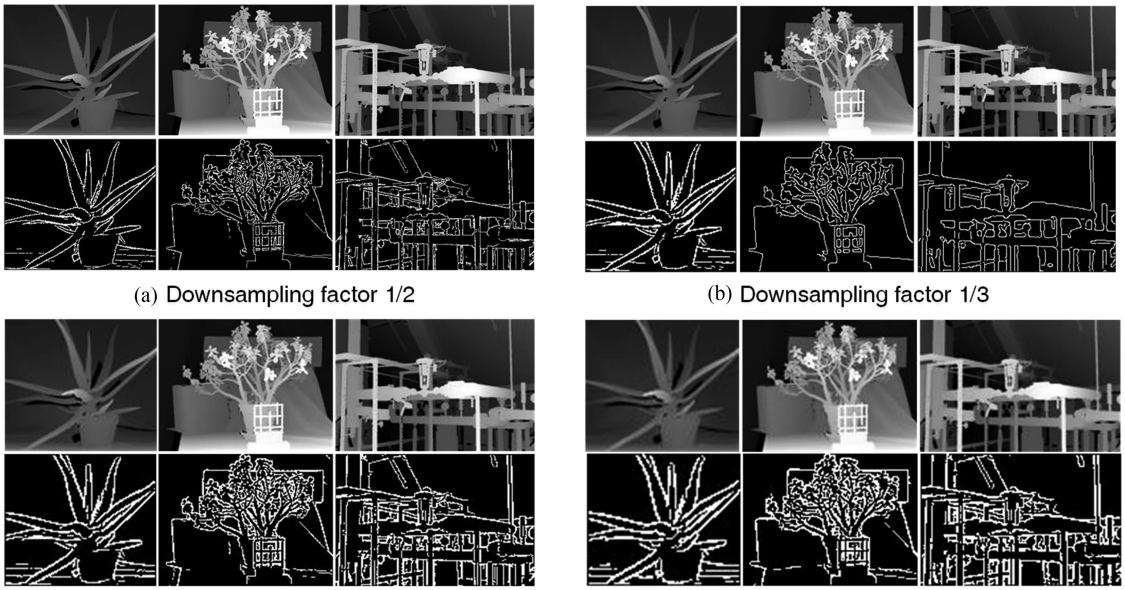
IEEE Photonics Journal
An IEEE Photonics Society Publication

Open Access

LBP-Based Edge Detection Method for Depth Images With Low Resolutions

Volume 11, Number 1, February 2019

Xinyu Wang
Jie Cao
Qun Hao
Kaiyu Zhang
Zihan Wang
Saad Rizvi



(a) Downsampling factor 1/2

(b) Downsampling factor 1/3

(c) Downsampling factor 1/4

(d) Downsampling factor 1/5

DOI: 10.1109/JPHOT.2018.2884772
1943-0655

LBP-Based Edge Detection Method for Depth Images With Low Resolutions

Xinyu Wang,[1] **Jie Cao**,[1,2] **Qun Hao**,[1] **Kaiyu Zhang**,[1] **Zihan Wang**,[1] **and Saad Rizvi** [1]

[1]School of Optics and Photonics, Beijing Institute of Technology, Key Laboratory of Biomimetic Robots and Systems, Ministry of Education, Beijing 100081, China
[2]NUS Suzhou Research Institute, Suzhou 215123, China

DOI:10.1109/JPHOT.2018.2884772

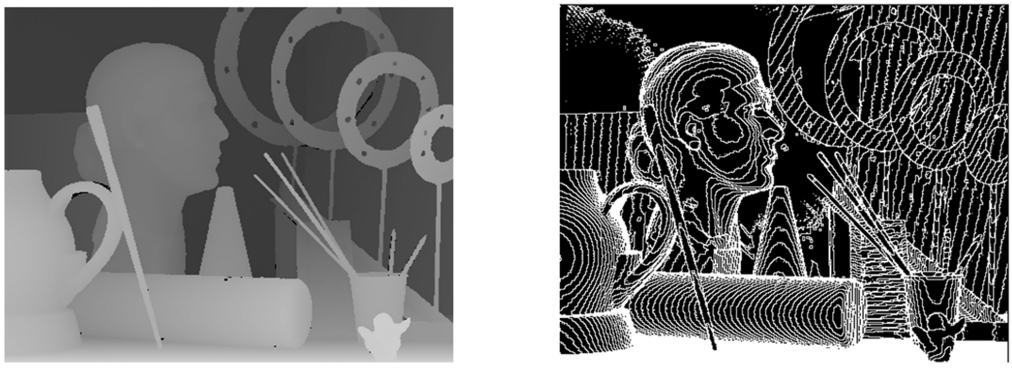
Manuscript received November 6, 2018; revised November 25, 2018; accepted November 27, 2018. Date of publication December 3, 2018; date of current version December 28, 2018. This work was supported in part by the National Natural Science Foundation of China under Grants 61605008, 61871031, and 61875012, in part by the Natural Science Foundation of Beijing Municipality under Grant 4182058, and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20160375. Corresponding author: Jie Cao (e-mail: ajieanyyn@163.com).

