A Robust Linear-Parabolic Model for Lane Following

Cláudio Rosito Jung and Christian Roberto Kelber
UNISINOS - Universidade do Vale do Rio dos Sinos
Ciências Exatas e Tecnológicas
Av. UNISINOS, 950. São Leopoldo, RS, Brasil, 93022-000
crjung@exatas.unisinos.br, kelber@eletrica.unisinos.br

Abstract

In this paper, we address the problem of lane detection and lane tracking. A linear model is used to approximate lane boundaries in the first frame of a video sequence, using a combination of the edge distribution function and the Hough Transform. A new linear-parabolic model is used in the subsequent frames: the linear part of the model is used to fit the near vision field, while the parabolic model fits the far field. The proposed technique demands low computational power and memory requirements, and showed to be robust in the presence of noise, shadows, lack of lane painting and change of illumination conditions.

1. Introduction

Researchers in the areas of computer vision and intelligent vehicles have been devoting great efforts to develop machine vision systems. Cameras installed inside a vehicle can be used for constant monitoring of the road, detecting in advance tendencies of lane departure or collision with other vehicles [1]. Also, Autonomous Guided Vehicles (AGVs) can rely on computer vision systems for unsupervised navigation. In both applications, it is important to obtain robust information about road boundaries.

Many road boundary detection/following systems have been developed in the past years. However, several conditions can decisively degrade the performance of lane detection techniques:

- shadows: trees, buildings and other vehicles project shadows on the road, creating false edges;
- solar position: direct sunlight may saturate the acquired images, or cause specular reflexes;
- climate: natural fenomena (such as fog, rain or snow) may degrade significantly the quality of the images;
- occlusion: other vehicles may cause partial or complete occlusion of road boundaries.

In this paper, we propose a linear lane model for the first image of the video sequence (initial detection), and a linear-parabolic model for all the remaining frames (road following). In Section 2, some existing lane detection and following techniques are reviewed. Section 3 describes our method for initial road boundary detection based on a linear function. In Section 4, our linear-parabolic model for lane following is presented. Section 5 contains some experimental results of the proposed method, for video sequences with different illumination conditions, shadows and weak road paintings. Finally, the conclusions and ideas for future work are given in the last Section.

2. Related Work

Many methods for road segmentation and lane following have been proposed in the past years. Different approaches, such as watersheds, deformable models and particle filtering were used to tackle these problems.

Kluge [2] proposed a method for estimating road curvature and orientation based on isolated edge points, without the need of grouping them. This system works if at most 50% of input edge points are noisy, which may not happen in practical situations (due to weak road markings, shadows, etc.).

Beucher and his colleagues [3, 4] worked on road segmentation and obstacle detection based on watersheds. Their techniques consist of applying a temporal filter for noise reduction (and connection of ground markings), followed by edge detection and watershed segmentation. Such methods demand a relatively high computational cost and the resulting road boundaries are typically jagged (due to the watershed transform).

Another class of lane detection methods [5, 6, 7] rely on top-view (birds eye) images computed from images acquired by the camera. These methods are reliable in obtaining lane orientation in world coordinates, but require online computation of the top-view images.

Apostoloff and Zelinsky [8] proposed a lane tracking system based on particle filtering and multiple cues. In fact, this method does not track the lanes explicitly, but it computes parameters such as lateral offset and yaw of the vehicle with respect to the center of the road. Although the method appears to be robust under a variety of conditions (shadows, different lighting conditions, etc.), it cannot be used to estimate curvature or detect if the vehicle is approaching a curved part of the road.

Deformable road models have been widely used for lane detection [9, 10, 11, 12, 13, 14]. These techniques attempt to determine mathematical models for road boundaries. In general, simpler models (e.g. linear functions) do not provide an accurate fit, but they are more robust with respect to image artifacts. On the other hand, more complex models (such as parabolic functions and splines) are more flexible, but also more sensitive to noise. Hence, there is a trade-off between accuracy of the fit and robustness with respect to image artifacts.

A typical application for lane detection/following algorithms is the development of lane departure warning systems. Parameters such as orientation and distance of the vehicle with respect to lane boundaries can be used for lane departure detection [15, 16]. As expected, the performance of such systems is highly dependent on the accuracy of the parameters obtained from the lane detection/following algorithm (orientation, lateral offset, etc.).

In this paper, we propose a novel technique for lane detection/following. Basically, our approach consists of two steps:

1. Lane detection - this first step consists of detecting lane boundaries from the initial image (the first frame of the video sequence), using a linear lane model.
2. Lane following - this step consists of updating the detection obtained in the previous frame to the subsequent one, using a new linear-parabolic lane model. The linear part of the model is used in the near field (locally, the road is assumed to be straight), and the parabolic part is utilized in the far field (such that incoming curves can be efficiently detected).

