Proceedings of the IEEE ITSC 2006
2006 IEEE Intelligent Transportation Systems Conference
Toronto, Canada, September 17-20, 2006

Real-time Lane Detection in Various Conditions and Night Cases

Hong Wang, Qiang Chen
State Key Laboratory of Intelligence Technology & Systems
Department of Computer Science & Technology, Tsinghua University, Beijing 100084, P.R.China
wanghong@tsinghua.edu.cn, cq00@mails.tsinghua.edu.cn

Abstract— **In this paper, we propose a real-time lane detection algorithm based on a hyperbola-pair lane boundary model and an improved RANSAC paradigm. In stead of modeling each road boundary separately, we propose a model to describe the road boundary as a pair of parallel hyperbolas on the ground plane. A fuzzy measurement is introduced into the RANSAC paradigm to improve the accuracy and robustness of fitting the points on the boundaries into the model. Our method is able to deal with existence of partial occlusion, other traffic participants and markings, etc. Experiment in many different conditions, including various conditions of illumination, weather and road, demonstrates its high performance and accuracy.**

I. INTRODUCTION

Lane detection is a critical component in Driver Assistance System in order to provide meaningful and consistent road shape information for navigation purpose. In the real road land detection problem, environments are usually affected by many factors and complicated. There are straight and curvilinear lanes, and many other traffic markings except for the lane markings exist. Moreover, the lane boundary might be occupied by other traffic participants, blurred and broken into several parts, etc. Fig. 1 shows some samples, including straight/curvilinear lanes, existing crosswalk/arrows, vehicle occupation and environment at night.

Instead of considering each road boundary separately like most of existing methods, we propose to use a pair of road boundaries in our road shape model. In order to fit the hyperbola-pair road model, we introduce a real-time algorithm to extract edge points on lane boundaries. Given the boundaries observation and the road model, we adopt the RANSAC method [1] , as well as an improvement of fuzzy measurement, to estimate the model parameters, which demonstrates the capacity of our hyperbola-pair road model and improves the robustness of the algorithm.

II. RELATED WORK

Different lane detection algorithms have been applied in several experimental systems, which have been reported since the 1990s. CMU developed several systems on their NavLab experimental platform series, such as UNSCARF [2] in 1991, SCARF [3] in 1993 and RALPH [4] in 1995. Then, more and more systems were reported to be developed for detecting lanes, of which some typical ones are GOLD [5] in 1998 by Parma University; LOIS [6] in 1998 by Michigan University. Nowadays, detection of lanes becomes an essential part of many such systems. In 2003, THMR-5 developed by Tsinghua University [7] was reported to have this ability.

Fig. 1. Some samples of different road environments.

Edge information was mostly used in such algorithms [8] [9] [10] [11] [12] . Most of them used different gradient operators to extract edge information of lane markings, including the magnitude and the orientation. A large number of different strategies and methods [13] [14] [15] [16] were then used to find out the edge of the lane boundaries. LANA algorithm [17] was based on a novel set of frequency domain features that capture relevant information concerning the magnitude and orientation of spatial edges, which can also be considered as another representation of edge.

Some other algorithms based on pixel clustering and area segmentation [2] [3] [18] [19] were reported, which mainly aimed to distinguish the road and non-road area. As the pixel clustering operation is time-consuming, the image to be processed cannot be large, so that the accuracy of the algorithms were not so high.

Deformable templates have been widely used in many algorithms [20] . They attempt to determine mathematical models to fit the lane boundaries. The choice of the models is of great importance, since a simple model may be robust and easily calculated, but may not provide an accurate result, and a complex model may be flexible and suitable for various shapes of road, but may be affected easily by disturbance. The hyperbola-pair model proposed in this paper is also a deformable template, and is developed on the base of Kluge's work [14]. The origin model is suitable for single-side lane markings firstly, but in the work by Yu and Jain [16], it was emphasized that some parameters of the model was the same, if the lanes were parallel on the same road. The considerable feature is also used in our algorithms, to form an extended equation to fit the road shape.

