

Principal direction-based Hough transform for line detection

Yao Zhao · Haibin Pan · Changping Du ·
Yao Zheng

Received: 13 August 2014 / Accepted: 7 October 2014 / Published online: 17 March 2015
© The Optical Society of Japan 2015

Abstract A robust and fast line detection method based on Hough transform (HT) is proposed in this paper. Edge pixels are extracted based on the summation and ratio of principal curvatures. Probabilistic sampling on the edge pixels is applied to reduce the count of voting. Then a one-to-one voting strategy is applied by taking advantages of the information of principal direction. The principal direction is also conducive for the successive accurate line segment extraction. The experiments demonstrate that the proposed method shows better locating accuracy and computation efficiency compared with several significant variations of HT.

Keywords Line detection · Hough transform · Principal direction · Principal curvature · Line segment extraction

1 Introduction

The automatic detection of lines in images is a classic problem in image processing and computer vision. Automatic detection of lines has been employed in a wide range

of applications, including robot navigation [1], marine celestial navigation [2], camera calibration [3], industrial inspection [4], and character recognition [5]. Some detection methods have been proposed, e.g., chain code [6], ridgelet transform [7], LSD [8], EDLines [9], and orthogonal direction scanning [10].

Compared with other methods, the Hough transform (HT) [11, 12] is an efficient tool for detecting straight lines in images, even in the presence of noise and missing data, being a popular choice for the task. The standard HT (SHT) votes the edge pixels to a parameter space, and the determination of parameters of lines can be converted to the problem of peak identifying. The HT is conceptually simple and easy to implement. However, the limitations of the SHT are that it is time-consuming and not qualified for real-time applications, and also it is unable to locate line accurately in complex background. Several improved Hough transforms [13–17] have been proposed to expedite the method. Gradient-based HT [13] estimates the line direction via gradient direction. Within the estimation error of 10° , the total voting number can be reduced to 11 % of that of SHT. Probabilistic HT [14] uses a random sampling scheme to decrease the voting pixels, and experiment with competitive result has been obtained with as few as 2 % of edge pixels being sampled. Recently, the PPHT (progressive probabilistic HT) has been proposed. It takes a random sampling scheme, which is similar with the probabilistic HT, to vote. Afterward, it extracts lines along with the voting. To reduce the count of voting, it will remove the edge pixels which are on the extracted lines. Additionally, it takes an “unvoting” procedure to remove the “votes” which are cast by these removed edge pixels. However, the gradient-based HT methods and the probabilistic HT methods are still difficult to perform in real time without loss of accuracy. To decrease the voting times, kernel HT

Y. Zhao · H. Pan (✉) · C. Du · Y. Zheng
Institute of Aerospace Information Technology, Zhejiang University, Yuquan Campus, Hangzhou, 310027 Zhejiang, People's Republic of China
e-mail: phbzll@zju.edu.cn

Y. Zhao · H. Pan · Y. Zheng
Joint Laboratory of Flight Vehicle Ocean-based Measurement and Control, Zhejiang University, Yuquan Campus, Hangzhou, 310027 Zhejiang, People's Republic of China

H. Pan
Department of Electronic and Information Engineering,
The Hong Kong Polytechnic University, Hung Hom, Kowloon,
Hong Kong

(KHT) [17] uses a cluster-wise voting scheme. It clusters the potential line pixels at first and the voting of every cluster is limited in a small region in the parameter space. It can produce clear parameter space and is thrifty in the use of memory; additionally, the parameters need not be tuned. In contrast with the above methods using a one-to-many voting scheme, randomized HT (RHT) [16] uses a two-to-one voting scheme for line detection, that is, two edge pixels are selected randomly and the parameters of the line which are determined by these two point are used for voting. This method is efficient when there are a few long lines in the edge map. However, KHT and RHT will degrade when the background is complex.

In this paper, we propose a robust and fast line detector based on HT. There are several differences with other popular HT methods. Firstly, the edge pixels are extracted based on principal curvatures rather than Canny edge detector. Secondly, the minimum principal direction is used to determine the direction of the line [18, 19]. As the direction of the line can be estimated by the principal direction accurately, the θ parameter can be fixed; therefore, every edge pixel only needs to vote on one cell in parameter space. Similar to the probabilistic HT, a sampling process on the edge pixels is also applied for further reducing the voting count. The accuracy of line extraction will not degrade when less than a half of edge pixels are sampled for voting. In the proposed method, the voting count is usually less than 0.5 % of that of the SHT. Additionally, benefiting from the direction information of edge pixels, the successive line segment extraction can be more accurate than that only using the Canny edge map.

The proposed method comprises three steps. (1) Edge pixels based on principal curvatures are extracted. (2) Edge pixels are sampled for voting in parameter space. (3) Line parameters are obtained by peaks localization in parameter space, and line segments are extracted by the above edge map. The details are presented in the following sections.

