Similarity-Aware Patchwork Assembly for Depth Image Super-Resolution

Jing Li Zhichao Lu Gang Zeng Rui Gan Hongbin Zha
Key Laboratory on Machine Perception, Peking University

Abstract

This paper describes a patchwork assembly algorithm for depth image super-resolution. An input low resolution depth image is disassembled into parts by matching similar regions on a set of high resolution training images, and a super-resolution image is then assembled using these corresponding matched counterparts. We convert the super-resolution problem into a Markov Random Field (MRF) labeling problem, and propose a unified formulation embedding (1) the consistency between the resolution enhanced image and the original input, (2) the similarity of disassembled parts with the corresponding regions on training images, (3) the depth smoothness in local neighborhoods, (4) the additional geometric constraints from self-similar structures in the scene, and (5) the boundary coincidence between the resolution enhanced depth image and an optional aligned high resolution intensity image. Experimental results on both synthetic and real-world data demonstrate that the proposed algorithm is capable of recovering high quality depth images with $\times 4$ resolution enhancement along each coordinate direction, and that it outperforms state-of-the-arts [14] in both qualitative and quantitative evaluations.

1. Introduction

Depth images become more and more popular and have been extensively used in modern applications such as interactive free-viewpoint video [13], semantic scene analysis [10], and human pose recognition [18], thanks to the widespread 3D imaging hardwares like Kinect and TOF cameras. The upper limit on the precision and spatial resolution of sensing devices affects their application performance, especially when the scene to be captured is in a large scale where close and fine scans are cumbersome and labor intensive, and also when the capturing devices have to be placed in a long distance.

Depth image super-resolution is intrinsically an ill-posed problem. Most of prior arts either exploit additional data from the scene, such as multiple depth images from nearby viewpoints [17, 1] and an aligned high resolution image [23, 15], or utilize prior shape knowledge, such as self-similarity structures [11] and a training database for rectangular patches [14]. To the best of our knowledge, however, the object composition has never been taken into consideration.

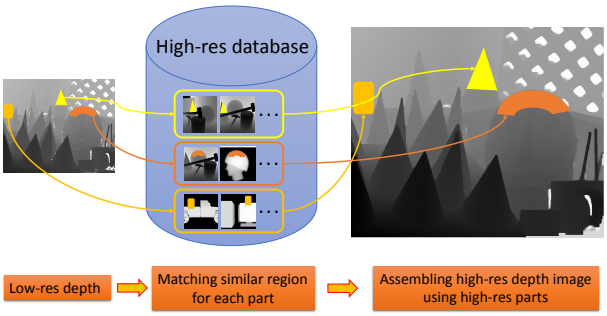

Figure 1. An overview of the patchwork assembly algorithm.

Two key observations are that many types of natural and man-made objects (*e.g.* buildings, cars, human bodies, *etc.*) can be disassembled into similar parts, and that by using these parts one can assemble a plausible copy of a different object of the same type. These motivate us to design a patchwork [26, 25] assembly algorithm for depth image super-resolution. We first collect high resolution depth images from different scenes and objects to build a training database. The input low resolution image is then disassembled into regions representing object parts which are matched with those on the training images. We finally assemble the super-resolution image using these corresponding matched regions from the high resolution training images. An overview of the patchwork assembly algorithm is shown in Fig. 1.

Segmenting an image into semantic parts itself is a traditional hard problem, which tends to be more complicated than the problem at hand. Here we are interested by a different definition of object part: an object region that commonly exists on several objects of a same type, or a corresponding depth image region that repeats on different images. Thus instead of explicitly extracting object parts based on their appearances from a single input image, we disassemble the input image into parts by detecting regions with similar appearance from the training images.

Another fact is the interplay between disassembling and assembling steps: the similarity based disassembling could provide basic materials for assembling, while a quality-aware assembling provides useful feedbacks for disassembling. That is to say, the two steps should be considered within a unified optimization framework. We formulate this patchwork assembly problem as a MRF labeling problem, namely that we aim to optimally segment the input low resolution depth image into several regions with different labels which correspond to a high resolution counterpart on training images. It allows us to jointly optimize the consistency between the resolution enhanced image and the original input, the similarity of disassembled parts with the corresponding regions on training images, the depth smoothness in local neighborhoods, and the boundary coincidence between the resolution enhanced depth image and an optional aligned high resolution intensity image.

Moreover, the scene may contain self-similar structures providing an additional clue [24] to further improve the enhanced depth image [11]. We address the self-similarity by adding soft constraints with regard to label and shape consistency between repeating structures in the MRF based optimization process.

1.1. Related Work

Due to their simplicity, classic upsampling techniques like nearest neighbor, bilinear, or bicubic interpolations are most convenient treatments for resolution enhancement. Without additional knowledge about the scene, they often produce jagged steps or blurry boundaries, which prevents their usages from demanding applications. Fattal [4] proposed to generate sharp edges based on a statistical edge dependency relating certain edge features of two different resolutions. Yang *et al*. [22] chose to selectively adjust the local gradients to restore antialiased edges. The bilateral filtering [20] smoothes images by means of a nonlinear combination of nearby depth values, which is prone to error at depth edges. The guided image filtering [9] considers the content of a guidance image, and performs an edge-preserving smoothing operation. These techniques either are difficult to extend in areas of texture or tend to smooth out sharp geometric details.