Using inverse projection mapping (IPM) is another way of considering the problem [21], by carrying out the detection on top-view images. In that case, characteristics of the road can be used and the results are much more convenient for vehicle control. As the top-view images are generated during the process, an accurate calibration of the camera (that is, a stable running environment for the system), as well as a powerful calculation ability is needed.

Some other techniques are reported, too. A robust algorithm, based on particle filtering and multiple cues was reported to have great performance under different situations by N. Apostoloff and A. Zelinsky [22]. And the edge distribution function (EDF) method was supposed by Lee [23] in 2002, which estimated the lane orientation by analyzing the direction of edges in the image. It was suitable for straight lines, but might encounter failure with a curving lane, and was easily disturbed by noise.

In this paper, we propose a hyperbola-pair model to describe lanes. It can accurately fit lanes with various typical shapes. Moreover, the parameters have definite relationship with the calibration data and the characteristics of the lane (i.e. the curvature), which makes the calculation of the vehicle's location and orientation much easier. In our algorithm, an explicit edge detection is carried out, so it is also based on the edge information.

III. Overview of Our Algorithm

A CCD camera is fixed on the rear-view mirror to view the road. To simplify the problem, we assume that it is setup to make the baseline horizonal, which assures the horizon in the image is parallel to the x-axis. Otherwise, we can adjust the image using the calibration data of the camera to make it.

Each lane boundary marking, usually a rectangle (or approximate), forms a pair of edge lines. In our algorithm, the "inner" edges, which is closer to the vehicle (observer), is used to describe the lane boundaries.

By scanning the image, points on the inner edges of the markings forming the two boundaries are retrieved. Then, some fitting method, which is based on the RANSAC paradigm to get high accuracy and robustness, is used to get the parameters of the hyperbola-pair model. Also, some tracking technique is applied to reduce the effect of disturbance and errors.

After the lane boundaries are detected, the position and orientation of the vehicle, as well as the curvature of the road can be easily calculated, with the calibration data of the camera.

IV. The Hyperbola-pair Lane Boundary Model

We adopt the general road model proposed in [14], which assume that each lane boundary forms a shape of hyperbola in the image plane, which can be expressed by the following formula [14], given the road boundary point (u, v) in image plane.

$$u = \frac{k}{v-h} + b(v-h) + c \tag{1}$$

In this formula, h is the y-coordinate of the horizon; k, b and c are the parameters of the curve, which can be calculated from the shape of the lane on the ground and the calibration data of the camera. The shape is shown in Fig. 2.

Due to occlusion of road boundaries, one road lane might be invisible. Direct estimation of road parameters usually leads to erroneous road shape. Based on the observation that most road lanes,or boundaries are parallel and it is not likely to occlude both road boundaries, we propose to use hyperbola-pair road model.

Our hyperbola-pair road model is built on the property of the hyperbola road model proposed in [14] that the road model of parallel parabolas in ground plane has the same parameters of k and c. We provide a proof as follows:

According to [14], a road lane boundary of curvature k' is approximated by a parabola of the form

$$x = k'y^2 + b'y + c' \tag{2}$$

and a pair of parallel boundaries have the same k' and b'.

In the case where the camera is fixed straight forward and tilted downwards, assuming perspective projection, the pixel (u, v) in the image maps to the point (x, y) on the ground , the projection can be formulated as

$$\begin{aligned}(v-v_0)v_f &= \tan(\arctan(\frac{H}{y}) - \arctan(v_0 v_f)) &(3)\\ (u-u_0)u_f &= \frac{x}{\sqrt{y^2+H^2}\cos(\arctan((v-v_0)v_h))} &(4)\end{aligned}$$

where $v = 0$ is the position of the horizon, H is the height of focal center, (u_0, v_0) is the center pixel of the image (the optical axis), u_f, v_f is the pixel size divided by focal length, . The curve is then formulated as

$$\begin{aligned}u &= \frac{1}{u_f v_f H\sqrt{1+u_f^2+v_f^2}}(k'H^2(1+v_0^2v_f^2)\frac{1}{v}\\ &+ v_f^2(c' - b'Hv_0v_f + k'H^2v_0^2v_f^2)v \\ &+ Hv_f(1+v_0^2v_f^2)(b' - k'Hv_0v_f)) + u_0\end{aligned} \tag{5}$$

