Multimed Tools Appl (2017) 76:22979–22998
DOI 10.1007/s11042-016-4184-6

Fast lane detection based on bird's eye view and improved random sample consensus algorithm

Yong Ding[1] · Zheng Xu[2] · Yubin Zhang[3] · Ke Sun[3]

Received: 29 May 2016 / Revised: 13 November 2016 / Accepted: 18 November 2016 /
Published online: 25 November 2016

Abstract In order to ensure the safety of drivers, Advanced Driving Assistance System (ADAS) has drawn more and more attention. The Lane Departure Warning system is one of the most important parts of ADAS. However, fast and stable lane marking detection is the precondition of it under complex background. In this paper, we proposed a new lane detection method through bird's eye view and improved RANSAC (Random Sample Consensus) algorithm based on the inspiration that extraction of road features from remote sensed images. According to the bird's eye view of the road image, we can recognize the line marking through Progressive Probabilistic Hough transform instead of lane detection. Then, the detected lines are grouped by a new distance-based weighting scheme and we can get the fields of candidate lanes. For each of the fields, lanes are refined through improved RANSAC algorithm and fitted by double models. Hence, the road orientation can be predicted by the curvature and straight line's slope. At last, our experimental results indicated that the lane detection algorithm has good robustness and real-time under various road environment.

Keywords Bird's eye view · Lane detection · Random sample consensus

1 Introduction

With the popularization of vehicles and the increasing number of traffic accidents, traffic safety has become a widespread social problem of concern to active safety car was driving increasing attention.

✉ Zheng Xu
xuzheng@shu.edu.cn

[1] School of Computer Science and Information Security, Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin, China

[2] The third research institute of the ministry of public security, Shanghai, China

[3] School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin, China

Only in 2014, one statistic said it was estimated 1203 road accidents have happened and resulted in 451 people has lost their lives in China. [http://www.statisticbrain.com/car-crash-fatality-statistics-2/] But this problem didn't appear merely in China. An estimated 1.2 million people are killed in road crashes every year around the world, and as many as 50 million are injured [15].

With the development of video processing and cloud computing, many public security problems can be based on the processing of video data processing technology through video surveillance systems [9] [27]. Of course, these theories carry out the breakthrough of traditional technology of road traffic safety risk. Currently, Advanced Driving Assistance System (ADAS) has rapidly grown and arisen as a contribution to traffic safety, including Forward Collision Warning (FCW), Lane Departure Warning (LDW), Pedestrian Detection (PD), Intelligent Cruise Control (ICC) and so on. Recently not only luxury cars but some entry level cars are equipped with ADAS applications, such as Automated Emergency Braking System (AEBS) [17]. Road and lane detection is a critical component in ADAS in order to provide meaningful and consistent road shape information for navigational purposes.

Lane detection may not seem very complex, when the road markings are clear and the lane has a well-defined geometry. But the variety and uncertainty of road environments will have an effect on the accuracy of lane detection. It is a difficult problem that how to gain useful information from complex video or picture data. Hence it always requires complex vision algorithms, such as video structured description technology [25, 26]. The main problems of detection are the high processing cost of detection and the unconstrained road environment [18]. The common interference factors for lane detection are as follows:

- The road surface always has different shades because of illumination variations. Moreover the presence of shadows (projected by trees, buildings, bridges, or other vehicles) produces artifacts onto the road surface and thus alternates the road texture.
- Various lane markings (continuous, dashed) may occur, and their geometries always show straight line or curve. But there isn't a single model to describe the markings accurately. Further, road splitting or merging and the interference from roadside objects or shadows could worsen the detection.
- Road area may be significantly occluded by the other vehicles and pedestrians.
- Various bad weather (foggy, rainy, snowy, etc.) may occur.

Apart from the above, there are still some factors, and we don't list them all here. Above all, the urban scenario is not simple. Some features that are similar with lane markings would lead to error detection, but also they would be hardly predicted where the obstacles appear and where roads have complex textures that produce edges.

In ADAS, a variety of sensors could be used, such as monocular vision, stereo, LIDAR, vehicle dynamics information obtained from car odometer or inertial measurement unit (IMU) with global positioning information obtained using global positioning system (GPS) and digital maps. For lane detection, camera is the most important sensors and vision-based algorithm is the most prominent research area in consideration of the cost factor. Detecting lane markings from a digital road image through some vision algorithms is the typical method. Its process includes four basic steps: preprocessing, feature detection, fitting, and tracking.

The purposes of preprocessing are removing noise and preparing the image for the subsequent steps. And the researchers focus on how to extract the feature of lane and which model fit it well. Hence, the general algorithms can be divided into two classes roughly: based on the lane feature and based on geometric information.

Ideally, lane markings are white or yellow lines on a dark pavement. Thus, the basic features are usually used by some researchers such as edge, gradient and intensity, especially the edge feature in grayscale images because of lanes can create strong edges on the road [11, 23]. In other words, there are large gradients between lane and road. So edges of the input image can be extracted by some edge detector. Canny edge detector [2] has been widely applied because of its anti-interference. Then the Hough transform [16] is used to find the line instead of lane detection based on the clear detected edge, but it will waste much time. In addition, the modified Hough transform methods have been proposed for much quicker and enhanced memory efficiency [12]. However, methods based on edge and Hough transform have many problems in detecting curved lanes, sensitivity in various illumination conditions and road patterns [8]. But in color image, color feather is the most obvious feather. Some methods [4, 20] usually convert RGB to HSI or custom color spaces because RGB color space is difficult to express the lane color information. Then the luminance and chrominance components of a pixel are separately modeled. So this method will be sensitive to illumination.

In order to describe and fit lane markings well, some lane geometric models are proposed, including, line model [5], spline model [22], linear-parabolic model [10], quadratic curve model [13], hyperbola-pair model [3] and so on. However, it is not easy to define geometric information because the camera shakes and road environment frequently changed, especially when the lanes are blurred or the road image has much noise.

According to these methods, some researchers have presented some novel algorithms that select appropriate feather to extract and suitable model to fit. For example, Aly [1] proposed a three degree spline model to fit curve based on bird's eye view. But many parameters need to control through human operation and the speed of processing is not very fast. Zhang [21] proposed a multi-step curved lane detection algorithm that used the Hough transform extract the parameters of line and apply hyperbola-pair model to fit the curve. Son [19] presented an extract lane feather method through comparing Y-component values between lane color and other color in YCbCr space under various illumination conditions, and used the connected component clustering method in binary lane image. Even though the proposed method is invariant against various illumination changes, it is still difficult to handle several extreme conditions such as strong light reflection, blur lane marks and lane cracks. However, clustering will make computation complicate.