With additional depth images captured at nearby locations, Schuon *et al*. [17] developed an optimization framework embedding data fidelity and geometry prior in order to produce high quality depth maps. Cui *et al*. [1] used a probabilistic scan alignment approach to fuse noisy scans to achieve 3D shapes with high quality. Rajagopalan *et al*. [16] proposed a super-resolution method through induced camera motion and used MRF to fuse multiple low resolution image with an edge-adaptive prior. Hahne and Alexa [8] combined depth images taken with differing integration times to decrease the noise in each image. Izadi *et al*. [12] designed a GPU-based pipeline to track the 3D pose of the sensor and reconstruct 3D models by merging low resolution images from Kinect in real-time. These multiple-image-based methods can only be used to capture static scenes, and moreover the quality of their results are sensitive to the errors in estimating viewpoint locations.

For intensity image super-resolution, most related to ours, Freeman *et al*. [5, 6] converted the super-resolution problem into an MRF multi-class labeling problem, but the formulation is based on the rectangular patch and is hard to capture objects and parts. Glasner *et al*. [7] combined multi-image-based methods with example-based methods and used redundant information from repeating patches to reconstruct a high resolution image. Sun *et al*. [19] computed over-segmentation of the image based on texture similarity, extracted descriptor from each region and compared texture from an external database. Yang *et al*. [21] employed sparse representation to retrieve high resolution image based on sparse linear combinations from an atom dictionary learned from corresponding low and high resolution patches. Compared with intensity images, depth images generally contain more noise with non-Gaussian distribution and are of lower resolution. The descriptors for image intensity and texture are also different from those for geometric entities.

With an aligned intensity image, Diebel and Thrun [2] proposed an MRF formulation based on the observation that discontinuities in range and intensity tend to co-align. Park *et al*. [15] further extended the MRF framework with a non-local means term to preserve thin structures in the image. Yang *et al*. [23] employed a cross bilateral filter to iteratively refine the input low-resolution range image, and the high resolution intensity image is used to build a cost volume. Dolson *et al*. [3] also used joint bilateral filter to upsample range data, but their method relies on a monocular image sequence and a stream of sparse range measurements to produces a dense, high-resolution dynamic depth map of the scene. Stereo cameras are used in [27] to improve the depth map, which leads to better performance than a single intensity image. In our algorithm, we treat the aligned intensity image as optional since the image is either sometimes unavailable or needs additional concerns about registration and synchronization. In the cases when an aligned intensity image is available, we combine it as an additional constraint in our MRF formulation.

The proposed approach also differs from state-of-the-arts with a similar training database [14] in several aspects: 1) Ours employs an assembly based strategy, matches regions on the input depth image with those on the training images, and propagates the corresponding regions along their neighborhoods to form matched object parts, while [14] only considers individual patches and does not match their neighbors to obtain object parts; 2) The proposed algorithm achieves

(a) Training images by rendering (b) Training images from [14]

Figure 2. Example of the training images from different sources.

pixel-wise accuracy in high-resolution image, while the prior art [14] computes a unique correspondence for each rectangular patch (*i.e.* 12×12) on high resolution image; 3) Our energy formulation naturally encodes similar part detection, patchwork assembly, self-similarity enhancement within a same framework. To the best of our knowledge, our algorithm is the first depth image super resolution algorithm that employs a patchwork assembly strategy and formulates the problem as a pixel-wise MRF labeling problem.

2. Problem Statement

In order to make the depth image super resolution problem well defined, we employ a training database $\{D_k^H, k = 1, 2, \cdots, K\}$ (see Fig. 2) as prior shape knowledge similar to [14], where D_k^H is a high-resolution depth image and can be obtained by either projecting a synthetic model or collecting with modern high-end laser scanners.

Given an input low-resolution depth image D^L, the proposed patchwork assembly algorithm aims to disassemble D^L into parts $D^L = \bigcup_l R_l^L$, such that each part R_l^L is similar to a certain high-resolution region R_l^H on a training image D_k^H in the database, and assembles the resolution enhanced image $D^H = \bigcup_l R_l^H$ using these corresponding regions $\{R_l^H\}$.

Thus we formulate the depth image super resolution problem as a multiple label image segmentation problem. Let $L(\mathbf{x})$ denote the label of a pixel $\mathbf{x}$ on the high-resolution depth image, and we use labels to distinguish different regions, such that R_l^H is mapped as

$$R_l^H(\mathbf{x}) = D^H(\mathbf{x}), \text{ with } L(\mathbf{x}) = l. \tag{1}$$

More precisely, a label presents a specific index indicating the location of the corresponding component from training images for assembling, together with the transformation of this component, namely that $l = (k, \mathbf{d}, s, t)$ with k being the index of training image in the database, $\mathbf{d}$ being the coordinate displacement between a corresponding region on the input depth image and its counterpart on the k-th training image, and s and t being the scaling and translation of the corresponding component along the depth direction:

$$R_l^H(\mathbf{x}) = s \cdot D_k^H(\mathbf{x} + \mathbf{d}) + t, \text{ given } l = (k, \mathbf{d}, s, t), \tag{2}$$

and hence if given a label assignment L we can compute the high-resolution depth image D^H as

$$D^H(\mathbf{x}) = s \cdot D_k^H(\mathbf{x} + \mathbf{d}) + t, \text{ if } L(\mathbf{x}) = l = (k, \mathbf{d}, s, t). \tag{3}$$

We thus convert the depth image super resolution problem into a multiple label image segmentation problem.

