

Depth Image Enhancement Using Local Tangent Plane Approximations

Kiyoshi Matsuo[†]

[†]Hokuyo Automatic Co., LTD.

k-matsuo@hokuyo-aut.co.jp

Yoshimitsu Aoki[‡]

[‡]Keio University

aoki@elec.keio.ac.jp

Abstract

This paper describes a depth image enhancement method for consumer RGB-D cameras. Most existing methods use the pixel-coordinates of the aligned color image. Because the image plane generally has no relationship to the measured surfaces, the global coordinate system is not suitable to handle their local geometries. To improve enhancement accuracy, we use local tangent planes as local coordinates for the measured surfaces. Our method is composed of two steps, a calculation of the local tangents and surface reconstruction. To accurately estimate the local tangents, we propose a color heuristic calculation and an orientation correction using their positional relationships. Additionally, we propose a surface reconstruction method by ray-tracing to local tangents. In our method, accurate depth image enhancement is achieved by using the local geometries approximated by the local tangents.

We demonstrate the effectiveness of our method using synthetic and real sensor data. Our method has a high completion rate and achieves the lowest errors in noisy cases when compared with existing techniques.

1. Introduction

Accurate and high-resolution shape measurement is required in many application fields such as autonomous grasping [25], autonomous navigation [9], and building information modeling (BIM) [17]. Although three dimensional (3D) measurement has become generally available after the release of many consumer RGB-D cameras such as Microsoft KinectTM [3], SoftKinetic[®] DS311 [2], and ASUS Xtion PRO LIVE [1], challenges for computer vision in this area still remain [12, 16, 22, 23, 30]. Because these RGB-D cameras were primarily designed for human interaction applications, the acquired depth images are of low-resolution and corrupted by noise. However, if these popular devices could capture accurate and dense shapes, other exciting applications, such as fine-grained object recognition, precise indoor navigation, and fine manipulation, could become practical.



Figure 1. Top row: original sensor data (left) and aligned color image (right) captured by a consumer RGB-D camera [2]. Bottom row: 3D point clouds of a doll head that visualizes the difference between the enhanced data using the Markov random field (MRF) [6] (left) and our method (right). The granular noise was reduced by our method.

The goal of this paper is to enhance both the accuracy and resolution of a noisy low-resolution depth image captured by a consumer RGB-D camera. To achieve this two-fold enhancement, we use the aligned high resolution color image. Although the aligned pixel-coordinates are widely used in this problem setting [6, 7, 12, 13, 16, 18, 20, 26, 34], these global coordinates are not suitable for handling the geometry of the measured shapes. For example, for a depth image, general smoothing/upsampling filters smooth the surfaces to be parallel to the image plane [13, 18, 20, 34]. Because the image plane has no geometric relationship with the measured surfaces, the geometries of the surfaces are not taken into account, and are sometimes corrupted by these filters.

In this paper, to reconstruct accurate surfaces, we introduce local tangent planes as local linear approximations of the measured surfaces. Our method is composed of two steps, a step that estimates the local tangent planes of the uncorrupted surfaces from a noisy low-resolution depth im-

age and a step that reconstructs surfaces using the estimated local tangent planes. In the reconstruction step, we first reconstruct coarse surfaces using segmentation defined by the contact relations between local tangent planes, and then we refine them by smoothing the normal direction components on each local tangent plane. Smoothing the normal direction components can preserve the local surface geometries that were linearly approximated by the local tangents. The measured data-driven estimation of the local tangent planes and the surface reconstructions based on them enhance both the accuracy and resolution of a depth image without any prior knowledge of the shape models.

We demonstrate the accuracy and noise robustness of our method by conducting experiments on synthetic data and real sensor data captured by a consumer RGB-D camera [2]. Compared with the previous methods in [6, 13, 24], our method can more accurately reconstruct smooth surfaces.

The main contributions of this paper are three-fold:

- We propose a new estimation method for local tangent planes from images captured by an RGB-D camera. A color heuristic principle component analysis (PCA) and method to correct the orientations of the calculated local tangent planes using their positional relationships are introduced to improve the accuracy.
- We propose a new surface reconstruction method from local tangent planes. A parallelizable evaluation of their contacts using ray-tracing and a smoothing of normal direction components are introduced to determine the geometries from local tangent planes.
- We demonstrate the performance of the proposed depth enhancement method using synthetic and real sensor data, and we show that it is more accurate and robust to noise than existing methods.

2. Related Work

Various methods for depth image enhancement have been proposed. In this section, enhancement methods that aim to produce geometrically accurate as well as visually appealing results are reviewed, focusing on their handling of the geometries of measured shapes.

2.1. Model Based Methods

One strategy to enhance depth images leverages the shape models of measured environments. For urban environments, depth image enhancement methods that fit planes or quadratic surfaces to each region of the image have been proposed [10, 15, 28, 32]. In other methods, solutions of some partial differential equations are used as shape models [27, 31]. These methods can achieve highly-accurate, high-resolution reconstruction of surfaces only when the environments have a suitable geometric structure.

2.2. Model Free Methods

For general situations, there are many methods that use local geometric conditions.

