Vision-based Lane Analysis: Exploration of Issues and Approaches for Embedded Realization

R. K. Satzoda and Mohan M. Trivedi
Computer Vision and Robotics Research Laboratory
University of California, San Diego
rsatzoda@eng.ucsd.edu, mtrivedi@ucsd.edu

Abstract

Lane feature extraction is one of the key computational steps in lane analysis systems. In this paper, we propose a lane feature extraction method, which enables different configurations of embedded solutions that address both accuracy and embedded systems' constraints. The proposed lane feature extraction process is evaluated in detail using real world lane data, to explore its effectiveness for embedded realization and adaptability to varying contextual information like lane types and environmental conditions.

1. Role of Lane Analysis in IDAS

Intelligent driver assistance systems (IDAS) are increasingly becoming a part of modern automobiles. Reliable and trustworthy driver assistance systems require accurate and efficient means for capturing states of vehicle surroundings, vehicle dynamics as well as state of the driver in a holistic manner [12].

Among the different modules for active driver safety framework, lane analysis using monocular cameras contributes to its efficiency in multiple ways. Firstly, lane analysis, i.e. lane estimation and tracking, aids in localizing the ego-vehicle motion, which is the one of the very first and primary steps in most IDAS like lane departure warning (LDW), lane change assistance etc. [12, 7]. Next, lane analysis is also shown to aid other vehicle surround analysis modules. For example in [10] lanes are used to detect vehicles more robustly because vehicles are assumed to be localized to their ego lanes. Similarly lane detection is shown to play a significant role in predicting driver intentions before lane changes occur [12, 7] etc.

There are a number of computer-vision-based lane analysis methods reported in literature as shown in recent works [6, 2, 4, 3, 1]. Most of these works address the robustness of the vision algorithms in different road scenarios. However, as pointed by Stein in [11] titled, "The challenge of putting vision algorithms into a car", there is a need to explore lane analysis approaches for embedded realization. Attempts have been made to realize embedded solutions for lane estimation and tracking [9, 5] etc. but as indicated in [5], most of them have been architectural translations of *some parts* of existing lane detection algorithms. In this paper, we propose a lane feature extraction method that addresses some of these issues related to embedded realization.

2. Lane Analysis & Embedded Vision

Different variants of lane analysis techniques have been proposed in literature such as [6, 2, 3] etc. A detailed survey of lane analysis methods is presented in [6] and [1]. An effective lane analysis method [6, 1] comprises of three main steps: (1) lane feature extraction, (2) outlier removal or post processing, and (3) lane tracking. Pixel level filtering operations like steerable filters etc. are applied on the entire image or regions of interest (usually the lower half of the input image) to extract lane features. A further post processing and outlier removal is performed using techniques like RANSAC [2], Hough transform [8] etc. in order to improve the robustness. Inverse perspective mapping (IPM) of the input image is also performed to transform the input image into world coordinate system (WCS) [6]. In addition, lane models and vehicle dynamics from CAN data are used to track lanes across time using Kalman filtering etc.

Considering that IDAS are implemented on battery powered embedded platforms inside a car, attempts have been made to implement lane detection systems on embedded platforms in [9, 5] etc. However, as indicated previously, most of these are partial systems with the exception of the full system implemented in [5]. For example, in [9] lane detection is implemented using steerable filters on an FPGA platform. However, this is only the lane feature extraction module of a comprehensive and robust lane analysis method called VioLET in [6]. One of the very few complete lane analysis systems is reported in [5], which includes a pipelined architecture for lane feature extraction,

978-0-7695-4990-3/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPRW.2013.91

lane model fitting and tracking, and implemented on an FPGA platform using DSP48 cores of Spartan FPGAs.

In [11], different kinds of embedded constraints are elaborated that decide the feasibility of employing a computer vision task in a car, which is an excellent example of a complex embedded system. These constraints bring together the requirements from two different disciplines - computer vision and embedded engineering. In other words, robustness is the key performance index for a computer vision algorithm but real-time operation, limited hardware resource utilization, energy efficiency are the key metrics for embedded realization. With the two together in active driver safety framework, the reliability and dependability of computer vision algorithms that run on resource constrained computing platforms is another challenge that needs to be satisfied.

