Proceedings of the 8th International
IEEE Conference on Intelligent Transportation Systems
Vienna, Austria, September 13-16, 2005

Single camera lane detection and tracking

David SCHREIBER, Bram ALEFS, Markus CLABIAN

Abstract--**In this paper we present a method to detect and track straight lane boundaries, using a forward-looking single camera mounted on-board. The method proved to be robust, even under varying lighting conditions, and in the presence of heavy shadowing cast by vegetation, vehicles, bridges, etc. Moreover, false positive hardly occur. The lane markings are being continuously detected even when the vehicle is performing maneuvers such as excursion or lane change. The performance is achieved by a novel combination of methods, namely, by first finding vanishing point candidates for the lane boundaries, and later by the careful selection of line segments given a vanishing point, and finally by using a smart heuristics to select the whole set of four lane boundaries. Suppression of false positives is further achieved by tracking the vanishing point, and by constraining the lane width based on recent history.**

Index Terms—**Hough Transform, Hypotheses verification, Lane Detection and tracking, Vanishing point detection and tacking.**

I. INTRODUCTION

Lane detection and tracking is an important part of the ongoing research concerning vision-based intelligent vehicles. To name a few applications: To enhance the accuracy of tracking the leading vehicle for intelligent cruise control; to restrict the region of interest where obstacle are to be found; to provide a lane-departure warning system; to indicate cars cutting-in. The amount of work on the subject of lane detection and tracking is voluminous, thus we restrict our brief literature survey to methods that use a single camera on-board and that are the most relevant to our work. More survey details can be found, for example, in [1].

There is a natural division between methods that treat the lane boundaries as straight lines and methods that consider them as more general curves, especially parabolic curves, approximating circular arcs. Parabolic lane detection is done, notably, using a deformable template approach (the LOIS system [1]), and using Hough transform [2]. Both methods assume a parabolic curve in the 3D world. Detection in the image is done either by maximizing a Bayessian likelihood function [1] or by using a multi-resolution Hough transform [2]. The advantage of assuming a parabolic curve on the road plane rather than in the image plane is that in the Euclidean world, one can assume that all parabolic lane boundaries in the ground plane have approximately the same curvature and have parallel tangents at their x intercepts. Furthermore, it can be shown ([2]) that the 3D Hough space required for estimating the 3 parameters of the image curve can be decomposed into 2D and 1D space.

The main problem with such an approach is that usually the lane boundary seems to be almost linear at the lower part of the image, and there is only a relatively small area below the horizon where the lane seems to be curved. However, as the image curve is inversely proportional to the y coordinate [2], it is clear that the area near the horizon is more sensitive to noise. Therefore, such method seems to be appropriate only for specific data, such as shown in [1]-[2]. An alternative approach would be to detect the parabolic curves directly on the image plane. However, in such a case, one needs to solve simultaneously for three independent parameters.

In this paper, we concentrate solely on straight lane markings. In the special case where the radius of curvature of the lane is too small, and the lane boundary appears to be curved also in the near field, an additional method should be used. For example, ego-motion information could provide us the indication when to switch from straight- to non-straight-line boundary detection. It is striking, that although a great effort was invested in the literature to detect straight lane boundaries, almost no effort was done to detect first the vanishing point of the boundary lines (at least not in the single camera approach). In fact, a typical image consists of many lines which are parallel in Euclidean space – lanes, road and pavement boundaries, fence contours, even vehicle contours – all of which converge to the vanishing point.

In [3], detection of vanishing points was discussed. We find one of the approaches proposed there to be useful for the purpose of lane detection. A 2D Hough transform is used, where each data point (location and slope) votes directly for vanishing points. The difficulty with this method is that the vanishing point might be arbitrary far away, out of the image, requiring a large Hough space, or even might be located at infinity, in the case of lines which are parallel to the horizon. However, in the case of lane detection, one can safely exclude the case where the lanes are perpendicular to

Manuscript received January 15, 2005.

