# Multiple Lane Detection Algorithm Based on Novel Dense Vanishing Point Estimation

Umar Ozgunalp, Rui Fan, Xiao Ai, and Naim Dahnoun

***Abstract*—The detection of multiple curved lane markings is still a challenge for advanced driver assistance systems today, due to interference such as road markings and shadows cast by roadside structures and vehicles. The vanishing point $\mathbf{V_p}$ contains the global information of the road image. Hence, $\mathbf{V_p}$-based lane detection algorithms are quite insensitive to interference. When curved lanes are assumed, $\mathbf{V_p}$ shifts with respect to the rows of the image. In this paper, a $\mathbf{V_p}$ for each individual row of the image is estimated by first extracting a $\mathbf{V_{py}}$ (vertical position of the $\mathbf{V_p}$) for each individual row of the image from the v-disparity. Then, based on the estimated $\mathbf{V_{py}}$'s, a 2-D $\mathbf{V_{px}}$ (horizontal position of the $\mathbf{V_p}$) accumulator is efficiently formed. Thus, by globally optimizing this 2-D $\mathbf{V_{px}}$ accumulator, globally optimum $\mathbf{V_p}$ s for the road image are extracted. Then, estimated $\mathbf{V_p}$ s are utilized for multiple curved lane marking detection on nonflat road surfaces. The resultant system achieves a detection rate of 99% in 1862 frames of six stereo vision test sequences.**

***Index Terms*—Lane detection, stereo vision, v-disparity, dynamic programming, vanishing point detection.**

## I. Introduction

ACCORDING to statistics [1], around 70% of all reported road accidents in Great Britain are a result of driver error or slow reaction time. Fortunately, the computation power available today makes it possible to utilize ADAS to prevent or minimize the consequences of these accidents. By using specialized algorithms, ADAS predicts driver intent, warns the driver about possible lane departure or collision, as well as many more functionalities.

Lane detection is one of the key elements of ADAS [2] and it is necessary for lane departure warning. Due to the changing environment, the input image can be noisy and lane detection can be a challenging task. For example, changing light conditions or the lack of consistent painting can affect the lane detection significantly. Thus, some assumptions are commonly made in the algorithms to increase the performance such as constant road width, constant lane painting width, consistent road texture and a flat road [3]. For instance, in [4], assuming parallel lanes, parallel road models are fitted into the novel feature map using RANSAC to minimize the effect of the local noise and decrease the computational complexity (fewer parameters to estimate if road models are parallel to each other). An important property of the input image is perspective mapping. During the image capturing process, the vision sensor maps the three dimensional world information into a two dimensional image. During this process, all parallel lines in the world coordinate system converge on a vanishing point $(V_p)$ in the image coordinate system. Under the assumption that the lanes are parallel to each other, $V_p$ can be used to improve the system robustness significantly. Thus, many researchers [5]–[9] are focused on $V_p$ based lane detection algorithms, since $V_p$ contains global information and $V_p$ based algorithms are less sensitive to local noise, such as occlusions or shadows. Previously developed $V_p$ based lane detection algorithms demonstrated robust results by first detecting $V_p$ and then detecting lanes based on this global information. However, they still have limitations, such as having a flat road assumption, a straight lane model or the ability to detect only the current lane.

In [7], the algorithm assumes intrinsic and extrinsic parameters of the camera are known and that the vehicle is travelling parallel to the road. Thus, $V_p$ can be estimated from these parameters and lines crossing $V_p$ can be searched by using the 2D Hough transform. In [5] and [6], the algorithms first detect and track $V_p$ (including the horizon line). Therefore, this eliminates the assumption of knowing extrinsic camera parameters and the assumption of the vehicle traveling parallel to the road. As a second step, the algorithms search for line pairs crossing $V_p$ for each lane (a lane is a light stripe on a darker background. Thus, each lane has two boundaries).

With a single $V_p$, only a linear lane model can be used and algorithms using a single $V_p$ are only suitable for the roads with limited curvature, such as motorways. In [8] and [9], $V_p$s are detected by segmenting the image into horizontal bands and detecting $V_p$s iteratively, starting from the bottom band and moving through the upper bands. In each iteration, a new $V_p$ is detected by restricting the search range depending on the previous detections from the lower bands. In [10] and [11], multiple $V_p$s are detected by segmenting the image into horizontal image bands and creating a $V_p$ accumulator for each band. Then, starting from the bottom band and moving through the top band, $V_p$ is tracked by a particle filter. Existing $V_p$ based lane detection algorithms detect either a single [5]–[7] $V_p$ or a few [8]–[11] $V_p$s. While single $V_p$ based algorithms are based on only the linear lane model, multiple $V_p$ based algorithms utilize a non-global iterative approach to detect multiple $V_p$s.

In [12], estimating the horizon line from stereo vision is proposed. This algorithm uses a 3D input and, independent of the lane markings in the image, can detect the horizon line

Manuscript received August 13, 2015; revised January 8, 2016 and May 3, 2016; accepted June 15, 2016. Date of publication August 5, 2016; date of current version February 24, 2017. The Associate Editor for this paper was J. M. Alvarez.

The authors are with the Department of Electrical and Electronic Engineering, University of Bristol, Bristol BS8 1UB, U.K. (e-mail: Umar.Ozgunalp@bristol.ac.uk; ranger.fan@bristol.ac.uk; Xiao.Ai@bristol.ac.uk; Naim.Dahnoun@bristol.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2016.2586187

robustly and accurately. Based on the detected horizon line from the stereo vision, $V_{px}$ (the horizontal position of $V_p$) can be detected more robustly and accurately. Furthermore, the computational complexity of the $V_p$ detection algorithm would also decrease, since the 2D $V_p$ detection problem could be reduced into a 1D detection problem. However, the algorithm described in [12] still has limitations. A single horizon line in the complete image restricts the algorithm to a flat road assumption.

In [13] and [14], a ridge detector as a feature extractor followed by a modified sequential RANSAC is used, where a reliable feature extractor can improve the overall system performance greatly. In [15], a compositional hierarchical model is used to remove hard constrains on the lane geometry as imposed by lane models such as clothoids and splines. Thus, the algorithm can detect lanes with a wide range of lane topologies. The methods mentioned above use a single sensor as an input. However, by incorporating multiple sensors, more robust results can be obtained. For instance in [16], information from multiple sensors such as a Global positioning System (GPS), Internal Measurement Unit (IMU), Light Detection and Ranging (LIDAR), and a camera are fused together for lateral distance measurement. Also, in [17], based on the fusion of LIDAR and vision data, an optimal-drivable-region and lane detection system is described, where the proposed algorithm can handle both structured and unstructured roads.

In this paper, the main purpose of the proposed algorithm is to detect the painted lane markings on the road, where the proposed algorithm can be extended to detect both the painted lane markings and the road boundaries as indicated in Section VII. The novel elements proposed in this paper include a global way to estimate $V_p$ using dynamic programming (both in the horizontal direction $(V_{px})$ and in the vertical direction $(V_{py})$ for each individual row of the image, utilization of this $V_p$ curve for multiple curved lane detection on non-flat surfaces, estimating the change in lateral offset of the car in a global way (in pixels) and utilizing this lateral offset for Signal to Noise Ratio (SNR) improvement.

## II. Experimental Set-Up and Calibration

### A. Experimental Set-Up

In our stereo camera rig, two Point Grey Flea3 (FL3-GE-13S2C-CS) cameras have been used. These cameras have 3.75 $\mu$m sensors and they can capture up to 1.3 MP images at 31 fps. Synchronization has been achieved by triggering the cameras using a pulse width modulation signal (the same signal for both of the cameras) using an Arduino board. The base line of the cameras is set to 34 cm. An example set-up is illustrated in Fig. 1.

### B. Disparity Map Estimation

The initial component of the algorithm is the disparity map estimation [18]. The disparity map estimation, which outputs the 3D world information, is useful for both extracting the vertical profile of the road and segmenting it. Although there are several applicable stereo vision algorithms available in the literature, only a limited number of algorithms can achieve good accuracy while working in real-time, such as [19] and [20]. In this paper, we have used our previously published algorithm [21] to acquire the disparity map. This algorithm is suitable for our application due to its good accuracy and high computational efficiency. In Fig. 2, input stereo images and their corresponding calculated disparity map are illustrated.