3. Feature Extraction Method for Context-aware Lane Analysis

Lane feature extraction is one of the key steps in real-time lane analysis, which includes both lane estimation and tracking. The robustness of the entire lane analysis system depends directly on reliable lane features that need to be extracted from the road scene. This also implies that there is a direct relationship between the efficiency of lane feature extraction process and the robustness of the system. Adding more computer vision algorithms for lane feature extraction in order to improve robustness can directly impact the efficiency of the system. Also, the robustness, and hence the efficiency, of this feature extraction step is dependent on vehicle surround conditions like road types, weather conditions like fog, wet roads etc., environmental changes in road scene like shadows, road surface etc., and the availability of other data sources like road maps etc. These factors - application requirements (eg. safety critical systems demand higher robustness), environmental and weather conditions, road information, lane types etc. constitute the context in which lane analysis is to be performed. Therefore, this context plays an important role in the robustness and efficiency of the lane feature extraction step. A detailed exploration of the lane feature extraction step that can cater to such contextual information is worthy of further study.

We will now elaborate the proposed lane feature extraction method that enables such a study and exploration. The proposed method will then be studied in detail to examine its robustness and its relationship with the different embedded systems' constraints. In the proposed method, an input image I is first transformed into WCS image I_W using IPM. In the IPM image, we select n_B scan bands that are sampled at predetermined coordinates in I_W in the direction of the road. The lane features are extracted from these scan bands instead of scanning the entire image. In order to find the lane features, each scan band B_i is filtered using a 5×5 steerable filter kernel H_π corresponding to $180°$, which highlights the intensity variations in the horizontal x direction of the image, i.e. $F_{B_i} = B_i \otimes H_\pi$. Referring to Fig. 1, we have $n_B = 2$ and after applying the steerable filter on the top band, we get the F_{B_i} as the filtered output. The lane edges form pairs of negative and positive filtering outputs, i.e. the the dark to light transition edge will give negative convolution outputs and vice versa for the light to dark transition edge. Two thresholds, a postive T_+ and a negative T_- threshold, are applied on the filtered output as shown Fig. 1 resulting in the binary maps E_+ and E_- respectively.

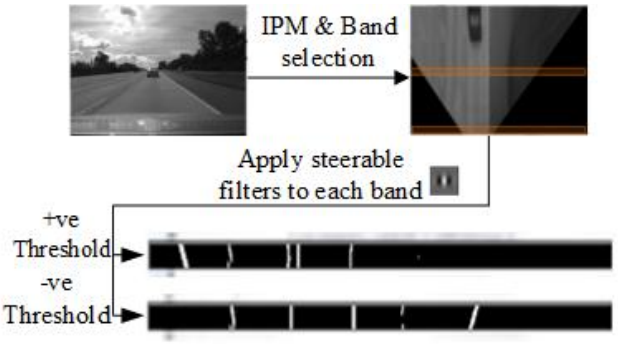

Figure 1. Generating steerable filter output from bands.

It can be seen that E_+ and E_- have non-lane features also. We now propose *shift and match* technique to extract lane features and eliminate non-lane features from each band. In order to do this, we compute the horizontal projection vectors $\mathbf{p}_+$ and $\mathbf{p}_-$ for E_+ and E_- as shown in Fig. 2. Peaks are formed in these projection vectors where there are clusters of pixels in E_+ and E_-. Since the dark→light and light→dark transitions in a lane marking are separated by δ pixels in the IPM image I_W, the peaks corresponding to the lane edges in $\mathbf{p}_+$ and $\mathbf{p}_-$ are also separated by a small δ. In order to capture these pairs of transitions of lanes, $\mathbf{p}_+$ is shifted by δ places to the left and multiplied with $\mathbf{p}_-$ resulting in the vector $\mathbf{K}_{B_i}$ for scan band B_i, i.e.,