2 Principal direction and Hough transform

2.1 Principal curvature and principal direction

Principal curvature and principal direction have been applied for detecting curvilinear structures [19], edges and ridges [18, 20, 21], and recognizing object [22]. This method shows good performance on the road detection and blood vessel detection. In the proposed method, principal curvature and principal direction are incorporated with HT for straight line detection.

Principal curvature and principal direction can be calculated approximately by the eigenvalues and

eigenvectors of the Hessian matrix [19]. In detail, the two eigenvalues of the Hessian matrix, k_1 and k_2 , are the principal curvatures, and the orientations determined by eigenvectors, θ_1 and θ_2 , are the principal directions. Furthermore, the principal curvature k_1 denotes the one which owns larger absolute value. The form of the Hessian matrix is given as

$$H = \begin{bmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{bmatrix}, \quad (1)$$

where f_{xx} , f_{xy} , f_{yx} , and f_{yy} denote the second-order partial derivatives. The surface of an image is discrete and the derivatives are calculated approximately by finite central differences. The summation of eigenvalues of Hessian matrix, i.e., $k_1 + k_2$, can be calculated by the trace of the matrix,

$$k_1 + k_2 = \text{tr}(H) = f_{xx} + f_{yy}. \quad (2)$$

It indicates that we can calculate the summation without the detailed k_1 and k_2 . The principal curvatures and principal directions are shown in Fig. 1.

Conventional methods use the gradient magnitude and gradient direction to extract edge pixels or estimate line direction. However, we find that using principal curvature and principal direction to extract edge pixels and estimate line direction will yield better performance, especially with the complex background. In the edge pixel extraction procedure, the ratio of principal curvatures can contribute a lot for declining clutter pixels. The principal curvatures indicate how the surface bends along two perpendicular directions. For a pixel of a line, one of its principal curvatures is a large value and the other is a small value. The ratio of two curvatures [i.e., the larger one of $\text{abs}(k_1/k_2)$ and $\text{abs}(k_2/k_1)$] is a large value, whereas for a pixel of a clutter its two principal curvatures are both large values and the ratio is a moderate value; therefore, the ratio contributes to remove clutter pixels. For comparison, the gradient only represents the steepest decrease direction (usually, approximately perpendicular to the line direction) and does not indicate the information along the line direction; therefore it is difficult to refine edge pixels. The principal direction also gets better estimating accuracy than gradient direction. Thus, the principal curvature and principal direction are more proper for line detection than the gradient magnitude and gradient direction. The detection results of the principal curvature method and the gradient method are shown in Fig. 2. All the parameters except the threshold for peak locating are set to the same default values. The thresholds are adjusted to keep the count of detected lines same or closed. The gradient method makes several mistaken detections with the complex background, especially on the noise image.

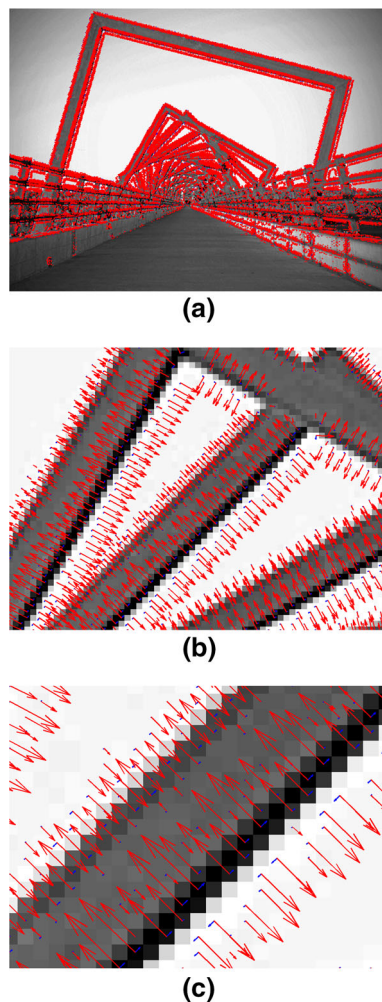


Fig. 1 The principal curvatures and principal directions of edge pixels. **a** Holistic image; **b, c** part of **a**. Maximum principal curvatures and minimum principal curvatures (together with corresponding principal directions) are drawn in red and blue, respectively

2.2 Edge pixel extraction

Different from the conventional HT methods, the proposed method takes the principal curvature rather than the Canny edge detector for edge extraction. The main reason is that the Canny operator is time-consuming (about 4 ~ 6 ms for a 512 × 386 image; for comparison the proposed method takes only 4 ~ 9 ms for the total line detection). Other reasons are that the Canny operator is not compatible with the principal curvature calculation and not instrumental for accurate line segment extraction.

