An Improved Linear-Parabolic Model for Lane Following and Curve Detection

Cláudio Rosito Jung and Christian Roberto Kelber
UNISINOS - Universidade do Vale do Rio dos Sinos
PIPCA - Programa Interdisciplinar de Pós-Graduação em Computação Aplicada
Av. UNISINOS, 950. São Leopoldo, RS, Brasil, 93022-000
crjung@unisinos.br, kelber@eletrica.unisinos.br

Abstract

In this paper, we propose a new model for lane tracking and curve detection. We use a linear-parabolic model for each lane boundary, and apply constraints to link both lane boundaries based on the expected geometry of the road. The parabolic part of the model, which fits the far field, is then used to analyze the geometry of the road ahead (straight, right curve or left curve), with applications in driver's assistance systems and road inspection. Experimental results indicate that introduced geometric constraints result in a more consistent fit if compared to the individual fitting of each lane boundary, and that the parabolic part of the model can be effectively used to keep the driver informed about the geometry of the road in front of him/her.

1 Introduction

Traffic related accidents pose severe social and economic problems. A report by the United Nations [9] indicated that 1.26 million people died worldwide due to traffic accidents in 2000, corresponding to an average toll of 3,000 fatalities a day.

Several researchers are developing new techniques for intelligent vehicles and/or intelligent roads, aiming to increase the safety of drivers and pedestrians [11]. A potentially low-cost solution relies on computer vision algorithms, through the installation of a video camera in the interior of the vehicle. In fact, the use of on-board cameras can be used for several driver's assistance functions, such as lane departure warning systems and obstacle detection [4].

In particular, this work deals with lane following, which is a fundamental problem in the development of driver's assistance systems. We propose an improvement to our linear-parabolic lane model [10], by imposing geometric constraints that relate left and right lane boundaries (in the original work, both lane boundaries are computed independently). We also use the second derivative of the parabolic part of the model to estimate incoming curves in the far field, indicating the driver if the road ahead is approximately straight, right-curved or left-curved.

The remainder of this paper is organized as follows. In Section 2, some existing lane detection and following techniques are reviewed. Section 3 briefly revises the linear-parabolic model, and also describes the proposed improvement to the model. Section 4 presents our approach for curve detection, and Section 5 contains experimental results. Finally, conclusions and ideas for future work are drawn 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 [12] 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 [19, 5] 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 [15, 2, 3] relies 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 [1] 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 [17, 16, 14, 18]. 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 tradeoff between accuracy of the fit and robustness with respect to image artifacts.

Guichard and Tarel [6] proposed a robust method for curve detection by combining perceptual grouping and a filter similar to Kalman's, with applications in autonomous driving. However, their technique does not explore temporal continuity, and presents a relatively high computational cost.

Jung and Kelber [10] used a linear-parabolic model for lane tracking. The linear part is used to fit lane borders in the near vision field, while the parabolic part fits the far field. Despite the good results obtained by this technique, lane boundaries are computed independently, and inconsistencies could arise (such as one lane indicating a positive curvature and the other indicating a negative one).

In this work, we improve the linear-parabolic model by imposing constraints based on the geometry of the road, and explore the curvature of the parabolic part of the model to detect incoming curves. Next, the original and improved versions of the linear-parabolic model are explained.

3 The Linear-Parabolic Model

3.1 Original Formulation

Let us assume that we have a camera installed in the interior of the vehicle, aligned with the central axis of the vehicle. The viewing area of the camera is divided in two regions (near and far fields), which are separated by a threshold x_m, as shown in Figure 1.