Abstract: Depth edge represents contours of depth images. However, captured images often have low resolution. Furthermore, conventional edge detection methods fail to handle depth images with low resolution. To cater depth images with low resolution, we propose a novel edge detection method with high accuracy and computational speed. The proposed method modifies the local binary pattern and selects an adaptive threshold to obtain a rough edge map. Then, it removes irrelevant edges to obtain the result. Experimental results show that the proposed method can rapidly acquire accurate depth edge maps and is applicable to depth images with low resolution. The proposed method is compared with and outperforms other renowned edge detection methods.

Index Terms: Depth images, edge detection, low resolution, LBP.

1. Introduction

An efficient edge detector is necessary to the extraction of object contours from depth edges. Edge detector is significant in various depth scenes [1] such as target recognition [2], human posture recognition [3], [4], and robot navigation [5]–[7]. In contemporary applications, depth images can be acquired through the use of time-of-flight (TOF) cameras [8], [9], binocular cameras [10], structured light cameras [11] and through light detection and ranging(LIDAR) [12], [13]. However, images captured through these techniques often have low resolution mostly because of the limit in processing speed or bandwidth of the camera. For example, the resolution of Mesa Imaging's TOF camera (product model: SR4000) is 176 $\times$ 144, and the resolution of Microsoft's Kinect V1 is 320 $\times$ 240. Moreover, the resolution of a novel single photon-counting camera produced by Photon Force is only 32 $\times$ 32. Although the resolution of depth images can be enhanced using multiple methods [14]–[16], they are usually impractical for real-time application given that the processing time is great. In addition, artifacts and blur produced by these methods [17] are problematic. So it's necessary to deal with low resolution images in some cases. Low resolution images are used in many practical applications such as target detection [18], [19], facial recognition [20], pose estimation [21], autonomous exploration [22], and edge detection [23]. However, acquiring useful edge information from low resolution depth images is relatively challenging and lacks extensive



(a) (b)

Fig. 1. Original LBP operator (a) Depth image. (b) LBP texture map.

study. Therefore, the current work focuses on developing an efficient edge detection method for depth images with low resolution.

Many edge detection methods have been proposed in the past. Most of these conventional methods use gradients in the image to detect edges. For example, Lejeune *et al*. proposed a rapid jump edge detection method for 3D cameras based on the principles of canny edge detector [24]. Similarly, Fu *et al*. applied canny edge detection algorithm to locate the sky area accurately [25]. However, these methods often have poor edge detection accuracy. An alternative scheme using aligned color images to detect edges in the depth image was also proposed. Schwarz *et al*. proposed to detect edges in a depth image and corresponding aligned color image and got a fine result [26]. This method was also adopted by Camplani *et al*. [27]. However, these methods are often noise-sensitive and may introduce false edges from color images. Recently, learning-based methods gained significant attention from the research community. For instance, Ferstl *et al*. proposed to learn a dictionary of edge priors from an external database of high and low resolution samples [28]. Similarly, Xie *et al*. proposed to learn the high resolution edge map from the edge map of the upsampled low resolution depth image [29]. Although these methods work effectively, they are often complex and require substantial processing time.

In summary, the existing edge detection methods are often accompanied by low accuracy and high processing time for depth images with low resolution. To counter these problems, an efficient edge detection method is proposed with high accuracy and computational speed for depth images with low and high resolutions. In the proposed method, the local binary pattern (LBP) is modified and an adaptive threshold is selected to obtain a rough edge map. Then, all irrelevant edges are removed to obtain the result. The proposed method is exhaustively tested for its performance by setting up several experiments.

2. Method

For the depth image, the pixel values that belong to the same plane change continuously, generally increasing or decreasing towards a specific direction. However, the depth values of pixels belonging to different planes change discontinuously and often show large changes. The discontinuities in an image are the edges. This characteristic of depth image indicates that the texture information can be used to detect edges. For this purpose, LBP operator is selected due to its low computational complexity and high sensitivity to details. The original LBP operator obtains a texture map from a depth image, as shown in Fig. 1. Fig. 1(a) shows the depth image, and Fig. 1(b) shows the LBP texture map of Fig. 1(a). These figures show that the LBP operator accurately detects and retains all edges. All irrelevant textures should be removed to obtain the final edge map accurately. So we modify and improve the performance of the original LBP operator for depth edge detection given that it cannot be used directly in depth edge detection.

