International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 34, No. 2 (2020) 2055005 (18 pages)

DOI: 10.1142/S0218001420550058

World Scientific
www.worldscientific.com

# Lane Marking Detection Based on Segments with Upper and Lower Structure

Junjie Huang

*Department of Automation, University of Science and Technology of China, Hefei 230026, P. R. China*

*Institute of Applied Technology
Hefei Institutes of Physical Science
Chinese Academy of Sciences, Anhui Engineering Laboratory for Intelligent Driving Technology and Application, Innovation Research Institute of Robotics and Intelligent Manufacturing
Chinese Academy of Sciences
Hefei 230088, P. R. China
junjieh@mail.ustc.edu.cn*

Zhiling Wang*, Huawei Liang†, Linglong Lin‡ and Biao Yu§

*Institute of Applied Technology
Hefei Institutes of Physical Science
Chinese Academy of Sciences, Anhui Engineering Laboratory for Intelligent Driving Technology and Application, Innovation Research Institute of Robotics and Intelligent Manufacturing
Chinese Academy of Sciences
Hefei 230088, P. R. China
*zlwang@hfcas.ac.cn
†hwliang@iim.ac.cn
‡keith_lin@rntek.cas.cn
§byu@hfcas.ac.cn*

Fei Dong¶ and Yan Xu‖

*Shanghai Fire Research Institute of MEM
Shanghai 200032, P. R. China
¶dongfei@shfri.cn
‖xuyan@shfri.cn*

Received 20 December 2018
Accepted 19 March 2019
Published 14 June 2019

† Corresponding author.

An effective and accurate lane marking detection algorithm is a fundamental element of the intelligent vehicle system and the advanced driver assistant system, which can provide important information to ensure the vehicle runs in the lane or warn the driver in case of lane departure. However, in the complex urban environment, lane markings are always affected by illumination, shadow, rut, water, other vehicles, abandoned old lane markings and non-lane markings, etc. Meanwhile, the lane markings are weak caused by hard use over time. The dash and curve lane marking detection is also a challenge. In this paper, a new lane marking detection algorithm for urban traffic is proposed. In the low-level phase, an iterative adaptive threshold method is used for image segmentation, which is especially suitable for the blurred and weakened lane markings caused by low illumination or wear. In the middle-level phase, the algorithm clusters the candidate pixels into line segments, and the upper and lower structure is used to cluster the line segments into candidate lanes, which is more suitable for curve and dashed lane markings. In the high-level phase, we compute the highest scores to get the two optimal lane markings. The optimal strategy can exclude interference similar to lane markings. We test our algorithm on Future Challenge TSD-Lane dataset and KITTI UM dataset. The results show our algorithm can effectively detect lane markings under multiple disturbance, occlusions and sharp curves.

*Keywords*: Lane marking detection; intelligent vehicle; upper and lower structure; dashed and curve road.

## 1. Introduction

Lane marking detection plays an important role in Lane Departure Warning System, Intelligent Vehicle System and Intelligent Transportation System, which are to ensure road safety. The objective of lane marking detection is to extract lane markings from a noisy background and locate their position. The extension of application scenarios poses challenges for lane marking detection. Because there are vehicles, pedestrians, shadows of trees, ground characters, guided arrows, crosswalk, abandoned old lane markings, ruts, blurred lane markings on the road. And there are car lights and low illumination at night. Besides, the lane marking detection on the dashed and curve road is still a challenge. The key problem of dashed and curve detection is how to select the correct candidate points for fitting.

According to the initial image used, the lane marking detector can be divided into two categories: original image-based and inverse perspective mapping (IPM) image-based. In the original image, vanishing point is a very useful feature. It can reduce the search space and filter out the false detections that do not pass through it.[9,16,19] However, it is not easy to estimate the vanishing point correctly in an image with a complex background. Su *et al.*[19] used v-disparity to find the accuracy position of the vanishing point. Yoo *et al.*[9] used a probabilistic voting procedure based on the line segment strength to estimate the vanishing point of lanes. In addition, a flaw in original image is that object is big when near and small when far. When the near lane marking is missing, it will cause the failure of lane marking detection. Moreover, the geometric characteristics of lane markings cannot be well utilized on the original image. The use of inverse perspective map can make better use of the geometric characteristics of lane markings.[2,7,8,15] Smart *et al.*[17] proposed a stereo pre-filter to remove out-of-plane regions from the IPM, which could reduce the number of false

positive lane marking-candidates. It is important to note that the effect is limited by the accuracy of stereo matching. This paper aims at lane marking detection under monocular camera, but any stereo algorithm can be used to reduce false detection.

For the lane marking detection methods, Refs. 19, 16 and11 mentioned that the technique of detecting lane markings commonly used are categorized into two classes: feature-based and model-based. Li *et al.*[12] proposed that lane detection usually consists of two parts: feature detection and model fitting. Niu *et al.*[14] proposed four processes in lane marking detection: lane feature extraction, noise reduction, model fitting and lane generation. No matter what steps, the important prerequisite is robust image segmentation or initial image binarization. Image segmentation is the low-level but very important phase. Under-segmentation will cause the loss of lane marking information. Over-segmentation will bring too much interference, which is not conducive to subsequent operation. The feature-based method is the most commonly used in the image segmentation. The selection of feature types and their thresholds are the key points. There are mainly two methods: empirical threshold method and adaptive threshold method. Empirical thresholds are generally fixed, which are more suitable for single scenarios. In order to extend its applicability, many algorithms have enhanced gradient or grayscale. Reference 3 used histogram equalization to enhance the contrast between lane marking and road. Reference 4 used gradient enhancement, but eventually the gradient threshold was fixed at 10. In order to adapt to different lighting and various road conditions, adaptive threshold methods were proposed. Otsu algorithm obtains the adaptive threshold through maximizing the variance of the histogram. But it is more suitable for the segmentation of two kinds of objects with great differences. The image segmentation method used in generic obstacle and lane detection system (GOLD) is more suitable for the blurred images.[2] But if too much smoothing is done, the weakened lane markings will be missed. In this paper, we propose an iterative adaptive threshold method, which can eliminate a lot of disturbances without eliminating too much information of lane markings.