Yang *et al.* used the joint bilateral upsampling (JBU) filter [20] to upsample a low-resolution depth image using an aligned color image [34]. In JBU filtering, the co-occurrence of depth and color continuities is used as the local geometric condition. Many methods based on this filter have been proposed [5, 7, 13, 18, 22]. For example, to prevent over smoothing, Garcia *et al.* modified the JBU filter to favor depth discontinuities using the gradient magnitude of a low-resolution depth image [13], and Liu *et al.* used geodesic distances in the color image [22]. Dolson *et al.* extended the filter temporally to determine a depth data stream as captured by LiDAR scanners, combining it with a Gaussian framework [7]. Such local filtering techniques can be parallelized because the filtering calculations at each pixel are independent of each other. Therefore, the connections of local geometric conditions around each pixel cannot be considered.

Global optimization methods can handle the connections of local geometric conditions at each pixel [6, 12, 14, 16, 21, 26]. Diebel and Thrun proposed a depth image enhancement method formulated as an optimization of a Markov random field (MRF), where the smoothness term is weighted according to the texture derivatives of an aligned color image [6]. To maintain finer details, Park *et al.* extended this MRF formulation [26]. They added various information about the image gradient, color image segmentation, edge saliency, and non-local means as MRF terms. Some methods that use sparse representations of images have been proposed [16, 21]. Sparseness in a depth image is caused by the local smoothness of the measured shapes. However, these global optimization methods mainly focus on the segmentation information, that is, the geometric conditions defined only by the projected contours of the measured surfaces on the image plane. The geometries of the interior regions of surfaces are not fully exploited. Recently, Herrera *et al.* used the second derivatives of depth images to reconstruct interior geometries [14]. Although such a second-order smoothness prior reconstructs locally flat surfaces without favoring fronto-parallel planes [33], the orientation of local tangents is a feature that has not been deeply explored.

Local geometries of surfaces have been used to fill holes of 3D data [8, 19, 27]. Kochanowski *et al.* used local tangent planes calculated from dense 3D points surrounding each hole [19]. Because their purpose is to fill local holes of dense points, it is difficult to apply these methods to sparse points captured by a consumer RGB-D camera.

Matsuo and Aoki introduced a depth interpolation procedure based on local tangent planes [24]. In this work, to detect surface regions, local tangent planes were estimated

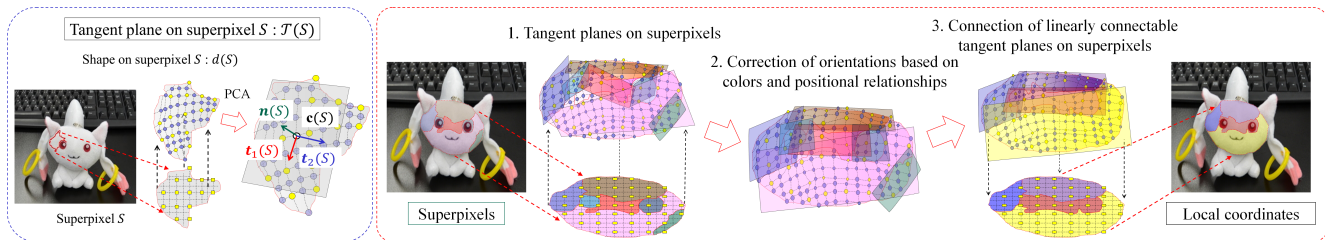


Figure 2. Outline of estimation of local coordinates and local linear approximations.

from a pair comprising a low-resolution depth image and aligned color image. Our work is an improvement of this work. Their estimation of local tangent planes does not work well in the presence of noise because they simply applied PCA to the shapes on each superpixel [11] of the color image. Therefore, we improve the noise robustness by introducing a color heuristic PCA and orientation correction based on their positional relationships. Additionally, we also introduce the usage of local geometries defined by the orientations of the local tangents. The interior geometries of surfaces are used to improve the reconstruction accuracy.

3. Proposed Method

We first introduce the estimation procedure for the local tangents as a local linear approximation of the measured surfaces (Sections 3.1 and 3.2). Because the depth enhancement accuracy strongly depends on the approximation accuracy of local tangents, we divide the local tangent estimation process into three additional steps. This procedure is shown on the right of Figure 2. In Section 3.3, we describe the surface reconstruction procedure. The procedure is composed of two steps, surface region detection and refinement. A conceptual diagram of the surface refinement is shown on the right of Figure 3.

3.1. Tangent Planes on Superpixels

In our method, tangent planes of shapes over small regions are used to calculate linear approximations of measured surfaces. Superpixels [11] of a color image are used as the basic shape-linear regions. This concept was proposed in [24]. Therefore, we first review it and introduce some notations. In the following sections (Sections 3.1.1, 3.1.2, and 3.1.3), our improvements are described.

To estimate local tangents, Matsuo and Aoki applied PCA to each 3D point cloud defined by the locally upsampled depth information on the superpixels [24]. They used JBU [20] filters on each superpixel. Although smooth surfaces are well approximated by this method, in noisy cases, the tangent planes of uncorrupted surfaces cannot be estimated.

In our method, tangent plane $\mathcal{T}(S)$ on superpixel S is used as a 2D plane that has finite width in 3D space. The

center $\mathbf{c}(S)$ of the tangent plane is the center of point cloud $d(S)$ that is defined by locally upsampled depth information on superpixel S . The tangent plane is spanned by the two eigenvectors that correspond to the two largest eigenvalues of the covariance matrix of point cloud $d(S)$. In the left of Figure 2, the conceptual diagram of a tangent plane on a superpixel is shown. The lengths in each direction are defined as the several-fold length of the square root of each eigenvalue. The coefficient l of the length is set to six in our experiments, to use the information of the 3D points that exist inside $\pm 3\sigma$ regions. We decide the sign of the normal vector $\mathbf{n}(S)$ such that the inner product with the center $\mathbf{c}(S)$ is not positive. If a tangent plane's length in the depth direction is larger than threshold d_{th} , we call that tangent plane a steep tangent plane. We set this threshold $d_{\text{th}} = 28.0$ mm.