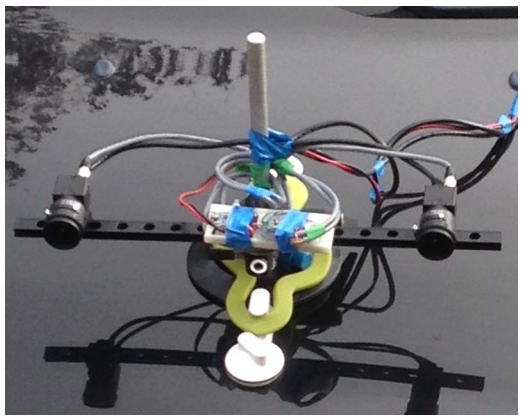


Fig. 1. Example experimental set-up.

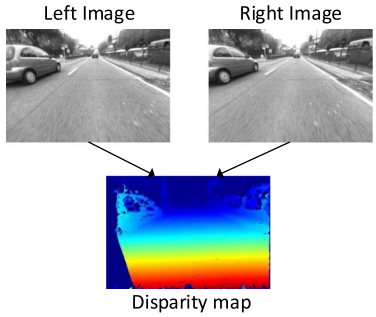



Fig. 2. Disparity map estimation from stereo images.

### C. v-disparity Map and Roll Angle Correction

In the previous section, 3D world information is extracted by stereo vision. Then, this information can be used for extracting the road profile of the road. There are various approaches in the literature to estimate the road profile. For instance, in [22], the road model estimation is based on fitting a 2D quadratic road surface model to the depth map, where outlier elimination is achieved by a RANSAC based approach. Similar to this approach, in our previous paper [23], the road model is estimated by fitting a 2D quadratic road surface model to the depth map, where outliers are eliminated based on efficiently estimated surface normals. Quadratic surface fitting is an accurate method to estimate the road model. Thus, this is mainly adopted for applications that need high precision, such as for detection of small obstacles with a height in terms of few centimeters or pothole detection such as we described in our previous paper [24].

Another widely applied method for road model estimation is the v-disparity map estimation [25]–[27]. The v-disparity map estimation algorithm creates a histogram of disparities for each row of the image and then maps them to the 2D v-disparity map. This can be seen from Fig. 3. In Fig. 3(a), the input image

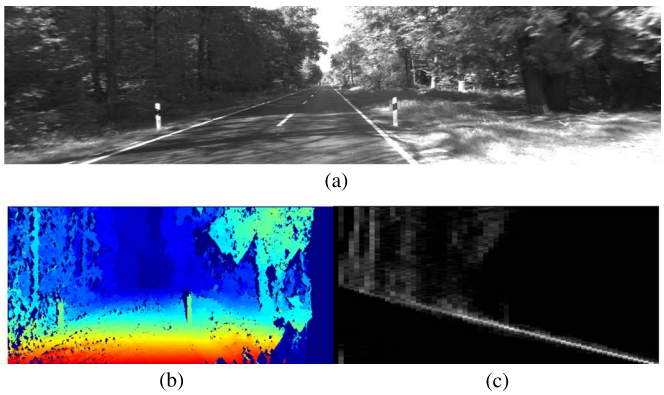



Fig. 3. v-disparity map. (a) Input image. (b) Disparity map of the input image. (c) v-disparity map created using the estimated disparity map.

is shown. In Fig. 3(b), the disparity map of the input image is shown, and in Fig. 3(c), the estimated v-disparity map is shown, where the vertical axis of the v-disparity map is the row number and the horizontal axis of the v-disparity map is the accumulated disparity values.

In the Euclidean coordinate system, the road can be modeled with a road surface model, where $x$ and $z$ are the coordinate variables for estimating $y$ [22]. When the roll angle is zero and the road is assumed to be flat in the $x$-axis, the road model can be defined by only $z$ (distance) vs. $y$ (height). A point in the disparity domain $[u, v, d]$, can be converted to a point in the Euclidean domain $[x, y, z]$ with the following equations [28]:

$$z = \frac{f.b_s}{d}, \quad x = \frac{u.z}{f}, \quad y = \frac{v.z}{f} \tag{1}$$

where $f$ is the focal length, $b_s$ is the base line, $v$ is the vertical coordinate of the pixel, $u$ is the horizontal coordinate of the pixel, $d$ is the disparity, and $[x, y, z]$ is the coordinate of the pixel in Euclidean space (the origin of the image coordinate system is set to the centre of the image). From the equations above, it can be seen that the disparity $(d)$ is inversely proportional to the distance $(z)$ and $v$ is directly proportional to $y$. Thus, the road profile can be estimated using the v-disparity map ($d$ vs. $v$), which corresponds to estimating ($z$ vs. $y$) in the Euclidean domain, where the transformation between these domains is not linear.

The main advantage of using the v-disparity map is to reduce the road model estimation from 2D quadratic surface estimation into line estimation. This greatly reduces the computational complexity. However, it assumes all the road pixels on the same row of the disparity map have the same or similar disparity values. This means that the roll angle needs to be zero, where the roll angle is one of the extrinsic camera parameters (camera height $(h)$, pitch angle $(\theta)$, yaw angle $(\psi)$ and roll angle $(\gamma)$). In Fig. 4, extrinsic camera parameters are illustrated.

Many state of the art road model estimation algorithms [29] only estimate the road profile in the $z$-axis while assuming the road profile in the $x$-axis is linear and the roll angle is zero. Since, the roll angle does not change significantly over time (not more than a few degrees), this is a fair assumption for calibrated cameras. For instance, many available data-sets such as KITTI data-sets are already calibrated and there is no need

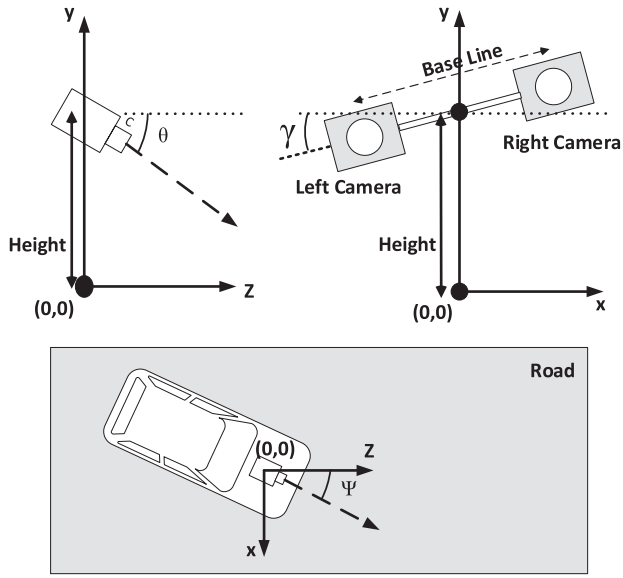



Fig. 4. Extrinsic camera parameters, where camera height $h$, pitch angle $\theta$, yaw angle $\psi$, and roll angle $(\gamma)$ are illustrated.

for roll angle estimation for v-disparity map based road model estimation. However, the experimental set-up used in this paper is manually installed onto the vehicle using air-suction pads, as can be seen in Fig. 1, and a large amount of roll angle can be introduced due to this initial installation. Although the roll angle does not change significantly over time, the roll angle introduced during camera installation onto the vehicle needs to be estimated as a part of the calibration process.