In [10], both lane boundaries (left and right) are approximated by independent linear-parabolic functions, that are composed by linear functions in the near field, and parabolic functions in the far field. The expression for the linear-parabolic model $f^k(x)$ can be written as[1] :

[1] In this work, we changed notation if compared to the original formulation in [10]

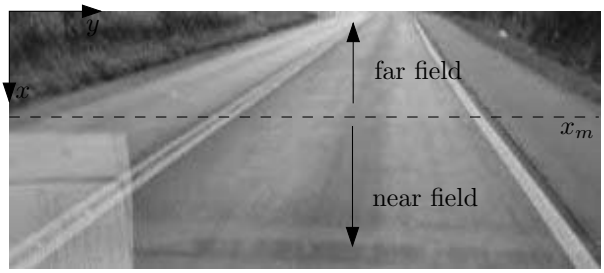

Figure 1. The initial frame of a video sequence, with our coordinate system and the definition of the near and far fields.

$$f^k(x) = \begin{cases} a^k + b^k(x - x_m), & \text{if } x > x_m \\ a^k + b^k(x - x_m) + c^k(x - x_m)^2, & \text{if } x \leq x_m \end{cases}, \quad (1)$$

where $k \in \{r, l\}$ denotes which lane boundary we are referring to (right or left). It should be noticed that f^k is a continuous and differentiable function, b^k represents the local orientation of the lane boundary, and c^k is related to its curvature in the far field of the image domain. To determine parameters (a^k, b^k and c^k) for the current frame, we compute image edges in a neighborhood of detected lanes in the prior frame, and minimize the following weighted square error:

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

where $(x_{n_i}^k, y_{n_i}^k)$, for $i = 1, ..., m$, denote the m coordinates of the non-zero pixels of the thresholded edge image in the near field, and $M_{n_i}^k$ the respective magnitudes, for $k \in \{r, l\}$. Analogously, $(x_{f_j}^k, y_{f_j}^k)$ and $M_{f_j}^k$, for $j = 1, ..., n$, represent the same characteristics for the n edge pixels in the far field.

It is easy to show that E^k is minimized when the following 3×3 linear system is solved:

$$(\mathbf{A}^k)^T \mathbf{W}^k \mathbf{A}^k \mathbf{c}^k = (\mathbf{A}^k)^T \mathbf{W}^k \mathbf{b}^k, \quad (3)$$

where

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

$$\mathbf{W}^k = \begin{bmatrix} M^k_{n_1} & & & & & & \\ & \ddots & & & & & \\ & & M^k_{n_m} & & & & \\ & & & M^k_{f_1} & & & \\ & & & & \ddots & & \\ & & & & & M^k_{f_n} \end{bmatrix},$$

$\mathbf{c}^k = \left[a^k, b^k, c^k\right]^T$ and $\mathbf{b}^k = \left[y^k_{n_1}, \ldots, y^k_{n_m}, y^k_{f_1}, \ldots, y^k_{f_n}\right]^T$.

In the original formulation, the procedure above is applied independently for each lane boundary (i.e. $k = l$ and $k = r$ lead to uncoupled linear systems). Next, we extend this approach and impose constraints based on the expected geometry of the road.

3.2 Improved Linear-Parabolic Model

It is clear that left and right lane boundaries must present some kind of coherence. For example, it is expected that the orthogonal distance between them at each point should be constant (in world coordinates). However, if camera parameters are not known, it is not easy to use this condition in image coordinates. In this work, we explore vanishing points to provide coherence between both lane boundaries, as explained next.

Guiducci [7, 8] worked on road models and automatic camera calibration. He assumed that for planar high speed roads (radius of curvature typically larger than 1000 m) and for vehicle heading directions forming a small angle with the road direction (typically smaller than 5°), then the x-coordinate of the vanishing points of the tangents to road borders in the near and far fields are the same, as shown in Figure 2.

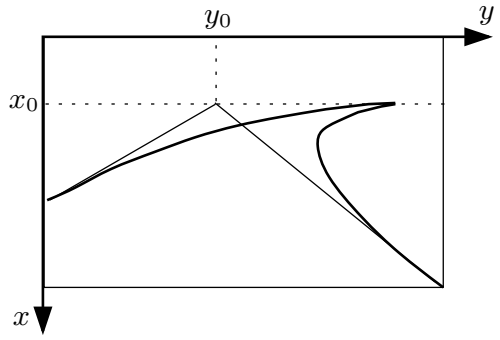

Figure 2. Projection of a road in image coordinates.