Image segmentation followed by the noise reduction stage. At this stage, many feature-based and model-based detection methods are proposed. Feature-based methods are very effective on simple and clean roads. But for complex road surface, some features will be weakened or even invalid. Commonly used features are edge, color, intensity, geometric shapes etc. But they are sensitive to noise and parameter threshold. The more complex the road, the less the common characteristics. Reference 20 used several low-level visual features so that the detector could not fail when one of the features gives incorrect results. References 7 and 8 used different features in different road scenes, which can improve detection performance. There are also systems that use the intensity-bump (dark-bright-dark pattern or low-high-low intensity pattern) filter to extract the lane markings.[2,11,18,25] This feature pattern extractor is relatively stable. And in most scenarios, the lane marking itself can maintain this feature. The model-based method is insensitive to noise and partial

occlusions, and it is a strong detection method for a pre-determined model. But in fact we don't have this priori knowledge and the model is uncertain. The most common model is the straight line model via Hough Transform.[10,13,23] It has strong ability to remove noise, but its biggest flaw is that it cannot be used to detect curve lane. To increase the representation ability of road model and expand the scope of application, piecewise linear model,[24] clothoids,[6] quadratic model[5] and spline model[1,21] have been used for fitting the feature points. But all of them should be under the hypothesis of correct clustering. In many articles, the clustering itself is not correct when dealing with the curve lane. If there is a priori information of a map including lane marking and GPS conditions, the model information can be obtained.[22] Using the accurate model can remove the noise very well. If there is no prior knowledge, the model threshold is uncertain. This model is not effective for lane detection with noise. In this paper, the upper and lower structures of line segments can be used to cluster the segments correctly.

In this paper, on the basis of the principle of Zhang's calibration,[26] a small scale inverse perspective map is generated. The scale of the map is one pixel to 5 cm. In the inverse perspective map, the geometric information of lane markings are used to remove disturbances and cluster the lane candidate points into candidate line segments. Then, the line segments are collected for a further processing step. Using the upper and lower structure, segments belonging to the same lane line are clustered, which provides reliable detection of curve lane markings. Finally, in the lane generation phase, the optimal left and right lane markings are extracted by the optimal selection strategy.

This paper is organized as follows. Section 2 introduces the generation method of the inverse perspective map, candidate pixels, candidate line segments, the upper and lower structure and the optimal selection strategy. Section 3 shows the experiment results. Section 4 gives concluding remarks.

## 2. Method

Figure 1 shows our algorithmic processing flow. First, we convert the original image into IPM image, and use the proposed iterative adaptive threshold method to get the candidate pixels. Then, the continuity feature is used to cluster candidate pixels into candidate segments, and the upper and lower structure is to get candidate lane markings. Finally, the optimal strategy is used to get the final lane markings.

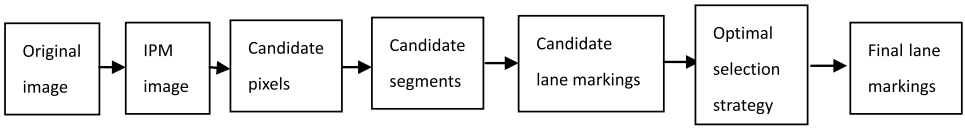



Fig. 1. Our proposed algorithm flow chart.

(a) (b)

Fig. 2. (a) The original image. (b) The IPM image.

### 2.1. *Inverse perspective mapping*

The pinhole camera model performs perspective projection. It transforms the parallel lines on the ground into intersecting lines. In this paper, Zhang's plane calibration method is used to eliminate the influence of perspective transformation and generate IPM image. Figure 2 shows the original image from the camera and the IPM image. The resolution of IPM image is 512*1000. The actual size represented by one pixel is 5 cm.

### 2.2. *Candidate pixels*

The first phase of the algorithm is to get the candidate pixels belonging to lane markings. It is the low-level but very important process. Our wish is to get all pixels of lane markings without noises. However, for the complex application environments, there is currently no algorithm to achieve such results. In order to obtain better detection performance, many algorithms sacrifice some lanes with dim light and more interference, which are just suitable for clean and brighter lane markings. So, the goal of the phase is aimed at extracting as many lane marking pixels as possible while minimizing interference. That is, it will not cause the missed detection of the weak lane markings, and will not cause trouble to the subsequent detection phase. The proposed adaptive threshold method is more adaptable to the complex environment, which is operated on the IPM.

In this step, we first transform the image into a gray level image. Then we select two relative stable features: low-high-low intensity pattern and image contrast. The assumption is that the lane marking has higher brightness than the surrounding ground, which is illustrated in Eqs. (1) and (2). For the outdoor multi-lighting environment, we choose a more stable contrast image rather than gray image to segment. $G$ denotes the contrast image.

$$G(x,y) = \begin{cases} (f_+(x,y) + f_-(x,y))/2, & \text{if } (f(x,y) > \max(f(x,y+T_w), f(x,y-T_w)) \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

where

$$\begin{aligned} f_+(x,y) &= f(x,y) - f(x,y+T_w), \\ f_-(x,y) &= f(x,y) - f(x,y-T_w). \end{aligned} \tag{2}$$

Here, $T_w$ denotes the width of a lane marking. In general, the width is 15 cm. Considering the impact of vehicle turbulence, image discreteness and overexposure in highlight areas, we set $T_w = 4$ pixels. Then we get the image $G$. For any row of the image $G$, we can get multiple consecutive horizontal segments. Find the midpoint of the horizontal line segment and optimize the midpoint value according to Eq. (3). Get enhanced midpoint image $G_{\text{mid}}$.