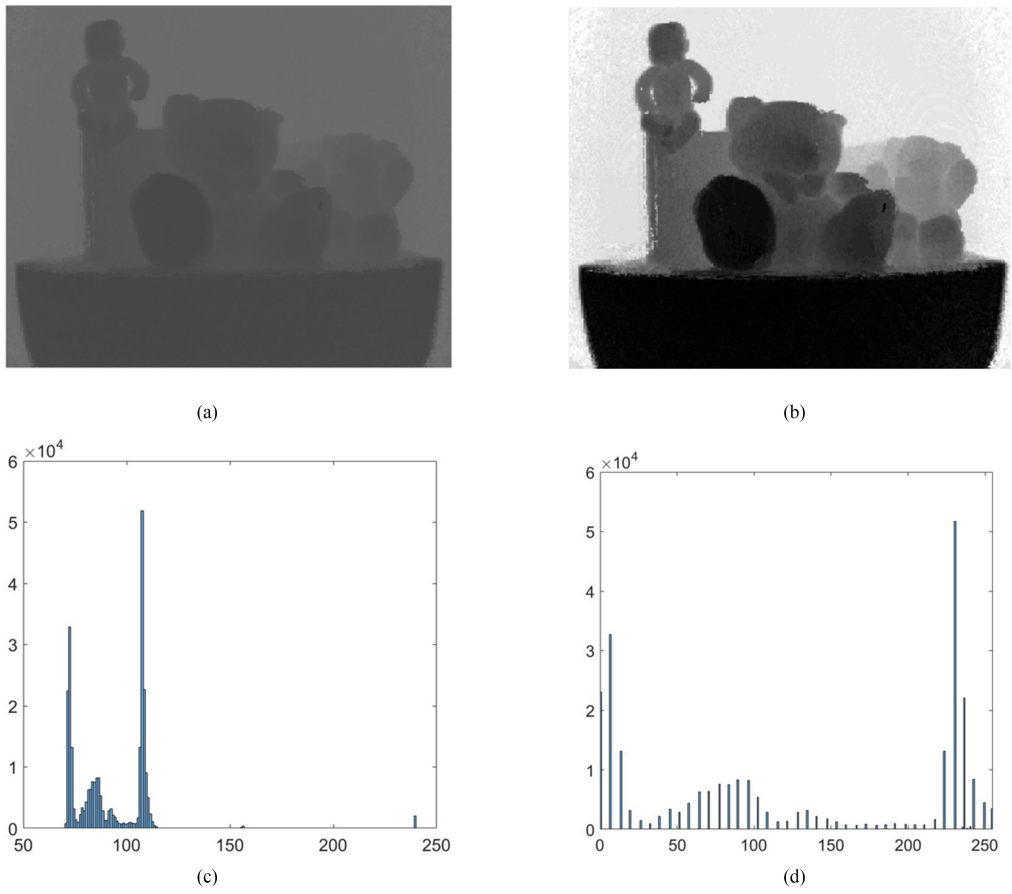


Fig. 2. Adjusting the contrast of depth image. (a) original depth image. (b) adjusting depth image. (c) grayscale histogram of (a). (d) grayscale histogram of (b).

The basic idea of the proposed method is to adjust the contrast of depth image and obtain a rough edge map through the use of modified LBP operator. Finally, all irrelevant edges are removed. The method is discussed in detail as follows.

2.1 Adjusting the Contrast of Depth Image

Depth images often have insufficient contrast. Common methods for adjusting image contrast include linear transformation and nonlinear transformation. However, nonlinear transformation may strengthen the contrast of the background excessively, linear transformation is more suitable for processing depth images. A common linear transformation is expressed as follows:

$$G = \begin{cases} 0, & i < low \\ S(i), & low \leq i \leq high \\ 255, & i > high \end{cases} \tag{1}$$

$$S(i) = \frac{255}{high - low}(i - low), \tag{2}$$

where i is the original gray value of the depth image, G is the adjusted gray value, and low and $high$ are two thresholds determined by the given depth image. The lower and upper bounds of the threshold imply that: pixels less than low threshold value accounts for 1% of the entire image and pixels more than high threshold value accounts for 1% of the entire image. The contrast of an image can be enhanced via these thresholds, as shown in Fig. 2. Figs. 2(a) and (b) show the original depth

image and the map with contrast adjustment, respectively. Figs. 2(c) and (d) are their grayscale histograms. The depth edges of Fig. 2(b) are clearer than that of Fig. 2(a), and the gray levels of the image are more evenly distributed, as shown in the comparative histograms of Figs. 2(c) and (d).

2.2 Using Modified LBP Operator

The LBP operator was introduced by Ojala *et al.* [30] for texture classification. Given a center pixel in the 3 × 3 pattern, the LBP value is obtained by comparing its gray scale value with its neighborhoods [31] with the following formula:

$$LBP(x, y) = \sum_{p=0}^{7} 2^p \cdot f(i_p - i_c) \tag{3}$$

$$f(m) = \begin{cases} 1, & if\ m \geq 0 \\ 0, & else \end{cases}, \tag{4}$$

where i_c denotes the gray value of the center pixel *(x, y)*, i_p is the gray value of its neighbors, and p stands for the number of neighbors. We modify the LBP operator to use it in depth edge detection, that is, Depth LBP (DLBP). The DLBP is defined as follows:

$$DLBP(x, y) = D\left(\sum_{p=0}^{7} L(i_p - i_c)\right), \tag{5}$$

$$L(j) = \begin{cases} 1, & if\ j \geq i_T \\ 0, & else \end{cases}, \quad i_T = i_{\max} - i_{aver}, \tag{6}$$

$$D(k) = \begin{cases} 1, & if\ k \geq 1 \\ 0, & else \end{cases}, \tag{7}$$

where $i_{\max}$ represents the maximum value of 3 × 3 pattern, and i_{aver} is the average of the 3 × 3 pattern. i_T is an adaptive threshold used to determine whether an edge exists between pixel i_c and i_p. If $i_{\max}$ is much bigger than i_{aver} in this pattern, then it can be decided that an edge may exist in this pattern if i_p is also much bigger than i_c. After using DLBP, a further judgment is required, as follows:

$$R(x, y) = \begin{cases} DLBP(x, y), & if\ |i_{aver} - i_c| \geq 1.5 \\ 0, & else \end{cases}. \tag{8}$$

If the average value of the 3 × 3 pattern is sufficiently close to the center pixel, then this pattern is the same plane, and no edge exists in this pattern. Consequently, a rough edge map is obtained by calculating *R(x, y)* of every pixel.