Comparing to (1), we can see that

$$
\begin{aligned}
k &= \frac{k'H(1+v_0^2v_f^2)}{u_f v_f\sqrt{1+u_f^2+v_f^2}} \\
b &= \frac{v_f(c'-b'Hv_0v_f+k'H^2v_0^2v_f^2)}{u_f H\sqrt{1+u_f^2+v_f^2}} \\
c &= \frac{(1+v_0^2v_f^2)(b'-k'Hv_0v_f)}{u_f\sqrt{1+u_f^2+v_f^2}} + u_0
\end{aligned} \tag{6}
$$

for parallel boundaries on the road, as k' and b' are the same, the parameters k and c will be the same too, and b will be determined by the parameter c' of the boundary.

In our algorithm, the two line formed by lane markings are grouped into one model. Assume that the two lane marking curves can be represented by

$$u = \frac{k}{v-h} + b^{(l)}(v-h) + c \tag{7}$$

$$u = \frac{k}{v-h} + b^{(r)}(v-h) + c \tag{8}$$

Then the model can be formulated by an extended equation:

$$u = \frac{k}{v-h} + b^{(l)}(v-h) + b^{(r)}(v-h) + c \tag{9}$$

For each point on the boundaries, $b^{(r)} = 0$ if it is on the left curve and $b^{(l)} = 0$ if on the right one. As long as two sets of points representing the boundaries of the lane are extracted, the model can be easily fitted by several methods.

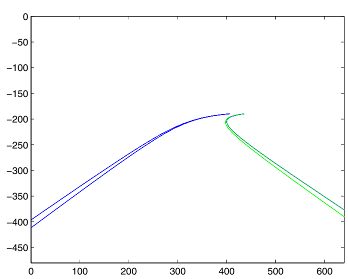

Fig. 2. Simulated lane model shape

By combining the two boundaries to a pair, more observations are involved into one model, but the number of parameters increases only from 3 to 4 and thus the accuracy and robustness is improved, especially when part of the boundaries are missing (i.e. uncleared markings in the far field) or occluded by other objects.

By scanning the inner edges of the lane markings, we collect two sets of points, each forms the boundary on one side of the lane. The points are organized into two lists expressed as $L^{(l)}$ and $L^{(r)}$ and

$$
\begin{aligned}
L^{(l)} &= \{(u_1^{(l)}, v_1^{(l)}), (u_2^{(l)}, v_2^{(l)}), \cdots, (u_m^{(l)}, v_m^{(l)})\} && (10)\\
L^{(r)} &= \{(u_1^{(r)}, v_1^{(r)}), (u_2^{(r)}, v_2^{(r)}), \cdots, (u_n^{(r)}, v_n^{(r)})\} && (11)
\end{aligned}
$$

In the ideal condition, all points are on the boundaries and no outliers is involved. From (1) (9)and its characteristics, let

$$
A = \begin{bmatrix}
\frac{1}{v_1^{(l)}-h} & v_1^{(l)}-h & 0 & 1 \\
\vdots & \vdots & \vdots & \vdots \\
\frac{1}{v_m^{(l)}-h} & v_m^{(l)}-h & 0 & 1 \\
\frac{1}{v_1^{(r)}-h} & 0 & v_1^{(r)}-h & 1 \\
\vdots & \vdots & \vdots & \vdots \\
\frac{1}{v_n^{(r)}-h} & 0 & v_n^{(r)}-h & 1
\end{bmatrix} \tag{12}
$$

$$\overline{x} = [\ k \quad b^{(l)} \quad b^{(r)} \quad c]^T; \tag{13}$$

$$b = [\ u_1^{(l)} \quad \cdots \quad u_m^{(l)} \quad u_1^{(r)} \quad \cdots \quad u_n^{(r)}\ ] \tag{14}$$

Obviously, the parameters of the model can be calculated by solving the linear equation $A\overline{x} = b$. Because $m + n \gg 4$, a least-square solution is solved.

