

Efficient vanishing point detection method in unstructured road environments based on dark channel prior

Weili Ding¹ ✉, Yong Li¹, Honghai Liu²

¹Institute of Electrical Engineering, Yanshan University, Qinhuangdao, Hebei 066004, People's Republic of China

²School of Computing, University of Portsmouth, Portsmouth PO1 3HE, UK

✉ E-mail: weiy51@ysu.edu.cn

Abstract: Vanishing point detection is a key technique in the fields such as road detection, camera calibration and visual navigation. This study presents a new vanishing point detection method, which delivers efficiency by using a dark channel prior-based segmentation method and an adaptive straight lines search mechanism in the road region. First, the dark channel prior information is used to segment the image into a series of regions. Then the straight lines are extracted from the region contours, and the straight lines in the road region are estimated by a vertical envelope and a perspective quadrilateral constraint. The vertical envelope roughly divides the whole image into sky region, vertical region and road region. The perspective quadrilateral constraint, as the authors defined herein, eliminates the vertical lines interference inside the road region to extract the approximate straight lines in the road region. Finally, the vanishing point is estimated by the meanshift clustering method, which are computed based on the proposed grouping strategies and the intersection principles. Experiments have been conducted with a large number of road images under different environmental conditions, and the results demonstrate that the authors' proposed algorithm can estimate vanishing point accurately and efficiently in unstructured road scenes.

1 Introduction

Detecting vanishing point in a road image is important for depth estimation [1], camera calibration [2], three-dimensional (3D) reconstruction [3], autonomous driving [4], robot navigation [5] and so on.

Over the past few decades, there have been a lot of algorithms for vanishing point detection in either structured or unstructured roads. The current methods can be generalised into three groups: texture based, edge based and prior based [6, 7].

Texture-based methods search for local or global oriented textures and vote the most likely road vanishing point by the textures [6–12]. In [10], Rasmussen estimated vanishing points using texture orientation and hard voting method. In [6], Kong *et al.* proposed the vanishing point detection method based on generalising Laplacian of Gaussian filters, and they use soft voting method to solve the disadvantage of hard voting. These methods are applied to estimate the vanishing point for the unstructured roads. However, those methods are sensitive to noise and are not good at road scenes with few texture features. They also have high computational complexity and cannot satisfy the real-time requirement.

Edge-based methods [13–16] first detect lines or parallel lines [14] in the road image and then intersections of any pair of lines will be computed, and the vanishing point will be estimated by the clustering algorithms. Here, the lines detection method is mainly based on Hough transformation [13] or line segment detector proposed by von Gioi [17], and the clustering algorithms include K-means [13], C-means [14], Expectation Maximization Algorithm (EM) [16], and so on. Most of these methods show the effectiveness and real feature of the vanishing point detecting in structured roads. However, the vanishing point detection results are affected by the accuracy of the straight lines detection in the road scene heavily. Thus, they may be ineffective when there are not enough edges or straight line segments in the unstructured road image.

Prior-based methods [18, 19] use machine learning algorithms to estimate the vanishing point. That is, these methods match the best candidate images in an image database and identify the vanishing point by refining the location of vanishing point based

on a probabilistic model of vanishing point. For example, some methods studied a global perspective structure matching scheme using image retrieval technique to extract the best candidate images in database [18], some methods extract integrating contextual 3D information with low-level features [19]. These methods may be more accurate, but they need a large scale training database according to the types of road scenes, and it is a huge task to mark the vanishing point for the training stage.

Analysing both the advantages and the disadvantages of the existing methods, we propose a new vanishing point detection method using the dark channel prior, the lines information and colour texture properties of the unstructured roads, which both the detection accuracy and computation time are significantly improved. Compared with the existing algorithm, the contributions of our proposed method are as follows:

First, our algorithm is starting from the image segmentation regions, which are generated based on the dark channel prior [20] and K-means algorithm. Thus, the weak edges and the region characteristics can be obtained in the first step. Then, all approximate straight lines are extracted from the segmented region contours using a fast straight line detection method based on cumulative errors. Furthermore, unlike direct line detection method and textures detection method based on Gabor or gLog, our method can obtain enough useful information and has a relatively small complexity.

Second, in order to extract the useful line segments in the road region, the vertical envelope and the perspective quadrilateral are defined to filter. To estimate the ultimate road vanishing point, new effective computing intersection strategies are defined. As a result, our proposed method achieves desirable detection accuracy as well as extremely fast computation time compared with the texture map-based method proposed by Kong *et al.* [6, 9], which is considered as the state-of-the-art method.

An outline of this paper is as follows: In Section 2, the description of our methods is given in detail. In Section 3, the experimental results are discussed, and the performances of our method are evaluated. Finally, the conclusions and the further works are shown in Section 4.