The proposed model combines the robustness of the linear function with the flexibility of the parabolic function, showing a good performance in the presence of noise, shadows and different illumination conditions. Next, we describe in detail these two steps.

3. Initial Lane Detection

In this stage, the first frame acquired by the camera is processed, and the two (left and right) lane boundaries are obtained automatically. Our coordinate system coincides with image coordinates, and a threshold x_m separates the near and far vision fields, as shown in Figure 1. The choice for x_m depends on the size and the quality of the acquired images, but a default value can be half of the image height.

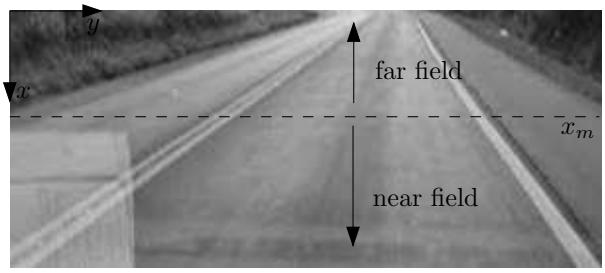

Figure 1. The initial frame of a video sequence, with our coordinate system and the definition of the near and far fields.

For the initial detection, we chose a linear model for the lane boundary, because of its simplicity and robustness. We also assume that the following conditions are satisfied in the first frame of the video sequence:

- the vehicle is initially located in a straight portion of the road;
- the vehicle is approximately aligned with the road;
- there are no linear structures in the image, except for the lane boundaries.

To detect the linear lane boundaries, we combine the *edge distribution function* approach adopted by Lee [15] with the Hough Transform [17].

3.1. The Edge Distribution Function

For the greyscale image $I(x, y)$, the gradient function $\nabla I(x, y)$ can be approximated by:

$$\nabla I(x,y) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right)^T \approx (D_x, D_y)^T , \qquad (1)$$

where D_x and D_y are differences computed in the x and y directions (this differences can be computed using the Sobel operator [18]). We can estimate the gradient magnitude and orientation using the following equations:

$$|\nabla I(x,y)| \approx |D_x| + |D_y|, \qquad (2)$$

$$\theta(x,y) = \tan^{-1}\left(\frac{D_y}{D_x}\right). \qquad (3)$$

To determine the orientation of the road boundaries, we compute the edge distribution function (EDF), which is the histogram of the gradient magnitude with respect to the orientation[1]. To compute this histogram, the angles $\theta(x, y)$

within the range $[-90°, 90°]$ were quantized in 90 subintervals (each one with length of 2°). A look-up table can be used to avoid the computation of $\tan^{-1}$ in Equation (3).

Assuming that lanes are the only significant linear objects in the image, and that the car is approximately aligned with the road, it is expected that the largest peak on the left portion of the histogram (corresponding to $\theta < 0$) will be related to the right lane boundary. Similarly, the largest peak on the right portion of the histogram (corresponding to $\theta > 0$) will be related to the left lane boundary. Figure 2 shows the EDF (a Gaussian filter was used to smooth the histogram) for the road image shown in Figure 1. The largest peak in the negative region of θ, which occurs at $\alpha_1 = -54°$, corresponds to the right boundary of the right lane; similarly, the largest peak in the positive region of θ, occurring at $\alpha_2 = 34°$, corresponds to the left boundary of the right lane. There is a second peak close to α_2, that is related to the left boundary of the left lane.

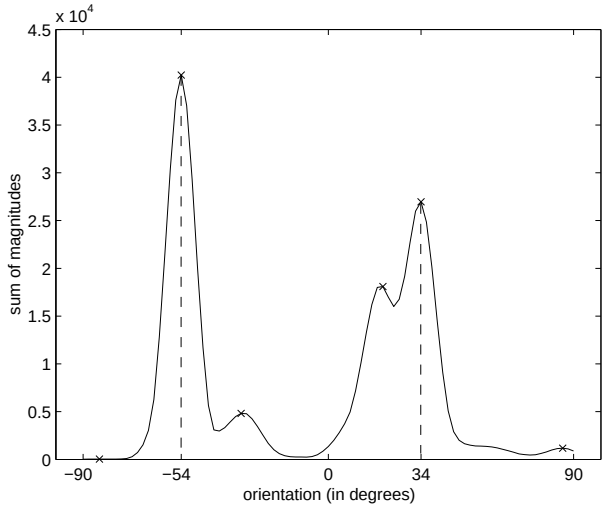

Figure 2. Smoothed Edge Distribution Function (EDF).