3.1.1 Color Heuristic PCA

We first calculate the tangent planes on superpixels using a method similar to [24]. These tangent planes are sensitive to outliers caused by measurement noise. We then extract the superpixels that have steep tangent planes to detect regions corrupted by noise. We also extract small superpixels that do not contain many pixels to refine their tangents. In our experiments, we set the threshold for the number of pixels n_{small} to 32. The tangent planes for these extracted superpixels are recalculated using the following color heuristic PCA. A color heuristic covariance matrix $\mathbf{A}_{\text{color}}(d(\tilde{S}))$ is defined as

$$\frac{1}{W} \sum_{\mathbf{x} \in d(\tilde{S})} w(|\mathbf{i}(\mathbf{x}) - \mathbf{i}_S|_1) (\mathbf{x} - \mathbf{c}_{\text{color}}(\tilde{S})) (\mathbf{x} - \mathbf{c}_{\text{color}}(\tilde{S}))^T, \quad (1)$$

where $d(\tilde{S})$ is the extended shape on superpixel S [24], w is a weight function, $\mathbf{i}(\mathbf{x})$ is the RGB-color of point \mathbf{x} , \mathbf{i}_S is the average RGB-color on superpixel S , and $\mathbf{c}_{\text{color}}(\tilde{S})$ is the color-difference weighted center point of $d(\tilde{S})$. The denominator W is the total weighted sum $W = \sum_{\mathbf{x} \in d(\tilde{S})} w(|\mathbf{i}(\mathbf{x}) - \mathbf{i}_S|_1)$. The notation $|\mathbf{i}|_1$ denotes the L^1 norm of RGB-color vector \mathbf{i} . The color-difference weighted center point of $d(\tilde{S})$ is similarly defined as

$$\mathbf{c}_{\text{color}}(\tilde{S}) = \frac{1}{W} \sum_{\mathbf{x} \in d(\tilde{S})} w(|\mathbf{i}(\mathbf{x}) - \mathbf{i}_S|_1) \mathbf{x}. \quad (2)$$

Here, we use the weight function $w(t) = \exp(-t)$. We use tangent planes that are defined by the eigenvalues and eigenvectors of this color heuristic covariance matrix $\mathbf{A}_{\text{color}}(d(\hat{S}))$, instead of the steep tangent planes. Because extended neighboring shape information is used, the smoothed tangent plane is calculated, even in noisy low-resolution situations. The color-difference weights cause the recalculated tangent plane to spread to neighboring regions that have the same color as the superpixel. Therefore, over smoothing around object boundaries is prevented.

3.1.2 Contact Evaluation by Ray-tracing

To improve the accuracy of the approximations, we correct tangent orientations using spatial relationships. In this section, we introduce our evaluation method of the spatial relationships by ray-tracing.

First, we define a distance $d_p(\mathcal{T}_1, \mathcal{T}_2)$ between two tangent planes \mathcal{T}_1 and \mathcal{T}_2 at pixel p . Light ray \mathcal{L}_p at pixel p is defined as the line through pixel p and the camera center. When $\mathcal{L}_p \cap \mathcal{T}_1 = \{\mathbf{q}_1\}$ and $\mathcal{L}_p \cap \mathcal{T}_2 = \{\mathbf{q}_2\}$, it is defined as

$$d_p(\mathcal{T}_1, \mathcal{T}_2) = |\mathbf{q}_2 - \mathbf{q}_1|_2. \quad (3)$$

If there are no or more than two intersection points, we define $d_p(\mathcal{T}_1, \mathcal{T}_2) = \infty$ to be the invalid distance value. This is the distance between the intersection points of the tangent planes and light ray \mathcal{L}_p . We then define the set of pixels $Q(\mathcal{T}_1, \mathcal{T}_2) = \{p : \text{pixel} \mid d_p(\mathcal{T}_1, \mathcal{T}_2) \neq \infty\}$. This is the set of pixels in the intersection of the projections of \mathcal{T}_1 and \mathcal{T}_2 in the image plane. We use the following value $d(\mathcal{T}_1, \mathcal{T}_2)$ as the distance between local tangent planes \mathcal{T}_1 and \mathcal{T}_2 .

$$d(\mathcal{T}_1, \mathcal{T}_2) = \max_{p \in Q(\mathcal{T}_1, \mathcal{T}_2)} \{d_p(\mathcal{T}_1, \mathcal{T}_2)\}. \quad (4)$$

Here, if the set $Q(\mathcal{T}_1, \mathcal{T}_2)$ is empty, there are no light rays to the pixels that intersect with both tangent planes \mathcal{T}_1 and \mathcal{T}_2 . In this case we set $d(\mathcal{T}_1, \mathcal{T}_2) = \infty$.