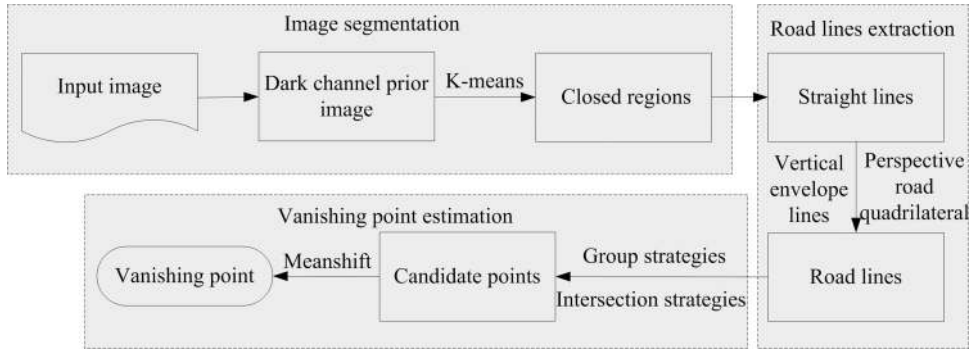


Fig. 1 Flowchart of the proposed algorithm



Fig. 2 Process of the image segmentation
(a) Original image, (b) Dark channel prior image, (c) Dark channel image smoothed, (d) Segmentation results

2 Proposed algorithm

The proposed vanishing point detection algorithm is mainly divided into three steps: image segmentation, road lines extraction and vanishing points detection, as shown in Fig. 1.

2.1 Image segmentation based on dark channel prior

Our vanishing point detection method relies on dark channel prior. According to the study of He [20], low values in the dark colour channel are caused by three main factors: (a) shadows of cars, buildings and urban glass windows or natural landscape projection; (b) brightly coloured objects or surfaces, such as green grass, trees, red or yellow flowers, leaves and so on; (c) darker objects or surfaces, such as dark-coloured trunks and stones. After observing thousands of road images, we find that a lot of road regions including the road surface and signs correspond to high values in the dark channel, and the green trees, the colour vehicle, pedestrian crossing or the shadows correspond to low values. As shown in Fig. 2, we see that the rutting in road region is more easily identified in the dark channel image, which provides a more meaningful segmentation between the road and the background. So, we provide a new algorithm to segment the road image as follows:

Given an image I , as shown in Fig. 2a, the dark channel image is computed based on the following formula [20]:

$$I^{\text{dark}}(x) = \min_{y \in \Omega(x)} \left(\min_{c \in \{r, g, b\}} I^c(y) \right), \quad (1)$$

where $\Omega(x)$ is a local region centred at pixel x , whose size is determined by the filter radius r

$$\Omega = 2 \times r + 1, \quad (2)$$

and I^c is a colour channel of I . I^{dark} satisfies the condition of $I^{\text{dark}} \rightarrow 0$. Then, we define the dark channel prior image as I^{dark} (Fig. 2b).

To avoid the impact of the rectangular edges in Fig. 2b, the Gaussian smoothing filter is used to obtain the smooth image I_G (Fig. 2c) pixel by pixel in the next step

$$G(u, v) = \frac{1}{2\pi\sigma^2} e^{-\frac{(r_g^2)/(2\sigma^2)}{2}}, \quad (3)$$

where $r_g^2 = u^2 + v^2$, σ is the standard deviation of the normal distribution. Here, r_g is at the range of 8–20, and σ is at the range of 0.8–2.

After smoothing, the road image is segmented into more meaningful regions in Fig. 2c. based on K-means clustering algorithm [21]. Thus, a set of segmented regions will be obtained, and the segmented image I_s , for which some samples are shown in

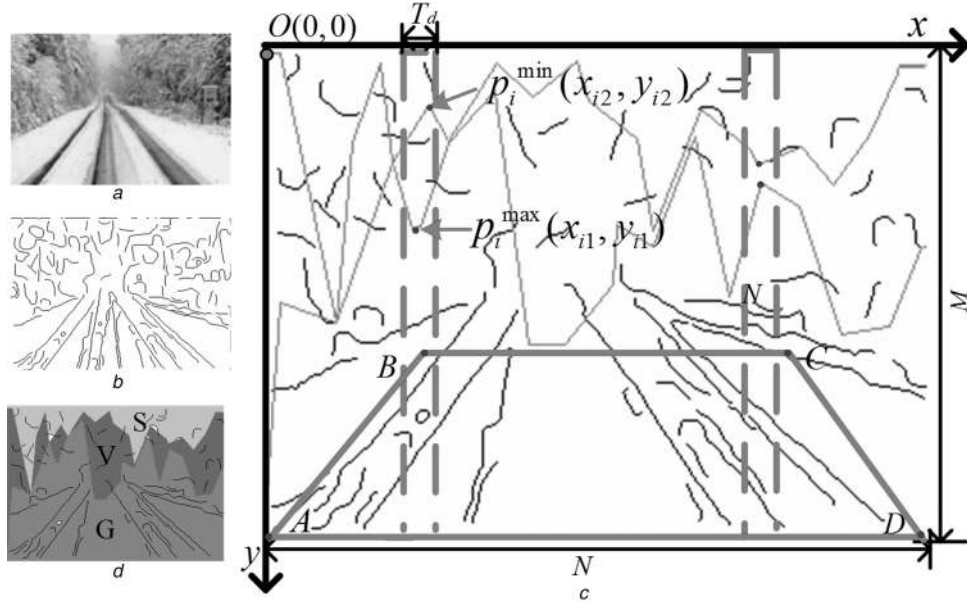


Fig. 3 Road lines extraction process

(a) Original image, (b) Approximate lines of the segmentation image, (c) Envelope image, (d) Region segmentation and road lines image

Fig. 2d will be generated finally. Here, $I_s = \sum_{i=1}^n R_i$, where n is the number of the segmented regions, and R_i means any segmented region. In this step, it is important to set a suitable parameter of the K-means clustering algorithm, and we will discuss it in Section 2.4.