To suppress noise, a Gaussian filter is applied initially. Gaussian filters can sustain the main structure of the surface and suppress point noises effectively. Subsequently, stepwise calculations, which can save computation time, are employed to extract edge pixels. Firstly, the summation of the two curvatures is calculated; if the absolute value of the summation of principal curvatures, i.e., $\text{abs}(k_1 + k_2)$,

is larger than a threshold S_{\min} , these pixels are supposed to be potential edge pixels or clutter pixels. Then the principal curvatures of these pixels are calculated; if the ratio between two principal curvatures, i.e., $\text{abs}(k_1/k_2)$, is larger than a threshold R_{\min} , the pixel is considered an edge pixel and then its principal directions are calculated. Edge pixel extraction together with the successive procedures is shown in Fig. 3 for better understanding.

2.3 Vote casting

The edge pixels are used for voting on the parameter space. In the conventional HT method, for each edge pixel, θ varies uniformly from $-\pi/2$ to $\pi/2$ and the corresponding ρ is calculated by

$$x \cos \theta + y \sin \theta = \rho. \quad (3)$$

In the proposed method, every pixel only needs to vote for one cell such that θ_1 is determinate and the ρ parameter can be calculated by

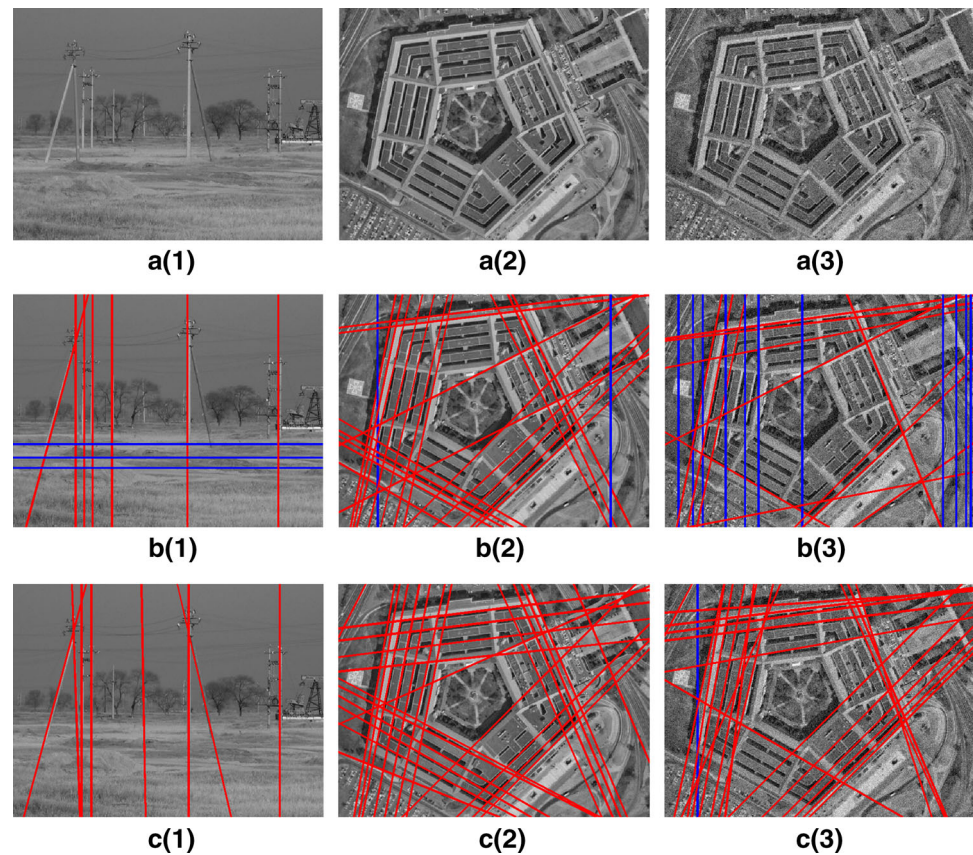
$$x \cos \theta_1 + y \sin \theta_1 = \rho, \quad (4)$$

where x and y are the coordinates of the pixel in the image, ρ is the distance from the origin of the image's coordinate system to the line and θ is the angle between the image's x -axis and the normal to the line. In the parameter space, θ is restricted to $[-\pi/2, \pi/2)$ and ρ is restricted to $[-R, R]$, where $R = \sqrt{w^2 + h^2}/2$, and w and h are the width and height of the image, respectively. Each cell in the parameter space represents a line.