$$\mathbf{K} = (\mathbf{p}_+ << \delta) \odot \mathbf{p}_- \tag{1}$$

where $\odot$ represents point-wise multiplication. Fig. 2 shows the result of the shift and match operation performed on $\mathbf{p}_+$ and $\mathbf{p}_-$ for the upper band selected in Fig. 1. It can be seen that we get peaks in $\mathbf{K}_{B_i}$ in Fig. 2 at the same locations as the left edge of each lane marking in the upper band in Fig. 1. The locations of the peaks in $\mathbf{K}_{B_i}$ for each scan band B_i are then used along with the road model to eliminate outliers.

In this paper, for the sake of illustration and simplicity, we limit the discussion to a simple straight road model in the IPM domain, i.e. we assume the road is straight (deviated only by a few pixels). Considering that input images are calibrated with WCS in IPM image, the lane marking positions

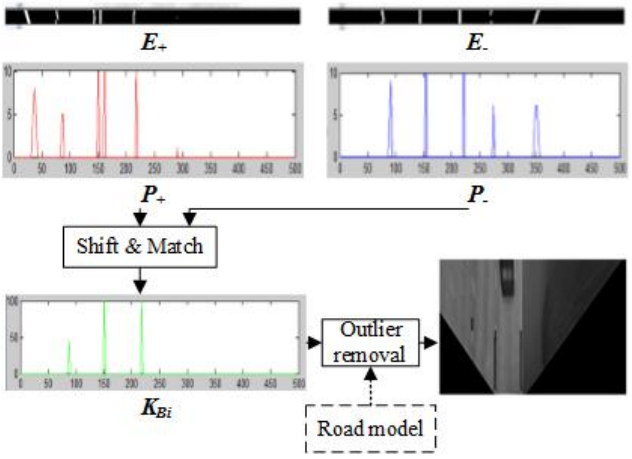

Figure 2. Illustrating shift and match operation for band B_i

can be predicted in a deterministic manner. Let us take the case of ego-lane. After calibration, if x_L and x_R correspond to the lane positions of the left and right lane markings of the ego-lane, the lane markings are expected to be in the vicinity of these lane positions. The peaks positions in $\mathbf{K}_{B_i}$ from each scan band are mapped to the predicted lane markings x_L and x_R. This mapping will eliminate any outliers that may be picked during the shift and match operation. The lane estimation output is shown in Fig. 2.

In order to cater for higher curvatures of the lanes, lane models like clothoid model can be also be used on the peak positions obtained in $\mathbf{K}_{B_i}$ to estimate curved lanes and also eliminate the outliers. Furthermore, lane tracking using Kalman filters using vehicle dynamics like yaw rate and steering angle information [6] increases the robustness of outlier removal tremendously.

Fig. 3(a) shows the overall lane analysis method using the proposed lane feature extraction method. Fig. 3(b) & (c) show two possible design options enabled by the proposed scan band based lane feature extraction. The filtering operation and shift-match operation that are applied on each scan band can be ported as a processing element (PE). A parallel architecture with each scan band being processed by one PE gives a parallel design option as shown in Fig. 3(b).

The second option shown in Fig. 3(c) is a pipelined option, which can offer a wide variety of design implementations. If one PE is used, we get a serial implementation, where each band is processed serially. The number of pipeline stages can be increased depending on the number of PEs that are used. This pipelined design option can also be used to control/predict the lane feature positions in each subsequent PE. In other words, if PE_0 detects lane features at specific positions, this information can be relayed to the PE_1 as positions around which lane features are expected. Vehicle dynamics and road model information can further aid in the overall robustness and efficiency of this implementation.

