PERGAMON

Robotics and Computer-Integrated Manufacturing 15 (1999) 111–120

Robotics
and Computer-Integrated
Manufacturing

Estimation of moving vehicle locations using wheel shape information in single 2-D lateral vehicle images by 3-D computer vision techniques[1]

Chih-Chiun Lai, Wen-Hsiang Tsai*

Department of Computer and Information Science, National Chiao Tung University, 1001 Hsueh Road, Hsinchu, Taiwan 30050, ROC

Received 10 February 1998; received in revised form 2 October 1998; accepted 13 November 1998

Abstract

An approach to the estimation of moving lateral vehicle locations for driving assistance using wheel shape information in single 2-D vehicle images by 3-D computer vision techniques is proposed. The location scheme is supposed to be performed on a vehicle with a camera mounted on the front bumper. An analytical solution is applied to estimate locations of the lateral vehicle. Firstly, the rear wheel shape of a lateral vehicle moving in a nearby lane is imaged. By using the Hough transform, the projected wheel shape, which is an ellipse, is detected. Secondly, the equation of the detected ellipse is used to infer the orientation angle of the lateral vehicle with respect to the camera view direction. Finally, the center of the ellipse shape is used to determine the relative position of the lateral vehicle with respect to the camera lens center. Moreover, an edge-point verification algorithm is utilized to extract the ellipse shape more precisely in the image processing stage. Both computer simulated and real images are tested and good experimental results show the effectiveness of the proposed approach for estimating lateral vehicle locations. The results are useful for driving assistance and vehicle collision avoidance and are discussed in detail. © 1999 Elsevier Science Ltd. All rights reserved.

Keywords: Estimation of lateral vehicle location; Driving assistance; Vehicle avoidance; Wheel shape; Ellipse detection; Computer vision

1. Introduction

An important application of autonomous vehicle researches by computer vision techniques is human driving assistance, for example, road following and collision avoidance. In a traffic environment, there are three views that a driver should take care: the front view, the rear view and the lateral views. Most researches [1–3, 6, 7] so far focus on vehicle or obstacle detection in the front view for collision avoidance, and on road or line following methods for vehicle guidance. There are some studies [4, 5] about observing the vehicle rear view. On the other hand, when driving a vehicle in a certain lane on a road, a driver should also notice whether there exists a vehicle in a nearby lane. This is necessary when the driver wants to drive into the nearby lane, or when a vehicle in a nearby lane is trying to get into the driver's lane. There are still very few studies on the detection and location of the lateral vehicle under such circumstances. In this study, the wheel shape information of the lateral vehicle is employed to locate the lateral vehicle, in which the wheel information is captured with a camera mounted on the driver's vehicle. The proposed approach is based on computer vision techniques.

In recent years, a lot of related studies about vehicle detection and location have been proposed [1–7]. For model-based location of vehicles, Ku and Tsai [1] and Lee et al. [2] both used a pre-learned model. By matching the image and the model information, the location of a vehicle is obtained, which is used for vehicle guidance. However, these approaches cannot be utilized to detect whether a lateral vehicle exists. They are used to estimate the location of the vehicle itself, but what we want to know here is the relative location of a lateral vehicle.

For detection and tracking of the vehicle in front, Charkari and Mori [3] proposed a method to detect a moving vehicle in front of a robot from the forward direction. The intensity value of the vehicles underneath is used to detect the vehicle boundary, and a dynamic tracking window is used to track the moving vehicle. Then the velocity of the vehicle is determined by the time interval of two consecutive image processing stages.

* Corresponding author. Fax: 00-88-63-572-1490.

[1] This paper is supported under the project NSC 85-2213-E-009-014 of National Science Council.

S0736-5845/99/$ – see front matter © 1999 Elsevier Science Ltd. All rights reserved.
PII: S0736-5845(99)00006-X

Sasaki et al. [4], and Yanaguchi and Nakajima [5] proposed rear-view observing systems using CCD cameras and ultrasonic sensors, respectively. However, these approaches [3–5] are also unsuitable for handling lateral views.

For vehicle trajectory supervision, Chapuis et al. [6] developed a system for the real-time control of the vehicle trajectory on a highway based on an on-board vision system using a single camera. The system has been designed to avoid damages (e.g., due to a sleepy driver). The method is based on the real-time extraction of the relative vehicle location with respect to the lateral road line, and the extracted vehicle location is used to determine the vehicle trajectory. By combining the vehicle trajectory information and some predefined danger criteria, a trajectory supervision module is used to warn the driver when an emergency situation is encountered. In Pomerleau and Jochem [7], an automated vehicle steering system was proposed, which decomposes the vehicle steering work into three steps: sampling the image, determining the road curvature, and assessing the lateral offset of the vehicle relative to the lane center. As road-departure warning, the system can also compare the assessed lateral offset relative to the lane center with the human driver's steering direction to warn the driver at a dangerous time. Similarly, these systems do not consider lateral vehicle information.

When an autonomous vehicle changes the lane to a nearby one, possible collision with a vehicle on the nearby lane should be avoided. Hence, the extraction of lateral vehicle information is emphasized in this study. A camera is mounted on the right side of the front bumper of the autonomous vehicle and the optical axis of the camera is directed to the lateral side at a certain angle. An illustration of the camera setup on a vehicle is shown in Fig. 1. A typical view captured by the camera is shown in Fig. 2. The geometric relation of the observing vehicle and a lateral vehicle is described in Fig. 3.

