A Robust Lane Detection Method Based on Vanishing Point Estimation Using the Relevance of Line Segments

Ju Han Yoo, *Student Member, IEEE*, Seong-Whan Lee, *Fellow, IEEE*, Sung-Kee Park, *Member, IEEE*, and Dong Hwan Kim, *Member, IEEE*

***Abstract*— In this paper, a robust lane detection method based on vanishing point estimation is proposed. Estimating a vanishing point can be helpful in detecting lanes, because parallel lines converge on the vanishing point in a projected 2-D image. However, it is not easy to estimate the vanishing point correctly in an image with a complex background. Thus, a robust vanishing point estimation method is proposed that uses a probabilistic voting procedure based on intersection points of line segments extracted from an input image. The proposed voting function is defined with line segment strength that represents relevance of the extracted line segments. Next, candidate line segments for lanes are selected by considering geometric constraints. Finally, the host lane is detected by using the proposed score function, which is designed to remove outliers in the candidate line segments. Also, the detected host lane is refined by using inter-frame similarity that considers location consistency of the detected host lane and the estimated vanishing point in consecutive frames. Furthermore, in order to reduce computational costs in the vanishing point estimation process, a method using a lookup table is proposed. Experimental results show that the proposed method efficiently estimates the vanishing point and detects lanes in various environments.**

***Index Terms*— Lane detection, line segment, vanishing point estimation, probabilistic voting, lane departure warning (LDW) system.**

I. Introduction

IN THE research of advanced driver assistance systems, there has been a growing interest in techniques that can improve driver safety [1]–[3]. Of these techniques, a lane departure warning (LDW) system alerts a driver if the vehicle deviates from a lane or a narrow road on arterial roads and freeways, and it operates to prevent unnecessary alerts only when the vehicle goes above a certain speed. The LDW system can help to reduce vehicle crashes that are caused by careless or drowsy driving. There has been much research on vision-based lane detection for the LDW system [4]–[10]. In these lane detection methods, color or edge information is utilized as a feature of the lane.

Various researches have been presented on the detection of lanes using color cues [11], [12], because lanes and roads have contrasting colors. However, it is difficult to discriminate them by using color differences between lanes and roads in various environments such as changing illumination, back-light, nighttime, and rainy conditions. Therefore, color cues have been limitedly used to detect roads because variations in the color on the roads are smaller than those on the lanes [13], [14].

One way to overcome this disadvantage of using the color cues for lane detection is to employ edge cues. Liu *et al.* [15] have detected lanes by using the Hough transform technique. However, their method has a drawback that the false positive rate can be high in an image with many spurious lines extracted from various safety markers on the road, from shadows, and so on. Therefore, other methods [16], [17] apply a ridge feature to detect lanes by using dark-bright-dark patterns in the lanes. Kang *et al.* [17] have proposed a multi-lane detection method that uses the ridge feature and the inverse perspective mapping (IPM) which is widely used to detect lanes because it can remove the perspective distortion on lines that lie in parallel in the real world [10], [18]–[20]. The IPM transforms an image from a camera view to a bird's eye view by using camera parameters. Thus, lanes are presented perpendicularly and have the same width in the transformed image, where simple filters or geometric constraints can be used to detect lanes. In the IPM-based technique, however, the effectiveness of mapping is reduced if there are obstacles in the road. In addition, the accuracy of mapping can decrease because vibrations occur in an image when a vehicle is in motion.

Unlike the approaches mentioned above, methods using a vanishing point have been applied to lane detection. Since lanes are parallel to each other, they pass through their vanishing point in the image plane. Using this property, the false detections can be decreased by filtering out lines that do not pass through the vanishing point. Therefore, the vanishing point has been widely used to detect lanes [21] and roads [22]–[24]. Furthermore, it can be used in various mobile applications where parallel lines detection plays an important

Manuscript received May 12, 2016; revised November 27, 2016 and February 21, 2017; accepted March 2, 2017. Date of publication March 28, 2017; date of current version December 7, 2017. This work was supported by the National Research Council of Science & Technology through the Korea Government (MSIP) under Grant CRC-15-04-KIST. The Associate Editor for this paper was J. Zhang. *(Corresponding authors: Seong-Whan Lee and Dong Hwan Kim.)*

J. H. Yoo is with the Department of Computer Science and Engineering, Korea University, Seoul 136-713, South Korea (e-mail: yoojuhan@kist.re.kr).

S.-W. Lee is with the Department of Brain and Cognitive Engineering, Korea University, Seoul 136-713, South Korea (e-mail: sw.lee@korea.ac.kr).

S.-K. Park and D. H. Kim are with the Center for Robotics Research, Korea Institute of Science and Technology, Seoul 136-791, South Korea (e-mail: skee@kist.re.kr; gregorykim@kist.re.kr).

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.2017.2679222

role, such as corridor detection [25]–[28], power transmission lines inspection [29]–[31], 3D reconstruction [32], [33], and so on.

In order to extract a vanishing point exactly, first, lines should be extracted correctly. However, it is difficult to extract lines exactly from an image because of noises. Grompone von Gioi *et al.* [34] have introduced a line segment detector (LSD) that considers a line segment as a rectangle that consists of aligned points with similar gradient angles. Each rectangle is validated by using the a contrario approach [35]. Also, their method provides a good false detection control. They show that the LSD method is not only robust to noise but also fast in their paper. Because of this advantage, their method has been used to detect lanes [36], [37]. In the proposed method, the LSD method is applied to estimate the vanishing point and to detect lanes by considering the relevance of line segments.

In this paper, we assume that a monocular camera is front-mounted in the middle of the car and that the road is flat for a short distance in front of the camera. Then, in the short distance area corresponding to the bottom region of the image, occlusion by other vehicles does not occur generally because vehicles keep a safe distance from each other. In the proposed lane detection algorithm, line segments are first extracted by using the above mentioned LSD method in an image and then the vanishing point of the lanes is estimated by using a probabilistic voting procedure. The voting function is defined based on the line segment strength that represents relevance of the extracted line segments. Next, lanes are detected by finding peaks in the score function that measures how high a test line passing through the vanishing point has directional support from the line segments. Furthermore, because the voting function has high computational costs in the proposed algorithm, a method that approximately computes the voting function by using a lookup table which is constructed in advance is proposed to reduce computational costs. However, lanes may still not be detected accurately in roads where there are noises such as various pavement markings, repaired marks, superannuated lane markings, and so on. These noises can result in intermittent lane detection failures. Therefore, in order to reduce these failures, we utilize inter-frame similarity that considers location consistency of the detected lanes and the estimated vanishing point in consecutive frames. Finally, the proposed method detects the host lane by using the inter-frame similarity.

The main contributions of this paper are threefold: 1) A new vanishing point estimation method robust to noise is proposed. The proposed probabilistic voting function is defined with line segment strength that represents relevance of extracted line segments. 2) A lookup table-based voting algorithm is proposed to reduce computational costs of the voting function. 3) The host lane is detected efficiently by finding peaks in the proposed score function based on geometric relationships between detected parallel lines and their estimated vanishing point. Then, it is refined by using inter-frame similarity that considers location consistency of the detected lanes and the estimated vanishing point in consecutive frames.

The remainder of this paper proceeds as follows. Section II reviews the previous works on lane detection. Section III explains the proposed lane detection method and Section IV describes an algorithm improving the accuracy of the proposed lane detection method by using consecutive frames. Experimental results are presented in Section V and Section VI describes applications for the proposed method. Finally, conclusions are given in Section VII.

II. Related Work

The IPM-based methods have been widely used to detect lanes. Aly [19] transformed an input image into an IPM image and then vertical direction lines are extracted by using a simplified Hough transform. Finally, lanes are estimated by utilizing the RANSAC spline fitting method. Borkar *et al.* [10] also used the IPM technique to estimate lanes. First, the IPM image of an input image is converted into a binary image by using the adaptive threshold method and then the lane marker candidates are selected by utilizing predefined lane templates. Then, outliers are eliminated by using RANSAC and lanes are tracked by using a Kalman filter. Recently, Kang *et al.* [17] have proposed a detection method for multi-lanes that are parallel to each other. They first extracted ridges on a road as lane features. To remove the perspective effect, they transformed the lane features from image coordinates to IPM coordinates. Then, a histogram of x coordinate values for each lane feature has four local maxima because, in the IPM image, the lane features that are extracted from the same lane are distributed with a small variance on the x axis. Therefore, the lanes are detected by clustering the lane features around each local maximum point. Similarly to [17], Hur *et al.* [38] have proposed a multi-lane detection method. However, they did not use the IPM technique. Instead, they extracted the lane features directly from an input image. They detected not only parallel lanes but also various nonparallel lanes, such as intersections and splitting and merging lanes, in urban environments. In their method, lane features are extracted by using the ridge property and are clustered into a small group. This group is described by a set of coefficients which are estimated by using conditional random fields to robustly detect multiple lanes.