If x_0 denotes the common x-coordinate of the vanishing points, then the linear part of the left and right lane boundaries should meet at $x = x_0$, and the parabolic part of both lane boundaries should also meet at $x = x_0$. These conditions imply that:

$$\begin{cases} (a^r - a^l) + (b^r - b^l)\Delta x = 0 \\ (a^r - a^l) + (b^r - b^l)\Delta x + (c^r - c^l)\Delta x^2 = 0 \end{cases}, \quad (4)$$

where $\Delta x = x_0 - x_m$. We estimate x_0 dynamically, by finding the intersection between the linear parts of both lane boundaries at each frame. Our experimental results show that estimated values for x_0 typically vary within a range of 2 or 3 pixels. It should be noticed that Guiducci's assumptions were made for planar roads only. Hence, estimated vanishing points may present significant variations for non-planar roads.

Then, we include the restrictions of Equation (4) in the minimization errors E^k (given by Equation (2)), obtaining a new **global** error E_{g}:

$$E_{\text{g}} = E^l + E^r + w_{\text{lin}}\left(a^r - a^l + (b^r - b^l)\Delta x\right)^2 + w_{\text{par}}(a^r - a^l + (b^r - b^l)\Delta x + (c^r - c^l)\Delta x^2)^2, \quad (5)$$

where w_{lin} and w_{par} are weights associated with the first and second lines of Equation (4), respectively (when $w_{\text{lin}} = w_{\text{par}} = 0$, minimizing E_{g} is equivalent to detecting right and left lane boundaries independently). To minimize the global error E_{g}, we need to solve the following 6×6 system:

$$\mathbf{A}^T\mathbf{W}\mathbf{A}\mathbf{c} = \mathbf{A}^T\mathbf{W}\mathbf{b}, \quad (6)$$

where $\mathbf{A}$, $\mathbf{W}$, $\mathbf{c}$ and $\mathbf{b}$ can be represented as block matrices:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}^r & \\ & \mathbf{A}^l \\ \mathbf{L} & -\mathbf{L} \end{bmatrix}, \quad \mathbf{L} = \begin{bmatrix} 1 & \Delta x & 0 \\ 1 & \Delta x & \Delta x^2 \end{bmatrix}, \quad (7)$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}^r & & \\ & \mathbf{W}^l & \\ & & \mathbf{W}_g \end{bmatrix}, \ \mathbf{W}_g = \begin{bmatrix} w_{\text{lin}} & 0 \\ 0 & w_{\text{par}} \end{bmatrix}, \quad (8)$$

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}^r \\ \mathbf{c}^l \end{bmatrix}, \quad \text{and } \mathbf{b} = \begin{bmatrix} \mathbf{b}^r \\ \mathbf{b}^l \\ 0 \\ 0 \end{bmatrix}. \quad (9)$$

Furthermore, it is easy to show that matrix $\mathbf{A}^T\mathbf{W}\mathbf{A}$ can be written as:

$$\begin{bmatrix} (\mathbf{A}^r)^T\mathbf{W}^r\mathbf{A}^r + \mathbf{L}^T\mathbf{W}_g\mathbf{L} & -\mathbf{L}^T\mathbf{W}_g\mathbf{L} \\ -\mathbf{L}^T\mathbf{W}_g\mathbf{L} & (\mathbf{A}^l)^T\mathbf{W}^l\mathbf{A}^l + \mathbf{L}^T\mathbf{W}_g\mathbf{L} \end{bmatrix}.$$

The computation of $\mathbf{L}^T\mathbf{W}_g\mathbf{L}$ is extremely fast ($\mathbf{L}$ is a 2×3 matrix and $\mathbf{W}_g$ is a 2×2 matrix), and consequently the computational burden of solving the coupled system given by Equation (6) is practically the same as solving the two uncoupled systems given by Equation (3). It should be noticed that the proposed geometric constraints result in a

trade-off between the individual fitting of each lane boundary and the consistency of lane boundaries (i.e. the individual fitting of both lane boundaries may not seem as accurate as if they were applied independently, but they are consistent). Larger values of w_{lin} and w_{par} result in more lane dependency, while smaller values lead to independency. This behavior was expected, since other authors [4] had already noticed this trade-off between consistency and flexibility of lane models.