To estimate the lateral vehicle location, an image of the vehicle is taken first. This is the start of an image processing stage. The Hough transform [8] is then applied to detect the projected shape of a wheel, which is an ellipse. Because the detected ellipse of the Hough transform may not be the exact projected shape of the wheel but a rough one, an edge-point verification procedure is used to compute the ellipse shape more precisely. After the polynomial coefficients of the ellipse equation are determined, the lateral vehicle orientation angle can be obtained by substituting the coefficients into a formula derived in this study. Finally, the center of the ellipse shape is computed and used to determine the relative position of the lateral vehicle with respect to the camera lens center by the backprojection principle. In this study, the projected shape of a wheel is an ellipse with a little rotation, and it is assumed that the ellipse is without rotation to speed up the location computation work. In addition, the heights of the centers of the wheels of the cars of different brands vary within a small range, and it is assumed to be fixed. Hence, some simulations are performed to test whether these assumptions yield tolerant errors.

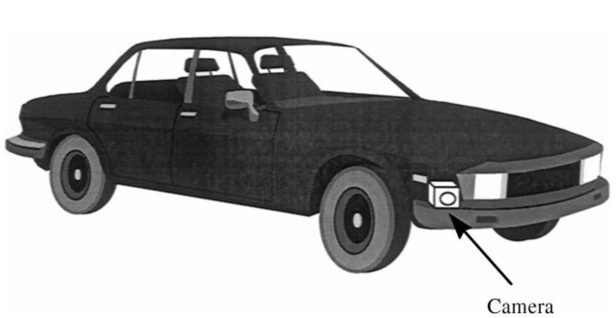

Fig. 1. Illustration of a vehicle mounted with a camera for imaging lateral views.

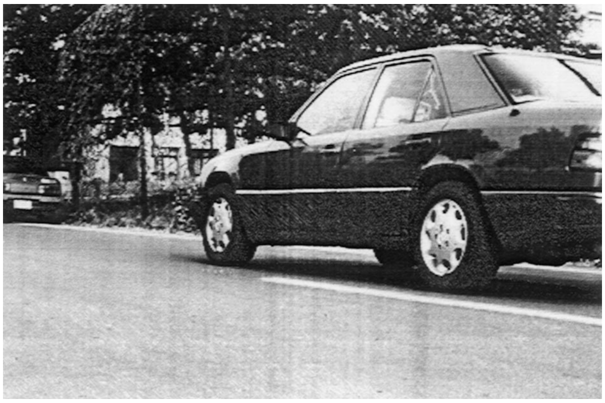

Fig. 2. A typical lateral image through the camera view.

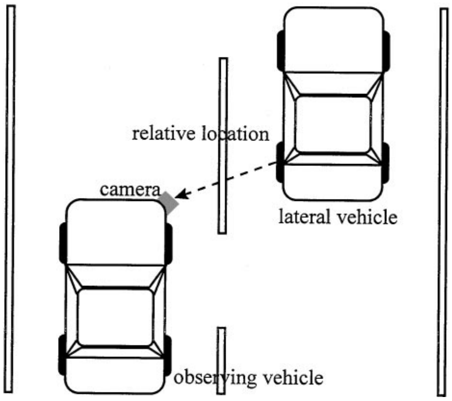

Fig. 3. Geometrical relation between two vehicles.

The major contribution of this study is that the wheel shape information of the lateral vehicle is utilized for the first time. By the estimation of the vehicle location, a lot of useful information is gathered and can be used for vehicle avoidance, lateral vehicle motion estimation, etc. These applications are important for autonomous vehicle navigation in the circumstance with multiple moving vehicles in multiple lanes.

The remainder of this paper is organized as follows. In Section 2, the proposed analytical method to estimate the location of a circle is described. In Section 3, the image processing techniques used to detect an ellipse shape are presented. In Section 4, some experimental results are described. The discussion is shown in Section 5. Finally, conclusions and some suggested future works are given in Section 6.

2. Estimation of 3-D location of a circular wheel shape

In this section, the formulation of the problem of 3-D location of a circular wheel shape and its solution is derived. We emphasize that the derived solution is analytical and so is more suitable for fast processing. First, we know that the surface normal vector of a wheel of a lateral vehicle is always parallel to the ground plane. Next, we calibrate the camera mounted on the autonomous vehicle in advance in such a way that the swing and tilt angles of the camera with respect to the ground plane are zero. Then, the orientation parameters used to describe the surface normal of a wheel shape can be simplified from three parameters down to one parameter, i.e., the pan angle θ. Hence, the lateral vehicle location with respect to the camera coordinate system can be described with four parameters, one parameter θ for the orientation and three parameters (X_c, Y_c, Z_c) for the position vector of the center of the wheel surface. The geometrical situation is shown in Fig. 4. The 3-D location problem is solved in two steps in this study. In the first step, the 3-D orientation parameter of the wheel which is circular in shape is derived. Second, the 3-D position parameters are derived based on the backprojection principle with the knowledge of the height of the wheel center.

Fig. 4 shows the camera coordinate system (CCS), denoted as $X{-}Y{-}Z$, and the image plane $Z = f$, where f is the focus length of the camera. The Z-axis is along the optical axis of the camera. The object coordinate system (OCS), denoted as $x{-}y{-}z$, is set up on the wheel of the lateral vehicle, and the z-axis is along the normal vector of the wheel circle through the circle center. The $X{-}Z$ plane and the $x{-}z$ plane are both parallel to the ground. The center of the circle O is chosen as the origin of the OCS, and its 3-D coordinates with respect to the CCS are (X_c, Y_c, Z_c). The image coordinate system (ICS) is denoted as $u{-}v$. The transformation from the OCS to the CCS [9] can be written as

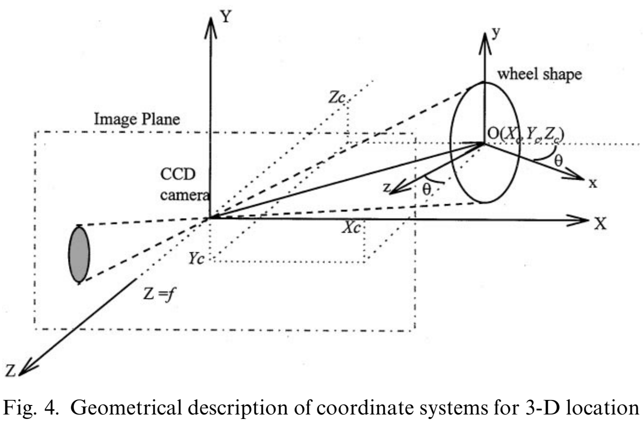

Fig. 4. Geometrical description of coordinate systems for 3-D location estimation of a circular wheel shape.