Lee [15] used the EDF to determine the orientation of each lane boundary. However, we want to detect these boundaries explicitly, by fitting a linear function. Let α be the orientation corresponding to the desired lane boundary. Also, let $g(x, y)$ be the directional edge image defined as:

$$g(x,y) = \begin{cases} |\nabla I(x,y)|, & \text{if } |\theta(x,y) - \alpha| < T_\alpha \\ 0, & \text{otherwise} \end{cases}, \quad (4)$$

where T_α is an angular threshold (in this work, we used $T_\alpha = 2°$, to match the quantization used in computation of the EDF). It should be noticed that $g(x, y)$ contains edge magnitudes of the original image $I(x, y)$ that are aligned with the direction α. These magnitudes will be mostly related to the lane boundary, but there will be also some pixels related to noise or other structures that are aligned with the lane. Figure 3 shows image $g(x, y)$ for $\alpha = -54°$, which corresponds to the right boundary of the right lane. Indeed, some isolated pixels with small magnitude that are not related to this lane boundary appear in the image.

1 Please, note that the gradient direction is orthogonal to the contour orientation.

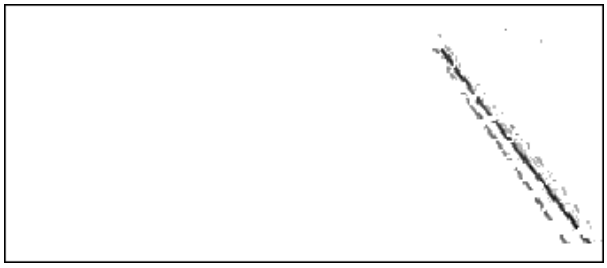

Figure 3. Magnitudes aligned with the right boundary of the right lane.

3.2. The Hough Transform

Applying the Hough Transform to a set of edge points (x_i, y_i) results in an 2D function $C(\rho, \theta)$ that represents the number of edge points satisfying the linear equation $\rho = x\cos\theta + y\sin\theta$. In practical applications, the angles θ and distances ρ are quantized, and we obtain an array $C(\rho_k, \theta_l)$. The local maxima of $C(\rho_k, \theta_l)$ can be used to detect straight line segments passing through edge points.

In our case, the orientation θ can be obtained from the EDF peak α described in Section 3.1 (in fact, we have $\theta = \alpha$, as shown in Figure 4). Thus, we have a one-dimensional search (only the parameter ρ_k). Also, instead of building $C(\rho_k, \alpha)$ by counting the number of edge pixels belonging to $\rho_k = x\cos\alpha + y\sin\alpha$, we use an alternative approach, based on the Gradient Weighted Hough Transform [19].

Let (x_i, y_i), for $i = 1, ..., n$, be the coordinates of non-zero pixels of the thresholded magnitude image $g(x, y)$ corresponding to the orientation α. We define $C(\rho_k, \alpha)$ as:

$$C(\rho_k, \alpha) = \sum_j g(x_j, y_j), \quad (5)$$

where (x_j, y_j) are all the edge pixels belonging to $\rho_k = x\cos\alpha + y\sin\alpha$. By summing edge magnitudes instead of counting the number of aligned pixels, we minimize the influence of edges with small magnitudes (typically related to noise). If ρ_α denotes the global maximum of $C(\rho_k, \alpha)$, the

lane boundary corresponding to the EDF peak α is given by the straight line:

$$y = -x \cot \alpha + \frac{\rho_\alpha}{\sin \alpha} \ . \quad (6)$$

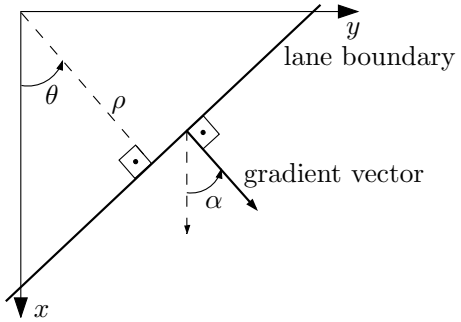

Figure 4. Relation between the EDF peak α and the parameters ρ and θ of the Hough Transform.

This line detection procedure is applied independently to each lane boundary, resulting in one linear model for each boundary. This initial detection is used to find the lane boundary region of interest (LBROI), which will be the search space for lane boundaries in the subsequent frame of the video sequence. In this work, a "thick" line was the chosen LBROI, and can be obtained by extending the linear function in the y direction w pixels to the right and w pixels to the left. The choice of w depends on the width of the lane markings (if they exist), the focal length of the camera and the travelling speed of the vehicle. The LBROIs shown in Figure 5 correspond to the lane boundaries depicted in Figure 1 (using $w = 8$ pixels).

It should be said that other variations of the Hough Transform, such as the Progressive Probabilistic Hough Transform [20], could be used to reduce the computational cost of the initial detection. However, this gain would be insignificant, because the procedure described in this Section is applied just one time.

