44/1.66/0.96718	3.01/2.24/0.99203
DEGR [54]	8×	3.97/0.55/0.97987	10.71/1.21/0.91533	6.35/1.37/0.96927
LapSRN [27]	8×	1.76/1.35/0.99760	6.60/2.23/0.96789	3.24/2.29/0.99273
GP2P [43]	8×	3.30/0.53/0.98368	12.09/0.98/0.94309	11.12/1.08/0.97345
Ours	8×	**1.47/1.62/0.99842**	**4.93/3.55/0.97756**	**2.34/2.79/0.99506**

4.3. Experiments on Synthetic Data

Table 2 presents the average results on three datasets in terms of RMSE, IFC (Information Fidelity Criterion) and SSIM (Structural Similarity Index) values at 2×, 4× and 8× upsampling (More detailed results are available in supplementary material.). IFC is claimed to have the highest correlation with perceptual scores for SR evaluation [60]. The best results are highlighted in bold, and the second results are underlined. According to work in [27], we set network depth $d = 10$ at each level for 2× and 4× models of LapSRN [27], and set $d = 5$ at each level for the 8× model of LapSRN . Table 2 shows that our method achieves the best results for all datasets at different magnification ratios. Our method, MSG and LapSRN outperform most other methods, which demonstrates the effectiveness of progressive strategy. In addition, our method constantly outperforms LapSRN and GP2P, which indicates that good edge maps can bring more gain than color images. In contrast, results of DEGR[54] and EG [20] are not satisfactory, although they also use edge maps. This is because accuracy of edge maps of EG is highly dependent on the external dataset, which degrades a lot at 8× upsampling due to the absence of the external dataset. DEGR adopts a handcrafted edge detector and cannot produce accurate depth edge maps, although the detector is effective in natural images. In a word, although edge map guided SR is a common idea, it is not easy to produce high quality edge maps in favor of the SR process.

To further demonstrate the effectiveness of the proposed method, we show the visual results at 8× upsampling on three datasets (from Fig. 6 to Fig. 8). Results of GF [53] have obvious halo artifacts. AR [31] and LN [12] produce a bit smooth edges, because common traditional regularization cannot handle severe degradation well at large magnification ratios. LN, SDF [15] and GP2P [43] have texture copy artifacts (such as the second scene in Fig. 6), since they lack explicit inconsistency handling. In addition, LN, SDF and AR also have scattering artifacts. EG [20] and DEGR [54] don't produce pleasing results, since they cannot produce good edge maps. Benefiting from large scale datasets, LapSRN achieves visually pleasing results. However, its results are slightly smooth, since

they do not utilize high quality guidance information. In contrast, MSG and our method have sharp edges and no artifacts. Our method has better structure details than MSG due to high-quality edge maps.

4.4. Experiments on Real-World Data

We capture 75 depth maps (512×424) and their corresponding color images (1920×1080) using Kinect-v2 to construct a real-world RGB-D dataset. The depth images have view disparity with the HR color images. Therefore, we warp the depth images to the view of the color images using the camera parameters. As shown the first row in Fig. 9, the warped depth image has missing structures along edges and random missing content in the whole image. Since there is no GT for the warped depth maps, we first reconstruct the warped depth maps using the method in [13] and then refine these depth maps via manual adjustment to synthesize pseudo GT. These warped depth images, color images and the pseudo GT depth images make up our dataset. We would like to point out that our dataset is different from NYU v2 [61] in two aspects. First, the images in NYU v2 are captured by Kinect-v1. Second, the inpainted HR depth images in NYU v2 have jagged artifacts, while our pseudo GT depth images are sharp and clean. To our knowledge, there is no dataset captured by Kinect-v2 with HR depth images for real-world depth SR research. Our dataset is the first to deal with the structural and random missing of real-world data.

Fig. 10 shows two recovered scenes. We can see that GF[53] and LN[12] produce overly smooth results. GF[53] has obvious ringing artifact and scattering artifact. Besides, GF[53], AR[31], and SDF[15] cannot preserve details well such as the arm of the second scene. DEGR [54] cannot recovery random depth missing well, since it directly extracts edge maps from LR depth images with depth missing. LapSRN[27] has jagged artifacts in the second scene. In contrast, our results have sharper edges and better details, which benefits from edge prediction subnetwork. However, we would like to point out that, the pro-

[2]the mean absolute difference between the ground truth and the corresponding result.