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & -X_c \\ 0 & 1 & 0 & -Y_c \\ 0 & 0 & 1 & -Z_c \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}. \tag{1}$$

Let (u, v) be the coordinates of the perspective projection of a 3-D point (X, Y, Z) on the image plane. The perspective transformation from the OCS$(X{-}Y{-}Z)$ to the ICS$(u{-}v)$ can be written as

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{f}{Z}\begin{pmatrix} X \\ Y \end{pmatrix}. \tag{2}$$

In addition, the projection of a circle in the image plane is an ellipse and the ellipse equation, which is assumed to have been obtained by some image processing techniques described later, can be expressed as

$$Au^2 + Buv + Cv^2 + Du + \mathrm{E}v + F = 0. \tag{3}$$

The circle equation in the 3-D OCS is expressed as

$$\begin{aligned} x^2 + y^2 &= R^2, \\ z &= 0. \end{aligned} \tag{4}$$

By combining Eqs. (1)–(4), the orientation parameter θ can be obtained. The desired solution is

$$\theta = \frac{1}{2}\cos^{-1}\left(\frac{-a_2 \pm \sqrt{a_2^2 - 4a_1a_3}}{2a_1}\right), \tag{5}$$

where

$$a = \frac{A}{C}, \quad b = \frac{D}{Cf}, \quad c = \frac{F}{Cf^2},$$

$$d = 2(1-c) - (a-c),$$

$$e = -(a-c),$$

$$a_1 = b^2 + e^2,$$

$$a_2 = 2de,$$

$$a_3 = d^2 - b^2.$$

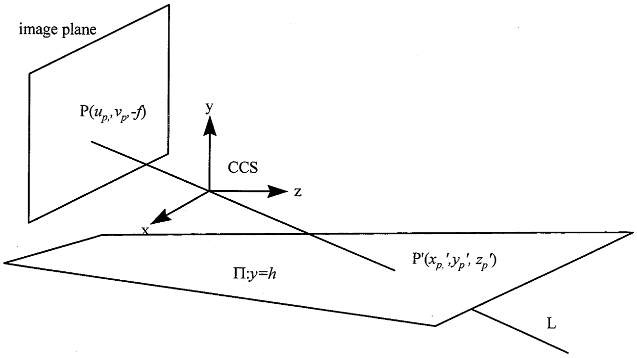

Fig. 5. Illustration of the backprojection principle for finding the space point corresponding to an image pixel.

Next, the position parameters of the wheel center can be derived by the backprojection principle which is described below. As shown in Fig. 5, after backprojecting the point P in the image into the CCS, we can get a line L which passes the lens center and P. The intersection point of the line L and the horizontal plane Π is the corresponding space point of P which we want. Denote this point as P′.

The equation of the horizontal plane Π can be set as $y = h$, where h is the height of the wheel center. Assume that point P in the image plane has the CCS coordinates $(u_p, v_p, -f)$ where (u_p, v_p) is the position in the image and f is the focus length. The desired CCS coordinates (x'_p, y'_p, z'_p) of the wheel center of the lateral vehicle can be solved to be

$$\begin{aligned} x'_p &= u_p \cdot \frac{h}{v_p}, \\ y'_p &= h, \\ z'_p &= -f \cdot \frac{h}{v_p}. \end{aligned} \tag{6}$$

3. Image processing techniques

As shown in Fig. 2, the wheel shape of a vehicle in an image is an ellipse. Hence, ellipse detecting techniques must be used. In this study, the sobel operator is first applied to extract edge points. Then, the Hough transform [8] for extracting ellipses is applied. Because the result might include ambiguous ellipses, an additional verification procedure is used to obtain a more precise candidate. There are four stages in the image processing works, which are shown as a flowchart in Fig. 6. Stages 3 and 4, the edge-verification algorithm, in Fig. 6 will be expressed as a detailed algorithm later.

In the first stage, the sobel operator is used to obtain the edge points of the input image. The threshold value for the sobel operator is set as 300 by our experimental experiences.

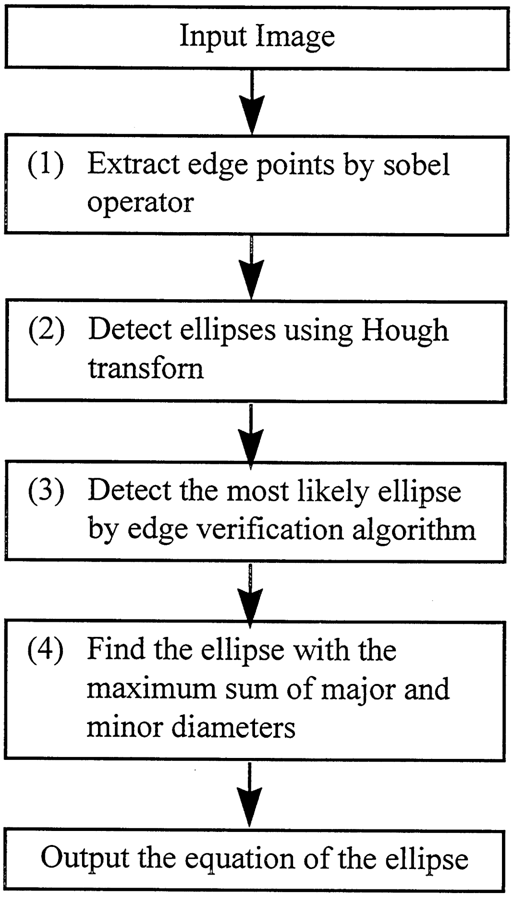
Input Image
(1) Extract edge points by sobel operator
(2) Detect ellipses using Hough transforn
(3) Detect the most likely ellipse by edge verification algorithm
(4) Find the ellipse with the maximum sum of major and minor diameters
Output the equation of the ellipse

Fig. 6. Flowchart of image processing processes.