Unfortunately, points that are not on the boundaries are usually involved in the lists. They may be generated by shadows on the road, other traffic markings, other vehicle in the front, etc. So the lease-square solution is usually not reliable and it ruins the result fitted model. So we apply another method of fitting the model based on the RANSAC paradigm instead.

V. The RANSAC Paradigm and Our Improvement

In the Least-Square Method, the total least-squares criterion places huge weight on large errors. So if one data points lay a long way from the a model that fits others well, the result fitted model will be biased heavily by it. That will reduce the robustness and accuracy greatly. The RANSAC paradigm is suitable for this kind of cases. It is described as follows:

Determine

n – the smallest number of points required
k – the number of iterations required
t – the threshold used to identify a point that fits well
d – the number of nearby points required to assert a model fits well

Until k iterations have occurred

1) Select n points uniformly from the data at random to form a sample.
2) From the n points, fit the model.
3) For each data point outside the sample
   a) Calculate the distance from the point to the model.
   b) If the distance is less than t, the point is close.
4) If there are more than d points close to the model, then there is a good fit. Refit the model using all the close points.

end

Use the best fit from the collection.

For example, in our algorithm the step of estimating the location of the horizon, two lines need to be retrieved from two lists of points. A simple RANSAC procedure is carried out on each list to get a good fit of the line.

According to 9, at least 4 points are needed to get a result model. However, not every sample of 4 arbitrary points can form a fit (i.e. 4 points on the same side). It must meet the requirement that $|A| \neq 0$.

As the image is digitally sampled, and the positions of points are represented by integers. Moreover, the shape of the land boundary is not an exact hyperbola. So almost all the points are at a distance to the calculated model more or less. It is difficult to find a threshold used to identify a point that fit well, because a large threshold may lead to a bad fit which has lots of close points but is far away from the actual boundary, and a small threshold may cause that no fit can get enough close points and the algorithm fails to get a result.

Actually, the original RANSAC paradigm uses a 0-1 logic in identify a point that fits well. And as an improvement to avoid the difficulty in determining the threshold, a fuzzy logic is used in our algorithm instead. In the test step of the RANSAC algorithm, we introduce a fuzzy criterion to measure whether a point is ”close” to the model.

We introduce a function $\rho(p; M)$ to describe how close is a point p to the model M fitted from the selected sample. Let $r(p; M)$ the distance between p and M. We define

$$\rho(p; M) = \begin{cases} 1 & r(p; M) \leq T_{min} \\ \dfrac{1}{8r(p; M) - 3} & T_{min} < r(p; M) \leq T_{max} \\ 0 & r(p; M) > T_{max} \end{cases} \quad (15)$$

While d is predefined, the model is a good fit if $\sum_{p \in L} \rho(p; M) \geq d$ and the best fit is the one with the largest $\sum_{p \in L} \rho(p; M)$.

By experiment, we choose $r(p) = |\dfrac{k}{p.y - h} + b(p.y - h) + c - p.x|$, $T_{min} = 0.5$, $T_{max} = 2$, and $d = 0.6$.

VI. The Lane Detection Algorithm

A. Estimation of The Position of the Horizon

Without any prior knowledge of the camera setting, the y-coordinate of the horizon must also be detected from the image. Affected by the projection transform, a pair of parallel line on the ground will be mapped to a pair of line intersecting on the horizon. This feature is used in our algorithm, by detecting the nearest pair of the lane marking segments. They are approximately parallel and straight, so there extend line will intersect at the vanishing point, on the horizon.