3. Energy Minimization

We propose to solve the above mentioned MRF labeling problem by minimizing the following energy functional

$$E_{total} = E_{data} + \lambda_l \cdot E_{label} + \lambda_d \cdot E_{depth} + \lambda_s \cdot E_{similar}, \tag{4}$$

where E_{data} penalises the dissimilarity between the input depth image D^L and the resolution enhanced image D^H, E_{label} penalises the label inconsistency among neighboring pixels, E_{depth} penalises the depth discontinuous boundaries, and $E_{similar}$ penalises the label and depth inconsistency between self-similar structures. $\lambda_l, \lambda_d, \lambda_s$ are weighting parameters controlling the importance of each term.

Data Term E_{data}: The data term ensures that the assembled high-resolution depth image D^H resembles the low-resolution input D^L. Given a label assignment $L(\mathbf{x}) = l = (k, \mathbf{d}, s, t)$, D^H can be calculated with Eqn. (3). Let $\mathbf{u}$ be a function that maps the coordinate $\mathbf{x}$ in the high-resolution image to a coordinate $\mathbf{u}(\mathbf{x})$ in the low-resolution image with the nearest neighbor (NN) interpolation. The NN interpolation is proven to be superior by [17], although the reliablity of this interpolation decreases with the increasing geometric details. To address this problem, we introduce a weighting scheme that characterises the interpolation reliablity:

$$\begin{aligned} E_{data} &= \sum_{\mathbf{x}} e^{-\alpha \cdot \varphi(\mathbf{x})} \cdot |D^H(\mathbf{x}) - D^L(\mathbf{u}(\mathbf{x}))|, \\ \text{with } \varphi(\mathbf{x}) &= \sum_{\mathbf{v} \in N_{\mathbf{u}(\mathbf{x})}} |D^L(\mathbf{u}(\mathbf{x})) - D^L(\mathbf{v})|, \end{aligned} \tag{5}$$

where $N_{\mathbf{u}}$ is a local neighborhood of $\mathbf{u}$, $\varphi(\mathbf{x})$ measures the magnitude of local geometric details around $\mathbf{x}$, and α is a controlling parameter for the negative exponential function.

Label Coherency Term E_{label}: The label coherency term emphasizes the label consistency between neighboring pixels on the high-resolution image. It encourages disassembling the depth image into a smaller number of larger parts, extracting similar components with a larger area of supporting regions, focusing more on comparing more global structures and ignoring tiny details, and approaching component- and module-level assembling. It naturally allows adding geometric details from training images based on rough structure similarity, even when local evidences are not efficiently captured by the low-resolution image and thus are insufficient for a data-driven modeling.

Since the interpolation reliability of low-resolution image becomes lower when detailed geometric changes occur, we should exploit more global structure to recover the details and thus we have:

$$E_{label} = \sum_{\mathbf{x}} \sum_{\mathbf{y}\in N_{\mathbf{x}}} \sigma(\varphi(\mathbf{x},\mathbf{y})) \cdot \delta(L(\mathbf{x}) \neq L(\mathbf{y}))$$
$$\text{with } \varphi(\mathbf{x},\mathbf{y}) = \frac{\varphi(\mathbf{x})+\varphi(\mathbf{y})}{2}, \quad (6)$$

where $\sigma(w) = \frac{1}{1+e^{-\alpha\cdot w}}$ is the sigmoid function and $\delta(true) = 1, \delta(false) = 0$ is the delta function. $\varphi(\mathbf{x},\mathbf{y})$ estimates the magnitude of geometric changes between neighboring $\mathbf{x}$ and $\mathbf{y}$ with $\varphi(\mathbf{x})$ defined in Eqn. (5). $N_{\mathbf{x}}$ denotes the local neighborhood of $\mathbf{x}$.

Depth Smoothness Term E_{depth}: The depth smoothness term ensures smooth transition in depth and penalizes sharp contrast with high curvatures in local neighborhoods. We also make this term sensitive to the estimated magnitude of local geometric changes $\varphi(\mathbf{x},\mathbf{y})$ as defined in Eqn. (6):

$$E_{depth} = \sum_{\mathbf{x}} \sum_{\mathbf{y}\in N_{\mathbf{x}}} e^{-\alpha\cdot\varphi(\mathbf{x},\mathbf{y})} \cdot |D^H(\mathbf{x}) - D^H(\mathbf{y})|, \quad (7)$$

where again we use $N_{\mathbf{x}}$ to denote the local neighborhood of $\mathbf{x}$, and α is a controlling parameter for the negative exponential function.

Self-Similarity Term $E_{similar}$: The self-similarity term enforces intrinsic constraints among similar parts in the scene (see Fig. 3). Let $(\mathbf{x}_i,\mathbf{x}_j)$ denote a pair of corresponding pixels in two similar parts on the depth image, and let $L(\mathbf{x}_i) = (k_i,\mathbf{d}_i,s_i,t_i)$ and $L(\mathbf{x}_j) = (k_j,\mathbf{d}_j,s_j,t_j)$. The depth values of $\mathbf{x}_i$ and $\mathbf{x}_j$ may be different since the similar parts are often distributed in different scales and depths, but their shapes should be disassembled into similar components and thus their labels are correlated. More precisely, $\mathbf{x}_i$ and $\mathbf{x}_j$ should be mapped into a unique location of the same component on a training image, which implies:

$$(k_i = k_j)\ \&\ (\mathbf{x}_i + \mathbf{d}_i = \mathbf{x}_j + \mathbf{d}_j), \quad (8)$$