As we know, Extraction of road features from satellite or aerial images has been a long-term topic of research. In the remote sensed image, the road surface and road direction can be shown obvious. If remote sensed image is captured by highly accurate aerial digital camera systems, the lane markings also would be shown clearly. In consideration of the cost of equipment and requirement of lane detection, we present this method through the reference of feature extraction from remote sensed images. First of all, we convert the front view of the road image to bird's eye view through inverse perspective mapping, then the generated bird's view image is filtered by a two dimensional Gaussian kernel and processed through a new threshold processing method. Secondly, we detect the key fields of lanes through progressive probabilistic Hough transform after thresholding on the filtered image and group the lines through a new distance-based weighting computing scheme. Thirdly, we take a new search mode and an improved RANSAC algorithm to determine the lane from the candidate lanes and use double models to fit the line and curve respectively. The algorithm of this paper can detect lane well on complex road environment and has an ability of real-time, running at 50 Hz on 640,480 images on a typical machine with Intel Core 2.6 GHz machine.

The following sections introduce each step of this algorithm in detail. Section 2 gives the image main pre-processing, including view transition (that is, a new image that represents the same scene as acquired from a different position), image filtering, image threshold processing. Section 3 introduces the coarse detection algorithm of candidate lanes after pre-processing. And the key lane fields are got through groups these candidates. Section 4 gives the lane selection algorithm and double models fitting algorithm. And the experiment results and the analysis of the computational complexities are shown in Section 5.

2 Pre-processing

2.1 Inverse perspective mapping

The road image is captured by the camera that installs in front of vehicle. So in Fig. 1a, we can see the top of a road image is always sky or other outlier information that makes little contribution to lane detection. Moreover, two lanes approximate intersect at one point that called vanish point in this image. It always determines road area and road direction.

Choosing the object field is a key to reduce the computational work. Therefore, we divide the lane fields into the near and far field (as shown in Fig. 1b, theIpart means the near field, theIIpart means the far field). In the near field, the lane would be seen as the straight line.

The curves always show in the far fields. Thus the far field is the critical area of curved lane recognition. When we use the line model to fit the fields of lane, lane of the far field can be fitted by a curve model separately based on the parameters of the line model. Of course, the height of the camera that load in the front of car has an effect on the fields of lane. So we can set reasonable ROI (Region of Interest) based on the position of the vanish point.

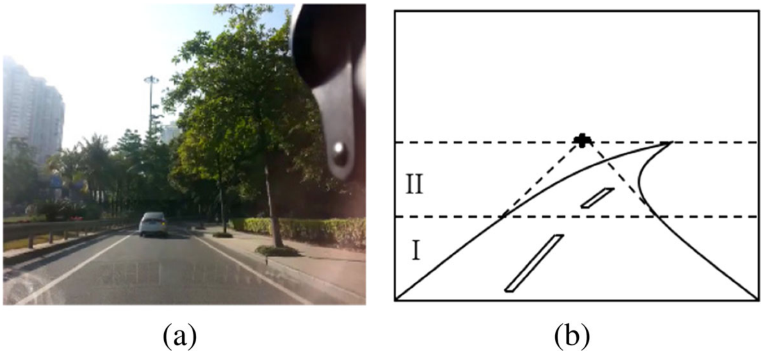


Fig. 1 **a** The road image sample, **b** The definition of the near and far fields

Because of the road is not absolutely flat, the vehicle driving will be jolted. If we detect the lanes from the front view, it won't be detected well because of the orientation of lane confirmed hardly. For recognizing the lane easily, we transform the front view of road image to bird's eye view through inverse perspective mapping (IPM) based on grey-level image. The bird's eye view changes the common view and makes lane be vertical. The ideal view is Fig. 2. According to the real road situation, we put the world coordinate system. The camera coordinate system with the origin at the camera and two-dimension image coordinate system. The image that camera captured is the projection that the three-dimensional Euclidean space mapped to the two-dimensional Euclidean image space, where:

- $W = \{X, Y, Z\} \in E^3$represents the 3D world space (world coordinate system), where the real world is defined. To elaborate on that a bit, the$X$direction of axis is moving forward along the road direction and parallel to ground, while the$Y$direction of axis is perpendicular to the axis$X$to right and also parallel to the ground. And the$Z$axis is perpendicular to the ground;
- $I = \{u, v\} \in E^2$ represents the 2D image space (image coordinate system), where the 3D scene is projected. The$u$direction is horizontal rightward and$v$direction is vertical down. According to the flatness assumption, the remapped image (that is road surface in W) is defined as the XY plane of the 3D world space, namely the $S \triangleq \{(X, Y, 0) \in W\}$ surface.

Based on the parameters above, the IPM transform process has mainly two steps, including $I \rightarrow S$ mapping and $S \rightarrow I$ mapping [14]. This process need some the camera intrinsic and extrinsic parameters, including optical center coordinates, pitch angle, yaw angle, the height of camera above ground and so on.

Assuming that the coordinates of the mounting position of the camera in the world coordinate system is $(d, l, h) \in W$.

The calibrated parameters of the camera are as follows: yaw angle γ (See Fig. 3a) represents the angle between the projection line η of the optical axis at the $Z = 0$ plane and X-axis; pitch angle θ (See Fig. 3b) represents the deviation angle of the optical axis to the $Y = 0$ plane; $2\alpha_u$ and $2\alpha_v$ represent the field of view of the camera in the horizontal and vertical directions, respectively.

The first step is $I \rightarrow S$ mapping. Choose a pixel (u, v) in the $m \times n$ image, the transform equation is:

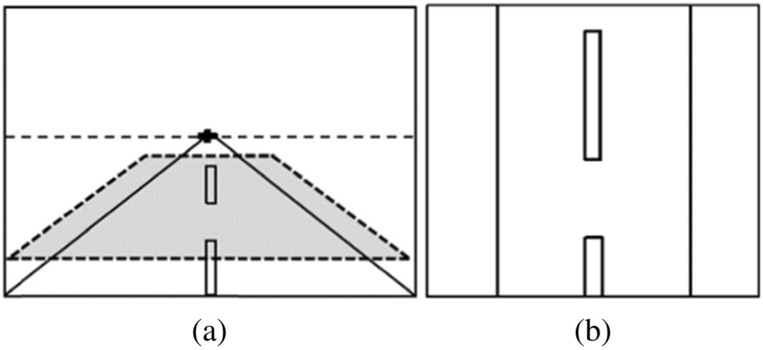
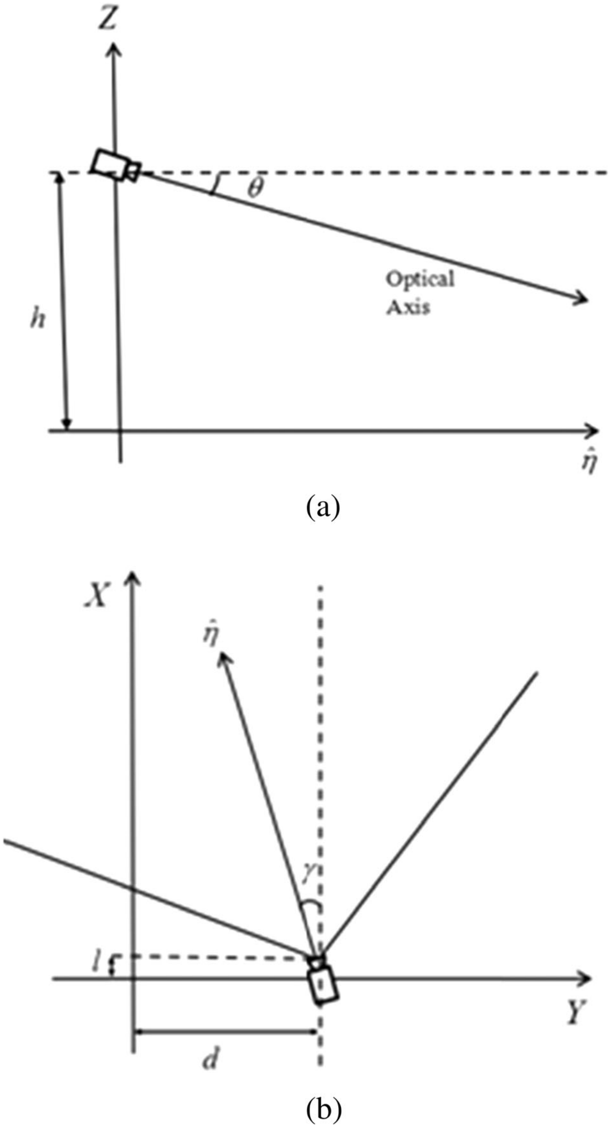


Fig. 2 The ideal image: **a** The front view of road image, **b** The bird's view of road image

Fig. 3 The camera calibration image: **a** the yaw angle, **b** the pitch angle



(a)

(b)

$$\begin{cases} x(u,v) = h\cot\left(\dfrac{2v\alpha_v}{n-1} - \alpha_v + \gamma\right)\cos\left(\dfrac{2u\alpha_u}{m-1} - \alpha_u + \theta\right) + l \\ y(u,v) = h\cot\left(\dfrac{2v\alpha_v}{n-1} - \alpha_v + \gamma\right)\sin\left(\dfrac{2u\alpha_u}{m-1} - \alpha_u + \theta\right) + d \\ \qquad z(u,v) = 0 \end{cases} \tag{1}$$

Eq.(1) returns the transformed point $(x, y, 0) \in S$ corresponding to point $(u, v) \in I$. Following similar algebraic and trigonometric manipulations, the inverse $S \rightarrow I$ mapping equation is:

$$\begin{cases} u(x,y,0) = \dfrac{(m-1)\times\left(\arctan\left(\dfrac{y-d}{x-l}\right)+\alpha_u-\theta\right)}{2\alpha_u} \\ v(x,y,0) = \dfrac{(n-1)\times\left[\arctan\dfrac{h}{\sqrt{(x-l)^2+(y-d)^2}}+\alpha_v-\gamma\right]}{2\alpha_v} \end{cases} \tag{2}$$

Eq. (2) is the projection transform. In fact, the array of pixels of coordinates $(x, y, 0) \in W$ is scanned and each pixel is assigned the value of the corresponding pixel of coordinates $(u(x, y, 0), v(x, y, 0))$.As a consequence, we can get the analogous top-view through transform each pixels in ROI of original image by Eq. (1) and (2).

2.2 Gauss processing and thresholding

After inverse perspective mapping, we have got the bird's view of road image. Some noise is generated by the image capture and the transformation among coordinate systems. In order to ensure the detection, we would like reduce image noise while the lane can be shown brighter and clearer on the darker background. In this paper, the image is filtered by a two dimensional Gaussian kernel which is a separable kernel accordingly. It will smooth the image in the vertical direction and sharpen in the horizontal direction.

The vertical direction is Gaussian low-pass filter and the convolution equation [1] is:

The horizontal direction is a second-derivative of Gaussian and the convolution equation [1] is:

$$f_v(y) = EXP\left(-\frac{y^2}{2\times\rho_y^2}\right) \tag{3}$$

$$f_u(x) = \frac{1}{\rho_x^2}EXP\left(-\frac{x^2}{2\times\rho_x^2}\right)\left(1-\frac{x^2}{\rho_x^2}\right) \tag{4}$$

where ρ_x and ρ_y are parameters that control the extent of filtering and have something to do with the width of lane.

After the filter processing, we can get the brighter and clearer lines that is the assumption of lanes on the dark background. Moreover, the advantages of the filter are its simplicity, low computational cost because of the separability of Gaussian kernel, and selection of the feature scale through the parameters ρ_x and ρ_y.

Accordingly, lanes in a filtered image are the lines of elongated, longitudinal, bright structures. In order to retain conspicuous lane feature as more as possible, then we keep and enhance the highest values of $p\%$.Other values that below the threshold are removed, that is assigned to be zero. And the setting of p in $m \times n$filtered image is:

$$p = \frac{mn-k}{4mn} \tag{5}$$

Assume the mean of filtered image values is μ and the standard deviation of them is σ.The k of Eq. (5) is the number of pixel points in $[\mu-\sigma, \mu+\sigma]$. As can be seen from the binary image (Fig. 4), the lane markings are more obvious. And they are parallel or near parallel each other. Furthermore, Fig. 4 shows the threshold process results in the continuous and dashed lane situation.

Lane markings can be recognized easier in the binary image, especially continuous lane situation. As we know, it will increase the difficulty for lane detection because of the dashed lane has less information. So we should take more adaptive lane detection algorithm, especially candidate lane fields' determination, which explained in the next section.

3 Candidate lane detection

As a general rule, the lanes always consist of the straight line or various curves. In consideration of the difficulty of curve detection, we can recognize the lines instead of lane detection based on bird's eye view through Progressive Probabilistic Hough transform [7] firstly. Thus the lane fields will be got, and then we can analyze the curve detection in these fields easier.

The polar equation of the normal Hough transform is:

$$r = x\sin\varphi + y\cos\varphi \tag{6}$$

where r is the vertical distance from the origin to the line, and the origin means the first pixel at left top image.φ is the angle between the vertical line and the x-axis (see Fig. 5).