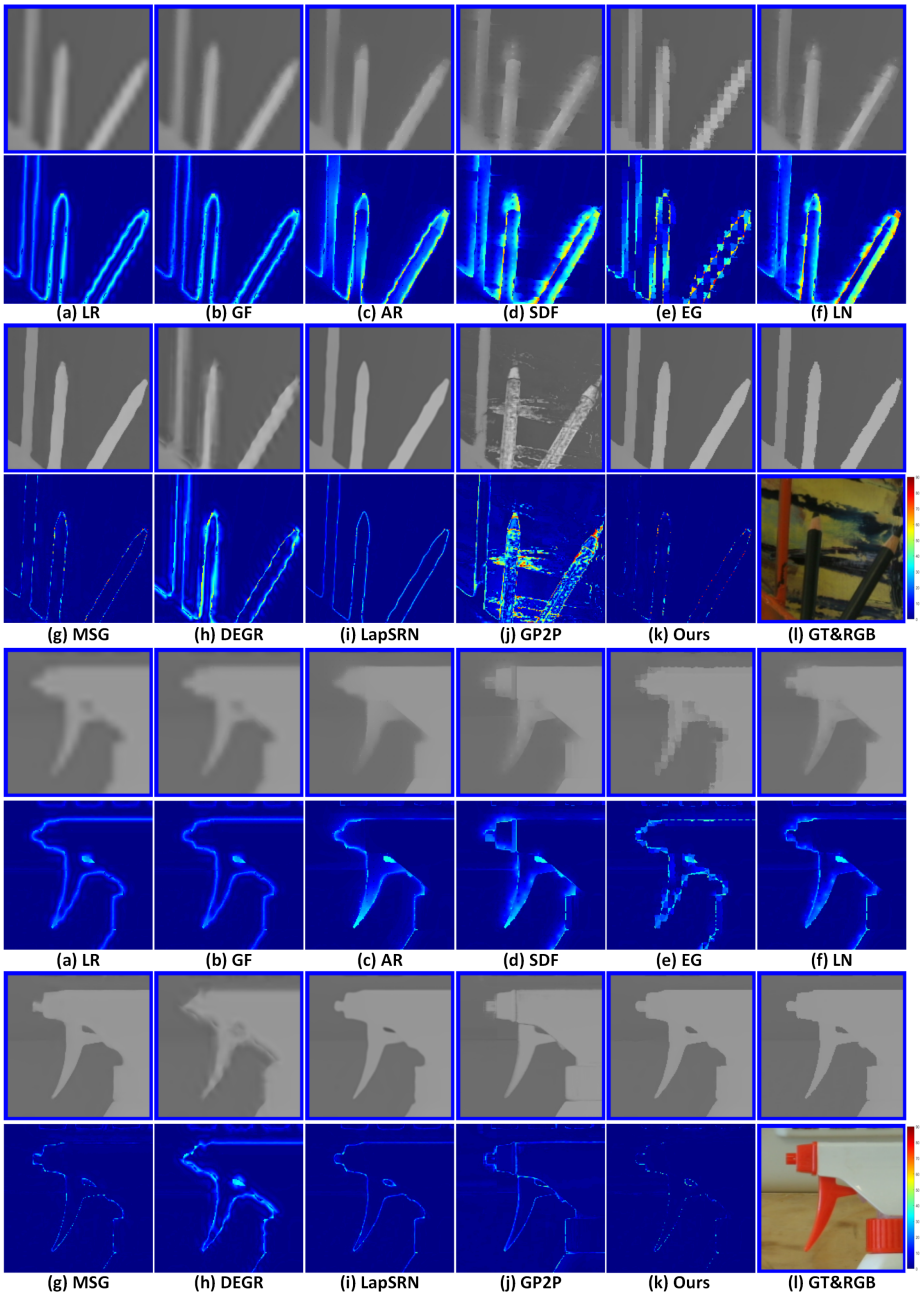

Figure 6: Visual comparison results at 8× upsampling for image "Art" and "Laundry" from Middlebury dataset. The cropped patches are generated by (a) the LR, (b) GF [53], (c) AR [31], (d) SDF [15], (e) EG [20], (f) LN [12], (g) MSG [42], (h) DEGR [54], (i) LapSRN [27], (j) GP2P [43], (k) the proposed method and (l) the ground truth and corresponding color image. The second row is the error map[2].

