MDPI

A Robust Road Vanishing Point Detection Adapted to the Real-World Driving Scenes

Cuong Nguyen Khac [1], Yeongyu Choi [1], Ju H. Park [2] and Ho-Youl Jung [1,*]

1   Department of Information and Communication Engineering, Yeungnam University,
    Gyeongsan 38544, Korea; cuongnguyenkhac73@ynu.ac.kr (C.N.K.); ygchoi11@ynu.ac.kr (Y.C.)
2   Department of Electrical Engineering, Yeungnam University, Gyeongsan 38544, Korea; jessie@ynu.ac.kr
*   Correspondence: hoyoul@yu.ac.kr; Tel.: +82-53-810-3545

Abstract: Vanishing point (VP) provides extremely useful information related to roads in driving scenes for advanced driver assistance systems (ADAS) and autonomous vehicles. Existing VP detection methods for driving scenes still have not achieved sufficiently high accuracy and robustness to apply for real-world driving scenes. This paper proposes a robust motion-based road VP detection method to compensate for the deficiencies. For such purposes, three main processing steps often used in the existing road VP detection methods are carefully examined. Based on the analysis, stable motion detection, stationary point-based motion vector selection, and angle-based RANSAC (RANdom SAmple Consensus) voting are proposed. A ground-truth driving dataset including various objects and illuminations is used to verify the robustness and real-time capability of the proposed method. The experimental results show that the proposed method outperforms the existing motion-based and edge-based road VP detection methods for various illumination conditioned driving scenes.

Keywords: VP detection; optical flow; autonomous vehicles; ADAS; FOE; RANSAC

1. Introduction

If a set of lines that are parallel in the 3D object space is projected onto a 2D image plane of a camera under perspective view, their images will form a set of lines that intersect at a point. The intersection point is called a vanishing point (VP) [1]. Generally, the number of VPs in an image depends on the number of sets of parallel lines in the 3D object space that are projected on the image. Among a lot of VPs presented in every frame of the driving videos, the most special VP is the intersection point of lanes or road boundaries. Hereafter, the VP is called R-VP (road VP) [2] in the rest of this paper.

R-VPs provide important information regarding driving scenes for ADAS (advanced driver assistance systems) and autonomous vehicles. For example, camera rotations [3], depth distance estimations, an instantaneous moving direction of a host vehicle, drivable areas of roads, etc., can be determined by using R-VPs.

Recently, deep learning (DL) has achieved a lot of impressive successes. Particularly, CNN (convolutional neural networks) has obtained great results in image classification, image recognition, etc. Some studies have applied DL to R-VP detection. CNN is used to train R-VP detectors in [4–6]. Regression ResNet-34 is used as an R-VP detector in [7]. A combination of CNN and heatmap regression is applied to R-VP detection, where features get from a modified version of HRNet [8]. These DL-based R-VP detectors detect R-VPs for limited driving situations. Moreover, the supervised learning method in general is the data-driven learning method that needs a huge size of training datasets with the exact labels in real-world driving scenes. As can be seen in [4–8], the number of images in the training datasets is limited. It is still difficult to realize a sophisticated DL-based R-VP detection that is suitable for driving scenes in practice.

Currently, geometrical feature-based R-VP detection methods for driving scenes are still being studied and utilized. Recently, the number of geometrical feature-based methods for R-VP detection increased significantly. These recent existing methods can be classified into four groups based on types of geometrical features used in the methods, such as line segment-based [9–29], edge-based [30–33], motion-based [34–37], and texture-based methods [38–52].

Most geometrical feature-based R-VP detection methods include three main processing steps: Step (1) Line segments are extracted from various features in driving images. Step (2) The line segments that tend to converge at R-VP locations are selected. Step (3) Intersections of the selected line segments are used to vote R-VPs through a voting process.