3.1.3 Correction of Orientations

We correct the orientations of the tangent planes on superpixels while considering the connections among neighboring tangent planes. We use $d(\mathcal{T}_1, \mathcal{T}_2)$ in Equation 4 to extract the neighbors of a superpixel. For each superpixel S , we extract the set of superpixels $\mathcal{N}_{\mathcal{T}}(S, d_{\text{th}})$ that have spatially neighboring tangent planes.

$$\mathcal{N}_{\mathcal{T}}(S, d_{\text{th}}) = \{S' : \text{superpixel} \mid d(\mathcal{T}(S), \mathcal{T}(S')) < d_{\text{th}}\}. \quad (5)$$

Here, d_{th} is a distance threshold. In our experiments, we set $d_{\text{th}} = 28.0$ mm, the same value as the definition of a steep tangent plane. To estimate an accurate normal vector, we apply PCA to the 3D point cloud $N(S, d_{\text{th}})$ that consists of the center points of the neighboring tangent planes as follows:

$$N(S, d_{\text{th}}) = \{\mathbf{c}(S') \mid S' \in \mathcal{N}_{\mathcal{T}}(S, d_{\text{th}})\}. \quad (6)$$

We use the unit eigenvector that corresponds to the smallest eigenvalue of the covariance matrix of this 3D point cloud as the corrected normal vector on superpixel S . Each center point is robust to measurement noise because they are averaged 3d points. And they are widely distributed on the neighboring surfaces. Therefore, we simply apply PCA to only these center points to estimate local orientations of uncorrupted surfaces. Some results for our normal correction for noisy data are given in the supplementary materials.

3.2. Linear Approximations of Surfaces

We evaluate the normal similarities by quantizing the spherical coordinates of the normal vectors. To simplify the region connection processes, we use orientation quantizations. If the normal vectors have the same quantized spherical coordinates, then we assume that they are similar. In our implementation, the minus vector of the camera direction vector is used as the zenith direction of the spherical coordinate, and the two angle components, polar angle θ and azimuthal angle ϕ , are quantized by the same quantization width θ_{th} . In the circumpolar regions where the polar angle is near 0° or 180° , we use one quantized value, regardless of the azimuthal angle because the azimuthal angle component does not significantly change the orientation. Our quantization tables are shown in the left image of Figure 3. Colors used in the general normal maps represent orientations, and the grid indicates the quantization.

We connect superpixels that have spatially neighboring and similar normal tangent planes to determine the local linear approximations of surfaces. Here, we use the distance value between tangent planes (Equation 4), the distance threshold d_{th} , and a small value $\theta_{\text{th}}^{\text{small}}$ as the angle quantization width. The quantization width controls the tolerance of each linear approximation. In our experiments, d_{th} is set to 28.0 mm as before, and we set $\theta_{\text{th}}^{\text{small}} = 25^\circ$. Generated regions are used as local coordinates to handle local geometries of measured surfaces. We then recalculate the tangent planes on each local coordinate by applying PCA to the local shapes.

Next, we estimate the surface regions using these local coordinates. We detect them by connecting the local coordinates if they have spatially neighboring and similar normal tangents. We use a large normal vector quantization width $\theta_{\text{th}}^{\text{large}}$ in this step. In our experiments, we set this threshold $\theta_{\text{th}}^{\text{large}} = 80^\circ$. We simply implement this connection process by giving the same label to pairs of regions that have spatially neighboring and similar normal tangent planes.

3.3. Surface Reconstruction

In our method, the depth image is enhanced using the previously described local coordinates and surface segmentation. We first reconstruct coarse surfaces using the JBU filter [20]. Here, this filter only uses depth data on each

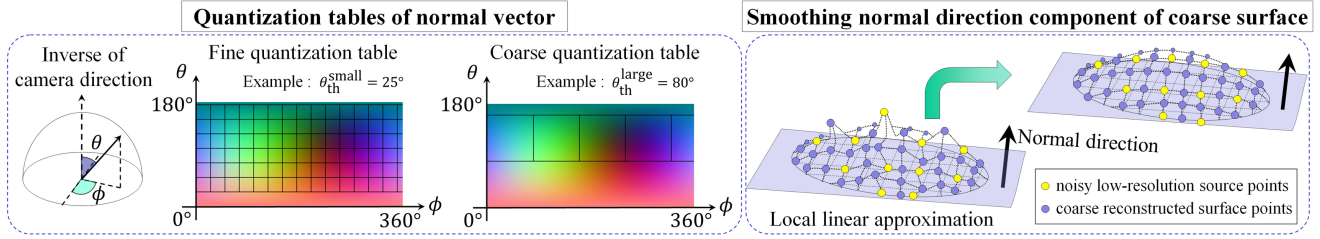


Figure 3. Quantization of the normal vector and smoothing normal direction of shape.

surface if the number of low-resolution depth data points is larger than threshold n_{source} . We interpolate the depth image using the depths from the same surface so that we do not add depth data from other surfaces. However, we do not interpolate on any surfaces where there are only a few source depth data points to ensure the accuracy of the interpolated depth image. In our experiments, we set the threshold $n_{\text{source}} = 5$.