Firstly, the canny operator is applied to form an edge image. The thresholds can be adjusted to adapt to the different road conditions.

A middle-to-side strategy, proposed in [7], is used in the search of the pair of the markings. On each horizontal scan-line, the search begins from the middle of the image to the left or to the right and finds out the first non-zero (edge) point on each side. The search steps run from the bottom of the image upwards. After a pair of boundary points are selected, the begin position of the next scan-line is update to the middle of the points. It works when the lane bends and the middle vertical line of the image intersects with one boundary. However, if the middle position of the bottom scan-line is not between a pair of boundary, the algorithm fails. We solve the problem by the tracking technology, which will be demonstrated later. Points on each side are stored in a list.

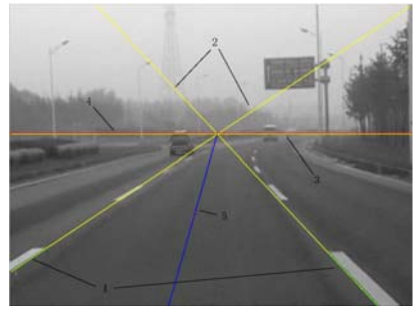

Fig. 3. The location of horizon. 1: The edge points on the first pair of lane markings (green points). 2: The extended line (yellow). 3: The currently detected horizon (yellow). 4: The tracked horizon (red). 5: The middle line of the lane (blue).

In the list, the distance between adjacent points must no greater than a threshold to form a continuous line. Moreover, the length of the list must be in a range, neither too short to be easily disturbed by noises nor too long to form a curve segment in some specific conditions (i.e. a solid curvilinear lane marking).

Finally, after the two lists of points are collected, two straight lines are located by linear fitting. A simple RANSAC implementation is applied to get the two lines.

From the pair of lines, some important information is retrieved for the further stages, which are used for guiding the search of boundaries and eliminating noise, including the y-coordinate position h of the horizon; the approximate width of the lane w_0, represented by the distance between the two lines on the bottom of the image; the middle position of the lane on the bottom of the image p_0.

Actually, these parameters give out a rough description of the lane region.

B. Fit the Hyperbola-pair Model

1) Collect the points on the boundaries: In the algorithm, the inner edges of the lane markings indicating the current lane is of interest. The model will be fitted using the points on these edges. These points are collected using the same middle-to-side strategy, but with the help of the last stage, some more restrictions are applied to make the search fast and accurate.

It can be easily proved that in the image the width of the lane in the image decreases approximately linearly while the scanned row moves upwards, since the formulated boundary curves are different only in the b parameters. In another word, the width is a linear function of $y - h$. This value of width on each row can be predicted from the h and w_0. Only the pair of edge points with a distance of threshold, or the points with a distance of approximate half the value to the middle of the lane, are accepted. This restriction reduces the error

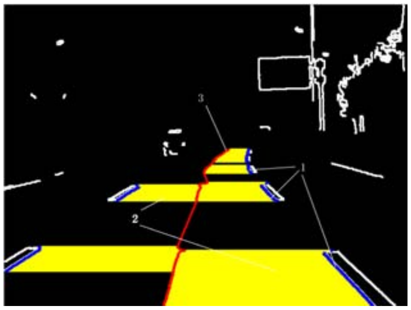

Fig. 4. The search for the inner edge points of the lane markings. 1: The points collected for model fitting (blue). 2: The scanned area from the middle position to both sides (yellow). 3: The beginning position of every searched row (red).

acceptance of fault edge points and improves the accuracy of the detection greatly. Two lists, consisting of the points on the inner edges of the lane markings separately, are formed after this stage.

