applied sciences

MDPI

Article

Depth Image Super Resolution Based on Edge-Guided Method

Dongsheng Zhou *, Ruyi Wang, Jian Lu and Qiang Zhang

Key Laboratory of Advanced Design and Intelligent Computing, Ministry of Education, Dalian University, Dalian 116622, China; wangruyimail@126.com (R.W.); lujian@dlu.edu.cn (J.L.); zhangq26@126.com (Q.Z.)
* Correspondence: donyson@126.com; Tel.: +86-411-87402045

Received: 26 December 2017; Accepted: 12 February 2018; Published: 18 February 2018

Abstract: Depth image super-resolution (SR) is a technique which can reconstruct a high-resolution (HR) depth image from a low-resolution (LR) depth image. Its purpose is to obtain HR details to meet the needs of various applications in computer vision. In general, conventional depth image SR methods often cause edges in the final HR image to be blurred or ragged. To solve this problem, an edge-guided method for depth image SR is presented in this paper. To get high-quality edge information, a pair of sparse dictionaries was applied to reconstruct edges of depth image. Then, with the guidance of these high-quality edges, a depth image was interpolated by using a modified joint bilateral filter. Edge-guided method can preserve the sharpness of edges and effectively avoid generating blurry and ragged edges when SR is performed. Experiments showed that the proposed method can get better results on both subjective and objective evaluation, and the reconstructed performance was superior to conventional depth image SR methods.

Keywords: depth image; super-resolution; sparse coding; joint bilateral filter

1. Introduction and Related Works

In recent years, with the rapid development of computer vision technology, the depth information of scenes becomes increasingly essential for many applications, such as 3D Reconstruction [1,2], Augmented Reality [3], Robot Navigation [4] and so on. Some active sensors [5], such as Kinect and PMD (Photonic Mixer Device), can easily acquire depth information of scenes. Then, this information will be used to create a depth image. However, due to the theoretical and practical limitations, the achievable resolution of any depth imaging device is usually too low to meet the needs of many practical applications. How to improve depth image resolution is an urgent problem that needs to be solved. One way to solve this problem is to apply some sophisticated vision sensors. However, these sensors are usually very expensive. Another way is to use super-resolution (SR) algorithm. Compared with expensive sensors, SR algorithm, not relying on hardware configuration, is evidently a low-cost approach. Inspired by the idea of color image SR, researchers have proposed many promising depth image SR methods [6–8] in recent years, which can improve the resolution of depth image effectively.

According to the difference of referenced information, depth SR can be mainly divided into four categories: (1) interpolation; (2) SR from LR depth image frames; (3) SR through fusing depth image and HR color image; and (4) example-based SR.

(1) Interpolation: There are many analytic methods for image interpolation, including nearest neighbor interpolation, bilinear and cubic interpolation [9]. However, when interpolation is done by a large factor, these analytic methods can cause image edge to be ragged and blurred because of the big value difference of pixels across edges. To solve the problem, Pang [10] presents an SR method based on bilinear interpolation and adaptive sharpening filter. This method can suppress effectively the

edge-blurred effect. Ning [11] proposes an improved cubic interpolation algorithm, which uses cubic interpolation to compute the pixels in smooth areas and uses edge-vector interpolation to compute the pixels near edges. Xie [8] presents an edge-guided approach. This method reconstructs sharp HR edges through Markov random field at first, and then an SR depth image can be interpolated under the guidance of these edges. With the help of edge-guided information, the sharpness of edges can be well preserved in the final SR image. At the same time, some bilateral filtering methods [12,13] can also preserve edge well.

(2) SR from LR depth image frames: The HR image can be reconstructed by fusing the complementary information among a few LR depth images captured from the same scene. Schuon [14] uses an optimization framework with a bilateral total-variation regularization term to solve such a SR problem. Rjagopalan [15] constructs an energy function through the Markov Random Field, and minimizes the energy function to get the HR image. Ismaeil [16] proposes a dynamic-scene SR method for depth image to deal with the problem of non-rigid body motion between LR images. Gevrekci [17] uses convex projection method to construct the imaging model of depth image sequence for depth image SR.

(3) SR through fusing depth image and HR color image: Most commercial depth cameras can get a depth image and a color image about the same scene simultaneously, and usually the resolution of the color image is higher than that of the depth image. Thus, the HR depth image can be reconstructed with the help of HR color image. Ferstl [18] calculates an anisotropic total variation diffusion tensor from HR color image, and then the tensor is used to reconstruct the SR depth image. Yang [19] combines bilateral filter with median filter to compute adaptively weights from the HR image, and then the depth image is interpolated according to these weights. Lo [20] proposes a depth image SR method based on joint trilateral filter. The method considers not only the weight of distance, but also the weight of pixel value and gradient.

(4) Example-based SR: This method learns the transformation between LR and HR image from example database, and then an HR depth image can be reconstructed through the learned transformation when an LR depth image is inputted. Yang [21] uses a sparse coding method to grasp the transformation. Therefore, HR image patches can be represented by a sparse linear combination of HR dictionary atoms. Zeyde [6] modifies the above sparse coding method, and uses the K-SVD [22] and the orthogonal matching pursuit (OMP) [23] to train an LR and HR dictionary pair. Xie [24] proposes a pairwise dictionary training method with local coordinate constraints for depth image SR. Timofte [7] clusters the dictionary atoms into sub-dictionaries using K-NN algorithm, and then the HR patches can be represented by the most suited sub-dictionary. Kim [25] presents an accurate color image SR method based on VGG-NET [26], which can also be applied to solve the depth image SR problem.

Although the above methods can effectively reconstruct SR depth image from LR input, some existing problems cannot be ignored as well. The methods of the first category can cause discontinuous regions jagged and blurred. The methods of the second category need to be subject to the rigorous assumption that adjacent images only have slight movements on the plane parallel to the focal plane of camera. This assumption is usually difficult to satisfy in practical scenarios. Depth image SR based on fusing depth image and HR color image needs first to obtain an HR color image which register fully with the depth image. The example-based method has a strong dependence on training databases. That is, the difference of training databases may have a great effect on experiment results.

To address these problems, in this study, an edge-guided method for depth image SR is presented. We first train a pair of sparse dictionaries to recover high-quality edge information, and then an HR depth image is interpolated with the guidance of these high-quality edges. This method is a mixture of the example-based method and the interpolated method. We make full use of the advantage of the two methods. In this way, the proposed method can achieve improved results that are comparable to current state-of-the-art methods. Our approach needs neither strict assumptions nor the assistance of HR color image, so it can be used to improve depth image resolution conveniently. At the same time, our approach not only can achieve the goal of preserving sharp edge in depth image SR, but also can get a better color image SR.

The remainder of this paper is organized as follows. A detailed overview of the proposed method is presented in Section 2. Section 3 reports and discusses the results of the experiments. Finally, Section 4 concludes the paper.

2. Proposed Method

In this section, we first present the general steps of our work. Then, the way we have built the LR and HR edge dictionaries is discussed. Afterward, we continue with the details of how to interpolate HR image by joint bilateral filter.

To keep blurred and jagged edges away from the final SR result, we present a novel depth image SR method, which employs joint bilateral filter based on edge guidance for LR-to-HR reconstruction. The general steps of the proposed depth image SR method are summarized and shown in Figure 1.