After coarse reconstruction of the surfaces, we smooth the normal direction components of 3D points on each local coordinate using a Gaussian filter. The right image of Figure 3 shows a conceptual diagram of this smoothing. Here, we avoid smoothing on the regions that have steep tangent planes (such as object boundaries), as the local geometries are not well approximated by the local coordinates. The explicit definition of smoothed point $\tilde{\mathbf{x}}$ of point \mathbf{x} in local coordinates R is

$$\tilde{\mathbf{x}} = \frac{1}{W} \sum_{\mathbf{x}' \in d(R)} w_{\text{Gauss}}(|\mathbf{x} - \mathbf{x}'|_1)(\mathbf{x}', \mathbf{n}(R))\mathbf{n}(R) + \mathbf{x}_{\mathcal{T}}, \quad (7)$$

where w_{Gauss} is the Gaussian weight function, $d(R)$ is the interpolated shape in region R , $\mathbf{n}(R)$ is the normal vector, $\mathbf{x}_{\mathcal{T}}$ is the tangential component of point \mathbf{x} , and W is the total weighted sum. If $\mathbf{t}_1(R)$ and $\mathbf{t}_2(R)$ are the orthonormal tangent vectors, then $\mathbf{x}_{\mathcal{T}} = (\mathbf{x}, \mathbf{t}_1(R))\mathbf{t}_1(R) + (\mathbf{x}, \mathbf{t}_2(R))\mathbf{t}_2(R)$. To refine the interpolated shapes, we smooth them while preserving the local geometries that are linearly approximated by the local coordinates.

4. Experimental Results

We tested our method by conducting two sets of experiments for quantitative and qualitative evaluations. The quantitative evaluations were performed using the well-known Middlebury Stereo Datasets [29] (Section 4.1). In addition, real sensor data captured by a consumer RGB-D camera were used for the qualitative evaluations (Section 4.2). We used the same settings for the parameters of our method in all experiments. A summary of these parameters is given in Table 1. Here, we compared our results with the results of some other methods. The purpose of this comparison was to evaluate the effectiveness of the proposed local tangent coordinates. Therefore, we selected basic and representative methods that use the pixel

d_{th}	n_{small}	l	$\theta_{\text{th}}^{\text{small}}$	$\theta_{\text{th}}^{\text{large}}$	n_{source}
28.0 mm	32 pixel	6.0	25.0°	80.0°	5 pixel

Table 1. Parameters on our experiments.

global coordinates, bilinear interpolation, MRF [6], and pixel weighted average strategy (PWAS) [13]. Additionally, we compared our method with an existing local tangent plane based method [24].

4.1. Quantitative Evaluation on Synthetic Data

We used the depth images of the Middlebury datasets as the ground truth and used them to generate noisy low-resolution images by downsampling the vertical and horizontal axes at rates of (1/4, 1/4), (1/4, 1/2), (1/2, 1/4), (1/2, 1/2), (1/2, 1), (1/2, 1), and (1, 1). The Middlebury datasets provide various images of three different sizes. We used the smallest images (463 × 370 pixels) of the scenes *Art*, *Books*, *Dolls*, and *Moebius* because it is difficult to interpolate an image that contains local variations in color and shape. The following measurement model (Equation 8) was used to generate simulated noise, and we used the high-resolution color images directly.

We used a flash lidar device measurement model [4] to simulate measurement noise. In this model, the covariance of the distance measured at each pixel is proportional to the square of the true distance and inversely proportional to the cosine of the angle of incidence of the measuring light ray. When the ground truth value was d_{true} , we generated a random number for the measured distance d at pixel p from the following Gaussian distribution

$$p(d | d_{\text{true}}, p) \propto \exp\left(-\frac{(d - d_{\text{true}})^2}{2\sigma(p, d_{\text{true}})^2}\right), \quad (8)$$

$$\sigma(p, d_{\text{true}}) = k \frac{d_{\text{true}}^2}{\cos(\alpha)}. \quad (9)$$

Coefficient k controls the level of the added noise. We varied k from 0.0 to 5.0×10^{-6} in 1.0×10^{-6} increments. At the maximum noise level $k = 5.0 \times 10^{-6}$, the covariance of the measured distance to a target that is 1 m away is 5 mm. For reference, the coefficients of some range sensors are given in the supplementary materials.