From the segmentation results in Fig. 2, we see that the rutted structures of the road region are well segmented based on the dark channel and clustering method. That is to say the weak edges in the road region can be detected successfully by extracting the boundaries of the rutted regions. Since the line segments of the rutted structures contribute to the vanishing point detection it is important to detect line segment of the rutted regions in the unstructured road image.

2.2 Lines extraction from the road region

Generally, the vanishing point is estimated by the intersection of two tangent lines which are the nearest road edges or main direction lines in the road region. So, extracting lines from the road region is very important and we detect them by the following steps.

2.2.1 Straight lines detection: To obtain the accurate vanishing point of the road, first we extract line segments from all contours of the segmented regions as follows:

Given an closed contour $C_i = \{p_{ij}\}$, $j=1, 2, \dots, m_i$, which corresponds to a segmented region R_i , we segment it as a set of line segments using an incremental recognition process. That is, we start at some point of the contour and perform a line segment recognition with an allowable error. The last point of the previous recognition is taken to be the first point of computation of the next recognition until all points of the contour have been examined. Here, the line segment recognition algorithm follows a simple principle: we take the first m_p points of the given contour as c_i , $c_i = \{p_{ij}\} (j=1, 2, \dots, m_p)$, and fit them into a straight line $y = a_i x + b_i$ using the least squares fitting algorithm. Here, a_i is the slope of that detected straight line. Then, we add the next points to the recognised line segment until there is a point whose distance d_{err} to the fitted straight line satisfies $d_{err} > T_{err}$. Here, T_{err} is a threshold value used to judge whether a point is on the fitted straight line [22]. In this way, each contour is further decomposed into several line segments and the orientation of each line segment can be written as [23, 24] follows:

$$\theta_i = \tan^{-1} a_i. \quad (4)$$

On the basis of the above algorithm, the approximate straight lines in the road region can be obtained. Here we define the number of the extracted line segments is n , all detected line segments can be described as follows:

$$P_{ll} = \{l_i | \theta_i\}, \quad i = 1, 2, \dots, n. \quad (5)$$

Taking Fig. 3a as an example, the straight lines in Fig. 3b are extracted by using the above method.

2.2.2 Region classification: According to our previous studies [14], the envelope of the vertical lines can be used to segment the road image into three parts: road region, sky region and vertical region as shown in Fig. 3d. According to the extracted line segments, if the main direction angle of a line segment l_i satisfies

$$\theta_i \in [90 - \delta, 90 + \delta], \quad (0 < i \leq n). \quad (6)$$

We define it as a vertical line. Here, the value of δ is the threshold to detect the vertical lines.

To filter out the interference lines from the vertical lines in the road region, we define a *perspective quadrilateral*, which has four vertices A, B, C and D (see Fig. 3c for example). Here, if the image size is $M \times N$, then the vertice coordinates in Fig. 3c are set as $A(0, M)$, $B(0.25 \times N, 0.55 \times M)$, $C(0.75 \times N, 0.55 \times M)$ and $D(N, M)$ in our algorithm. Then the vertical lines, which are not in the defined perspective quadrilateral, will be extracted and noted as $V_{ll} = \{l_i\}$, ($i = 1, 2, \dots, n_1, n_1 < n$). Next, the envelope of vertical lines V_{ll} is detected as follows [14].

First, a small rectangular window whose size is $M \times T_d$ (see the dotted lines in Fig. 3c) is set, and the image is divided into several sub-images. Then, the highest point $p_i^{\max}(x_{i1}, y_{i1})$, $i = 1, \dots, N/T_d$ and the lowest point $p_i^{\min}(x_{i2}, y_{i2})$ of the vertical lines in each sub-image will be computed. Finally, all highest points and lowest points are connected in a sequential order (see the two grey polygonal lines in Fig. 3c), and these points are defined as the vertical envelope whose vertices are

$$Q = \{(1, y_{11}), p_1^{\max}, \dots, p_{N/T_d}^{\max}(M, y_{T_d}), (M, y_{T_d}), p_{N/T_d}^{\min}, (1, y_{12})\} \quad (7)$$

2.2.3 Road lines extraction: On the basis of the above definition of the vertical envelope, as shown in Fig. 3d, the input image will

be divided into three regions: G (road region), V (vertical region) and S (sky region). Due to the road vanishing point is primarily determined from the line segments in the road region, we define the line segments in the road region as *road lines*, and extract them from the road region using the following criterion.

CRITERION: If the main direction θ_i and the length d_i of any line segment l_i satisfy:

$$\begin{cases} l_i \cap G \neq \emptyset \\ \theta_i \in (\delta', 90 - \delta) \cup (90 + \delta, 180 - \delta') \\ d_i > \max\{M, N\} \times \eta \end{cases}, \quad (8)$$

where δ' is the horizontal lines detection threshold. We define the line l_i as a road line in the road region. Moreover, l_i in the road region will be divided into one of the following groups, that is:

$$l_i \in \begin{cases} G_1 & \text{if } \theta_i \in [\delta', 90 - \delta) \\ G_2 & \text{if } \theta_i \in (90 + \delta, 180 - \delta'] \end{cases}. \quad (9)$$

In this paper, we select the image whose size is normalised to 180×240 pixels. Here, $\eta \in [0.03, 0.05]$ is the proper value range.