For the reasons discussed above, while estimation of other extrinsic camera parameters is not necessary, the roll angle needs to be estimated for the initial frame to minimize distortion on the v-disparity map due to the roll angle introduced during the camera installation. Thus, the roll angle is only estimated in the first frame and the same roll angle is used for the rest of the video sequence. In this paper, the roll angle is estimated by fitting a plane $(d(r, c) = p1 + p2.c + p3.r)$ to a small patch from the near field in the disparity map, where $d$ stands for the disparity value, $r$ stands for the row value in the disparity image, $c$ stands for the column value in the disparity map, and $p1$, $p2$, and $p3$ are the estimated parameters of the plane. The reason a patch from the near field is used rather than a patch from the far field is it is less likely to appear on an obstacle. Then, since $\tan(\gamma) = \delta r/\delta c$, the roll angle is calculated as $\gamma = a\tan(-p2/p3)$. Using the estimated roll angle, both the input image and the disparity map are rotated using the affine transform. A high roll angle introduced by manually installing a stereo camera onto a vehicle can introduce high distortion. With the roll angle correction, it is seen that the algorithm can remove this distortion from the v-disparity map and create a better v-disparity map. In Fig. 5(a), the original road image is shown. In Fig. 5(b), the disparity map of the image is shown and the patch used for the roll angle estimation is depicted by the black box. In Fig. 5(c), the rotated road image is shown. In Fig. 5(d), the rotated disparity is shown. In Fig. 5(e), the v-disparity map of the original image is shown and, in Fig. 5(f), the v-disparity of the rotated disparity map is shown. As is seen from

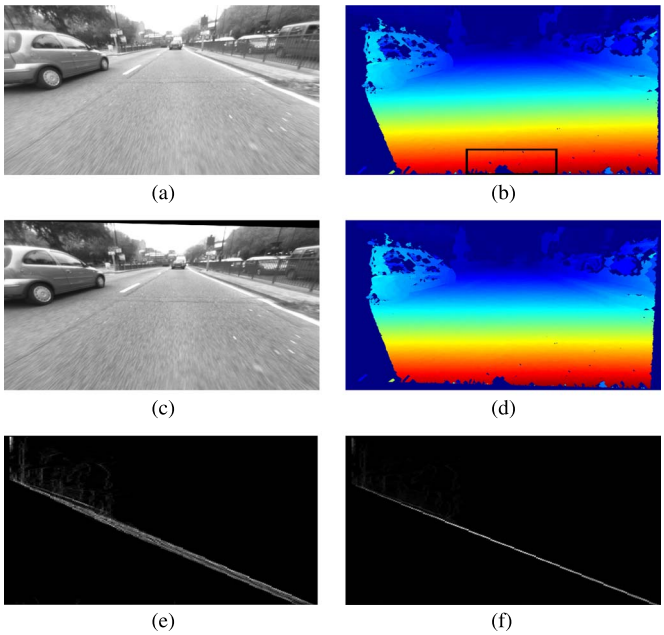



Fig. 5. Roll angle correction. (a) Input image. (b) Disparity map of the input image. The patch used for the roll angle estimation is depicted by the black box. (c) Road image after roll angle correction. (d) Disparity map after roll angle correction. (e) v-disparity map of the original image. (f) v-disparity map after roll angle correction.

Fig. 6. Example road image with a lack of lane painting in the near field [30].

Fig. 5(e) and (f), after the roll angle correction, the v-disparity map becomes sharper. Thus, the vertical profile of the road can be estimated more accurately and the road can be segmented more precisely.

## III. Dense and Global Vanishing Point Estimation

### A. Energy Minimization Based on Dynamic Programming

Traditionally, for the $V_{px}$ estimation of curved lanes, algorithms segment the image into a few horizontal image bands and, iteratively, detect $V_{px}$ of the bands from the bottom band to the upper bands (assuming the road is flat and, therefore, $V_{py}$ is the same for all the bands). However, this approach is not global and, while estimating $V_{px}$ of the current band, the algorithms completely ignore the information supplied in the upper bands. This non-global approach may lead to misdetection. For example, misdetection will occur if there is a higher level of noise in the near field or lack of lane painting. An example of such a case is demonstrated in Fig. 6, where there is no lane painting in the near field. However, there is enough information in the complete image to detect the lanes.

In this section, an optimization algorithm based on dynamic programming is described. In the proposed algorithm, energy minimization using dynamic programming is utilized twice (in this section while optimizing v-disparity map and in Section III-D while optimizing $V_{px}$ accumulator).

In the proposed algorithm, $V_p$ is optimized globally by minimizing the energy function in equation (2) (Please note that, the v-disparity map and the $v_{px}$ accumulator are multiplied by $-1$ for the sake of being consistent with literature and "minimize" energy instead of "maximize"). The data term, $E_{\text{data}}$, penalizes the disagreement in $V_p$. In other words, it depends on the total vote each accumulator gets. The road profile can be assumed to be piecewise smooth. Thus, $V_p$ should also be piecewise smooth. The smoothness term, $E_{\text{smooth}}$, penalizes the change in $V_p$ and ensures smoothness, where $\lambda$ is a smoothness constant

$$E(r) = E_{\text{data}}(r) + \lambda E_{\text{smooth}}(r). \tag{2}$$

The first stage in which dynamic programming [31] has been used is in optimizing the v-disparity map to extract the vertical profile of the road. Dynamic programming has been used to search a path in the v-disparity map which starts from the right and goes to the left (starting from the left would also give exactly the same result). Since the upper rows are further away from the camera, they should have either a decreasing disparity value or the same disparity value (the input disparity map does not have sub-pixel accuracy). Thus, only this pattern is searched.

The v-disparity has two axes, $r$ (row number) and $d$ (disparity). Let $m(r)_d$ be a value on the v-disparity map with a position of row number of $r$ and disparity of $d$. Then, equation (2) can be solved iteratively (column by column in the v-disparity, starting from $d = d_{\max}$ to $d = 1$). In the first iteration, there is no $E_{\text{smooth}}$. Thus, $E = E_{\text{data}}$ and equation (2) can be estimated as

$$E(r)_{d=d\max} = m(r)_{d=d\max}. \tag{3}$$

Please note that r is variable and $E(r)_{d=d\max}$ has a different value for each $r$. Then, for the following iterations, $E$ can be estimated based on previous iterations as:

$$E(r)_d = m(r)_d + \min_{\tau=0}^{\text{range}-} \left[E(r+\tau)_{d+1} + \lambda \cdot \tau\right]. \tag{4}$$

While moving through the upper rows of the image, $z$ (distance) of the road should increase. Since the disparity ($d$) is inversely proportional to the distance ($z$), while moving through the upper rows of the image, the disparity value should be either the same or decreasing. Thus, while optimizing the v-disparity map, the range in equation. (4) is set to $[-7, 0]$. In each iteration ($d$ is decremented and the v-disparity map is processed column by column), index values of the minimum (which gives direction information) are saved into a buffer with the same size of v-disparity map (shown by the arrows in Fig. 7 and there is a different index for each $r$). Then, once $E(r)$ is calculated for the final column which corresponds to $E(r)_{d=1}$ (depicted by the brown line in Fig. 7), a minimum is selected (depicted by the red dot). Then, the algorithm backtracks along the path starting from the last column to the first column by using previously recorded directions (in Fig. 7, the tracked path is illustrated by green arrows). In this way,

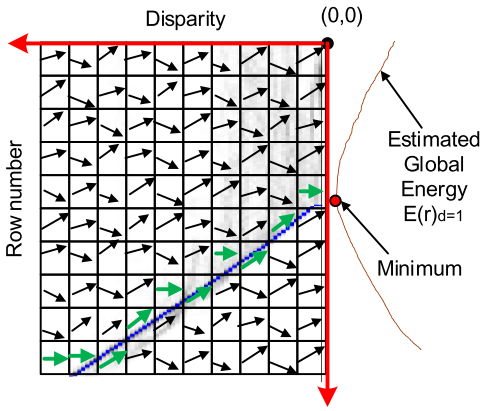



Fig. 7. Vertical road profile estimation using dynamic programming on the v-disparity map, where indexes saved into a buffer are illustrated by the arrows. When the global minimum is estimated in the last iteration, depicted by the red dot at the end, the algorithm estimates the path by following the indexes saved previously into a buffer. For instance, in the illustrated figure, the algorithm followed indexes indicated by the green arrows.

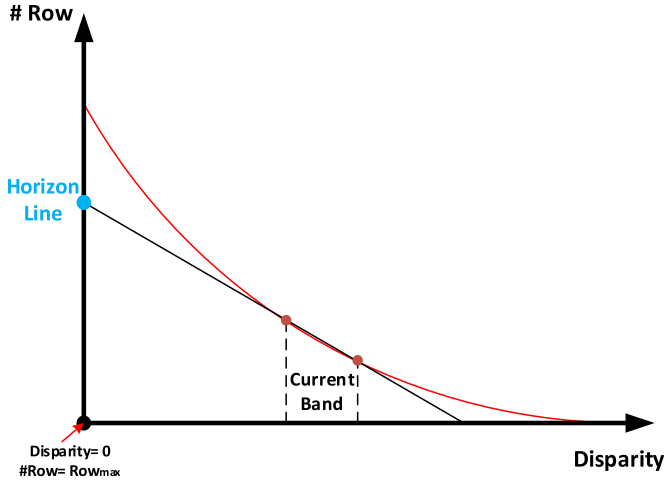



Fig. 8. Estimation of a horizon line for an image section based on a known vertical road profile.

dynamic programming searches all possible paths efficiently with connectivity restriction and the resultant path would have energy as described in equation (2).

