Chapter 6
Kinect Depth Recovery Based on Local Filters and Plane Primitives

M.A. Esfahani and H. Pourreza

6.1 Introduction

These days RGB-D cameras, especially Kinect (introduced by Microsoft in 2010), is providing depth map besides the color image of the capturing point of view by triangulating specific infrared patterns [FrEtAl13]. This new feature is beneficial for wide number of problems in the area of Computer Vision, especially for mobile robots to understand the scene and improve their knowledge about its geometry. To create an accurate road map from the input RGB-D data collected by the mobile robots, a significant constrain is to have an accurate depth map which helps to have a better understanding of the desired scene. Having an accurate depth map as input is also an important point in wide number of other problems [ZhEtAl17, ChEtAl16].

Captured depth map using Kinect sensor suffers from both holes and invalid measurements called noise. Holes are the pixels that depth sensor was unable to compute any depth value for them; because of the lighting conditions or being a glass or mirror in front of the IR camera. Invalid measurements which are mostly called as noise in the literature are also involved in the captured depth map due to the lightning condition, the way that the IR pattern is reflecting to the camera, the properties of the object surface that IR pattern is facing with, and finally lacking in calibration and measurement of disparities. It is also important to notify that the value of noise increases according to the distance exponentially (Figure 6.1).

Overall, the problem of depth recovery breaks down into two parts of depth hole filling and fixing invalid measurements or briefly called denoising. To visualize the problem and get familiar with this issue, Figure 6.2 exemplifies holes in a depth map which captured by a Kinect sensor. In the presented depth map, brown points are holes and no value is measured for them. There exist also invalid measurements

M.A. Esfahani • H. Pourreza (✉)
Ferdowsi University of Mashhad, Mashhad, Iran
e-mail: Mahdi.Abolfazli@stu-mail.um.ac.ir; hpourreza@um.ac.ir

C. Constanda et al. (eds.), *Integral Methods in Science and Engineering, Volume 2*,
DOI 10.1007/978-3-319-59387-6_6

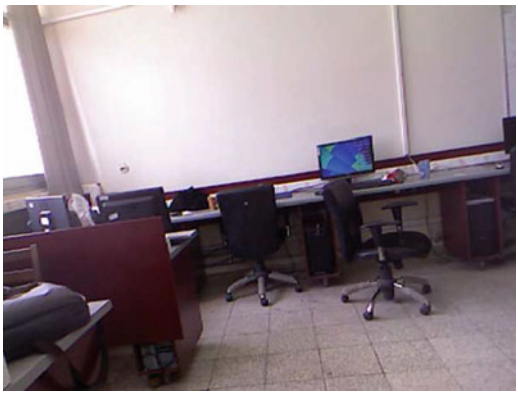

Fig. 6.1 RGB image captured with Kinect sensor

Fig. 6.2 Depth image captured with Kinect sensor

in the illustrated depth map. Figure 6.1 is the correspondence RGB image for the captured depth map and it is obvious that its resolution is more than the output of the depth sensor. The RGB camera of the Kinect sensor has the resolution of 640x480 while its depth resolution is 320x240.

Mentioned properties of the Kinect sensor and also its characteristic make it critical to remove both the invalid measurements and also filling the holes without estimated depth value. To that, recent works [YaEtAl07, DoEtAl10] applied Bilateral Filters (BF) [ToMa98] on the depth map to reduce the noise of inaccurate measurement. Sudden changes on the border of objects help BF to determine a subjective area and be able to estimate a reliable depth value for that. While it focuses on the border of objects, it is not recommended for denoising depth maps that contain high number of holes, since holes also describe a type of border to it.

To overcome with this issue, using the correspondence color image is recommended. Most of the recent works [ChEtAl12, RiEtAl12, CaSa12] used Joint Bilateral Filter (JBF) [KoEtAl07] to add properties of color image into their computations as a guidance. Despite fixing the problem of existence holes and its high performance, it works worst in the areas where the foreground and background have same color attributes. Chen et al. [ChEtAl15, ChEtAl13] formulated the

problem as an energy minimization function that merges behavior of BF, JBF, and also Joint Trilateral Filter (JTF) [LiEtAl10]. While their method performs well, they have not included the rich information of the scene structure in their minimization function which is the focus of this chapter and helps to have a better understanding about the effect of pixels on each other. Each of the mentioned filters describes and analyzes in the next steps in detail.