On the other hand, vanishing point-based methods also have been used to detect lanes or roads to reduce a false detection rate. Wang *et al.* [21] have proposed a lane detection algorithm based on the B-Spline snake model. To initialize this model and track lanes, the vanishing point is estimated by using a Canny/Hough estimation of vanishing points (CHEVP). However, this method has a disadvantage of having a high false detection rate under shadowy or illuminated conditions. Rasmussen [39] has introduced an ill-structured road detection method. First, they computed texture orientation for all pixels of an input image by using multi-scaled Gabor filters. Then, the vanishing point is estimated by voting in a predefined area, and the estimated vanishing point is tracked from frame to frame. However, this method has a drawback that the vanishing point can not be estimated exactly in case its location is rapidly changed in each consecutive frame. In Kong *et al.*'s method [22], texture orientation using Gabor filters was applied to lane detection. They defined

the confidence score of the texture orientation, and estimated a vanishing point by using an adaptive soft voting scheme in a local voting area using high confidence voters. Moghadam and Dong [40] have proposed a vanishing point estimation method that can be applied to structured and unstructured roads. Their method computes a dominant orientation of a pixel by using four Gabor filters, which is similar to Rasmussen's method [39] and Kong *et al.*'s method [22]. Then, Moghadam and Dong [40] defined rays of each pixel by using these dominant orientations, and carried out a voting procedure by using the relationship between a pixel and a ray. Finally, a pixel with a maximum number of votes is selected as a vanishing point.

As mentioned above, these methods calculate the orientation of pixels by using a Gabor filter, and then estimate the vanishing point by using various voting functions. Unfortunately, in their methods, false detections increase considerably in noisy images. For robustness to noise, Liu *et al.* [36] applied the LSD method to detect lanes. They first defined a region of interest (ROI) where the lanes are located, and extracted line segments by using the LSD method in this region. These line segments are filtered out by utilizing orientation constraints because they may include line segments extracted from non-lane markings. Then, out of intersection points of pairs of line segments, a point with maximum intersection number is selected as a vanishing point of lanes. Finally, the host lane is detected by considering a geometric distance between the vanishing point and the reserved line segments. However, they did not consider relevance of line segments, and so their method may yield an unsatisfactory performance in noisy environments because irrelevant line segments are extracted from noisy areas or background clutter. Similar to the proposed vanishing point estimation method, Yuan *et al.* [41] have introduced a vanishing point estimation method with a probabilistic framework. They converted estimation of a vanishing point in the image space into estimation of line parameters in the parameter space. They first extracted straight lines by using the progressive probability hough transform and then the maximum a posteriori was used to estimate a vanishing point of the lines. However, their method has a disadvantage that it is difficult to estimate the vanishing point when a variation of lines between consecutive frames is large, and that it also needs a separate process to initialize the parameters.

Recently, Jung *et al.* [9] have proposed a lane detection method using spatiotemporal images. The spatiotemporal image is obtained from video and generated by accumulating a set of pixels which are extracted on a horizontal scan-line with a fixed location in each frame along a time axis. Then, in the spatiotemporal image, lanes are detected by using hough transform. Their method is robust to short-term noises such as missing lanes or occlusion by cars, however, not only using the spatiotemporal image needs a period of time to accumulate the pixels but also it is not easy to detect lanes accurately on a road with various pavement markings because they use heuristic constraints to find an initial position of lanes. Gu *et al.* [42] enhanced lanes in an image by using an extremal-region enhancement kernel and select lane candidates in the edge map that is obtained by applying the canny edge detector and the hough transform. Finally, lanes are detected by using a rule-based probability. They presented a simple and efficient method for detecting lanes, however, their method is not robust to scenes with broken lane markings, shadows, or various pavement markings.

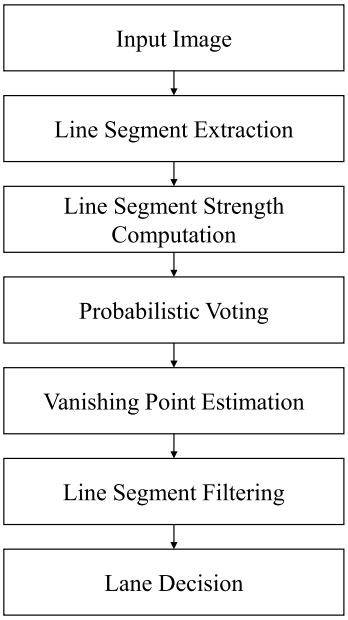

Fig. 1. Block diagram of the proposed methods.

III. Lane Detection

An overview of the proposed lane detection method is shown in Fig. 1. In order to detect lanes, line segments are first extracted by using the LSD method [34] on the input image, and the line segment strength is computed for each line segment. Then, a vanishing point of lanes is estimated by using a probabilistic voting procedure with the line segment strength. The extracted line segments are filtered out by considering whether their orientations are similar to those of the lanes or not, then reserved line segments are selected as candidate line segments. Finally, lanes are detected from those candidate line segments.

A. Vanishing Point Estimation

Parallel lines in a real world environment meet at a single point in an image because of the perspective effect that occurs in the process of projection from a 3D to a 2D image. This point is called the vanishing point. The vanishing point of lanes can be calculated as an intersection point of a pair of line segments extracted from lanes. The vanishing point can also be used to find lines parallel to the lanes. Therefore, if the vanishing point of line segments parallel to lanes is found, it can be helpful in detecting lanes.

In the image, a line segment consists of aligned pixels. Errors that occur during the line segment extraction process—pixel alignment error caused by curves, clutter around a line, and so on—make it difficult to estimate the vanishing point exactly. As shown in Fig. 2, the intersection points of line segments that are extracted from the lanes do not meet at a

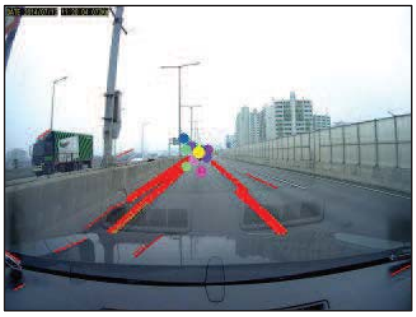

Fig. 2. Example of line segments and intersection points. The circles of various colors are intersection points of line segments (red). The yellow circle is a vanishing point of the lanes.

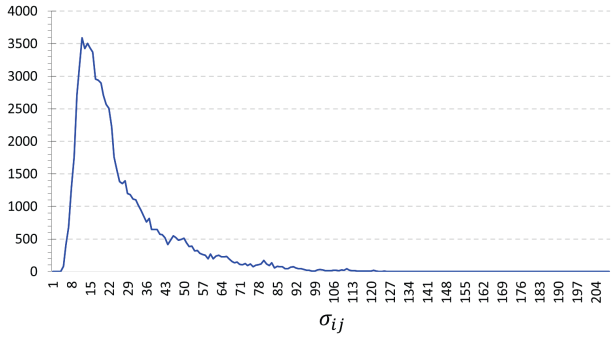

Fig. 3. Histogram of standard deviations for intersection points of pairs of line segments extracted in real road images.

single point because of the pixel alignment error. Therefore, this paper first proposes a robust vanishing point estimation method using a probabilistic voting procedure.

In order to consider the pixel alignment error, the line segment strength τ, which represents how well orientations of pixels forming a line segment are aligned, is introduced in the proposed method. τ is defined to assign higher values to longer and well-aligned sharper line segments, as follows:

$$\tau_i = \frac{l_i}{w_i}, \tag{1}$$

where l_i is the length and w_i is the width of a line segment extracted by the LSD method [34]. Line segments with higher τ values are considered more relevant. Line segments whose orientations are similar to those of the lanes are shown in Fig. 2. The thickness of the line segments represents the line segment strength.