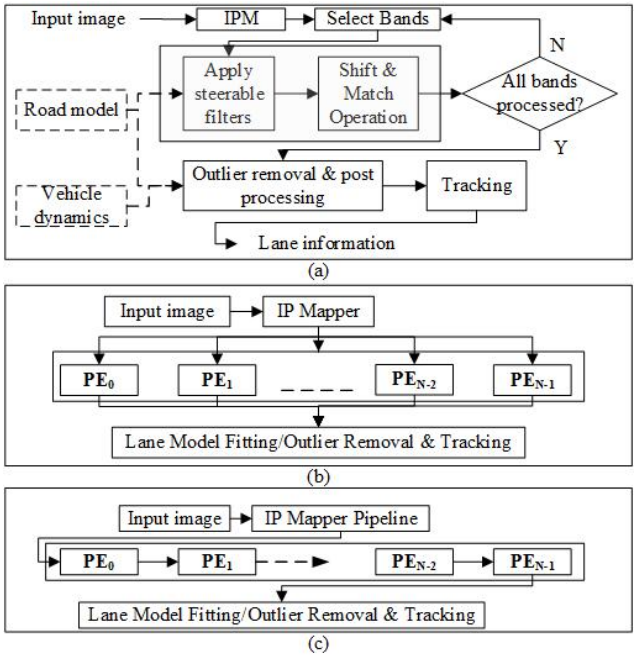

Figure 3. Design options possible for the lane analysis method: (a) Lane analysis using the proposed scan-band based lane feature extraction method, (b) Parallel architecture with each PE catering to each scan band and extracting from all scan bands in parallel, (c) Pipelined architecture with each PE also acting as a controller to predict positions in the next PE.

4. Experimental Studies

In this section, we present a detailed study of the proposed lane feature extraction method to address robustness and the constraints posed by embedded platforms [11]. We present the possible scenarios and tradeoffs between robustness and metrics for embedded realization, that are possible using the proposed technique. We also present the different configurations that can be explored for different conditions and user requirements. As indicated previously, lane tracking is not considered in the scope of evaluations and is considered for future work. Therefore, for the study presented in this paper, lanes are assumed to be detected if the lanes are present in the ground truth and the proposed technique is able to determine the lane features in "correct" positions in the frame. The proposed technique is evaluated using the test video datasets obtained by LISA-Q testbed [6]. The results are presented for five different test image sequences that are listed in Table 1, each dataset having a minimum of 250 image frames that are captured at 10-15 frames a second.

Firstly, Fig. 4 shows some sample images with lanes

Table 1. Dataset description

Set 1	Freeway lanes		Set 2	Freeway with vehicles	
Set 3	Freeway concrete surface		Set 4	Freeway circular reflectors	
Set 5	Urban road with shadows				

that are extracted from complex road scenes by applying the proposed lane feature extraction method on input images from the datasets listed in Table 1. It can be seen that the proposed algorithm is able to extract lanes in varying lane conditions like cracks (Fig. 4(a)-(d)), presence of vehicles (Fig. 4(e)), presence of strong shadows (Fig. 4(e)-(h)). The proposed method is also able to extract lanes with circular reflectors as shown in Fig. 4(f)&(g).

Figure 4. Sample results of the proposed lane analysis method showing lane detection is the complex road scenes.

Fig. 5 shows detection accuracy results of the lanes in datasets 1, 2 and 3, in which we are evaluating the detection of dashed lane markings (i.e. no circular reflectors or solid lane boundaries). The effect of changing the number of scan bands and the scan band width on detection accuracy is shown in Fig. 5. It is evident that reducing the number of scan bands will reduce the detection accuracy of the lane features because depending on the position of the lane marker and the speed of the vehicle, the scan band at a particular coordinate may fail to detect the lane marking (which we consider as failed detection). Therefore, having more scan bands increases the detection rate as seen in Fig. 5 for both cases of the scan band width, i.e. 10 and 5 pixels. The detection accuracy with 8 scan bands is over 90% in all test datasets. This is an important observation because this implies that for the IPM images of size 360×500, processing just 8 scan lines with 10 pixels each is sufficient to get a detection rate of 95%, instead of processing the entire 360×500 sized image (which is usually the case in most conventional methods). This figure also plots the detection accuracy for varying scan band width, i.e. $w_B = 10$ and 5 in Fig. 5. A higher scan width captures more information, implying better detection rate. Therefore, it is expected that bands with width of 5 pixels have lesser detection rate. However, it is noteworthy that as the scan lines increase to 8, the detection rate is nearing 90-95% in both the cases of scan band width. The implication of this on computation cost will be discussed later.