In this algorithm, if the far field is occluded, due to the smoothness term in equation (2), the estimated path would have a flattened region at the end. Thus, the unreliable area in the far field can be estimated. Once this region is removed from the estimated $V_{py}s$, the 1D output (a $V_{py}$ for each row, i.e. the blue line in Fig. 7) is fitted to a quadratic equation. The advantages of the fitting stage include the estimation of the road profile at the far field and reducing the number of the output parameters from a few hundred (depends on the image resolution) to a few. Thus, tracking in the time domain would be much more efficient.

### B. Horizon Line Calculation for Each Row of the Image

The horizon line $(V_{py})$ estimation is an important step for $V_p$ detection. Some algorithms assume it is fixed and can be estimated by camera parameters [8] (ignoring the camera shakes) and some algorithms estimate a single horizon line [6]. For a flat road, a single horizon line can be estimated. However, for a non-flat road, the horizon line is continually changing according to the elevation of the road. [25], and [32] described a solution to the pitch angle estimation based on stereo vision which is essentially the same as estimating the horizon line for a flat road. In this section, estimating the horizon line for each row of the image for non-flat roads is proposed. Projection of the flat road to the v-disparity map is a straight line since the disparity of the road should decrease linearly and the road profile in this case can be estimated by using straight line detectors such as the Hough transform. However, for a non-flat road, the projection of the road is not a straight line (for example, it can be modeled with a quadratic model). The approach taken in this paper is to estimate the horizon line by taking two points from the boundaries of a band (their disparity values are already estimated using dynamic programming as in Section III-A) and, since they are close to each other, the change in elevation is small and that piece of the road can be assumed to be flat. By using these two points, a line equation can be calculated and the cross section of this line and the column on the v-disparity map which has 0 disparity value is the horizon line for that section of the road. After calculating the horizon line for this section, another 2 points are taken which are shifted one row above and the same process is applied to estimate a horizon line for this section. This process is iteratively calculated until the last estimated row of the road (see Fig. 8). Alternatively a derivative can be used to estimate the line equation of the locally planar surface. However, in this paper, the shifting band approach is preferred, since the shifting band approach is used in $V_{px}$ estimation with known boundaries which will be further discussed in Section III-C.

### C. Forming Accumulator for $V_{px}$

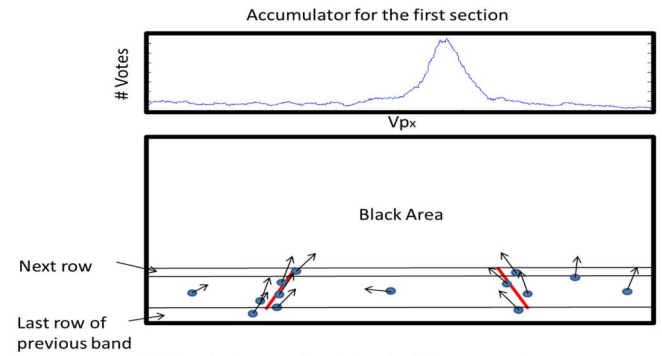
$V_p$ is composed of two values, $V_{px}$ and $V_{py}$. In the previous sections, $V_{py}$ is already estimated for each individual row of the image. The initial step of the proposed approach is to take a segment from the near field of the image and to form an accumulator for $V_{px}$. Then, all the edge points are voted to this accumulator according to each edge point's orientation and position. This initial step (forming the 1D $V_{px}$ accumulator) is similar to the method used in our previous paper [8]. In this work, instead of using a fixed horizon line, it is estimated by relying on the 3D information acquired by stereo vision.

The proposed approach then shifts the current band slightly up and creates another vanishing point accumulator. Computational efficiency is achieved by adding the edge point's votes which appear one row above the current band to the initially calculated accumulator and subtracting the edge point votes which appear on the bottom row of the current band from the initially calculated accumulator.

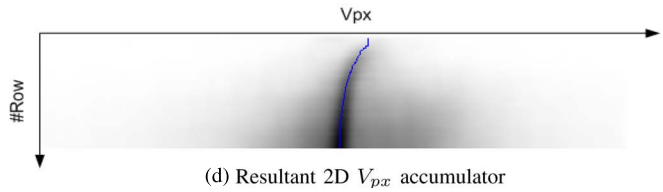
Furthermore, the far field of the road may contain a higher curvature. So, thinner bands are desirable for the upper bands of the image. Formation of thinner bands is achieved during the shifting process by subtracting more than one row from the bottom of the previous band, while adding only one row to the top of the previous band. As a result, the initial band to final band thickness ratio is adjustable.

(a) Original Image

(b) Feature map




(c) Shifting horizontal band for the $V_{px}$ accumulator




(d) Resultant 2D $V_{px}$ accumulator

Fig. 9. Creating the 2-D $V_{px}$ accumulator. (a) Input image. (b) Estimated feature map after postprocessing (see Section IV for more details). (c) Creating 1-D $V_{p_x}$ accumulators and updating it (after saving the 1-D accumulator into a 2-D accumulator according to its row number) iteratively for efficiency. (d) Created 2-D $V_{p_x}$ accumulator, where the optimization result is shown by a blue line on the accumulator.

By this approach, instead of creating an accumulator from scratch for every band, the algorithm updates the previous one by only processing and adding the top row (the vote positions of the bottom band are already calculated when adding them. Only subtraction is needed). Once an accumulator is updated for a band, the algorithm saves it to a 2D accumulator.

This process is demonstrated in Fig. 9. In Fig. 9(a), the input image is shown. In Fig. 9(b), the extracted feature image is shown (feature map segmentation will be described in Section IV). In Fig. 9(c), the process of creating the 2D $V_{p_x}$ accumulator is demonstrated. In Fig. 9(d), the extracted 2D $V_{p_x}$ accumulator is shown (the blue line in Fig. 9(d) is the estimated $V_{p_x}$s and the estimation method will be introduced in the next section).

### D. Estimating $V_{px}$

The vanishing point can be described as the cross section point between the tangent line of the lane and the horizon line. Thus, if the road model is a polynomial with an order of n, then, the projection of this road model should also be the same order polynomial, as shown in

$$Vpx(r) = u(r) - \frac{d}{dr}u(r) \cdot (r - Hz) \tag{5}$$

where $u(r)$ is the equation of the lane model. Thus, we can assume the vanishing point is changing gradually instead of having sudden jumps. In previous sections, the 2D $V_{px}$ accumulator has already been constructed. Similar to Section III-A, this 2D accumulator can be optimized for a gradually changing output by dynamic programming to estimate the best path and acquiring a $V_{px}$ for each row. For this purpose, equation (3) and equation (4) can be rewritten as equation (6) and equation (7), where the range is set to $[-4, 4]$

$$E(V_{px})_{r=r\max} = m(V_{px})_{r=r\max} \tag{6}$$

$$E(V_{px})_r = m(V_{px})_r + \min_{\tau=\text{range}-}^{\text{range}+} \left[E(V_{px}+\tau)_{r+1} + \lambda \cdot \tau\right]. \tag{7}$$

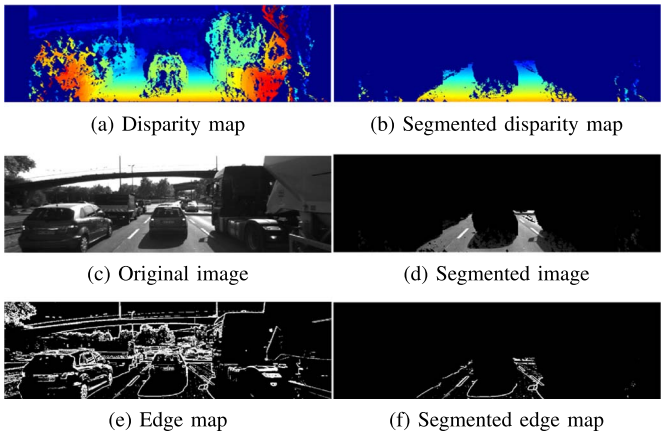
For the feature map in Fig. 9(b), the resultant 2D $V_{px}$ accumulator is shown in Fig. 9(d). In Fig. 9(d), a 2D $V_{px}$ versus row position accumulator is demonstrated. The more votes a cell gets the darker it seems and the blue line on the top of the figure is the optimization result for this accumulator.