For a pair of line segments (L_i, L_j), the intersection point of (L_i, L_j) is considered to be a Gaussian distribution in an image plan Ω, as follows:

$$P_{V_{ij}}(x, y; m_{ij;x}, m_{ij;y}, \sigma_{ij;x}, \sigma_{ij;y}) = \frac{1}{2\pi\sigma_{ij;x}\sigma_{ij;y}} e^{-[\frac{(x-m_{ij;x})^2+(y-m_{ij;y})^2}{2\sigma_{ij;x}\sigma_{ij;y}}]}, \tag{2}$$

where $(x, y) \in \Omega$, $(m_{ij;x}, m_{ij;y})$ is the intersection point of (L_i, L_j) in image coordinates, and $\sigma_{ij;x}$ and $\sigma_{ij;y}$ are standard deviations along the x-axis and y-axis, respectively. In this equation, the Gaussian model is assumed to be isotropic, and then

$$\sigma_{ij} = \sigma_{ij;x} = \sigma_{ij;y} = \sqrt{\sigma_i^2 + \sigma_j^2}, \tag{3}$$

where $\sigma_i = \alpha \cdot (1/\tau_i)$ and α is a scale factor. This Gaussian model has a sharper and narrower distribution for a pair of (L_i, L_j) with higher line segment strength values, and assigns higher probabilities of being the vanishing point to the area around the intersection point. The overall voting function to estimate the vanishing point is as follows:

$$P_W(x, y) = \sum_{i=0}^{N-2} \sum_{j=i+1}^{N-1} P_{V_{ij}}(x, y; m_{ij;x}, m_{ij;y}, \sigma_{ij}), \tag{4}$$

where N is the number of line segments. Finally, the vanishing point of the lanes can be estimated as follows:

$$V(x, y) = \arg\max_{x,y}[P_W(x, y)]. \tag{5}$$

B. Approximate Computation for the Voting Function

In our previous work [43], the probabilistic voting process is computationally expensive, because it computes Gaussian distribution $P_{V_{ij}}$ for all pixels in a voting area for the intersection points of all pairs of line segments. However, the distribution of $P_{V_{ij}}$ is determined by only σ_{ij} with the line segment strength, so that $P_{V_{ij}}$ can be constructed in advance as a lookup table by using a predefined σ_{ij}. Using the lookup table leads to a reduction in computational costs, because it is not necessary to compute the Gaussian function for all pixels in the voting area. More information about the efficiency of using the lookup table will be presented in the experimental results. First, an approximate computation method using the lookup table for P_W is explained.

Let $\widetilde{\Omega}_{ij}$ be a subset of image plane Ω. It is an M by M square image with a center point $(m_{ij;u}, m_{ij;v})$. The lookup table $T_{ij}(u, v, \sigma_{ij})$ can be expressed as follows:

$$T_{ij}(u, v, \sigma_{ij}) \equiv P_{V_{ij}}(x, y; m_{ij;x}, m_{ij;y}, \sigma_{ij}), \tag{6}$$

where $(u, v) \in \widetilde{\Omega}_{ij}$, $u = x - m_{ij;x}$ and $v = y - m_{ij;y}$. Then, (4) is represented by

$$\widetilde{P}_W(x, y) = \begin{cases} \sum_i \sum_j T_{ij}(u, v, \sigma_{ij}) & (x, y) \in \widetilde{\Omega}_{ij} \\ 0 & otherwise. \end{cases} \tag{7}$$

When the lookup table is constructed, a range of σ_{ij} must be defined. If an input image has a size of $R \times S$, l_i and w_i are in the range of 1 to $\sqrt{R^2 + S^2}$, and 1 to $\sqrt{RS}$, respectively [34], then, σ_{ij} is in the range of $\alpha\sqrt{2/(R^2 + S^2)}$ to $\alpha\sqrt{2RS}$. However, it is inefficient to compute the lookup table for all σ_{ij}. Therefore, in this paper, the range of σ_{ij} is determined experimentally. Fig. 3 shows a histogram of standard deviations for all intersection points of line segments where each σ_{ij} is measured in a real road environment. The line segments are extracted from a total of 923 images acquired by the camera on arterial roads. The number of intersection points is approximately 77000, and the bin size

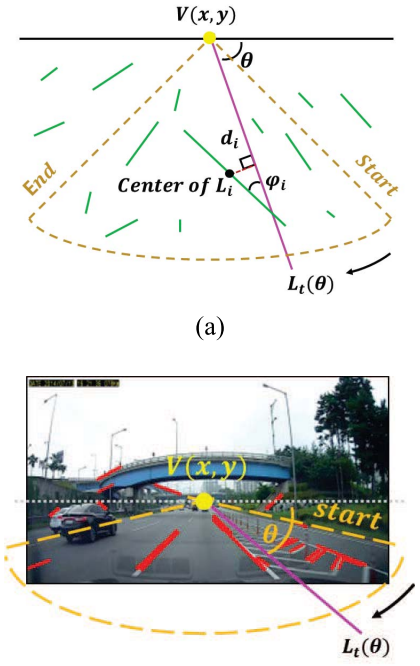

Fig. 4. The proposed line segment filtering: (a) parameters to select candidate line segments, (b) searching candidate line segments in a fan-shaped area.

is set to 1. In Fig. 3, it is observed that σ_{ij} is only increased to about 150 in a real road environment. Therefore, in this paper, the lookup table for σ_{ij} with a range from 0 to 150 is constructed. A typical values for M is 200.

C. Detection of Lanes

When the vanishing point of the lanes is estimated, line segments that pass through the vanishing point can be found easily in an image. Most of the line segments are from the lanes, however, they may contain line segments that are not parallel to lanes in the 3D space but pass through the vanishing point by chance in the projected 2D image. In this paper, therefore, a robust lane detection method is proposed to reduce the influence of these outlier line segments.

1) Filtering: In this step, line segments that do not converge to the vanishing point are first filtered out because they are not parallel to lanes and therefore can not be lanes. In an image obtained from the proposed lane detection method, the lanes are located in a fan-shaped area centered on the estimated vanishing point, as shown in Fig. 4(a) and (b). This property can help lanes to be efficiently detected by searching only in the fan-shaped area. First, a test line $L_t(\theta)$ is defined as a line at an angle θ that rotates clockwise around the vanishing point with an angle step of $\Delta\theta$, as shown in Fig. 4(a). d_i is the shortest distance from $L_t(\theta)$ to a center point of L_i, and φ_i is the acute angle between L_i and $L_t(\theta)$. In this paper, both distance and angle are employed together because the direction of each line segment can be arbitrary even though d_i is close to zero. Finally, line segments are selected using the following constraints: $d_i < d_t$ and $\varphi_i < \varphi_t$, where d_t and φ_t are threshold values. Line segments that satisfy these constraints are selected as candidate line segments for lanes, and are considered further in the following decision step. Fig. 4(b) shows the process of searching the candidate line segments in a fan-shaped area of the input image.

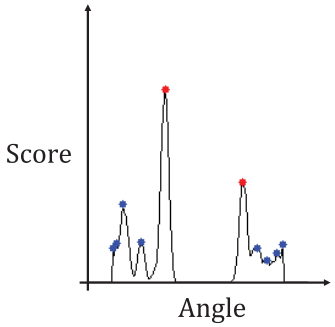

Fig. 5. Score function for lane detection.

2) Decision: For lane detection, candidate line segments that converge at a vanishing point are selected. The selected candidate line segments, however, can contain line segments that are extracted from the surrounding environment and are not parallel to lanes in the 3D space. Since these outliers can affect lane detection, in this step, a lane decision method robust to outlier line segments based on a score function is proposed, and the score function is defined as follows:

$$S_{L_t(\theta)} = \sum_i \tau_i \exp(-d_i \sin(\varphi_i)). \tag{8}$$

$S_{L_t(\theta)}$ has a higher value if d_i and φ_i have lower values or τ_i has a higher value. Therefore, a high score value means that the probability of the overlap between $L_t(\theta)$ and L_i extracted from the lanes is high. Then we can consider that a set of $L_t(\theta)$ with higher scores corresponds to lanes. Fig. 5 shows a plot of $S_{L_t(\theta)}$ with respect to each angle value within the fan-shaped region. However, the local peaks (shown as blue points) can make it difficult to find the lanes correctly. Therefore, $S_{L_t(\theta)}$ is smoothed by mean filtering for noise reduction, and the flooding watershed algorithm [44] is applied to find two peaks (shown as red points) that correspond to the host lane.

IV. Efficient Consecutive Lane Detection

Generally, lane detection should be carried out for all consecutive frames of a video sequence. In a road image, it can contain various pavement markings, repaired marks, shadows of trees or buildings and so on. They can be seen around lanes, which can cause a problem in detecting lanes if they occlude lanes or have orientations that are similar to those of lanes. For a certain image with superannuated or broken lanes that do not appear in the image, lane detection sometimes can not be possible.

Therefore, there have been methods developed to detect lanes by considering consecutive frames [21], [45], [46]. Wang *et al.* [21] detected lanes by using a lane model based on B-spline curves with some control points. They iteratively update parameters of the points in order to increase the lane detection accuracy until differences between parameters in t and $t-1$ frame are smaller than a pre-defined threshold value. Ruyi *et al.* [45] have proposed a lane detection method using a particle filter and a modified Euclidean distance transform.

Algorithm 1 Algorithm for the Proposed Lane Detection in Consecutive Frames

Data: Input image I_t
Output: Estimated $\tilde{V^t}$ and $\tilde{\theta^t}$
1: **for** $t = 1, 2, \ldots, T$ (T is the number of frames) **do**
2: (V^t, θ^t) = Detection(I_t);
3: Validation(V^t, θ^t);
4: (E_V, E_θ) = Update(Q_V, Q'_V, Q_θ);
5: $\tilde{V^t} \leftarrow E_V$;
6: $\tilde{\theta^t} \leftarrow E_\theta$;
7: **end for**

For a lane detection robust to complex road situations, they introduced some lane point sets: one is predicted from $t-1$ frame, and the others consist of points that are estimated from t frame. The reliability of the sets is estimated by using maximum likelihood estimation and the final lane point set is selected as one with the largest reliability value. Wu *et al.* [46] have improved the lane detection accuracy by using a Kalman filter. They assumed that lanes do not substantially change between consecutive two frames and then the lane in t frame is estimated by using the detected lane in $t-1$ frame.