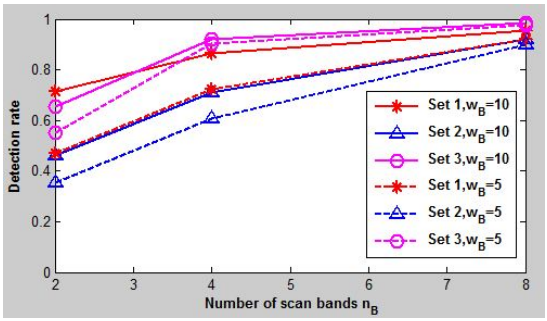

Figure 5. Detection rate versus number of scan bands for scan band width = 10 and 5.

It can also be seen that for a band width of 10 pixels, the difference in accuracy between $n_B = 8$ and 4 is less than 20% in each dataset. Therefore, one can decide to go for 4 scan bands instead of 8, trading off accuracy by less than 20% for half the number of processors.

Let us now consider the main operations involved in the proposed method. Each $k \times k$ filtering operation involves k^2 multiplications, $k^2 - 1$ additions and 1 comparison. Assuming all operations are of equal complexity (simplified model), the total number operations in filtering n_B scan bands of width w_B each and length N_w is equal to $2n_B w_B N_w k^2$. The next step involves horizontal projections in each band, which is $w_B N_w n_B$ addition operations. The shift and add operation involves N_w multiplications

and comparisons per band resulting in a total of $2N_w n_B$ operations. Therefore, the total number of operations for lane feature extraction in the proposed method is given by

$$N_{prop} = 2n_B w_B N_w (k^2 + 1) \quad (2)$$

This is a simplified model but it is sufficient to evaluate qualitatively the effect of scan bands on the overall computation cost efficiency. Fig. 6 shows a scatter plot between number of operations N_{prop} and the detection rate for different possible number of scan bands and scan band widths. The top left corner in the graph, i.e. high accuracy but less number of operations, is the ideal place to be in and we can see in Fig. 6 that using 8 scan bands of width $w_B = 5$ gives similar detection rate as 8 bands of $w_B = 10$ but at 50% lesser number of operations. Also, when compared to conventional methods wherein the entire image is processed for filtering alone, the proposed method gives orders of magnitude savings in the number of operations. Other constraints for embedded realization like total computation cycles, latency, energy cost, total memory accesses etc. are also directly related to the number of operations by different factors.

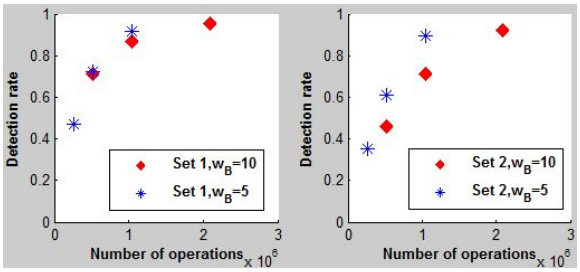

Figure 6. Number of operations versus detection rate for different scan band widths in Set 1 and Set 2.

The effectiveness of the proposed technique to detect circular reflectors using the proposed scan band based lane feature extraction is illustrated in Fig. 7. It can be seen that a detection accuracy of 85% is obtained using 8 scan bands with each band of 10 pixels. A comparison on the effect of reducing the scan bands and their width is also shown in Fig. 7. It can be seen that reducing the scan band width also reduces the detection rate. For the same number of scan bands but scan band width reduced to 5 pixels, the detection rate has been reduced to about 40%. This is because thinner scan bands fail to completely and conclusively capture the circular reflectors. Therefore, having wider scan bands and more number of scan bands to sample as many reflectors as possible is desirable to get higher accuracy.