and we thus define $E_{similar}$ as:

$$E_{similar} = \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in S} [s(\mathbf{x}_i,\mathbf{x}_j) \cdot \sigma(\tfrac{\varphi(\mathbf{x}_i)+\varphi(\mathbf{x}_j)}{2}) \cdot \delta(k_i \neq k_j \mid \mathbf{x}_i + \mathbf{d}_i \neq \mathbf{x}_j + \mathbf{d}_j)], \quad (9)$$

where σ and δ are the sigmoid and delta functions as in Eqn. (6). $\varphi(\mathbf{x})$ measures the magnitude of local geometric details around $\mathbf{x}$ as in Eqn. (5). S is the set of all corresponding pixel pairs in similar parts. $s(\mathbf{x}_i,\mathbf{x}_j)$ measures the similarity of the pair, like the cross-correlation scores between the depth neighborhoods of the two pixel. We give the design details of this function later in Eqn. (13).

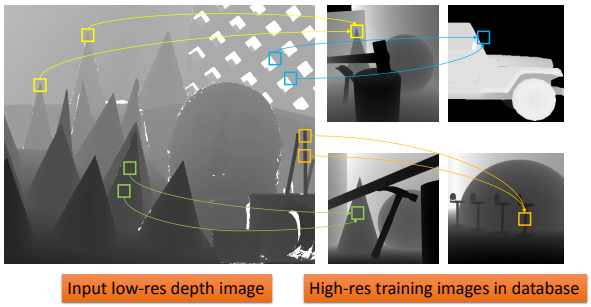

Figure 3. An example of self-similar structures in scene and their counterpart in training images.

3.1. With Additional Intensity Image

An additional high resolution intensity image is often available from modern scanners, such as Kinect and TOF cameras. Prior arts [23, 15] anisotropically diffuse depth values based on the boundaries shape on intensity image. We fuse the optional intensity image I by including it in the energy functional.

It is very likely that the variation in geometry and appearance are concurrent. The low resolution depth image may fail to capture some details but the high resolution intensity image provides cues. We adapt the magnitude estimation function of geometric details, $\varphi(\mathbf{x})$ and $\varphi(\mathbf{x},\mathbf{y})$ in Eqn. (5) and (6), by encoding both depth and intensity information:

$$\varphi_c(\mathbf{x}) = \sum_{\mathbf{v}\in N_{\mathbf{u}(\mathbf{x})}} |D^L(\mathbf{u}(\mathbf{x})) - D^L(\mathbf{v})| + \gamma_1 \cdot \sum_{\mathbf{y}\in N_{\mathbf{x}}} \|I(\mathbf{x}) - I(\mathbf{y})\|_2, \quad (10)$$

$$\varphi_c(\mathbf{x},\mathbf{y}) = \frac{\varphi(\mathbf{x})+\varphi(\mathbf{y})}{2} + \gamma_2 \cdot \|I(\mathbf{x}) - I(\mathbf{y})\|_2, \quad (11)$$

where we assume the high resolution intensity image I and depth image D^H share the same coordinate system. γ_1,γ_2 are used to control the importance of intensity information.

Replacing $\varphi(\mathbf{x})$ with newly defined $\varphi_c(\mathbf{x})$ in E_{data}, $E_{similar}$ in Eqns. (5,9) and $\varphi(\mathbf{x},\mathbf{y})$ with $\varphi_c(\mathbf{x},\mathbf{y})$ in E_{label}, E_{depth} in Eqns. (6,7) results in the fused energy terms.

3.2. Optimization Algorithm

With the patchwork assembly pipeline we have converted the depth image super resolution problem into a MRF labeling problem, and designed the energy functional in Eqn. (4) to fuse the criteria and constraints in order to describe the optimization goal. Different from classic multiple label image segmentation, the label $L(\mathbf{x}) = l = (k,\mathbf{d},s,t)$ in the proposed formulation is in a much higher dimension space, and even with discretization, the complexity of such a problem is unaffordable for modern computing hardwares.

Candidate Label Calculation We simplify the problem by limiting the labels $L(\mathbf{x})$ with a smaller number of possible candidates. The strategy is to compare the low-resolution local neighborhood of $\mathbf{x}$ in the input image with high resolution local neighborhoods in all training images in the database. The most similar candidates with top scores are kept and the parameters $l = (k, \mathbf{d}, s, t)$ for each candidate label are calculated based on the transformation of the corresponding locations. This patch-based similarity detection is inspired by [14], where the input is treated as a collection of non-overlapping patches of rectangular shape, and the patches are then compared with a patch database. Different from their method, we aims to calculate the best matching candidates for every pixel locations, and moreover we must guarantee that neighboring pixels share a portion in their labels such that they can be merged to one label segment and thus regarded as in a same object part in the patchwork assembly pipeline.

For each pixel location $\mathbf{u}$ in the input low resolution image, we calculate the SSD (Sum of Squared Differences) distance between the normalized low-resolution depth patch centered at $\mathbf{u}$ and the normalized downsampled high resolution patches from all training depth images in the database. We select the N most similar patches, and for each of which we calculate the index k of the depth image in the database, the displacement $\mathbf{d}$ of the coordinates between the patch center and $\mathbf{u}$, and the relative scaling and translation s and t between this patch and the patch centered at $\mathbf{u}$. The above parameter assignment forms a possible candidate label $l_{\mathbf{u}} = (k_{\mathbf{u}}, \mathbf{d}_{\mathbf{u}}, s_{\mathbf{u}}, t_{\mathbf{u}})$. Hence the N most similar patches provide N candidate labels $\mathcal{L}_{\mathbf{u}} = \{l_{\mathbf{u}}\}$.