D. Schreiber, B. Alefs and M. Clabian are with the Advanced Computer Vision GmbH - ACV, Donau-City-Strasse 1 Vienna, Austria. (e-mail: david.schreiber@acv.ac.at).

the direction of motion and, using geometric considerations, one can also limit the region of interest where the vanishing point should be searched for. Under these circumstances, the method is fast and simple.

In a recent publication [4], detection of the vanishing point in the context of intelligent vehicles was done, for the purpose of detecting road borders in infrared images. This work is similar to ours, but differs from it in several important aspects. In [4], there is no method presented to detect the vanishing points. It is assumed that there is only one vanishing point in the image, and that its vertical position is constant (no oscillations of the camera). Given a vanishing point, and allowing only a small deviation in its location, the two border lines of the road are detected via a 2D Hough transform in which data points vote for lines. Tracking is done by allowing a small change in the orientation of the lines in the next frame, and the deviation of the vanishing point is accounted for by computing the pair of lines for each pixel in the 3x3 neighbourhood of the vanishing point.

In contrast, we present a novel and more general approach as follows. We detect the vanishing point, allowing large deviations in its location from frame to frame, due to pitch and yaw change of the ego-car relative to the road. Furthermore, we do not assume that the image data corresponds only to one vanishing point. As we are searching for lane markings rather than road borders, we are looking for four rather than two lines, and we employ a smart heuristics to detect these four boundary lines, in the presence of other competing lines and in the presence of other vanishing points. In addition, given a vanishing point, we manage to detect the lane boundaries using 1D rather than 2D Hough transform.

We have built a method that is robust, fast and simple on one hand, while being general enough on the other hand. We do not assume that the car is always moving more or less in the centre of the lane, where its optical centre is parallel to the lane boundaries. Rather, we have assumed only some general and necessary assumptions, as follows: (a) The road is planar. (b) The lane markings are straight lines, parallel to each other in Euclidean space. (c) Lane boundary cannot be perpendicular to the direction of motion, i.e. to appear as horizontal lines in the image. (d) The lane markings are brighter than the road.

II. Definitions and Formulas

The images we have used are of size of 400x768 pixels, taken with a horizontal field of view of 37°. As the alignment of the chip relative to the road was not perfect (about 1° pitch and yaw), each image was rectified by software. Some of the images were captured on a bigger chip and then asymmetrically clipped. Thus, on some of the images shown, the vertical position of the optical centre is not in the middle of the image.

The drawing in Fig. 1 depicts a camera travelling within a lane, as seen from above in Euclidean space. The lane is marked on the road by left and right white bands, or *lane markings*. L and R denote the distance measured from the optical axis to the middle of the left and right lane markings, respectively. ΔL and ΔR represent the width of the left and right lane markings, respectively. Lane detection means detecting the *lane boundary*, i.e. bounding the left and right lane markings, each by two line segments, as shown in Fig. 2. Going from left to right in the image, we denote these four lane boundaries as *LL* (left-lane-left-line), *LR* (left-lane-right-line), *RL* (right-lane-left-line) and *RR* (right-lane-right-line).

We define the 3D camera coordinate frame, such that the Z axis points forward, in the direction of motion, while X and Y are in the image plane, where the origin coincides with the optical centre of the camera. In the image plane, the direction of the x coordinate should be inverted. Therefore, we have the following perspective projection relation

$$x = -f\frac{X}{Z}, \quad y = f\frac{Y}{Z} \tag{1}$$

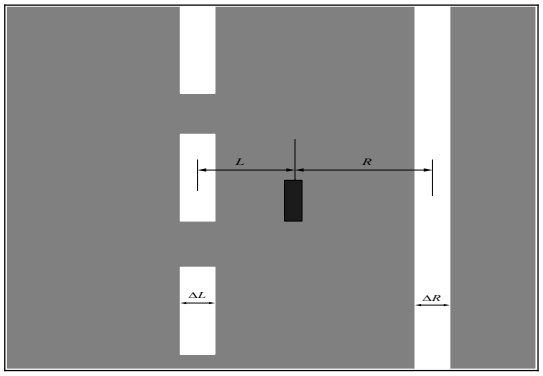

Fig. 1: Definition of lane width and lane marking width.