It is found that detection accuracy is not impaired if a fraction of image pixels are used for voting. However, the whole processing time can be reduced due to the reduction of voting count. Results show that half of image pixels used for voting will nearly cause no degradation; a better performance can even be obtained after some time. When processing a large image, like 1600 × 1200, sampling about only a quarter of the edge pixels will produce a competitive result. In the proposed method, the sampling rate is inversely proportional to the width (or height) of the image. The edge map and voting map (i.e., parameter space) are shown in Fig. 4. The edge map is not like the Canny edge map which has continuous lines. It comprises discrete points.

For the conventional HT method, every voting will increase one “vote” on the corresponding cell in the parameter space, whereas the gradient-based HT [13] will increase the count of gradient magnification “votes” in each voting on the corresponding cell. This is to say, a pixel with large gradient will contribute more on the corresponding cell. In the proposed method, the increment value of each voting is assigned to one or the maximum principal curvature for testing. The detection result on

Fig. 2 The detection results of the *gradient method* (row **b**) and the *principal curvature method* (row **c**). Image **a(3)** is yielded by adding noises to image **a(2)**. The standard variance of noises is 10. The mistaken detected lines are drawn with *blue color*



Pentagon (the Pentagon image is an ideal example for evaluating HT methods, for the reason that it contains some straight lines with gaps and much clutters) is shown in Fig. 5. The two strategies generated competitive results. However, using the strategy with the increment value of maximum principal curvature it is difficult to provide a proper default threshold for peak localization. In addition, it needs more computation memory as its accumulator is a 32-bit float type rather than an 8-bit integer type. Therefore, the strategy with the increment value of one is employed in the proposed method.

2.4 Line segments extraction

Peaks are located by a global threshold in parameter space. Also, only one peak is selected when several peaks are near each other. The global threshold is suggested to be 7 for the 512×386 images and most times it will yield an acceptable result. Line parameters can be determined by the peaks, and then the line pixels can be extracted by the line parameters. In some application scenarios, the line segments should be extracted after the lines are located. The proposed method uses the curvature edge map as well as the directions of the edge pixels to obtain accurate line segments. The conventional methods

would likely connect the clutter pixels to a line because the gaps between clutter pixels may be less than the given threshold of maximum gap. The proposed method is less likely to connect clutter pixels by mistakes. One reason is that it will decline the clutter pixels in the edge pixel extraction procedure. The other is that it will evaluate the principal directions of the pixels before connecting, and only the pixels possessing similar directions with line direction are connected. The principal directions of clutter pixels rarely are similar with the direction of the line, thus it can also reduce mistaken connecting. The best results accomplished by curvature edge map without direction information and with direction information are shown in Fig. 6.

For the fact that the edge pixels in the proposed method usually are isolated points and not linked with each other, the pixels of a line segment are searched around the line. In detail, a pixel is supposed to be a line segment pixel if it is near a line and its maximum principal direction is similar with the line direction. Additionally, consistent with the conventional methods, the gap which is less than a threshold is filled with line segment pixels. The line segment which is shorter than a threshold is discarded. Usually, the line extraction procedure will make the line segments locate more precisely than the original lines.