2.3 Vanishing point detection

To improve the efficiency and accuracy of the vanishing point detection, it is very important to find the vanishing point candidates using a few straight lines. Moreover, we define the following principle to compute the vanishing point candidates of the road lines in the two groups.

PRINCIPLE 1: Consider any line $l_i \in G_1$, whose main direction angle and mid-point is θ_i and (x_i, y_i) , if there is a line $l_j \in G_2$, whose main direction angle and mid-point is θ_j and (x_j, y_j) , and they satisfy

$$\begin{cases} y_i, y_j \in [0.25 \times N, 0.75 \times N] \\ 180 < |\theta_i + \theta_j| < 240 \end{cases}, \quad (10)$$

then the intersection points of l_i and l_j will be defined as a vanishing point candidate.

PRINCIPLE 2: Select only one group from G_1 and G_2 . Here, the selected group contains a larger number of road lines. Then we consider any two lines l_i, l_j in it, if the main direction angle of l_i and l_j are θ_i and θ_j , and they meet

$$|\theta_i - \theta_j| > 5, \quad (11)$$

then the intersection points of l_i and l_j will be defined as a vanishing point candidate.

After obtaining vanishing point candidates from the two groups of the road lines, we estimate the position of the vanishing point using the meanshift clustering algorithm [25]. In order to reduce the detection error of the vanishing point, we chose the meanshift clustering algorithm. Moreover, it is not needed to set the number of the clusters and the clustering radius is uncertain. Given n_p vanishing point candidates in the 2D space, the bandwidth of meanshift algorithm in our study is set as h , and the value $m_h(V)$ is computed as follows:

$$m_h(V) = \frac{\sum_{i=1}^Z G(V_i - V/h)w(V_i)V_i}{\sum_{i=1}^Z G(V_i - V/h)w(V_i)} \quad (12)$$

Then, the V is equalled to $m_h(V)$ as the V_{i+1} until

$$\|m_h(V) - V\| < \varepsilon \quad (13)$$

Here, we define $\varepsilon = 1 \times 10^{-3} \times \text{bandwidth}$, $G(X)$ is the kernel and $w(V_i)$ is the weight value of V_i .

Finally, the vanishing point candidates are classified as N clusters. Then, we select one cluster centre $V_{cp}(x_{ij}, y_{ij})$, which includes the most number of sampling points, as the ultimate road vanishing point.

2.4 Selection of parameter values

First, we discuss the two parameters of the Gaussian smoothing filter. We selected 25 different types of images, and tested the segmentation results of 25 images with different values of r_g and σ . As shown in Fig. 4a, the numbers of segmented regions are affected by the values of r_g and σ slightly. In our study, we find $r_g = 18$ and $\sigma = 1.8$ are the optimal values.

In our algorithm, the vanishing point location is mainly effected by the different values of k . That is, the accuracy of the algorithm depends on the number of the segmented regions. As shown in Fig. 4b, the white points are the vanishing points location estimated by our method with different value of k , and the black points are the ground truth locations of the vanishing points. The first row to the fourth row shows different vanishing points locations when the value of k is set as 5, 8, 12 and 15, respectively. We see that the vanishing point location of the second row and the third row are closer to the ground truth than the other two rows. The value of $k = 8$ and 12 are more suitable than $k = 5$ and 15, respectively. So, the recommended value of k is from 8–12.

The proper clustering bandwidth is a crucial parameter. Here, we tested several images with different bandwidth length, and we recorded their time consumption. As shown in Fig. 4c, the black points are the ground truth locations of the vanishing points. The white points are locations estimated by our method with different bandwidths. Moreover, the numbers in every image are the running times. As shown in Fig. 4c, in the most cases, the larger the value of h is, the lesser the time cost. Moreover, the vanishing point locations are more accurate when the bandwidth is between 20 and 40. Considering the accuracy and efficiency, we think the proper bandwidth is 30.

2.5 Complexity of the algorithm

Given an $N \times N$ image, the first step of our method is the image segmentation based on dark channel prior and K-means, which are $O(N^2)$ and $O(N^2 \times K \times t)$ operations, separately. All approximate straight lines are extracted from the segmented region contours and are based on cumulative errors. Assuming that there are m pixels of contours and the number of approximate straight lines is n , the detection process is an $O(m)$ operation. Then, l road lines are selected based on vertical envelope lines and perspective quadrilateral constraint, in which the process is an $O(n)$ operation. Next, the candidate points are computed based on the proposed grouping strategies and the intersection principles, which is a $O(2^l)$ operation. Finally, the vanishing point is estimated by the meanshift clustering method, which is a $O(4^l)$ operation. Therefore, the time complexity of the proposed method is $O(N^2) + O(N^2 \times K \times t) + O(m) + O(n) + O(2^l) + O(4^l) = O(N^2)$.