2.3 Remove Isolated Edges

Although a rough depth edge map is obtained by the DLBP operator, some isolated edges remain in the depth image. We use a mask to remove these edges. In general, the removal of isolated edges is high when the width of the mask is small. However, if the mask has an extremely small width, then some large isolated edges are left. After several tests and comparison, the width of the mask is optimized to a value of 5, as shown in Fig. 3. If none of the outermost pixels of the mask is an edge pixel, then the other pixels in the mask are set to zero. Similarly, if none of the second ring pixels of the mask is an edge pixel, then the center pixel is set to zero. In this approach, the isolated edges are removed, and the final depth edge image is obtained.

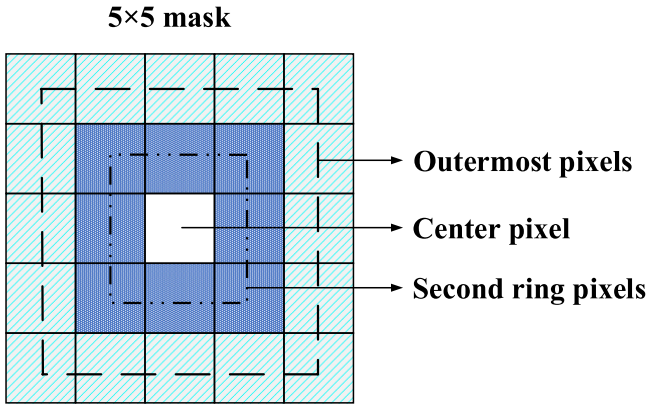


Fig. 3. 5 × 5 mask for removing isolated edges.

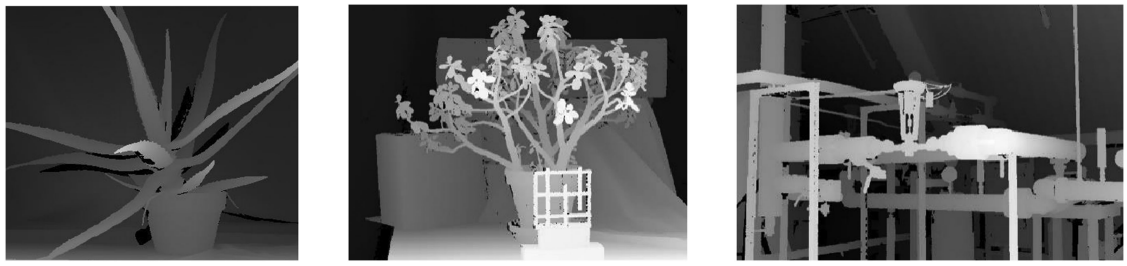


Fig. 4. Three original depth images: Aloe (427 × 370), Jade plant (659 × 497), and Pipes (735 × 485).

3. Experimental Results

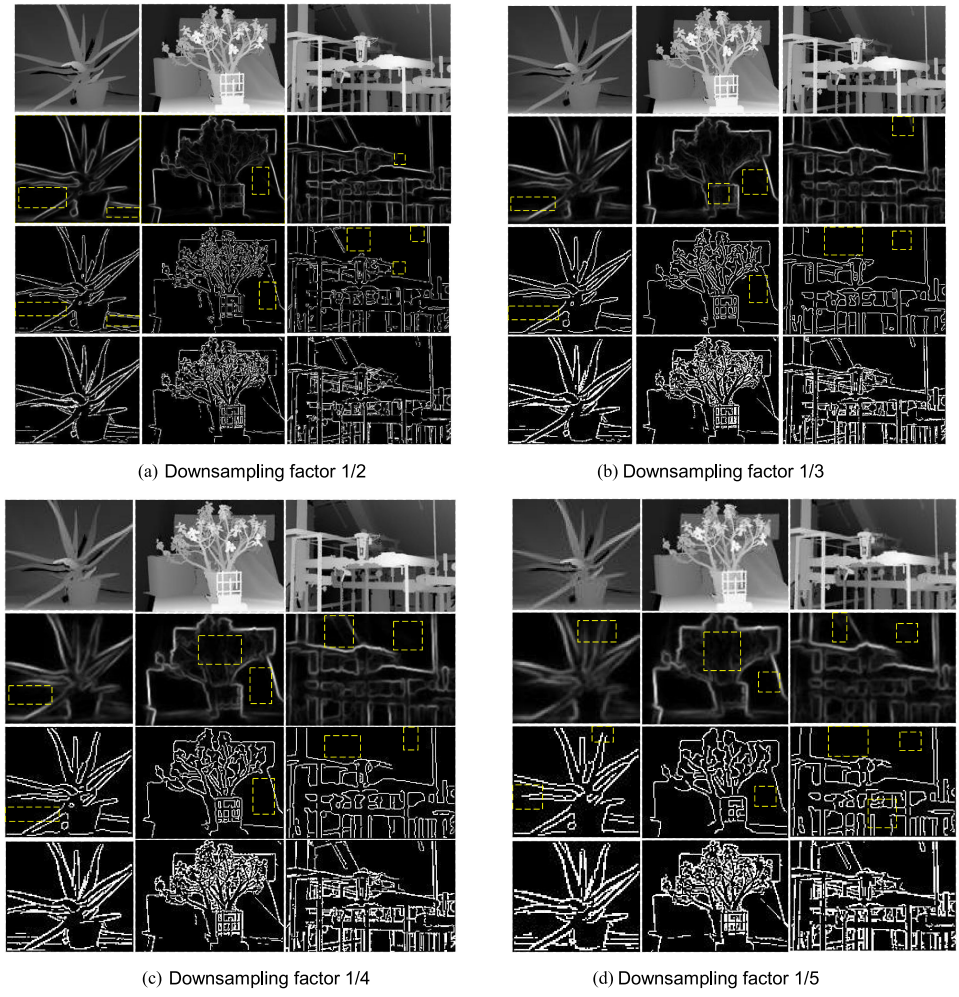
Several experiments are set to test the proposed method. First, depth images with low resolutions are tested using the proposed method and other existing renowned methods. For the evaluation of the performance, the F1 and time results are compared. Finally, the consistency of proposed method is discussed.