Here the low resolution patch is not upsampled with a deterministic interpolation method for comparison at the upsampled scale due to the introduction of noise by such an interpolation. Similar to [14], the high resolution training images in the database are prefilterred and downsampled to make them the same size and comparable to low resolution patches on the input image.

Given a pixel location $\mathbf{x}$ on the high-resolution image, let $\{\mathbf{u}_k\}$ denote the M closest pixel locations on the input low-resolution image. We define $N \times M$ candidate labels $\mathcal{L}_{\mathbf{x}}$ for $\mathbf{x}$ as the following form in order to guarantee that neighboring pixels share a large portion in their labels:

$$\mathcal{L}_{\mathbf{x}} = \bigcup_{k=1}^{M} \mathcal{L}_{\mathbf{u}_k}. \tag{12}$$

Self-Similarity Detection We search for similar patches for every pair of locations $(\mathbf{u}_i, \mathbf{u}_j)$ on low resolution image. Let $SSD_{\mathbf{u}_i,\mathbf{u}_j}$ denotes the Sum of Squared Differences distance between the two normalized depth patches centered at $\mathbf{u}_i$ and $\mathbf{u}_j$. We select patch pairs with sufficient small SSD scores and filter out planar patches. We then extract and merge pixel correspondences to form the self-similarity set. The self-similarity score of a pixel pair is defined as

$$s(\mathbf{x}_i, \mathbf{x}_j) = e^{-\beta \cdot SSD_{\mathbf{u}(\mathbf{x}_i),\mathbf{u}(\mathbf{x}_j)}}, \tag{13}$$

where $\mathbf{u}(\mathbf{x})$ is the nearest neighbor coordinate mapping from the high resolution to the low resolution. β is a controlling parameter for the negative exponential function.

Graph-cuts Optimization We choose to use graph-cuts to minimize our energy functional in Eqn. (4), since max-flow-based optimizations are proven to achieve a global minimum solution while their complexities remain in the order of polynomial time in terms of the number of the underlying graph nodes and edges. We use α-expansion to solve the converted multiple label segmentation problem.

4. Experimental Results and Discussion

4.1. Implementation Details

Unless otherwise indicated, all experiments were run with the same parameters and with the same training data. α in Eqn. (5) is set to 5.0. γ_1 and γ_2 in Eqns. (10). are set to 30.0 and 3.0 respectively. β in Eqn. (13) is set to 3.0. The algorithm is implemented by C++ and runs on a 64-bit computer with Intel 3.40GHz CPU and 8GB memory.

Training Database Preparation We generate the high resolution training images by rendering 50 different 3D models of various types to the camera coordinates of different viewpoints and viewing directions. For each model we produce 8 views by rotating around z axis. We also include 58 synthesized images produced by [14]. Some examples of training image can be found in Fig. 2. Similar to [14], we locate and prune redundant planar patches from training images by detecting depth discontinuities using an edge detector. The amount of remaining non-planar patches used in our experiments is around 10 million.

4.2. Results and Discussion

We evaluated the proposed algorithm with data from various sources, including structured light, TOF cameras, Microsoft Kinect and synthesized data. We show the resolution enhanced image for a single depth image and compare it with the nearest neighbor (NN) interpolation, bilateral filtering [20], and patch-based method [14]. Thanks to the patchwork assembly pipeline, our method is capable of recovering fine details, including sharp edge boundaries and detailed geometric variations. Moreover, these prior arts favor TOF sensors over Kinect due to the interference from missing regions around depth boundaries on Kinect images, while our method is robust against such effect.