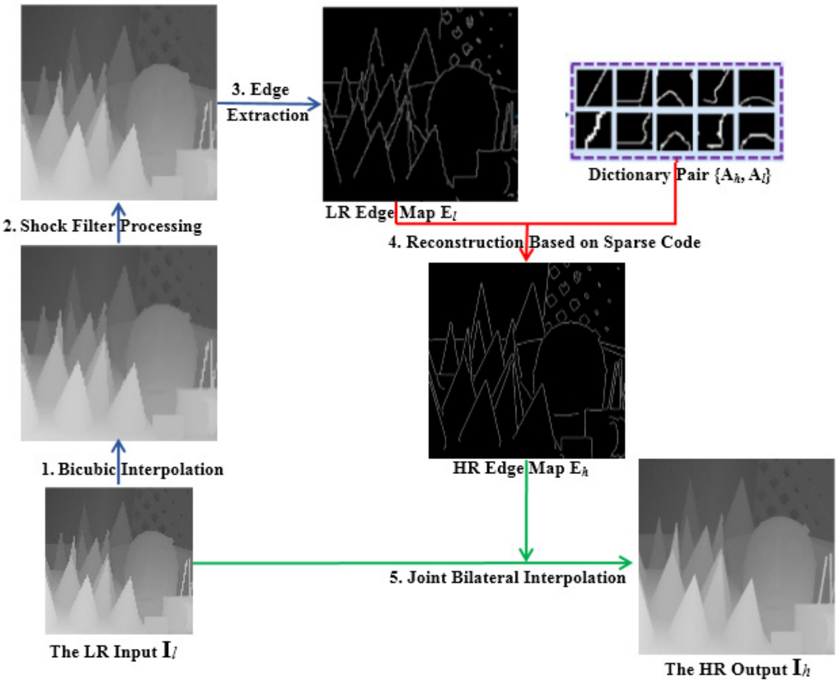


Figure 1. Pipeline of the edge-guided depth image SR.

To avoid computational complexity caused by the different size between LR image and the final HR image, we first use simple interpolation algorithm (bicubic interpolation) to magnify the input LR image $\mathbf{I}_l$ to the same size as the final HR image. However, interpolation algorithm can cause blurred and jagged effects near edges, so we use a Shock filter [27] to clear the magnified image for further process.

Edges provide essential structural information to describe the objects in the scene. Thus, we first focus to recover HR edges before reconstructing the whole image. As illustrated in Figure 1, LR edge map $\mathbf{E}_l$ is extracted from the preprocessed LR image. Edge map preserves only the primary structure information and abandons widespread smooth area. This leads to it having very strong sparseness. Thus, we choose sparse coding method to recover HR edge map $\mathbf{E}_h$ in our method.

After getting HR edge map $\mathbf{E}_h$, depth image $\mathbf{I}_h$ will be interpolated by a modified joint bilateral filter under the guidance of the high-quality edge map. The usage of bilateral filter can not only preserve the edge sharpness but also suppress noise further.

From the above introduction, the proposed method mainly includes two important parts: (1) edge recovery using sparse coding method; and (2) edge-guided depth interpolation using bilateral filter. The details on these two parts will be discussed in the following subsections.

2.1. Edge Recovery Using Sparse Coding

In this section, we first present some notation for our work. Then, the way we have built the LR and HR dictionaries for edge map recovery is discussed.

2.1.1. Sparse Dictionary Training

The LR and HR images are represented as $\mathbf{z}_l \in \mathbf{R}^{N_l}$, and $\mathbf{y}_h \in \mathbf{R}^{N_h}$, where $N_h = s^2 \cdot N_l$, and $s > 1$ is some integer scale-up factor. The blur operator is denoted by $\mathbf{H} : \mathbf{R}^{N_h} \to \mathbf{R}^{N_l}$, and the decimation operator for a factor s is denoted by $\mathbf{D} : \mathbf{R}^{N_h} \to \mathbf{R}^{N_l}$. The acquisition model of how to generate an LR image from an HR image can be described as:

$$\mathbf{z}_l = \mathbf{D}\mathbf{H}\mathbf{y}_h + \mathbf{v} \tag{1}$$

where $\mathbf{v}$ is an additive noise in the acquisition process. Given $\mathbf{z}_l$, the problem is to find $\hat{\mathbf{y}} \in \mathbf{R}^{N_h}$ such that $\hat{\mathbf{y}} \approx \mathbf{y}_h$. That is, $\|\hat{\mathbf{y}} - \mathbf{y}_h\|_2$ tends to zero. To avoid the complexities caused by the different resolutions between $\mathbf{z}_l$ and $\mathbf{y}_h$, it is assumed that the image $\mathbf{z}_l$ is scaled-up by a simple interpolation operator $\mathbf{Q} : \mathbf{R}^{N_l} \to \mathbf{R}^{N_h}$ (e.g., bicubic interpolation) that fills out the missing pixels between the original pixels in the input LR image. The scaled-up image shall be denoted by $\mathbf{y}_l$ and it satisfies the relation:

$$\mathbf{y}_l = \mathbf{Q}\mathbf{z}_l = \mathbf{Q}(\mathbf{D}\mathbf{H}\mathbf{y}_h + \mathbf{v}) = \mathbf{Q}\mathbf{D}\mathbf{H}\mathbf{y}_h + \mathbf{Q}\mathbf{v} = \mathbf{U}\mathbf{y}_h + \overline{\mathbf{v}} \tag{2}$$

The reconstruction problem now is cast to process $\mathbf{y}_l \in \mathbf{R}^{N_h}$ and produce a result $\hat{\mathbf{y}}_h \in \mathbf{R}^{N_h}$, which will get as close as possible to the original HR image, $\mathbf{y}_h \in \mathbf{R}^{N_h}$.

The algorithm we propose operates on patches extracted from $\mathbf{y}_l$, aiming to estimate the corresponding patch from $\mathbf{y}_h$. Let $\mathbf{p}^k = \mathbf{R}_n^k\mathbf{y}$ be an image patch of size $n \times n$ centered at location k and extracted from the image $\mathbf{y}$ by the linear operator $\mathbf{R}$. The stride d is used for spatially shifting of image patches. Hence, the LR and HR patches are extracted as:

$$\mathbf{p}_l^k = \mathbf{R}_n^k\mathbf{y}_l, \mathbf{p}_h^k = \mathbf{R}_n^k\mathbf{y}_h \tag{3}$$

It shall be further assumed that $\mathbf{p}_l^k$ and $\mathbf{p}_h^k$ can be represented sparsely by coefficients $\mathbf{q}^k$ over the dictionary pair $\mathbf{A}_l$ and $\mathbf{A}_h$, respectively, namely:

$$\mathbf{p}_l^k = \mathbf{A}_l\mathbf{q}^k, \mathbf{p}_h^k = \mathbf{A}_h\mathbf{q}^k \tag{4}$$

To acquire such a dictionary pair $\mathbf{A}_l$ and $\mathbf{A}_h$, we choose to apply jointly dictionary training method proposed in ref. [7].

The flow of training dictionary pair is summarized in Algorithm 1. The first step is to construct the training set. A set of HR training images $\left\{\mathbf{y}_h^j\right\}_j$ are collected, LR images $\left\{\mathbf{y}_l^j\right\}_j$ are constructed using scale-down operator $\mathbf{U}$ and pairs of matching patches that form the training database $\left\{\mathbf{p}_h^k, \mathbf{p}_l^k\right\}_k$, are extracted. After finishing training database preparation, we can enter into dictionary learning stage.