In this work, w_{lin} and w_{par} are chosen adaptively. If n_{lin} represents the total number of edge points and M_{lin} represent the mean edge magnitude in the near field (and n_{par}, M_{par} represent the same characteristics in the far field), then:

$$\begin{aligned} w_{\text{lin}} &= 0.05 M_{\text{lin}} n_{\text{lin}}, \\ w_{\text{par}} &= 0.05 M_{\text{par}} n_{\text{par}}, \end{aligned} \tag{10}$$

meaning that these weights are 5% of the number of edge pixels in the respective viewing area (near or far) multiplied by the mean edge magnitude in the corresponding region.

An example of the proposed improvement is illustrated in Figure 3. The result of the original formulation of the linear-parabolic model is shown in Figure 3(a), while the results of the modified model is shown in Figure 3(b). It can be noticed that edges produced by the vehicle in front of the camera affects the fitting process in the original formulation. On the other hand, the geometric constraints imposed in the modified version of the model provide consistent lane boundaries.

Figure 3. (a) Original formulation of the model. (b) Improved version.

Another situation that could result in wrong fitting results for the original formulation is shown in Figure 4. This Figure illustrates an exit to the road, which is indicated by a solid lane marking (while the right lane border is indicated by a dotted marking). Figures 4(a)-(b) show two frames of the video sequence near the exit, and detected lane boundaries using the original linear-parabolic model. It can be noticed that this model tends to follow the solid line, providing a misleading result. Figures 4(c)-(d) show the same two frames, but with the improved version of the model. The geometric consistency between both lane boundaries guarantees consistent lane boundaries.

4 Curve Detection

After applying the improved linear-parabolic model for lane following, we have the functions $f^r(x)$ and $f^l(x)$ at each frame of the video sequence. As mentioned before, coefficients c^r and c^l are related to the curvature of right and left lane boundaries in image coordinates (these information could be used to obtain actual curvature in world coordinates, if camera parameters were known). If $c^r(t)$ and $c^l(t)$ denote the curvature parameters of both lane boundaries at frame t, we compute a global curvature measure given by:

$$c(t) = c^r(t) + c^l(t). \tag{11}$$

In practice, measured curvatures c^r and c^l (and hence, $c(t)$) present high-frequency noise, due to irregularities of the pavement or bad conditions of road painting. On the other hand, changes in $c(t)$ due to the actual curvature of the road should occur progressively, indicating the use of a low-pass filter. For noise removal, we use a first order Chebyshev type I causal filter [13], with 15 dB of peak-to-peak ripple in the passband, and cutoff frequency $w_c = 0.1$, where $w = 1$ corresponds to half of the sample rate (in most cases, video sequences are acquired at 15 or 30 frames per seconds). The resulting filtered curvature $c_f(t)$ can be obtained by solving the following recursive equation:

$$c_f(t) - 0.9444 c_f(t-1) = 0.0278\,(c(t) + c(t-1)) \tag{12}$$

When the road in the far field is approximately straight, $c_f(t)$ should be close to zero; if there is a left turn in the far field, $c_f(t)$ should be negative; in a right turn, $c_f(t)$ should be positive. To detect such patterns, we define a threshold $T > 0$, and for each frame t the road ahead is classified as:

$$\begin{cases} \text{straight}, & \text{if } |c_f(t)| < T \\ \text{left turn}, & \text{if } c_f(t) < -T \\ \text{right turn}, & \text{if } c_f(t) > T \end{cases} \tag{13}$$

The choice of T clearly depends on camera parameters. However, for the camera setups used in this work, the value $T = 0.1$ produced good results for all experiments.

5 Experimental Results

In this section, we evaluate the performance of the improved linear-parabolic model for lane following and curve detection. All video sequences were captured/digitized with a resolution of 240×320 pixels.