Similar to above mentioned methods, we propose an efficient lane detection method using consecutive frames. The proposed method detects lanes finally by considering inter-frame similarity for location consistency of the detected lanes and the estimated vanishing point in consecutive frames. This can help to compensate for intermittent lane detection failure and result in a low false positive rate. The framework of the proposed consecutive lane detection method can be found in Algorithm 1. As mentioned in Section III, V and θ denote a location of the vanishing point and angles of the host lane, respectively. For the sake of convenience, we consider that θ means both θ_{left} and θ_{right}, which are angles of left and right boundary of the host lane, respectively. Note that θ_{left} and θ_{right} are computed independently of each other. In order to verify consecutive detection of lanes, three queues, Q_V, Q'_V, and Q_θ are used in the proposed method. The queue has a FIFO(first in first out) structure and it is appropriate to decide a partial continuity of data in consecutive frames. Note that two queues, Q_V and Q'_V, are used in terms of V, while only one queue, Q_θ is used with regard to θ. Let's denote Q_V as a queue for the vanishing point, and Q_θ as a queue for lane angles, then a mean and a standard deviation of Q_V are defined as E_V and σ_V, also a mean of Q_θ is defined as E_θ.

The basic functions inside the main loop are:

- $Detection()$: Lane detection is carried out in t frame by using the proposed lane detection method as mentioned in Section III. It yields a location of the vanishing point and angles of the host lane, then they are stored in the queues.
- $Validataion()$: This function validates whether a vanishing point in t frame, V^t, and lane angles in t frame, θ^t, are reliable or not. Not only V^t is inserted into Q_V but also Q'_V is cleared if $||V^t - E_V||_2$ is smaller than κ_v, and V^t is inserted into Q'_V otherwise. Also, θ^t is inserted into Q_θ if θ^t satisfies the following constraints, $\beta_1 < \theta^t_{left} < \beta_2$ and $\beta_3 < \theta^t_{right} < \beta_4$ where β_1, β_2, β_3, and β_4 are constants.
- $Update()$: This function updates queues in order to estimate a vanishing point and lane angles. Q_V is replaced with Q'_V if the size of Q'_V is bigger than κ_n and σ'_V of Q'_V is smaller than κ_v, and Q'_V is cleared otherwise. Finally, E_V and E_θ are computed.

where $\beta_1 = 125°$, $\beta_2 = 150°$, $\beta_3 = 30°$, and $\beta_4 = 55°$. Also, κ_v and κ_n are user-defined thresholds and are set to 5 and 3, respectively.

V. Experimental Results

A. Detection of Lanes

To evaluate the performance of the proposed method, experiments were carried out on five categories in our database and other lane detection methods [9], [36], [42] were tested on our database to compare with our lane detection method. Also, in order to compare Aly's lane detection method with our lane detection method, experiments were carried out on four categories in the Caltech database [19]. Our code and database used in the experiments are available on our website: https://sites.google.com/site/juju1006s.

1) Our Database: The database contains consecutive images (reduced to 320×240) from five different types of environments: in daytime, in a tunnel, in a rainy day, in a backlight, and in nighttime (from top to bottom rows in Fig. 6). They are acquired by a front-mounted camera in the car on arterial roads and consist of 525, 676, 222, 359, and 263 images, respectively. For the daytime sequence, it is obtained in the daytime and includes scenes with large illumination changes or shadows, where confusing line segments of which orientations are similar to those of lanes are extracted. The tunnel sequence consists of images with tunnel lighting with various colors and illumination changes at entry and exit of the tunnel. For the rainy day sequence, it includes scenes with lanes that are occluded by windshield wiper. The backlight sequence is obtained when the sun is directly in front of the car and the nighttime sequence consists of images with curves and car leaving road. Overall, our database includes scenes with various pavement markings, broken lane markings, shadows, and so on.

Fig. 6 depicts sample results of the proposed lane detection method for each sequence. The detected lanes are represented as straight lines and we set an effective detection range along the y axis in an image as 120 to 150 in order to remove a bonnet area in the bottom side of an image. Overall, it shows good lane detection performance, although there are vehicles around the host lane and are various pavement markings, broken lane markings, and shadows on the road. The last and second last columns illustrate cases that the proposed lane detection fails. Most of the failed cases are caused by shadows or repaired marks of which orientations are similar to those of the lanes on the road. Also, the lane detection fails when the line segment detector fails to extract line segments from lanes, which sometimes occurs due to deterioration of painted lane markings or broken lanes.

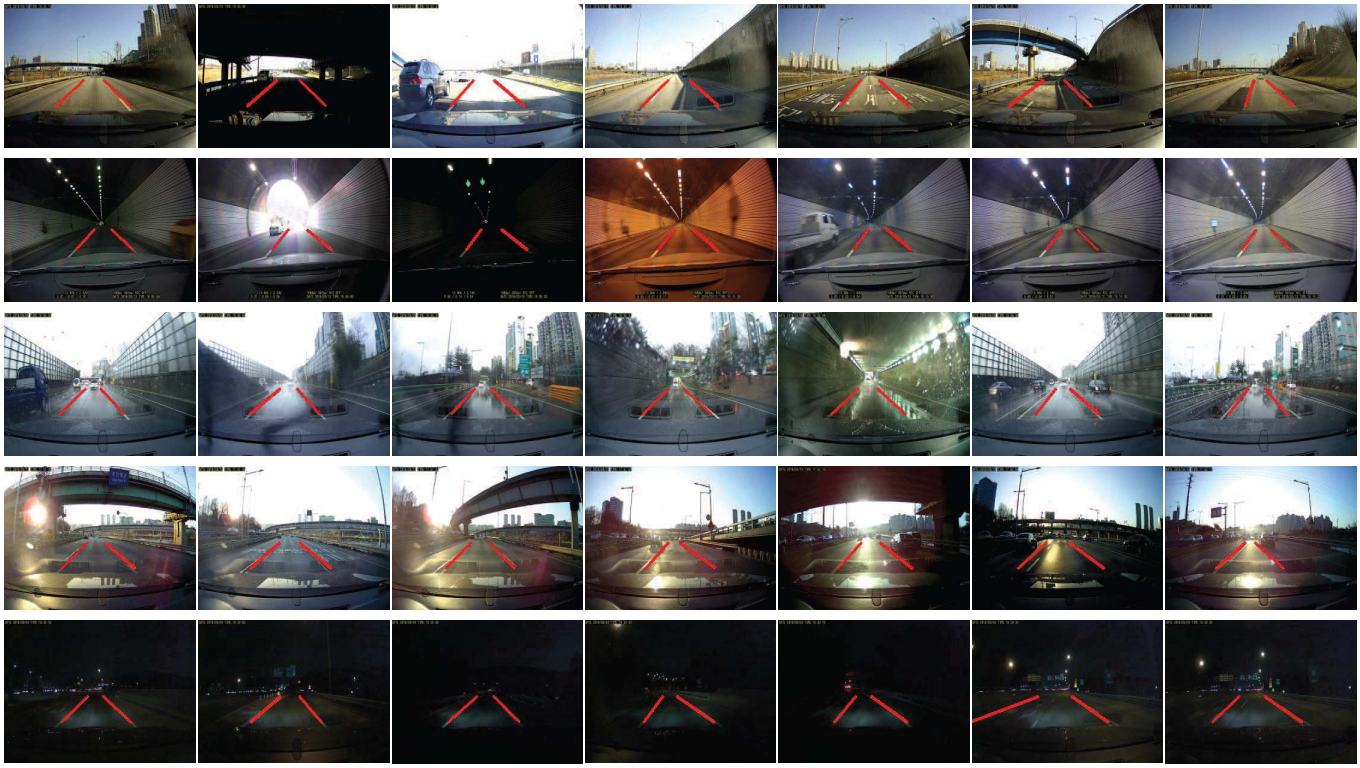

Fig. 6. Sample results of the proposed lane detection method. Red lines refer to the detected lanes.