An experiment was also conducted to see the affect of changing the scan band sizes across different scan bands in a single frame. The scan bands nearer to the ego-vehicle were given higher weight by having thicker bands ($w_B = 10$) as

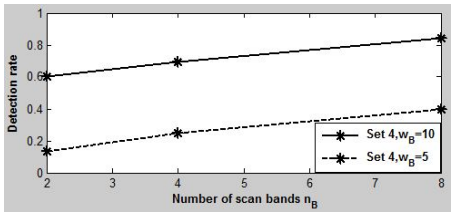

Figure 7. Detection rate versus number of scan bands for Set 4 with circular reflectors.

compared to farther scan bands with $w_B = 5$. Different permutations were used to find if such hybrids can give better detection accuracy for lesser number of operations. Fig. 8 shows the scatter plot with some of these varying options. The option VARY_5_5_5 _10_10_10 is one particularly interesting design option. It gives a detection accuracy of nearly 90%, which is the same as design options with $n_B = 8$ processors for scan band widths $w_B = 5$ and 10 both. However, it uses only 6 processors instead of 8. In terms of number of operations, the design option with $w_B = 5$ is better but this varying scan band width design option is better choice if we want to reduce the number of processors.

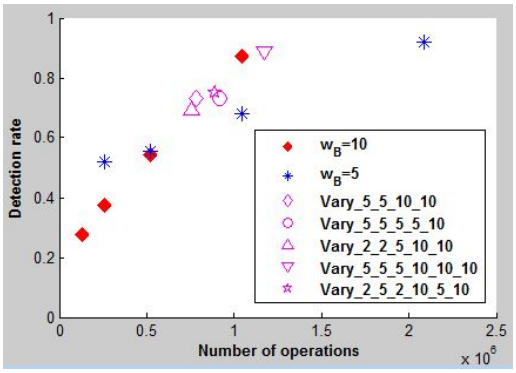

Figure 8. Detection rate versus number of operations with varying band sizes of different scan bands in the same frame.

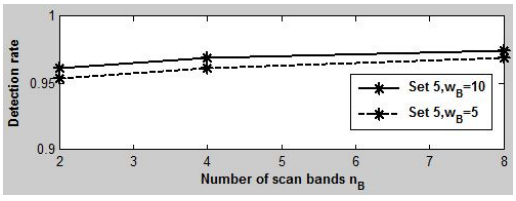

Figure 9. Detection rate for urban lane scenario with solid lane in Set 5.

Fig. 9 shows the detection rates for varying scan bands to detect solid right lane in urban road context (Set 5). It can be seen that detection rates of over 90% are achieved for all band widths and any number of scan bands. Also,

Table 2. Design configurations by varying n_B and w_B

	$n_B \downarrow$ $w_B \downarrow$	$n_B \downarrow$ $w_B \uparrow$	$n_B \uparrow$ $w_B \downarrow$	$n_B \uparrow$ $w_B \uparrow$
Lane Types				
• Solid	√	√		
• Dashed		√ ↔ √		
• Circular Reflectors bot dots				√
Environmental				
• Sunny day	Depends on lane types above			
• Night scene				√*
• Foggy conditions		√*		
• Rainy conditions		√*		
Embedded Constraints				
• Parallel processing			√	√
• Low area constraint	√	√		
• Pipelining			√	√
• Low Memory Resources	√		√	
• Timing	Depends on hardware configuration			

* This also depends on placement of scan bands.

the dataset was chosen such that there are heavy shadows of trees in the images (which usually is the case in most urban road scenarios). These detection rates imply that it is an overkill if more than 2 processors are running when the system is detecting solid lanes.

In Table 2, we present possible recommendations of the different configurations that are possible based on the user requirements, road and environmental conditions. Firstly, we consider the types of lane markings and what combination of scan band size w_B and number n_B could give acceptable detection rates. For example, solid lanes require minimal number of scan bands and can also work with smaller band sizes. However, circular reflectors need higher number and wider band sizes also. Similarly, certain combinations of n_B and w_B are suited for specific environmental conditions. For example, in foggy and rainy conditions, it is desirable to extract lanes from the road surface closest to ego vehicle. Therefore, lesser number of bands but wider bands closer to the vehicle are sufficient for robust extraction.