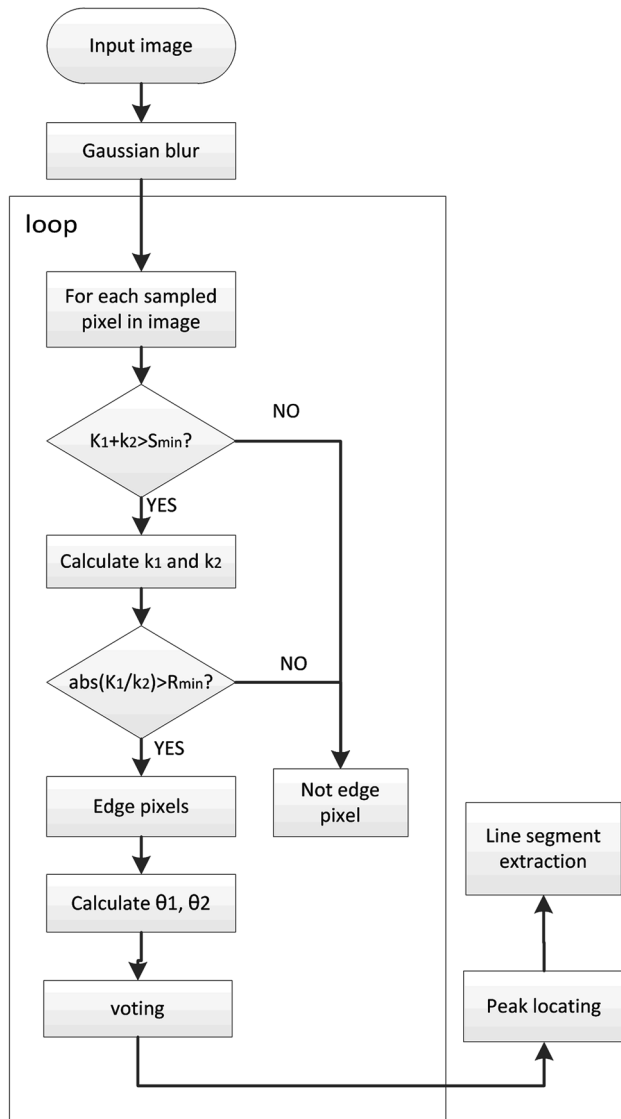


Fig. 3 The block chart of the proposed method

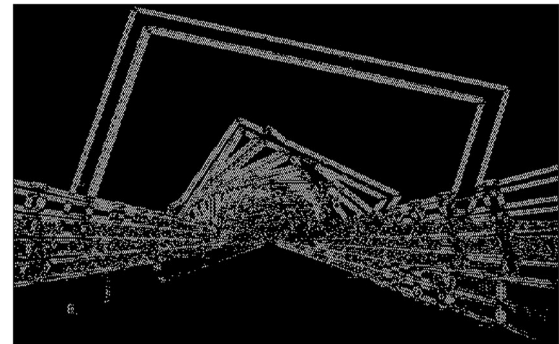
3 Experiments and evaluations

In this section, experiments on several images are presented for illustration and comparison. The images are grayscale with a size of 384×512 . The standard deviation of the Gaussian filter is 1.2. The minimum summation of principal curvatures (S_{\min}) is 6. The minimum ratio (R_{\min}) is 5. The resolution of θ and ρ is 1° and 1 pixel, respectively. The suppression neighborhood for locating peaks in the parameter space is $1/50$ of row count and $1/25$ of column count. The detection results of three images are shown in Fig. 7. The proposed method can both extract lines in clear background and complex background.

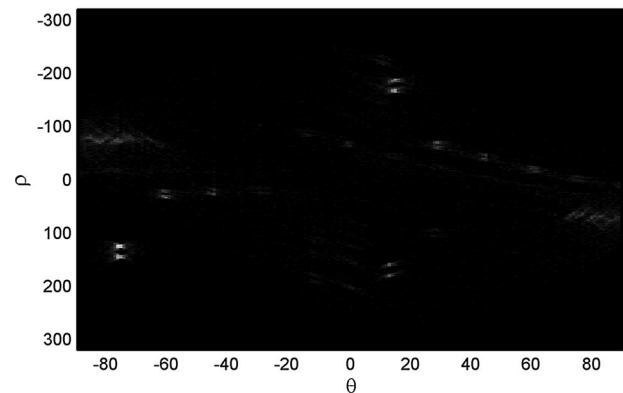
The comparisons between the proposed method and several important variations of HT, e.g., PPHT (progressive probabilistic HT) [15] and KHT [17] (kernel HT), are



(a)



(b)



(c)

Fig. 4 The edge map [image (b)] and voting map [image (c)] of the image (a)

presented in Fig. 8. Edge detection is implemented by the Canny operator in OpenCV, and the two thresholds are 100 and 300 for all images. SHT and PPHT are implemented by OpenCV, and the resolution of θ and ρ are 1° and 1 pixel, respectively. One thing to note is that the outputs of PPHT are line segments, which are plotted as lines to make a comparison with other methods. The parameters adopted in KHT are the ones suggested by the method proposers, that is, the minimum size of cluster is 10, the minimum deviation of cluster is 2, the resolution of θ and ρ are 0.5° and 0.5 pixels, the sigma value dedicated to the voting region is 2 and the minimum

Fig. 5 The most important 30 lines with the increment value of maximum principal curvature (*left*) and one (*right*)

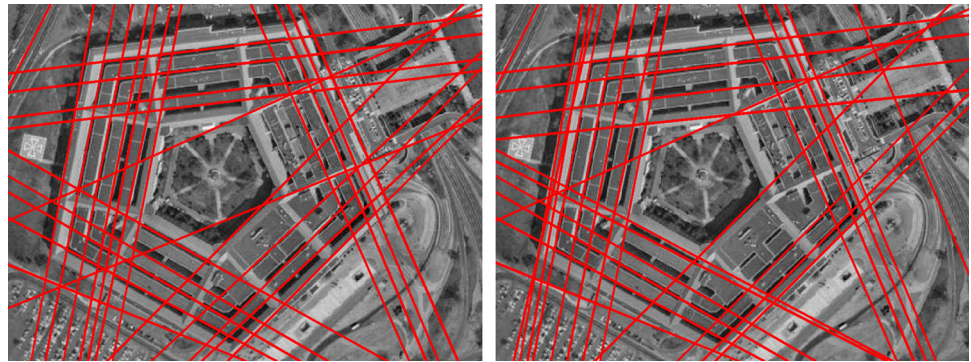


Fig. 6 Results of line segment extraction with (*left*) and without (*right*) pixel direction information