For every frame, each detected lane is compared to manually drawn ground truth lanes and the following detection criteria are applied to determine if the lane detection succeeds or not. Let us denote a set of points on the detected and ground truth lanes as $D = \{p_1, p_2, \ldots, p_{n_D}\}$, and $G = \{q_1, q_2, \ldots, q_{n_G}\}$, respectively. For each point $p_i \in D$, we compute the nearest distance, d_i^p to G. Similarly, we do the same for $q_j \in G$ and get the nearest distance, d_j^q to D as follows:

$$d_i^p = \min_j^{n_G} \parallel p_i - q_j \parallel_2, \quad d_j^q = \min_i^{n_D} \parallel q_j - p_i \parallel_2 . \tag{9}$$

Then, mean distances $(\bar{d}^p, \bar{d}^q)$ and median distances $(\hat{d}^p, \hat{d}^q)$ are computed. In order to decide whether they are the same, we apply the following criteria as in [19]:

$$\min(\bar{d}^p, \bar{d}^q) < t_1, \quad \min(\hat{d}^p, \hat{d}^q) < t_2, \tag{10}$$

where t_1 and t_2 mean pre-defined thresholds. Finally, the lane detection rate, R is defined as follows:

$$R(\%) = \frac{m^l + m^r}{2T}, \tag{11}$$

where m^l and m^r denote the number of correctly detected left and right lanes, respectively. Also, T is the total number of images as in Algorithm 1. The parameter values used in this experiment are $\alpha = 100$, $d_t = 2$, $\varphi_t = 20$, and $t_1 = t_2 = 5$.

We have simulated three different lane detection methods [9], [36], [42] to evaluate the performance of the proposed lane detection method in terms of detection rate on our database. Liu *et al.*'s method [36] first sets a ROI before extracting line segments from an input image. In order to find the upper boundary of the ROI automatically, they use an assumption that the upper boundary locates at a row with the minimum mean intensity value in the input image. It may work well on their database [36] and Caltech database [19]. However, this assumption has a limitation that it can only be applied to images in which an average intensity of pixels on a vanishing line of lanes is smaller than average intensities of other rows. As shown in Fig. 7, for a sample image of our database, their method can not find the upper boundary of the ROI. Therefore, in order to exclude cases that their method fails to detect lanes due to a wrong ROI, we modified Liu *et al.*'s method to fix the upper boundary of the ROI and compared it with ours. In the modified Liu *et al.*'s method, the upper boundary of the fixed ROI is equal to that of our method.

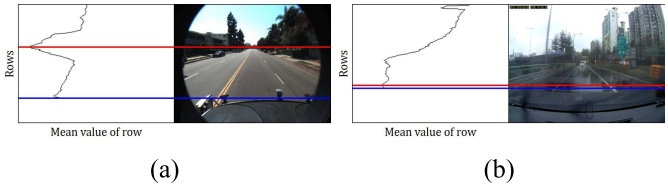

Fig. 7. Example results of the ROI extraction in [36]. The computed upper boundary and the fixed lower boundary of the ROI are represented by a red and a blue line, respectively. The left image in each result shows mean values of pixel intensities on each row in the input image [36]. (a) a success case of the ROI extraction on Caltech database [19], (b) a failed case of the ROI extraction on our database.

Table I presents the detection rates and Fig. 8 shows sample results of the lane detection on our database. Overall, the proposed method outperforms others. It is shown that there is

TABLE I
COMPARISON OF LANE DETECTION RATES(%) ON OUR DATABASE

	Liu *et al.*'s [36]	Modified Liu *et al.*'s [36]	Gu *et al.*'s [42]	Jung *et al.*'s [9]	Ours
Daytime	75.24	82.95	75.24	95.43	**96.95**
Tunnel	21.60	69.50	78.26	**99.93**	99.39
Rainy Day	33.78	72.97	56.76	92.12	**98.20**
Backlight	7.52	53.20	69.92	83.15	**91.50**
Nighttime	37.64	68.06	61.03	85.17	**98.48**

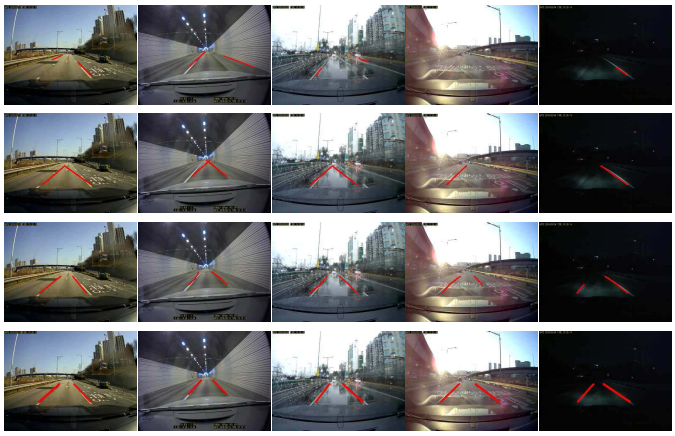

Fig. 8. Sample lane detection results. Modified Liu *et al.*'s, Gu *et al.*'s, Jung *et al.*'s, and ours (top to bottom).

a considerable difference between lane detection rates of the original Liu *et al.*'s method and ours. The proposed method shows a good performance with detection rates of over 90%, however, the detection rates of the original Liu *et al.*'s method are lower than 76% because there are cases that not only their assumption can not be efficiently applied to various road environments but also lanes and their vanishing point can not be estimated correctly due to heuristic constraints. Note that the lane detection rates of our method are still higher than those of the modified Liu *et al.*'s method.

In Gu *et al.*'s method [42], they enhance lines in intensity and simply select lines that are the closest to the middle of an image as lane candidates. However, lanes can not detected accurately in roads where there are noises such as various pavement markings, repaired marks, superannuated lane markings and so on. Consequently, the detection rates of their method are under 79% on our database including various road environments.

Jung *et al.*'s method [9] shows slightly better performance than ours in the tunnel sequence. In the other 4 different types of sequences, however, the proposed method outperforms theirs. They use a simple motion compensation to align lanes of consecutive frames because lanes are not located consistently in the spatiotemporal images. The compensation error, however, can be significantly increased when the time difference between frames is longer due to computation time or the image difference between frames is larger due to high speed of a car, which can yield inaccurate lane detection. Also, their method has a limitation that their heuristic constraints can not efficiently deal with various noisy environment such as backlight or pavement markings.

Fig. 9. Sample lane detection results on the Caltech database. The first and second rows are sample results of Aly's method [19]. The third and fourth rows are sample results of our method.

TABLE II
LANE DETECTION RATES(%) ON CALTECH DATABASE

	Aly's [19]	Ours
Cordova1	91.60	**94.40**
Cordova2	75.37	**87.44**
Washington1	**92.43**	89.61
Washington2	92.46	**98.28**

2) The Caltech Database: The Caltech database contains consecutive images (640×480) from four different types of environments: in cordova1, in cordova2, in washington1, and in washington2 (left to right in Fig. 9). They consist of images that are obtained in the daytime and include many shadows. Also, it is not easy to detect lanes exactly on their database because orientations of line segments extracted in the shadows are similar to those of lanes. The first and second rows in Fig. 9 show sample results of lane detection using Aly's method [19] and the detected lanes are represented by green color spline curves. The third and fourth rows in Fig. 9 depict the sample lane detection results using our lane detection method and the detected lanes are represented by red color line segments.

For comparison, we manually built ground truth lanes on the Caltech database and applied the same parameter values as used in experiments on our database. The parameter values for the detection criteria, t_1 and t_2, are set to 15 and 20, respectively. These values are the same as in [19]. The lane detection rates computed by (11) are summarized in Table II. Overall, our method performs better than Aly's method except for the washington1. The proposed method considers inter-frame similarity for location of the detected lanes and the estimated vanishing point in consecutive frames, whereas Aly's method detects lanes independently in each frame. Therefore, our method outperforms Aly's method on the Caltech database in which there are various shadows such as the cordova1, the cordova2, and the washington2. The inter-frame similarity,

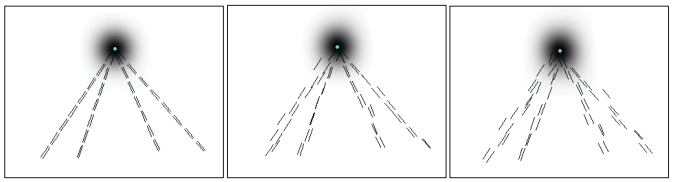

Fig. 10. Line segments whose endpoints are degraded by Gaussian noise with various standard deviations, and estimated vanishing points (cyan): (a) ground truth, (b) with a standard deviation of 3, (c) with a standard deviation of 5.

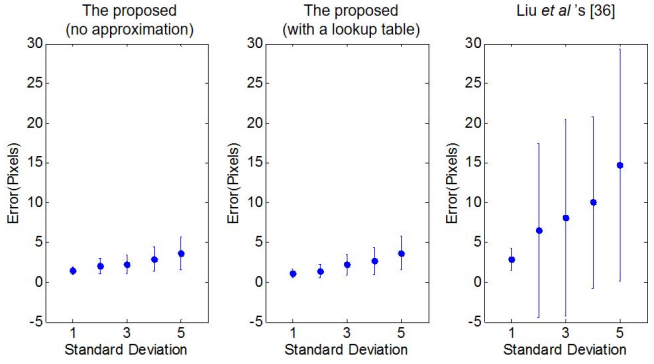

Fig. 11. Vanishing point estimation error on noisy line segment images.

however, can sometimes be disturbed in a sequence of changing lanes, and so it can yield slightly incorrect lane detection results. This occurred in the washington1 images and the proposed lane detection shows a performance of about 3% lower than that of Aly's. On average, lane detection rates of Aly's method and ours are 87.97% and 92.43%, respectively.