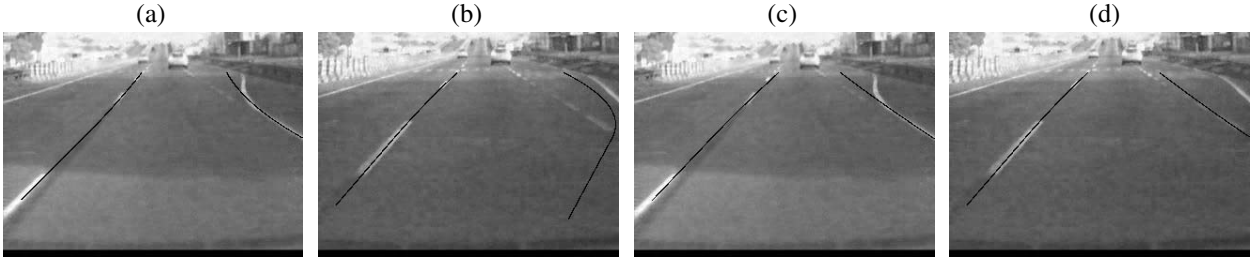

Figure 4. (a)-(b) Original formulation of the model. (c)-(d) Improved version.

Figure 5 illustrates some frames of a video sequence acquired close to sunset time with a digital camera set at video mode. This camera offers very poor illumination control, and some frames are over-illuminated while others are under-illuminated, as it can be noticed in Figures 5(a) and 5(b). Furthermore, this video sequence alternates frames in which the road is filled with shadows with frames in which the road was illuminated by direct sunlight. Despite all these image artifacts, our model correctly fits both lane boundaries.

In the second example, we consider a video sequence with 1450 frames illustrating a curvy road, acquired at 30 FPS. Figure 6(a) shows the filtered curvature $c_f(t)$ across time, and the regions of straight road, left curve and right curve[2]. According to Equation (13), the road was identified as an alternated sequence of left and right turns, connected by short segments of straight road. More specifically, the road is classified as a right turn in the following range of frames: 34-126, 505-629 and 915-1102; it is classified as left turn during frames: 238-433, 691-885 and 1140-1297; straight road in frames: 1-34, 126-238, 433-505, 629-691, 885-915, 1112-1140, and 1297-1450. Figure 6(b) illustrates some key frames of this video sequence, and visual inspection indicates that the method correctly indicated all road conditions. Furthermore, these frames illustrate that the improved linear-parabolic lane model is robust with respect to considerable illumination changes (for example, frame 629 is much darker than frame 915).

The third example consists of a video sequence with 1890 frames, also illustrating a curvy road and acquired at 30 FPS. Figure 7(a) shows $c_f(t)$ across time, and the geometry of the road ahead (straight,left or right)[3]. According to the proposed metric, a straight portion of the road is initially detected, followed by a left curve (frames 104-277), then a small straight portion (frames 277-353) connecting a right turn (frames 353-512). Frames 512-850 relate to a long portion of straight road, followed by a left turn (frames 850-1067) and a straight portion (frames 1067-1181). Frames 1362-1700 indicate a long right turn with a small linear segment in the middle (frames 1506-1546), and the road during frames 1700-1890 is identified as straight (with a very small right curve in frames 1794-1812). Some key frames of this video sequence are illustrated in Figure 7(b), and it can be observed that the estimated geometry of the road matches the actual video sequence. However, the small right curve detected in frames 1794-1812 can barely be noticed in the video sequence.

The computational burden of the proposed technique depends on the number of thresholded edge pixels needed to form matrices $\mathbf{A}$ and $\mathbf{W}$. However, an implementation of our technique in C++, running in a portable computer powered by a Pentium Centrino 1.7MHz processor and 1GB RAM memory, can process up to 80 frames per second of typical video sequences with resolution 320×240.

6 Discussion and Conclusions

In this work, we presented an improved linear-parabolic model for lane boundaries, by including constraints related to the geometry of the road. We also used the second derivative of the parabolic portion of the model (far field) to estimate the conditions of the road ahead (in terms of straight, left-curved or right-curved).