For LR dictionary $\mathbf{A}_l$, the K-SVD dictionary training procedure [22] is applied to LR patches $\left\{\mathbf{p}_l^k\right\}_k$, resulting in the dictionary $\mathbf{A}_l$:

$$\mathbf{A}_l, \{\mathbf{q}^k\} = \underset{\mathbf{A}_l, \{\mathbf{q}^k\}}{\operatorname{argmin}} \sum_k \|\mathbf{p}_l^k - \mathbf{A}_l\mathbf{q}^k\|^2 \quad s.t. \|\mathbf{q}^k\|_0 \le L \tag{5}$$

A side product of this training is the sparse representation coefficients vectors $\left\{\mathbf{q}^k\right\}_k$ that correspond to the training patches $\left\{\mathbf{p}_l^k\right\}_k$. $\| \|_0$ is the zero norm, and $\|\mathbf{q}^k\|_0$ is used to count the nonzero entries of vector $\mathbf{q}^k$. L is a constant that controls sparsity.

The next step is the high-resolution dictionary construction. Recall that we assume that the HR patch $\mathbf{p}_h^k$ can be approximated by $\mathbf{p}_h^k = \mathbf{A}_h\mathbf{q}^k$. The dictionary $\mathbf{A}_h$ is therefore sought such that this approximation is as exact as possible, i.e.,

$$\mathbf{A}_h = \underset{\mathbf{A}_h}{\operatorname{argmin}} \sum_k \|\mathbf{p}_h^k - \mathbf{A}_h\mathbf{q}^k\|_2^2 = \underset{\mathbf{A}_h}{\operatorname{argmin}} \|\mathbf{P}_h - \mathbf{A}_h\mathbf{Q}\|_F^2 \tag{6}$$

where the matrix $\mathbf{P}_h$ is constructed with the HR training patches $\left\{\mathbf{p}_h^k\right\}_k$ as its columns, and similarly, $\mathbf{Q}$ contains $\left\{\mathbf{q}^k\right\}_k$ as its columns (give that $\mathbf{Q}$ has full row rank). $\| \|_F$ is the Frobenius norm [28]. The solution of the least-squares problem is given by the following expression:

$$\mathbf{A}_h = \mathbf{P}^h\mathbf{Q}^T\left(\mathbf{Q}\mathbf{Q}^{\mathrm{T}}\right)^{-1} \tag{7}$$

Algorithm 1.

Input: A set of HR training images $\left\{\mathbf{y}_h^j\right\}_j$

Output: LR-HR dictionary pairwise $\{\mathbf{A}_l, \mathbf{A}_h\}$

Step 1. Construct training set: use scale-down operator $\mathbf{U}$ to construct LR images $\left\{\mathbf{y}_l^j\right\}_j$ from HR training images $\left\{\mathbf{y}_h^j\right\}_j$ and extract pairs of matching patches that form the training database $\left\{\mathbf{p}_h^k, \mathbf{p}_l^k\right\}_k$ from images $\left\{\mathbf{y}_h^j\right\}_j$ and $\left\{\mathbf{y}_l^j\right\}_j$.

Step 2. LR dictionary training: apply K-SVD [22] dictionary training procedure to train LR patches $\left\{\mathbf{p}_l^k\right\}_k$, resulting with LR dictionary $\mathbf{A}_l$ and the sparse representation coefficients vectors $\left\{\mathbf{q}^k\right\}_k$.

Step 3. HR dictionary training: HR dictionary $\mathbf{A}_h$ is trained using the sparse representation coefficients vectors $\left\{\mathbf{q}^k\right\}_k$ to match corresponding LR one

2.1.2. Edge Map Recovery

Once we get LR-HR dictionary pair $\{\mathbf{A}_l, \mathbf{A}_h\}$, high-quality edge map $\mathbf{E}_h$ can be represented by a sparse linear combination of HR dictionary atoms. Before starting reconstruction, we first process the input LR image $\mathbf{I}_l$ to obtain an LR edge map $\mathbf{E}_l$. The process can be divided into the following three steps:

(1) The input image $\mathbf{I}_l$ is interpolated to the same size as the desired HR image using bicubic interpolation algorithm, producing an LR image $\hat{\mathbf{I}}_l$.

(2) Shock filter [27] is applied to suppress zigzag effect produced by up-sampling interpolation.

(3) Canny operator is used to extract edge $\mathbf{E}_l$ from $\hat{\mathbf{I}}_l$.

Then, the HR edge map $\mathbf{E}_h$ can be reconstructed from $\mathbf{E}_l$ using the LR-HR dictionary pair $\{\mathbf{A}_l, \mathbf{A}_h\}$. The process is described in Algorithm 2.

Algorithm 2.

Input: LR-HR dictionary pairwise $\{\mathbf{A}_l, \mathbf{A}_h\}$ and edge map $\mathbf{E}_l$
Output: High-quality edge map $\mathbf{E}_h$
Step 1. Extract patches $\left\{\mathbf{b}_l^k\right\}_k$ from edge map $\mathbf{E}_l$;
Step 2. Patches $\left\{\mathbf{b}_l^k\right\}_k$ can be represented by the atoms of LR dictionary $\mathbf{A}_l$, and the side product is the corresponding sparse coefficients $\left\{\mathbf{c}^k\right\}_k$;
Step 3. Multiply the obtained sparse coefficients $\left\{\mathbf{c}^k\right\}_k$ by HR dictionary $\mathbf{A}_h$ to find HR patches $\left\{\mathbf{b}_h^k\right\}_k$;
Step 4. The high-quality edge map $\mathbf{E}_h$ can be constructed by merging these HR patches $\left\{\mathbf{b}_h^k\right\}_k$, and the overlap regions of image patches are processed by the method of Zeyde [6].

2.2. Edge-Guided Depth Interpolation

In this section, we first introduce some notation during interpolation. Then, the method of discriminating pixels distribution is discussed.

2.2.1. Modified Joint Bilateral Filter

After obtaining HR edge image $\mathbf{E}_h$, HR depth image $\mathbf{I}_h$ can be interpolated through a modified joint bilateral filter with the guidance of $\mathbf{E}_h$. For each pixel p in the target HR depth image $\mathbf{I}_h$, its value can be interpolated by a local neighborhood of LR image:

$$\mathbf{I}_h(p) = \frac{1}{k_p} \sum_{q \in N(p)} \mathbf{I}_l(q\downarrow) \cdot f_s(\| p\downarrow - q\downarrow \|) \cdot f_r(\mathbf{E}_h, p, q) \tag{8}$$

where $N(p)$ is an $s \times s$ neighborhood window centered at pixel p. $p\downarrow$ and $q\downarrow$ represent pixel coordinate corresponding to pixel p and pixel q in the LR depth image $\mathbf{I}_l$, and only integer coordinate is considered. $f_s(\cdot)$ is a Gaussian kernel with standard deviation σ and mean value 0, which is used to weight the correlation of different pixel in the neighborhood. k_p is a normalizing factor. $f_r(\cdot)$ is a binary indicator, which determines whether or not two pixels are on the same side of the edge. The indicator is defined as:

$$f_r(\mathbf{E}_h, p, q) = \begin{cases} 1 & if\ pixel\ p\ and\ pixel\ q\ are\ at\ the\ same\ side\ of\ \mathbf{E}_h \\ 0 & otherwise \end{cases} \tag{9}$$

The concrete form of $f_r(\cdot)$ can be created by discriminating the distribution of pixels p and q.