In the second stage, a two-pass Hough transform procedure is applied. The inputs to the Hough transform are the coordinates (x_i, y_i) of detected edge points, and the parameter space is a four-dimensional counting space, i.e., a space of (x_0, y_0, a, b) in which the parameters come from the ellipse equation $(X - x_0)^2/a^2 + (Y - y_0)/b^2 = 1$, where (x_0, y_0) is the center of an ellipse and (a, b) are the minor and major diameters. The original input image is a 762 × 506 image. First, the image is reduced to 190 × 126 for finding the ellipse quickly. After the first-pass of the Hough transform, a rough location of the center and two diameters are obtained. Next, a rectangular region in the original half-resolution (381 × 253) image obtained from the detected parameters is processed by the Hough transform again to get a more precise solution. This situation is shown in Fig. 7. In the real implementation, the first two stages are combined and the details of the Hough transform used in this study is described in Algorithm 1 below. Assume that X_{min}, X_{max}, Y_{min}, Y_{max}, A_{min}, A_{max}, B_{min} and B_{max} are the minimum and the maximum of x_0, y_0, a and b, and are known in advance.

Algorithm 1: Hough transform for ellipses

Input: An image with ellipses.

Output: The counting space indexed with x, y, a and b.

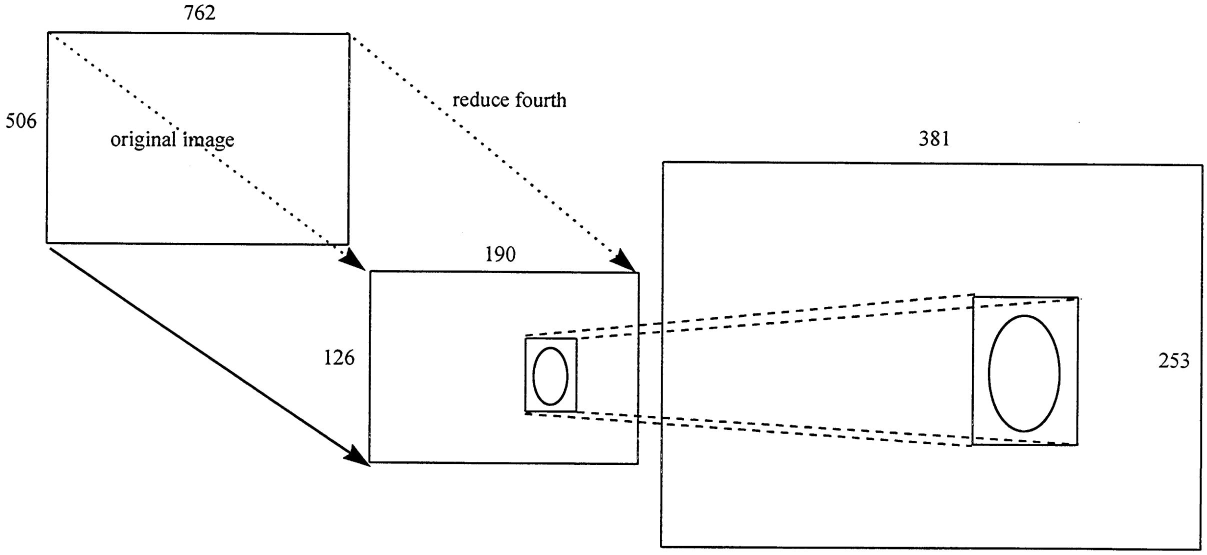

Fig. 7. Illustration of a 2-pass Hough transform.

Steps:

Step 1: for all pixels in the image, i.e., for $x = X_{\min}$ to $X_{\max}$ and for $y = Y_{\min}$ to $Y_{\max}$, do Steps 2–4.

Step 2: Apply the sobel operator: Compute the gradient for each pixel, which is denoted as (GX, GY). If the magnitude of the gradient is larger than a threshold G_t, the corresponding pixel is taken as an edge point and do Steps 3 and 4.

```
Step 3: For $a = A_{\min}$ to $A_{\max}$ do
    for $b = B_{\min}$ to $B_{\max}$ do
        increment the value of the cell indexed with $(x_0, y_0, a, b)$, according to the values of $GX$, $GY$, $a$ and $b$ using the following equations:
```

$$\text{angle} = \arctan\left(\frac{GX}{GY}\right) - \frac{\pi}{2},$$

$$\xi = \tan(\text{angle}),$$

$$\mathrm{d}x = \frac{a}{\sqrt{1 + (b^2/a^2\xi^2)}},$$

$$\mathrm{d}y = \frac{b}{\sqrt{1 + (a^2\xi^2/b^2)}},$$

$$x_0 = x + \text{Sign}\, X(GX, GY)\mathrm{d}x$$

$$y_0 = y + \text{Sign}\, Y(GX, GY)\mathrm{d}y.$$

Step 4: End.