The essence of Hough transform is the mapping between image space and parameter space. In other words, this method involves transforming each of the pixel points into a sine curve in a parameter space, which is defined by the parametric representation used to describe lines in the image space. In fact, we just transform the nonzero pixels in $x-y$ image space to some sine curve in the $r-\rho$ parameters space. Thus the problem of detecting collinear points can be converted to the problem of finding concurrent curves. In general, these n curves will intersect in $n(n-1)/2$ points corresponding to the lines between all pairs of figure points. Exactly collinear subsets of figure points can be found, at least in principle, by finding coincident points of intersection in the parameter plane [16].

As we know, the lanes always display as a group of nearly parallel straight lines and vertical state in bird's eye view. So the value range of r will be near zero. Based on the fundamental theories of Hough transform, setting reasonable search space can save more operative time.

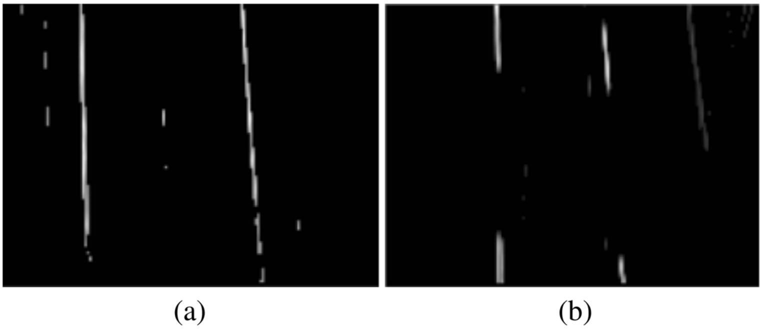
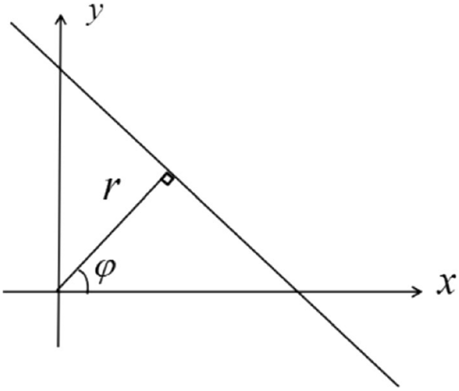

(a) (b)

Fig. 4 Binary image: **a** the continuous lane situation, **b** the dashed lane situation

Fig. 5 The normal parameters for a line



According to Hough transform basic theory, we set the acceptable parameters' range in r and φ, quantize the $r-\rho$ plane into a rectangular grid. So the value range of r and φ:

$$r_{\min} \le r \le r_{\max}, -\varphi_{\max} \le \varphi \le \varphi_{\max} \tag{7}$$

where r is the size of the retina, since points outside this rectangle correspond to lines in the picture plane that do not cross the retina.

According to these, we divide the confined plane into some small spaces and each subspace has a two-dimensional array of accumulators $A(r, \varphi)$ that the initial value of it is zero.

The next step is that we choose the effective pixels from the nonzero pixels randomly and put them into the set s.When the set s is null, the algorithm has finished. We choose a point (x_i, y_j) while this point is deleted from set s.The corresponding curve given by Eq.(6) is entered in the array by incrementing the count in each cell along the curve. That is, when the φ is searched in its range successively, and the corresponding r can be calculated through Eq.(6). Moreover, if the value of φ is in $[-\Delta\varphi, \Delta\varphi]$, the relevant accumulator cell would be added λ:

$$A(r, \varphi) = A(r, \varphi) + \lambda \tag{8}$$

or the relevant accumulator cell would be added one:

$$A(r, \varphi) = A(r, \varphi) + 1 \tag{9}$$

where $\Delta\varphi$ is range parameter and $\Delta\varphi < \varphi_{\max}$, λ is weight parameter.

After updating the relevant accumulator cell, the candidate lines are detected when the value of $A(r, \varphi)$ is bigger than the threshold. Otherwise, it doesn't stop calculating next φ until $A(r, \varphi)$ is over the threshold or r has gone through all the optional values.

After that, we choose the next point from set s until the set is null. Through this improved Hough transform, we can get some lines and treat these as candidate lanes.

To speed up the following algorithm, we can classify these candidate lanes. Because of the point and line is dual in Hough transform, (r_i, φ_i) and (r_j, φ_j) $(i \neq j)$ represent two detected lines that are random selected from the n detected candidate lanes. Then, we group the lines through the distance-based weighting equation:

$$d = \left|r_i - r_j\right| + \sigma\left|\varphi_i - \varphi_j\right|, i, j = 1, 2, 3, \ldots n \tag{10}$$

where σ is weight parameter, and

$$\varphi_k = \begin{cases} \varphi_k & \varphi_k \geq 0 \\ \pi + \varphi_k & \varphi_k < 0 \end{cases}, k = 1, 2, 3 \ldots \tag{11}$$

The distance of each candidate lanes has much to do with the d value. The smaller this value, the closer distance between two lines. When d is smaller than specified threshold, the two lines will be assigned a grouped collection. Through the first calculation of Eq. (10), we can get one or two collections. Then the next candidate is chosen and compared with other determined collections. By parity of reasoning, we can get some the values of d between each two lines through choosing the detected line. Some candidate lanes are grouped into collections by means of this scheme.

Furthermore, the fields of each grouped candidate lanes can be separated through some rectangle frames. Each rectangle height is all fixed at image height and width is determined by the intersection points of the parts image edge and the lines in each collections.

4 Line and curve fitting

The right lane is selected from each candidate lane collections. To ensure the algorithm adapt to different shapes of the lane, we utilize the improved RANSAC algorithm to select and double models to fit the reasonable lines and splines.

As you know, the road lane is consist of lines and curves. In the near of road fields, the lanes are thought as some straight lines. Therefore, the straight lane is fitted by the line model in the global field. But in the far of road fields, we fit the curves by the third degree Bezier spline, which is defined by:

$$P(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \tag{12}$$

where $t \in [0, 1]$, $P(0) = P_0$, $P(1) = P_3$. The shape of a standard third degree Bezier spline is controlled by the point P_1 and P_2 (Fig. 6).So getting the four control points of spline is the key to fitting spline.

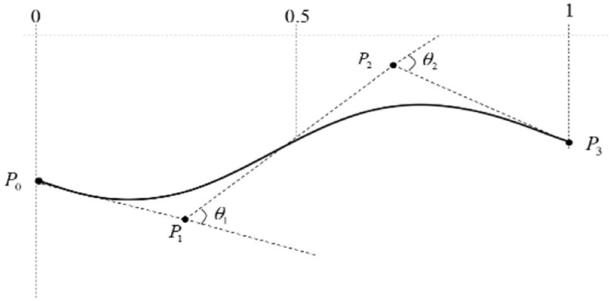


Fig. 6 Third degree Bezier spline

The RANSAC algorithm can be regarded as optimize fitting algorithm. For each candidate lane fields, the three main steps are choosing random samples, line and curve fitting and the selection scheme based on the degree of fitting.