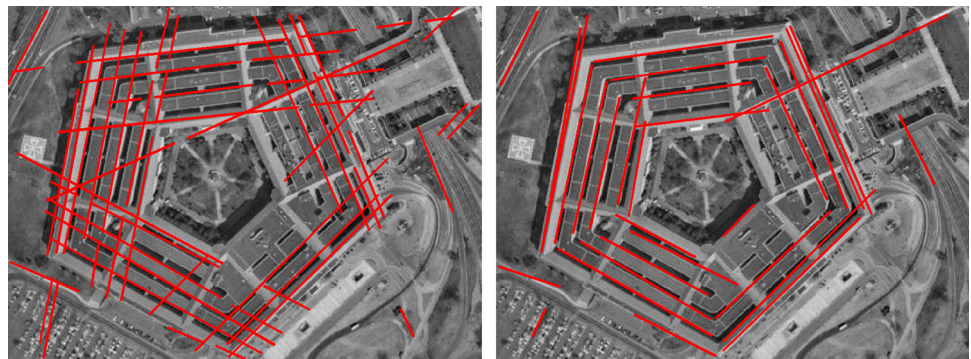


Fig. 7 Detection results for three images

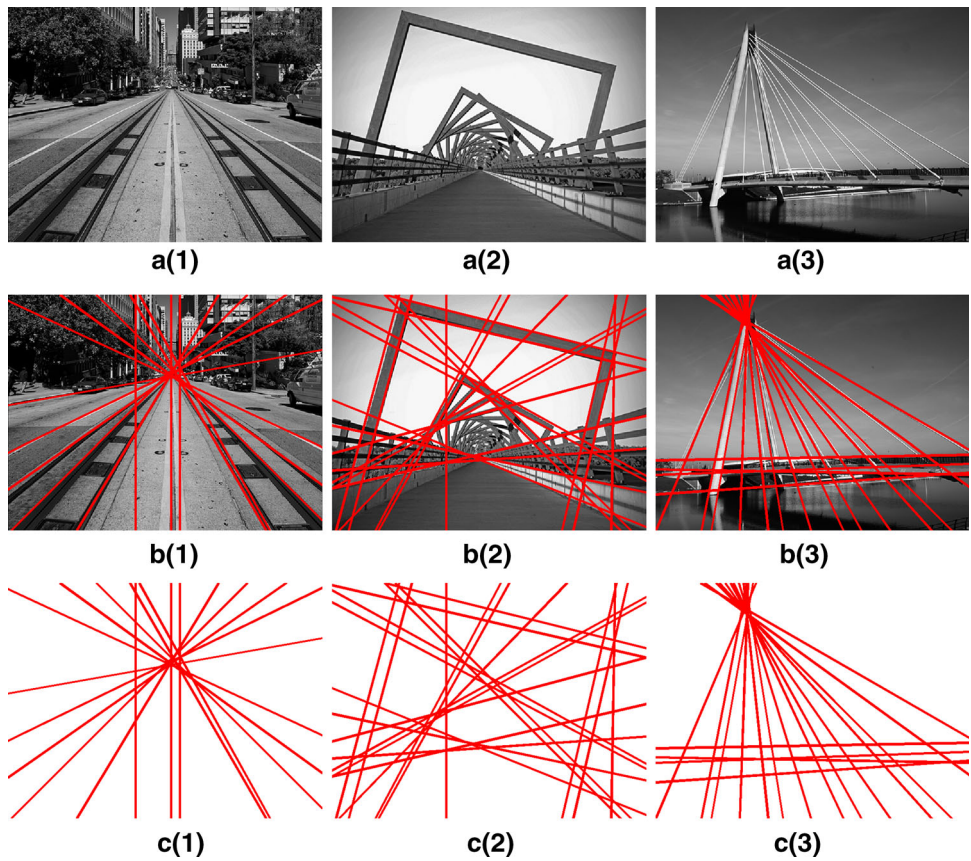
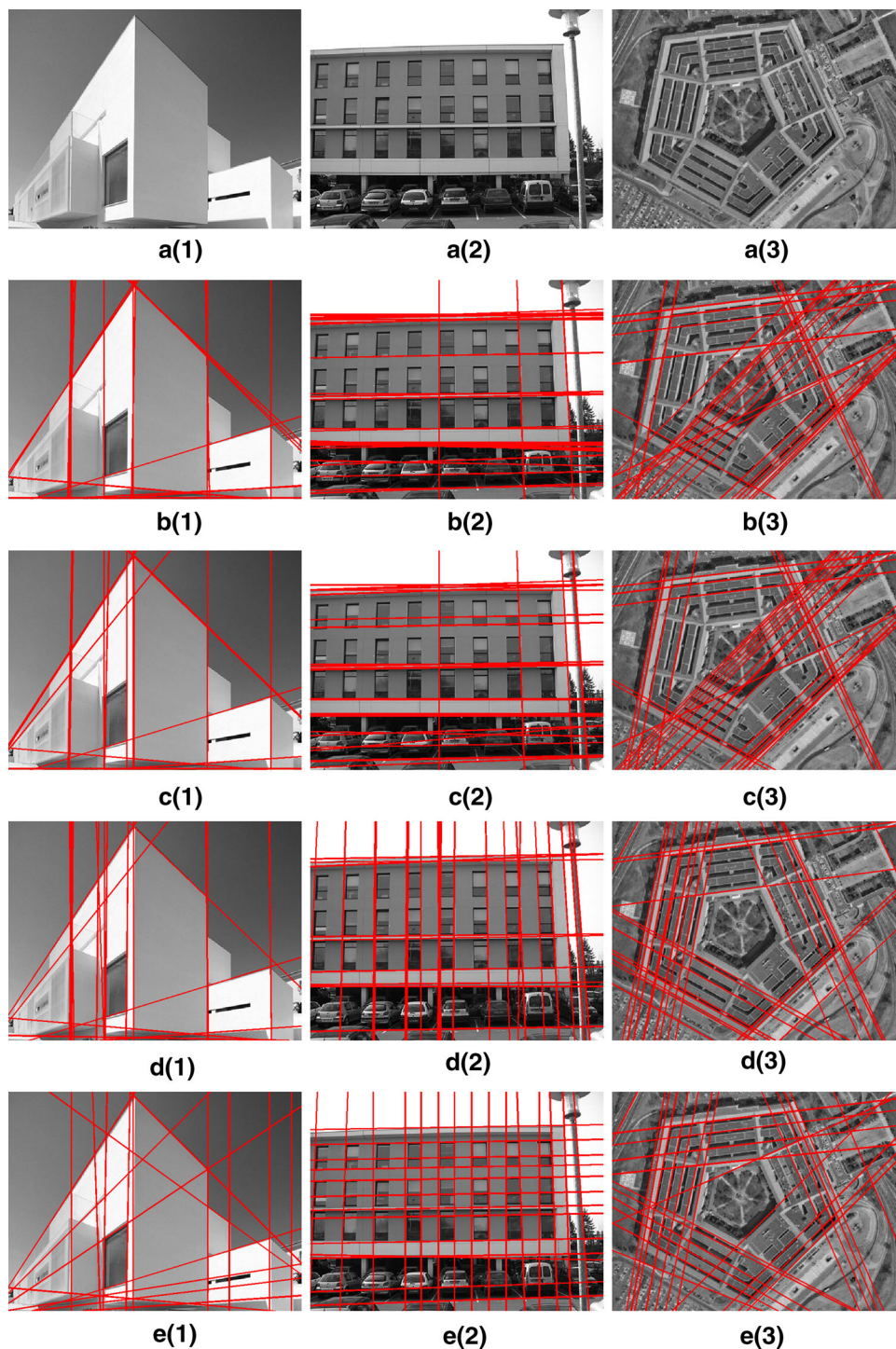