To focus on such filters and see how they work envisage that image I and its correspondence depth map Z exists. Hence, the recovered depth value for each pixel in the depth map describes as

$$Z' = \sum_{j \in \Omega_i} \alpha_{ij} Z(j).$$

where Ω_i is set of points with valid depth which are neighbor of pixel i and α_{ij} is the normalized weight that shows the effect of pixel j on pixel i and defines as

$$\alpha_{ij} = \frac{\beta_{ij}}{\sum_{j \in \Omega_i} \beta_{ij}}$$

where the weight β_{ij} is

$$\beta_{ij} = \begin{cases} G_S(i,j)G_Z(i,j), for\ BF \\ G_S(i,j)G_I(i,j), for\ JBF \\ G_S(i,j)G_I(i,j)G_Z(i,j), for\ JTF \end{cases}$$

and defines according to the type of the filter that system is using. In this equation, G_S, G_I, and G_Z are probability density functions and mostly define as Gaussian probability density function in spatial, color, and depth domain, respectively. Each of these shows the pairwise effect of each pair of pixels in each of the spaces. Since all of the introduced filters are local, their value for the center pixel, called β_{ij}, is equal to 1. According to relation of distance and noise subject to the Kinects' characteristic, this value is too large for the center pixel and effects worst. Adaptive methods also introduced to handle this issue, but they have not achieved grateful results [ChEtAl15, ChEtAl13].

Using each of the BF, JBF, and JTF filters benefits us to understand the scene in a specific manner. To use the pros of all of the introduced filters and reduce effect of their cons and limitations, a minimization framework that merges all these introduces. In this framework, effective features of different filters come together and combine with the structure of planes that model the scene. Using structure of the scene helps to have an initial guess for the holes and also reduces the measurement noise, since points in the 3D coordinate are standing near each other in a meaningful way and planes describe that well. Rest of the chapter is going to discuss about the way planes of the scene extracts and models the scene, the energy minimization function based on the structure of the scene, and comparing its results with the result of basic filters.

6.2 Proposed Method

In order to benefit from the structure of the scene in depth recovery and formulate that, this part goes towards modeling the structure of the scene using primitives. Efficient Ransac [ScEtAl07] is a method that helps in this process. In this chapter, an efficient modified version of that uses: Parallel RANSAC [CoEtAl15], which extracts planes of the desired scene using normal vector map of the input point cloud. Since it uses normal vectors to extract independent planes, it is possible to run this method parallel and benefit from the high speed of Graphical Processing Unit (GPU), and get a higher probability of best plane extraction by increasing number of iterations.

RANdom SAmple Consensus, briefly called RANSAC, classifies input data into two classes of inliers and outliers iteratively. In each iteration, it selects subset of data points and fits a model, e.g., line or plane, on them. The final result of the iterative RANSAC is the model that fits high number of inliers. Since RANSAC works iteratively on a subset of data points, its probability of fitting an accurate model improves by increasing its number of iterations. For instance, to fit a plane model, three points in the space are required. Hence, if the probability of extracting primary plane and selecting sampling point on that plane be ρ and u, respectively, the minimum number of iterations that require to fit the model calculates using Equation (6.1). For this reason, having more number of iterations the probability of fitting exact model improves.

$$N = \frac{log(1-\rho)}{log(1-u^3)}. \tag{6.1}$$

Alehdaghi et al. [FiBo81] introduced Parallel RANSAC based on GPU to extract planes, and showed that its computation is linear in order. To make RANSAC parallel, it is essential to determine independent parts that fitted model would not have any overlap with the other parts or segments. To make independent parts and extract plane model from them, it is conceivable to extract normal vectors, segment the image based on them, break down the global problem into small parts, and run RANSAC locally on each of the segmented boundaries. Segmenting according to the normal vector extracts the parts with high potential of having the same plane, since points of a plane are going to have same normal vector.

After estimating correspondent plane to each point of the desired scene, it is suitable to model the scene and determine an initial guess for all the points. Using this initial guess, the structure of the scene used as a part of the depth hole filling process. In the next step, the difference of the normal vector of each point with its neighbors included as a part of the minimization function to reduce the existence measurement noise besides the guidance of local filters. Next steps go toward formulating it using Kinects' characteristics.