$$G_{\text{mid}}(x,y) = \begin{cases} \max(G(x,y_{\text{start}}),\ldots,G(x,y_{\text{end}})), & \text{if } y = (y_{\text{start}}+y_{\text{end}})/2 \\ 0, & \text{otherwise} \end{cases}. \tag{3}$$

Here, $y_{\text{start}}$ and $y_{\text{end}}$ represent the starting point and end point of a horizontal line segment respectively. We choose a region of interest (ROI) with a width of 8 m and a height of 15 m masked by red as shown in the second column of Fig. 4, which reduces the impact outside the road and emphasizes the influence of road surface. In such ROI of gray image, image segmentation are performed by Otsu's method. The results of segmentation are shown in the third column of Fig. 4. It can be seen that the performance is not good in the case of night environment and weak lane markings. To optimize the results, we performed the following iterative operations

First, using Otsu's method to segment the image $G_{\text{mid}}$. We can get the initial threshold value $T = T_{\text{first}}$. For any line in the ROI of the image $G_{\text{mid}}$, if there are two points satisfy Eq. (4), the value $R_{\text{num}}$ is increased by one. $\text{dis}(j-\bar{j})$ means the distance between two points. Its value is approximately equal to the width of one lane as illustrated in Fig. 3. Here, the distance ranges from 3.2 m to 4.2 m. Once $R_{\text{num}} \geq 70(3.5\,\text{m})$, $T$ is the final threshold. Otherwise, $T = T - 1$. Iterative cycle operation is performed as illustrated in Function 1. To reduce the time consuming of iterative operation, we use the two-dimensional array to record pixel values as shown in Eq. (5). This operation does not take into account the time consuming caused by iteration. Function 1 illustrates the whole operation process in detail.

$$\begin{cases} G_{\text{mid}}(i,j) \geq T \\ G_{\text{mid}}(i,\bar{j}) \geq T, \quad \bar{j} \in W_j = W_{j1} \cup W_{j2}, \\ 84 \text{ pixels} \geq \text{dis}(j-\bar{j}) \geq 64 \text{ pixels}, \end{cases} \tag{4}$$

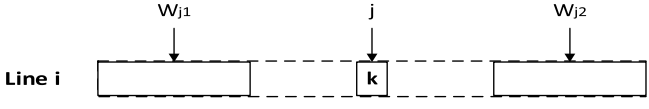



Fig. 3. Determine whether the line satisfies the counting condition. Once finding the two points that satisfy formula (4), count the row and end the search of the row.

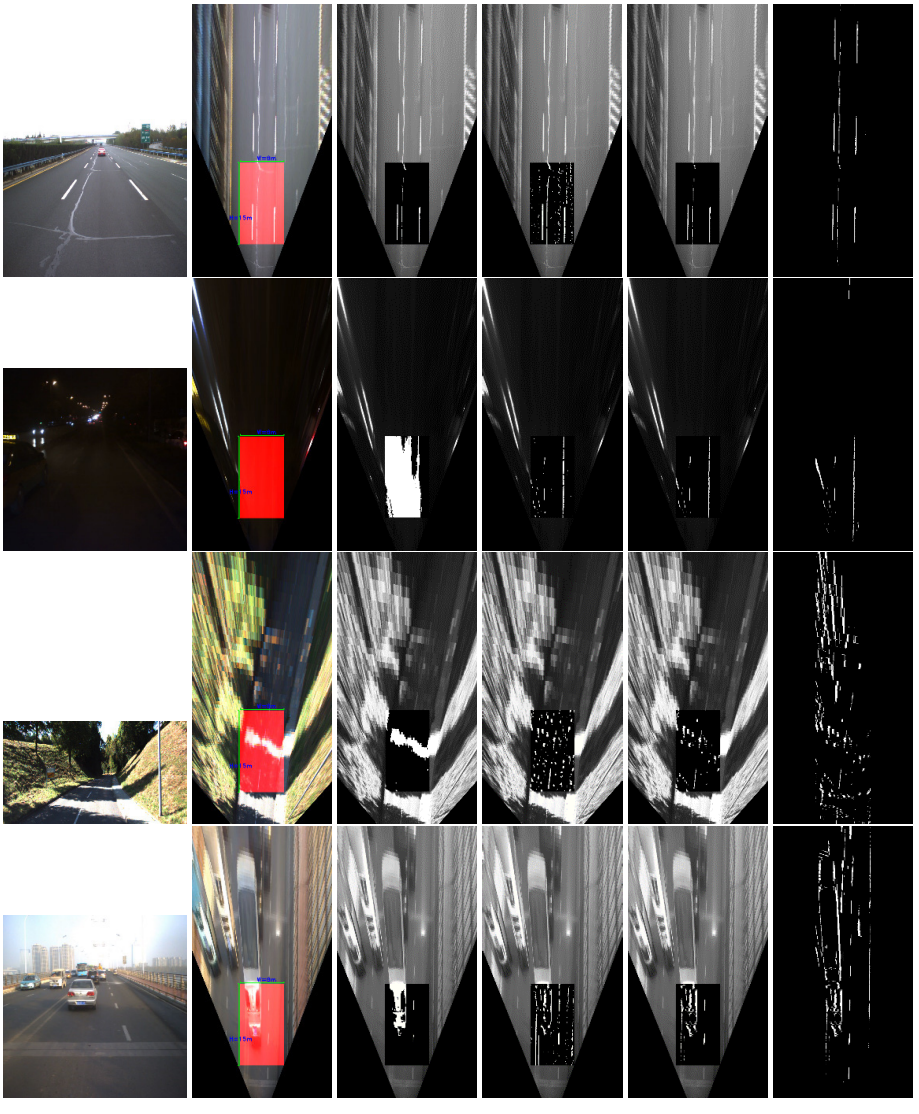


Fig. 4. The segmentation results. The first column shows the original image. The second column, the third column, the fourth column and the fifth row are the IPM with ROI masked by red, the segmentation results using Otsu's method, GOLD method and our proposed method, respectively. The last column are the segmentation results by use of our proposed method, in the ROI with a width of 10 m and a height of 50 m (color online).