In practical experimental situations, it is possible that the parameter set with the maximum cell counting value is not the exact solution. However, the more precise solution must be in the neighborhood of one of the cells with larger cell counting values, so a verification procedure is applied in the third stage. The cells with counting values larger than 70% of the maximum counting value are checked in the following way. Let the ellipse curve inferred by each of the obtained parameter sets overlap the image of extracted edge-points. The one with the maximum number of overlapping edge points is considered to be a candidate with a good parameter set. However, this edge verification process could not resolve the existence of ambiguous ellipses completely. This situation happens because there are some holes in the wheel and the edge points of those holes can also be detected as candidate ellipses. By inspecting the experimental results, we see that the desired precise ellipse is often the largest in shape of all candidates obtained in the third step and surrounds the others. Hence, the ellipse with the maximum sum of the major and the minor diameter values is selected to be the detected ellipse of a wheel in this study. The details of the verification algorithm is described in Algorithm 2.

Algorithm 2: Verification for removing ambiguous ellipses obtained from the Hough transform

Input: The counting space obtained in Algorithm.

Output: The parameters of the best fitting ellipse.

Steps:

Step 1: Find the cell with the maximum cell counting value (denoted as ***MC***) after the Hough transform.

Step 2: Select all cells with their cell counting values $\boldsymbol{C} > 0.7 \times \boldsymbol{MC}$. Let these cells be denoted as C_i, $i = 1, 2, \ldots$.

Step 3: For each cell C_i, calculate the equation of the corresponding ellipse using the corresponding parameters. Check the 60 points on the ellipse curve with 0, 6, 12, 18, …, 254 degrees to see whether these points overlap an edge point in the edge-point image. Count the number of overlapping points, and denote it by ON_i.

Step 4: If there is only one maximum in all of the ON_i values, output the ellipse with the parameters of the cell corresponding to the maximum as the desired result and halt. Otherwise, continue.

Step 5: Let the maximum of the ON_i values be denoted as MON. For each cell C_j with the MON value, calculate the sum of the values of the major and minor diameters of the corresponding ellipse, and denote the result as DS_j. Find the cell $C_{\max}$ with the maximum DS_j and output the ellipse with the parameters of the cell $C_{\max}$ as the desired result and halt.

4. Experimental results

4.1. Results of image processing

The proposed ellipse extracting algorithm has been implemented on a SUN SPARC 10 workstation and several real images have been tested. An example of the image processing result of ellipse detection is shown in this section. The results of the ellipse extraction process is shown in Fig. 8. Figure 8a is an input image. By applying the sobel edge operator, the resulting edge-point image is obtained and shown in Fig. 8b. Then, the Hough transform was applied. The detected ellipse was drawn in Fig. 8c with black pixels as the result of wheel shape detection. In Fig. 9, more different types of wheels are tested. The experimental results show that the proposed approach can work effectively for different types of wheels.

A reason for using the Hough transform is that the ability to extract ellipses in complex situations is preferred. In fact, the image is captured by a camera mounted on the right-hand side of the front bumper of the autonomous vehicle. In the camera view, only the bottom of the car and the road are seen. Hence, there will not be many artificial objects shown in the image. However, two real images in more complex situations are tested and the results are shown in Fig. 10. In Fig. 10a, there is a man behind the vehicle. In Fig. 10b, the vehicle is on the grass. The experimental results show that the applied method is robust in more complex situations.

In this study, every extraction cycle takes about 1 min on an average. The speed can be reduced if a hardware implementation of the algorithm is carried out. In addition, it is helpful to reduce the processing time if a tracking window is used in processing the image sequence, because the counting space of the Hough transform is downsized.

Comparing the proposed approach with other ellipse detection methods, the ellipse detector using geometric symmetry [11], least-squares fitting [12], and the least median of squares [12], we see that the three methods above are not robust in the presence of outliers in edge points. Before the three methods above are applied, a pre-processing process to detect the candidate edge points of an ellipse must be used. Otherwise, the ellipse will be missed because of some noisy points. This pre-processing is usually not easy to do in real complex situations. However, the experimental results in this study have shown that the proposed approach is robust for real images.

4.2. Experimental results of simulation for location estimation

In the perspective projection model, the projection of a circle is an ellipse with rotation in most cases. However, it is assumed in this study that the projection of a wheel is an ellipse without rotation, i.e., an ellipse of the equation $(X - x_0)^2/a^2 + (Y - y_0)^2/b^2 = 1$. A reason is to speed up the Hough transform in detecting ellipse shapes because accordingly the dimensionality of the Hough counting space can be reduced by one. That is, we use an ellipse without rotation to approximate a real ellipse with rotation of a small angle in our experiments. Consequently, an error factor for the orientation parameter is introduced. A computer simulation process is designed in this study to test whether the error is in a tolerable range for real applications. First, a circle is projected onto the image plane, in which the center of the circle is located in

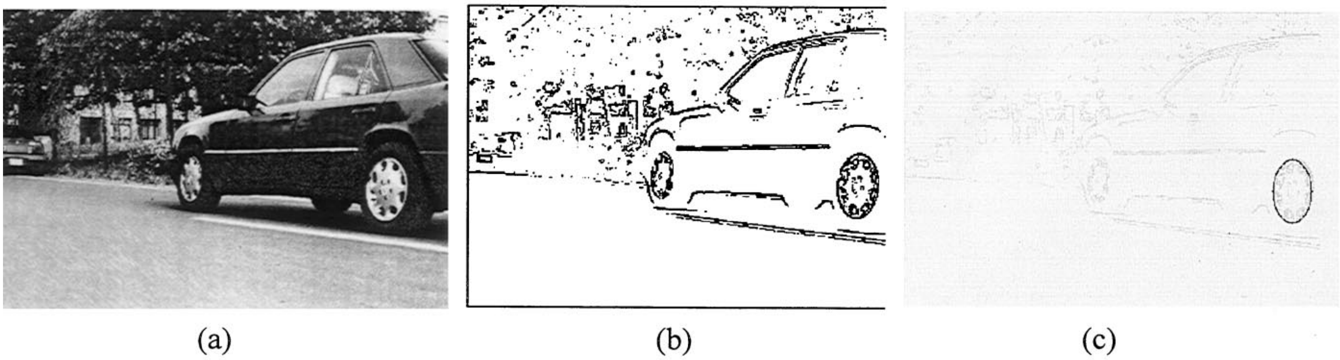

Fig. 8. (a) An example of input image. (b) Result of edge detection. (c) Result of ellipse detection.