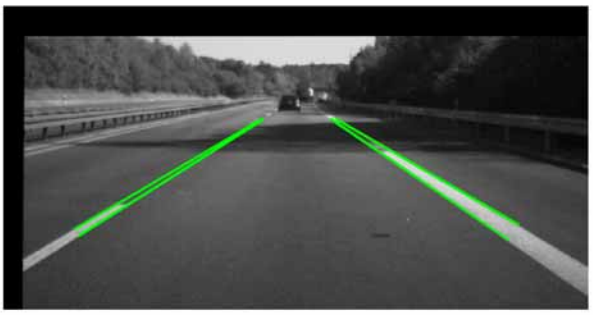

Fig. 2: Detection of four lane boundaries.

Let us assume that the road plane is not fully parallel to the optical axis but rather tilted with angle θ (pitch angle). In addition, assume that the lane markings (which are parallel

to each other in 3D), are rotated relative to the optical axis and that the amount of rotation is ϕ (yaw angle). In this case it is not difficult to show that the equation of a straight lane boundary *in the image* is given by (2)

$$y = \frac{-H\cos(\phi)}{\cos(\theta)X - \sin(\phi)\sin(\theta)H}x + \frac{f(-\sin(\theta)X - \sin(\phi)\cos(\theta)H)}{\cos(\theta)X - \sin(\phi)\sin(\theta)H}$$

where (X, Y, Z) are the coordinates in the un-rotated camera frame, (x, y) are the image coordinates, H is the height of the optical centre from the road plane, and the 4 different lane boundaries correspond to different values of X. It can be further shown that the intersection point of lines defined by (2) is given by

$$(x_0, y_0) = -f(\tan(\phi)/\cos(\theta), \tan(\theta),) \quad (3)$$

Thus, all parallel lines lying on our θ-tilted road plane and having slope $\tan(\phi)$ will converge to the vanishing point defined in (3).

Given the location of the vanishing point and the slopes of the four boundaries of the lane markings in the image, designated by a_{LL}, a_{LR}, a_{RL} and a_{RR}, respectively, it is possible to compute from (2) the 3D lateral Euclidean distance of the lane markings from the optical axes of the camera: (4)

$$L = -H\frac{\cos(\phi) - a_L\sin(\phi)\sin(\theta)}{a_L\cos(\theta)},\ R = H\frac{\cos(\phi) - a_R\sin(\phi)\sin(\theta)}{a_R\cos(\theta)}$$

where a_L is the average of a_{LL} and a_{LR}, and a_R is the average of a_{RL} and a_{RR}. In addition, we would like to estimate the error involved in the computation of L and R. Substituting the relation $a = \tan(\alpha)$ in (4) the following error estimate can be derived: (5)

$$dL = \left| H\frac{\cos(\phi)}{\cos(\theta)}\frac{1}{\sin^2(\alpha_L)} \right| d\alpha,\ dR = \left| H\frac{\cos(\phi)}{\cos(\theta)}\frac{1}{\sin^2(\alpha_R)} \right| d\alpha$$

Assuming $d\alpha = 1°$, we conclude that for angle bigger than 30°, the measurement error is less than 10 centimetres. In fact, this is usually the case, unless under unlikely driving conditions. Similarly, one can extract expressions for the width of the lane markings, ΔL and ΔR, as well as for the corresponding errors, $d\Delta L$ and $d\Delta R$.