Optical flows of image points are generated by the movement of an observer that is a camera of a host vehicle in the R-VP detection problem. The convergence point of the optical flows of stationary image points is called the focus of expansion (FOE) of the host vehicle [53]. The great advantage of the FOE is that the FOE forms from the self-movements of the host vehicle. In other words, the FOE formation of a host vehicle does not depend on geometrical feature extractions from lanes, road boundaries, and other lines. Furthermore, if a host vehicle moves in parallel with road boundaries and the roads are almost flat, its FOE will coincide with R-VP [2]. That is why FOEs of host vehicles can be used to detect efficiently R-VPs for both structured roads and unstructured roads in the existing motion-based R-VP detection methods. This paper carefully examines the main processing steps of the existing motion-based R-VP detection methods. Some disadvantages in these processing steps are analyzed and considered to design our proposed method. In this paper, a new motion-based R-VP detection method is proposed to overcome the shortcomings of the existing motion-based methods.

The novel contributions of this paper are outlined below.

1. Stable motion vector detection is proposed to detect the stable vectors from several consecutive frames.
2. Stationary point-based motion vector selection is proposed to keep useful motion vectors and to reduce unhelpful ones for the R-VP detection.
3. Angle-based RANSAC voting is proposed to reduce efficiently the contributions of the outliers.
4. The proposed R-VP detection method that consists of stable motion detection, stationary point-based motion vector selection, and angle-based RANSAC (RANdom SAmple Consensus) voting achieves high accuracy and robustness for real-world driving scenes.

The next section discusses related research work. Section 3 explains the proposed method in detail. Section 4 shows experimental results. We conclude in Section 5.

2. Related Work

2.1. Main Processing Steps of The Existing R-VP Detection Methods

Most existing R-VP detection methods consist mainly of three steps. In Step 1, the line segment-based methods extract line segments from lane markers and road boundaries. The edge-based methods and texture-based methods extract pieces that are similar to very short straight lines and convert them into line segments in a variety of ways. Motion-based methods estimate line segments from motion vectors in driving scenes using pairs of two consecutive frames. In Step 2, the estimated line segments that tend to converge at the R-VP location are selected by using various constraints. In Step 3, intersection points generated by the lines selected from step 2 are voted to find R-VP. Because the number of intersection points is usually very high, voting algorithms are always used. The most commonly used voting method is the RANSAC-based method.

The accuracy of each method above depends heavily on the type of geometrical features that the method uses. Although the line segment-based methods have achieved high accuracies, they are well-suited only for structured roads. It is known that the edge-based methods, texture-based methods, and motion-based methods can detect R-VPs for

both structured roads and unstructured roads. However, texture-based methods cannot be applied to real-time R-VP detection applications because of their high complexity. The edge-based methods are highly sensitive to illuminations due to the character of edge detectors. Eventually, the motion-based methods seem to be the most suitable method to detect R-VPs for both structured roads and unstructured roads because they do not depend on extracting features from the lanes, edges, and texture of the driving scenes. However, the existing motion-based R-VP detection methods for driving scenes still have some limitations as shown in the next sub-section. Therefore, this paper attempts to enhance the performance of the methods.

In general, various kinds of line segments are estimated from input driving video frames. The line segments that tend to converge at R-VP locations are selected by using various constraints. Because the road surface is not completely flat, the translation of the host vehicles is not completely pure, which causes a small fluctuation of the R-VPs. To compensate for the fluctuation, possible intersection points are firstly found and an intersection point with the highest vote is detected as R-VP by using voting methods such as MLE (maximal likelihood estimator) voting [9], probabilistic voting [10], line-soft-voting [12], cell-based voting [16], and RANSAC-based voting [18,21,22,24]. The next sub-sections summarize the three main processing steps of existing motion-based R-VP detection described in Figure 1.
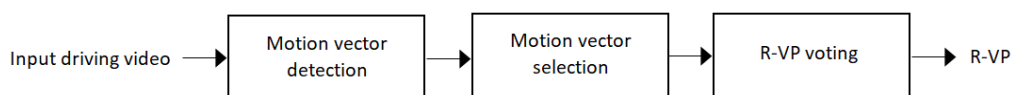

Figure 1. The main processing of existing motion-based road vanishing points (R-VP) detection methods.