3.1 Edge Results on Depth Images With Low Resolutions

The proposed method is compared with the canny edge detector [32] and SE-MS method [33]. The canny edge detector is a powerful edge detection method and widely used in many applications. The SE-MS method proposed by Piotr Dollár *et al.* is capable of real-time frame rates while achieving state-of-the-art accuracy. The depth images in Figs. 4(a) to (c) are from Middleburry 2006 datasets [34] and 2014 datasets [35]. These depth images have high resolution. Then each depth image is downsampled to 1/2, 1/3, 1/4, and 1/5 for testing purposes.

The edge results are shown in Fig. 5. Figs. 5(a) to (d) are four comparative results with four down-sampling factors. The images of the first row in each figure are depth images with low resolution, whereas the images of the following rows are the edge detection results by SE-MS, Canny, and the proposed method. The proposed method performs better than other two methods under different resolutions. In the results of SE-MS, the loss of edge information is extremely high, and the details on the image can hardly be recognized. For example, in the scene “Aloe”, the edges of the leaves and flowerpot are unclear and lack details. In the case of canny edge detector, many distorted edges are observed and some edges are lost. For example, in the scene “Pipes”, the edges of many pipes are distorted and lost. On the contrary, the proposed method retains most edges and shows effective performance even under the downsampling factor of 1/5, as shown in Fig. 5.

For fair comparison, linear transformation of contrast proposed in chapter 2.1 is also performed in SE-MS and Canny to prove the effectiveness of proposed method and results are shown in Fig. 6.



(a) Downsampling factor 1/2 (b) Downsampling factor 1/3

(c) Downsampling factor 1/4 (d) Downsampling factor 1/5

Fig. 5. Each original image is downsampled to 1/2, 1/3, 1/4, and 1/5. The images on the first row are downsampled depth images, and that on the other rows are edge results by SE-MS, Canny, and the proposed method.

Compared with Fig. 5, there are no significant difference in the results of Fig. 6. As a result, the proposed method proves it has better performance than other two methods on depth images with low resolutions.

3.2 Comparison of F1 Results

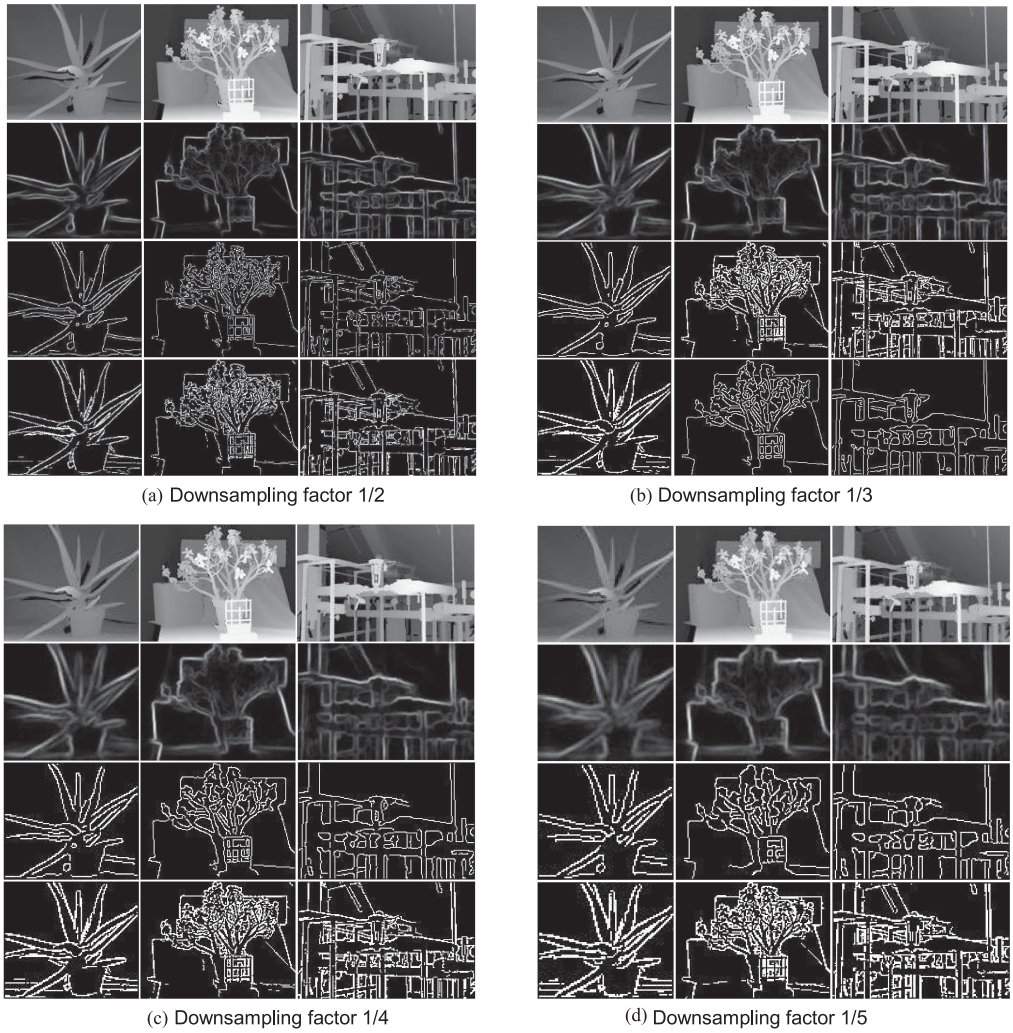
F1 is used to quantitatively evaluate the proposed method. F1 is the harmonic mean of *precision* and *recall* traditionally, as follows