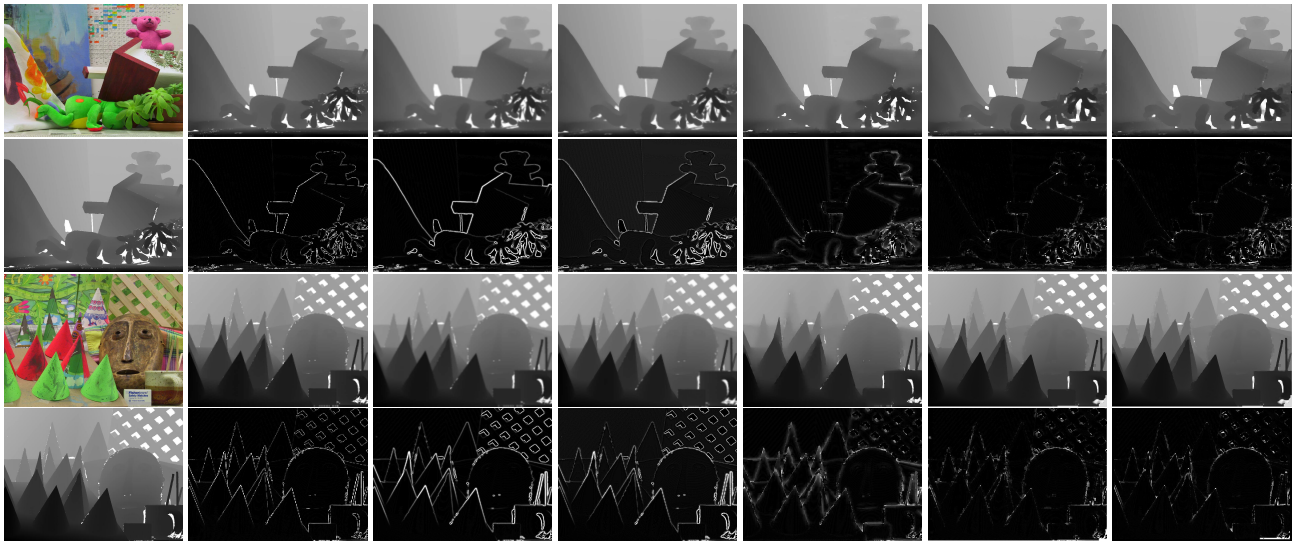

Figure 5. $\times 4$ resolution enhancement using Middlebury data of scene "teddy" and "cones". The first column shows color image and the corresponding groundtruth image. Columns $2-7$ are results using nearest neighbor, bilateral filtering [20], guided image filtering [9], patch-based method [14] and our method (without and with intensity image). For each scene, the first row shows the upsampling results, and the second row shows the corresponding error map compared with the groundtruth.

In cases when the additional intensity image is available, we also compare our results with guided image filtering [9], which performs an edge-preserving smoothing operator like popular bilateral filter, but has better behavior near the edges. In our experiments, we use the additional intensity image as guidance when applying guided image filtering [9] on the input low resolution image.

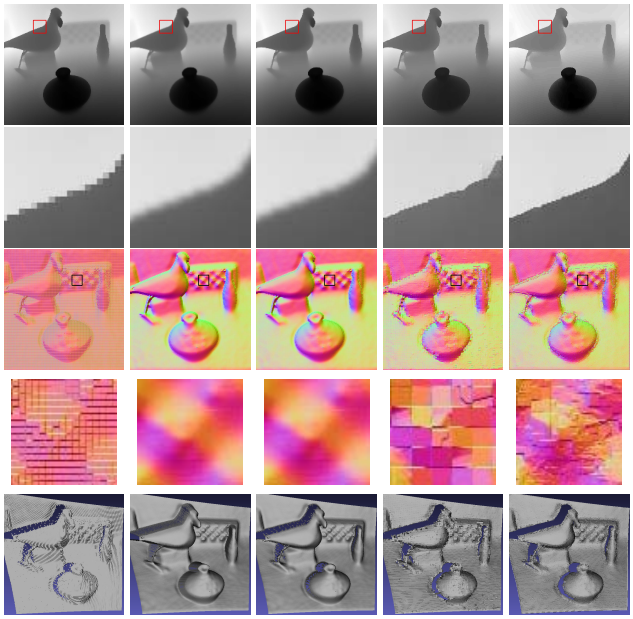

Figure 4. $\times 4$ resolution enhancement using a TOF image. The first row shows results of the enhanced depth maps, with the region within red square is zoomed and shown in the second row. The third row shows the corresponding normal map and the forth row shows the zoomed region in the black square. The last row shows 3D model of the reconstruction. We compare our results (column 5) with nearest neighbor (column 1), bilateral filtering [20] (column 2), guided image filtering [9] (column 3) and patch-based method [14] (column 4).

Qualitative Comparison Fig. 4 shows a $\times 4$ super resolution result using TOF data of size 800×800. The nearest neighbor interpolation produces jagged steps on boundaries with sharp surface normal variations. The bilateral filtering and guided image filtering both produce over-smoothed surface normals with blurry boundaries. Patch-based method recovers sharp details with some artifacts due to the patch-based matching strategy (see the enlarged regions in Fig. 4). The surface normal also jumps across the patch boundaries. Our patchwork assembly method recovers sharp and accurate boundary geometries with smooth transition of surface normal at pixel-level. The running time for this image is 270 seconds.

Fig. 5 shows two $\times 4$ upsampling results from the Middluebury dataset. We show the error map between the enhanced image and the groundtruth, and our method produces least error compared with other methods. The images are of size 750×900, and it takes 528 seconds to compute the "cones" image and 439 seconds to compute the "teddy" image.

Fig. 6 shows $\times 4$ upsampling results for real scenes captured using Kinect. The Kinect depth image is more challenging because it contains more error with missing regions around depth discontinuous boundaries. Our method preserves sharp depth boundaries and small geometric struc-

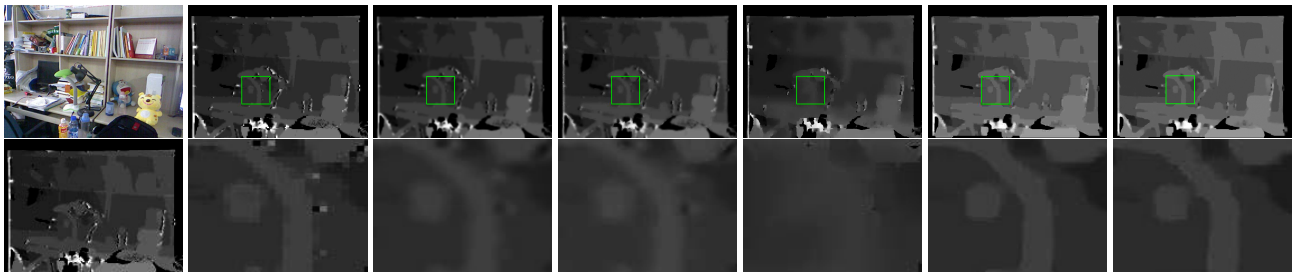

Figure 6. $\times 4$ upsampling using Kinect data from a real scene. The first column shows input color and depth images. Columns $2-7$ represent results using nearest neighbor, bilateral filtering [20], guided image filtering [9], patch-based method [14] and our method (without and with intensity image). The first row corresponds to upsampling results, and the second row show the zoomed region within the green box of the first row.