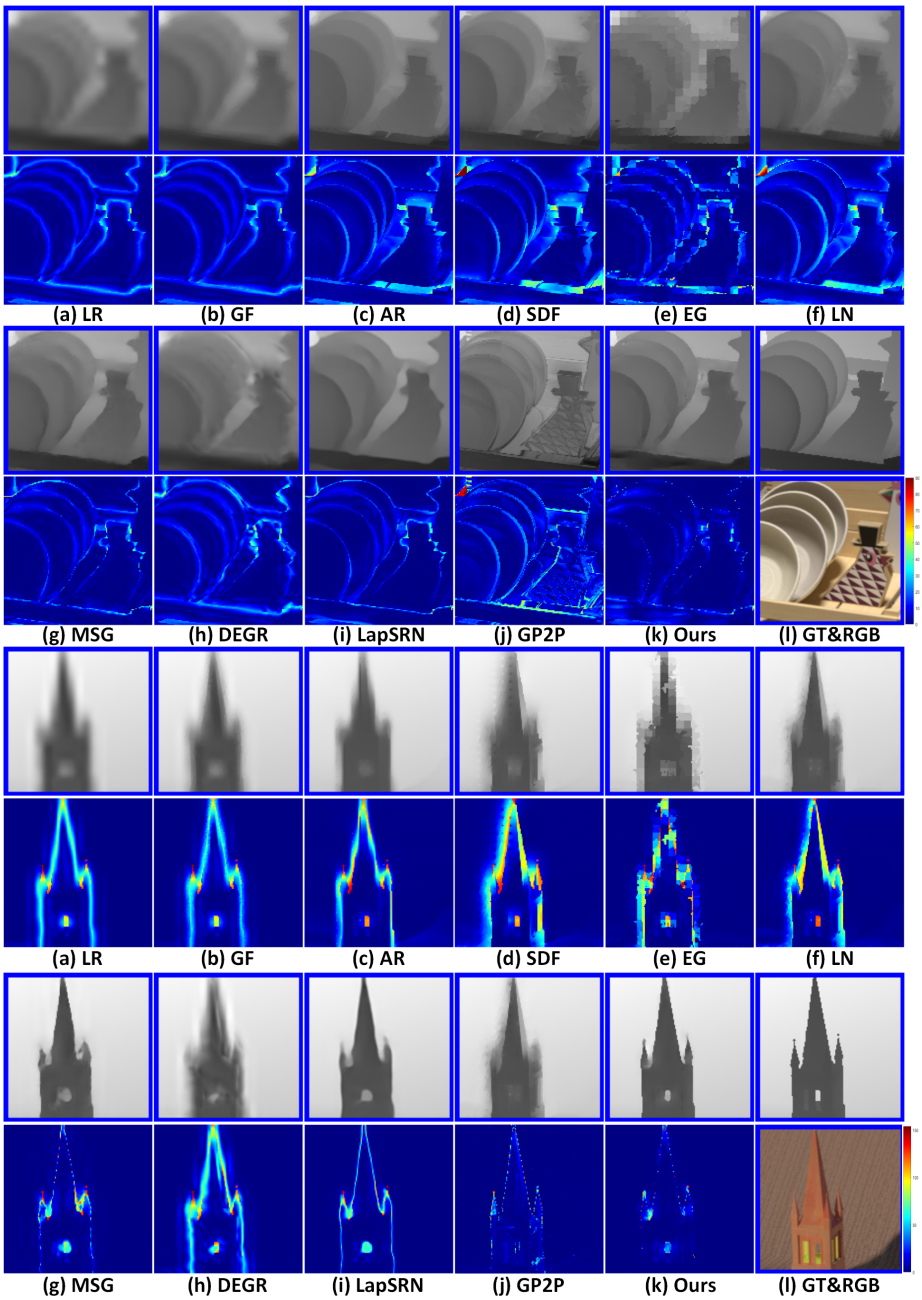

Figure 7: Visual comparison results at 8× upsampling for image "Dishes" and "Tower" from LFD dataset. The cropped patches are generated by (a) the LR, (b) GF [53], (c) AR [31], (d) SDF [15], (e) EG [20], (f) LN [12], (g) MSG [42], (h) DEGR [54], (i) LapSRN [27], (j) GP2P [43], (k) the proposed method and (l) the ground truth and corresponding color image. The second row is the error map.