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{9}$$

$$precision = \frac{TP}{TP + FP}, \tag{10}$$

$$recall = \frac{TP}{TP + FN}, \tag{11}$$

where *precision* is the number of accurate positive results divided by the number of all positive results, and *recall* is the number of accurate positive results divided by the number of positive



(a) Downsampling factor 1/2 (b) Downsampling factor 1/3

(c) Downsampling factor 1/4 (d) Downsampling factor 1/5

Fig. 6. Edge results with linear transformation of contrast. The images on the first row are downsampled depth images, and that on the other rows are edge results by SE-MS, Canny, and the proposed method.

results that should have been returned. F1 can be interpreted as a weighted average of precision and recall, and it is a synthetic indicator. Better edge result often has larger F1.

The F1 results of the proposed method and other two edge detectors are shown in Fig. 7. Evidently, the proposed method shows improved F1 results in most scenarios. For example, in the results of “Aloe”, the F1 value of the proposed method is approximately 0.5 (for different downsampling factors), whereas, the results of the other two methods are smaller than 0.5. The other two scenes also have similar results. Furthermore, compared with the other two methods, the F1 of the proposed method has insignificant change with the decrease in resolution, thereby showing better reliability than the other methods. The main reason is that the proposed method is based on the LBP algorithm, which is sensitive to details of low resolution images, and the local adaptive threshold is self-adaptive over the entire image. Although SE-MS method and canny edge detector have acceptable performance on original images, they are poor in handling low resolution depth images. Therefore, the proposed method clearly surpasses the other two methods in terms of accuracy.

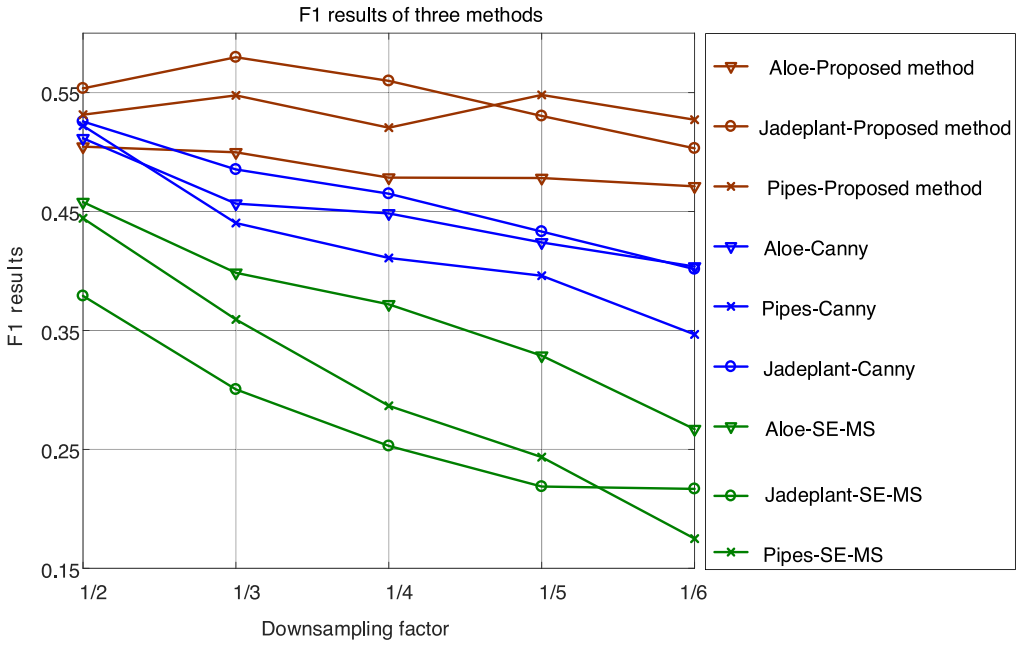


Fig. 7. F1 results of three methods.