There are some characteristics that neighbor pixels have in any input depth map. For instance, there exists less depth difference in the smooth areas or large

error exists in the border of objects. Combining all these characteristics together a minimization energy function which consists of a fidelity and data term could be signified. This minimization consists of two terms to combine the two characteristics mentioned above. Hence, the minimization energy function defines as

$$\min_{Z'(i)} E_r(Z'((i)) + \lambda E_d(Z'(i))$$

where E_r and E_d are the regularization term and data term, respectively, and Z' presents the recovered depth map. λ is a trade-off factor between data and regularization term. This minimization function was firstly introduced by Chen et al. [ChEtAl15, ChEtAl13]. In the next steps we are going to define the properties of the both regularization and fidelity terms and include the effect of the scene structure in computations.

Data term includes the fact that accuracy of measured depth decreases as the distance between the object and Kinect sensor increases, and also the fact that states the depth on the smooth areas of objects is reliable and is unreliable on their boundary. According to these, the data term defines as

$$E_d(Z'(i)) = \frac{1}{2} \sum_{i \in \Omega_d} w_i (Z'(i) - Z(i))^2$$

where Ω_d is the subset of points with a valid measured depth values. This equation goes toward minimizing the weighted squared difference between the recovered depth value and the original one according to their information quality. In this part, the weight w plays grate rule and defines as

$$w_i = \frac{Z_{max}^2 - Z_{avg_i}^2}{Z_{max}^2 - Z_{min}^2}$$

to have more focus on the reliable depth values which are nearer to the Kinect sensor. In this equation, Z_{max} and Z_{min} are the max and min distance that Kinect sensor can measure and Z_{avg_i} is average depth of boundary around pixel i with reliable and valid depth values. Beside the mentioned characteristics of Kinect sensor, it is important to include the point that difference between a point and its neighbor in a smooth region is too small. Hence, the regularization term defines as

$$E_r(U(i)) = \frac{1}{2} \sum_{i \in \Omega_s} \sum_{j \in \Omega_i} w_{ij} (U(i) - U(j))^2$$

where Ω_s is the subset of points with valid neighborhood and Ω_i is each of the neighbors of pixel i in that subset. w_{ij} plays an important rule to classify similar and dissimilar pixels subject to their region; it checks that by locating boundaries with sudden changes using different types of information.

While neighboring pixels have to be similar with low difference, they have to be dissimilar in the sudden changes of depth and colors where an edge exists. The coefficient w_{ij} controls this behavior by considering color and depth images. It defines as the normalized coefficient

$$w_{ij} = \frac{\beta_{ij}}{\sum_{j \in \Omega_i} \beta_{ij}}$$

where

$$\beta_{ij} = \begin{cases} G_S(i,j)G_I(i,j) & i \notin \Omega_d, j \in \Omega_i \\ G_S(i,j)G_I(i,j)G_Z(i,j)G_N(i,j) & i \in \Omega_d, j \in \Omega_i \end{cases}$$

and Ω_d states pixels with valid depth value and Ω_i defines subset of pixels who are neighbor to pixel i. As mentioned, G_S , G_I, and G_Z are the probability density function in the spatial, color, and depth domain. Beside these parameters which focus on the depth and color images independently, G_N applies the theorem that difference between normal vectors of the points that are on a same plane has to be minimum. The mentioned probability density functions define as

$$G_S = exp(\frac{-||i-j||^2}{\sigma_S^2})$$

$$G_I = exp(\frac{-||I(i)-I(j)||^2}{\sigma_I^2})$$

$$G_Z = exp(\frac{-||Z(i)-Z(j)||^2}{\sigma_Z^2})$$

$$G_N = exp(\frac{-||N_Z(i)-N_Z(j)||^2}{\sigma_N^2})$$

with variances σ_S, σ_I , σ_Z, and σ_N. I is intensity, Z is the depth, and N_Z is the normal vector of each pixel. The variance controls the effective area of similarity in each of the spaces. In sum, B_{ij} describes the pairwise relation between pixels i and j. This weight includes the difference of normal vectors when there exists valid depth value for pixel i and helps to include structure of the scene in our computations.

6.3 Experimental Results

The presented method is implemented and tested under the linux OS using OpenCV and Point Cloud Library (PCL). To evaluate the results of the proposed method, Middlebury datasets [Mi] that simulates Kinects' characteristic is used. In the problem of denoising Kinect depth map, a ground truth of the depth map is required

Table 6.1 Comparing Mean Absolute error of the proposed method with Chen et al. [ChEtAl15] on the Middlebury datasets

Dataset	Chen et al. (JTF) [ChEtAl15]	Proposed Method
Art	0.0073	0.0050
Book	0.0110	0.0087
Doll	0.0064	0.0043
Laundry	0.0229	0.0225
Moebius	0.045	0.044
Reinder	0.0061	0.0037

Fig. 6.3 A simulated depth input from the Middlebury dataset (Black points are the holes and no value exists for them)

Fig. 6.4 Recovered depth map for Figure 6.3

to figure out accuracy of the hole filling and denoising. Table 6.1 illustrates the Mean Absolute Error (MAE) of the depth recovery on Middlebury datasets.