$$\begin{cases} R[i][k] = 1, \quad \text{if } G_{\text{mid}}(i,j) == k \&\& G_{\text{mid}}(i,W_j) \geq k. \\ A[i][k] = A[i][k-1] + R[i][k], \\ A[i][0] = R[i][0], \\ (i,j) \in \text{ROI}, \quad k \in [0,255], \end{cases} \tag{5}$$

**Function 1.** Adaptive threshold generation process
**Input:** Two-dimensional array $A[i][256], i \in \text{ROI}$
**Output:** the Adaptive threshold
**Function begin**
**Variable initialization**: $T = T_{\text{first}}, R_{\text{num}} = 0$
**While** $(T \geq 2 \&\& R_{\text{num}} < 70)$
  **for** $(i \in \text{ROI})$
    $R_{\text{num}} = 0$
    **if**$((A[i][255] - A[i][T]) \geq 2)$
      $R_{\text{num}} + +;$
    **end if**
    **if** $(R_{\text{num}} \geq 70)$
      break;
    **end if**
  **end for**
  $T = T - 1;$
**end while**
$T_{\text{final}} = T$
**Function end**

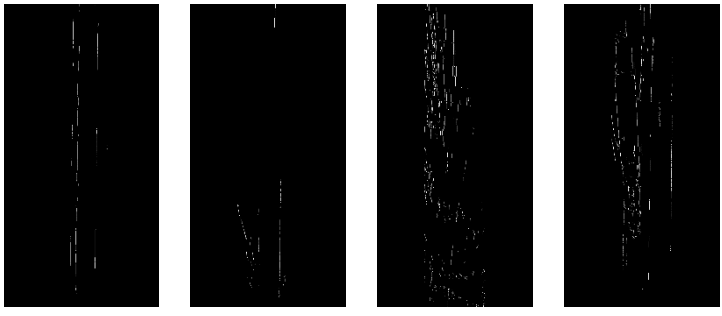
Once the final threshold is obtained, we use it to segment the image in the ROI of lane detection with a width of 10 m and a height of 50 m. Figure 4 shows the segmentation results by using Otsu's method, GOLD method and our adaptive threshold method. It can be seen that our algorithm has strong applicability to light, and can achieve better segmentation effect in various scenarios, especially for weak lanes.

### 2.3. *Candidate segments*

We use the continuity feature of lane marking to cluster pixels into small segments. The continuity feature can remove the sparse noises. But the length of dashed line segment varies with road grade. If there is no prior information about lane length, we will set the length to 2 m according to the minimum dashed segment length in rules. Due to the lane marking are sometimes blocked and worn out, in this paper, we set the length threshold of line segment to 30 (1.5 m).

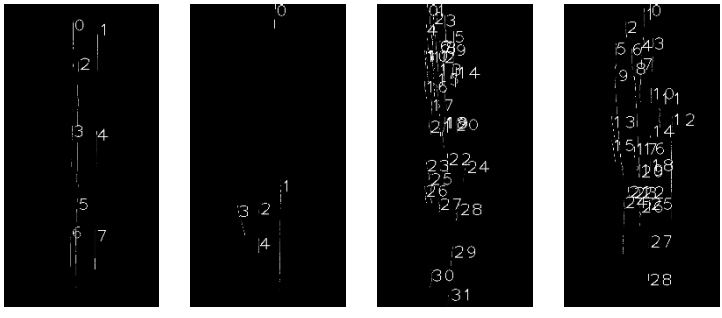
First, we get the midpoint images of the binary images, which are as shown in Fig. 5. Find the midpoint of every horizontal line segment and set the gray value of midpoint to 255. Then we use function 2 to obtain the segments as shown in Fig. 6. This process removes the interference of a lot of miscellaneous points.

From Fig. 6, we can see the candidate pixels are clustered into segments. These segments are sorted according to the line numbers.



(a) (b) (c) (d)

Fig. 5. The four images are the midpoint maps of the bottom images of Fig. 4.



(a) (b) (c) (d)

Fig. 6. The four images are the candidate segments after using continuity feature.

**Function 2.** Candidate segments generation process
**Input:** midpoint image $I_M$
**Output:** line segments image
**Function begin**

1. Scan the midpoint image $I_M$ from left to right and from top to bottom.
2. If $f(x, y) \neq 0$
3. Find the nearest white point in the range of threshold width in the next line.
4. If it was found, perform repeated operations in the next line, and record pixel position. Remove this pixel from the midpoint image. If not, interval++;
5. If interval is greater than the interval threshold, generate a line segment by the stored pixel points. If the length of line segment is less than 30 (1.5 m), discard it. Otherwise it will be retained
6. Continue scan the midpoint image until all pixels are removed

**Function end**

### 2.4. *Upper and lower structure*

Compared with the straight lane markings and the solid lane markings, the detection of the curve and dashed lane markings is a difficult point. In this paper, we use the upper and lower structures, which can cluster the curved dashed segments into one lane marking.

If the two line segments are to be judged as upper and lower structures, the following conditions must be satisfied:

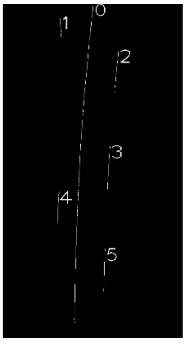
(1) The starting point of a line segment is located below the end of another line segment. And the distance between the two points is within 10 m.
(2) The two lines are fitted by quadratic curve. If the fitting error is less than a certain threshold, the upper-lower relationship is determined. We only keep those fitted lines that are one lane wide from the vehicle's location.