The geometric constraints introduced in this work generate more consistency between lane boundaries if compared to the original formulation [10]; on the other hand, detected lane boundaries may not be as accurate as if they were treated independently (specially in high-curvature turns). For curve detection, consistency is more important than local accuracy (it is important that both lane boundaries indicate the same curvature signal), and then the improved model is more adequate. In fact, the filtered curvature measure $c_f(t)$ can be effectively used to predict right or left portions of the road in the far field, and could be included in

[2]Ticks in the x-axis correspond to frames 34, 126, 238, 433, 505, 629, 691, 885, 915, 1102, 1140 and 1297, and were not included in the graphic due to lack of space

[3]Ticks in the x-axis correspond to frames 104, 277, 353, 512, 850, 1067, 1181, 1227, 1362, 1506, 1546, 1700, 1794 and 1812, and were not included in the graphic due to lack of space

Figure 5. Some frames of video sequence 1 and fitted lane boundaries.

a driver's assistance system. Other possible applications of our curve detection algorithm regard road inspection (e.g. to verify if traffic signs indicating curves are correctly placed along the road), or to forward feed a control system for autonomous guidance.

Future work will include a study on camera calibration to can obtain the actual road curvature in world coordinates, and comparisons of our curve detection procedure with ground truth (roads with known geometry). We also plan to discuss questions related to driving assistance, such as: How early must a curve be detected to inform the driver? Should the driver be warned about low-curvature incoming turns?

7 Acknowledgements

The authors would like to thank Brazilian agency CNPq for supporting this work.

References

[1] N. Apostoloff and A. Zelinsky. Robust based lane tracking using multiple cues and particle filtering. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 558–563, Columbus, OH, June 2003.

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

[3] M. Bertozzi and A. Broggi. GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Transactions on Image Processing*, 7(1):62–81, 1998.

[4] M. Bertozzi, A. Broggi, M. Cellario, A. Fascioli, P. Lombardi, and M. Porta. Artificial vision in road vehicles. *Proceedings of the IEEE*, 90(7):1258–1271, July 2002.

[5] S. Beucher and M. Bilodeau. Road segmentation and obstacle detection by a fast watershed transformation. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 296–301, October 1994.

[6] F. Guichard and J.-P. Tarel. Curve finder combining perceptual grouping and a kalman like fitting. In *Proceedings of IEEE International Conference on Computer Vision*, volume 2, pages 1003–1008, 1999.

[7] A. Guiducci. Parametric model of the perspective projection of a road with applications to lane keeping and 3d road reconstruction. *Computer Vision and Image Understanding*, 73(3):414–427, March 1999.

[8] A. Guiducci. Camera calibration for road applications. *Computer Vision and Image Understanding*, 79(2):250–266, August 2000.

[9] J. Hagen. Road safety crisis. *UN Chronicle Online Edition.*, 2004. `http://www.un.org/Pubs/chronicle/2004/webArticles/012204\_road\_safety.asp`.

[10] C. R. Jung and C. R. Kelber. A robust linear-parabolic model for lane following. In *Proceedings of SIBGRAPI*, pages 72–79, Curitiba, PR, October 2004.

[11] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis. A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21(4):359–381, April 2003.

[12] K. Kluge. Extracting road curvature and orientation from image edge points without perceptual grouping into features. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 109–114, October 1994.

[13] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1989.

[14] J. Park, J. Lee, and K. Jhang. A lane-curve detection based on an LCF. *Pattern Recognition Letters*, 24(14):2301–2313, October 2003.

[15] D. A. Pomerleau. RALPH: Rapidly adapting lateral position handler. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 506–511, Detroit, USA, 1995.

[16] R. Risack, N. Mohler, and W. Enkelmann. A video-based lane keeping assistant. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 506–511, Dearborn, MI, October 2000.

[17] Y. Wang, D. Shen, and E. Teoh. Lane detection using spline model. *Pattern Recognition Letters*, 21(6-7):677–689, June 2000.

[18] Y. Wang, E. Teoh, and D. Shen. Lane detection and tracking using B-snake. *Image and Vision Computing*, 22(4):269–280, April 2004.