2.2. Motion Vector Detection and Selection

In the motion-based R-VP detection methods [34–37], motion vectors are used as line segments. Most existing motion-based methods detect a fixed number of corner points in a frame by using corner detectors such as the Shi-Tomasi corner detector [54]. These detected corner points are tracked in the next frame using the Lucas-Kanade method [55]. The corner detection and tracking process are repeated for every pair of two consecutive frames. The corresponding points between two frames form motion vectors. As the sets of detected corner points between frames are often unstable, the estimated motion vectors lead to errors of the R-VP detection. Furthermore, the lengths of the estimated motion vectors are usually short because corner points are tracked only between two consecutive frames. These unstable and short motion vectors make it difficult to detect the FOE of host vehicles with high accuracy.

Figure 2a shows motion vectors (indicated as red lines) estimated by the existing motion-based R-VP detection methods. The white dots are the heads of the motion vectors. The remaining endpoints are tails. The direction from a tail to a head and the length of a motion vector represent movement information of a corner point. These motion vectors provide a lot of useful information for a wide variety of computer vision applications. However, in the point of view of the R-VP detection problem, these motion vectors still cause some errors for R-VP detection. In detail, most existing motion-based R-VP detection methods assume that longer motion vectors are more convergent to R-VP locations [34–37]. However, it can be seen in Figure 2a that the short vectors also coincide on the lanes and road boundaries. The short vectors also contribute well to the R-VP detection. These less distinguishable motion vectors do not provide useful information for selecting vectors that are useful for the R-VP detection problem. It is necessary to enhance motion vector detection that facilitates R-VP detection.
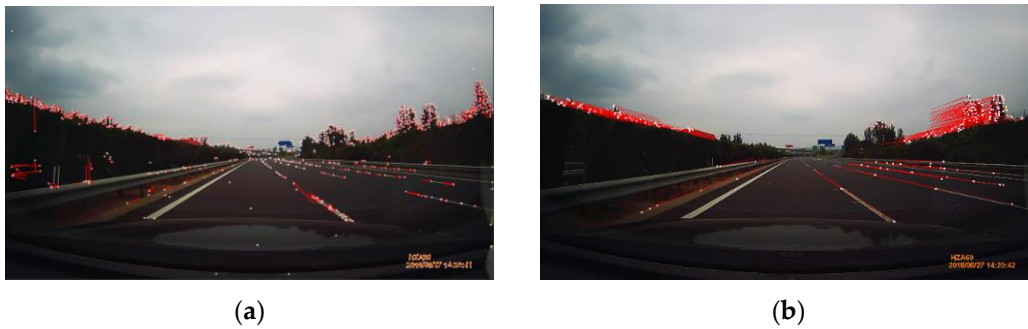
(**a**)　　　　　　　　　　　　　　　　(**b**)

Figure 2. Examples of motion vectors detected from (**a**) two consecutive frames by the existing method and (**b**) several consecutive frames by the proposed stable motion detection method.

The existing motion-based methods attempt to select useful vectors. Motion vectors that are long and not belonging to a certain area where other moving objects exist are selected [34–37]. In the methods, they eliminate the vectors in the area of on-coming and in-parallel driving vehicles. However, we observe that the motion vector selection also eliminates useful vectors for R-VP detection. It is helpful to keep the motion vectors coming from stationary objects.

2.3. Voting Method

In motion-based methods, motion vectors are used as line segments. Because the number of intersections of the line segments is often very high and these lines converge at a small location instead of a single point, a voting process is needed. The RANSAC-based voting has been used commonly for low computational complexity. A minimum number of line segments are randomly selected to generate an estimated R-VP as a hypothesis R-VP. The hypothesis R-VP is voted by the remaining motion vectors. In the existing methods, perpendicular distances between the hypothesis R-VP and every line segment are calculated and used as a metric to vote the hypothesis R-VP. The voting process is applied iteratively for other hypotheses. A hypothesis R-VP that gets the highest vote is detected as R-VP. The distance-based metric does not guarantee to assign the higher scores to the useful vectors (inliers) that tend to highly converge at the R-VP locations. As shown in Section 3.3, the two motion vectors (indicated by the red arrows) have the same perpendicular distance from the hypothesis R-VP (indicated by the blue dot), even though they converge in completely different directions. In this paper, an efficient RANSAC-based voting method is proposed to exclude most of the outliers from the voting process.

