LARGE-AREA DEPTH RECOVERY FOR RGB-D CAMERA

Zengqiang Yan⋆*, Li Yu*⋆*, and Zixiang Xiong*†

⋆ School of Electron. Inf. & Commun., Huazhong Univ. of Sci. & Tech., Wuhan, China
† Department of ECE, Texas A&M University, College Station, USA
Email: ZengqiangYan@hust.edu.cn, hustlyu@hust.edu.cn, zx@ece.tamu.edu

ABSTRACT

In this paper, a large-area depth recovery method for RGB-D camera is proposed. Considering that pixels along edges between different regions usually share similar depth values, we first select reliable pixels along edges of large-area depth missing regions and project them into the world coordinate system. Then, by examining the distribution of these pixels, we apply a weighted least squares method to approximate the surface function. With the help of the surface function, missing depth values can be recovered correctly. To the best of our knowledge, this is the first recovery method focusing on large areas of missing depth information. Qualitative evaluation demonstrates the effectiveness of the proposed method.

Index Terms— Depth recovery, surface approximation, point cloud processing, RGB-D camera

1. INTRODUCTION

Accurate depth information is essential for applications such as 3DTV [1], 3D reconstruction [2] and virtual viewpoint synthesis [3]. Different from color information that can be easily obtained, acquiring accurate depth information is more difficult. Till now, time-of-light (ToF) [4, 5] based technique and structured light [6, 7] based technique have been widely researched and used for depth acquisition.

In ToF cameras, depth information is obtained by calculating the phase differences between emitted lights and captured lights. Structured light based technique acquires depth information by projecting predesigned patterns and analyzing the deformation of patterns on the surface of target scene. For both techniques, the generated depth maps usually suffer from the problem of missing information. Fig. 1 shows the RGB-D data obtained by Kinect #1 (structured light based depth sensor), where Fig. 1 (a) is color image and Fig. 1 (b) is the corresponding depth map. In the depth map, there exist both large- and small-area depth missing regions.

To improve the accuracy of the depth map, many depth recovery methods have been proposed and most used color image as guidance. In [8], Kopf *et al.* applied an iterative joint bilateral filter through the help of color images. The joint bilateral filtering method was also used for depth map super resolution by Yang *et al.* [9]. In [10], Liu *et al.* conducted inpainting and upsampling for depth enhancement by applying color image guided anisotropic diffusion. Min *et al.* in [11] proposed a weighted mode filtering method based on a joint histogram for upsampling and depth enhancement. We tested these methods by conducting multiple contrast experiments, and results are provided in Section 3. From the results, we see that these methods show good performances on depth recovery of small regions, but cannot handle the large-area depth missing problem.

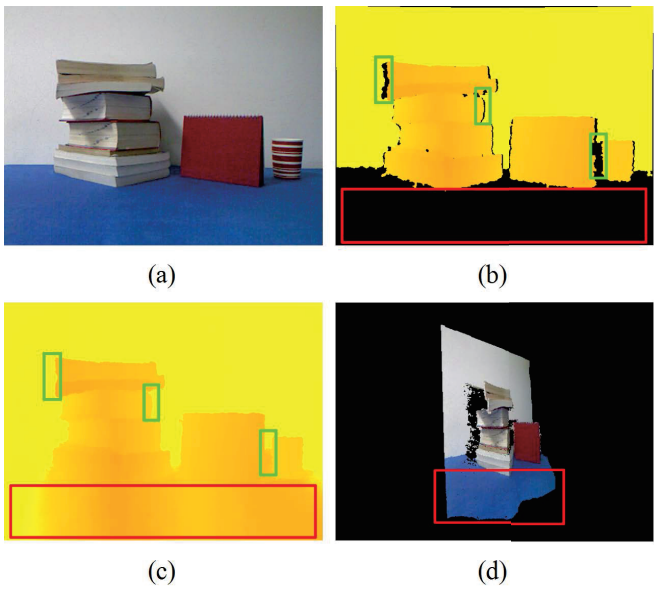


Fig. 1. RGB-D data captured by Kinect #1: (a) Color image. (b) Depth map. (c) Depth map processed by a joint bilateral filter [8]. (d) Point cloud constructed based on depth map (c).