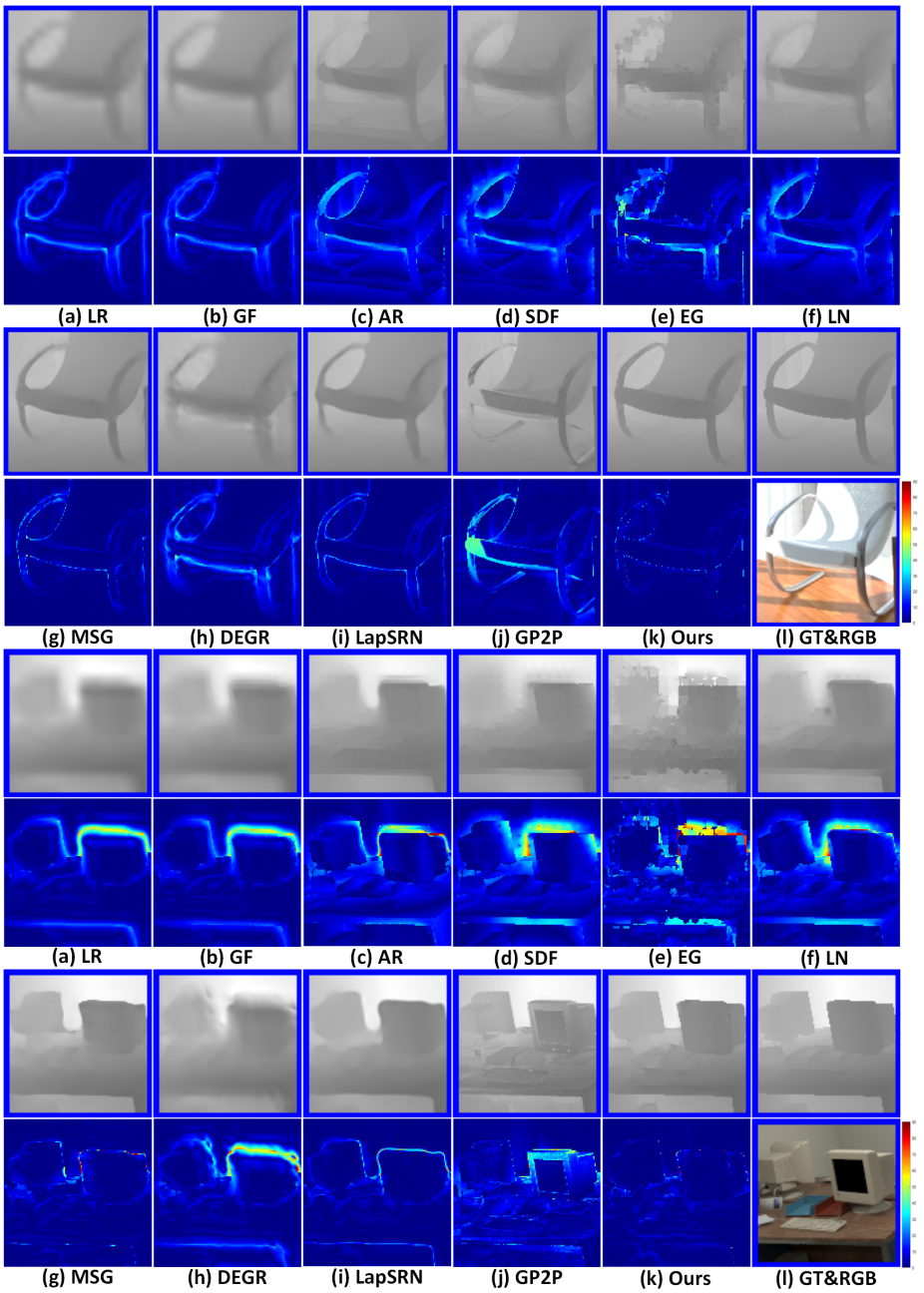

Figure 8: Visual comparison results at 8× upsampling for image "Scene 4" and "Scene 11" from LCL dataset. The cropped patches are generated by (a) the LR, (b) GF [53], (c) AR [31], (d) SDF [15], (e) EG [20], (f) LN [12], (g) MSG [42], (h) DEGR [54], (i) LapSRN [27], (j) GP2P [43], (k) the proposed method and (l) the ground truth and corresponding color image. The second row is the error map.