By scanning the images as described in the previous sections, a series of $V_p$ is estimated and each calculated vanishing point is estimated as for the row which is in the middle of the corresponding band. In this way, each $V_p$ is estimated except the rows under the middle of the first band. However, these rows are in the near field where the lanes tend to be straight and they can be estimated as the same as the $V_p$ of the first band.

## IV. Post Processing on Edge Map

The feature map used for the lane detection is the edge map. In this paper, to minimize salt and pepper noise, a variable kernel size median filter is used. Consequently, the median filter used has a large size in the near field and its size decreases directly proportionally to the estimated road disparity value for each image row (estimated using v-disparity).

For lane detection purposes, edges can be classified into a few categories: edges on the sky, the obstacles, the lane markings, the road markings and noise on the road such as caused by cracks and shadows. Since the vertical profile of the road is already calculated, this information can be used to eliminate further noise on the edge map and increase the SNR before calculating the horizontal profile of the road. Since some noise such as road cracks, shadows and road markings (apart from lane markings) appear on the road with the same disparity as the road, such noise cannot be eliminated by using the vertical profile of the road. However, most of the edge points caused by the obstacles and sky can be segmented. This is done by eliminating the edge points which have different disparity values (more than a few pixels) from the ones calculated by dynamic programming for their rows in the v-disparity map. Especially for the urban environment, this process can dramatically improve the SNR of the edge map, as seen in Fig. 10.



(a) Disparity map (b) Segmented disparity map

(c) Original image (d) Segmented image

(e) Edge map (f) Segmented edge map

Fig. 10. Feature map segmentation. (a) Input disparity map. (b) Segmented disparity map using the estimated vertical profile of the road. (c) Input left image. (d) Segmented left image using the segmented disparity map as a mask. (e) Estimated feature map. (f) Segmented feature map using the segmented disparity map as a mask.

In Fig. 10(a), the calculated disparity map is shown. In Fig. 10(b), the calculated disparity is segmented for the road. In Fig. 10(c), the original image is shown. In Fig. 10(d), the original image is segmented by using the segmented disparity map. In Fig. 10(e), the original edge map is shown and, in Fig. 10(f), the edge map is segmented by using the segmented disparity map.

## V. Lane Detection Using Estimated Dense Vanishing Point

### A. Forming the Likelihood Function

In the previous sections, a series of $V_p$ is estimated. $V_p$ can give direction and curvature information of the lanes. However, lateral positions of the lanes still remain unknown. There is only one unknown variable left to detect for the lane detection. A 1D accumulator has been formed with a width of 2∗image width. This approach is similar to the paper presented by [5]. For each possible value of intersection point (constructed candidate lane and bottom row of the image), a likelihood value is calculated by allowing edge points underneath the constructed lane vote for individual starting points. Each edge point $(e)$ votes according to the following equation:

$$V(e) = \nabla(e) \cdot \cos\left(\theta_e - \theta_{V_p}\right) \tag{8}$$

where, for each individual edge point, $V$ is the vote, $\nabla(e)$ is the gradient, $\theta_e$ is the angle of the edge point and $\theta_{V_p}$ is the angle between the edge point and the vanishing point (in the future, inclusion of the connectivity is also planned). For the image in Fig. 11(a), the green area is the segmented road area and only the edge points in this area are used for voting. The red lines on Fig. 11(a) are created road patterns for some example starting points. In Fig. 11(b), the estimated 1D accumulator is illustrated for the image in Fig. 11(a).

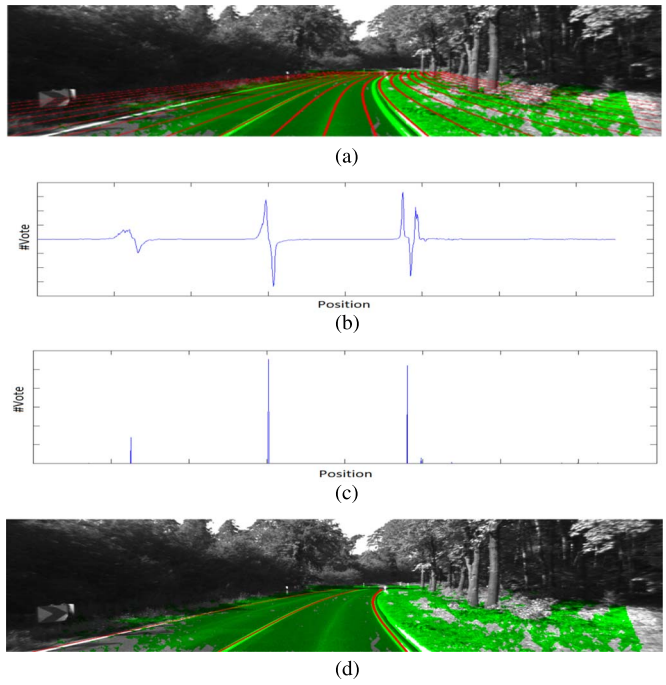



Fig. 11. Creating 1-D accumulators. (a) For an example input image, the ground plane is illustrated with the green area (excluding obstacles, sky and occluded areas in the disparity map), and some of the candidate lane markings are illustrated with red lines. (b) Projection of candidate lane markings to the 1-D accumulator (the size of this signal is twice of the image width). (c) One-dimensional signal after plus–minus peak pair selection. (d) Lane detection result.

### B. Peak Pair Selection

Due to the dark-light-dark transition of the lane markings, a lane marking is projected into a 1D likelihood accumulator as a plus–minus peak pair. This property can be seen in Fig. 11(b).

To detect these peak pairs, the algorithm initially finds plus and minus peak points and, secondly, for each plus peak point, finds minus peak points within a range and creates a Dirac function at the middle of the peak-pair in another accumulator. This process can be seen in Fig. 11. For the initial accumulator in Fig. 11(b), a new accumulator is created for peak pairs, as is seen in Fig. 11(c). Then, lateral offsets of the lanes are selected from this new accumulator. Starting from the highest peak, the algorithm eliminates all other peaks within the range ($\pm 1$ m) and, then, detects the next highest peak from the signal and eliminates all other peaks within the range ($\pm 1$ m). The algorithm iterates until the detected peak is lower then the selected threshold. The detected lanes for Fig. 11(b) are illustrated in Fig. 11(d).

### C. 1D Signal Noise Reduction

The proposed algorithm can detect multiple lanes. However, compared to the lane in which the vehicle is traveling, the lanes further from the vehicle are harder to detect, especially when these lanes are dashed. To detect multiple lanes consistently, a noise reduction step is applied. In this paper, via the estimated $V_p$s, the lane detection is reduced to a 1D problem. Let the noise-free signal be $x(n)$ and the estimated signal be

$$S_i(n) = x(n) + w(n) \tag{9}$$

TABLE I
DETECTION RESULTS OF THE PROPOSED ALGORITHM

| Sequence | Total Lane markings | Correct detection | Incorrect detection | Misdetection |
|---|---|---|---|---|
| Sequence 1 | 860 | 860 | 0 | 0 |
| Sequence 2 | 594 | 594 | 0 | 0 |
| Sequence 3 | 376 | 376 | 0 | 0 |
| Sequence 4 | 156 | 147 | 0 | 9 |
| Sequence 5 | 678 | 661 | 0 | 17 |
| Sequence 6 | 1060 | 1039 | 14 | 7 |
| total | 3724 | 3677 | 14 | 33 |

TABLE II
DETECTION RESULTS OF [8]

| Sequence | Total Lane markings | Correct detection | Incorrect detection | Misdetection |
|---|---|---|---|---|
| Sequence 1 | 860 | 848 | 12 | 0 |
| Sequence 2 | 594 | 550 | 44 | 0 |
| Sequence 3 | 376 | 332 | 44 | 0 |
| Sequence 4 | 156 | 139 | 17 | 0 |
| Sequence 5 | 678 | 571 | 107 | 0 |
| Sequence 6 | 1060 | 880 | 180 | 0 |
| total | 3724 | 3320 | 404 | 0 |

where $w(n)$ is white noise. For instance, $w(n)$ can be artifacts on the road which are directed to the $V_p$ but, however, do not consistently appear on consecutive frames. If the road thicknesses are the same for two consecutive frames, the noise-free signal for frame $i-1$ would be $x(n-L)$ where $L$ is the change in lateral offset. Thus, the estimated signal for frame $i-1$ can be defined as in

$$S_{i-1}(n) = x(n-L) + w(n). \tag{10}$$

Thus, the change in lateral offset $(L)$ between two consecutive frames can be estimated by applying cross-correlation between these two signals via equation (11). In this way, before detecting the lanes, the lateral offset change can be estimated

$$r_{i,i-1}(k) = \sum_{n=0}^{N} S_i(n) \cdot S_{i-1}(n+k). \tag{11}$$

The probability density function (PDF) of the change in lateral offset $(L)$ is then estimated by normalizing $r_{i,i+1}(k)$ for the range of $k$

$$\text{PDF}_{i,i-1}(k) = \frac{r_{i,i-1}(k)}{T} \tag{12}$$

where

$$T_{i,i-1}(k) = \sum_{k=-20}^{20} r_{i,i-1}(k). \tag{13}$$

Finally, signal alignment can be achieved by convolving PDF of $L$ with the next signal. For random noise reduction (improvement in SNR), a few signals for a few consecutive frames are iteratively aligned and added together as

$$\text{ST}_i(n) = S_i(n) + S_{i-1}(n) * \text{PDF}_{i,i-1}. \tag{14}$$

## VI. EXPERIMENTAL RESULTS

The proposed algorithm is compared with our previously published lane detection algorithm described in [8] (excluding the tracking step), where lanes are detected based on vanishing point estimation. To quantify the performance of the algorithm, we first tested the lane detection ratio.