Fig. 1 (c) is the depth map processed by the iterative joint bilateral filter [8], and recovered depth information is propagated from neighbor pixels. In Fig. 1 (c), small depth missing regions can be correctly recovered. However, the filter fails to recover large-area depth missing region labeled by the red rectangle. Fig. 1 (d) shows the corresponding point cloud,

Work supported by NSFC (Grant No. 61231010), NSF (Grant No. 1216001), and Research Fund for the Doctoral Program (20120142110015).

and the recovered large-area region is definitely wrong. Motivated by this, we focus on large-area depth missing problem.

Considering that neighbor pixels for large-area depth missing region inpainting are quite limited, we propose to recover depth values through the help of its surface function. In our method, we first select pixels that can be used for surface function calculation based on the edge and depth maps. Then the selected pixels are projected to the world coordinate system according to a standard projection matrix. After that, the surface function is calculated by applying a fitting method to the discrete points in the world coordinate system. Finally, missing depth values of the large-area region can be recovered based on the surface function. Experimental results demonstrate the effectiveness of our proposed method.

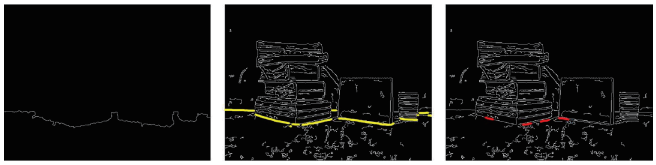
The remainder of this paper is organized as follows. In Section 2, we describe the process of our large-area depth recovery method. Experimental results are provided in Section 3. Section 4 concludes the paper.

2. PROPOSED METHOD

In our method, the key step is to select correct pixels to approximate the surface function. We first coarsely select edge pixels as candidates, and then remove noisy candidates through weight assignment. Details of the proposed method are provided in following subsections.

2.1. Candidates Selection

The first step of the proposed method consists of identification of large-area depth missing region and selection of candidates for further processing. In the depth map, pixels that have no depth values are identified as holes. In our method, we assume that there exists only one large-area depth missing region in each depth map. As a result, we can identify the large-area depth missing region *R* by selecting the hole with the largest size. For complex scenes, image segmentation algorithms [12, 13] can be used for identification.



(a) (b) (c)

Fig. 2. (a) Dilated contour map of the large-area depth missing region. (b) All candidates (labeled by the color yellow) in the edge map. (c) Selected pixels (labeled by the color red) for surface function calculation by removing noisy candidates.

The candidates selection part can be divided into coarse candidates selection and weight assignment. Pixels that can be used for approximating the surface function mainly exist in the edges around *R*, due to the fact that these pixels share the same or similar depth values with *R*.

In our method, we first dilate the contour map of the region as shown in Fig. 2 (a). Then, we apply a Canny edge detector to the color image to obtain the edge map I_e, and candidates can be obtained by

$$\pi_c = I_c \text{ AND } I_e, \tag{1}$$

where I_c is the dilated contour map and π_c is the candidate set as shown in Fig. 2 (b) (yellow superimposed pixels). When color image is unavailable, we use the contour map without dilatation. Then, each candidate is assigned with a weight calculated by

$$w(x,y) = K \cdot \phi(x,y) + M \cdot \frac{\sqrt{\sum_{(s,t)\in\Omega} [d(x,y) - d(s,t)]^2}}{H(x,y)+1}, \tag{2}$$

where K and M are coefficients, *H(x,y)* is the Hamming distance between the candidate and its nearest pixel in *R*, Ω represents a 5×5 neighborhood of pixel *(x,y)*, *d(x,y)* is the depth value of pixel *(x,y)*, and