3 Experimental results and analysis

To test the performance of the proposed algorithm, we compared the computation time and detection accuracy of our proposed algorithm with the gLoG filter-based method [6], Ding's method [15] and Kong's method [9]. Here, both Ding's method and our method is belonged to the edge-based methods. The other three are the texture-based methods, and the gLoG filter-based method is considered as the state-of-the-art approach. The dataset used to evaluate the performance of different approaches consists of 207 road images with various types of background, illumination, texture and colour information, and all images are normalised to the same size of 180×240 . In the dataset, one part is taken from the related literatures [6, 9], and the other part is downloaded from <http://www.image.baidu.com/>. To quantitatively measure the accuracy of vanishing point detection methods, the ground truth of every image in the dataset is calibrated first, where the ground truth is defined as the nearest visual vanishing point of the road. Then,

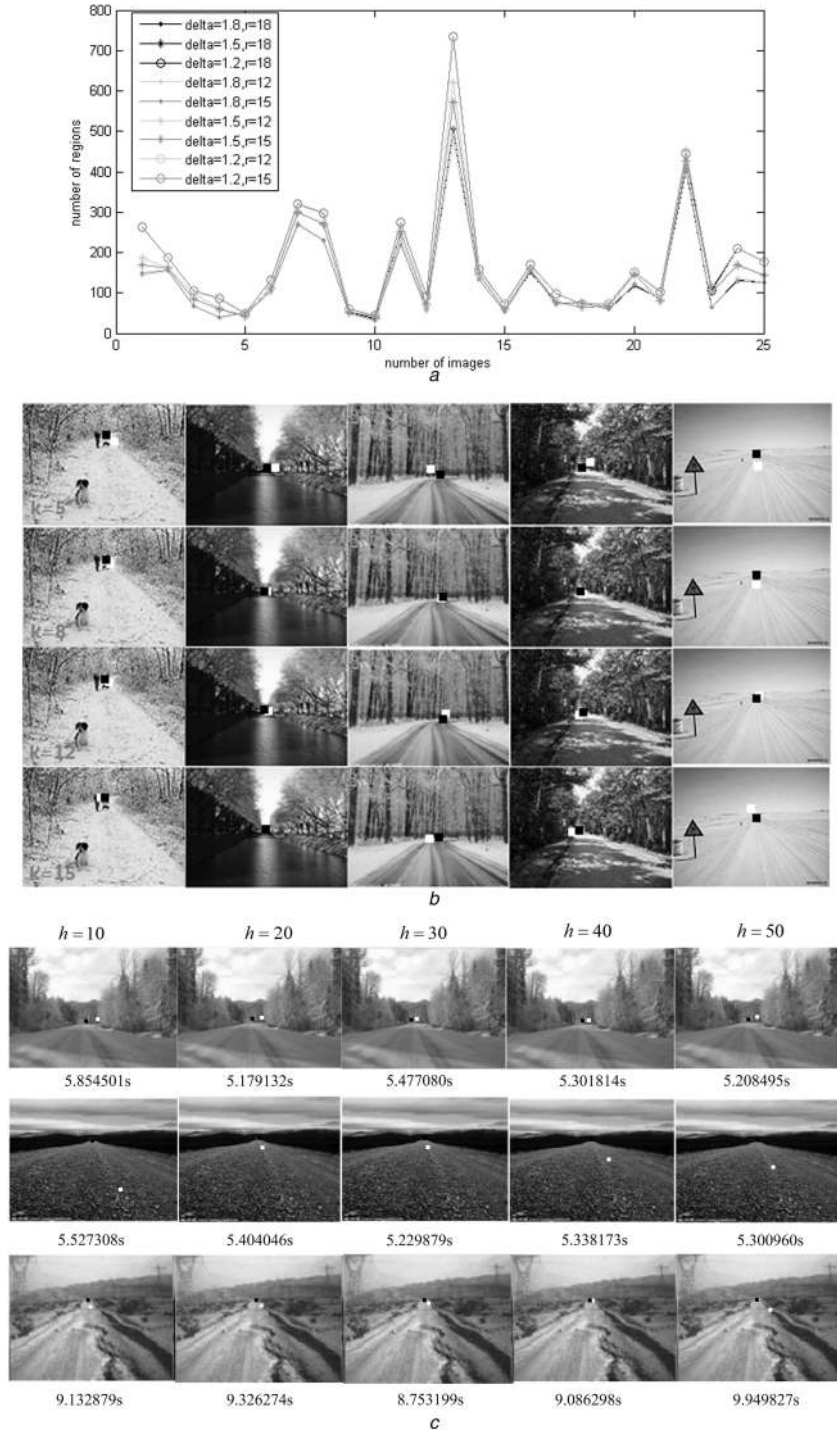


Fig. 4 Selection of parameter values

(a) Number of regions histograms with different rg and σ , (b) Vanishing point location compared with different cluster value of k , (c) Vanishing point location and run times compared with different bandwidths

the algorithms are running on *Matlab2014(a)* of the common PC with 3.3 GHz Intel i5 Processor and 8 GB memory space, and our algorithm used the following optimal parameters: $r = 7$, $r_g = 18$, $\sigma = 1.8$, $k = 8$, $m_p = 10$, $T_{err} = 10$ and $h = 30$.

3.1 Quantitative analysis of the comparison results

Fig. 5 shows the performance comparison results of different vanishing point detection approaches using 100 images, which are taken from the database of Kong [9], and various types of roads, background, illumination, texture and colour information have been included. As to the computation time, the curves in Fig. 5 show that our method and Ding's method are similar, but our method is faster than gLoG method [6] and the Kong's method [9]. Since the texture-based methods [6, 9] spent more time on the

voting stage, they usually cost a long time to detect the vanishing point. In order to perform the efficiency of our method more clearly, the average value and the standard deviation of the computation time of all methods are listed in Table 1. It is obvious that our method has the similar average value in run time with Ding's method. However, our method is more than 40 times faster than Kong *et al.* [9] method and Slow gLoG method. Besides, our method has the minimal standard deviation in computation time. This means that our method is more stable than the other three methods in computing speed.