4. Lane Following

After the initial segmentation in the first frame, we need to update the detection for the following frames. For the initial detection, a linear model was chosen, because it provides a robust automatic detection. However, this model is obviously not suitable for curved roads. In fact, many road/lane models have been proposed in the past years [9, 10, 11, 12, 13]. Simpler models demand less computational power and are usually less sensitive to noise; on the other hand, models with more degrees of freedom can provide a more accurate fit to the lane boundary, but are more likely to be affected by image artifacts.

Figure 5. LBROIs corresponding to the initial lane segmentation using the linear model.

In this paper, we propose a lane boundary model that is flexible enough to follow curved roads, robust with respect to road/backgroud variations (noise, shadows, weak lane markings), and that can provide information about lane orientation and/or curvature. This model is described next.

4.1. Lane Model Formulation

Our lane boundary model $f(x)$ is composed by a linear function in the near field, and a parabolic function in the far field:

$$f(x) = \begin{cases} a + bx, & \text{if } x > x_m \\ c + dx + ex^2, & \text{if } x \leq x_m \end{cases} \ , \quad (7)$$

where x_m represents the border between near and far fields. We impose continuity and differentiability conditions on the function f, such that $f(x_m^+) = f(x_m^-)$ and $f'(x_m^+) = f'(x_m^-)$. These conditions imply that:

$$\begin{cases} a + bx_m = c + dx_m + ex_m^2 \\ b = d + 2ex_m \end{cases} \ . \quad (8)$$

We can solve this system for the variables c and e, obtaining:

$$c = a + \frac{x_m}{2}(b - d) \ \text{ and } \ e = \frac{1}{2x_m}(b - d) \, . \quad (9)$$

Replacing these values back into Equation (8), we obtain:

$$f(x) = \begin{cases} a + bx, & \text{if } x > x_m \\ \frac{2a + x_m(b-d)}{2} + dx + \frac{(b-d)}{2x_m}x^2, & \text{if } x \leq x_m \end{cases} \ . \quad (10)$$

Hence, we need only three coefficients (a, b and d) to describe our lane boundary model. To determine these parameters, we apply a minimum weighted square error approach, fitting the proposed model to the detected edges within the LBROI. This procedure is applied independently for each lane boundary, and is described next.

4.2. Fitting the Lane Model

Let us consider that a lane boundary was detected in the previous frame, and the corresponding LBROI was obtained. The edge image $|\nabla I(x,y)|$ of the current frame is computed within the LBROI. Most of the edges will be related to the lane boundary, but some edges related to noise, road texture or other structures will also appear. To remove these undesired edges, we apply an adaptive threshold based on the mean magnitude M_{mean} of the edges. In fact, we remove all the edges with magnitudes smaller than $0.5M_{\text{mean}}$. Let $g(x,y)$ denote the thresholded edge image:

$$g(x,y) = \begin{cases} |\nabla I(x,y)|, & \text{if } |\nabla I(x,y)| \geq 0.5M_{\text{mean}} \\ 0, & \text{otherwise} \end{cases}. \tag{11}$$

It should be noticed that this adaptive threshold is not affected by varying illumination conditions, and does not required any *a priori* information about the contrast between the road and the image background (or the intensity of the lane markings, if they exist). Figure 6 shows the thresholded edge image $g(x,y)$ for the right LBROI depicted in Figure 5.

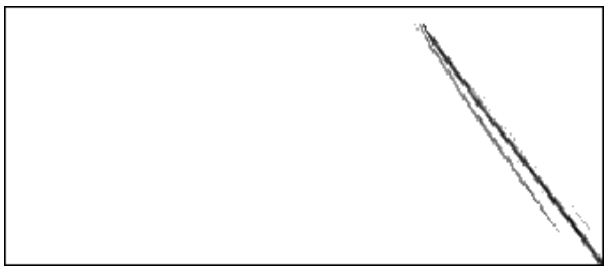

Figure 6. Thresholded edge image corresponding to the right LBROI.

Let (x_{n_i}, y_{n_i}), for $i = 1, ..., m$, denote the m coordinates of the non-zero pixels of the thresholded edge image $g(x,y)$ belonging to the near field, and $M_{n_i} = g(x_{n_i}, y_{n_i})$ the respective magnitudes. Analogously, let (x_{f_j}, y_{f_j}) and $M_{f_j} = g(x_{f_j}, y_{f_j})$, for $j = 1, ..., n$, represent the same characteristics for the n edge pixels in the far field.