4.1 Choosing random samples

For a grouped collection field, this samples from the nonzero points available in the region of interest where we have confirmed in the previous steps. The region is divided to two sub-regions, the samples for the line fitting are chosen in the entire region and for the spline fitting are chosen in sub-region 1 (Fig. 7). Moreover, we use a weighted sampling approach, with weights proportional to the pixel values of the threshold image.

4.2 Line fitting

For line fitting, we use the standard line equation as $y = ax + b$, and calculate the normal distance from every samples to the line. Assume we have$N$candidate lines, set the kth line is:

$$y = a_k x + b_k \tag{13}$$

Then calculate the normal distance based on$n$samples like (x_i, y_i).The distance formula is:

$$d_{ki} = \frac{|y_i - a_k x_i - b_k|}{\sqrt{a_k^2 + 1}}, i = 1, 2 \ldots n \tag{14}$$

At last, we calculate the summation of the distances:

$$d_k = \sum_{i=1}^{n} d_{ki}$$

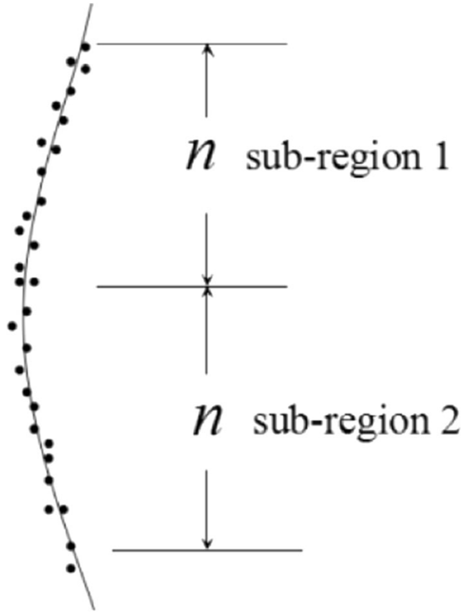


Fig. 7 The relationship between candidate points and sub-regions

When the d_k is smaller than the threshold value d^*, the line is the straight lane. If all of the d_k are larger than d^*, we should restart 4.1.If no candidate meets this condition, we can estimate whether straight line exists based on the proportion of nonzero pixels in this field. We will fine-tune the line through RANSAC algorithm if the nonzero pixels ratio is high enough.

We verified that, in cases where the two lane lines are dashed, there are few inlier points that fit the model and possibly a large number of outliers that have not been filtered out by the expected orientation criterion. So it has more advantages for candidates' selection after lane fields determined. There are some candidate lanes that contained a few of errors after Hough transform (Fig. 8), especially in dashed or other complex lane situations. And Fig. 8 is the Hough transform result of Fig. 4b, which is dashed lane situation.

Through straight lane selection algorithm, each grouped candidate lanes' fields can determine some accurate lanes more or less (Fig. 9).

4.3 Spline fitting

After getting straight lane, we also get each lane fields. And if the curve lane appears, it will always show in the far fields, that is the top half of bird's eye view. So we choose the sub-region 1 (Fig. 9) in each fields.

Firstly, we can get the control points through the property of the third degree Bezier spline based on the samples through the least square method and the definition of Bezier spline [1].

In normal RANSAC, we should calculate the normal distance from every point to the spline to decide the goodness to the spline. But this method would require solving a fifth degree equation for every such point and need much time to process. Hence, we propose a formula to decide the goodness to the spline by using an efficient iterative way. The formula define as:

$$\text{score} = s\left(1 + \frac{m_1 L + m_2 \theta}{2}\right) + m_3 \frac{m}{M} \tag{15}$$

Where *score* is to decide the goodness to the spline.s is the raw score for the spline (the sum of pixel values of the spline), L is the normalized spline length measure defined as $L = l/v - 1$ where l is the spline length and v is the image height and so $L = 0$ means we have a longer spline and $L = -1$means a shorter spline, θ means the degree of curvature and defined as $\theta = \theta'/2$,where $\theta' = \cos(\theta_1 - \theta_2)$, θ_1 and θ_2 are the angles between lines joining the spline's control points. (See Fig. 7). The m is number of nonzero pixels in the neighborhood of

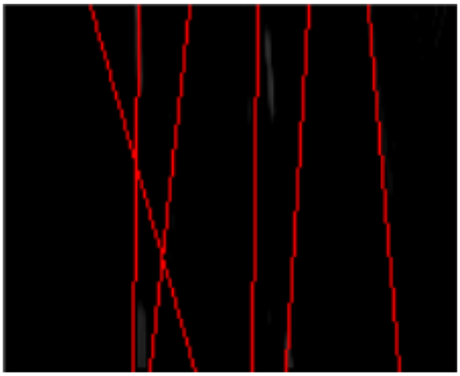


Fig. 8 The detected candidate lanes

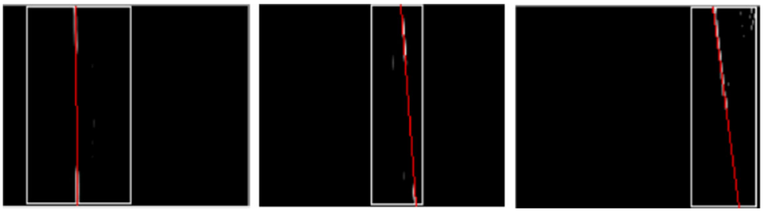


Fig. 9 The confirmed lanes in each fields

candidate lane and the M is number of all pixels. Furthermore, m_1 , m_2 and m_3 are regularization factors.

When the score is larger than the threshold value, the spline is the curve lane. If all of the score are smaller than the threshold value, we should restart choose random samples and calculate control points Bezier spline. And this formula also takes into account the straightness, the nonzero pixels ratio and length of the spline.

Above all, this paper set the screening mechanism to ensure the accuracy of lane detection and use double models to fit. The double models increase the probability of dashed lane fitting, but also have function of curve recognition. θ_1 and θ_2 determine if there is a curve ahead. When the curve lane appears, we can estimate the road orientation by comparing the slope between the straight lane and the P_0P_3 in the Bezier spline.

5 Experiment results

We use the image of 640×480 and approx. 24 frames to validate the proposed algorithm. For reliable verification, it is applied the same algorithm in various road environment. The algorithm was implemented by using OpenCV (Open source Computer Vision) that provides image processing library and Visual Studio 2012 installed Intel(R) Core (TM) i5 CPU@2.60GHz, 4GB RAM, the Windows 8 laptop.