Figure 9: Some examples in our constructed real-world RGB-D dataset. The first row represents the warped degraded depth images, which are captured by Kinect-v2 and then warped based on camera parameters. The warped degraded depth images have the same view and size as the corresponding color images shown in the third row, and meanwhile have structure missing and random missing. The second row represents the pseudo GT. The third row represents the high-quality color image captured by Kinect-v2. To better visualize degraded LR depth maps, we show the colorized version.

posed method cannot produce continuous edges in some cases due to large area depth missing. We will further improve this issue in the future work.

5. Conclusion

We propose a novel depth image SR method guided by an edge map. Despite many edge guided SR methods, it is difficult to produce high quality edge maps for traditional methods in favor of SR process. In this paper, we propose to predict a high quality edge map separately from color and depth multi-scale features. The SR subnetwork learns general and specific features from weight sharing and adaptive modules respectively via a cascade strategy. Compared with state-of-the-art SR methods, our method achieves the best results in different datasets. We further construct a benchmark dataset captured by Kinect-v2 to promote the research on real-world data.

Acknowledgment

Part of the work was done during Zhongyu Jiang's visiting in Cardiff University as a joint PhD student. The visit is supported by the China Scholarship Council (project number 201806250049).

This work was supported in part by the National Natural Science Foundation of China under Grant 61672378 and Grant 61771339.

References

[1] Y. Zhang, X. Liu, H. Liu, C. Fan, Depth perceptual quality assessment for symmetrically and asymmetrically distorted stereoscopic 3d videos, Signal Processing: Image Communication 78 (2019) 293–305.

[2] L. Chen, J. Zhao, A robust blind watermarking algorithm for depth-image-based rendering 3d images, Signal Processing: Image Communication (2020) 115935.