2.2.2. Discrimination of pixels distribution

Firstly, the set $\mathbf{C}_e$ is used to store the pixels on the edge. The pixels on the line segment between pixels p and q are stored in set $\mathbf{L}$. Pixels p and pixel q are on the same side of the edge if the intersection of sets $\mathbf{C}_e$ and $\mathbf{L}$ is null, as shown Figure 2a. The distribution of pixels p and q may have two situations when the intersection of sets $\mathbf{C}_e$ and $\mathbf{L}$ is not null. Pixels p and q are not on the same side of the edge in Figure 2b, but they are on the same side in Figure 2c. In this situation, we divide each neighborhood window into some sets according to the edge. The process is summarized in Algorithm 3.

As shown in Figure 2, white lines represent the edge pixels, the whole black portion is a to-be-divided area, and an image patch will be area-divided based on the connectivity of the black area. In addition, there are some special edge curve formats that need to be stated clearly. As shown in Figure 3, if the edge curve is not traversing the entire image patch, we think this is a special form of connectivity, that is, a form where interior space which is enclosed within the edge pixels is zero. Furthermore, the details of the algorithm are as follows.

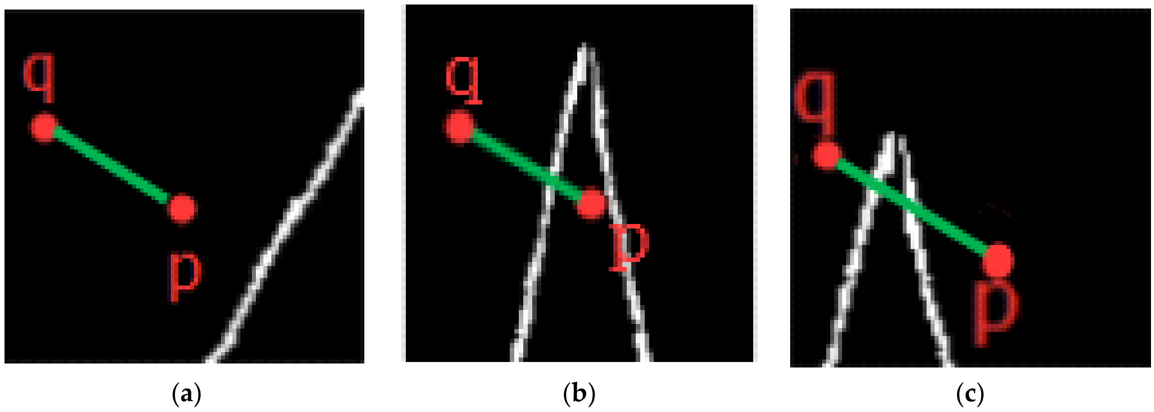


Figure 2. Distinguish of two pixels near edge.

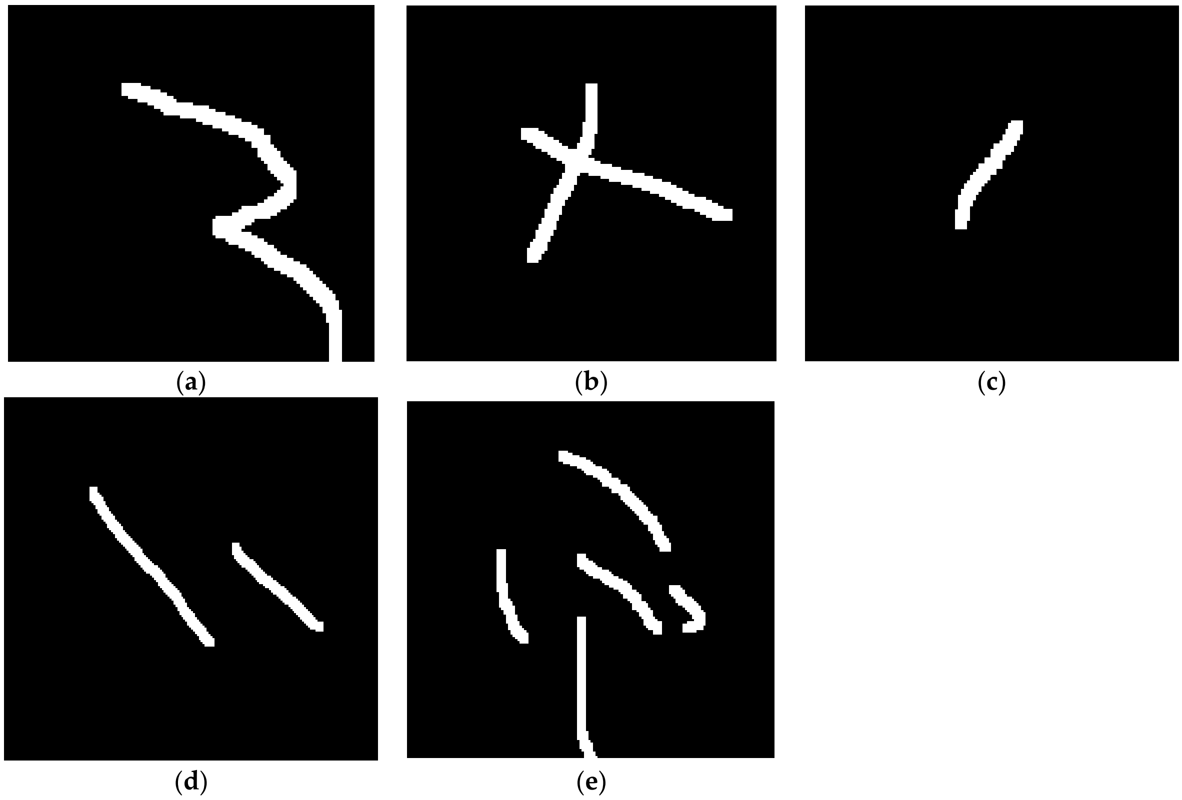


Figure 3. Some special forms of the edge curves.

Algorithm 3.

Input: An image patch **A** with edge pixels and the set $\mathbf{C}_e$

Output: Different sets $\mathbf{C}_i$ $(i = 1, 2, 3 \ldots \ldots n)$, where i is the index of sets, and n is the total number of sets.

Step 1. The initial pixel r is chosen randomly from **A**. The following will be sequentially obtained based on the coordinates;

Step 2. If $r \notin \mathbf{C}_e$, we assume it belongs to $\mathbf{C}_1$. If $r \in \mathbf{C}_e$, the algorithm returns to the Step 1;

Step 3. Adjacent pixels of the newly added pixels in set $\mathbf{C}_1$ are judged. If the adjacent pixel does not belong to $\mathbf{C}_e$, we add it into set $\mathbf{C}_1$;

Step 4. Repeat Step 3 until set $\mathbf{C}_1$ does not change;

Step 5. The remaining pixels are judged by the same method as $\mathbf{C}_1$;

Step 6. Area **A** is divided into different pixel sets $\mathbf{C}_i$ $(i = 1, 2, 3 \ldots \ldots n)$.