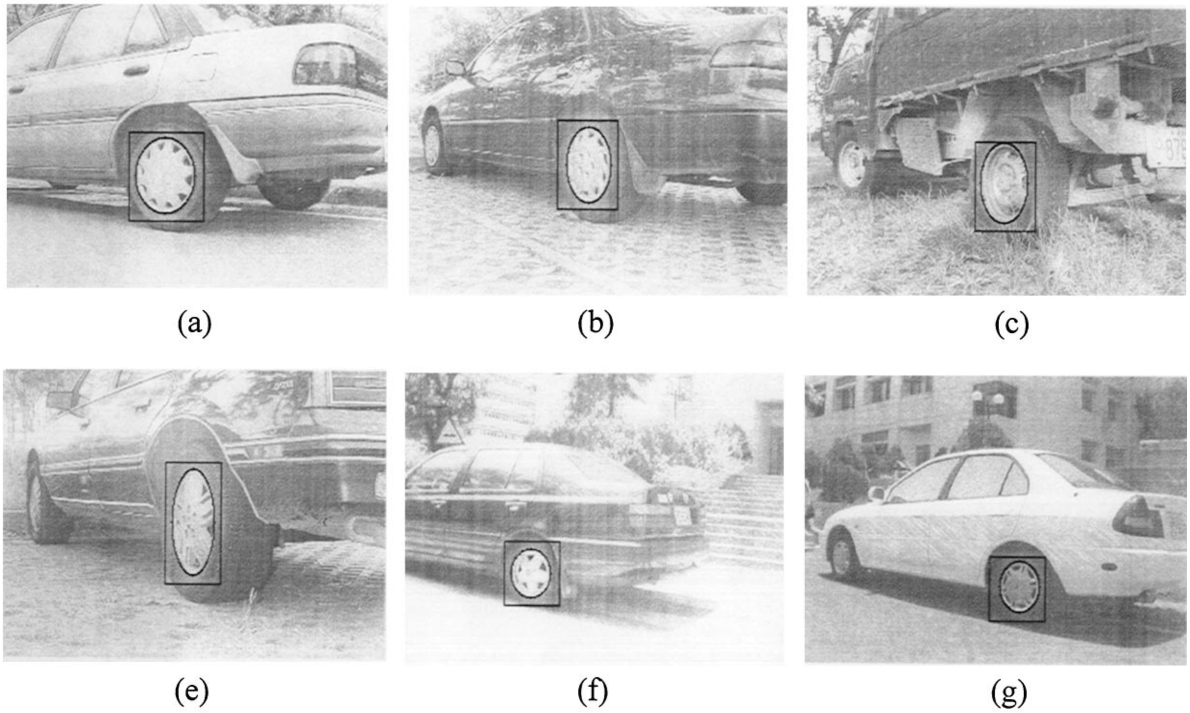

Fig. 9. Results of ellipse detection for different types of wheels.

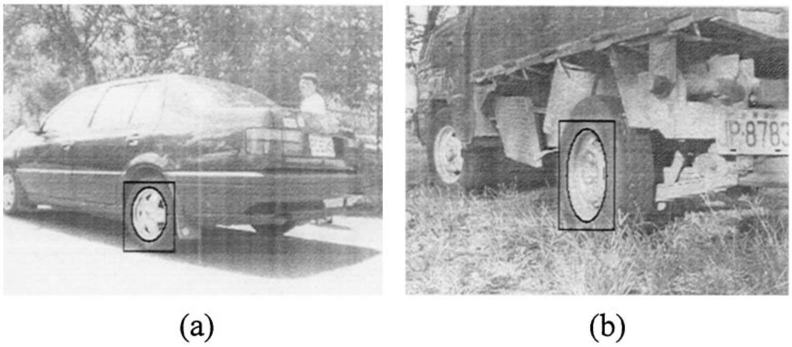

Fig. 10. Results of ellipse detection in more complex situations.

a range with Z_c from 100 to 350 cm in intervals of 50 cm, X_c from $-Z_c$ to 400 cm and rotation angle θ from 5 to 75° in intervals of 5°. Second, a least-square-fit method is used to approximate the precise ellipse with an ellipse with no rotation. At last, the parameters of the equation of the approximate ellipse are substituted into Eq. (5) in Section 2. The results of the error analysis are shown in Table 1. It is seen that the average error of the rotation angle is smaller than 1°. Hence, the approximation of an ellipse with rotation by an ellipse without rotation is feasible in practice.

In the backprojection stage, the height of the wheel center should be known in advance. By measuring the heights of the wheel centers of a lot of cars of different models and manufacturers, we obtained the range of the height values to be from 27.5 to 31.5 cm (see Table 2). Because in the process of lateral vehicle location, the kind of the wheels on the vehicle are unknown in advance, we assume that the heights of the wheel centers of all cars are the same. In this way, an error factor for the translation parameter is introduced, so another computer simulation experiment was conducted to test whether the error is

Table 1
Simulation results of orientation angle computation

θ (°)	5.0	15.0	25.0	35.0	45.0	55.0	65.0	75.0	85.0
Average error (°)	0.04	0.37	0.99	1.19	0.73	0.97	0.51	0.58	0.64