B. Analysis for Gaussian Noise

1) Vanishing Point Estimation Error: In order to evaluate the effectiveness of the proposed vanishing point estimation method, the vanishing point estimation was carried out on synthetic images with line segments whose endpoints are degraded by Gaussian noise with zero mean and various standard deviations denoted by σ_e, as shown in Fig. 10. In Fig. 10, intensity images represent the voting function, $P_W(x, y)$. The darker the pixel, the higher the probability of being the vanishing point. It is shown that the vanishing points are efficiently estimated on noisy line segments with a larger value of σ_e.

Fig. 11 shows the mean and variance of the position error of the estimated vanishing point when repeated 100 times for each σ_e of the Gaussian noise. In the proposed method without using a lookup table-based approximation, the estimated position error is less than about four pixels on average, although the standard deviation of the Gaussian noise increases to 5. The vanishing point estimation error of the proposed approach with a lookup table-based approximation is almost equal to that of the method without using the approximation. In Liu *et al.*'s method [36], however, the error increases to 15 pixels on average. From the results, we can see that the proposed vanishing point estimation method can efficiently

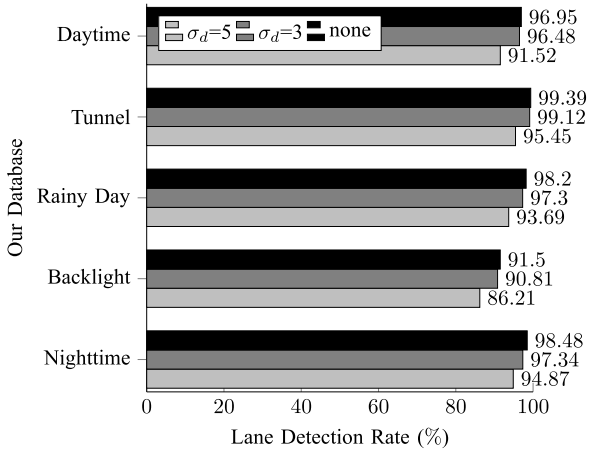

Fig. 12. Comparison of lane detection rates when the estimated vanishing point is corrupted by Gaussian noise.

Fig. 13. Example of salt-and-pepper noise: (a) original image, (b) image corrupted by salt-and-pepper noise.

estimate the vanishing point of the lanes even in noisy environments.

2) Lane Detection Error: We conduct experiments on our database to verify how well positions of lanes are estimated by the proposed lane detection method. Similar to the experiment in terms of vanishing point estimation error, lane detection was carried out on images with estimated vanishing points that are degraded by Gaussian noises with zero mean and various standard deviations denoted by σ_d. As shown in Fig. 12, the overall detection rates of the proposed lane detection method are over 90% although the detection rates are decreased significantly in case of adding Gaussian noise with a standard deviation of 5 to the vanishing point. This indicates that the proposed method can robustly detect the lanes, although estimated vanishing point of the lanes is severely distorted by Gaussian noise.

C. Analysis for Salt-and-Pepper Noise

The salt-and-pepper-like noise can be occurred in an image due to environmental factors such as rain, snow, or dust. Therefore, we demonstrate the effectiveness of the proposed lane detection method for salt-and-pepper noise. In order to do this, experiments were carried out on our database. Input images are corrupted by salt-and-pepper noise of which noise density is one percent. Fig. 13 presents a sample image with salt-and-pepper noise. As shown in Fig. 14, a difference in detection rates between with and without salt-and-pepper noise is less than about 2%. Also, it shows that the proposed method

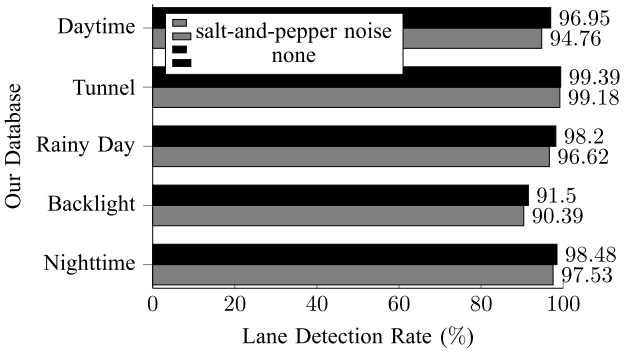

Fig. 14. Comparison of lane detection rates under salt-and-pepper noise.

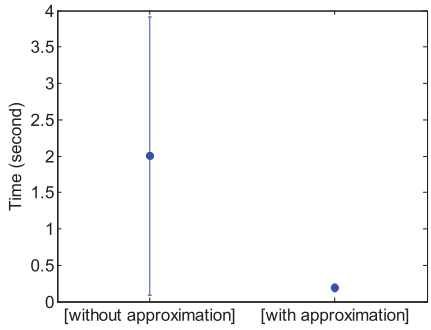

Fig. 15. Comparison of computation time of the proposed lane detection method.

yields satisfactory performance on rainy day images that are similar to images corrupted by salt-and-pepper noise.

D. Comparison of Computational Speed for a Lookup Table

To verify the computational efficiency of the proposed method with a lookup table-based approximation, the proposed method is compared to the method without using the approximation on our database. As shown in Fig. 15, the average computation time of the method without using the approximation is about 2 *sec* per image, whereas the proposed method with the approximation takes about 0.18 *sec*. In addition, the variance of the proposed method with the approximation is about 0.05 *sec*, however, the variance of the method without using the approximation is about 1.9 *sec*. The experimental result shows that the proposed method with the approximation is not only more stable because the variation of its computation time is smaller than that of the method without using the approximation but also faster than the method without using the approximation. All of the experiments were performed on a system with a 4.0 GHz CPU and 32 GB RAM.

VI. Applications

The proposed method extracts parallel lines by using a score function based on geometric relationships between lines and their estimated vanishing point, then lanes are detected by considering the inter-frame similarity for location of the detected lanes and the estimated vanishing point in consecutive frames. Consequently, we can take advantage of a parallel lines detection robust to noisy environments. Therefore, this framework can be used in various applications where detecting parallel lines plays an important role, such as corridor boundary detection of an indoor mobile navigation system or power transmission lines detection of a mobile power transmission lines inspection system. Also, the proposed method has an advantage in that the detection criteria can be controlled by adjusting only three specialized parameters such as α, d_t, and φ_t. Thus, it can be easily applied to various applications.

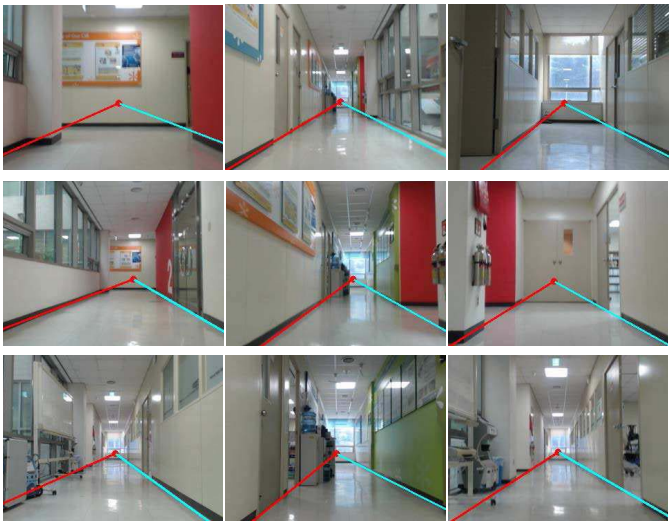

Fig. 16. Sample corridor floor detection results.

A. Indoor Mobile Navigation System

Mobile navigation systems operating inside a building should be able to determine their own locations correctly to carry out various services such as mobile object transportation [47], mobile bin picking [48], and mobile teaching [49]. In a corridor environment, however, it is difficult to estimate a location of the mobile system exactly because there may not be sufficient distinctive visual marks. Therefore, various types of structures in a corridor environment such as T-junctions, L-junctions, and ends, have been used for localization [50]. In order to detect these structures, the corridor floor can be an important cue. Thus, various methods have been proposed to detect the corridor floor, such as a laser-based method [25], an ultrasonic-based method [26] and vision-based methods [27], [28]. The vision-based methods using a single camera first estimate a vanishing point of a corridor from line segments which are extracted from boundaries of a corridor. Then, the corridor floor is detected based on the estimated vanishing point information. In this process, the proposed approach can be applied to finding the corridor floor.