Clustering method based on upper and lower structure is explained in detail in Function 3 and Fig. 7. $L_i$ denotes candidate line segment. $S_i$ stores the serial numbers of the lines that make up the upper and lower relationships. $\notin$ denotes that can't form the upper and lower structure. $\in$ denotes that can form the upper and lower structure.

If the total length of the segments contained in S is too short, S will be discarded. The total length is defined as the difference between the starting abscissa of the first line segment and the ending abscissa of the last line segment. In this paper, we set the length threshold to 6 meters. The remaining lines will be marked in red as shown in Fig. 8. Lines with upper and lower structures are connected by green straight lines.

### 2.5. *Optimal selection strategy*

Although a lot of disturbances are removed by the previous operations, there will still be very similar interference with lane markings.




(a)

| | | | | |
|---|---|---|---|---|
| $L_0$ | Initial value | | new | $S_0 = (0)$ |
| $L_1$ | $\notin S_0$ | | new | $S_1 = (1)$ |
| $L_2$ | $\notin (S_0, S_1)$ | | new | $S_2 = (2)$ |
| $L_3$ | $\notin (S_0, S_1), \in S_2$ | | | $S_2 = (2,3)$ |
| $L_4$ | $\notin (S_0, S_1, S_2)$ | $S_2' = S_2.pop = (2)$ $\notin S_2'$ | new | $S_3 = (4)$ |
| $L_5$ | $\notin (S_0, S_1, S_2', S_3), \in S_2$ | | | $S_2 = (2,3,5)$ |

(b)

Fig. 7. Explain the clustering process with concrete examples.

**Function 3.** Candidate lane markings generation process
**Input:** line segments $\mathrm{L}_i, i = 0, 1, \ldots, n$
**Output:** candidate lane markings $\mathrm{V} = \{S_0, S_1, \ldots\}$
**Function begin**
**Variable initialization**: $S_0 = (0), V = \{S_0\}$, iflag = false;
**for** $(i = 1; i \leq n; i++)$
iflag = false;
**for**$(j = 0; j < V.\text{size}; j++)$
$S_{\text{temp}} = S_j$
**while**$(S_{\text{temp}}.\text{size})$
**if**$(L_i \in S_{\text{temp}})$
**if**$(S_{\text{temp}}.\text{size} = S_j\text{size})$ $S_j = S_{\text{temp}} \bigcup i$; iflag = true; break;
**else New** $S = S_{\text{temp}} \bigcup i$; V.push_back$(S)$; iflag = true; break;
**end else**
**end if**
**else**
$S_{\text{temp}}$pop;
**end else**
**end if**
**end while**
**if** $(j == V.\text{size} - 1 \&\& \text{iflag} == \text{false})$
**New** $S = i$; V.push_back$(S)$;
**end if**
**end for**
**end for**
**Function end**

One of the most important features of our optimal selection strategy is pairing characteristics. In this paper, we use a simple and easy way to match them as illustrated in Eq. (6). Choose two lines from the candidate lane markings.

$$\begin{cases} W_{\min} < L_i < W_{\max}, & i = 1, 2, 3 \\ \text{abs}(L_m - L_n) < T_{\text{sub}}, & m, n = 1, 2, 3,\ m \neq n \end{cases}, \tag{6}$$

where $W_{\min}$ and $W_{\max}$ are the minimum and maximum width threshold of lane markings respectively. $L_i$ denotes the width between two lines in the same line. $L_1$, $L_2$, $L_3$ are the width of upper, middle, lower position, respectively. $T_{\text{sub}}$ denotes the width difference threshold. Their meanings are shown more intuitively in Fig. 9.

After pairing, we use the following strategy to select the best pair of lines:

(1) If there are line pairs, Eq. (7) is used to select the highest score pair as the final line pair.