To quantify the robustness of the system, the detection ratio of the algorithm is also estimated on sample sequences from KITTI datasets [33], and from the video sequences recorded by our camera and compared with [8]. Detailed results are shown in Tables I and II (only the closest lanes are taken into account in this table and detection rates on these sequences are subjectively estimated).

Sample detection results are illustrated in Figs. 12 and 13 (a sample video sequence will also be available at http://ieeexplore.ieee.org). In Fig. 12, the first four rows consist of results from the KITTI data-sets, where lane detection results are demonstrated under dense traffic and shadows. In Fig. 12, the second two rows consist of detection results from the video sequences recorded by our experimental set-up, where lane detection results are demonstrated for the cases when there are no road markings, when the vehicle is on the bridge with a vertical curve, when the vehicle is entering and leaving under a bridge, when there is only one painted lane, when there is saturation in the image due to exposure time adjustment after leaving under a bridge, and when there are road markings. Failure cases are demonstrated in the last two figures, where in the first one, the lane edges disappear due to the image saturation and cause the algorithm to misdetect lanes, and in the second one the algorithm detected the road marking instead of the lane marking. Road markings are also light stripes parallel to the vanishing point. Thus, they can cause the algorithm to fail. In the future, adding a tracking step to the algorithm is planned to be able to cope with such cases.

In Fig. 13, detection results under dense shadow are shown, where estimated feature maps (on the left) and the resultant detection results (on the right) are shown side by side. In this figure, the input video sequence is recorded with a single camera. Thus, a fixed horizon line is manually selected and given to the algorithm as an input. To estimate the feature map, the Sobel edge detector with a threshold of 80 is used. Although a relatively higher threshold can be used, as can be seen from the figure on the top left corner, almost all lane feature points are eliminated under shadow. In this sequence, the proposed $V_p$ detection itself only failed in 4 frames, resulting detecting lanes (both left and right) with incorrect shapes and 29 lane markings are misdetected when only ego lane markings are considered (out of 2092 frames).

As stated by the survey paper [3], there are no ground truth data-sets available for lane detection and there is a lack of accepted test protocols. Thus, many lane detection papers reported their results only qualitatively. For this reason, the detection of multiple lane markings is evaluated qualitatively. To do so, the intersection point between the detected lanes and the bottom row of the image are plotted for each frame, where, if there is an incorrect detection, a sudden jump is expected. The inputs of Fig. 14(a)–(c) are recorded using a stereo vision camera, where the input of Fig. 14(d) is recorded with a

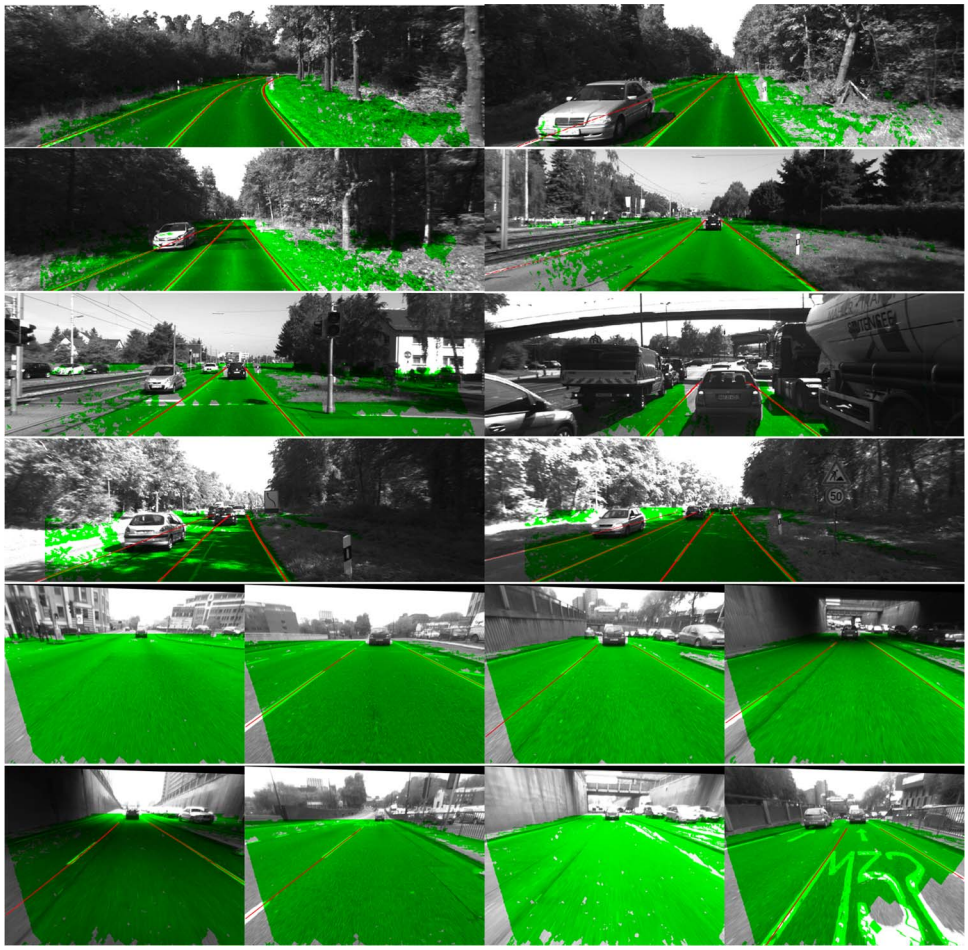


Fig. 12. Experimental results using video sequences recorded by stereo vision cameras, where the top four rows consist of detection results using the KITTI data set as the input and the bottom two rows consist of detection results using video sequences recorded by our stereo vision camera rig.

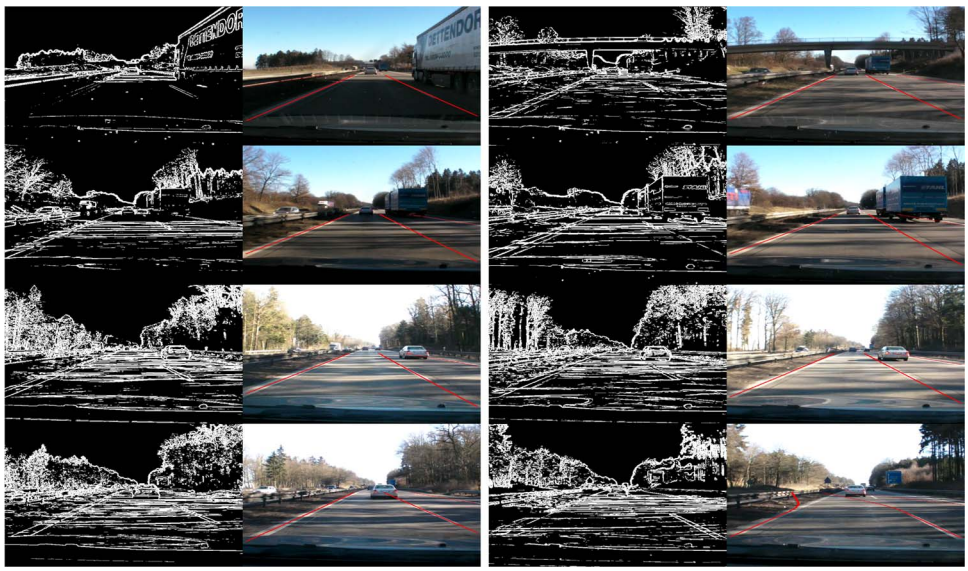


Fig. 13. Experimental results using a video sequence recorded by a single camera, where sample detection results are demonstrated under dense shadow. For the input frames, the estimated feature maps are shown on the left, and the lane detection results using these feature maps are shown on the right.