Let the width and height of the image are respectively W and H. The steps are demonstrated as follows:

$p \leftarrow p_0$, $y \leftarrow H$, $list_L \leftarrow \emptyset$, $list_R \leftarrow \emptyset$
while $y > h$ **do**
  $x_L, x_R \leftarrow p$; $b_L, b_R \leftarrow false$; $w \leftarrow \frac{y-h}{H-h} w_0$,
  repeat
    $x_L \leftarrow x_L - 1$
  until $image(x_L, y) \neq 0$ or $x_L < 0$
  repeat
    $x_R \leftarrow x_R + 1$
  until $image(x_R, y) \neq 0$ or $x_R \geq W$
  if $x_L \geq 0$ **then**
    $list_L \leftarrow list_L \cup \{(x_L, y)\}$, $b_L \leftarrow true$
  end if
  if $x_R \geq 0$ **then**
    $list_R \leftarrow list_R \cup \{(x_R, y)\}$, $b_R \leftarrow true$
  end if
  if b_L and b_R **then**
    $p \leftarrow \frac{x_L + x_R}{2}$
  else if b_L and not b_R **then**
    $p \leftarrow x_L + \frac{w_0}{2}$
  else if not b_L and b_R **then**
    $p \leftarrow x_R - \frac{w_0}{2}$
  end if
end while

Fig. 4 illustrates the process.

Since the parameter k controls the curvature of the lane, it should varies smoothly, but it is easily disturbed by noises and sometimes causes errors. So a simple tracking method is also applied to avoid this.

VII. Experiment Result

A database including a growing number of more than 50,000 frames is set up for the experiment. All the images are taken on the urban roads and highways in Beijing City, China. The database covers many conditions, including sunny and cloudy, daytime, dusk and nighttime , solid and dashed markings, straight and curvilinear roads, etc. Some difficult conditions, such as partly occupied markings, other traffic markings on the road, and so on.

TABLE I
DETECTION RATE AGAINST THE RATE OF OUTLIERS (WITHOUT TRACKING)

Rate of outliers	Frames (Rate)	Detected (Rate)
$> 40\%$	0 (0.00%)	372 (5.25%)
$30\% \sim 40\%$	121 (1.71%)	9 (7.44%)
$20\% \sim 30\%$	521 (7.35%)	331 (63.53%)
$10\% \sim 20\%$	1757 (24.77%)	1702 (96.87%)
$0\% \sim 10\%$	4284 (60.41%)	4281 (99.93%)
0	37 (0.52%)	37 (100.00%)
Total	7092 (100.00%)	6360 (89.68%)

The results of our test on the database indicate the robustness and real-time performance of the algorithm. The test was carried out on a PC with a Intel Pentium IV CPU of 3.0 GHz. The size of the image is 640×480. In the daytime conditions, the average time is 8.32 ms/frame and in the nighttime 7.62 ms per frame. The maximum values are respectively 13.50 ms and 10.96 ms. The model fits well and some typical result is shown in Fig.5.

The result also demonstrate the robustness of the RANSAC paradigm. We tested 6 typical video clips, including conditions in the daytime and night, to test the detection rate against the rate of outliers. The result indicates that the rate of outliers is less than 20% in most times (85.7% of frames), and in these frames the detection rate is higher than 95%. The qualitative result is shown in Table I.

VIII. Conclusion

In this paper, we propose a real-time algorithm to detect road lanes. The use of hyperbola-pair road model can deal with the occlusion and imperfect road condition. With the improvement of fuzzy logic, the improved RANSAC paradigm greatly increases the robustness and performance of the algorithm. The results of our experiment indicates that the algorithm meets the requirement of time performance and accuracy. Our lane detection system provides information for locating and orienting the vehicle, and is suitable for lane departure warning systems.

Acknowledgment

The authors would like to express their gratitude to Yu Qian and Cai Wenchao for their outstanding suggestions, to Zhu Wennan and all the other members involved in the project for their enthusiastic work.

References

[1] M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, pp. 381 – 395, June 1981.