[19] 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*, pages 166–172, June 1992.

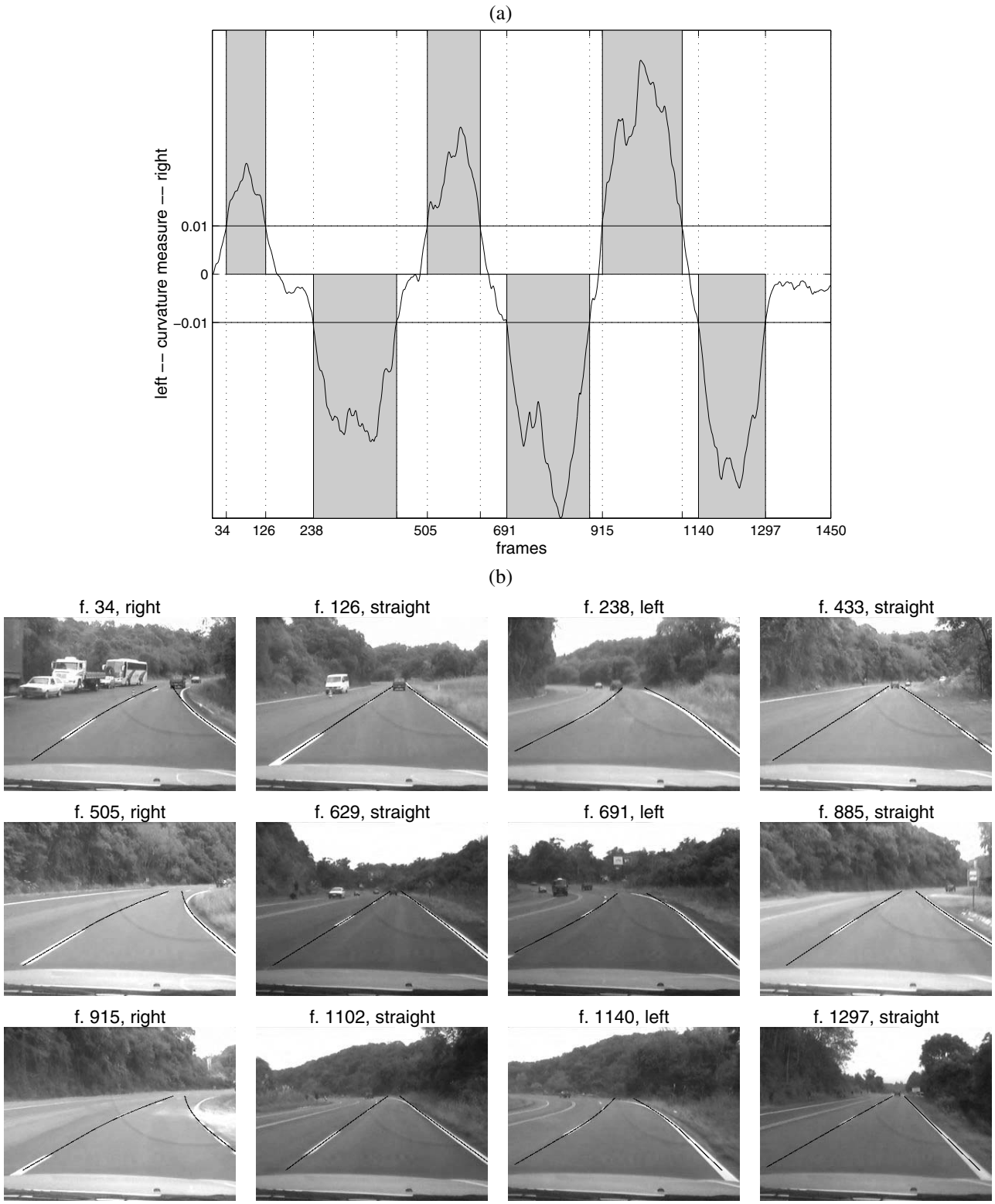

Figure 6. (a) Curvature measure for video sequence 2. (b) A selection of frames from this sequence.