single camera (3D input is not available). In Fig. 14(a), robust results are demonstrated for multiple lane detection under dense traffic. In this figure, there are also incorrect detections for the ego-lane. This error is due to the consistent artifact on the road with dark-light-dark transition. In Fig. 14(b), while ego-lane is separated with painted lanes, the other lane is separated with

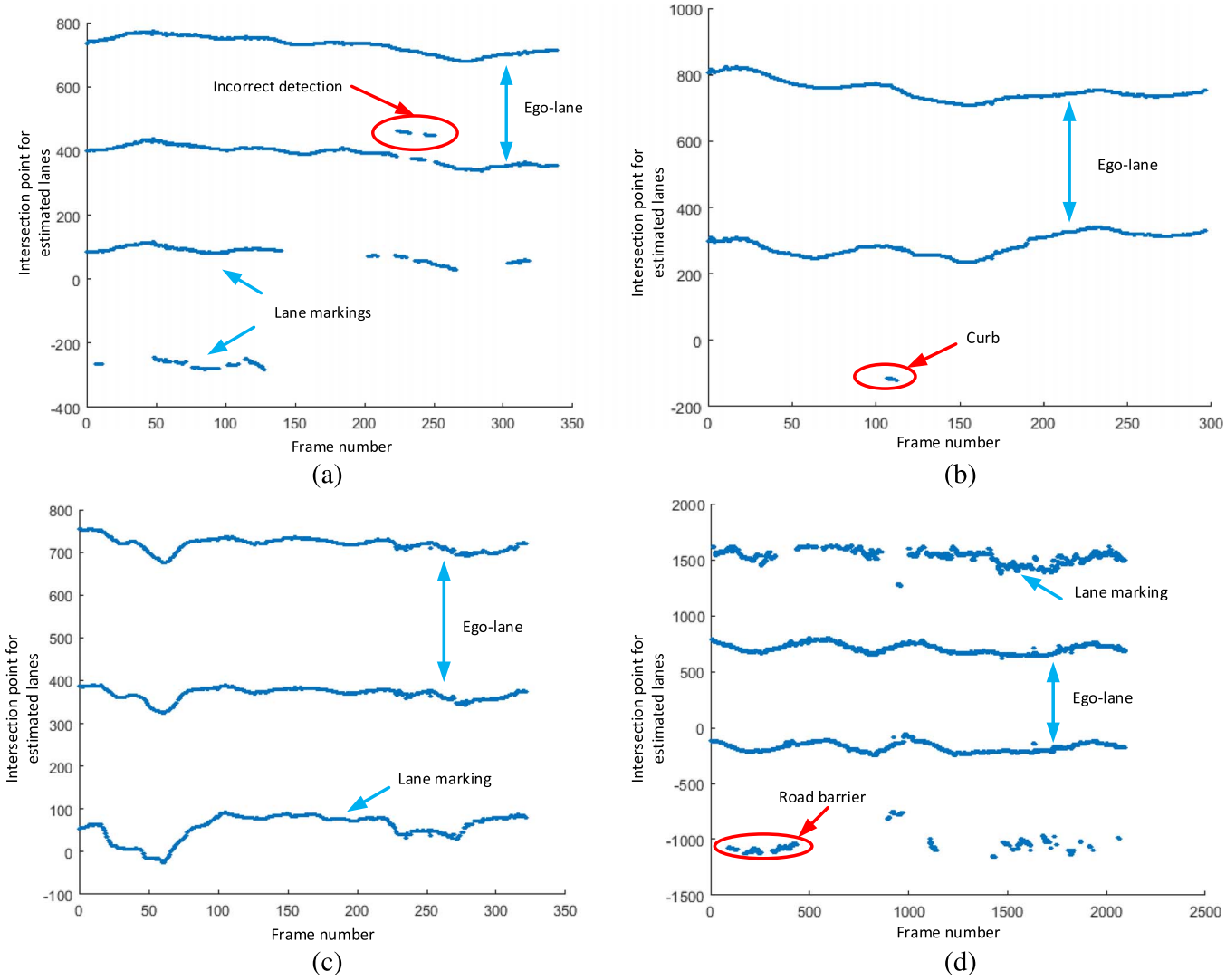



Fig. 14. The plots for intersection between the detected lane markings and the bottom row of the image for four different sequences are illustrated, where the example incorrect detections are illustrated with red ellipses. The inputs of (a), (b), and (c) are recorded with a stereo vision camera, where the input of (d) is recorded with a single camera.

only a curb. Apart from the dark-light-dark transition, curbs have similar properties to the lanes (curbs are also oriented to the $V_p$ and they are not very much higher than the road). Although detecting road boundaries is included in the future work (See Section VII for details), currently curbs are considered as incorrect detections. The input of Fig. 14(d) is recorded with a single camera (3D input is not available). Thus the extrinsic camera parameters are assumed to be known (the horizon line is manually selected and given as an input to the algorithm. Also, the road is assumed to be flat). Although the algorithm detected lanes robustly especially for the ego-lane, the road barrier next to the road is also incorrectly detected for many frames as a lane. The edges of the road barrier are directed to the $V_p$ and furthermore due to the illumination, dark-light-light transition can be occurred. Thus, the proposed algorithm detected the side of the road barrier as a lane. This sequence was recorded with a single camera and 3D input was not available. However, it would be possible to eliminate these errors with using 3D input and segmenting the road area. With this sequence, the robustness of the algorithm under dense shadow and the ability to run using a single camera input is demonstrated, if stereo vision input is not available. However, since the 3D input would eliminate several assumptions and eliminate noise from the objects above the ground, it is desirable to use this information. Furthermore, the lane detection is an ADAS application, where it is expected that many other algorithms would already need 3D input. The proposed algorithm is implemented both in MATLAB and C languages. The run-time of the algorithm for a single frame with a resolution of 1242 by 375 is 2.41 seconds in MATLAB, and 0.26 seconds in C language (using a single thread with the Intel I5-5300U CPU). The given run time excludes the disparity map estimation and the disparity map is assumed to be given as an input to the algorithm. However, many steps of the algorithm such as dynamic programming are suitable for parallel processing.

## VII. Future Work

In this paper, the algorithm searches for dark-light-dark patterns for the lanes which is a more robust way (compared to searching for a single boundary) to detect lanes and currently it can only detect painted lanes. In Fig. 11(a), it can be seen that there are both lane marking and a road boundary on the right hand side of the road. This is projected to the 1D likelihood accumulator, as in Fig. 11(b), as peaks. In that position, there is a plus peak just after the plus–minus peak pair, where the plus–minus peak pair is due to the lane marking and the plus peak just after this peak pair is due to the road boundary. If there were not any painted lane markings, there would be only a positive peak. To detect this road boundary, it is possible to modify the peak pair selection in an ad-hoc manner. Such as, if no lane is detected on either side of the road, the algorithm should also search for the single high peaks. Thus, the algorithm would be able to detect painted lanes robustly by using the dark-light-dark pattern of the lanes and, at the same time, it would be able to detect road boundaries, when painted lane markings are not available. Currently, the algorithm detects lanes accurately and robustly. However, further improvement is possible by applying tracking. For this purpose, it is also planned to fit estimated $V_{px}$ values and $V_{py}$ values of a frame into spline models and track estimated control points. Thus, tracking lanes would be possible by tracking a few parameters.

## VIII. Conclusion

In this paper, a novel lane detection algorithm is presented. The main novel elements of this paper include dense vanishing point estimation, the use of estimated vanishing points to detect lanes, estimating the change in lateral offset of the car in a global way and utilizing this estimation for SNR improvement. The $V_p$ contains the global information of the road image. Hence, $V_p$ based lane detection algorithms are quite insensitive to interference and they demonstrate robust results. The algorithm described in this paper, proposes a global approach for dense vanishing point estimation and it can detect multiple lanes with both horizontal and vertical curvature. Experimental results show that the proposed algorithm works robustly and accurately even in dense traffic. The run-time of the algorithm for a single frame with a resolution of 1242 by 375 is 2.41 seconds when implemented in MATLAB, and 0.26 seconds when implemented in C language (using a single thread with the Intel I5-5300U CPU). Furthermore, due to the flexibility of the described system, the user can simply plug a stereo camera rig (experimental set-up) onto a vehicle without concern about any of the external camera parameters (i.e. camera height, pitch, yaw or roll angle).