[2] Jill.D.Crisman, “UNSCARF, a color vision system for the detection of unstructured roads,” in *Proceedings of IEEE International Conference on Robotics and Automation*, Sacnunento, California, Apr 1991, pp. 2496–2501.

[3] ——, “SCARF: A color vision system that tracks roads and intersections,” *IEEE Transactions on Robotics and automation*, vol. 9, no. 1, Feb 1993.

[4] D. Pomerleau, “RALPH: rapidly adapting lateral position handler,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, Jun 25–26 1995, pp. 506–511.

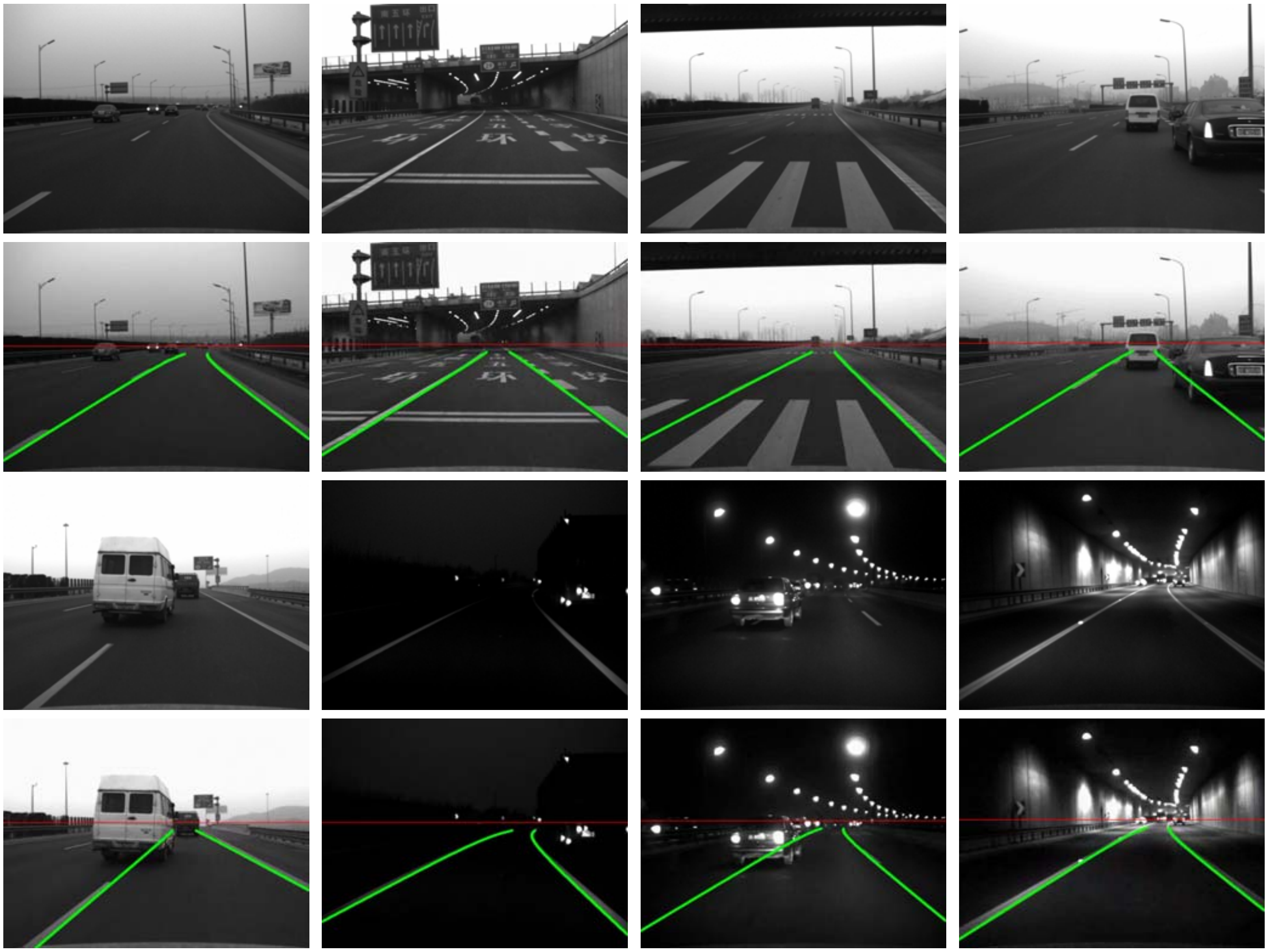

Fig. 5. Algorithm result in different conditions

[5] M. Bertozzi, “GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection,” *IEEE Transactions on Image Processing*, vol. 7, no. 1, pp. 62–81, Jan. 1998.

[6] K. Kluge and S. Lakshmanan, “A deformable-template approach to lane detection,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, Jun 25–26 1995, pp. 54–59.

[7] K. Ni and K. He, “THMR-V: An effective and robust high-speed system in structured road,” in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Oct 5–8 2003, pp. 4370–4374.

[8] A. Broggi, “Robust real-time lane and road detection in critical shadow conditions,” in *Proceedings of IEEE International Symposium on Computer Vision*, Nov 1995, pp. 353–359.

[9] A. Broggi and S. Bertè, “Vision-based road detection in automotive systems: A real-time expectation-driven approach,” *Journal of Artificial Intelligence Research*, pp. 325–348, 1995.

[10] F. Paetzold, U. Franke, and W. V. Seelen, “Lane recognition in urban environment using optimal control theory,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, Oct 2000, pp. 221–226.

[11] B. Southall and C. J. Taylor, “Stochastic road shape estimation,” in *Proceedings of IEEE International Conference on Computer Vision*, Vancouver, BC, Canada, July 2001.

[12] M. Kazui, M. Haseyama, and H. Kitajima, “The estimation of the vanishing point for automatic driving systems using a cross ratio,” *Systems and Computers in Japan*, pp. 31–40, 2001.