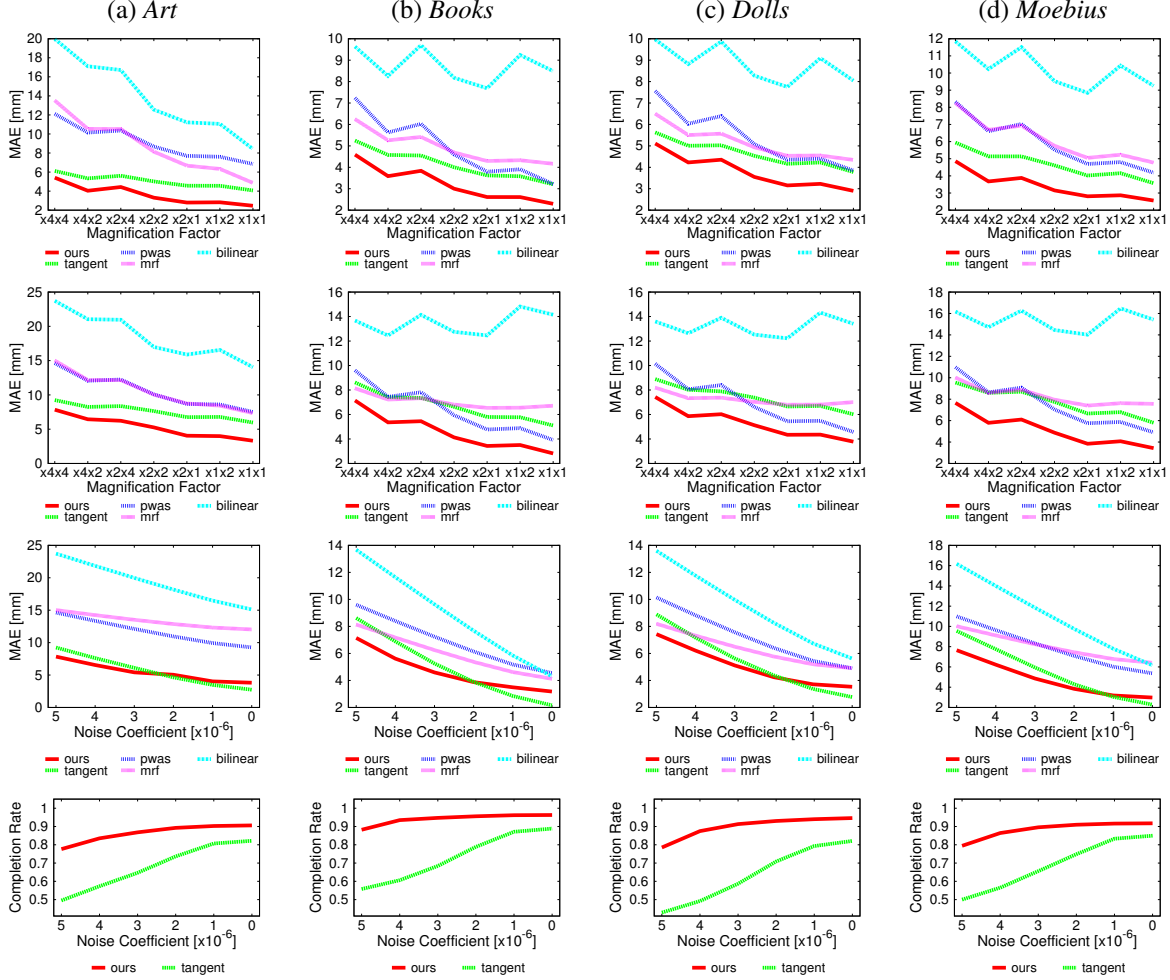


Figure 4. Quantitative comparisons: MAE scores under low level noise ($k = 3.0 \times 10^{-6}$), MAE scores under high level noise (5.0×10^{-6}), and MAE scores and completion rates for various noise levels (upsampling rate = 4×4).

We evaluated the interpolation accuracy and completion rate using the mean absolute error (MAE) scores and the following completion rate of interpolated depth image d .

$$\text{completion rate} = \frac{|P(d) \cap P(d_{\text{true}})|}{|P(d_{\text{true}})|}. \quad (10)$$

Here, $P(d)$ is the set of pixels that have depth value in depth image d , $|P(d)|$ is the number of pixels in $P(d)$, and d_{true} is the ground truth depth image.

Figure 4 shows the MAE scores using different magnification factors and fixed noise coefficients 3.0×10^{-6} and 5.0×10^{-6} , as well as MAE scores and completion rates using different noise coefficients and a fixed magnification factor 4×4 . In these graphs, the results of the proposed method, bilinear interpolation, MRF, PWAS, and the method described in [24] are shown using red, watery, purple, blue, and green lines, respectively. In the noisy situations, our method outperformed the other methods in terms of accuracy. Our method had regions where depth inter-

polation was not performed, similar to [24]. The tangent based method in [24] does not interpolate regions that have steep tangent planes caused by noise, hence its errors were small, but its completion rates were low. Our method had low errors and completion rates of over 78%, higher than the tangent based method.

In Figure 5, the absolute errors at each pixel are shown. In these images, pixels that have no depth value are colored black. All results of bilinear interpolation, MRF, and PWAS had large errors on the neighboring regions of surface boundaries. In contrast, the tangent plane based method [24] and our method did not interpolate the boundaries of surfaces. In these results, the tangent plane based method [24] did not interpolate regions that were not neighboring boundaries. In the bilinear interpolation and PWAS results, errors in the interior regions of surfaces were large. In the MRF results, errors in the interior regions of surfaces were smaller. However, in the results of our method, errors were even more reduced than in the MRF results.

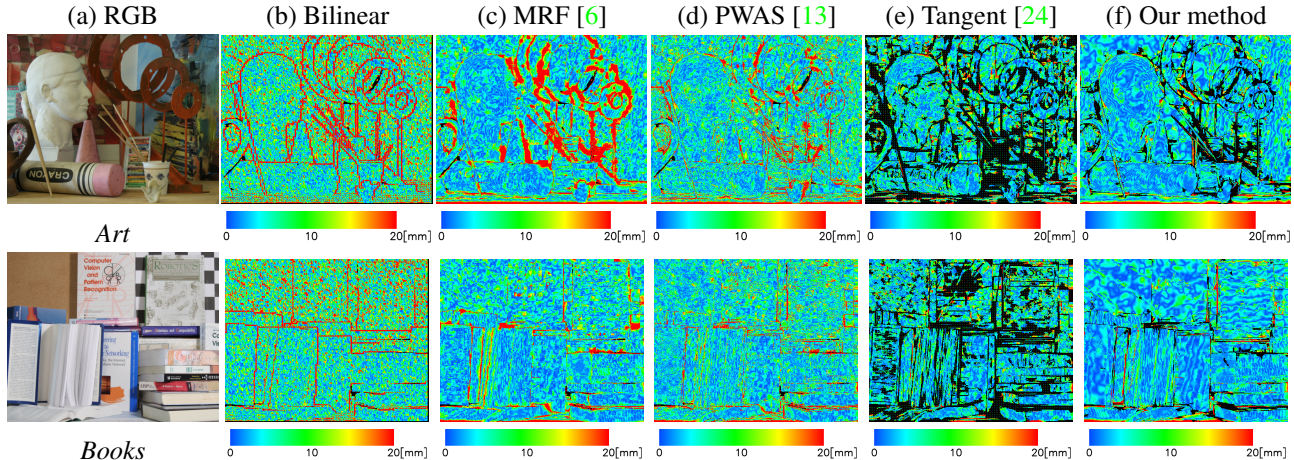


Figure 5. Resulting enhancement error at each pixel (Scene *Art* and *Books* in the Middlebury datasets, noise coefficient $k = 3.0 \times 10^{-6}$, upsampling rate = $\times 4 \times 4$).