Fitting the lane model (10) to the edge data results in a linear system with 3 unknowns and $n+m$ equations:

$$\begin{cases} a + bx_{n_i} = y_{n_i}, & i = 1...m \\ \frac{2a + x_m(b-d)}{2} + dx_{f_j} + \frac{(b-d)}{2x_m} x_{f_j}^2 = y_{f_j}, & j = 1...n \end{cases}. \tag{12}$$

Typically, $(n+m)$ will be much greater than 3, and this system will not admit an exact solution. However, we can find an approximated solution such that a specific error measure is minimized. Assuming that edges related to lane boundaries usually have larger magnitudes than edges related to other irrelevant structures (such as noise, road texture, etc.), we propose the following weighted square error measure:

$$E = \sum_{i=1}^{m} M_{n_i} \left[y_{n_i} - f(x_{n_i}) \right]^2 + \sum_{j=1}^{n} M_{f_j} \left[y_{f_j} - f(x_{f_j}) \right]^2. \tag{13}$$

This error is minimized when the following 3×3 linear system is solved:

$$\mathbf{A}^T \mathbf{W} \mathbf{A} \mathbf{c} = \mathbf{A}^T \mathbf{W} \mathbf{b}, \tag{14}$$

where

$$\mathbf{A} = \begin{bmatrix} 1 & x_{n_1} & 0 \\ \vdots & \vdots & \vdots \\ 1 & x_{n_m} & 0 \\ 1 & \frac{1}{2x_m}\left(x_{f_1}^2 + x_m^2\right) & -\frac{1}{2x_m}\left(x_{f_1} - x_m\right)^2 \\ \vdots & \vdots & \vdots \\ 1 & \frac{1}{2x_m}\left(x_{f_n}^2 + x_m^2\right) & -\frac{1}{2x_m}\left(x_{f_n} - x_m\right)^2 \end{bmatrix},$$

$$\mathbf{W} = \begin{bmatrix} M_{n_1} & & & & & \\ & \ddots & & & & \\ & & M_{n_m} & & & \\ & & & M_{f_1} & & \\ & & & & \ddots & \\ & & & & & M_{f_n} \end{bmatrix},$$

$$\mathbf{c} = [a, b, d]^T \text{ and } \mathbf{b} = [y_{n_1}, \ldots, y_{n_m}, y_{f_1}, \ldots, y_{f_n}]^T.$$

It should be noticed that $\mathbf{A}^T\mathbf{W}\mathbf{A}$ is a symmetric matrix. Hence, only a triangular portion (upper or lower) of the matrix must be computed, reducing the computation burden.

Figure 7 shows the proposed model fit to the second frame of our video sequence, where the dashed line indicates the border between the far field and the near field. Since this Figure illustrates a straight portion of the road, the parabolic part of the model is approximately linear. In Section 5, some results are shown for curved roads.

Similarly to the procedure applied in the initial segmentation, the LBROI is obtained by "thickening" the fitted function in the y direction, w pixels to the right and w pixels to the left.

In the subsequent frame, the edge image will be computed only in a small region delimited by the LBROI obtained in the current frame. The procedure described in Section 4.2 is then repeated for the remaining frames.

The proposed lane following technique requires low computational cost and memory needs. In summary, an edge detector is applied at the LBROIs (which is only a small fraction of the image), and edge magnitudes larger

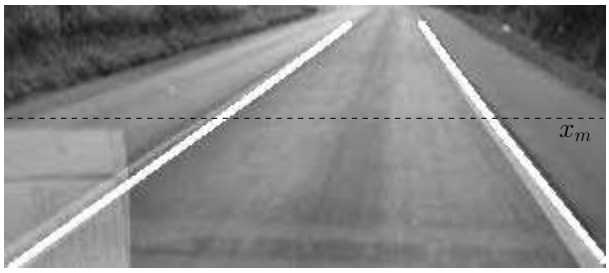

Figure 7. The proposed model superimposed to the original image (second frame of the video sequence). The dashed line indicates the border between the near and far fields.

than a threshold are used to compute the 3×3 symmetric linear system given in Equation (14). Since this procedure is applied to each lane boundary separately, the algorithm can be parallelized.

5. Experimental Results

We tested our lane detection/following algorithm for different video sequences containing several conditions that may degrade the accuracy of the proposed algorithm, such as varying illumination conditions, presence of shadows and weak painting of the lane markers. All video sequences in this work were obtained at 30 frames per second, with a resolution of 240×320 pixels.

The first video sequence depicts a road, and was obtained with a travelling speed of approximately 80 km/h. In fact, only the central portion of the images (140×320 pixels) was processed, because we needed to remove the horizon (top of the image) and the windshield wipers (bottom of the images). Figure 8 shows twelve frames of this sequence, separated by approximately 3 seconds. They are displayed from left to right, top to bottom. It can be noticed that both lane boundaries are efficiently detected, even in the presence of shadows (frames 6-8), the reflex of a book on the windshield (frames 1-7), different illumination conditions (please, notice that the asfalt is much brighter in frames 1-2 in comparison with other frames) and very poor painting conditions (frames 10-12). As expected, the accuracy of the model fitting is dependent on the conditions of the lane markers.