[13] K. Kluge and C. Thorpe, “Representation and recovery of road geometry in YARF,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, Jun – July 1992, pp. 114–129.

[14] K. Kluge, “Extracting road curvature and orientation from image edge points without perceptual grouping into features,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, 1994, pp. 109–114.

[15] A. Gern, R. Moebus, and U. Franke, “Vision-based lane recognition under adverse weather conditions using optical flow,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, 17-21 Jun 2002, pp. 652–657.

[16] B. Yu and A. Jain, “Lane boundary detection using a multiresolution hough transform,” in *Proceedings of International Conference on Image Processing*, 26-29 Oct 1997, pp. 748–751.

[17] C. Kreucher and S. Lakshmanan, “LANA: a lane extraction algorithm that uses frequency domain features,” *IEEE Transactions on Robotics and automation*, pp. 343–350, Apr 1999.

[18] A. K. Dawoud, S. G. Foda, and A. S. Tolba, “A robust neural network multi-lane recognition system,” in *Proceedings of the 12th International Conference on Microelectronics*, 14-16 Dec 1998, pp. 178–182.

[19] R. Risack, N. Mohler, and W. Enkelmann, “A video-based lane keeping assistant,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, 3-5 Oct 2000, pp. 356–361.

[20] C. R. Jung and C. R. Kelber, “A lane departure warning system based on a linear-parabolic lane model,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, Parma, Italy, Jun 14–17 2004, pp. 891 – 895.

[21] G. Y. Jiang, “Lane and obstacle detection based on fast inverse perspective mapping algorithm,” in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, 2000, pp. 2969–2974.

[22] N. Apostoloff and A. Zelinsky, “Robust vision based lane tracking using multiple cues and particle filtering,” in *Proceedings of IEEE Intelligent Vehicles Symposium*, 9-11 Jun 2003, pp. 558–563.

[23] J. W. Lee, “A machine vision system for lane-departure detection,” *Computer Vision andImage Understanding*, pp. 52–78, Apr 2002.