$$\phi(x,y) = \|\mathrm{p}(x,y) - p(m,n)\| \tag{3}$$

represents the RGB similarity between $p(x,y)$ and $p(m,n)$. For each candidate $p(x,y)$, we search for its nearest pixel $p(m,n)$ in the color image on the condition that the corresponding pixel $d(m,n)$ in the depth map belongs to *R*. Thus, by calculating $\phi(x,y)$ of each candidate, we identify candidates along reliable edges that have higher weights. Then, based on the weight similarity between candidates, we divide the candidates in Fig. 2 (b) into different segments.

2.2. Noisy Candidates Removal

According to the imaging principle of camera, we can transform the coordinates between the world coordinate system and the image coordinate system via

$$d(x,y) \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = P \bullet \begin{pmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{pmatrix}, \tag{4}$$

where *P* is the projection matrix that can be obtained by the calibration algorithm [14]. According to (4), the coordinates (X_W, Y_W, Z_W) in the world coordinate system can be obtained by multiplying the vector *(x,y,1)* with the inverse matrix of *P* and depth value *d(x,y)*. By examining the projection matrix *P*, we find that the coordinate transformation is linear. Thus in our method the projection matrix is set to the standard projection matrix of Kinect #1.

Based on (4), each segment is transformed to the world coordinate system. Then a least squares method is applied,

and approximated surface function of each segment can be obtained. In our method, we use the included angle between the surface and XOY plane to represent the fitting result of each segment. By looking into the statistical histogram of fitting results of all the segments in Fig. 3, we select the candidates that correspond to the peak in the histogram.

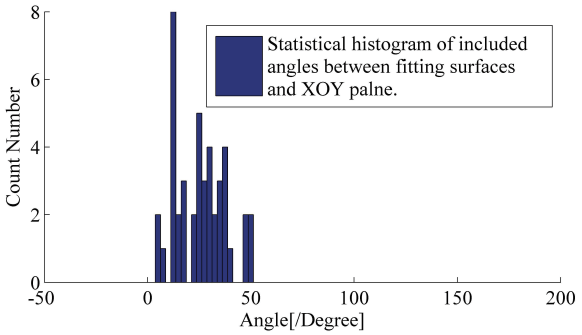


Fig. 3. Statistical histogram of fitting results of segments. Fitting result of each segment is represented by the included angle between the approximated surface and XOY plane.

2.3. Surface Function Approximation

At the beginning of this stage, we further remove noisy candidates. To void the situation that pixels selected for surface approximation are too concentrated, we add to the weight of each selected candidate obtained above a tuning factor $\varphi(x, y)$ calculated by

$$\varphi(x,y) = \sqrt{(x - \sum_{(u,v)\in S} \frac{u}{N})^2 + (y - \sum_{(u,v)\in S} \frac{v}{N})^2}, \quad (5)$$

where (u, v) is the coordinates of candidate and S represents the current candidates set with size *N*. After weight assignments, only high-weight candidates are selected for surface approximation as shown in Fig. 2 (c).

Then, we approximate the surface function of large-area depth missing region by studying the distribution of selected pixels in the world coordinate system as shown in Fig. 4 (a). In our method, the weights of the pixels are also taken into consideration to correct the surface function.

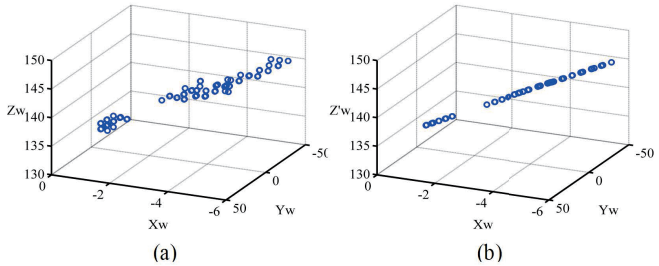


Fig. 4. (a) Distribution of the selected pixels (red pixels in Fig. 2 (c)) in the world coordinate system. (b) Distribution of the selected pixels by recalculating the coordinates according to the surface function.