5.1 Lane detection results

A sample image (Fig. 10a) is experimented by this method at first. Moreover, the figure shows most of lane area in the lower portion of the picture. Ideally, the lane lines always are parallel to each other. In fact, the road surface is not always flat, but also has ascent or descent. Because of

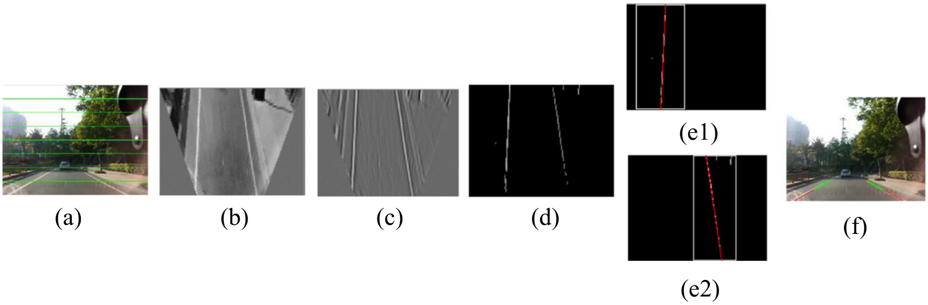


Fig. 10 **a** the sample image; **b** the actual bird's view; **c** the filtered image; **d** the threshold image; **e1**,**e2** the detected and grouped line image **f** the detection result image

IPM need some camera parameters that relate to the actual environment, the processing always has a little deviation. The Fig. 10 shows the processing results of each steps when IPM result is not well. We can see the bird's eye view and the filtered image in Fig. 10b, c. Through an adaptive threshold process based on bird's eye view, the lane information is more distinguished (see Fig. 10d).

Then we use Progressive Probabilistic Hough transform and a new distance-based group method and get the detected lines in each field of grouped candidate lanes. So each of grouped fields has more or less candidate lanes. To ensure the accuracy of lane detection, the RANSAC algorithm makes the lane fit well include straight lines and curves lastly. The Fig. 10e1, e2 show the grouped fields and determined straight lanes. The curve is fitted by the third Bezier spline through getting the key part that can reflect the feather of curve.

Above all, a sample of the output can be seen in Fig. 10f. And the green line refers to the detected curved and the red line refers to the detected straight lane. Because the straight line is the special shape of the curve. When the lanes are all straight, the green line will show straight. Therefore, the experiment of the sample proved that, in a tolerant error range of IPM, lane markings can be still detected well.

Furthermore, if we choose the spline fitting only, many lanes would not be detected by the more strict regression criterion, especially the dashed lane. Hence, we can keep the higher detection rate with double models.

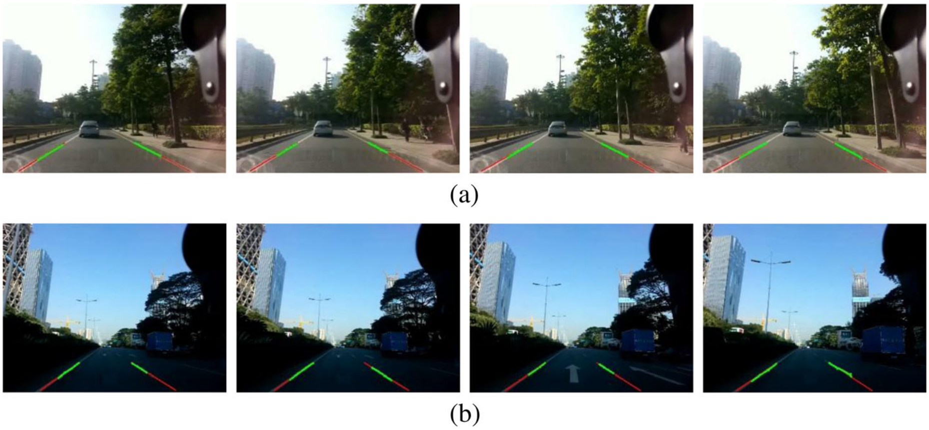
5.2 Detection results analyses in complex environments

The complexity of the road environment often embodies various illumination changes. Different light intensity has a significant effect on the color of the road surface and the clear extent of lane markings. To test the performance and stability of this vision-based method according to illumination changes, we consider illumination conditions for constructing the dataset.

In the actual scene, illumination conditions mainly include natural light and artificial light. The natural light changes are caused by the weather or the position of sun. And the changes of artificial light often occur at night or in the tunnel and are caused by the characteristics of various vehicle lamps and streetlamps on the roadside.

Figure 11a, b shows lane detection results with the proposed method for several road image sequences under front lighting or back lighting situation. We can see the obvious color difference and lane's color feature among different illumination conditions. Lane detection is not affected obviously by light intensity in daylight. Even if we test this algorithm at night, the lane still can be detected in slightly lower light (see Fig. 11c). In the night, many vehicles in the road can lead to many bright taillights in road image. Ahead of test camera, we can notice some bright vehicle taillights through Fig. 11c. Although these taillights affect the gradient of road image, the lanes still can be detected by this method.

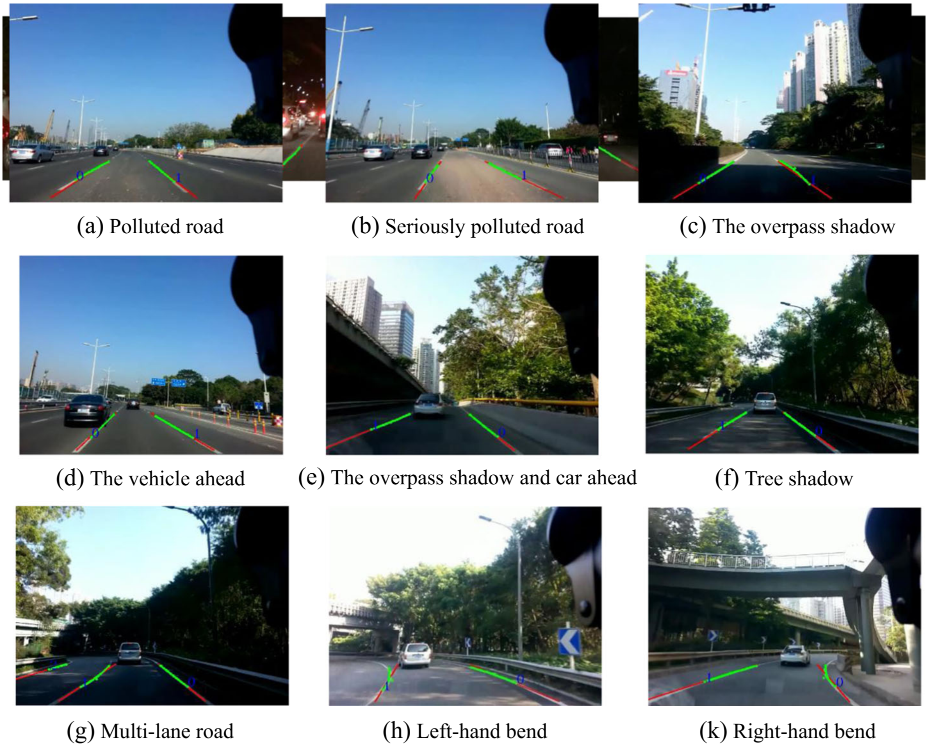
Apart from the influence of illumination condition, the road environment has usually the various vehicles, the various shadows, blot and so on. It is more difficult to detect the right lanes due to these disturbing factors. We collect some different complex situations of urban streets with shadow or vehicle ahead and test the robustness of this method. Some experiment results can be seen from the outputs (see Fig. 12).when a little stain appears on the road, the lane still can be detected (Fig. 12a, b). The information of lane markings in road image plays an important role on visual-based lane detection. So if the stain is up to cover the lane totally, lane detection will fail because of unable to distinguish out lane information. Various shadows