To express the accuracy of the vanishing point detection, the Euclidean distance in pixels for every image used in Fig. 5a is also calculated between the location of the detected vanishing point and the location of ground truth [7, 8]. Fig. 5b shows the cumulative errors in the Euclidean distance of our proposed method and the

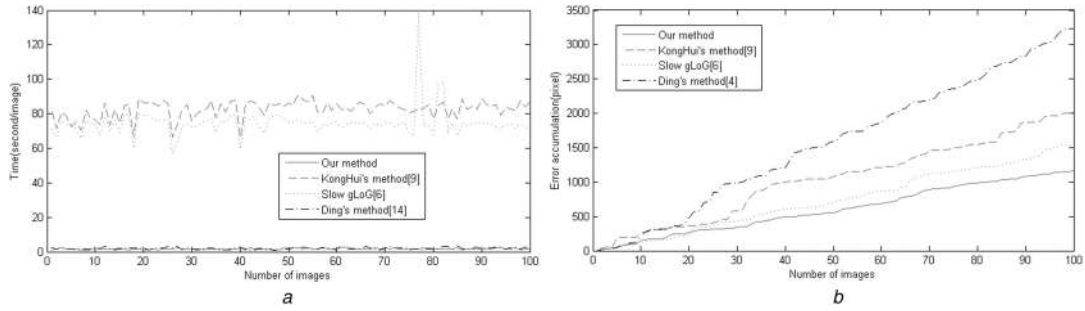


Fig. 5 Performance comparison of different methods

(a) Comparison results in terms of computation time, (b) Comparison results in terms of detection accuracy

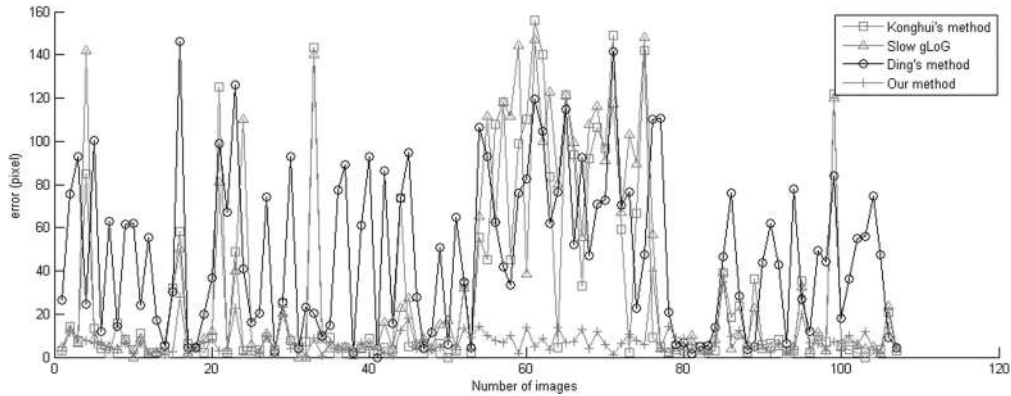


Fig. 6 Comparison of vanishing point detection methods in Euclidean distance errors between the location of the detected vanishing point and the location of ground truth

other three methods. The curves show that our proposed method achieves the best result among all the methods. The last four columns of Table 1 also list quantitative evaluation results of all methods in pixels. It indicates that our method achieves the best detection accuracy. The detection accuracy of our method is 23.9% better than the Kong's method, 60.6% better than the gLoG method and 63.8% better than the Ding's method. For the standard deviation error, our method achieves the minimal value. Thus, our method is more stable than the other three methods in detection accuracy.

Table 1 Run time and detection accuracy for different algorithms

Method	Our method	Ding <i>et al.</i> [14]	Kong <i>et al.</i> [9]	Slow gLoG [6]
time range (second/image)	1.2–2.8	0.9–3.9	57.3–138.5	66–99.9
average time (second/image)	1.76	1.95	74.55	82.66
standard deviation (second/image)	0.27	0.62	8.53	4.79
error range (pixel/image)	0–56.57	1.41–138.25	0–107.07	0–133.45
average error (pixel/image)	11.46	31.65	15.06	29.15
standard deviation error (pixel/image)	10.22	31.54	17.93	19.33

Table 2 Comparison of average Euclidean distance and standard deviation of Fig. 6

Method	Our method	Ding <i>et al.</i> [14]	Kong <i>et al.</i> [9]	Slow gLoG [6]
average error	7.3979	48.2832	29.7812	33.4943
standard deviation error	6.4921	36.6736	42.865	44.8373

Since the texture and background of the images in the database provided by Kong are still not very rich, we downloaded another 107 unstructured road images (See Fig. 9 for example) from <http://www.image.baidu.com/> to test the detection accuracy of our proposed algorithm with the other three methods. These images contain more complex texture and background. As shown in Fig. 6, the Euclidean distances of our method are compared directly with the gLoG filter-based method [6], Ding's method [15] and Kong's method [9]. We see that our method gives a more steady distance error than other three methods. The standard deviation errors listed in Table 2 also indicate this point. If we define that the Euclidean distance is not bigger than 25 pixels, then the detected vanishing point is regarded as correct detection. The correct detection rate of our method, Ding's method, Kong's method and gLoG method are 97.2, 36.4, 70.1 and 67.3, respectively. It means that our method achieves the best detection accuracy on the road images with complex texture and background.