Considering the distribution of the selected pixels in Fig. 4 (a), an ordinary weighted least squares approach [15] is applied to find the coefficients (α, β, γ) of the surface function

$$Z_W = \alpha X_W + \beta Y_W + \gamma \quad (6)$$

such that the weighted sum of squared distances,

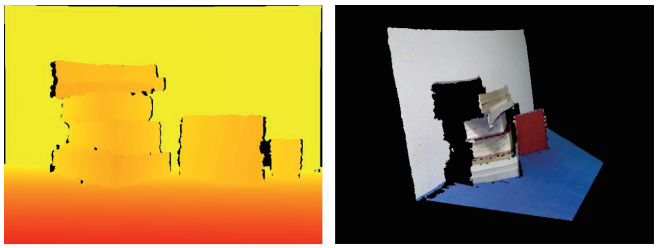
$$S(\alpha,\beta,\gamma) = \sum_{i=1}^{m} w_i(\alpha X_W^i + \beta Y_W^i + \gamma - Z_W^i)^2 \quad (7)$$

is minimal, where w_i is weight of the selected pixel and m the total number of selected pixels.

To show the performance of the fitting method, we recalculate the coordinates $(X_W, Y_W, Z_W^{'})$ of the selected pixels, where $Z_W^{'} = \alpha X_W + \beta Y_W + \gamma$. By comparing the distributions in Fig. 4 (a) and Fig. 4 (b), we can see that the fitting result effectively approximates the real distribution.

2.4. Depth Recovery

The final step of the proposed method is to recover the missing depth values according to the surface function. For a pixel belongs to *R*, its coordinates *(x,y)* in the image coordinate system are known. According to the transformation matrix in (4), each value of the coordinates (X_W, Y_W, Z_W) in the world coordinate system can be represented by a linear equation with its unknown depth value *d(x,y)*. By applying the coordinates (X_W, Y_W, Z_W) to the surface function (6), missing depth values of the whole region can be recovered.



(a) (b)

Fig. 5. (a) Recovered depth map by the proposed method. (b) Reconstructed point cloud based on the depth map (a).

Fig. 5 shows the depth map processed by the proposed method and the corresponding point cloud. Comparing the depth map and point cloud in Fig. 5 with that in Fig. 1 (c) and (d), we see that the depth values in Fig. 5 (a) change gradually and the constructed point cloud in Fig. 5 (b) is more realistic. We also see that the angle between the recovered surface and the background plane is slightly larger than 90 degrees. This is mainly influenced by the accuracy of the depth map.

3. EXPERIMENTAL RESULTS

In this section, we conduct contrast experiments on depth maps obtained by both structured light (Kinect #1) and ToF