(a)

(b)

Fig. 11 Lane detection results in various illumination conditions: **a** front lighting, **b** back lighting, **c** night

often generate obvious the edge of the shadows and mix the lane markings. Figure 12c, f, g, h show the well detection results in different shadow situations, including tree shadow, building shadow, bright shadow and so on. And a few vehicles on road have little influence on lane detection (see Fig. 12d). Except for straight lane detection, Fig. 12h, k show curve detection, including left turn and right turn. Because of setting ROI, the spline is not shown very long. We can also know the road orientation by the curvature of spline and slope of the straight line.


(a) Polluted road

(b) Seriously polluted road

(c) The overpass shadow

(d) The vehicle ahead

(e) The overpass shadow and car ahead

(f) Tree shadow

(g) Multi-lane road

(h) Left-hand bend

(k) Right-hand bend

Fig. 12 Lane detection results in some complicate but common road situation situation

The complexity of the road environment is hard to describe completely. When bad weather appears or lane information is covered, this method cannot recognize the correct lane.

And Lane detection is used in ADAS, so the demands are quite high for its robustness and real-time. Hence we should analyze the robustness of this method on the complicate situation and the efficiency of it. Next, we collect some road videos on different types of urban streets, with shadows and vehicles ahead, and including straight and curved streets. Thus, we choose some driving scenarios considering these factors. And then, statistical data are recorded for the correct lane number and accuracy equaling the number of correct lanes divided by the total sample lane number. The accuracy of our experiment is shown in Table 1.

Table 1 mainly falls into two parts: the detection rate of clips under comparatively simple disturbing factors and the overall detection rate.

Half of the lanes' sum in Video 1 are dashed lines, such as Fig. 12c. And this video contains bright shadow, building shadow and so on. The road environment of Video 2 is the most complex in these test videos. Tree shadow makes dappled patterns on road surface and mix lane information. And lanes in Video 3 are continuous lines mostly, such as Fig. 12e. Furthermore, Video 2 and Video 3 have curve situation. In Video 4, the lanes are almost dashed lines, and the road surface has much dirt and stains that cover the lane information.

The calculation of the detection rate is the proportion of correctly detected lanes' number and lanes' sum. According to Table 1, we can know detection rate under various disturbed road environment. And the experimental data indicate that the detection rate is mostly above 90% under various road environment. So the algorithm of this paper can detect lane well whatever the lane is continuous, or dashed. Moreover, we can find that the detection rate of Video 4 is lower than others because of more disturbing factors in it. And the major reason for this is that the traffic block and dusts of road surface result in lack of lane information. Although this method can resist the effect of shadows to some extent, guardrail shadow and tree shadow still can effect on the detection results.

At last we can set reasonable the number of iterations, the speed of the test run can be controlled in 35 ms. Because one second of video has 24 frames, the process time can't exceed 50 ms utmost. So this method is sufficiently fast for real-time applications.

To summarize the analysis of the results, this algorithm has higher accuracy under various road environment and better efficiency of this algorithm. However, when too many disruptive

Table 1 lane detection rate of the proposed algorithm in various environments

Test Video	Frame Number	Sum of Lanes	Sum of Detected Lanes	Detection Rate (%)
1	464 (total)	842	784	93.11%
	208 (shadow)	374	346	92.51%
2	481 (total)	942	874	92.78%
	264(tree shadow, curve)	528	483	91.48%
3	1097 (total)	2082	1978	95.00%
	148 (undisturbed)	296	294	99.32%
	211 (other vehicle)	422	404	95.73%
	240 (guardrail shadow)	480	441	91.88%
4	1202(total)	2404	2169	90.22%
	468(dirty road)	936	769	82.16%

factors appear, especially for interfered with similar feature, the probability of false detection will increase.

6 Conclusion

In this paper, we focused on lane detection which is important in driving safety. Although it has some limitations in challenging scenarios such as various shadows and blur lane markings situations, the proposed method shows stable detection performance under complex environment.

To ensure the stable and real-time of lane detection, we reduced the computational complexity by set adaptive Hough search space and group scheme. Moreover, the improved RANSAC algorithm can select the correct lane from candidate lanes effectively. Lastly, we used double models to fit straight lane and curve respectively which can avoid omitting detection of the dashed lane.

Experimental results of the proposed method showed an average detection rate of above 90% under various conditions with an execution time of 35 ms, which is sufficiently fast for real-time applications and lane departure warning system. What is more, we can get more accurate lane and spent less time based on reasonable parameter setting. Of course, in video data processing, we may introduce cloud computing technologies such as distributed storage and computing, data retrieval of huge and heterogeneous data, provide multiple optimized strategies to enhance the utilization of resources and efficiency of tasks [6, 24, 28–30].

Even though the proposed method is effective to resist interference especially for against various illumination changes, it is still difficult to handle several extreme conditions such as bad weather situation, strong light reflection, vehicle occlusion and so on. These extreme conditions may make lane marks being disappeared in captured images. It is difficult for vision-based lane detection in these conditions. To overcome these difficulties, we should use other sensors or the camera with higher resolution.