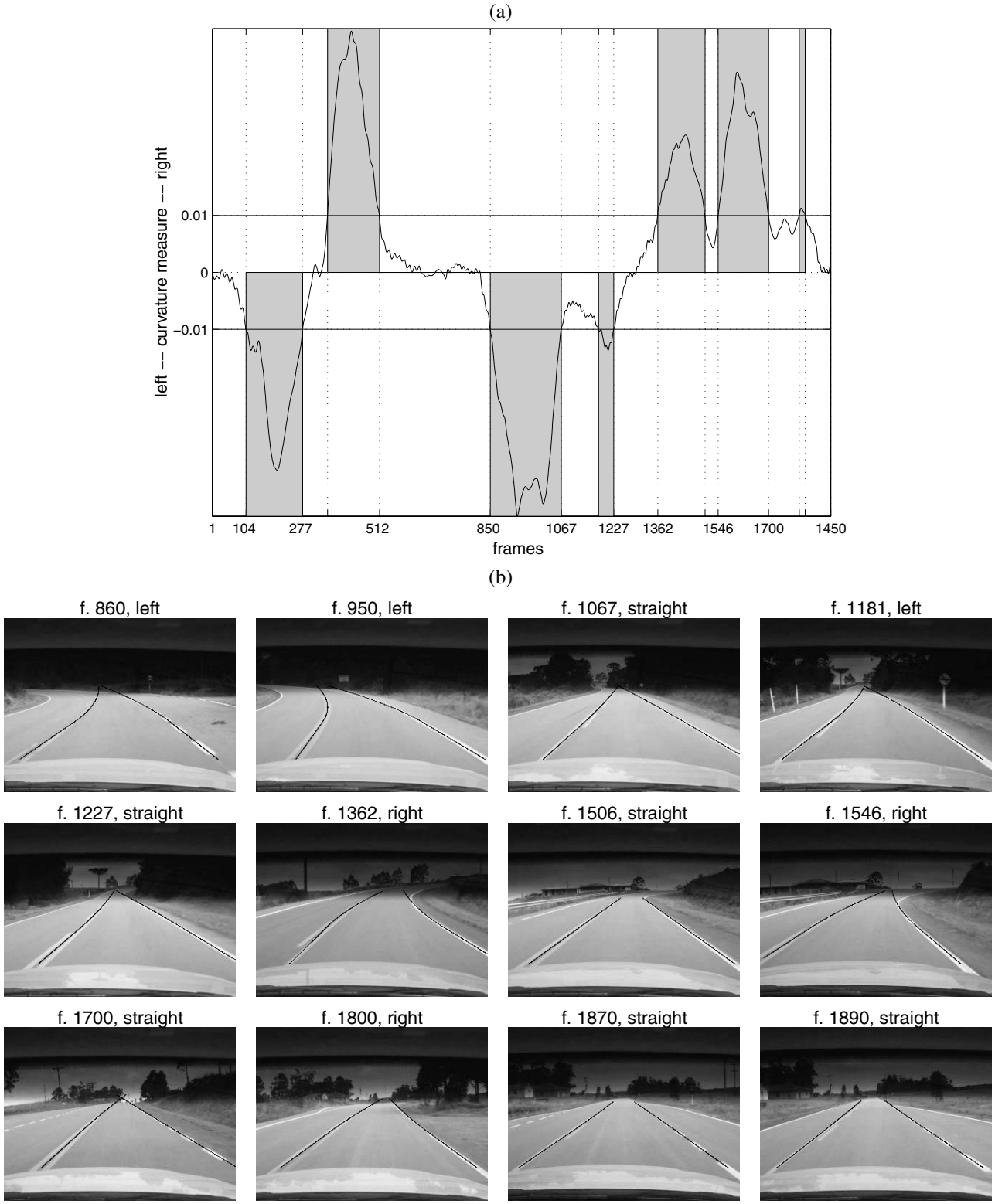

Figure 7. (a) Curvature measure for video sequence 3. (b) A selection of frames from this sequence.