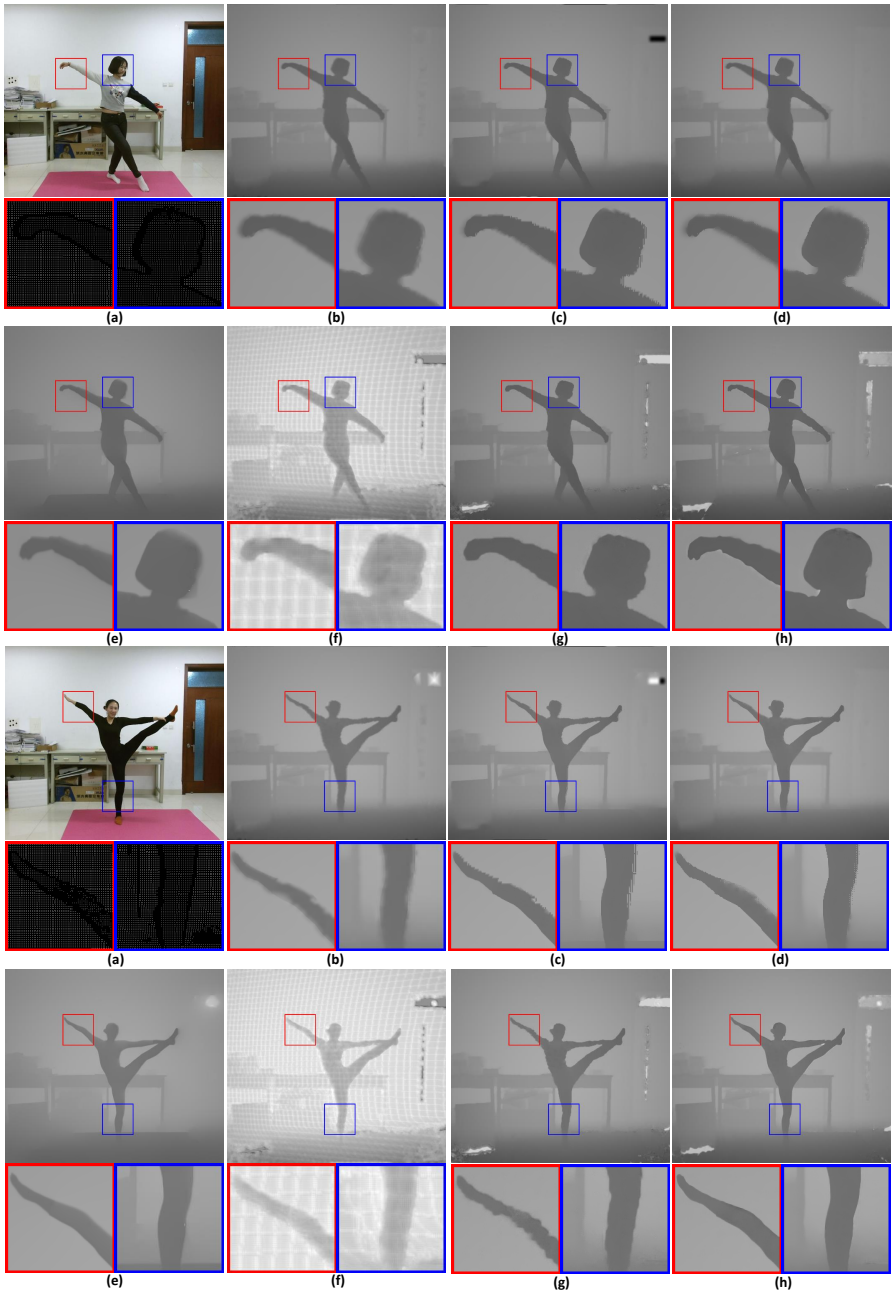

Figure 10: Visual comparison results of real depth maps for two scenes. From left to right and top to bottom, images are produced by (a) RGB and warped LR depth images, (b) GF[53], (c) AR[31], (d) SDF[15], (e) LN[12], (f) DEGR[54], (g) LapSRN[27] and (h) the proposed method.

[3] X. Li, Human–robot interaction based on gesture and movement recognition, Signal Processing: Image Communication 81 (2020) 115686.

[4] Q. Chang, Z. Xiong, Vision-aware target recognition towards autonomous robot by kinect sensors, Signal Processing: Image Communication (2020) 115810.

[5] G. Li, C. Li, Learning skeleton information for human action analysis using kinect, Signal Processing: Image Communication (2020) 115814.

[6] M. Kiechle, S. Hawe, M. Kleinsteuber, A joint intensity and depth co-sparse analysis model for depth map super-resolution, in: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE, 2013, pp. 1545–1552.

[7] M.-Y. Liu, O. Tuzel, Y. Taguchi, Joint geodesic upsampling of depth images, in: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE, 2013, pp. 169–176.

[8] Y. Zuo, Q. Wu, J. Zhang, P. An, Explicit edge inconsistency evaluation model for color-guided depth map enhancement, IEEE Transactions on Circuits and Systems for Video Technology 28 (2) (2016) 439–453.

[9] X. Ye, B. Sun, Z. Wang, J. Yang, R. Xu, H. Li, B. Li, Pmbanet: Progressive multi-branch aggregation network for scene depth super-resolution, IEEE Transactions on Image Processing x (2020) xx.

[10] Y. Zuo, Q. Wu, J. Zhang, P. An, Minimum spanning forest with embedded edge inconsistency measurement model for guided depth map enhancement, IEEE Transactions on Image Processing 27 (8) (2018) 4145–4159.

[11] W. Liu, X. Chen, J. Yang, Q. Wu, Variable bandwidth weighting for texture copy artifact suppression in guided depth upsampling, IEEE Transactions on Circuits and Systems for Video Technology 27 (10) (2017) 2072–2085.

[12] W. Dong, G. Shi, X. Li, K. Peng, J. Wu, Z. Guo, Color-guided depth recovery via joint local structural and nonlocal low-rank regularization, IEEE Transactions on Multimedia 19 (2) (2017) 293–301.

[13] Z. Jiang, Y. Hou, H. Yue, J. Yang, C. Hou, Depth super-resolution from rgb-d pairs with transform and spatial domain regularization, IEEE Transactions on Image Processing 27 (5) (2018) 2587–2602.

[14] M. Huang, X. Xiang, Y. Chen, D. Fan, Weighted large margin nearest center distance-based human depth recovery with limited bandwidth consumption, IEEE Transactions on Image Processing 27 (12) (2018) 5728–5743.

[15] B. Ham, M. Cho, J. Ponce, Robust image filtering using joint static and dynamic guidance, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2015, pp. 4823–4831.