## References


[1] "Reported road casualties Great Britain 2009," Stationery Office Dept. Transp., London, U.K., 2010.

[2] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 20–37, Mar. 2006.

[3] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vis. Appl.*, vol. 25, pp. 1–19, 2012.

[4] A. López, J. Serrat, C. Cañero, F. Lumbreras, and T. Graf, "Robust lane markings detection and road geometry computation," *Int. J. Automotive Technol.*, vol. 11, no. 3, pp. 395–407, 2010.

[5] D. Schreiber, B. Alefs, and M. Clabian, "Single camera lane detection and tracking," in *Proc. IEEE Intell. Transp. Syst.*, 2005, pp. 302–307.

[6] D. Hanwell and M. Mirmehdi, "Detection of lane departure on high-speed roads," in *Proc. ICPRAM*, 2012, vol. 2, pp. 529–536.

[7] B. Fardi and G. Wanielik, "Hough transformation based approach for road border detection in infrared images," in *Proc. IEEE Intell. Veh. Symp.*, 2004, pp. 549–554.

[8] Y. Wang, N. Dahnoun, and A. Achim, "A novel system for robust lane detection and tracking," *Signal Process.*, vol. 92, no. 2, pp. 319–334, 2012.

[9] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using B-snake," *Image Vis. Comput.*, vol. 22, no. 4, pp. 269–280, 2004.

[10] Y. Wang, L. Bai, and M. Fairhurst, "Robust road modeling and tracking using condensation," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 4, pp. 570–579, Dec. 2008.

[11] C. Rasmussen, "Grouping dominant orientations for ill-structured road following," in *Proc. IEEE CVPR*, 2004, vol. 1, pp. I-470–I-477.

[12] R. Labayrade and D. Aubert, "A single framework for vehicle roll, pitch, yaw estimation and obstacles detection by stereovision," in *Proc. IEEE Intell. Veh. Symp.*, 2003, pp. 31–36.

[13] X. Du, K. K. Tan, and K. K. K. Htet, "Vision-based lane line detection for autonomous vehicle navigation and guidance," in *Proc. IEEE 10th ASCC*, 2015, pp. 1–5.

[14] X. Du and K. K. Tan, "Vision-based approach towards lane line detection and vehicle localization," *Mach. Vis. Appl.*, vol. 27, no. 2, pp. 1–17, Feb. 2016.

[15] D. Topfer, J. Spehr, J. Effertz, and C. Stiller, "Efficient road scene understanding for intelligent vehicles using compositional hierarchical models," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 441–451, Feb. 2015.

[16] C. Rose, J. Britt, J. Allen, and D. Bevly, "An integrated vehicle navigation system utilizing lane-detection and lateral position estimation systems in difficult environments for GPS," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2615–2629, Dec. 2014.

[17] Q. Li, L. Chen, M. Li, S.-L. Shaw, and A. Nuchter, "A sensor-fusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 540–555, Feb. 2014.

[18] N. Lazaros, G. C. Sirakoulis, and A. Gasteratos, "Review of stereo vision algorithms: From software to hardware," *Int. J. Optomechatron.*, vol. 2, no. 4, pp. 435–462, 2008.

[19] N. Einecke and J. Eggert, "A two-stage correlation method for stereoscopic depth estimation," in *Proc. DICTA*, 2010, pp. 227–234.

[20] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Proc. ACCV*, 2010, pp. 1–14.

[21] Z. Zhang, X. Ai, and N. Dahnoun, "Efficient disparity calculation based on stereo vision with ground obstacle assumption," in *Proc. IEEE 21st EUSIPCO*, 2013, pp. 1–5.

[22] F. Oniga and S. Nedevschi, "Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection," *IEEE Trans. Veh. Technol.*, vol. 59, no. 3, pp. 1172–1182, Mar. 2010.

[23] X. Ai, Y. Gao, J. Rarity, and N. Dahnoun, "Obstacle detection using u-disparity on quadratic road surfaces," in *Proc. IEEE ITSC*, 2013, pp. 1352–1357.

[24] Z. Zhang, X. Ai, C. Chan, and N. Dahnoun, "An efficient algorithm for pothole detection using stereo vision," in *Proc. IEEE ICASSP*, 2014, pp. 564–568.

[25] R. Labayrade, D. Aubert, and J.-P. Tarel, "Real time obstacle detection in stereovision on non flat road geometry through v-disparity representation," in *Proc. IEEE Intell. Veh. Symp.*, 2002, vol. 2, pp. 646–651.

[26] Y. Gao, X. Ai, Y. Wang, J. Rarity, and N. Dahnoun, "U–V disparity based obstacle detection with 3D camera and steerable filter," in *Proc. IEEE IV*, 2011, pp. 957–962.

[27] S. Nedevschi *et al.*, "3D lane detection system based on stereovision," in *Proc. IEEE 7th Int. Conf. Intell. Transp. Syst.*, 2004, pp. 161–166.

[28] D. Demirdjian and T. Darrell, "Motion estimation from disparity images," in *Proc. IEEE 8th ICCV*, 2001, vol. 1, pp. 213–218.

[29] A. Wedel, H. Badino, C. Rabe, H. Loose, U. Franke, and D. Cremers, "B-spline modeling of road surfaces with an application to free-space estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 572–583, Dec. 2009.

[30] M. Aly, "Real time detection of lane markers in urban streets," in *Proc. IEEE Intell. Veh. Symp.*, 2008, pp. 7–12.

[31] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *Int. J. Comput. Vis.*, vol. 61, no. 1, pp. 55–79, 2005.
[32] A. D. Sappa, D. Gerónimo, F. Dornaika, and A. López, "Real time vehicle pose using on-board stereo vision system," in *Image Analysis and Recognition*. New York, NY, USA: Springer, 2006, pp. 205–216.
[33] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, 2013.

**Umar Ozgunalp** received the B.Sc. degree in electrical and electronics engineering from Eastern Mediterranean University, Famagusta, Cyprus, in 2007 and the M.Sc. degree in electronic communications and computer engineering from University of Nottingham, Nottingham, U.K., in 2009. He is currently working toward the Ph.D. degree with the Visual Information Laboratory, Department of Electrical and Electronic Engineering, University of Bristol, Bristol, U.K.

His research interests include computer vision and pattern recognition.

**Rui Fan** received the B.S. degree in information and electrical engineering from Harbin Institute of Technology, Harbin, China, in 2015. He is currently working toward the M.S. degree with the Visual Information Laboratory, Department of Electrical and Electronic Engineering, University of Bristol, Bristol, U.K.

His research interests include real-time digital signal processing, lane detection, and stereo vision.

**Xiao Ai** received the B.Sc. degree in electrical and electronics engineering from University of Bristol, U.K., in 2007 and he received the Ph.D. degree in electrical and electronics engineering from University of Bristol, U.K., in 2012.

He is a Postdoctoral Researcher with University of Bristol, Bristol, U.K. His Ph.D. specialized in 3-D imaging techniques and applications. His current research interests include embedded real-time signal processing, optoelectronics for spaceborne remote sensing, and automotive obstacle detection applications. He also has extensive experience in machine vision.

**Naim Dahnoun** received the Ph.D. degree in biomedical engineering from University of Leicester, Leicester, U.K., in 1990.

He was with the Leicester Royal Infirmary as a Researcher on blood flow measurements for femoral bypass grafts and then with University of Leicester as a Lecturer in digital signal processing (DSP). In 1993, he started new research in optical communication at the University of Manchester Institute of Science and Technology, Manchester, U.K., on wideband optical communication links before joining the Department of Electrical and Electronic Engineering, University of Bristol, Bristol, U.K., in 1994, where he is a Reader in Learning and Teaching of DSP. His main research interests include real-time digital signal processing applied to biomedical engineering, video surveillance, automotive, and optics.

In 2003, in recognition of the important role played by universities in educating engineers in new technologies such as real-time DSP, Texas Instruments (NYSE:TXN) presented the first Texas Instruments DSP Educator Award to Dr. Dahnoun for his outstanding contributions to furthering education in DSP technology.