According to the reported results, including structure of the scene in computations using plane primitives helps to reduce the MAE and have a better understanding of the scene. This reduction is due to the characteristic of selecting a better supporting regions for pixels in depth map and giving a more realistic pairwise weights using the normal vector of supporting plane of pixels. To have a better comparison, Figure 6.3 is an input depth map with a number of holes on it and Figure 6.4 shows the result of applying the proposed method on that input depth map.

To have a better comparison, Figure 6.5 shows a part of the result of applying Chen et al. [ChEtAl15] method on Figure 6.3, and Figure 6.6 shows result of the

Fig. 6.5 Focusing on a part of the recovered depth map of Figure 6.3 using Chen et al. [ChEtAl15] method

Fig. 6.6 Focusing on a part of the recovered depth map of Figure 6.3 using the proposed method

proposed method applied on that part. It illustrates that using structure of the scene besides the information that extracts from local filters helps to extract edges of depth map accurately. Comparing Figure 6.7 and Figure 6.8 also shows that the proposed method is able to detect holes in the ring and fix that parts. Since the points in the hole of the ring are not in the sample plane of the ring, they will not have effected by the points that are on the ring using the proposed method.

Figure 6.9 shows another depth input and results of applying Chen et al. and proposed method are illustrated in Figure 6.10 and Figure 6.11, respectively. Comparing the outputs, it is again clear that our method performs well on edges and keeps them by looking at the scene structure, while Chen et al. [ChEtAl15] method blurs the edges.

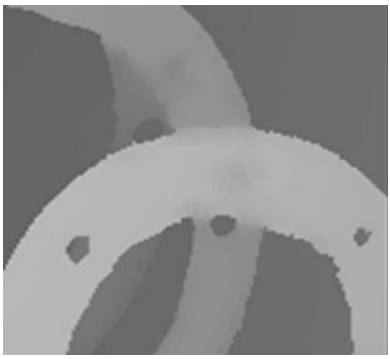

Fig. 6.7 Focusing on a part of the recovered depth map of Figure 6.3 using Chen et al. [ChEtAl15] method

Fig. 6.8 Focusing on a part of the recovered depth map of Figure 6.3 using the proposed method

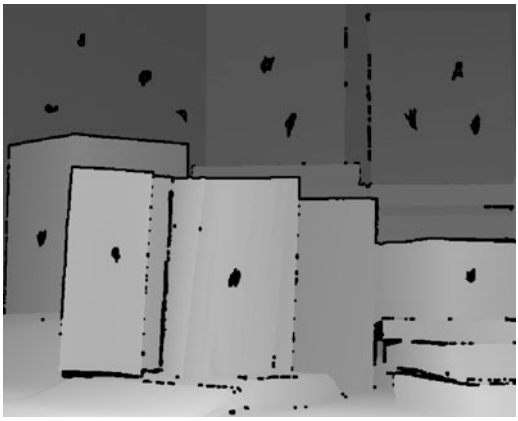

Fig. 6.9 A simulated depth input from the Middlebury dataset (black points are the holes and no value exists for them)

6.4 Conclusion

In this chapter, a novel approach for Kinect depth recovery based on both scene structure and the guidance of local filters based on color image and depth map is presented. Modeling scene structure using planes helps to get an initial guess for points with damaged or unknown depth value. Analyzing results shows that our

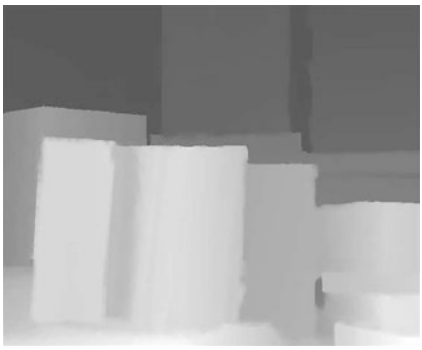

Fig. 6.10 Focusing on a part of the recovered depth map of Figure 6.7 using Chen et al. [ChEtAl15] method