The second video sequence depicts the internal avenue of our university, and was obtained with a travelling speed of approximately 40 km/h. The central portion of the images was analyzed (110×320 pixels), to remove the hood of the car and the horizon. Figure 9 shows twelve frames of this sequence. It can be observed that the proposed algorithm also works in unpainted roads, as long as there is enough contrast between the road and the background (in this video sequence, the algorithm captures the contrast between the road and the sidewalk to obtain the left lane boundary). Markings on the center the road (such as speed limits and arrows) do not affect our lane following procedure, as illustrated in frames 8 and 12. However, pedestrian crossings may cause some confusion to the algorithm, because edges close to lane boundaries are detected. In this case, our model tends to follow the crossing, as shown in the 6th frame. As the crossing disappears, the model correctly fits back the lane boundary. Also, it can be noticed that the algorithm may work in intersections, if there is enough information about lane boundaries (as shown in the 8th frame).

The third video sequence was obtained at night, and contains 1200 frames acquired at 30 fps. An additional problem of nocturnal images is saturation caused by headlights of incoming vehicles, as shown in Figure 10. This Figure illustrates a time period of 1.6 seconds, when a truck and a car were travelling towards the camera. Despite the strong edges caused by the headlights and local saturation, the proposed linear-parabolic model was robust enough to correctly fit lane boundaries.

6. Conclusions and Future Work

In this paper, we proposed a new lane detection/following algorithm. In the initial detection, a linear model is fitted to the lane boundaries, using a combination of the EDF and the Hough Transform. For the lane following step, a new lane boundary model was proposed, consisting of a linear model for the near field, and a parabolic model for the far field. This model is fitted to the lane boundaries by minimizing a weighted square error.

The proposed model combines the robustness of the linear model with the flexibility of the parabolic model. As shown in Section 5, our technique can accurately fit straight or curved portions of the road, performing well in the presence of shadows, image artifacts, varying lighting conditions, and poor painting of lane markings.

For future work, we intend to develop a lane departure warning system based on information (such as orientation, curvature and lateral offset) extracted from our linear-parabolic model. Another idea is to further extend our method for autonomous driving.

References

[1] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, vol. 21, pp. 359–381, April 2003.

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

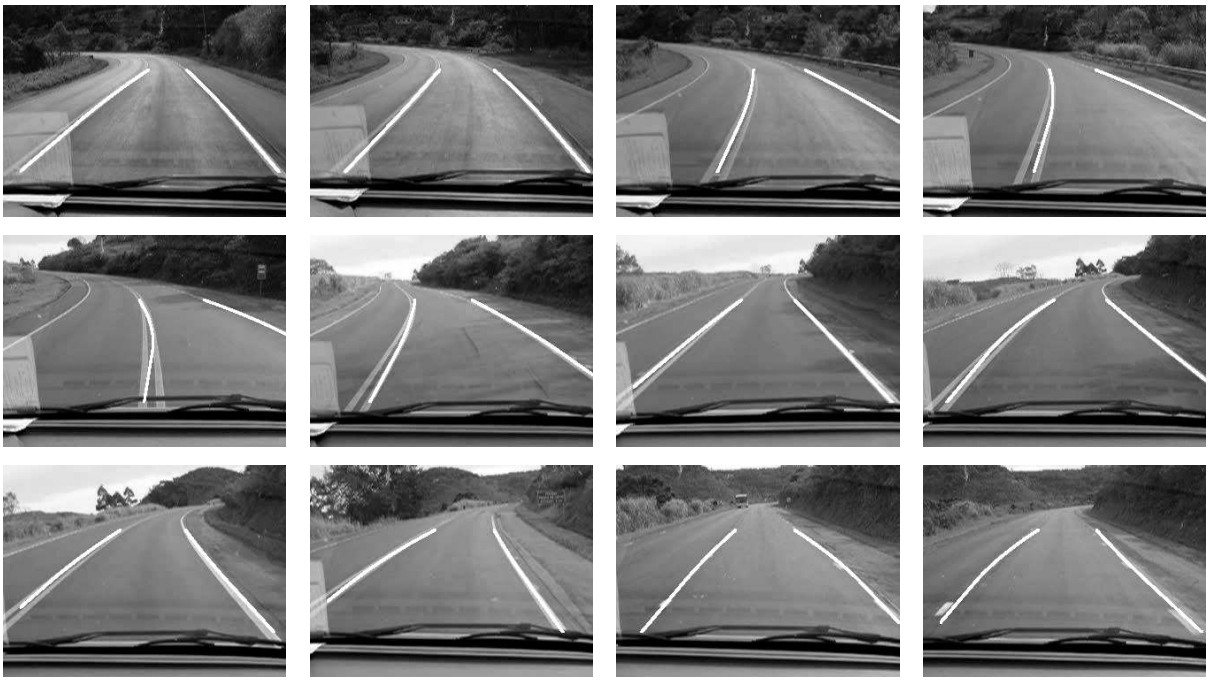

Figure 8. Twelve frames of the first video sequence.