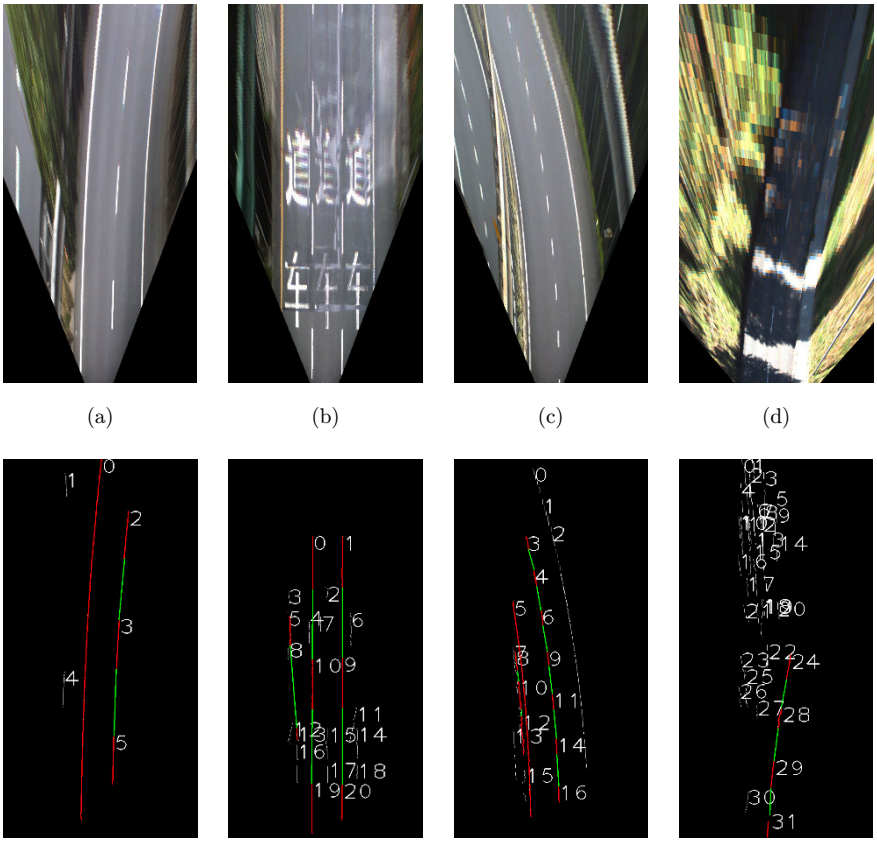


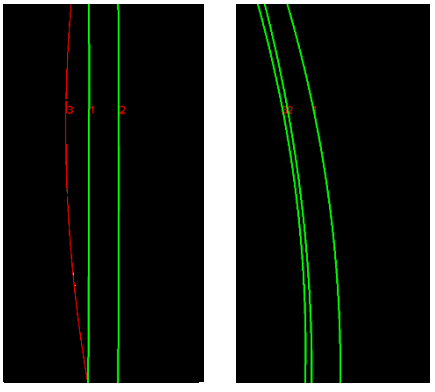
Fig. 8. The first row are the inverse perspective maps. The second row shows the result after clustering by upper and lower structure. The red lines are the candidate lane markings. We use the green lines connect the segments with upper and lower structure (color online).

$$F_{\text{score}} = w_1 * s_1 * B + w_2 * s_2 * L + w_3 * s_3 * \frac{1}{\sigma} + w_4 * s_4 * P, \tag{7}$$

where $B$ is the average gray value of pair lines. $L$ is the average length of pair lines. $\sigma = \max(\text{abs}(L_i - W_{\text{lane}}))$. We set $W_{\text{lane}} = 75$ pixels. $P$ is the parallelism of two lines. $P = \frac{1}{\max(\text{abs}(L_i - L_j))}$, $i, j = 1, 2, 3$. The meanings of $L_i$ is the same as Eq. (6). $s_1, s_2, s_3, s_4$ are the scale factors to normalize $B, L, \frac{1}{\sigma}, P$ to the interval $[0,\ 1]$. $w_1, w_2, w_3, w_4$ are weight values. Taking into account the impact of bumps and road slopes, we are more focused on the gray and length of the lane markings. We set $(w_1, w_2, w_3, w_4) = (0.3, 0.3, 0.2, 0.2)$.

(2) If there is no line pairs, it is very likely that only one lane will appear. We use Eq. (8) to select the best single line.

$$F_{\text{score}} = s_1 * B + s_2 * L. \tag{8}$$



(a) (b)

Fig. 9. The red and green lines are the fitting curves of candidate lane markings from the second column and the third column of Fig. 8. Green represents the success of pairing (color online).

## 3. Experiments

The proposed lane marking detection algorithm is implemented on the KITTI UM dataset and Future Challenge TSD-Lane dataset. For algorithm evaluation, we employ the F-1 derived from the precision and recall values on TSD-Lane dataset. They are illustrated in Eqs. (9)–(11). The visual results are shown in Fig. 10. TP means true positive. FP means false positive. FN means false negative.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{9}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{10}$$

$$F - 1 = \frac{2 * \text{Precison} * \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{11}$$

The KITTI UM dataset only provides ground truth for ego-lane estimates and not for the lane marking themselves. For intuitive comparison, we assess results with indicators as in Ref. 17. Correct Detection (CD): The lane marking estimates were either correctly detected, or were correctly identified as missing or not conforming to the model. Slight Misalignment (SM): The lane marking estimates were slightly misaligned with at least 1 lane marking line segment being missed by the estimates. Major Misalignment or False Alarm (MM): Either of the lane marking estimates miss the majority of true lane marking line segments or incorrectly identify a lane marking where there was not one. False Failure (FF): The algorithm declared a failure to find lane markings that were truly present.

The results of the algorithm are shown in Tables 1 and 2. It can be seen that our algorithm is effective especially for the curve road. This is because we use the upper

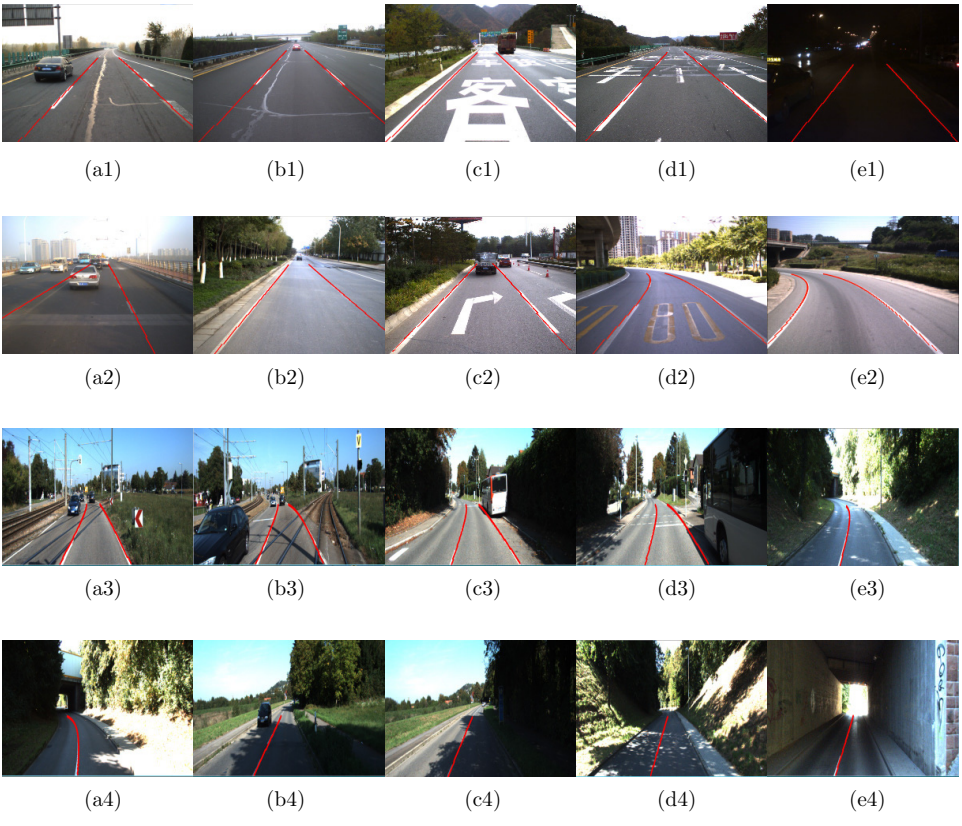



Fig. 10. The results of lane marking detection. The first two rows are the results on the TSD-Lane dataset. The last two rows are the results on KITTI UM dataset.