3. Proposed Method

The proposed method also has the same main processing steps as shown in Figure 1. However, the three processing steps are proposed to improve the robustness and real-time ability of the R-VP detection. The flowchart of the proposed R-VP detection method is shown in Figure 3. The following sub-sections describe three processing steps in detail.

3.1. Stable Motion Vector Detection

In contrast to most existing motion-based R-VP detection methods that obtain motion vectors from two consecutive frames, our motion vector detection estimates stable motion vectors over several consecutive frames. From a given first frame f_{t_0} at a time t_0, corner points are detected by using the Shi-Tomasi corner detector [54]. All corner points form an initial set, denoted as $P_{t_0} = \{p_{t_0}(1), p_{t_0}(2), \ldots, p_{t_0}(n_{t_0})\}$, where n_{t_0} is the number of corner points. Every point in the initial set is tracked over the successive frames f_{t_0+k} ($k = 1, 2, 3, \ldots$) by using the Kanade-Lucas optical flow method [55]. Any point in the initial set that is not tracked in the current frame f_{t_0+k} is deleted from the set P_{t_0}. Otherwise, the successive tracked points are updated into the tracked set P'_{t_0+k}. Even though the set is denoted as P'_{t_0+k}, only one set is used and updated for a different time $t_0 + k$. The initial

point $p_{t_0}(i) \in P_{t_0}$ and its corresponding tracked point $p'_{t_0+k}(i) \in P'_{t_0+k}$ have the same index i and indicate the pixel-wise coordinates.
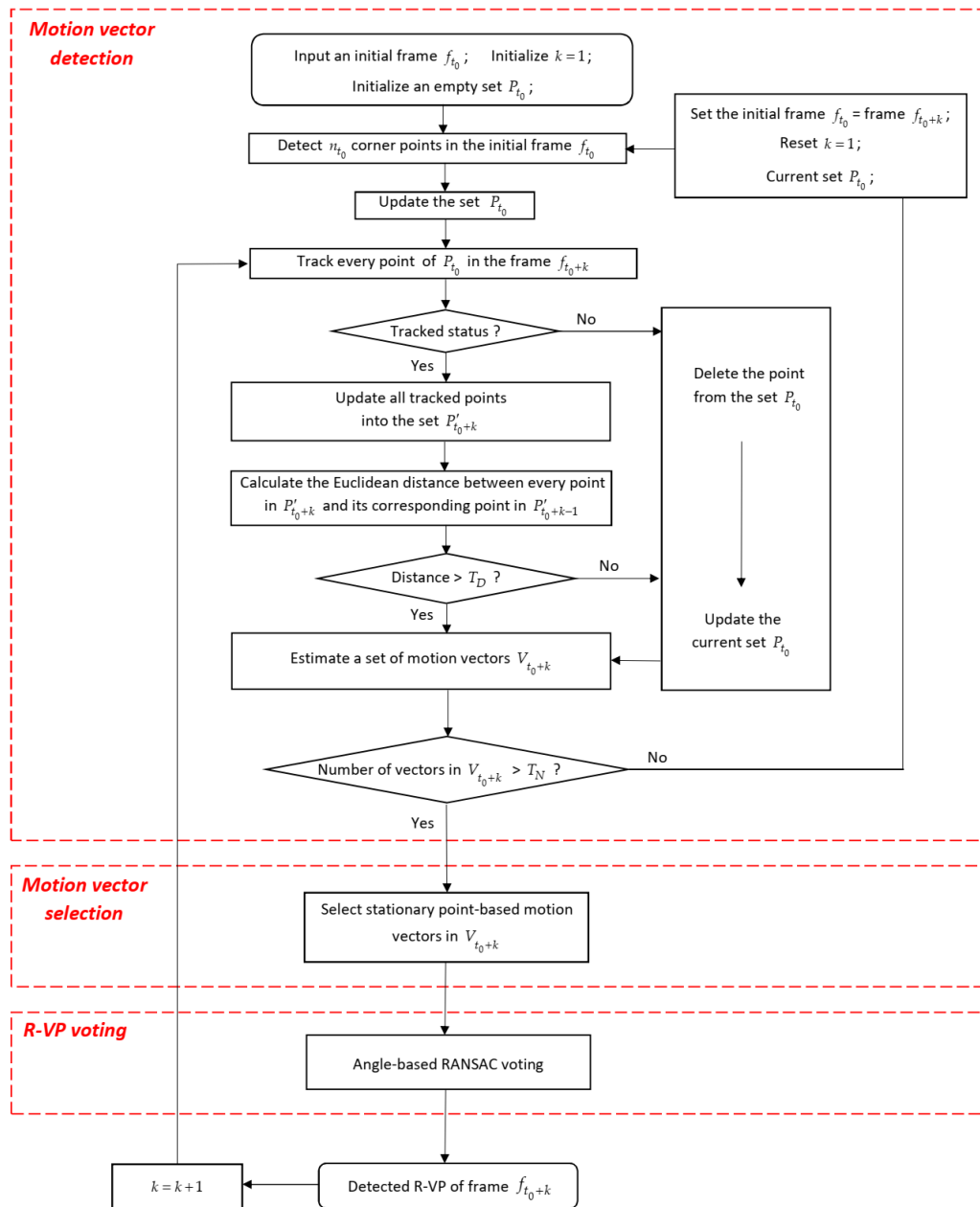