After determining the distribution of pixels, pixels p and q can be discriminated easily whether on the same side of the edge. They are on the same side of the edge when they belong to the same pixel set, otherwise they are not on the same side. Once the kernel functions of bilateral filter are determined, the HR depth image can be interpolated using Equation (9). When interpolation is performed, the Gaussian kernel $f_s(\cdot)$ also suppresses some noise for the depth values. In addition, with the guidance provided by the indicator $f_r(\cdot)$, only pixels at the same side of the edge will be considered during interpolation so that edges can be well preserved.

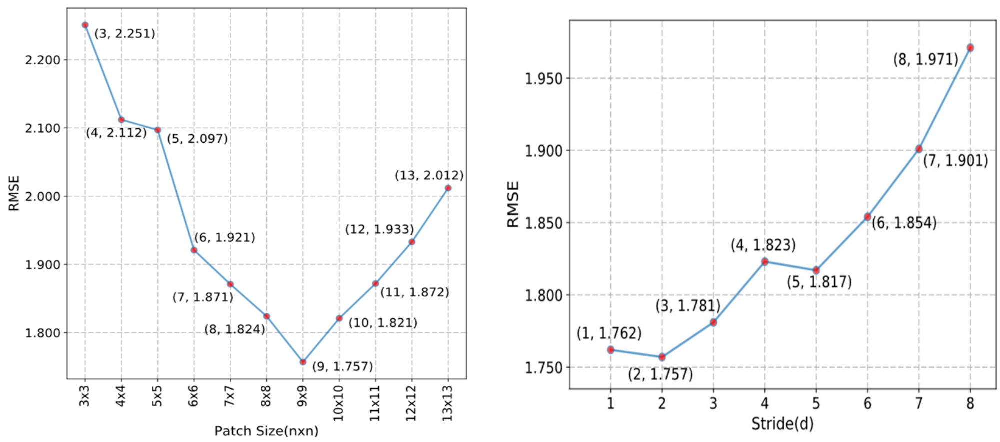
3. Experiments and Analysis

In this section, we analyzed the performance of our proposed depth image SR method and benchmarked it in quantitative and qualitative comparison with other state-of-the-art methods. All the experiments are implemented in a same experimental environment.

3.1. Test Environment and Parameter Setting

In our experiments, the programming tool was MATLAB (v.2016a) [29], and the test environment is the following. The processor was Intel(R) Xeon(R) CPU E5-2620 v3@ 2.40 Hz. Computer memory size was 64.0 Gb. The multithreading technology was used in the experiment. The proposed algorithm supports GPU computing, but it is not used. Test images were from the Middlebury Stereo database [30,31].

Some parameters were selected based on the smallest Root Mean Square Error (RMSE). We calculated average RMSE of 10 test images by varying the size of image patches from $n \times n = 3 \times 3$ to $n \times n = 13 \times 13$ per experiment. The results are depicted in Figure 4a. By comparison, we chose $n \times n = 9 \times 9$ as the size of image patch. Similarly, we also compared the stride d of patch selection in Figure 3b. The stride is determined to be 2. The size of neighborhood window was $s \times s = 7 \times 7$ when joint bilateral filter was performed. The reliability of the value s has been confirmed in [8]. The standard deviation $\sigma = 0.5$ for $f_s(\cdot)$ in Equation (8). Dictionaries were trained using the database from Yang [32] which consisted of 100,000 patches extracted from 30 training images.



(a) **(b)**

Figure 4. The sensitivity of patch size n and stride d.

3.2. Experimental Results and Comparative Analysis

To compare the proposed method quantitatively, we chose RMSE, the Peak Signal Noise Ratio (PSNR), Structural Similarity (SSIM) and Percent of Error (PE) [8] to evaluate experimental results. Tables 1–4 show experimental results of 10 test images from Middlebury Stereo database using different SR methods. These methods included: Neighbor Embedding with Locally Linear Embedding (NE + LLE) [33], Neighbor Embedding with Least Squares (NE + LS) [34], Neighbor

Embedding with Non-Negative Least Squares (NE + NNLS) [35], Global Regression (GR) and Anchored Neighborhood Regression (ANR) [36], Adjusted Anchored Neighborhood Regression for Fast Super-Resolution (AANR) [7], Accurate Image Super-Resolution Using Very Deep Convolutional Networks (CNN) [25], the sparse coding method of Yang [21], the modified sparse coding method of Zeyde [6], and edge-guided method based Markov random field of Xie [8].

Table 1. RMSE values on the Middlebury Stereo database with scaling factor of 4.

RMSE ×4	Bowling	Artl	Aloe	Cones	Indian	Venus	Warrior	Tsukuba	Hand	Dove
NE + LLE	1.841	2.567	2.370	1.380	0.820	0.654	3.645	2.861	1.828	1.004
NE + LS	1.811	2.512	2.332	1.357	0.802	0.634	3.630	2.860	1.839	1.000
NE + NNLS	1.812	2.523	2.326	1.356	0.803	0.637	3.627	2.867	1.820	1.012
GR	1.986	2.838	2.585	1.492	0.906	0.726	3.864	3.142	2.004	1.088
ANR	1.807	2.634	2.429	1.393	0.836	0.662	3.723	2.946	1.885	1.022
AANR	1.855	2.713	2.478	1.456	0.855	0.674	3.707	2.972	1.925	1.043
CNN	2.238	3.798	3.245	1.778	0.987	0.845	4.424	3.505	2.174	1.214
Yang	2.112	3.361	2.865	1.514	0.908	0.733	4.360	3.186	2.026	1.098
Zeyde	1.803	2.494	2.329	1.338	0.798	0.635	3.620	2.844	1.832	0.989
Xie	1.766	2.935	2.583	1.240	0.771	0.617	4.081	3.009	1.926	1.010
Ours	**1.662**	**2.368**	**2.242**	**1.230**	**0.705**	**0.541**	**3.325**	**2.768**	**1.796**	**0.935**

Table 2. SSIM values on the Middlebury Stereo database with scaling factor of 4.

SSIM ×4	Bowling	Artl	Aloe	Cones	Indian	Venus	Warrior	Tsukuba	Hand	Dove
NE + LLE	0.916	0.712	0.875	0.874	0.987	0.937	0.893	0.827	0.983	0.988
NE + LS	0.923	0.735	0.884	0.890	0.988	0.951	0.903	0.814	0.984	0.989
NE + NNLS	0.923	0.728	0.884	0.887	0.989	0.948	0.902	0.829	0.984	0.989
GR	0.899	0.656	0.847	0.850	0.984	0.931	0.875	0.833	0.978	0.985
ANR	0.916	0.711	0.873	0.879	0.987	0.944	0.892	0.782	0.983	0.988
AANR	0.924	0.750	0.880	0.891	0.987	0.953	0.906	0.801	0.985	0.990
CNN	0.922	0.745	0.865	0.880	0.987	0.951	0.904	0.843	0.983	0.988
Yang	0.909	0.677	0.857	0.861	0.985	0.940	0.884	0.795	0.980	0.986
Zeyde	0.925	0.740	0.885	0.893	0.988	0.950	0.905	0.839	0.984	0.989
Xie	0.946	0.791	0.908	0.916	**0.992**	**0.971**	0.931	0.855	0.989	**0.993**
Ours	**0.953**	**0.804**	**0.912**	**0.917**	0.991	0.968	**0.936**	**0.879**	**0.990**	0.992

Table 3. PSNR values on the Middlebury Stereo database with scaling factor of 4.