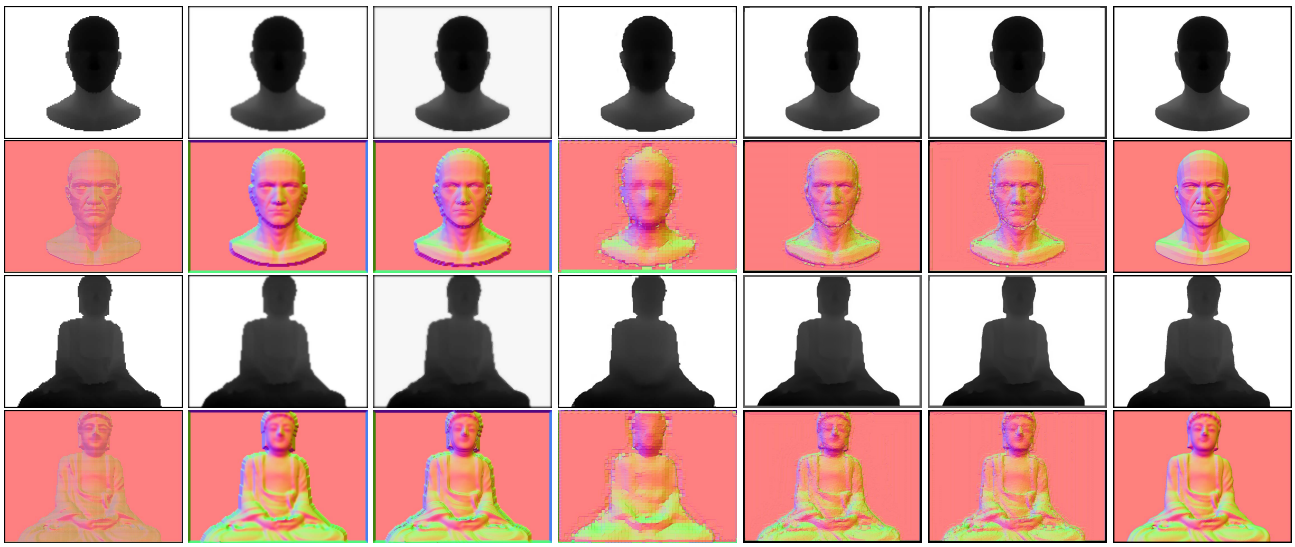

Figure 7. $\times 4$ resolution enhancement using synthesized data without intensity images. The columns represent results using nearest neighbor, bilateral filtering [20], guided image filtering [9], patch-based [14] result and our result. The first and third rows show upsampling results, and the second and fourth rows show the corresponding normal map. The normal map also shows the difference between the proposed patchwork assembly algorithm and the patch-based method in recovering rich geometric details.

tures, while prior arts either smooth out depth boundaries or neglect such thin structures. This Kinect image is of size 640×480, and it takes about 350 seconds to finish the computation.

Fig. 7 shows two results from synthesized data. The corresponding normal maps are calculated directly from the depth images. Results using bilateral filtering and guided image filtering produces large errors at sharp depth edges, while results from patch-based method tends to ignore detailed geometry. The proposed patchwork assembly method is capable of recovering rich details on eyes, nose, mouth, arms and hands. The two images are of size 640×480, and it takes about 600 seconds on average to generate the high-resolution images.

Quantitative Evaluation Tab. 1 shows a quantitative comparison between our method with nearest neighbor, bilateral filtering and patch-based method [14]. We also compare our method with guided image filtering [20] by using additional cues from an intensity image. Our method outperforms these prior arts on most of the examples. Bilateral filtering on Buddha data works a little better than ours, but the depth boundaries are over-smoothed and the geometry is not visually pleasing. We also notice the performance boost given the additional intensity image.

Table 1. Quantitative comparison.

	Head	Buddha	Cones	Teddy	Lab1	Lab2
NN	0.1306	0.1131	0.0235	0.0254	0.8253	0.7917
Bilateral	0.1093	**0.0929**	0.0227	0.0243	0.7999	0.7600
Patch [14]	0.1337	0.1080	0.0201	0.0246	0.8623	0.7696
Ours	**0.1042**	0.0954	**0.0181**	**0.0198**	**0.7885**	**0.7342**
guided	0.1112	0.1102	0.0217	0.0237	0.8040	0.7738
Ours+color	**0.0541**	**0.0623**	**0.0162**	**0.0193**	**0.7860**	**0.7318**

5. Conclusion

With the patchwork assembly algorithm, we have cast the depth image super-solution problem into an MRF multiple label image segmentation problem and proposed an energy minimization functional to jointly optimize constraints from different information sources, including the input low-resolution depth image, a high resolution training image database, self-similar scene structures, and an optional aligned intensity image. Particularly in some extreme cases when only a small size of training images are available or if the input scene is different from those in the training images, the proposed algorithm achieves a trade-off jointly considering the above criteria and can still enhance the input resolution with a relatively high precision thanks to the proposed pixel-wise MRF labeling formulation. The work can be also extended with inputs from multiple viewpoints or to dynamic environments with additional concerns about calibration, registration and synchronization, but the extensions are beyond the scope of this paper.