III. Detecting vanishing points

Our lane detection method is based on edge information. We use the Canny edge detector [5]. To have accurate orientation measurement, we have used the 5x5 moment-based method [6]. Vanishing points are detected using a 2D Hough transform [7], where we vote directly for the location of the vanishing point. Usually more than one vanishing point will be detected in the image, only one of them corresponds to the lane boundaries. Assuming a vanishing point (x_0, y_0), a data point (x_i, y_i) and the corresponding measured gradient $\tan(\alpha_i)$, the following line segment is defined:

$$(y_i - y_0)\cos(\alpha_i) = (x_i - x_0)\sin(\alpha_i) \quad (6)$$

In case that a priori knowledge regarding the location of (x_0, y_0) is given (see section VI.), the Hough space is restricted to a relatively small portion of the image. We vote for all Hough cells with which the line-segment (6) intersects. We have used a weighted two-resolution Hough transform.

IV. Detecting straight line segments

The detection of the straight line segments, given a vanishing point, is more robust than the case where a vanishing point is not given, as the location of the vanishing point restricts the interpretation of data points in terms of straight lines. Given the vanishing point (x_0, y_0), a data point (x_i, y_i) votes for the following line slope

$$\tan(\alpha_i) = (y_i - y_0)/(x_i - x_0) \quad (7)$$

We use a 1D Hough transform to vote for the slopes of the lane boundaries. At this stage it is crucial to introduce error handling. Suppose that the perpendicular distance from a data point to the line on which it is supposed to lie is bounded by d pixels. Then, the error of the slope, due to this displacement error is bounded by $\Delta\alpha_i = \text{atan}(d / r_i)$, where r_i is the radial distance from the vanishing point. Given a bound on the slope error, $\Delta\alpha_i$, we vote for all slope values in the interval $[\alpha_i - \Delta\alpha_i, \alpha_i + \Delta\alpha_i]$. Fig. 3 shows an image, with the relevant data pixels marked in blue, and the corresponding vanishing point in green. The resulting 1D Hough space is shown in Fig. 4. The four peaks in the Hough space corresponding to the lane boundaries of Fig. 3 are marked as LL, LR, RL and RR. When computing the orientation of the line segment defined by data pixel (x_i, y_i) and the vanishing point (x_0, y_0), by means of (7), the sign of the resulting angle is adapted to fit the measured gradient. This sign adaptation helps to separate between the four peaks of the lane boundaries in the 1D

Hough space. Finally, given a peak in the Hough space (orientation), and given its support pixels, we improve the orientation of the line using a robust least-squares fit.

Fig. 3: Data pixels used for voting.

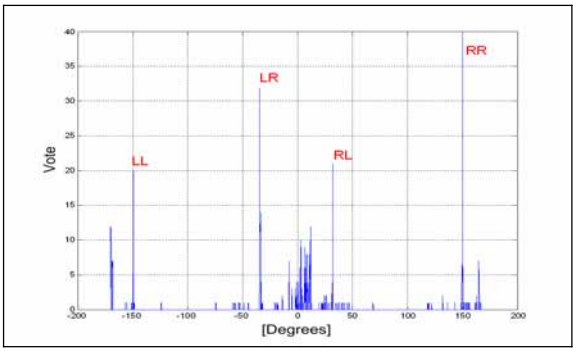

Fig. 4: The result of voting with the data pixels shown in the figure 3.

V. Generating and Testing Lane-Boundary Hypotheses

Having detected all line segments in the image, and given a vanishing point (x_0, y_0), we use a heuristics, explained below, to detect the four lane boundaries, denoted by their slopes, α_i, i=1,…,4. A set composed of (x_0, y_0) and $\{\alpha_i\}$ will be denoted a *hypothesis*.