PSNR ×4	Bowling	Artl	Aloe	Cones	Indian	Venus	Warrior	Tsukub	Hand	Dove
NE + LLE	42.827	39.942	40.634	45.328	49.852	51.818	36.896	38.999	42.901	48.096
NE+LS	42.972	40.099	40.776	45.475	50.045	52.084	36.932	39.001	42.850	48.126
NE + NNLS	42.966	40.091	40.798	45.480	50.034	52.038	36.938	38.980	42.938	48.130
GR	42.171	39.069	39.878	44.653	48.984	50.912	36.389	38.186	42.104	47.396
ANR	42.691	39.716	40.419	45.250	49.678	51.713	36.712	38.759	42.632	47.935
AANR	42.761	39.460	40.245	44.864	49.485	51.553	36.748	38.669	42.451	47.762
CNN	40.667	36.538	37.764	43.131	48.237	49.587	35.214	37.237	41.396	46.444
Yang	41.016	37.600	38.655	44.526	48.960	50.821	35.340	38.064	42.008	47.314
Zeyde	43.008	40.193	40.784	45.599	50.085	52.071	36.954	39.049	42.882	48.219
Xie	42.124	38.778	39.332	46.260	50.384	52.317	35.915	38.560	42.447	48.044
Ours	**44.255**	**41.015**	**41.110**	**46.332**	**51.163**	**53.459**	**37.694**	**39.605**	**43.552**	**48.707**

To make the tables readable, we marked the top three reconstruction methods in the four tables. The value in bold is the best. The value with single underline is the second best, and this is the value that is closest to the best value in the optimal direction. Likewise, the value with double underlines denote the third best, which is closest to the second best value in the optimal direction. From the tables, we can conclude that the RMSE and PSNR values of our method both rank the first in the test results. There were seven SSIM values ranked first and three SSIM values ranked third using our method in the test results. Three PE values of our result were first, and seven PE values were second. These objective measurements showed that our method can get good performance compared with other methods.

Table 4. PE values on the Middlebury Stereo database with scaling factor of 4.

PE ×4	Bowling	Artl	Aloe	Cones	Indian	Venus	Warrior	Tsukuba	Hand	Dove
NE + LLE	6.725	24.519	16.890	9.050	2.403	2.690	9.427	15.243	4.293	2.730
NE + LS	6.216	22.944	15.948	8.155	2.230	2.412	8.638	14.415	4.260	2.584
NE + NNLS	6.482	23.725	16.246	8.478	2.303	2.583	8.918	14.730	4.332	2.782
GR	8.438	30.248	20.611	11.200	3.191	3.539	10.776	18.229	5.137	3.719
ANR	7.017	25.183	17.360	9.052	2.506	2.781	9.912	15.182	4.534	2.911
AANR	5.274	21.084	14.741	7.385	2.047	1.967	7.640	12.816	3.293	2.248
CNN	4.232	19.169	13.454	6.993	1.795	1.696	8.077	11.340	2.624	1.690
Yang	8.208	30.547	19.799	9.955	2.867	2.996	12.692	17.170	5.341	3.430
Zeyde	6.040	22.714	15.751	7.968	2.208	2.447	8.228	14.240	4.036	2.544
Xie	2.405	10.694	8.299	**2.829**	**0.951**	**0.505**	**2.575**	**4.239**	**0.918**	**0.608**
Ours	**2.384**	**10.676**	**8.257**	3.294	1.076	0.641	2.603	4.586	1.089	0.751

We also provided visual assessments on test image "cones" and "tsukuba". The ground-truth HR image and the final SR images using the top five methods in objective evaluation tables (4× scaling factor) are shown in Figures 5 and 6, and note that except for Figures 5a and 6a, all the remaining experimental images for comparison are all generated by ourselves after repeating the original algorithms.

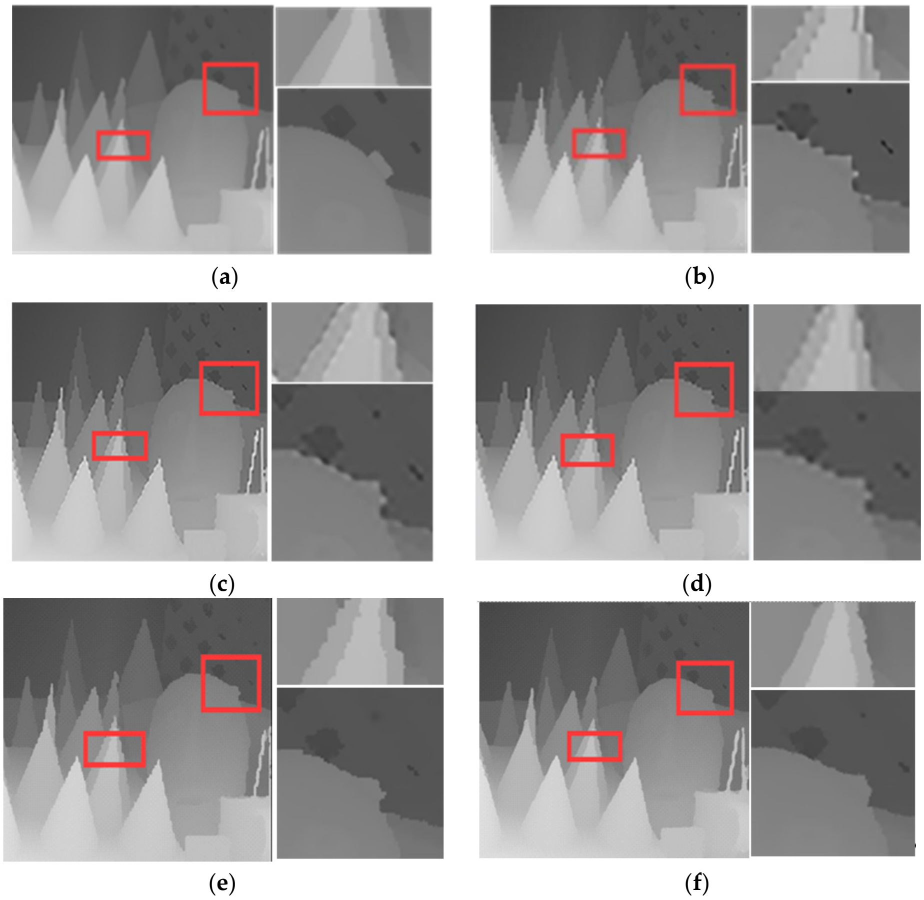


Figure 5. Visual comparisons of "cones": (**a**) ground truth (reprinted with permission from [37]); (**b**) Kim [25]; (c)AANR [7]; (**d**) Zeyde [6]; (**e**) Xie [8]; and (**f**) our method.