Acknowledgement

This work is supported by National Natural Science Foundation of China (NSFC) 61375022, National Key Basic Research Program of China (NKBRP) 2011CB302200, Research Fund for the Doctoral Program of Higher Education of China (RFDP) 20100001120023, and National Nature Science Foundation of China (NSFC) 91120004, 61005037, 90920304.

References

[1] Y. Cui, S. Schuon, D. Chan, S. Thrun, and C. Theobalt. 3d shape scanning with a time-of-flight camera. In *CVPR*, pages 1173–1180. IEEE, 2010. 1, 2

[2] J. Diebel and S. Thrun. An application of markov random fields to range sensing. In *NIPS*, 2005. 2

[3] J. Dolson, J. Baek, C. Plagemann, and S. Thrun. Upsampling range data in dynamic environments. In *CVPR*, pages 1141–1148. IEEE, 2010. 2

[4] R. Fattal. Image upsampling via imposed edge statistics. *ACM Trans. Graph.*, 26(3):95, 2007. 2

[5] W. T. Freeman, T. R. Jones, and E. C. Pasztor. Example-based super-resolution. *CGA*, 22(2):56–65, 2002. 2

[6] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *IJCV*, 40(1):25–47, 2000. 2

[7] D. Glasner, S. Bagon, and M. Irani. Super-resolution from a single image. In *ICCV*, pages 349–356. IEEE, 2009. 2

[8] U. Hahne and M. Alexa. Exposure fusion for time-of-flight imaging. *Comput. Graph. Forum*, 30(7):1887–1894, 2011. 2

[9] K. He, J. Sun, and X. Tang. Guided image filtering. *TPAMI*, 35(6):1397–1409, 2013. 2, 6, 7

[10] D. Holz, R. Schnabel, D. Droeschel, J. Stückler, and S. Behnke. Towards semantic scene analysis with time-of-flight cameras. In *RobuCup*, volume 6556 of *Lecture Notes in Computer Science*, pages 121–132. Springer, 2010. 1

[11] M. Hornacek, C. Rhemann, M. Gelautz, and C. Rother. Depth super resolution by rigid body self-similarity in 3d. In *CVPR*, pages 1123–1130. IEEE, 2013. 1, 2

[12] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. A. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. J. Davison, and A. W. Fitzgibbon. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *UIST*, pages 559–568. ACM, 2011. 2

[13] C. Kuster, T. Popa, C. Zach, C. Gotsman, and M. H. Gross. Freecam: A hybrid camera system for interactive free-viewpoint video. In *VMV*, pages 17–24. Eurographics Association, 2011. 1

[14] O. Mac Aodha, N. D. F. Campbell, A. Nair, and G. J. Brostow. Patch based synthesis for single depth image super-resolution. In *ECCV*, volume 7574 of *Lecture Notes in Computer Science*, pages 71–84. Springer, 2012. 1, 2, 3, 5, 6, 7

[15] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I.-S. Kweon. High quality depth map upsampling for 3d-tof cameras. In *ICCV*, pages 1623–1630. IEEE, 2011. 1, 2, 4

[16] A. N. Rajagopalan, A. V. Bhavsar, F. Wallhoff, and G. Rigoll. Resolution enhancement of pmd range maps. In *DAGM-Symposium*, volume 5096 of *Lecture Notes in Computer Science*, pages 304–313. Springer, 2008. 2

[17] S. Schuon, C. Theobalt, J. Davis, and S. Thrun. Lidarboost: Depth superresolution for tof 3d shape scanning. In *CVPR*, pages 343–350. IEEE, 2009. 1, 2, 3

[18] J. Shotton, T. Sharp, A. Kipman, A. W. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore. Real-time human pose recognition in parts from single depth images. *Commun. ACM*, 56(1):116–124, 2013. 1

[19] J. Sun, J. Zhu, and M. F. Tappen. Context-constrained hallucination for image super-resolution. In *CVPR*, pages 231–238. IEEE, 2010. 2

[20] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV*, pages 839–846, 1998. 2, 5, 6, 7

[21] J. Yang, J. Wright, T. S. Huang, and Y. Ma. Image super-resolution via sparse representation. *TIP*, 19(11):2861–2873, 2010. 2

[22] L. Yang, P. V. Sander, J. Lawrence, and H. Hoppe. Antialiasing recovery. *ACM Trans. Graph.*, 30(3):22, 2011. 2

[23] Q. Yang, R. Yang, J. Davis, and D. Nistér. Spatial-depth super resolution for range images. In *CVPR*. IEEE Computer Society, 2007. 1, 2, 4

[24] G. Zeng and L. J. V. Gool. Multi-label image segmentation via point-wise repetition. In *CVPR*. IEEE Computer Society, 2008. 2

[25] G. Zeng, S. Paris, L. Quan, and F. X. Sillion. Progressive surface reconstruction from images using a local prior. In *ICCV*, pages 1230–1237. IEEE Computer Society, 2005. 1

[26] G. Zeng, S. Paris, L. Quan, and F. X. Sillion. Accurate and scalable surface representation and reconstruction from images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(1):141–158, 2007. 1

[27] J. Zhu, L. Wang, R. Yang, and J. Davis. Fusion of time-of-flight depth and stereo for high accuracy depth maps. In *CVPR*. IEEE Computer Society, 2008. 2