Table 2
Simulation results of backprojection of the wheel center

Height of wheel center (cm)	Height of camera											
	67.5 (cm)		68.5 (cm)		69.5 (cm)		70.5 (cm)		71.5 (cm)		72.5 (cm)	
	X (%)	Z (%)	X (%)	Z (%)	X (%)	Z (%)	X (%)	Z (%)	X (%)	Z (%)	X (%)	Z (%)
27.5	5.00	5.00	4.88	4.88	4.76	4.76	4.65	4.65	4.54	4.54	4.44	4.44
28.5	2.56	2.56	2.50	2.50	2.44	2.44	2.38	2.38	2.32	2.32	2.27	2.27
29.5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
30.5	2.70	2.70	2.63	2.63	2.56	2.56	2.50	2.50	2.44	2.44	2.38	2.38
31.5	5.55	5.55	5.40	5.40	5.26	5.26	5.12	5.12	5.00	5.00	4.87	4.87

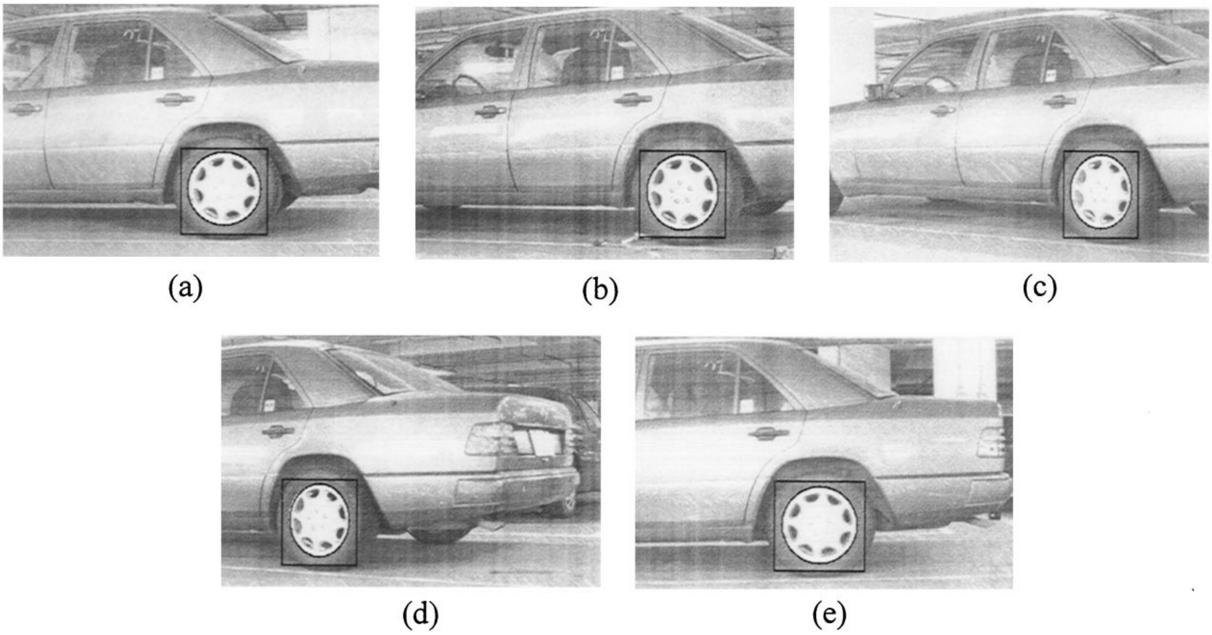

Fig. 11. Results of image processing of image set 1.

tolerable or not. First, the wheel center was projected onto the image plane by a set of the precise translation parameters. Then the backprojection process derived in Section 2 was applied under the assumption that the heights of the wheel centers of all cars are 29 cm, which is the average value of the range of the wheel center heights from 27.5 to 31.5 cm. The results are shown in Table 4. The average error rate of the translation parameters are small than 5%, so the assumption of constant wheel center heights is considered feasible in practice.

4.3. Experimental results of real images for location estimation

The proposed method was applied to two sets of images. In the first set, each image is independent. The results of image processing of the first set are shown in Fig. 11. In the second set, three images of a sequence were used. The results of image processing of the second set are shown in Fig. 12. The results of location of the two sets are listed in Tables 3 and 4, respectively. In each of these tables, the column 'M' of each parameter is measured manually and was used as a reference to test the precision of the corresponding location result. The column 'm' of each parameter was obtained by the formula derived in Section 2. At last, the column 'e' of each table is the error measurement. For the translation parameters, i.e., for X and Z, the value of column 'e' specifies an error rate, which is the ratio of the difference between column 'M' and column 'm' to the value of 'M'. For the rotation parameter, the value of column 'e' denotes the difference between column 'M' and column 'm'. As shown in Tables 3 and 4, the error rate values are less than 5%, so the proposed method is feasible for locating a lateral vehicle.

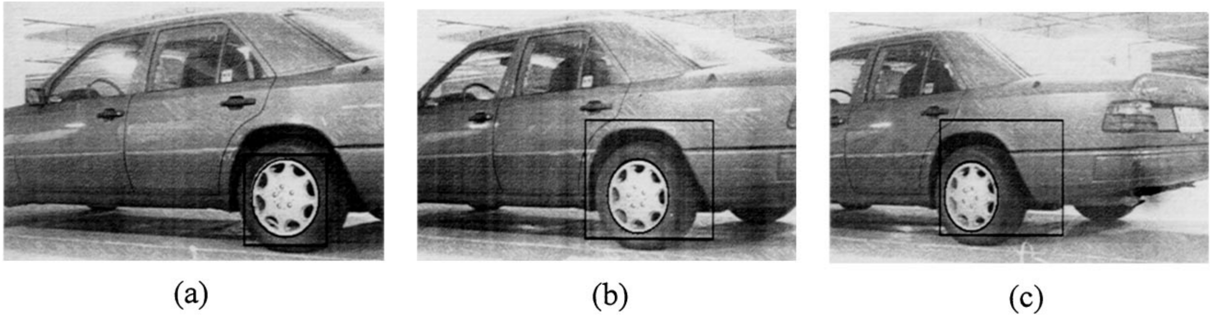

Fig. 12. Results of image processing of image set 2.