3.2 Visual comparison results analysis

To provide a visual comparison, the images used in Fig. 5 are divided into urban road images, lane images, desert road images, road images after snow/rain, road images of the night and the road images with vehicles or pedestrians. Fig. 7 shows some results of vanishing point detection in different kinds of road images. In each group, the first row is the calculated vanishing point location of our proposed method and the second row is the location of the ground truth. We see that our method is able to accurately detect vanishing point in all types of road scenes, which means that our method is robust to different illumination conditions, road textures and backgrounds.

We also compared the detection results in visual of our proposed algorithm with the result of Ding *et al.*, Kong *et al.* [9], Moghadam and Starzyk [8] and Slow gLoG [6]. Fig. 8 shows a variety of representative road images used in [7] and Fig. 5. The white points are vanishing points detected by different methods, and the black points are the ground truth locations of vanishing points. We see that our algorithm gives more accurate results, and the vanishing point locations detected by our method are very close to the ground truth locations. However, the remaining algorithms



Fig. 7 Vanishing point detection results of our method

(a) Structured roads, (b) Unstructured roads, (c) Desert roads, (d) Snow or rain roads, (e) Dark roads, (f) Road images with vehicles or pedestrians

give an unsatisfying result both in straight and curved road images.

When considered the more complex road texture and background provided by the images used in Fig. 6 (Fig. 9 shows some examples). We found that Kong's method and gLoG method is often a failure to extract the vanishing point in these images. Compared with them, the vanishing point can be successfully detected by our method. There are two main reasons. One is that our method can filter out interfering information in background information by using the envelope of the vertical lines. The other is that our method obtains a more smoother road surface after using the dark channel prior method.

However, the performance of the proposed method is not always good. In some cases, as shown in Fig. 10, the proposed

method may achieve large errors of the vanishing point detection results. For instance, if there is no obvious vertical lines in the image (Figs. 10a and b), or the colour difference between the background and the road is small (Fig. 10c), the location of the detected vanishing point is not accurate. Moreover, if some big obstacles appear in the road region of the image (see Fig. 10d for example), our method may have a larger detection error.

4 Conclusion

Vanishing point detection for unstructured road scenes is important for road scene understanding. In this study, the proposed algorithm estimates the vanishing point based on the dark channel prior, and it calculates a series of vanishing point candidates using a new

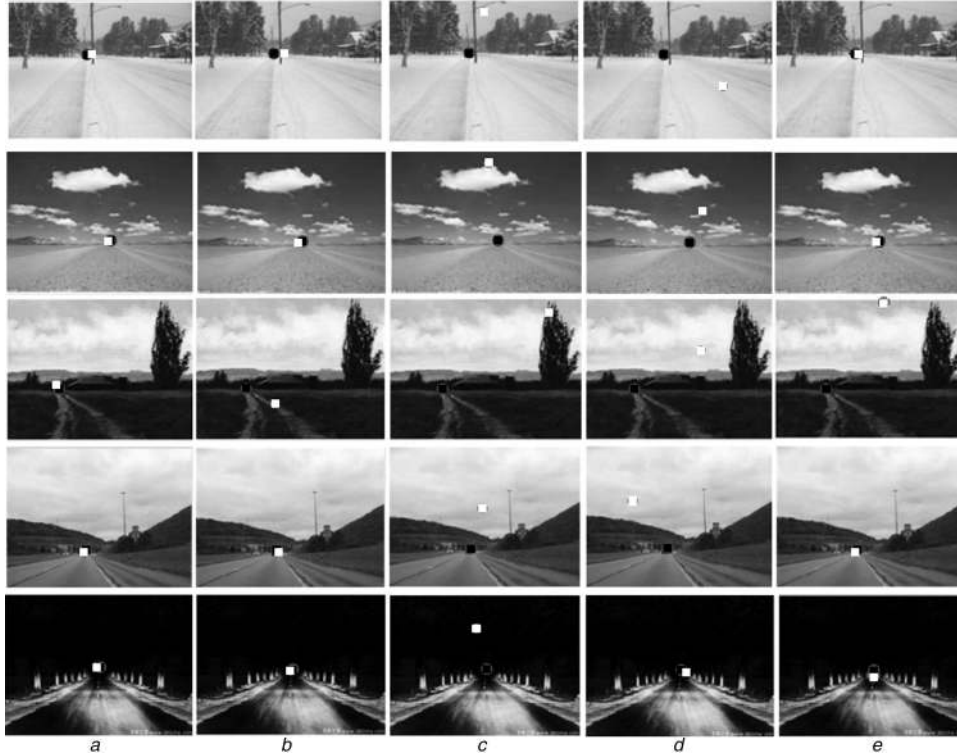


Fig. 8 Examples of vanishing point detection results by using
 (a) Our proposed method, (b) Ding's method [14], (c) Kong's method [9], (d) Moghadam's method [8], (e) Slow gLoG method [6]



Fig. 9 Examples of road vanishing point detection results on road images with complex texture and background. The black points denote the detection results of our proposed method, the black squares are the detection results of Ding's method, the black triangles are the detection results of Kong's method and the black stars represent the detection results of gLoG method



Fig. 10 Vanishing point detection results with large error
 (a) Vanishing point detection result on a mountain road image without obvious vertical lines, (b) Vanishing point detection result on a snow road image without obvious vertical lines, (c) Vanishing point detection result on a road image which has a small colour difference in the background and the road surface, (d) Vanishing point detection result on the road image with obstacles.

detection grouping and intersection detecting strategies. In view of the computation time and detection accuracy, the proposed algorithm gives an encouraging results, and it can be applied to many typical unstructured road scenes. However, further research is needed to improve the algorithm to enable it to solve more complex vanishing point detection scenarios and to expand its range of application, especially to road scenes with more than three curved paths.