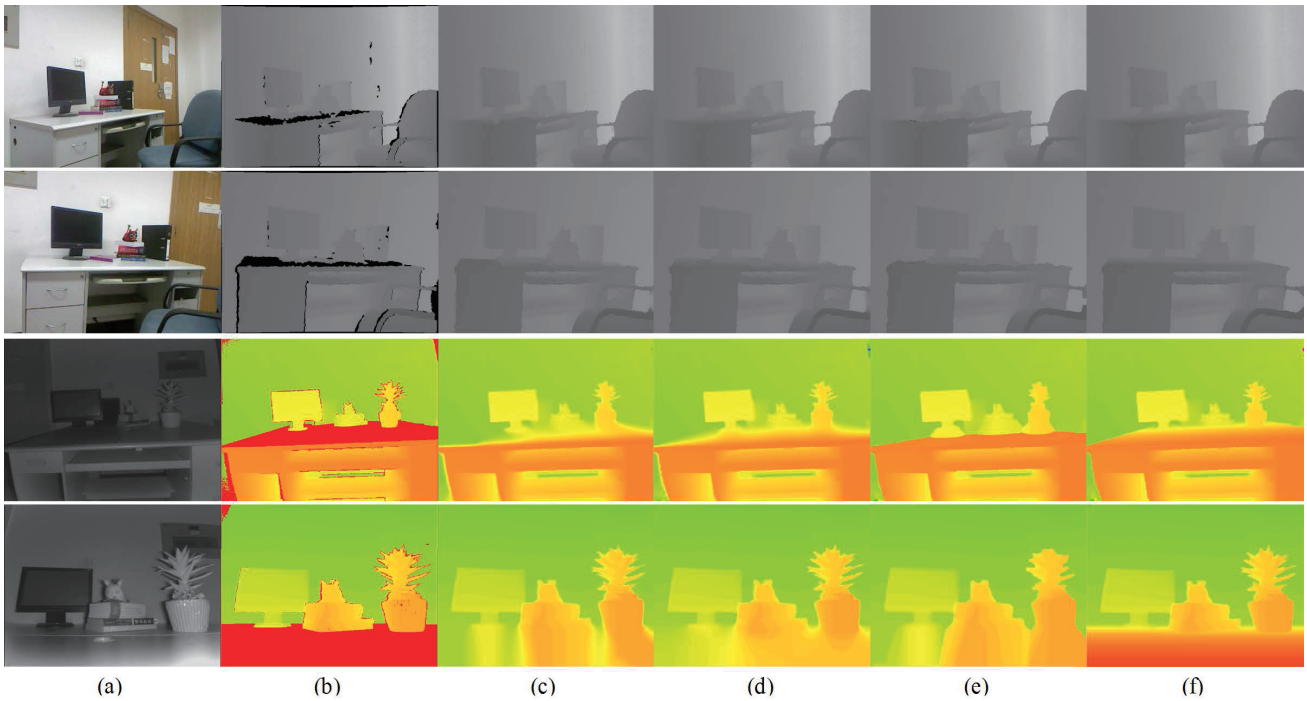


Fig. 6. Experimental results of different scenes for Kinect #1 (rows 1-2) and Kinect #2 (rows 3-4): (a) Color images (Kinect #1) and infrared images (Kinect #2). (b) Depth maps captured by depth sensors (for Kinect #2, we manually mark the large-area depth missing regions and remove wrong depth values in the regions). (c) Depth maps processed by the iterative joint bilateral filter (JBF) [8]. (d) Depth maps processed by the anisotropic diffusion based method [10]. (e) Depth maps processed by the weighted mode filter [11]. (f) Depth maps obtained by the proposed method and median filter.

(Kinect #2), and compare results from different depth recovery methods.

Fig. 6 shows experimental results from different depth recovery methods. For Kinect #1, color images in Fig. 6 (a) can be used as guidance for JBF [8], AD [10] and WMF [11], and depth maps contain plenty of depth missing regions. For small depth missing regions, depth values can be effectively recovered by JBF, AD and WMF. However, in JBF, AD and WMF, since only few edge pixels that share similar RGB information with the large-area depth missing regions are available, recovered depth values are badly influenced by local pixels in the neighborhood. Besides, in JBF and AD, edge regions between large-area depth missing region and neighbor regions become blur. For Kinect #2, since the resolution and viewpoint of the color image are different from that of the depth map, we use infrared images as guidance and manually mark the large-area depth missing regions. Though infrared images might influence the performances of JBF, AD and WMF, the main reason that limits their performances is because the number of available neighbor pixels for filling large-area depth missing regions is quite limited. In Fig. 6, depth maps processed by JBF, AD and WMF are wrong, and the depth values between large-area depth missing regions and neighbor regions are discontinuous.