In the next part of Table 2, we consider the different configurations for n_B and w_B that comply with certain embedded constraints/requirements. A combination of the selections between the different categories can be used to give a user-constrained embedded realization of an accurate lane feature extraction system.

5. Conclusions

In this paper, we proposed a lane extraction method that is shown to provide a way to explore the different configurations for embedded realization, based on the user requirements and the context in which the lane analysis is to be done. It is shown that the two design parameters, i.e. number of scan bands and width of scan bands, can be used to get an embedded vision system that caters to robustness as well as computation cost efficiency. The proposed technique enables to further study a possible adaptable lane analysis solution that takes into account the road and environmental conditions.

6. Acknowledgement

This research was performed at the UCSD Computer Vision and Robotics Research Lab, and LISA: Laboratory for Intelligent and Safe Automobiles. We are grateful for the sponsors supporting these labs and to our colleagues. We also thank the reviewers for their constructive comments.

References

[1] A. Bar Hillel, R. Lerner, D. Levi, and G. Raz. Recent progress in road and lane detection: a survey. *Machine Vision and Applications*, Feb. 2012. 1

[2] A. Borkar, M. Hayes, and M. T. Smith. A Novel Lane Detection System With Efficient Ground Truth Generation. *IEEE Trans. on Intell. Trans. Sys.*, 13(1):365–374, Mar. 2012. 1

[3] R. Gopalan, T. Hong, M. Shneier, and R. Chellappa. A Learning Approach Towards Detection and Tracking of Lane Markings. *Intelligent Transportation Systems, IEEE Transactions on*, 13(3):1088–1098, 2012. 1

[4] N. Hautière, J. Tarel, and D. Aubert. Towards fog-free in-vehicle vision systems through contrast restoration. *2007 IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)*, pages 1–8, 2007. 1

[5] R. Marzotto, P. Zoratti, D. Bagni, A. Colombari, and V. Murino. A real-time versatile roadway path extraction and tracking on an FPGA platform. *Computer Vision and Image Understanding*, 114(11):1164–1179, Nov. 2010. 1

[6] J. McCall and M. Trivedi. Video-Based Lane Estimation and Tracking for Driver Assistance: Survey, System, and Evaluation. *IEEE Trans. on Intell. Trans. Sys.*, 7(1):20–37, Mar. 2006. 1, 3

[7] J. McCall, M. Trivedi, and D. Wipf. Lane Change Intent Analysis Using Robust Operators and Sparse Bayesian Learning. *2005 IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR'05) - Workshops*, 3:59–59, 2005. 1

[8] S. S. Sathyanarayana, R. K. Satzoda, and T. Srikanthan. Exploiting Inherent Parallelisms for Accelerating Linear Hough Transform. *IEEE Trans. on Image Proc.*, 18(10):2255–2264, 2009. 1

[9] E. Shang, J. Li, X. An, and H. He. Lane Detection Using Steerable Filters and FPGA-based Implementation. *2011 Sixth International Conference on Image and Graphics*, pages 908–913, Aug. 2011. 1

[10] S. Sivaraman and M. M. Trivedi. Improved Vision-Based Lane Tracker Performance Using Vehicle. *2010 IEEE Intelligent Vehicles Symposium*, pages 676–681, 2010. 1

[11] F. Stein. The challenge of putting vision algorithms into a car. *2012 IEEE Conf. on Comp. Vis. and Pat. Recog. Workshops*, pages 89–94, June 2012. 1, 2, 3

[12] M. M. Trivedi, T. Gandhi, and J. McCall. Looking-In and Looking-Out of a Vehicle: Computer-Vision-Based Enhanced Vehicle Safety. *IEEE Trans. on Intell. Trans. Sys.*, 8(1):108–120, Mar. 2007. 1