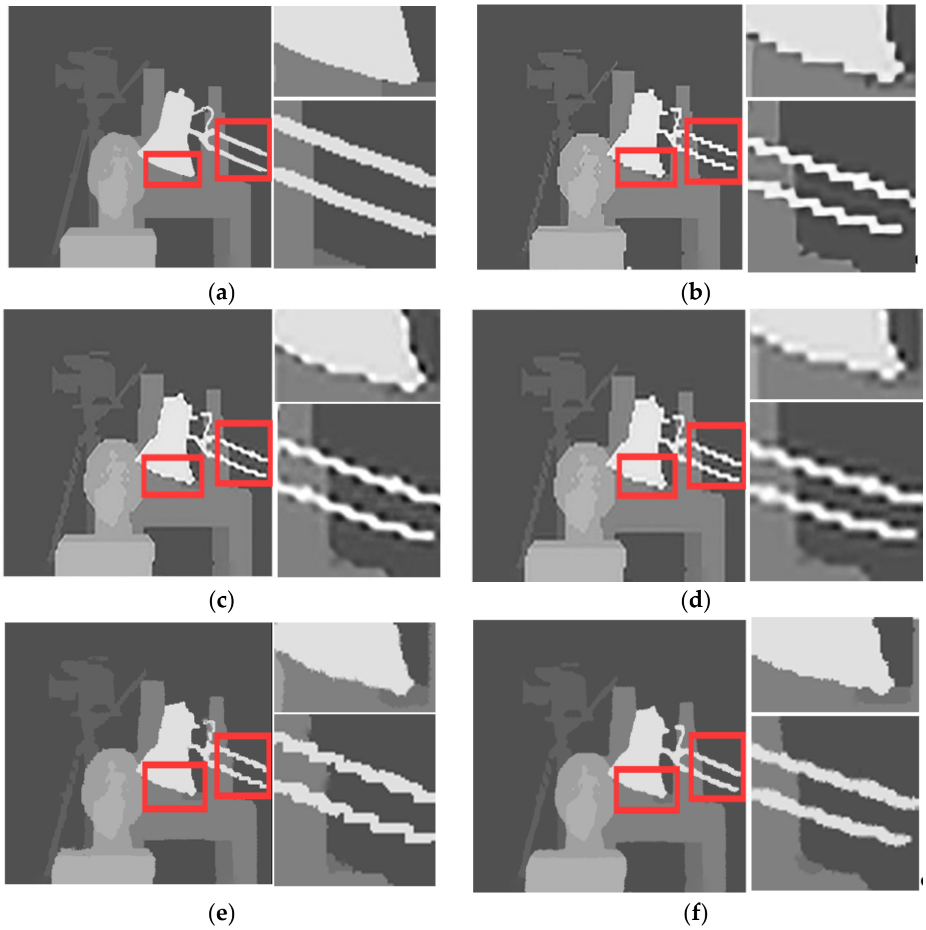


Figure 6. Visual comparisons of "tsukuba": (**a**) ground truth (reprinted with permission from [38]); (**b**) Kim[25]; (**c**)AANR [7]; (**d**) Zeyde[6]; (**e**) Xie[8]; and (**f**) our method.

From the above Figures, we can see that, in the SR result of Kim et al. [25], serious zigzag effect exists. Zeyde [6] and Timofte [7] could relieve zigzags using sparse coding method, but still introduced many artifacts around edges. The method of Xie [8] could get good results, but the detail information of edges could not be reconstructed very well compared with our method. In Figures 5f and 6f, we can see clearly that our reconstructed depth images not only avoided blurred edges, but also reduced zigzags near edges and preserved sharpness of edges.

4. Conclusion and Future Works

Conventional SR methods can cause edges to be blurred and jagged. Aiming at solving this problem, this paper proposes an edge-guided SR method. First, high-quality edge information is reconstructed based on generating a dictionary from pairs of HR and their corresponding LR edge patches. Then, with the guidance of these recovered edges, the SR depth image is interpolated by a joint bilateral filter. The guidance of high-quality edge information can improve the performance of SR algorithm resulting in sharper SR depth image. The quantitative and qualitative analyses of the experimental results showed the superiority of the proposed technique over conventional and state-of-the-art techniques.

There are still some shortages of the proposed method. The running time is higher than some methods shown in Table 5. The process of dictionary pair requires acquiring a database from external HR-LR images. In the future, we will further improve the proposed method in the following ways. (1) Database Construction: We will construct an image pyramid by interpolating the inputted image across different scales, and then database can be extracted from image pyramid. (2) Dictionary Training: We will use an optimal approach to train a sparse dictionary so that the running time can be reduced.

Table 5. Running time on the Middlebury Stereo database with scaling factor 4.

Time (s)	Bowling	Artl	Aloe	Cones	Indian	Venus	Warrior	Tsukuba	Hand	Dove
GR	2.7	0.7	1.6	1.3	5.2	1.3	3.9	0.8	5.3	5.5
ANR	3.2	0.8	1.8	1.5	5.8	1.4	4.8	1.0	6.1	6.4
NE + LS	9.1	2.5	4.9	4.2	17.0	4.1	13.1	2.4	17.3	18.5
NE + NNLS	56.3	15.1	29.6	27.0	104.8	26.0	66.6	10.4	104.3	110.4
NE + LLE	11.7	3.0	6.6	5.4	21.5	5.3	15.7	2.9	21.8	21.9
AANR	3.4	0.9	2.0	1.6	6.4	1.6	4.5	0.9	6.4	6.2
CNN	6.7	2.1	4.8	3.5	12.9	3.6	10.0	2.3	13.7	12.9
Zeyde	5.3	1.4	3.1	2.7	9.9	2.4	6.7	1.3	9.9	10.6
Yang	986.2	260.4	726.9	541.5	1937.3	485.1	864.7	177.5	1786.7	1419.8
Xie	594.9	517.4	864.6	608.9	913.7	141.3	759.9	373.8	469.5	417.7
Ours	92.1	74.7	105.0	101.3	177.4	56.9	115.6	51.7	76.4	75.3

Acknowledgments: This work is supported by the National Natural Science Foundation of China (Nos. 61751203, 61603066, and 61300015), the Program for Dalian High-level Talent's Innovation (2015R088), Program for Changjiang Scholars and Innovative Research Team in University (No. IRT_15R07), and Program for Liaoning Distinguished Professor.