In our method, we set higher K in (2) to make full use of RGB information for Kinect #1, while for Kinect #2 we set lower K to reduce the influence brought by infrared images. Besides, for Kinect #2, only the contour map of the large-area depth missing region is used for candidates selection. Thus, reliable edge pixels can be selected to approximate surface functions, and depth maps obtained by our proposed method are more accurate. In Fig. 6 (f), recovered depth values along edges of the large-area depth missing regions are continuous, and recovered values in the regions are smooth.

4. CONCLUSION

In this paper, we propose a large-area depth recovery method for both structured light and ToF. Different from traditional depth map recovery algorithms, we approximate the surface function of the region by selecting reliable edge pixels. Using the surface function as guidance, missing depth values can be effectively recovered. Experimental results show that our method gives good performance on large-area depth recovery. We point out that accurate depth maps can be obtained by the combination of our proposed method and traditional depth enhancement methods. As for future work, we plan to migrate from our current texture-dependent approach to texture-independent methods.

5. REFERENCES

[1] A. Smolic, K. Mueller, N. Stefanoski, J. Ostermann, A. Gotchev, G. B. Akar, G. Triantafyllidis, and A. Koz, "Coding algorithms for 3DTVła survey," *IEEE Trans. Circuits Syst. Video Technol.*, 17(11): 1606-1621, 2007.

[2] A. Saxena, S. H. Chung, and A. Y. Ng, "3-d depth reconstruction from a single still image," *Int. J. Comput. Vis.*, 76(1): 53-69, 2008.

[3] W. Li, J. Zhou, B. Li, and M. I. Sezan, "Virtual view specification and synthesis for free viewpoint television," *IEEE Trans. Circuits Syst. Video Technol.*, 19(4): 533-546, 2009.

[4] B. Bartczak, and R. Koch, "Dense depth maps from low resolution time-of-flight depth and high resolution color views," in *Proc. of Springer Advances in visual computing*, 2009, pp. 228-239.

[5] J. H. Cho, S. Y. Kim, Y. S. Ho, and K. H. Lee, "Dynamic 3D human actor generation method using a time-of-flight depth camera," *IEEE Trans. Consum. Electron.*, 54(4): 1514-1521, 2008.

[6] X. Zhang, Y. Li, and L. Zhu, "Color code identification in coded structured light," *Appl. Opt.*, 51(22): 340-5356, 2012.

[7] H. J. W Spoelder, F. Vos, E. Petriu, and F. Croen, "A study of the robustness of pseudorandom binary-array-based surface characterization," *IEEE Trans. Instrum. Meas.*, 47(4): 833-838, 1998.

[8] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," *ACM TOG*, 21(3): p. 96, 2007.

[9] Q. Yang, R. Yang, J. Davis, and D. Nistr, "Spatial-depth super resolution for range images, in *Proc. CVPR*, 2007, pp. 1-8.

[10] J. Liu, and X. Gong, "Guided Depth Enhancement via a Anisotropic Diffusion," in *Proc. PCM*, 2013, pp. 408-417.

[11] D. Min, J. Lu, and Do, M.N., "Depth Video Enhancement Based on Weighted Mode Filtering," *IEEE Trans. Image Process.*, 21(3): 1176-1190, 2012.

[12] J. Chen, T.N. Pappas, A. Mojsilovic, and B. Rogowitz, "Adaptive perceptual color-color image segmentation," *IEEE Trans. Image Process.*, 14(10): 1524-1536, 2005.

[13] H. Choi, and R.G. Baraniuk, "Multiscope image segmentation using wavelet-domain hidden Markov models," *IEEE Trans. Image Process.*, 10(9): 1309-1321, 2001.

[14] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11): 1330-1334, 2000.

[15] R. Scitovski, Š. Ungar and D. Jukić, "Approximating surfaces by moving total least squares method," *Appl. Math. Comput.*, 93(2): 219-232, 1998.