[16] Y. Li, D. Min, M. N. Do, J. Lu, Fast guided global interpolation for depth and motion, in: European Conference on Computer Vision, Springer, 2016, pp. 717–733.

[17] Y. Kim, H. Jung, D. Min, K. Sohn, Deeply aggregated alternating minimization for image restoration, in: IEEE International Conference on Computer Vision and Pattern Recognition, Vol. 1, 2017.

[18] Y. Li, J.-B. Huang, N. Ahuja, M.-H. Yang, Deep joint image filtering, in: European Conference on Computer Vision, Springer, 2016, pp. 154–169.

[19] K.-H. Lo, Y. Wang, K.-L. Hua, Edge-preserving depth map upsampling by joint trilateral filter, IEEE Trans. Cybern 13 (2017) 1–14.

[20] J. Xie, R. S. Feris, M.-T. Sun, Edge-guided single depth image super resolution, IEEE Transactions on Image Processing 25 (1) (2016) 428–438.

[21] Y.-W. Tai, S. Liu, M. S. Brown, S. Lin, Super resolution using edge prior and single image detail synthesis, in: 2010 IEEE computer society conference on computer vision and pattern recognition, IEEE, 2010, pp. 2400–2407.

[22] D. Ferstl, M. Ruther, H. Bischof, Variational depth superresolution using example-based edge representations, in: IEEE International Conference on Computer Vision, 2015, pp. 513–521.

[23] X. Liu, D. Zhai, R. Chen, X. Ji, D. Zhao, W. Gao, Depth super-resolution via joint color-guided internal and external regularizations, IEEE Transactions on Image Processing 28 (4) (2018) 1636–1645.

[24] S. Mandal, A. Bhavsar, A. K. Sao, Depth map restoration from undersampled data, IEEE Transactions on Image Processing 26 (1) (2017) 119–134.

[25] S. Yang, J. Liu, Y. Fang, Z. Guo, Joint-feature guided depth map super-resolution with face priors, IEEE transactions on cybernetics 48 (1) (2016) 399–411.

[26] Y. Wen, B. Sheng, P. Li, W. Lin, D. D. Feng, Deep color guided coarse-to-fine convolutional network cascade for depth image super-resolution, IEEE Transactions on Image Processing 28 (2) (2018) 994–1006.

[27] W.-S. Lai, J.-B. Huang, N. Ahuja, M.-H. Yang, Deep laplacian pyramid networks for fast and accurate super-resolution, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 624–632.

[28] J. Li, Z. Lu, G. Zeng, R. Gan, H. Zha, Similarity-aware patchwork assembly for depth image super-resolution, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2014, pp. 3374–3381.

[29] J. Kopf, M. F. Cohen, D. Lischinski, M. Uyttendaele, Joint bilateral upsampling, in: ACM Transactions on Graphics (ToG), Vol. 26, ACM, 2007, p. 96.

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

[31] J. Yang, X. Ye, K. Li, C. Hou, Y. Wang, Color-guided depth recovery from rgb-d data using an adaptive autoregressive model, IEEE Transactions on Image Processing 23 (8) (2014) 3443–3458.

[32] X. Liu, D. Zhai, R. Chen, X. Ji, D. Zhao, W. Gao, Depth restoration from rgb-d data via joint adaptive regularization and thresholding on manifolds, IEEE Transactions on Image Processing.

[33] W. Liu, X. Chen, J. Yang, Q. Wu, Robust color guided depth map restoration, IEEE Transactions on Image Processing 26 (1) (2017) 315–327.

[34] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rüther, H. Bischof, Image guided depth upsampling using anisotropic total generalized variation, in: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE, 2013, pp. 993–1000.

[35] Y. Zhang, Y. Feng, X. Liu, D. Zhai, X. Ji, H. Wang, Q. Dai, Color-guided depth image recovery with adaptive data fidelity and transferred graph laplacian regularization, IEEE Transactions on Circuits and Systems for Video Technology 30 (2) (2019) 320–333.

[36] Y. Kim, B. Ham, C. Oh, K. Sohn, Structure selective depth superresolution for rgb-d cameras, IEEE Transactions on Image Processing 25 (11) (2016) 5227–5238.

[37] C. Guo, C. Li, J. Guo, R. Cong, H. Fu, P. Han, Hierarchical features driven residual learning for depth map super-resolution, IEEE Transactions on Image Processing 28 (5) (2018) 2545–2557.