Acknowledgements This research has been funded by Guangxi Natural Science Foundation Project No. 2014GXNSFCA118014, Innovation of Guangxi Graduate Education No.XJYC2012020

References

1. Aly M (2008) Real time detection of lane markers in urban streets[C]. IEEE Intelligent Vehicles Symposium. IEEE, Eindhoven, pp. 7–12
2. Canny J (1986) A computational approach to edge detection [J]. IEEE Trans Pattern Anal Mach Intell 8(6): 679–698
3. Chen Q, Wang H (2006) A real-time lane detection algorithm based on a hyperbola-pair model[C]. IEEE Intelligent Vehicles Symposium. IEEE, Tokyo, p 510–515
4. Chin KY, Lin SF (2005) Lane detection using color-based segmentation. In IEEE intelligent vehicles symposium [C]:706–711
5. Collado, Hilario C, Escalera A, Armingol JM (2005) Detection and classification of road lanes with a frequency analysis[C]. IEEE Intelligent Vehicles Symposium. Nevada, USA, 7883
6. Fan C, Wang L, Liu P, Lu K, Liu D (2016) Compressed sensing based remote sensing image reconstruction via employing similarities of reference images. Multimedia Tools Appl 75(19):12201–12225
7. Galamhos C, Matas J, Kittler J (1999) Progressive probabilistic Hough transform for line detection[C]. Proceedings of computer vision and pattern recognition Fort Collins, co, IEEE, USA, 23–35

8. Hsiao PY, Yeh CW (2006) A portable real-time lane departure warning system based on embedded calculating technique[C]. 2006 I.E. 63rd Vehicular Technology Conference 6:2982–2986
9. Hu C, Xu Z, Liu Y, Mei L (2015) Video structural description technology for the new generation video surveillance systems[J]. Frontiers of Computer Science 9(6):980–989
10. King HL, Kah PS, Li-Minn A (2009) Lane detection and kalman-based linear-parabolic lane tracking [C]. Proceedings of IEEE international conference on intelligent human-machine systems and cybernetics. IEEE, Hangzhou, p 351–354
11. Kong H, Audibert JY, Ponce J (2010) General road detection from a single image [J]. IEEE Trans Image Process 19(8):2211–2220
12. Kuk JG, An JH, Ki H, Cho (2010) Fast lane detection and tracking based on Hough transform with reduced memory requirement [C]. In IEEE conference on intelligent transportation systems 1344–1349
13. Lu WN, Zheng YC, Ma YQ, et al. (2008) An integrated approach to recognition of lane marking and road boundary[C]. Proceedings of International Workshop on Knowledge Discovery and Data Mining. University of Adelaide, Australia, p 649–653
14. Massimo B, Alberto B, Alessandra F (1998) Stereo inverse perspective mapping: theory and applications [J]. Image Vis Comput 16:585–590
15. Peden M, Scurfield R, Sleet D, Mohan D, Hyder A, Jarawan E, Mathers C (2013) Global plan for the decade of action for road safety 2011–2020[R]. UN: World report on road traffic injury prevention (5):1–21
16. Richard OD, Peter EH (1972) Use of the hough transformation to detect lines and curves in picture [J]. Graphics and Image Processing 15(1):11–15
17. Ryosuke O, Kajiwara Y, Kazuaki T (2014) A survey of technical trend of ADAS and autonomous driving [C]. VLSI design, automation and test (VLSI-DAT), 2014 International Symposium on 4:1–4
18. Sibel Y, Gökhan Y, Ekrem D (2013) Keeping the vehicle on the road: a survey on on-road lane detection systems [J]. ACM Comput Surv 46(1):1–43
19. Son J, Yoo H, Kim S et al (2015) Real-time illumination invariant lane detection for lane departure warning system [J]. Expert Syst Appl 42(4):1816–1824
20. Sun TY, Tsai SJ, Chan V (2006) HSI color model based lane-marking detection[C]. In IEEE conference on intelligent transport system 1168–1172
21. Wang J, An X (2010) A multi-step curved lane detection algorithm based on hyperbola-pair model[C]. IEEE International Conference on Automation and Logistics. HK and Macao 132–137
22. Wang Y, Teoh E, Shen D (2004) Lane detection and tracking using b-snake [J]. Image Vis Comput 22(4): 269–280
23. Wang JG, Lin CJ, Chen SM (2010) Applying fuzzy method to vision-based lane detection and departure warning system [J]. Expert Syst Appl 37(i):113–126
24. Wang L, Song W, Liu P (2016) Link the remote sensing big data to the image features via wavelet transformation. Clust Comput 19(2):793–810
25. Xu Z, Hu C, Mei L (2006) Video structured description technology based intelligence analysis of surveillance videos for public security applications [J]. Multimedia Tools Appl 75(19):12155–12172
26. Xu Z, Liu Y, Mei L, Hu C, Chen L (2015) Semantic based representing and organizing surveillance big data using video structural description technology [J]. J Syst Softw 102:217–225
27. Xu Z, Mei L, Liu Y, Hu C, Chen L (2016a) Semantic enhanced cloud environment for surveillance data management using video structural description [J]. Computing 98(1–2):35–54
28. Xu Z, Mei L, Hu C, Liu Y (2016b) The big data analytics and applications of the surveillance system using video structured description technology [J]. Clust Comput 19(3):1283–1292
29. Xu Z, Zhang H, Hu C, Mei L, Xuan J, Choo K-KR, Sugumaran V, Zhu Y (2016c) Building knowledge base of urban emergency events based on crowdsourcing of social media. Concurrency and Computation: Practice and Experience 28(15):4038–4052
30. Xu Z, Zhang H, Sugumaran V, Choo K-KR, Mei L, Zhu Y 2016d Participatory sensing-based semantic and spatial analysis of urban emergency events using mobile social media. EURASIP J Wireless Comm and Networking 44. doi:10.1186/s13638-016-0553-0

Yong Ding male, born in 1975. Professor in Guilin University of Electronic Technology. His main research interests include cryptography and information security.

Zheng Xu was born in Shanghai, China. He received the Diploma and Ph.D. degrees from the School of Computing Engineering and Science, Shanghai University, Shanghai, in 2007 and 2012, respectively. He is currently working in the third research institute of ministry of public security and as a postdoc at Tsinghua University, China. His current research interests include topic detection and tracking, semantic Web and Web mining. He has authored or co-authored more than 70 publications including IEEE Trans. On Fuzzy Systems, IEEE Trans. On Automation Science and Engineering, IEEE Trans. On Cloud Computing, IEEE Trans. On Emerging Topics in Computing, IEEE Trans. on Systems, Man, and Cybernetics: Systems, etc.

Yubin Zhang male, born in 1990. Postgraduate student of Guilin University of Electronic Technology. Major in applied mathematics. His main research interests include image processing, computer vision and pattern recognition.

Ke Sun male, born in 1972. Associate Professor in Guilin University of Electronic Technology. His main research interests include image processing, computer vision and pattern recognition.