4.2. Qualitative Evaluation on Real RGB-D Data

We applied our method to real images captured by a consumer RGB-D camera. We captured depth images of size 124×92 and aligned color images of size 620×460 using a SoftKinetic® DS311 [2]. Because the field of views of the time of flight camera and the color camera are different, we cropped the images to the commonly measured regions.

Figures 1 and 6 show the measured real sensor data and enhanced data as 3D point clouds. To ensure the surface smoothness, different viewpoints from the real camera viewpoints were used. In Figure 6, the surface of a reconstructed book in the center of the scene shows differences for each method. The surface reconstructed by PWAS was too rough, and the result of the tangent based method in [24] had some holes. MRF reconstructed a smooth surface, but the characters on the surface are not readable. In the results of our method, we can read the characters, just as in the detailed color image. Similarly, as shown in the detailed images of Figure 1, some of a rough doll surface was reconstructed by MRF. Conversely, our method reconstructed a smooth doll surface.

4.3. Parallelizability and Processing Time

The local tangent based method can be primarily implemented by parallel processing. In our method, tangent planes on superpixels can be calculated by parallel processing over superpixels. Our ray-tracing for the contact evaluation can also be parallelized over all pixels. Once distances at each pixel (Equation 3) are calculated, they can be efficiently integrated to detect the distances between each tangent (Equation 4) by creating a reference table.

In this paper, we propose a partially parallelizable depth image enhancement method, but not its parallel implementation. Therefore, we have not optimized the implementation for our experiments. However, we have implemented

<i>Art</i>	Surface segmentation	Surface reconstruction	total
Tangent [24]	15.39 sec	4.36 sec	19.75 sec
Ours method	6.51 sec	12.48 sec	18.99 sec
<i>Books</i>	Surface segmentation	Surface reconstruction	total
Tangent [24]	17.28 sec	5.18 sec	22.46 sec
Ours method	7.94 sec	22.87 sec	30.81 sec

Table 2. Comparisons of processing times (upsampling of $\times 4 \times 4$).

our method and tangent plane based method in [24] using OpenMP so that we can approximately evaluate their processing times. We ran our experiments on an Intel Core i7-2600 CPU (3.40 GHz) with 16 GB RAM. Table 2 shows the comparisons of the processing times. Because our condition to connect superpixels is strict enough to omit many unnecessary connection candidates of each superpixel, the connection process is efficiently achieved. Therefore, the processing times for the surface segmentation of our method were smaller than those of the method in [24]. However, processing times for the surface reconstruction were larger. Because in our implementation, our shape refinement process is parallelized in each local coordinate, larger processing time is needed for larger local coordinates. Therefore, for the scenes where large flat surfaces exist, such as *Books*, total processing times were large, which were over 30 sec.

5. Conclusion

In this paper, we proposed a depth image enhancement method using local tangent planes as local linear approximations. Our method is composed of two processes, an estimation process of local tangent planes and a reconstruction process of the surfaces from the local tangents. To improve the noise robustness of the local tangent estimation, we introduce a color heuristic PCA and orientation correc-



Figure 6. Visual evaluation on a real sensor data. Left images in each figure show the details of the book in the center of the scene. Our method reconstructed a so smooth surface of the centered book that the characters on the cover are readable, just as in the detailed color image.

tion using their positional relationships. In the reconstruction process, we use not only their positional relationships, but also their spatial orientations. Here, we introduce a local smoothing of the normal direction components of surfaces.

Experiments using synthetic data and real data captured by a consumer RGB-D camera were conducted, and both results show that our method can successfully enhance noisy low-resolution depth images. In noisy cases when the previous local tangent based method in [24] could not be applied, our method could estimate accurate local tangents and achieved a high completion rate of over 78%. Compared with the existing methods that use the global coordinates of the image plane, the interior regions in particular of surfaces were reconstructed more accurately by our method. These experimental results show the effectiveness of our local tangent based depth enhancement.

Because our method uses only a single pair of depth and color images, multiple spatial and temporal viewpoints are not needed. Therefore, our method is beneficial for 3D measurement in places where multiple viewpoints can not be set, like a disaster site, or 3D measurement of moving objects. However, large processing times were required by our method. Our method can not enhance a depth image in one capturing time of general consumer RGB-D cameras. This is a large drawback of our method. Although our method is applicable to some applications, like building a 3D map for autonomous navigation, further improvement is needed to expand the application field. There are some room of improvement in the use of local tangent planes for the reduction of the processing time. We would like to continue to improve our method by focusing on the geometric nature of local tangent planes.