[38] Y. Zuo, Q. Wu, Y. Fang, P. An, L. Huang, Z. Chen, Multi-scale frequency reconstruction for guided depth map super-resolution via deep residual network, IEEE Transactions on Circuits and Systems for Video Technology 30 (2) (2019) 297–306.

[39] L. Zhao, H. Bai, J. Liang, B. Zeng, A. Wang, Y. Zhao, Simultaneous color-depth super-resolution with conditional generative adversarial networks, Pattern Recognition 88 (2019) 356–369.

[40] O. Voynov, A. Artemov, V. Egiazarian, A. Notchenko, G. Bobrovskikh, E. Burnaev, D. Zorin, Perceptual deep depth super-resolution, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 5653–5663.

[41] X. Deng, P. L. Dragotti, Deep coupled ista network for multi-modal image super-resolution, IEEE Transactions on Image Processing 29 (2019) 1683–1698.

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

[43] R. d. Lutio, S. D'Aronco, J. D. Wegner, K. Schindler, Guided super-resolution as pixel-to-pixel transformation, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 8829–8837.

[44] Y. Zuo, Y. Fang, P. An, X. Shang, J. Yang, Frequency-dependent depth map enhancement via iterative depth-guided affine transformation and intensity-guided refinement, IEEE Transactions on Multimedia (2020) xx.

[45] X. Deng, P. L. Dragotti, Deep convolutional neural network for multi-modal image restoration and fusion, IEEE Transactions on Pattern Analysis and Machine Intelligence (2020) xx.

[46] X. Song, Y. Dai, D. Zhou, L. Liu, W. Li, H. Li, R. Yang, Channel attention based iterative residual learning for depth map super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 5631–5640.

[47] X. Gu, Y. Guo, F. Deligianni, G.-Z. Yang, Coupled real-synthetic domain adaptation for real-world deep depth enhancement, IEEE Transactions on Image Processing 29 (2020) 6343–6356.

[48] S. Xie, Z. Tu, Holistically-nested edge detection, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1395–1403.

[49] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, X. Bai, Richer convolutional features for edge detection, in: Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on, IEEE, 2017, pp. 5872–5881.

[50] H. Zheng, M. Ji, H. Wang, Y. Liu, L. Fang, CrossNet: An end-to-end reference-based super resolution network using cross-scale warping, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 88–104.

[51] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-c. Woo, Convolutional lstm network: A machine learning approach for precipitation nowcasting, in: Advances in neural information processing systems, 2015, pp. 802–810.

[52] J. He, C. Dong, Y. Qiao, Modulating image restoration with continual levels via adaptive feature modification layers, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11056–11064.

[53] K. He, J. Sun, X. Tang, Guided image filtering, in: European conference on computer vision, Springer, 2010, pp. 1–14.

[54] W. Yang, J. Feng, J. Yang, F. Zhao, J. Liu, Z. Guo, S. Yan, Deep edge guided recurrent residual learning for image super-resolution, IEEE Transactions on Image Processing 26 (12) (2017) 5895–5907.

[55] D. J. Butler, J. Wulff, G. B. Stanley, M. J. Black, A naturalistic open source movie for optical flow evaluation, in: A. Fitzgibbon et al. (Eds.)

(Ed.), European Conf. on Computer Vision (ECCV), Part IV, LNCS 7577, Springer-Verlag, 2012, pp. 611–625.

[56] http://vision.middlebury.edu/stereo/data/.middlebury datasets.

[57] K. Honauer, O. Johannsen, D. Kondermann, B. Goldluecke, A dataset and evaluation methodology for depth estimation on 4d light fields, in: Asian Conference on Computer Vision, 2016, pp. 19–34.

[58] A. Handa, T. Whelan, J. Mcdonald, A. J. Davison, A benchmark for rgb-d visual odometry, 3d reconstruction and slam, in: IEEE International Conference on Robotics and Automation, 2014, p. 15241531.

[59] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the 22nd ACM international conference on Multimedia, ACM, 2014, pp. 675–678.

[60] C. M. C.-Y. Yang, M.-H. Yang, Single-image superresolution: A benchmark (2014) x.

[61] N. Silberman, D. Hoiem, P. Kohli, R. Fergus, Indoor segmentation and support inference from rgbd images, in: European Conference on Computer Vision, Springer, 2012, pp. 746–760.