Fig. 8 Detection results for several methods. Rows (b), (c), (d), and (e) show the results of SHT, PPHT, KHT and the proposed method, respectively. The figure exhibits the 20, 30, and 30 (or the most approximate count) most important lines detected by each algorithm



height of kernel is 0.002. The computation environment includes C++ with the OpenCV library on Windows 32-bit system and Pentium Dual-core with 3.2 GHz CPU and 3 GB RAM.

The proposed method shows better line detection performance when compared with the other methods, especially on the complex background, like the image of Pentagon. The voting count of several methods is shown in

Table 1. The voting count of PPHT, KHT and the proposed method varies with the parameters. For comparison, the parameters are tuned to generate the same or approximate count of lines. The voting count of PPHT includes the “unvoting” count. One thing to note is that KHT does not employ the pixel-wise voting scheme, but a cluster-wise voting scheme. A cluster may contain hundreds of pixels, and the voting scheme makes KHT own a seemingly small

Table 1 Voting count of SHT, PPHT, KHT and the proposed algorithm on the images shown in Fig. 8 and an additional large-size image

	SHT	PPHT	KHT	Proposed
White building (512 × 384)	1,039,320	1,318,320	1,516	6,912
Gray building (512 × 384)	3,819,960	4,661,280	3,213	21,580
Pentagon (512 × 384)	7,649,640	8,352,460	28,963	25,506
Gray building (1600 × 1200)	16,616,700	21,565,080	931	47,781

Table 2 Comparison of the performance between SHT, PPHT, KHT and the proposed algorithm on the images shown in Fig. 8 and an additional large-size image

	SHT (ms)	PPHT (ms)	KHT (ms)	Proposed (ms)
White building (512 × 384)	14.50	14.64	7.94	3.77
Gray building (512 × 384)	34.00	39.31	9.32	9.39
Pentagon (512 × 384)	58.45	69.01	16.64	8.78
Gray building (1600 × 1200)	166.48	234.56	50.40	30.00

voting count, Additionally, the time of voting in KHT is only a small fraction of the whole processing time [17]. It is shown that the voting count of the proposed method is much less than that of the SHT and PPHT.