A. Finding LL, LR, RL and RR candidates

The line segment defined by the vanishing point and by the mid-bottom of the image divides the image into left and right lane boundaries. The orientation of this line defines the *partition angle* β. Given a list of line slopes, $\{\alpha_i\}$, found from the histogram such as in Fig. 4, we choose LL, LR, RL and RR candidates according to the following formula,

$$
\begin{aligned}
LL &= \{\alpha_i \mid (\alpha_i \le \beta + d\beta) \wedge (PerpGrad_i > 0)\} \\
LR &= \{\alpha_i \mid (\alpha_i \le \beta + d\beta) \wedge (PerpGrad_i < 0)\} \\
RL &= \{\alpha_i \mid (\alpha_i \ge \beta - d\beta) \wedge (PerpGrad_i > 0)\} \\
LR &= \{\alpha_i \mid (\alpha_i \ge \beta - d\beta) \wedge (PerpGrad_i < 0)\}
\end{aligned}
\tag{8}
$$

where we have used $d\beta = 1°$, and where PerpGrad denotes the gradient in a direction perpendicular to the segment, averaged over the support data.

B. Finding left and right pairs

GGiven LL, LR, RL and RR candidates, we group them into left and right pair candidates, (LL, LR), and (RL, RR). For example, finding all left pairs is done as follows. We try to match all combinations of LL and RL pairs. For each such pair (α_i, α_j), we compute the total width of the lane marking, *LMwidth*, and the correponding width error, *dLMwdth*. A left pair $(\alpha_i \in LL,\ \alpha_j \in LR)$ is valid if the following condition holds true:

$$
\begin{aligned}
&(\alpha_j > \alpha_i) \\
&\wedge\ (LMwidth \ge minLMwidth - dLMwidth) \\
&\wedge\ (LMwidth \le maxLMwidth + dLMwidth)
\end{aligned}
\tag{9}
$$

were the minimal and maximal width of one lane marking are denoted as as minLMwidth and maxLMwidth, respectively. The search for right-pairs is done similarly.

C. Finding a hypothesis

To get a list of hypotheses, we try to match all possible left and right pairs. Given left and right pairs, *(LL, LR)* and *(RL, RR),* we compute the average slopes for each pair, a_L and a_R. We then compute the Euclidean distance to the left and right lane markings, R and L, using (4). Denoting *W=L+R*, the width error as *dW*, and the thresholds for the minimal and maximal total lane width as *minW* and *maxW*, a valid hypothesis is obtained if the following hold true,

$$minW - dW \le W \le maxW + dW \tag{10}$$

where *minW* and *maxW* are set in accordance with [8].

D. Grading a hypothesis

Each valid hypothesis receives a grade. The grade is first computed separately for the left and right pairs, and then the two sub-grades are combined. A grade of each pair is composed of the following five components, based on both edge and grey-level information.

Mean support-size of a pair: The support-size of a line is the number of data pixels whose perpendicular distance from the line is less than or equal to 1.5 pixels.

Mean perpendicular gradient of a pair: Given line orientation and the support data, we compute the mean grey-level-based gradient in a direction which is perpendicular to the line.

Similarity of a pair: We test for similarity between the two lines comprising the pair, in terms of the distribution of their data support.

Mean continuity of a pair: In [9], the role of higher-order statistics in the verification of hypotheses was discussed. In our case it means that we favour a line that matches the data by a small number of large pieces over a line that matches the data by a large number of short unconnected pieces.

Mean lateral accuracy: Finally, the last criterion is concerned with the lateral accuracy, i.e. the perpendicular distance of the support pixels from the line.

VI. The Role of History – Pruning Hypotheses and Tracking

The purpose of tracking is twofold: First, detection of the lane in previous frames should limit the search region for the lane in the current frame, thus reducing running time. Second, the information from previous frames serves as a consistency measure of the results of the current frame. This helps to prune wrong hypotheses. The geometric information the algorithm passed from frame to frame is the location of the vanishing point as well as the total width of the lane.

A. Tracking the Vanishing point

Based on our video sequences, we have set a bound of 0.7° (14 pixels) on the instantaneous pitch angle. In order to estimate the bound on the yaw angle, we have used available data concerning the relation between the maximal possible velocity V and the minimal radius of curvature R [8]. Using also the relation between the rate of change (angular velocity) of the yaw angle in circular motion $d\phi / dt = V / R$, we estimate the maximal yaw change between 2 frames as 2.9° (60 pixels) for the horizontal position of the vanishing point. Currently our system cannot deal with lanes of such a small curvature, as the lane markings in those cases are circular arcs over the entire image. In conclusion, we have used the value of 18 pixels to limit the rate of change of both horizontal and vertical location of the vanishing point. This limitation of the search region reduces the running time, as well as the number of detected vanishing points.