Figure 3. Flowchart of the proposed R-VP detection method.

Detail procedures to detect stable motion vectors are as follows. If the corner points are not tracked in the current frame f_{t_0+k} or have a very short displacement (smaller than a predefined distance T_D) between the previous frame f_{t_0+k-1} and the current frame f_{t_0+k}, the points in the initial set are deleted. The number of remaining corner points at the frame f_{t_0+k} decreases to $n_{t_0+k}(\leq n_{t_0+k-1})$, then the initial set and tracked set are now updated as $P_{t_0} = \{p_{t_0}(1), p_{t_0}(2), \ldots, p_{t_0}(n_{t_0+k})\}$ and $P'_{t_0+k} = \{p'_{t_0+k}(1), p'_{t_0+k}(2), \ldots, p'_{t_0+k}(n_{t_0+k})\}$, respectively. The lifetime of tracked points depends on how fast or slow the movement of the host vehicle is. The corner point tracking process performs iteratively until the number of the remaining points n_{t_0+k} is less than a pre-defined threshold number T_N. Whenever the number of remaining points is less than T_N, new corner points are detected in the actual

frame and appended to the initial set. The new detected corner points and remaining old ones keep a sufficient number of points to estimate stable motion vectors.

The motion vector set $V_{t_0+k} = \{v_{t_0+k}(1), v_{t_0+k}(2), \ldots, v_{t_0+k}(n_{t_0+k})\}$ is obtained from every remaining point in the tracked set and its corresponding point in the initial set. The length of each motion vector $\|v_{t_0+k}(i)\|$ is the distance between a point $p_{t_0}(i)$ and its corresponding point $p'_{t_0+k}(i)$ at the current frame f_{t_0+k}. Figure 2b shows that the estimated motion vectors of the proposed method are much longer than the ones taken from two consecutive frames. Compared to Figure 2a, most redundant and short motion vectors are eliminated in Figure 2b. As long motion vectors appear over many successive frames, they are considered to be stable motion vectors. Besides, many long motion vectors coincide with lanes and road boundaries. The stable motion vectors are very helpful to improve the performance of R-VP detection.