References

- [1] Asus xtion pro live. http://www.asus.com/Commercial_3D_Sensor/Xtion_PRO_LIVE/.
- [2] Softkinetic® ds311. http://www.softkinetic.com/Portals/0/Documents/PDF/WEB_20130527_SK_DS311_Datasheet_V3.0.pdf.
- [3] Xbox 360 kinect™. <http://www.xbox.com/en-US/kinect>.
- [4] D. Anderson, H. Herman, and A. Kelly. Experimental characterization of commercial flash lidar devices. In *International Conference of Sensing and Technology*, volume 2, 2005.
- [5] D. Chan, H. Buisman, C. Theobalt, and S. Thrun. A Noise-Aware Filter for Real-Time Depth Upsampling. In *Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications - M2SFA2 2008*, Marseille, France, 2008. Andrea Cavallaro and Hamid Aghajan.
- [6] J. Diebel and S. Thrun. An application of markov random fields to range sensing. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 291–298. MIT Press, Cambridge, MA, 2006.
- [7] J. Dolson, J. Baek, C. Plogemann, and S. Thrun. Upsampling range data in dynamic environments. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1141–1148, june 2010.
- [8] D. Doria and R. Radke. Filling large holes in lidar data by inpainting depth gradients. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 65–72, june 2012.
- [9] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard. 3-d mapping with an rgb-d camera. *Robotics, IEEE Transactions on*, 30(1):177–187, Feb 2014.
- [10] G. Facciolo and V. Caselles. Geodesic neighborhoods for piecewise affine interpolation of sparse data. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 365–368, nov. 2009.
- [11] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.
- [12] D. Ferstl, C. Reinbacher, R. Ranftl, M. Ruether, and H. Bischof. Image guided depth upsampling using anisotropic total generalized variation. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 993–1000, Dec 2013.
- [13] F. Garcia, B. Mirbach, B. Ottersten, F. Grandidier, and A. Cuesta. Pixel weighted average strategy for depth sensor data fusion. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 2805–2808, sept. 2010.
- [14] D. Herrera C., J. Kannala, L. Ladický, and J. Heikkilä. Depth map inpainting under a second-order smoothness prior. In J.-K. Kämäräinen and M. Koskela, editors, *Image Analysis*, volume 7944 of *Lecture Notes in Computer Science*, pages 555–566. Springer Berlin Heidelberg, 2013.
- [15] Z. Jia, Y.-J. Chang, T.-H. Lin, and T. Chen. Dense interpolation of 3d points based on surface and color. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 869–872, sept. 2011.
- [16] M. Kiechle, S. Hawe, and 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*, pages 1545–1552, Dec 2013.
- [17] C. Kim, H. Son, and C. Kim. Automated construction progress measurement using a 4d building information model and 3d data. *Automation in Construction*, 31(0):75–82, 2013.
- [18] J. Kim, J. Lee, S.-R. Han, D. Kim, J. Min, and C. Kim. A high quality depth map upsampling method robust to misalignment of depth and color boundaries. *Journal of Signal Processing Systems*, 75(1):23–37, 2014.
- [19] M. Kochanowski, P. Jenke, and W. Straßer. Analysis of texture synthesis algorithms with respect to usage for hole-filling in 3d geometry. In *Annual Meeting on Information Technology and Computer Science*, pages 1–6, Feb 2008.
- [20] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. *ACM Trans. Graph.*, 26(3), July 2007.
- [21] Y. Li, T. Xue, L. Sun, and J. Liu. Joint example-based depth map super-resolution. In *Multimedia and Expo (ICME), 2012 IEEE International Conference on*, pages 152–157, July 2012.
- [22] M.-Y. Liu, O. Tuzel, and Y. Taguchi. Joint geodesic upsampling of depth images. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 169–176, June 2013.
- [23] S. Lu, X. Ren, and F. Liu. Depth enhancement via low-rank matrix completion. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3390–3397, June 2014.
- [24] K. Matsuo and Y. Aoki. Depth interpolation via smooth surface segmentation using tangent planes based on the superpixels of a color image. In *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on*, pages 29–36, Dec 2013.
- [25] J. Papon, T. Kulvicius, E. Aksoy, and F. Worgotter. Point cloud video object segmentation using a persistent super-voxel world-model. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 3712–3718, Nov 2013.
- [26] J. Park, H. Kim, Y.-W. Tai, M. Brown, and I. Kweon. High quality depth map upsampling for 3d-tof cameras. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1623–1630, nov. 2011.
- [27] S. Park, X. Guo, H. Shin, and H. Qin. Shape and appearance repair for incomplete point surfaces. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1260–1267 Vol. 2, Oct 2005.
- [28] A. Sampath and J. Shan. Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *Geoscience and Remote Sensing, IEEE Transactions on*, 48(3):1554–1567, March 2010.
- [29] D. Scharstein and C. Pal. Learning conditional random fields for stereo. In *Computer Vision and Pattern Recognition*,

2007. *CVPR '07. IEEE Conference on*, pages 1–8, june 2007.
- [30] J. Shen and S.-C. S. Cheung. Layer depth denoising and completion for structured-light rgb-d cameras. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 1187–1194, 2013.
- [31] K. Varadarajan and M. Vincze. Surface reconstruction for rgb-d data using real-time depth propagation. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 723–724, nov. 2011.
- [32] R. Wang, J. Bach, J. MacFarlane, and F. Ferrie. A new up-sampling method for mobile lidar data. In *Applications of Computer Vision (WACV), 2012 IEEE Workshop on*, pages 17–24, jan. 2012.
- [33] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon. Global stereo reconstruction under second-order smoothness priors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12):2115–2128, Dec 2009.
- [34] Q. Yang, R. Yang, J. Davis, and D. Nister. Spatial-depth super resolution for range images. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, june 2007.