Fig. 16 depicts sample results on corridor floor detection. The estimated vanishing point of a corridor is shown by a red point and the detected corridor boundaries are represented by two different colors, red and cyan. The detected corridor floor region is defined as the inside area between the detected corridor floor boundaries. The images on the first and the second rows depict sample results of detecting a corridor floor with few obstacles for each of 3 types of structures of the corridor: T-junctions, L-junctions, and ends. In these images, boundaries of the corridor floor are partly missing

because of open doors or junctions. However, the proposed method estimates the vanishing point of the corridor and detects boundaries of the corridor floor correctly. Also, as shown in the images on the third row in Fig. 16, the proposed method can efficiently estimate the vanishing point and the corridor floor boundary, although there are obstacles in the corridor.

The database for corridor floor detection consists of images acquired by a single USB camera along four different paths in a building. The total number of images is 732 and the detection criteria is as follows: the detected corridor floor region should overlap at least 95% of the ground truth region. The detection accuracy on this database is 98.3%. The parameter values used in the experiments are equal to those used in the experiments on lane detection.

B. Mobile Power Transmission Lines Inspection System

As power consumption is increased, the importance of power transmission lines(PTLs) maintenance is being emphasized for an efficient electrical energy transportation. On PTLs, there are various structures such as dampers, clamps, and insulators. These structures and PTLs should be regularly inspected and repaired to ensure reliable power supply to consumers. However, it is not only expensive but also dangerous for human inspectors to inspect them manually. Therefore, there has been growing interest on autonomous system-based approaches [51]. For an autonomous inspection system, it is important to detect the structures because they can block a path of the system when the inspection system moves along the PTLs. In order to detect the structures, PTLs detection can be utilized efficiently because the structures are around the PTLs. Thus, the PTLs detection plays an important role in PTLs inspection systems [29]–[31]. PTLs detection problem is similar to that of lane detection on a road since PTLs are also placed in parallel to each other in the real world. Therefore, the proposed approach can also be applied to the PTLs detection problem.

Fig. 17 shows our PTLs database and sample PTLs detection results on the database. The database that is used to evaluate the performance of the PTLs detection consists of three kinds of image sets acquired by the inspection system [30], and the numbers of images are 101, 70, and 184, respectively. The estimated vanishing point can be located not only within the image but also outside of the image because of a viewpoint change. For the purpose of visualization, therefore, an extra black area is padded at the bottom of each image to represent the vanishing point that might be located outside of the input image. In Fig. 17, the detected PTLs are represented by four different colors, and the estimated vanishing point is shown by a red point. The detection criteria are as follows: First, the orientation error of the detected PTLs should be within 5°. Second, the detected PTLs should overlap at least 50% of the ground truth positions. The detection accuracies for each image set are 97.02%, 94.29%, and 95.65%, respectively. The parameter values used in the experiments are $\alpha = 10$, $d_t = 15$, and $\varphi_t = 15$.

As depicted on the first row in Fig. 17, the test images contain many line segments whose orientations are similar to those of the PTLs since there are many strong line segments extracted from the ceiling and walls of the indoor demonstration room. However, the results show that the vanishing point of the PTLs can be robustly estimated and each of the PTLs can be correctly detected by using the proposed method. In outdoor scenes such as those shown on the second and the third rows in Fig. 17, one can see that the proposed approach can also detect the PTLs efficiently, even though there still are many noisy line segments that come from background structures in images.

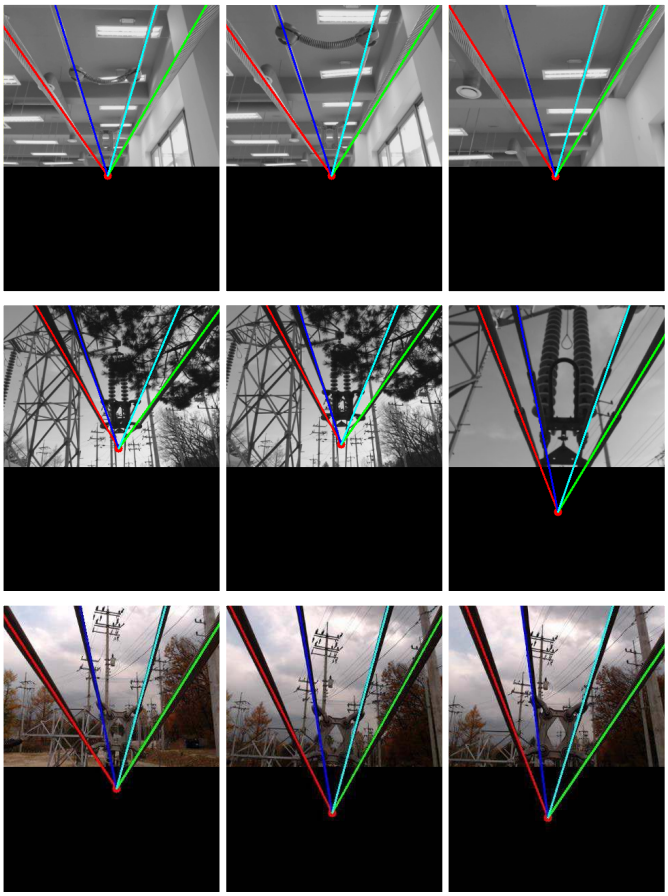

Fig. 17. Sample power transmission lines detection results.

VII. Conclusion

A robust lane detection method based on vanishing point estimation using the relevance of line segments has been described in this paper. The proposed method uses a probabilistic voting procedure based on the line segment strength to estimate the vanishing point of lanes correctly from a noisy image. In images obtained from the front-mounted camera in a vehicle, lanes are located in a fan-shaped area centered on the estimated vanishing point. Based on this observation, the proposed method detects lanes by searching only in this fan-shaped area. Then, the score function is defined to estimate lanes by using geometric relationships between the line segments and the estimated vanishing point. Finally, lanes are detected by applying flooding watershed algorithm for this score function, and the location of the detected host lane is

refined by utilizing the inter-frame similarity that considers location consistency of the detected lanes and the estimated vanishing point in consecutive frames. Furthermore, a method using a lookup table is proposed to reduce computational costs in the vanishing point estimation process. Through experimental results, it was demonstrated that the proposed method can estimate the vanishing point of lanes efficiently and also detect lanes in various environments. In addition, it was shown that the proposed approach could be successfully applied to various applications such as corridor floor detection and power transmission lines detection.

In the future, we hope to estimate lanes which are in a long distance ahead of the car correctly. Also we would like to extend the proposed approach to unstructured road detection.

REFERENCES

[1] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 4, pp. 6–22, Apr. 2014.

[2] A. Vahidi and A. Eskandarian, "Research advances in intelligent collision avoidance and adaptive cruise control," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 3, pp. 143–153, Sep. 2003.

[3] A. M. Kumar and P. Simon, "Review of lane detection and tracking algorithms in advanced driver assistance system," *Int. J. Comput. Sci. Inf. Technol.*, vol. 7, no. 4, pp. 65–78, Aug. 2015.

[4] S. Yenikaya, G. Yenikaya, and E. Düven, "Keeping the vehicle on the road: A survey on on-road lane detection systems," *Assoc. Comput. Mach. Comput. Surv.*, vol. 46, no. 1, pp. 2:1–2:43, 2013.

[5] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vis. Appl.*, vol. 25, no. 3, pp. 727–745, 2014.

[6] J. 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.

[7] V. Gaikwad and S. Lokhande, "Lane departure identification for advanced driver assistance," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 910–918, Apr. 2015.

[8] H. Yoo, U. Yang, and K. Sohn, "Gradient-enhancing conversion for illumination-robust lane detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1083–1094, Sep. 2013.

[9] S. Jung, J. Youn, and S. Sull, "Efficient lane detection based on spatiotemporal images," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 289–295, Jan. 2016.

[10] A. Borkar, M. Hayes, and M. T. Smith, "A novel lane detection system with efficient ground truth generation," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 365–374, Mar. 2012.

[11] T.-Y. Sun, S.-J. Tsai, and V. Chan, "HSI color model based lane-marking detection," in *Proc. Intell. Transp. Syst. Conf.*, Sep. 2006, pp. 1168–1172.

[12] H.-Y. Cheng, B.-S. Jeng, P.-T. Tseng, and K.-C. Fan, "Lane detection with moving vehicles in the traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 571–582, Dec. 2006.

[13] Y. He, H. Wang, and B. Zhang, "Color-based road detection in urban traffic scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 309–318, Dec. 2004.

[14] M. A. Sotelo, F. J. Rodriguez, L. Magdalena, L. M. Bergasa, and L. Boquete, "A color vision-based lane tracking system for autonomous driving on unmarked roads," *Auto. Robots*, vol. 16, no. 1, pp. 95–116, Jan. 2004.

[15] H.-J. Liu, Z.-B. Guo, J.-F. Lu, and J.-Y. Yang, "A fast method for vanishing point estimation and tracking and its application in road images," in *Proc. Int. Conf. ITS Telecommun.*, Jun. 2006, pp. 106–109.