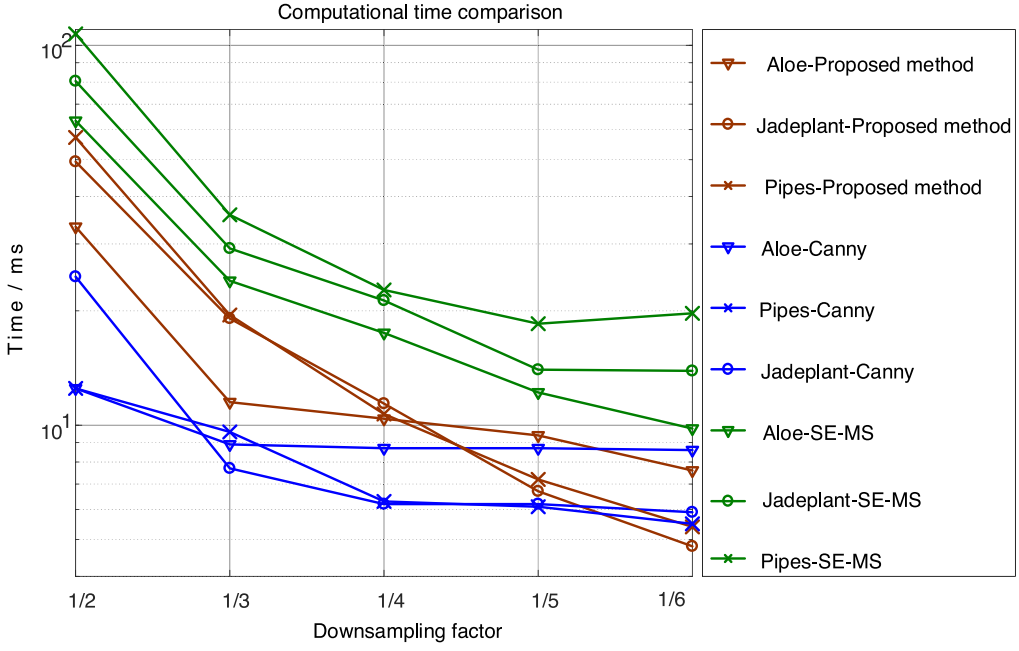


Fig. 8. Comparison of computational times.

3.3 Comparison of Computational Times

To further validate the effectiveness of the proposed method, we compare the computational times of different methods in Fig. 8. The computational time is evaluated on the platform of Intel i3-2120 CPU with 8 GB memory. All algorithms are implemented on Matlab (version of r2015a) and all other conditions are exactly the same. Fig. 8 shows that all computational times are less than 110 ms. The proposed method requires a little more time than the Canny edge detector. However, given its

Fig. 9. Edge detection results on depth images with high resolutions. The first rows are depth images with high resolutions, and the other rows are edge detection results by SE-MS, Canny, and the proposed method.

detection accuracy, this difference is insignificant. Moreover, the proposed method is faster than SE-MS. The results show that the proposed method is potential for real-time applications.

3.4 Discussion on the Consistency of Proposed Method

To further test the proposed method, high resolution depth images are also used. As shown in Fig. 9, all methods produce similar results given that depth edges are effectively detected by these methods. Moreover, the detected edges are clear and accurate. Furthermore, the results of the proposed method on depth images with high resolution are similar to those with low resolution, thereby confirming that the proposed method has a consistent performance on depth images with high and low resolutions.

4. Conclusions

A novel edge detection method applicable to depth images with low resolution is proposed. We modify the LBP, select an adaptive threshold to obtain a rough edge map, and further remove irrelevant edges to obtain the result. The F1 result of the proposed method is approximately 0.5, whereas the results of the other two methods (SE-MS and Canny) are smaller than 0.5. Furthermore, the computational time of the proposed method is less than 110 ms, making it suitable for real-time applications. Experimental results indicate that the proposed method is more effective than the traditional methods in detecting depth edges on depth images with low resolution. The advantages of this method include simplicity, rapid processing, and accuracy. It can be used for applications,

such as image segmentation, target recognition, and resolution enhancement. Future work will involve performance evaluation for the proposed method in real-life depth scenes.

References

[1] S. M. A. Hasan and K. Ko, "Depth edge detection by image-based smoothing and morphological operations," *J. Comput. Des. Eng.*, vol. 3, no. 3, pp. 191–197, 2016.

[2] F. Chen and W. Wang, "Target recognition in clutter scene based on wavelet transform," *Opt. Commun.*, vol. 282, no. 4, pp. 523–526, 2009.

[3] J. Song, L. Liu, W. Huang, Y. Li, X. Chen, and Z. Zhang, "Target detection via HSV color model and edge gradient information in infrared and visible image sequences under complicated background," *Opt. Quantum Electron.*, vol. 50, no. 4, pp. 175–187, 2018.

[4] C. Shi *et al.*, "SCECam: A spherical compound eye camera for fast location and recognition of objects at a large field of view," *Opt. Express*, vol. 25, no. 26, pp. 32333–32345, 2017.

[5] J. W. Dong and H. M. Lv, "A new method of extraction of mobile robot navigation line based on edge detection," *Adv. Mater. Res.*, vol. 143–144, pp. 482–486, 2011.

[6] Z. Xian, X. He, J. Lian, X. Hu, and L. Zhang, "A bionic autonomous navigation system by using polarization navigation sensor and stereo camera," *Auton. Robots*, vol. 41, no. 5, pp. 1–12, 2017.

[7] Y. Xie, W. Wang, J. Guo, and K. M. Lee, "Edge detection using structured laser pattern and vision for mobile robot navigation," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, 2011, pp. 910–915.

[8] F. Villa *et al.*, "SPAD smart pixel for time-of-flight and time-correlated single-photon counting measurements," *IEEE Photon. J.*, vol. 4, no. 3, pp. 795–804, Jun. 2012.

[9] F. Li *et al.*, "CS-ToF: High-resolution compressive time-of-flight imaging," *Opt. Express*, vol. 25, no. 25, pp. 31096–31110, 2017.