B. Constraining the lane width

The width of the lane, $R + L$, is a parameter which does not change over time as much as the vanishing point does. Therefore, we consider the average total lane width over a short history. Namely, the total lane width in the next image should be within 10% of the mean history value. This constraint significantly helps to prune hypotheses, where some of the lines do not correspond to true lane markings.

C. Lane crossing

Fig. 6 shows a sequence consisting of 400 frames on a highway. The red curve represents the distance from the optical centre to the middle of the left lane marking, while the green one represents the distance to the right lane marking. The black curve above describes the sum of the two, i.e. the total lane width, all measured in meters. The few gaps in the data correspond to cases where the visibility of the lane markings was poor (no effort was made at this point to fill the gaps using ego-motion information). Nevertheless, the lane crossing phenomena shown in Fig. 6 is very convincing. The lane change took roughly 50 frames. During this process, as well as during the whole sequence, the total lane width was kept constant within the measurement error.

Computing the average total lane width from the whole sequence, we obtain a width of 3.84 meters, and a standard deviation of 0.08 meters, in accordance with known values for highways' lane width [8]. Fig. 7 shows some representative frames of the lane crossing.

VII. Results

The images of Fig. 8 demonstrate the success of the method under varying difficult conditions. Fig. 8.a. shows a curved lane on a bridge, not fully planar itself. Nevertheless, reasonable lane detection is achieved. In Fig. 8.b, left lane marking is not available. Boundary of pavement was detected instead. Finally, Fig. 8.c. shows detection under heavy shadowing cast by trees and truck.

VIII. Conclusions and Future Work

A simple yet robust method was present to detect and track straight lane markings in images, using a forward-looking single camera. In case of curved lane boundaries, where fitting straight lines hardly makes sense, a different method should be employed. Indication when to switch between the linear and non-linear methods could be obtained from the ego-motion information, namely the value of the yaw angle. Further future work includes cases where one of the lane marking is missing, and instead there is a pavement boundary or a strip of a structured road. In addition, augmenting geometric information with ego-motion information, should significantly improve performance.

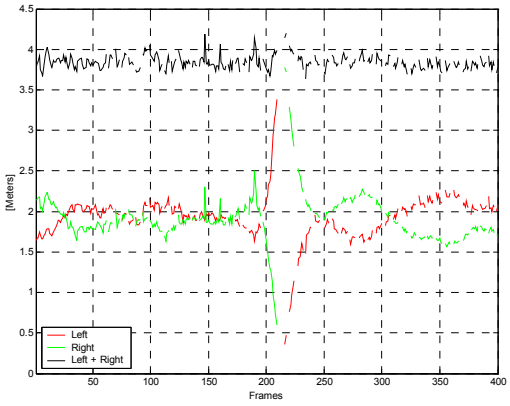

Fig. 6: Euclidean distances measured over 400 frames. The red(green) curve represents the distance to the left(right) lane marking. The black curve represents the sum of the two, i.e. the total width of the lane.