Table 3
Result of location for image set 1

	X (cm)			Z (cm)			θ (°)		
	M	m	E (%)	M	m	e (%)	M (°)	m (°)	e (°)
Fig. 11a	24.00	23.4	2.5	−452.36	−474.78	4.96	−15.37	−16.50	1.13
Fig. 11b	49.5	51.67	4.38	−449.28	−466.67	3.87	−25.79	−25.65	−0.14
Fig. 11c	55.90	57.27	2.45	−470.69	−491.89	4.5	−39.44	−39.74	0.30
Fig. 11d	−56.50	−58.34	3.25	−458.53	−451.24	1.59	−28.40	−28.81	0.39
Fig. 11e	0.00	−2.85	—	−444.00	−443.9	0.02	−2.58	−12.67	10.09

Table 4
The result of location for image set 2

	X (cm)			Z (cm)			θ (cm)		
	M	m	e (%)	M	m	e (%)	M (°)	m (°)	e (°)
Fig. 12a	52.80	55.33	4.79	−424.73	−423.26	0.35	−36.91	−37.19	0.28
Fig. 12b	19.00	19.34	1.80	−451.60	−443.90	1.70	−36.91	−35.74	−1.17
Fig. 12c	−30.8	−31.91	3.60	−470.99	−451.24	4.19	−36.91	−35.70	−1.21

5. Discussion

The orientation and translation parameters obtained by our method are useful information in the application of vehicle avoidance. Because the surface normal of the rear wheel shape is perpendicular to the moving direction of a lateral vehicle, we can derive its moving direction from the relative angle between the surface normal of the rear wheel and the camera. Then, the trajectory of the lateral vehicle can be predicted. Hence, a dangerous situation could be avoided.

By the experimental results, we have shown that the method developed in this paper is reliable in various situations. However, the error of the rotation angle, as shown in results of Fig. 11e of Table 3, is larger than 10° which seem considerably large. This situation happens when the rotation angle is considerably small, say smaller than 10°. The projection of the wheel is almost an ellipse without rotation in this situation and the major and minor diameters are different only in one or two pixels. This causes an error with one pixel in the result of image processing, which in turn causes an error of the rotation angle from 2 to 10°. It is thus concluded that the error is increased when the rotation angle is near 0°. However, this situation with a rotation angle from 0 to 10° means that the lateral vehicle is almost in the same direction as that of the observing vehicle, and the error will not cause dangerous consequences. Hence, this situation may not be considered serious, that is, safety in driving assistance can still be maintained.

6. Conclusions and future works

A new approach to the detection of moving vehicle locations for driving assistance using single 2-D lateral vehicle images by 3-D computer vision techniques has been proposed. The rear wheel shape information in a lateral vehicle image is used to infer an analytical solution of the orientation angle and the relative position of the lateral vehicle with respect to the driver's vehicle. The Hough transform is applied to detect the ellipse shape of the wheel in the image. To improve the accuracy of extracted ellipse shapes, an ellipse edge-point verification algorithm has been developed. Both computer simulated and real images were tested and good experimental results prove the effectiveness of the proposed approach.

To extend the ability of the autonomous vehicle for driving assistance or collision avoidance, it would be interesting to estimate the motion of a lateral vehicle by a sequence of images when the time interval of two contiguous images is known. Firstly, the relative direction of a lateral vehicle can be obtained in a single image. Secondly, the speed of the observing vehicle is easy to obtain. Combining the time interval of two contiguous images, the motion of a lateral vehicle can be estimated. In addition, when the information of both the front and rear wheel shapes is used, the trajectory of a lateral vehicle can be predicted much better. These works are very helpful for collision avoidance and are worth further studying.

References

[1] Ku PY, Tsai WH, Model-based guidance of autonomous land vehicle for indoor navigation. Proc of Workshop on Computer Vision, Graphics and Image Processing, Taipei, Taiwan, ROC, 1989;165–74.

[2] Lee PS, Shen YE, Wang LL. Model-based location of automated guided vehicles in the navigation sessions by 3D computer vision. J Robotic Systems 1994;11(3):181–95.

[3] Charkari NM, Mori H. A new approach for real time moving vehicle detection. Proc. 1993 IEEE/RSJ Int Conf on Intelligent Robots and Systems, Yokohama, Japan, 1993:273–78.

[4] Sasaki K, Ishikawa N, Otsuka T, Nakajima M. 3-D image location surveillance system for the automotive rear-view. Vehicle Navigation and Information Systems Conf Proc, Yokohama, Japan, 1994:27–32.

[5] Yamagchi, Nakajima. Development of an observation system for the automotive rear view using a fiber grating visual sensor. Sensor Technol 1993;13(4):18–22.

[6] Chapuis R, Potelle A, Brame JL, Chausse F. Real-time vehicle trajectory supervision on the highway. Int J Robotics Res 1995; 14(6):531–42.

[7] Pomerleau D, Jochem T. Rapidly adapting machine vision for automated vehicle steering. IEEE Expert 1996;11(2):19–27.

[8] Ballard DH. Generalizing the Hough transform to detect arbitrary shapes. Pattern Recognition 1981;13(2):111–22.

[9] Haralick RM, Shapiro LG. Computer and robot vision, vol. 2. Reading, MA, USA: Addison-Wesley, 1993.

[10] Gonzalez RC, Woods RE. Digital image processing. Reading, MA, USA: Addison-Wesley, 1992.

[11] Ho CT, Chen LW. A fast ellipse/circle detector using geometric symmetry. Pattern Recognition 1995;28(1):117–24.

[12] Zhang Z. Parameter estimation techniques: a tutorial with application to conic fitting. Image Vision Computing 1997; 15:59–76.