[10] Z. Jia *et al.*, "Improved camera calibration method based on perpendicularity compensation for binocular stereo vision measurement system," *Opt. Express*, vol. 23, no. 12, pp. 15205–15223, 2015.

[11] S. Zhang and S. T. Yau, "Absolute phase-assisted three-dimensional data registration for a dual-camera structured light system," *Appl. Opt.*, vol. 47, no. 17, pp. 3134–3142, 2008.

[12] L. Ye, G. H. Gu, W. J. He, H. D. Dai, J. Lin, and Q. Chen, "Adaptive target profile acquiring method for photon counting 3-D imaging lidar," *IEEE Photon. J.*, vol. 8, no. 6, Dec. 2016, Art. no. 6805510.

[13] G. Kim and Y. Park, "LIDAR pulse coding for high resolution range imaging at improved refresh rate," *Opt. Express*, vol. 24, no. 21, pp. 23810–23828, 2016.

[14] Y. M. Seong and H. W. Park, "A high-resolution image reconstuction method from low-resolution image sequence," in *Proc. 16th IEEE Int. Conf. Image Process.*, 2009, pp. 1181–1184.

[15] J. Zhu, W. Zhai, Y. Cao, and Z. J. Zha, "Co-occurrent structural edge detection for color-guided depth map super-resolution," in *Proc. Int. Conf. Multimedia Model.*, 2018, pp. 93–105.

[16] Y. X. Mao, B. J. Zhao, D. M. Yan, and H. W. Jia, "Super-resolution reconstruction from multiple defocused infrared images of stationary scene," *IEEE Photon. J.*, vol. 9, no. 5, Oct 2017, Art. no. 7000716.

[17] K. Cai, J. Shi, S. Xiong, and G. Wei, "Edge adaptive image resolution enhancement in video sensor network," *Opt. Commun.*, vol. 284, no. 19, pp. 4446–4451, 2011.

[18] F. Deboeverie, G. Allebosch, D. V. Haerenborgh, P. Veelaert, and W. Philips, "Edge-based foreground detection with higher order derivative local binary patterns for low-resolution video processing," in *Proc. Int. Conf. Comput. Vis. Theory Appl.*, 2014, pp. 339–346.

[19] T. C. Phung, S. I. Yong, J. C. Koo, and H. R. Choi, "Edge identification of a small object through a low-resolution tactile sensor array," *Int. J. Precis. Eng. Manuf.*, vol. 11, no. 2, pp. 247–254, 2010.

[20] M. H. Chang, H. S. Kim, J. H. Shin, and K. S. Park, "Facial identification in very low-resolution images simulating prosthetic vision," *J. Neural Eng.*, vol. 9, no. 4, 2012, Art. no. 046012.

[21] M. Zhang, K. Li, and Y. Liu, "Head pose estimation from low-resolution image with hough forest," in *Proc. Chin. Conf. Pattern Recognit.*, 2010, pp. 1–5.

[22] V. N. Murali and S. T. Birchfield, "Autonomous exploration using rapid perception of low-resolution image information," *Auton. Robots*, vol. 32, no. 2, pp. 115–128, 2012.

[23] T. C. Phung, S. I. Yong, J. C. Koo, and H. R. Choi, "An enhanced edge tracking method using a low resolution tactile sensor," *Int. J. Control Automat. Syst.*, vol. 8, no. 2, pp. 462–467, 2010.

[24] A. Lejeune, S. Piérard, M. V. Droogenbroeck, and J. Verly, "A new jump edge detection method for 3D cameras," in *Proc. Int. Conf. 3D Imag.*, 2011, pp. 1–7.

[25] H. Fu, B. Wu, Y. H. Shao, and H. Y. Zhang, "Perception oriented haze image definition restoration by basing on physical optics model," *IEEE Photon. J.*, vol. 10, no. 3, Jun. 2018, Art. no. 3900816.

[26] S. Schwarz, M. Sjostrom, and R. Olsson, "A weighted optimization approach to time-of-flight sensor fusion," *IEEE Trans Image Process*, vol. 23, no. 1, pp. 214–225, Jan. 2014.

[27] M. Camplani, T. Mantecon, and L. Salgado, "Depth-color fusion strategy for 3-D scene modeling with kinect," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1560–1571, Dec. 2013.

[28] D. Ferstl, M. Rüther, and H. Bischof, "Variational depth superresolution using example-based edge representations," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 513–521.

[29] J. Xie, R. S. Feris, and M. T. Sun, "Edge guided single depth image super resolution," in *Proc. Int. Conf. Image Process.*, 2014, pp. 3773–37777.

[30] T. Ojala and I. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognit.*, vol. 29, no. 1, pp. 51–59, 1996.

[31] M. Subrahmanyam, R. P. Maheshwari, and R. Balasubramanian, "Local maximum edge binary patterns: A new descriptor for image retrieval and object tracking," *Signal Process.*, vol. 92, no. 6, pp. 1467–1479, 2012.
[32] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
[33] P. Dollar and C. L. Zitnick, "Structured forests for fast edge detection," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 1841–1848.
[34] H. Hirschmuller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.
[35] D. Scharstein *et al.*, "High-resolution stereo datasets with subpixel-accurate ground truth," *Pattern Recognit.*, vol. 8753, pp. 31–42, 2014.