[16] J. G. Kuk, J. H. An, H. Ki, and N. I. Cho, "Fast lane detection & tracking based on Hough transform with reduced memory requirement," in *Proc. Int. IEEE Conf. Intell. Transp. Syst.*, Sep. 2010, pp. 1344–1349.

[17] S.-N. Kang, S. Lee, J. Hur, and S.-W. Seo, "Multi-lane detection based on accurate geometric lane estimation in highway scenarios," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 221–226.

[18] M. Bertozzi and A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Trans. Image Process.*, vol. 7, no. 1, pp. 62–81, Jan. 1998.

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

[20] G. Liu, F. Wörgötter, and I. Markelić, "Combining statistical Hough transform and particle filter for robust lane detection and tracking," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2010, pp. 993–997.

[21] 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.

[22] H. Kong, J.-Y. Audibert, and J. Ponce, "General road detection from a single image," *IEEE Trans. Image Process.*, vol. 19, no. 8, pp. 2211–2220, Aug. 2010.

[23] H. Kong, S. E. Sarma, and F. Tang, "Generalizing Laplacian of Gaussian filters for vanishing-point detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 408–418, Mar. 2013.

[24] Z. Zhou, F. Farhat, and J. Z. Wang. (2016). "Detecting vanishing points in natural scenes with application in photo composition analysis." [Online]. Available: https://arxiv.org/abs/1608.04267

[25] R. Carelli and E. O. Freire, "Corridor navigation and wall-following stable control for sonar-based mobile robots," *Robot. Auto. Syst.*, vol. 45, nos. 3–4, pp. 235–247, Dec. 2003.

[26] U. Farooq, G. Abbas, S. O. Saleh, and M. U. Asad, "Corridor navigation with fuzzy logic control for sonar based mobile robots," in *Proc. IEEE Conf. Ind. Electron. Appl.*, Jul. 2012, pp. 2087–2093.

[27] W. Shi and J. Samarabandu, "Corridor line detection for vision based indoor robot navigation," in *Proc. Can. Conf. Elect. Comput. Eng.*, May 2006, pp. 1988–1991.

[28] E. Bayramoglu, N. A. Andersen, N. K. Poulsen, J. C. Andersen, and O. Ravn, "Mobile robot navigation in a corridor using visual odometry," in *Proc. Int. Conf. Adv. Robot.*, Jun. 2009, pp. 1–6.

[29] S. Fu, Z. Liang, Z. Hou, and M. Tan, "Vision based navigation for power transmission line inspection robot," in *Proc. IEEE Int. Conf. Cognit. Inform.*, Aug. 2008, pp. 411–417.

[30] J. H. Yoo, D. H. Kim, S. Lee, and S.-K. Park, "A robust power transmission lines detection method based on probabilistic estimation of vanishing point," *J. Korea Robot. Soc.*, vol. 10, no. 1, pp. 9–15, 2015.

[31] C. Hu, G. Wu, H. Cao, and X. Xiao, "Obstacle recognition and localization based on the monocular vision for double split transmission lines inspection robot," in *Proc. Int. Conf. Image Signal Process.*, Oct. 2009, pp. 1–5.

[32] G. Vouzounaras, P. Daras, and M. G. Strintzis, "Automatic generation of 3D outdoor and indoor building scenes from a single image," *Multimedia Tools Appl.*, vol. 70, no. 1, pp. 361–378, May 2014.

[33] X. Liu, Y. Zhao, and S.-C. Zhu, "Single-view 3D scene parsing by attributed grammar," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 23–28.

[34] R. Grompone von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, Apr. 2010.

[35] A. Desolneux, L. Moisan, and J.-M. Morel, *From Gestalt Theory to Image Analysis: A Probabilistic Approach*, 1st ed. New York, NY, USA: Springer, 2008.

[36] W. Liu, S. Li, and X. Huang, "Extraction of lane markings using orientation and vanishing point constraints in structured road scenes," *Int. J. Comput. Math.*, vol. 91, no. 11, pp. 2359–2373, 2014.

[37] Z. Nan, P. Wei, L. Xu, and N. Zheng, "Efficient lane boundary detection with spatial-temporal knowledge filtering," *Sensors*, vol. 16, no. 8, p. 1276, 2016.

[38] J. Hur, S.-N. Kang, and S.-W. Seo, "Multi-lane detection in urban driving environments using conditional random fields," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2013, pp. 1297–1302.

[39] C. Rasmussen, "Grouping dominant orientations for ill-structured road following," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun./Jul. 2004, pp. I-470–I-477.

[40] P. Moghadam and J. F. Dong, "Road direction detection based on vanishing-point tracking," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 1553–1560.

[41] J. Yuan, S. Tang, X. Pan, and H. Zhang, "A robust vanishing point estimation method for lane detection," in *Proc. Chin. Control Conf.*, Jul. 2014, pp. 4887–4892.

[42] J. Gu, Q. Zhang, and S.-I. Kamata, "Robust road lane detection using extremal-region enhancement," in *Proc. Asian Conf. Pattern Recognit.*, Nov. 2015, pp. 519–523.

[43] J. H. Yoo, D. H. Kim, and S.-K. Park, "A new lane detection method based on vanishing point estimation with probabilistic voting," in *Proc. IEEE Int. Conf. Consum. Electron.*, Jan. 2015, pp. 204–205.

[44] J. Goutsias and H. J. A. M. Heijmans, *Mathematical Morphology*. Amsterdam, The Netherlands: IOS Press, 2000.
[45] J. Ruyi, K. Reinhard, V. Tobi, and W. Shigang, "Lane detection and tracking using a new lane model and distance transform," *Mach. Vis. Appl.*, vol. 22, no. 4, pp. 721–737, 2011.
[46] P.-C. Wu, C.-Y. Chang, and C. H. Lin, "Lane-mark extraction for automobiles under complex conditions," *Pattern Recognit.*, vol. 47, no. 8, pp. 2756–2767, Aug. 2014.
[47] W. Chung, G. Kim, M. Kim, and C. Lee, "Integrated navigation system for indoor service robots in large-scale environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, vol. 5. Apr./May 2004, pp. 5099–5104.
[48] M. Nieuwenhuisen *et al.*, "Mobile bin picking with an anthropomorphic service robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 2327–2334.
[49] M. Kuderer, H. Kretzschmar, and W. Burgard, "Teaching mobile robots to cooperatively navigate in populated environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 3138–3143.
[50] Y.-B. Park and I. H. Suh, "Predictive visual recognition of types of structural corridor landmarks for mobile robot navigation," in *Proc. IEEE Int. Symp. Robot Human Interact. Commun.*, Sep. 2010, pp. 391–396.
[51] J. Katrasnik, F. Pernus, and B. Likar, "A survey of mobile robots for distribution power line inspection," *IEEE Trans. Power Del.*, vol. 25, no. 1, pp. 485–493, Jan. 2010.

Ju Han Yoo (S'14) received the B.S. degree in electronic engineering from Kyunghee University, South Korea, in 2005 and the M.S. degree from Sogang University, South Korea, in 2009. He is currently working toward the Ph.D. degree with Korea University, South Korea. Since 2010, he has been with Korea Institute of Science and Technology. His research interests include object recognition and image segmentation and computer graphics.

Seong-Whan Lee (F'10) received the B.S. degree in computer science and statistics from Seoul National University, Seoul, South Korea, in 1984, and the M.S. and Ph.D. degrees in computer science from Korea Advanced Institute of Science and Technology in 1986 and 1989, respectively. In 2001, he stayed with the Department of Brain and Cognitive Sciences, MIT, as a Visiting Professor. He is currently the Hyundai-Kia Motor Chair Professor with Korea University, where he is the Head of the Department of Brain and Cognitive Engineering. His research interests include pattern recognition, computer vision, and brain engineering.

Sung-Kee Park (M'16) received the B.S. and M.S. degrees in mechanical design and production engineering from Seoul National University, Seoul, South Korea, in 1987 and 1989, respectively, and the Ph.D. degree in computer vision from Korea Advanced Institute of Science and Technology, Seoul, in 2000. He has been with the Center for Robotics Research, Korea Institute of Science and Technology (KIST), Seoul. During his period at KIST, he held a visiting position with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, in 2005, where he researched object recognition. He is currently a Principal Research Scientist with KIST. His research interests include cognitive visual processing, object recognition, visual navigation, and human–robot interaction.

Dong Hwan Kim (M'03) received the B.S., M.S., and Ph.D. degrees from Seoul National University, Seoul, South Korea, in 1999, 2001, and 2006, respectively. From 2006 to 2007, he was a Senior Engineer with Samsung Electronics, Company, Ltd., Suwon, South Korea. In 2007, he joined Korea Institute of Science and Technology (KIST), Seoul, and during his period at KIST, he was with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, as a Visiting Research Associate, from 2007 to 2010. He is currently a Senior Researcher with the Center for Robotics Research, KIST. His research interests are in computer vision, pattern recognition, and image processing.