Author Contributions: Ruyi Wang wrote this manuscript. Dongsheng Zhou, Jian Lu and Qiang Zhang contributed to the writing, direction, and content, and revised the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pan, H.; Guan, T.; Luo, Y.; Duan, L.; Tian, Y.; Yi, L.; Zhao, Y.; Yu, J. Dense 3D reconstruction combining depth and RGB inf-ormation. *Neurocomputing* **2016**, *75*, 644–651. [CrossRef]
2. Henry, P.; Krainin, M.; Herbst, E.; Ren, X.; Fox, D. Color-D Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments. In Proceedings of the 12th International Symposium on Experimental Robotics, Delhi, India, 18–21 December 2010; Springer: Berlin/Heidelberg, Germany, 2014; pp. 647–663.
3. Lee, K.; Hou, X. Augmented Reality Using 3D Depth Sensor and 3D Projection. U.S. Patent Appl. 15/172,723, 3 June 2016.
4. Correa, D.S.O.; Sciotti, D.F.; Prado, M.G.; Sales, D.O.; Wolf, D.F.; Osorio, F.S. Mobile robots navigation in indoor environments using kinect sensor. In Proceedings of the 2012 Second Brazilian Conference on Critical Embedded Systems, Campinas, Brazil, 20–25 May 2012; pp. 36–41.
5. Zhen-Liang, N.; Zhen, S.; Chao, G.; Xiong, G.; Nyberg, T.; Shang, X.; Li, S.; Wang, Y. The application of the depth camera in the social manufacturing: A review. In Proceedings of the IEEE International Conference on Service Operations and Logistics, Beijing, China, 10–12 July 2016; pp. 66–70.
6. Zeyde, R.; Elad, M.; Protter, M. On single image scale-up using sparse-representations. In Proceedings of the International Conference on Curves and Surfaces, Avignon, France, 24–30 June 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 711–730.
7. Timofte, R.; Smet, V.D.; Gool, L.V. A+: Adjusted Anchored Neighborhood Regression for Fast Super-Resolution. In Proceedings of the Asian Conference on Computer Vision, Singapore, 1–5 November 2014; Springer: Cham, Switzerland, 2014; pp. 111–126.
8. Xie, J.; Feris, R.S.; Sun, M.T. Edge-guided single depth image super resolution. *IEEE Trans. Image Process.* **2016**, *25*, 428–438. [CrossRef] [PubMed]
9. Prajapati, A.; Naik, S.; Mehta, S. Evaluation of Different Image Interpolation Algorithms. *Int. J. Comput. Appl.* **2012**, *58*, 466–476. [CrossRef]
10. Pang, Z.; Dai, H.; Tan, H.; Chan, D. An improved low-cost adaptive bilinear image interpolation algorithm. In Proceedings of the 2nd International Conference on Green Communications and Networks, Chongqing, China, 14–16 December 2012; Springer: Berlin/Heidelberg, Germany, 2013; pp. 691–699.

11. Ning, L.; Luo, K. An Interpolation Based on Cubic Interpolation Algorithm. In Proceedings of the International Conference Information Computing and Automation, Chengdu, China, 20–22 December 2007; pp. 1542–1545.
12. Tomasi, C.; Manduchi, R. Bilateral filtering for gray and color images. In Proceedings of the Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271), Bombay, India, 7 January 1998; pp. 839–846.
13. Kopf, J.; Cohen, M.F.; Lischinski, D.; Uyttendaele, M. Joint Bilateral Upsampling. *ACM Trans. Graph.* **2007**, *26*, 5–9. [CrossRef]
14. Schuon, S.; Theobalt, C.; Davis, J.; Thrun, S. LidarBoost: Depth superresolution for ToF 3D shape scanning. In Proceedings of the 2009 the 22nd International Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 343–350.
15. Rajagopalan, A.N.; Bhavsar, A.; Wallhoff, F.; Rigoll, G. Resolution enhancement of pmd range maps. In Proceedings of the Joint Pattern Recognition Symposium, Munich, Germany, 10–13 June 2008; Springer: Berlin/Heidelberg, Germany, 2008; pp. 304–313.
16. Al Ismaeil, K.; Aouada, D.; Mirbach, B.; Ottersten, B. Dynamic super resolution of depth sequences with non-rigid motions. In Proceedings of the 2013 20th IEEE International Conference on Image, Melbourne, Australia, 15–18 September 2013; pp. 660–664.
17. Gevrekci, M.; Pakin, K. Depth map super resolution. In Proceedings of the 2011 18th IEEE International Conference on Image, Brussels, Belgium, 11–14 September 2011; pp. 3449–3452.
18. Ferstl, D.; Reinbacher, C.; Ranftl, R.; Ruether, M.; Bischof, H. Image guided depth upsampling using anisotropic total generalized variation. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 993–1000.
19. Yang, Q.; Yang, R.; Davis, J.; Nister, D. Spatial-depth super resolution for range images. In Proceedings of the 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.
20. Lo, K.H.; Wang, Y.F.; Hua, K.L. Edge-Preserving Depth Map Upsampling by Joint Trilateral Filter. *IEEE Trans. Cybern.* **2017**, *13*, 1–14. [CrossRef] [PubMed]
21. Yang, J.; Wright, J.; Huang, T.S.; Ma, Y. Image super-resolution via sparse representation. *IEEE Trans. Image Process.* **2010**, *19*, 2861–2873. [CrossRef] [PubMed]
22. Aharon, M.; Elad, M.; Bruckstein, A. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Trans. Signal Process.* **2006**, *54*, 4311–4322. [CrossRef]
23. Tropp, J.A.; Gilbert, A.C. Signal Recovery from Random Measurements Via Orthogonal Matching Pursuit. *IEEE Trans. Inf. Theory* **2007**, *53*, 4655–4666. [CrossRef]
24. Xie, J.; Chou, C.C.; Feris, R.; Sun, M.-T. Single depth image super resolution and denoising via coupled dictionary learning with local constraints and shock filtering. In Proceedings of the 2014 IEEE International Conference on Multimedia and Expo (ICME), Chengdu, China, 14–18 July 2014; pp. 1–6.
25. Kim, J.; Kwon Lee, J.; Mu Lee, K. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654.
26. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556v6.
27. Gilboa, G.; Sochen, N.; Zeevi, Y.Y. Image enhancement and denoising by complex diffusion processes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 1020–1036. [CrossRef] [PubMed]
28. Böttcher, A.; Wenzel, D. The Frobenius norm and the commutator. *Linear Algebra Appl.* **2008**, *8*, 1864–1885. [CrossRef]
29. MATLAB. 2016. Available online: www.mathworks.com/products/matlab (accessed on 12 December 2017).
30. Baker, S.; Scharstein, D.; Lewis, J.P.; Roth, S.; Black, M.J.; Szeliski, R. A database and evaluation methodology for optical flow. *Int. J. Comput. Vis.* **2011**, *92*, 1–31. [CrossRef]
31. Scharstein, D.; Hirschmüller, H.; Kitajima, Y.; Krathwohl, G.; Nešić, N.; Wang, X.; Westling, P. High-resolution stereo datasets with Subpixel-accurate ground truth. In Proceedings of the German Conference on Pattern Recognition, Münster, Germany, 2–5 September 2014; Springer: Cham, Switzerland, 2014; pp. 31–42.

32. Yang, J.; Wright, J.; Huang, T.; Ma, Y. Image super-resolution as sparse representation of raw image patches. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
33. Chang, H.; Yeung, D.Y.; Xiong, Y. Super-resolution through neighbor embedding. In Proceedings of the 2004 IEEE Computer Socity Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 27 June–2 July 2004; pp. 1275–1282.
34. Chan, T.M.; Zhang, J.; Pu, J.; Huang, H. Neighbor embedding based super-resolution algorithm through edge detection and feature selection. *Pattern Recognit. Lett.* **2009**, *30*, 494–502. [CrossRef]
35. Bevilacqua, M.; Roumy, A.; Guillemot, C.; Alberi Morel, M.-L. Neighbor embedding based single-image super-resolution using Semi-Nonnegative Matrix Factorization. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012; pp. 1289–1292.
36. Timofte, R.; De Smet, V.; Van Gool, L. Anchored neighborhood regression for fast example-based super-resolution. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 1920–1927.
37. Charstein, D.; Szeliski, R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vis.* **2002**, *47*, 7–42. [CrossRef]
38. Hirschmüller, H.; Scharstein, D. Evaluation of cost functions for stereo matching. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.