The proposed method also shows significantly better computational efficiency than SHT and PPHT and is even faster than the real-time KHT method for most images, which is shown in Table 2. It can process an image of size 512 × 386 as fast as 275 fps and a large image of size 1600 × 1200 in about 36 fps on an out-of-date processor (Pentium 32-bit processor).

4 Conclusion

In this paper, a fast and robust HT for line detection is proposed. Edge pixels are extracted based on the principal curvatures and one-to-one voting is cast based on the principal directions. For acceleration, only a fraction of the edge pixels are sampled for voting. The voting count in the proposed method is less than 0.5 % of that in the conventional HT. The capability and effectiveness of the proposed method are demonstrated with several testing images. The comparisons between the proposed method and other important HT variants prove that our method performs better than the other methods in terms of computation efficiency and location accuracy, especially on complex images.

Acknowledgments This work is supported by the National Natural Science Foundation of China under Grant No. 61008048, the Open Fund of Joint Laboratory of Flight Vehicle Ocean-based Measurement and Control under Grant No. FOM2013OF001 and the Fundamental Research Funds for the Central Universities of China under Grand No. 2014FZA4029.

References

1. G. Yammine, E. Wige, F. Simmet, D. Niederkorn, A. Kaup, *IEEE International Conference on Vehicular Electronics and Safety*, 24–27 July 2012
2. C. Li, Y. Zheng, Y. Yuan, Y. Yang, *China Satellite Navigation Conference (CSNC)*, ed. by J. Sun, J. Liu, Y. Yang, S. Fan. Lecture notes in electrical engineering, vol. 160 (Springer, Heidelberg, 2012)
3. D. Farin, J. Han, P. H. de With, *IEEE International Conference on Multimedia & Expo (ICME)*, 6–8 July 2005
4. F.C. Tien, C.H. Yeh, K.H. Hsieh, *Int. J. Prod. Res.* **42**, 2477 (2004)
5. C.L. Liu, S. Jaeger, M. Nakagawa, *Pattern Anal. Mach. Intell.* **26**, 198 (2004)
6. G. Q. Lu, X. Hong-guo, Y. B. Li, *IEEE International Conference on Vehicular Electronics and Safety*, 14–16 Oct 2005
7. B. Hou, F. Liu, L. Jiao, *Sci. China Ser. E Technol. Sci.* **46**, 141 (2003)
8. R.G. Von Gioi, J. Jakubowicz, J.M. Morel, G. Randall, *Pattern Anal. Mach. Intell.* **32**, 722 (2010)
9. C. Akinlar, C. Topal, *Pattern Recogn. Lett.* **32**, 1633 (2011)
10. K. Yang, S. Sam Ge, H. He, *Comput. Vis. Image Underst.* **115**, 1207 (2011)
11. P. V. Hough, U.S. Patent No. 3,069,654, 1962
12. R.O. Duda, P.E. Hart, *Commun. ACM* **15**, 11 (1972)
13. F. O’Gorman, M.B. Clowes, *IEEE Trans. Comput.* **100**, 449 (1976)
14. N. Kiryati, Y. Eldar, A.M. Bruckstein, *Pattern Recogn.* **24**, 303 (1991)
15. J. Matas, C. Galambos, J. Kittler, *Comput. Vis. Image Underst.* **78**, 119 (2000)
16. L. Xu, E. Oja, P. Kultanen, *Pattern Recogn. Lett.* **11**, 331 (1990)
17. L.A. Fernandes, M.M. Oliveira, *Pattern Recogn.* **41**, 299 (2008)
18. R.M. Haralick, *Comput. Vis. Gr. Image Process.* **22**, 28 (1983)
19. C. Steger, *Pattern Anal. Mach. Intell.* **20**, 113 (1998)
20. T. Lindeberg, *Int. J. Comput. Vis.* **30**, 117 (1998)
21. M. E. Martínez-Pérez, A. D. Hughes, A. V. Stanton, S. A. Thom, A. A. Bharath, K. H. Parker, *Medical Image Computing and Computer-Assisted Intervention*. (Springer, Heidelberg, 1999), pp. 90–97
22. H. Deng, W. Zhang, E. Mortensen, T. Dietterich, L. Shapiro, *IEEE Conference on Computer Vision and Pattern Recognition*, 17–22 June 2007