and lower structure clustering method, which can correctly cluster the pixels belonging to the curve. TSD-Lane 168 and TSD-Lane 183 are the night scene datasets. In the two datasets, the precision is 98.68% and the recall is 93.75%. This is mainly due to the fact that we use the adaptive threshold when using low-light-low feature, which makes our algorithm less sensitive to light changes. The use of strong feature composed of multiple simple features and the use of optimal selection strategy make our algorithm can reduce the effect of disturbances. In the datasets with disturbances, the precision is 99.07% and the recall is 98.40%. But there are some

Table 1. The performance under TSD-Lane dataset.

| Dataset | Type | Frames | Precision (%) | Recall (%) | F-1 (%) |
|---|---|---|---|---|---|
| TSD-Lane-86, 94, 121, 132, 156, 159, 182 | Curve lane | 430 | 99.88 | 99.65 | 99.76 |
| TSD-Lane-80, 108, 172, 177, 179, 180, 181 | Straight lane with disturbances | 220 | 99.07 | 97.73 | 98.40 |
| TSD-Lane-168, 183 | Weak light | 40 | 98.68 | 93.75 | 96.15 |

Table 2. The performance under KITTI UM dataset.

| Method | Dataset (Frames) | CD | SM | MM | FF |
|---|---|---|---|---|---|
| Stereo filtering[17] | KITTI UM road (96 frames) | 50 | 20 | 8 | 18 |
| Proposed method | | 84 | 5 | 4 | 3 |

(a) (b) (c)

Fig. 11. Failed detection results.

limitations in our algorithm. When two lane markings are not parallel as shown in Figs. 11(a) and 11(b), we just detect only one lane marking. This is because we use the characteristics of parallelism and the optimization strategy. Meanwhile, we use the low-high-low pattern. Considering the change in the width of the lanes caused by the vehicle bump or a ramp, the width threshold of the lane marking is increased, but it also causes a miss detection due to the lane marking extension caused by right-reflection as shown in Fig. 11(c).

## 4. Conclusion

This paper supplies a method to detect lane markings under complex urban environment. In this paper, on the basis of the principle of Zhang's calibration, we generate an inverse perspective map. For the IPM, it is easy to use the geometric information of lane markings, such as width, length continuity, interval and so on. First, we propose an iterative adaptive threshold method to generate candidate points. It can not only retain the information of weakened lane, but also not bring much disturbances. Then we use continuous feature to cluster the lane candidate points into candidate line segments. The line segments are collected for a further processing step. Using the upper and lower structure and curvature information of segments, segments belonging to the same lane line are clustered, which provides reliable detection of curve lane markings. Finally, in the lane generation phase, an optimal strategy is used to extract two optimal left and right lane markings.

## Acknowledgments

We would like to acknowledge all of our team members, whose contributions were essential for this paper. We would like to thank the National Key Research and Development Program of China (Nos. 2017YFD0700303, 2016YFD0701401 and

2018YFD0700602), the Key Program of 13th five-year plan, CASHIPS, with granted No. KP-2017-13, the Anhui Provincial Natural Science Foundation with granted No. 1808085QF213, the National Science Foundation of Anhui Province with Grant No. KJ2016A233, the Youth Innovation Promotion Association of the Chinese Academy of Sciences (Grant No. 2017488), the Research Program of Fire Bureau of Ministry of Public Security (Grant No. 2017XFGG04) and the Technological Innovation Project for New Energy and Intelligent Networked Automobile Industry of Anhui Province.

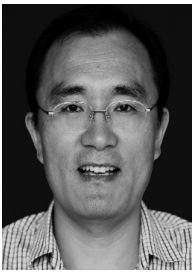
## References

1. M. Aly, Real time detection of lane markers in urban streets, *Intelligent Vehicles Symp.* (IEEE, 2008), pp. 7–12.
2. M. Bertozzi and A. Broggi, GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection, *IEEE Trans. Image Process.* **7**(1) (1998) 62–81.
3. A. Borkar, M. Hayes and M. T. Smith, Lane detection and tracking using a layered approach advanced concepts for intelligent vision systems, *Int. Conf.,* September 28-October, Acivs, Bordeaux, France, 2009.
4. W. Chao, W. Huan, Z. Chunxia *et al.*, Lane detection based on gradient-enhancing and inverse perspective mapping validation, *J. Harbin Eng. Univ.* **35**(9) (2014) 1156–1163.
5. C. D'Cruz and J. J. Zou, Lane detection for driver assistance and intelligent vehicle applications, *Int. Symp. Communications and Information Technologies* (IEEE, 2007), pp. 1291–1296.
6. C. Gackstatter, P. Heinemann, S. Thomas and G. Klinker, Stable road lane model based on clothoids, in *Advanced Microsystems for Automotive Applications*, eds. G. Meyer and J. Valldorf (VDI-Buch, Springer, Berlin, Heidelberg, 2010), pp. 133–143.
7. J. Huang, H. Liang, Z. Wang and M. Tao, Robust lane marking detection under different road conditions, *IEEE Int. Conf. Robotics and Biomimetics* (IEEE, 2013), pp. 1753–1758.
8. J. Huang, H. Liang, Z. Wang, Y. Song and Y. Deng, Lane marking detection based on adaptive threshold segmentation and road classification, *IEEE Int. Conf. Robotics and Biomimetics* (IEEE, 2015), pp. 291–296.
9. H. Y. Ju, S. W. Lee, S. K. Park and H. K. Dong, A robust lane detection method based on vanishing point estimation using the relevance of line segments, *IEEE Trans. Intell. Transp. Syst.* **99** (2017) 1–13.
10. J. G. Kuk, J. H. An, H. Ki and N. Cho, Fast lane detection & tracking based on Hough transform with reduced memory requirement, *Int. IEEE Conf. Intelligent Transportation Systems* (IEEE, 2010), pp. 1344–1349.
11. C. Lee and J. H. Moon, Robust lane detection and tracking for real-time applications, *IEEE Trans. Intell. Transp. Syst.* **19**(12) (2018) 4043–4048.
12. H. Li and F. Nashashibi, Robust real-time lane detection based on lane mark segment features and general a priori knowledge, *IEEE Int. Conf. Robotics and Biomimetics* (IEEE, 2012), pp. 812–817.
13. Z. Q. Li, H. M. Ma and Z. Y. Liu, Road lane detection with gabor filters, *Int. Conf. Information System and Artificial Intelligence* (IEEE, 2017), pp. 436–440.
14. J. Niu, J. Lu, M. Xu, P. Lv and X. Zhao, Robust lane detection using two-stage feature extraction with curve fitting, *Pattern Recognit.* **59** (2015) 225–233.