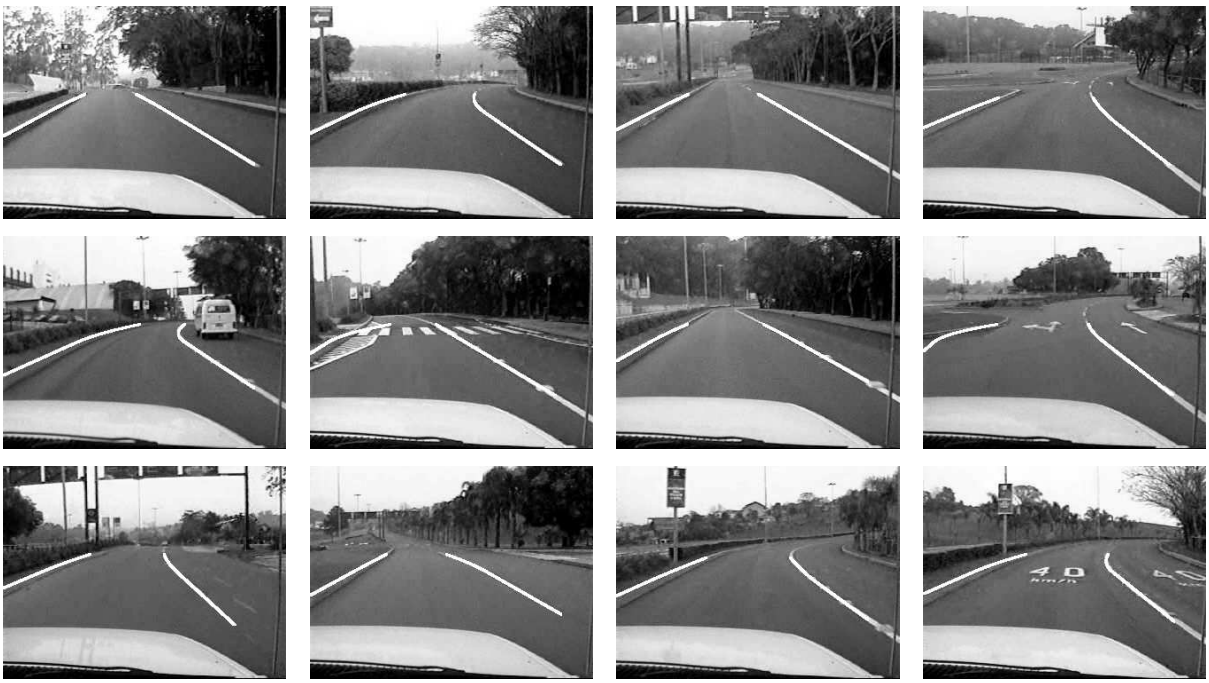

Figure 9. Twelve frames of the second video sequence.

[3] X. Yu, S. Beucher, and M. Bilodeau, "Road tracking, lane segmentation and obstacle recognition by a mathematical morphology," in *Proceedings of IEEE Intelligent Vehicles Symposium*, pp. 166–172, June 1992.

[4] S. Beucher and M. Bilodeau, "Road segmentation and obstacle detection by a fast watershed transformation," in *Pro-*