5 Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61005034) and the Natural Science Foundation of Hebei Province (No. F2012203185).

6 References

- [1] Ding, W.L., Li, Y., Wang, W.F., *et al.*: 'Depth estimation of urban road image based on contour understanding', *Acta Opt. Sin.*, 2014, **34**, (7), p. 0715001
- [2] Zhang, Z.: 'A flexible new technique for camera calibration', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2000, **22**, (11), pp. 1330-1334

- [3] Wei, W.: 'Performance evaluating of some methods in 3D depth reconstruction from a single image' Blekinge Institute of Technology, School of Engineering. Independent thesis, 2009.
- [4] Miguel, A.S., Francisco, J.R., Luis, M., *et al.*: 'A color vision-based lane tracking system for autonomous driving on unmarked roads', *J. Auton. Robots*, 2004, **16**, (1), pp. 95–116
- [5] McCall, J.C., Trivedi, M.M.: 'Video based lane estimation and tracking for driver assistance: survey, system, and evaluation', *IEEE Trans. Intell. Transp. Syst.*, 2006, **7**, (1), pp. 20–37
- [6] Kong, H., Sarma, S.E., Tang, F.: 'Generalizing Laplacian of Gaussian filters for vanishing-point detection', *IEEE Trans. Intell. Transp. Syst.*, 2013, **14**, (1), pp. 408–418
- [7] Lu, X.Q.: 'New efficient vanishing point detection from a single road image based on intrinsic line orientation and color texture properties', *Opt. Eng.*, 2012, **51**, (3), p. 037001
- [8] Moghadam, P., Starzyk, J.A.: 'Fast vanishing-point detection algorithm in unstructured environments', *IEEE Trans. Image Process.*, 2012, **21**, (1), pp. 425–430
- [9] Kong, H., Audibert, J.-Y., Ponce, J.: 'Vanishing point detection for road detection'. Proc. IEEE CVPR, 2009, pp. 96–103
- [10] Rasmussen, C.: 'Grouping dominant orientations for ill-structured road following'. Proc. IEEE CVPR, 2004, pp. 1-470–1-477
- [11] Kong, H., Audibert, J.-Y., Ponce, J.: 'General road detection from a single image', *IEEE Trans. Image Process.*, 2010, **19**, (8), pp. 2211–2220
- [12] Alon, Y., Ferencz, A., Shashua, A.: 'Off-road path following using region classification and geometric projection constraints'. Proc. IEEE Conf. on Computer Vision Pattern Recognition, 2006, pp. 689–696
- [13] Ebrahimpour, R., *et al.*: 'Vanishing point detection in corridors: using Hough transform and K-means clustering', *IET Comput. Vis.*, 2012, **6**, (1), pp. 40–51
- [14] Ding, W.L., Li, Y., Wang, W.F., *et al.*: 'Vanishing point detection algorithm for urban road image based on the envelope of perpendicular and parallel lines', *Acta Opt. Sin.*, 2014, **34**, (10), p. 1015002
- [15] Ding, W.L., Li, Y.: 'Efficient vanishing point detection method in complex urban road environments', *IET Comput. Vis.*, 2015, **9**, (4), pp. 549–558
- [16] Kosecká, J., Zhang, W.: 'Video compass'. European Conf. on Computer Vision (ECCV), 2002, **4**, pp. 476–490
- [17] R.G.J. von Gioi, J., Morel, J.-M., Randall, G.: 'LSD: a fast line segment detector with a false detection control', *IEEE Trans. PAMI*, 2010, **32**, pp. 722–732
- [18] Alvarez, J.M., Gevers, T., Lopez, A.M.: '3D scene priors for road detection'. Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, 2010, pp. 57–64
- [19] Wu, Q., Zhang, W., Chen, T.: 'Prior-based vanishing point estimation through global perspective structure matching'. Proc. of IEEE Int. Conf. on Acoustic Speech and Signal Processing, 2010, pp. 2110–2113
- [20] He, K.M., Sun, J., Tang, X.O.: 'Single image haze removal using dark channel prior', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011, **33**, (12), pp. 2341–2353
- [21] MacQueen, J.B.: 'Some methods for classification and analysis of multivariate observations'. Proc. Fifth Berkley Symp. on Mathematical Statistics and Probability, 1967, pp. 281–297
- [22] Ding, W.L., Wang, W.F., Zhang, X.G., *et al.*: 'Extracting straight lines from building image based on edge orientation image', *Acta Opt. Sin.*, 2010, **30**, (10), pp. 2904–2910
- [23] Ding, W.L., Wang, W.F.: 'A novel line detection algorithm based on endpoints estimation'. Sixth CISP, 2013, **1**, pp. 400–404
- [24] Guru, D.S., *et al.*: 'A simple and robust line detection algorithm based on small eigenvalue analysis', *Pattern Recognit. Lett.*, 2004, **25**, pp. 1–13
- [25] Comaniciu, D., Meer, P.: 'Mean shift: a robust approach toward feature space analysis', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002, **24**, (5), pp. 503–519