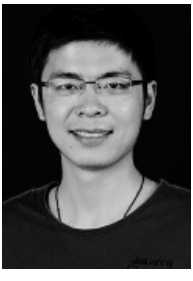
15. U. Ozgunalp and N. Dahnoun, Lane detection based on improved feature map and efficient region of interest extraction, *Signal and Information Processing* (IEEE, 2016), pp. 923–927.
16. U. Ozgunalp, R. Fan and X. Ai and N. Dahnoun, Multiple lane detection algorithm based on novel dense vanishing point estimation, *IEEE Trans. Intell. Transp. Syst.* **18**(3) (2017) 621–632.
17. M. Smart and S. L. Waslander, Stereo augmented detection of lane marking boundaries, *IEEE Int. Conf. Intelligent Transportation Systems* (IEEE, 2015), pp. 2491–2496.
18. U. Suddamalla, S. Kundu, S. Farkade and A. Das, A novel algorithm of lane detection addressing varied scenarios of curved and dashed lanemarks, *Int. Conf. Image Processing Theory, TOOLS and Applications* (IEEE, 2016), 87–92.
19. Y. Su, Y. Zhang, T. Lu, J. Yang and H. Kong, Vanishing point constrained lane detection with a stereo camera, *IEEE Trans. Intell. Transp. Syst.* (2017) 1–6.
20. G. Tejus, S. S. Harshit and C. Debashish, Robust lane detection using multiple features, *IEEE Intelligent Vehicles Symp.* (IEEE, 2018), pp. 1470–1475.
21. Y. Wang, E. K. Teoh and D. Shen, Lane detection and tracking using B-Snake, *Image Vision Comput.* **22**(4) (2004) 269–280.
22. Y. Wu, N. Zhang, T. Zhou and W. Yan, Research of lane detection and tracking methods based on multi-sensor fusion, *Appl. Res. Comput.* **35**(2) (2018) 600–607.
23. J. Xiao, S. Li and B. Sun, A real-time system for lane detection based on FPGA and DSP, *Sens. Imag.* **17**(1) (2016) 1–13.
24. W. Xiaojin, W. Zengcai and Z. Lei, Lane detection combining piecewise linear model with vanish point detection, *Mechatron.* **5** (2017) 41–46.
25. H. Yoo, U. Yang and K. Sohn, Gradient-enhancing conversion for illumination-robust lane detection, *IEEE Trans. Intell. Transp. Syst.* **14**(3) (2013) 1083–1094.
26. Z. Zhang, A flexible new technique for camera calibration, *IEEE Trans. Pattern Anal. Mach. Intell.* **22** (2000) 1330–1334.

**Junjie Huang** is currently working toward the Ph.D. degree in the Department of Automation, University of Science and Technology of China, Hefei, China, and Institute of Applied Technology, Hefei Institutes of Physical Science Chinese Academy of Science. Her research interests are computer vision, intelligent vehicle perception system.

**Zhiling Wang**, Doctor, Researcher, Master Tutor, Executive Deputy Director of Intelligent Vehicle Technology Research Center, Institute of Applied Technology, Hefei Institutes of Physical Science Chinese Academy of Science. He is mainly engaged in the research of driverless vehicles, environmental perception and understanding, machine vision and machine learning.



**Huawei Liang**, Researcher, Doctoral Supervisor, Deputy Director, Institute of Applied Technology, Hefei Institutes of Physical Science Chinese Academy of Science. He has been engaged in the robotics, intelligent vehicle technology and systems, detection technology and automation device, pattern recognition and intelligent system, control theory and control engineering.



**Linglong Lin**, Doctor, Associate Researcher. He received the Ph.D. degree from the University of Chinese Academy of Sciences, Hefei, China, in 2016. He is mainly engaged in the research of driverless vehicles, environmental perception and understanding.

**Biao Yu** is an Associate Professor at the Institute of Applied Technology, Hefei Institutes of Physical Science, Chinese Academy of Sciences, China. He received his Ph.D. and M.Sc. degrees from School of Mechanical and Automotive Engineering, Hefei University of Technology in 2013 and 2010, respectively. His research interests include evolutionary computation, mobile robot navigation and localization.

**Fei Dong** is a researcher affiliated with Shanghai Fire Research Institute of MEM. She received a Master's degree in Mechanical engineering. Her research interests include development of fire-fighting robotics and fire extinguishing technology.

**Yan Xu** is a researcher affiliated with Shanghai Fire Research Institute of MEM. He holds a Bachelor's degree in computer network technology. His research interests include development of fire-fighting robotics and fire extinguishing equipment technology.