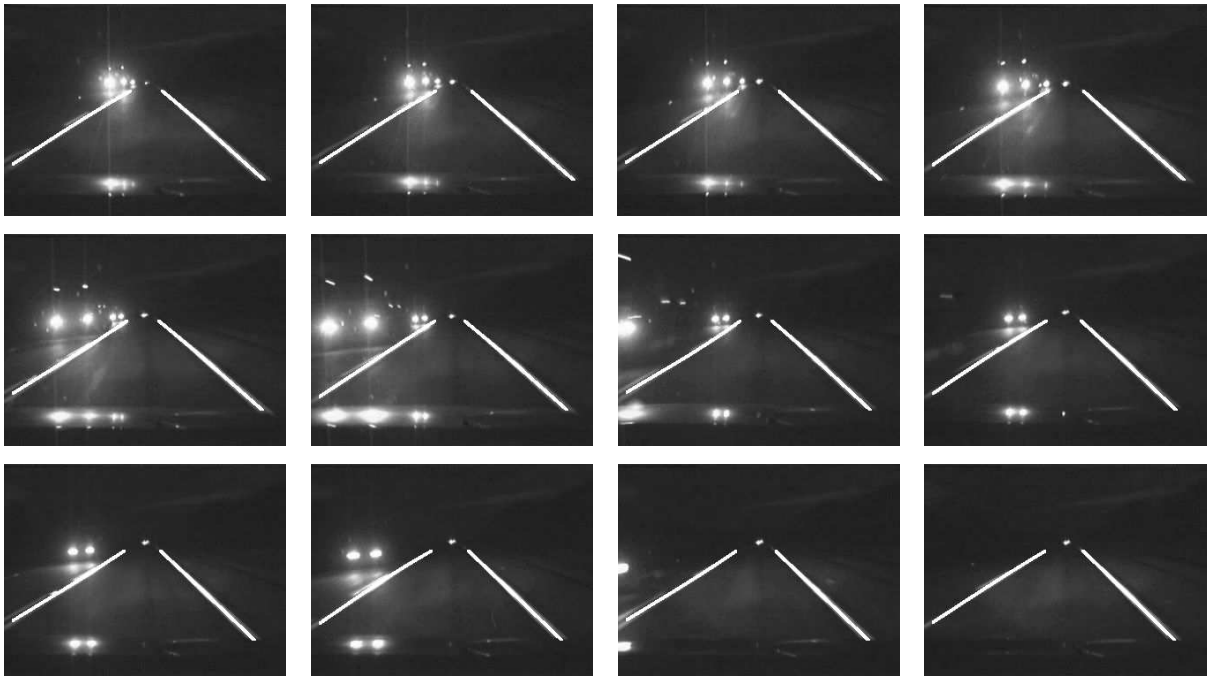

Figure 10. Twelve frames of the third video sequence.

ceedings of IEEE Intelligent Vehicles Symposium, pp. 296–301, October 1994.

[5] D. A. Pomerleau, "Ralph: Rapidly adapting lateral position handler," in *Proceedings of IEEE Intelligent Vehicles Symposium*, (Detroit, USA), pp. 506–511, 1995.

[6] M. Bertozzi and A. Broggi, "Real-time lane and obstacle detection on the gold system," in *Proceedings of IEEE Intelligent Vehicles Symposium*, pp. 213–218, 1996.

[7] M. Bertozzi and A. Broggi, "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, 1998.

[8] N. Apostoloff and A. Zelinsky, "Robust based lane tracking using multiple cues and particle filtering," in *Proceedings of IEEE Intelligent Vehicles Symposium*, (Columbus, OH), pp. 558–563, June 2003.

[9] W. Enkelmann, G. Struck, and J. Geisler, "Roma: A system for model-based analyis of road markings," in *Proceedings of IEEE Intelligent Vehicles Symposium*, (Detroit, USA), pp. 356–360, 1995.

[10] Y. Wang, D. Shen, and E. Teoh, "Lane detection using catmull-rom spline," in *Proceedings of IEEE Intelligent Vehicles Symposium*, (Stuttgart, Germany), pp. 51–57, 1998.

[11] Y. Wang, D. Shen, and E. Teoh, "Lane detection using spline model," *Pattern Recognition Letters*, vol. 21, pp. 677–689, June 2000.

[12] R. Risack, N. Mohler, and W. Enkelmann, "A video-based lane keeping assistant," in *Proceedings of IEEE Intelligent Vehicles Symposium*, (Dearborn, MI), pp. 506–511, October 2000.

[13] J. Park, J. Lee, and K. Jhang, "A lane-curve detection based on an lcf," *Pattern Recognition Letters*, vol. 24, pp. 2301–2313, October 2003.

[14] Y. Wang, E. Teoh, and D. Shen, "Lane detection and tracking using B-snake," *Image and Vision Computing*, vol. 22, pp. 269–280, April 2004.

[15] J. Lee, "A machine vision system for lane-departure detection," *Computer Vision and Image Understanding*, vol. 86, pp. 52–78, April 2002.

[16] J. W. Lee, C.-D. Kee, and U. K. Yi, "A new approach for lane departure identification," in *Proceedings of IEEE Intelligent Vehicles Symposium*, (Columbus, OH), pp. 100–105, June 2003.

[17] P. Hough, "Method and means for recognizing complex patterns," in *US Patent*, 1962.

[18] W. K. Pratt, *Digital Image Processing*. New York: John Wisley and Sons, 1991.

[19] F. O'Gorman and M. Clowes, "Finding picture edges through collinearity of feature points," *IEEE Transactions on Computers*, vol. 25, no. 4, pp. 449–456, 1976.

[20] J. Matas, C. Galambos, and J. Kittler, "Robust detection of lines using the progressive probabilistic hough transform," *Computer Vision and Image Understanding*, vol. 78, pp. 119–137, April 2000.