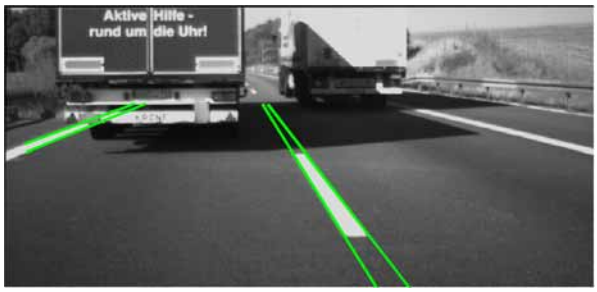

Fig.7.a: Frame 207 from lane crossing sequence

Fig.7.b: Frame 214 from lane crossing sequence

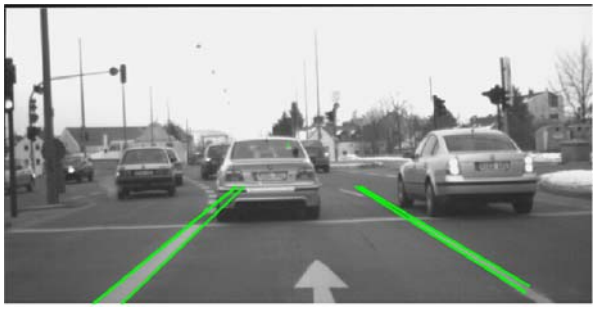

Fig. 8.a: Detection of a curved lane on a non-planar bridge.

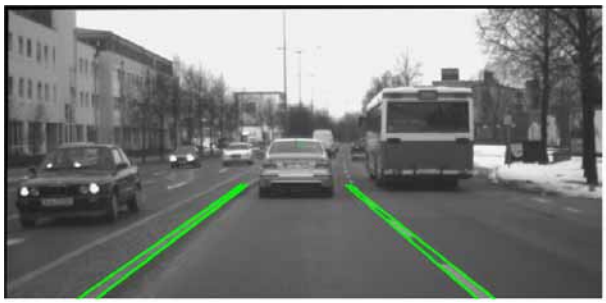

Fig. 8.b: Left lane marking is not available. Boundary of pavement was detected instead.

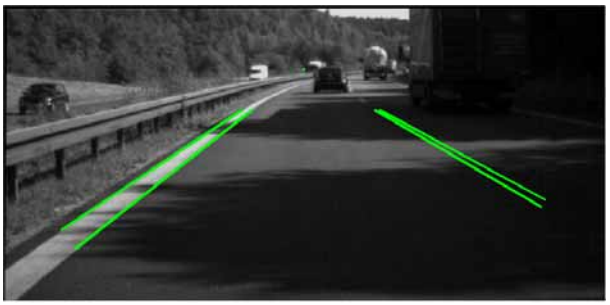

Fig. 8.c**:** Detection under heavy shadowing cast by trees and truck.

REFERENCES

[1] C. Kreucher, S. Lakshmanan and K. Kluge, "A Driver Warning System Based on the LOIS Lane Detection Algorithm", IEEE International Conference on Intelligent Vehicles, Stuttgart, Germany, October 1998.

[2] B. Yu and A. Jain, "Lane Boundary Detection Using a Multiresolution Hough Transform", International Conference on Image Processing (ICIP '97) , October 26 - 29, 1997 , Washington, DC, pp. 748-751.

[3] V. Cantoni, L. Lombardi, M. Porta and N. Sicard, "Vanishing Point Detection: Representation analysis and New Appraoches ", 2001 IEEE proceedings of the 11th international conference on image analysis and processing (8ICIAP 2001).

[4] B. Fardi and G. Wanielik, "Hough Transformation Based Approach For Road Border Detection in Infrared Images", 2004 IEEE intelligent vehicles symposium, university of parma, parma , italy, june 14-17, 2004.

[5] J. Canny, "Computational Approach to Edge Detection", PAMI, Vol.8(6), pp.679-698, 1986

[6] E.P.Lyvers and O.R.Mitchell, "Precision Edge Contrast and Orientation Estimation", IEEE Trans on Pattern Analysis and Machine Intelligence, Vol. 10(6), Nov 1988, pp. 927-937.

[7] R. O. Duda and P. E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures". Comm .1972, ACM 15(1): 11-15.

[8] http://www.isb.rwth-aachen.de/lehre/Umdrucke/VP-Grundlagen-Kap04-WS03-04.pdf

[9] T. M. Breuel, "High-Order Statistics in Visual Object Recognition", Memo #93-02, June 1993, IDIAP.