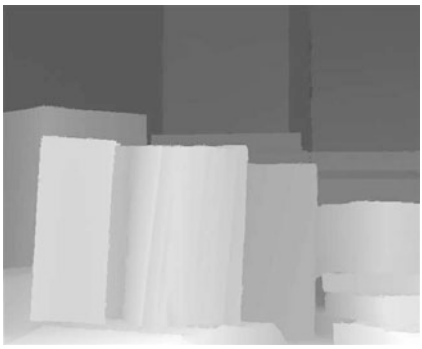

Fig. 6.11 Focusing on a part of the recovered depth map of Figure 6.7 using the proposed method

method is able to keep edges and also detects supporting regions of similar pixels perfectly. As the future work, we are going to model the scene using some other primitives like sphere and also benefit from deep Convolutional Neural Networks (CNN) to understand the model of both RGB image and the depth map.

References

[CaSa12] Camplani, M., Salgado, L.: Efficient spatio-temporal hole filling strategy for kinect depth maps. In: IST/SPIE Electronic Imaging, SPIE Proceedings, vol. 8290, pp. 82900E–82900E. International Society for Optics and Photonics (2012)

[ChEtAl12] Chen, L., Lin, H., Li, S.: Depth image enhancement for Kinect using region growing and bilateral filter. In: 21st International Conference on Pattern Recognition (ICPR), pp. 3070–3073. IEEE (2012)

[ChEtAl13] Chen, C., Cai, J., Zheng, J., Cham, T.J., Shi, G.: A color-guided, region-adaptive and depth-selective unified framework for Kinect depth recovery. In: IEEE 15th International Workshop on Multimedia Signal Processing (MMSP), pp. 007–012. IEEE (2013)

[ChEtAl15] Chen, C., Cai, J., Zheng, J., Cham, T.J., Shi, G.: Kinect depth recovery using a color-guided, region-adaptive, and depth-selective framework. ACM Trans. Intell. Syst. Technol. (TIST) **6**(2), 12 (2015)

[CoEtAl15] Alehdaghi, M., Esfahani, M.A., Harati, A.: Parallel RANSAC: speeding up plane extraction in RGBD image sequences using GPU. In: 5th International Conference on Computer and Knowledge Engineering (ICCKE), pp. 295–300. IEEE (2015)

[ChEtAl16] Chen, X., Kristian H., Yinhai, W.: Kinect–based pedestrian detection for crowded scenes. Comput. Aided Civ. Inf. Eng. **31**(3), 229–240 (2016)

[DoEtAl10] Dolson, J., Baek, J., Plagemann, C., Thrun, S.: Upsampling range data in dynamic environments. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1141–1148. IEEE (2010). ISO 690

[FiBo81] Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)

[FrEtAl13] Freedman, B., Shpunt, A., Machline, M., Arieli, Y.: U.S. Patent No. 8,493,496. U.S. Patent and Trademark Office, Washington, DC (2013)

[KoEtAl07] Kopf, J., Cohen, M.F., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. ACM Trans. Graphics (TOG) **26**(3), 96 (2007)

[LiEtAl10] Liu, S., Lai, P., Tian, D., Gomila, C., Chen, C.W.: Joint trilateral filtering for depth map compression. In: Visual Communications and Image Processing, SPIE Proceedings, vol. 7744, pp. 77440F–77440F. International Society for Optics and Photonics (2010). ISO 690

[Mi] Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007). Minneapolis (2007)

[RiEtAl12] Richardt, C., Stoll, C., Dodgson, N.A., Seidel, H.P., Theobalt, C.: Coherent spatiotemporal filtering, upsampling and rendering of RGBZ videos. Comput. Graph. Forum **31**(2pt1), 247–256 (2012)

[ScEtAl07] Schnabel, R., Wahl, R., Klein, R.: Efficient RANSAC for point cloud shape detection. Comput. Graphics Forum **26**(2), 214–226 (2007)

[ToMa98] Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: Sixth International Conference on Computer Vision, pp. 839–846. IEEE (1998)

[YaEtAl07] Yang, Q., Yang, R., Davis, J., Nistér, D. Spatial-depth super resolution for range images. In: Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2007)

[ZhEtAl17] Zhu, M., Canjun, Y., Wei, Y., Qian, B.: A Kinect-based motion capture method for assessment of lower extremity exoskeleton. In: Wearable Sensors and Robots, pp. 481–494. Springer, Singapore (2017)