3.2. Stationary Point-Based Motion Vector Selection

In general, there are three different types of motion vectors in the driving scenes when a camera of a host vehicle moves. The first is motion vectors that are generated from the approaching of stationary objects such as lanes, road boundaries, trees, traffic signs, buildings, etc. Two other motion vectors occur from other moving objects. The second type of vector is generated from other accelerating, overtaking, and land changing vehicles. The third one occurs from on-coming or decelerating vehicles. Figure 4a shows an example of the three types of motion vectors. In this Figure, the numbers 1, 2, and 3 indicate motion vectors from the stationary background, an accelerating vehicle, and a decelerating vehicle, respectively. Type 2 and type 3 motion vectors are numerous and might generate many incorrect R-VPs. Moreover, if the number of moving objects increases, their appearances will affect strongly the accuracy of R-VP detection. To achieve high performance of the R-VP detection, it is important to reduce the motion vectors type 2 and type 3 as much as possible, maintaining type 1.
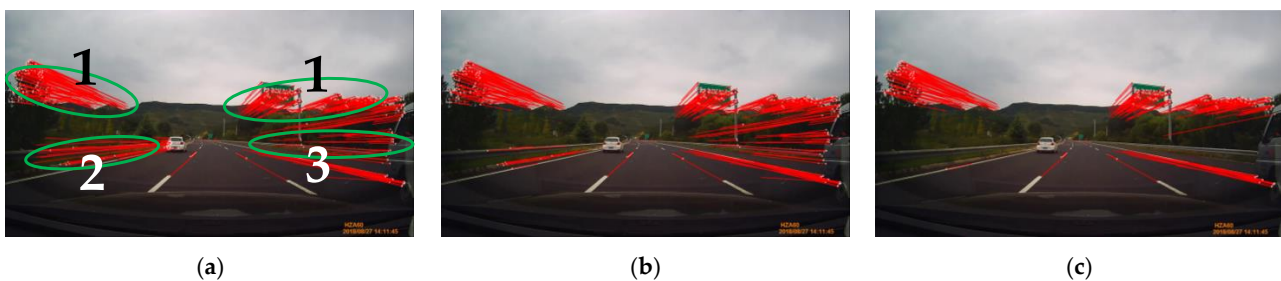

(a)　　　　　　　　　　(b)　　　　　　　　　　(c)

Figure 4. (**a**) Three types of motion vectors in a driving scene, where motion vectors types 1, 2, and 3 come from the stationary background, accelerating, and decelerating vehicles, respectively. The motion vectors (**b**) when applying motion vector type 2 reduction sub-step and (**c**) when applying both motion vector type 2 and 3 reduction sub-steps.

In the following sub-sections, the proposed stationary point-based motion vector selection method is introduced to reduce two different types of motion vectors respectively, based on the experimental observations.

3.2.1. Reduction of Motion Vector Type 2

It is observed that a current tracked point $p'_{t_0+k}(i)$ belonging to motion vector type 2 is closer to the image center when k increases. Then, the motion vector set V_{t_0+k} at the time $t_0 + k$ is updated to maintain motion vectors of which the head points $p'_{t_0+k}(i)$ get farther away from the image center p_c as follows;

$$V_{t_0+k} = \left\{ v_{t_0+k}(i) \mid \|p'_{t_0+k}(i) - p_c\| > \|(p_{t_0}(i) - p_c)\|, v_{t_0+k}(i) \in V_{t_0+k} \right\} \tag{1}$$

The above constraint does not guarantee to eliminate all motion vector type 2. It is observed that the motion vector with a completely different direction has equal or very

similar $\|p_{t_0}(i) - p_c\|$ and $\|p'_{t_0+k}(i) - p_c\|$. For this case, an extended actual head point denoted as $p'_{ext\_t_0+k}(i)$ is calculated using Equation (2) and applied to Equation (1) to eliminate efficiently the motion vector type 2.

$$p'_{ext\_t_0+k}(i) = p'_{t_0+k}(i) + l\frac{v_{t_0+k}(i)}{\|v_{t_0+k}(i)\|} \tag{2}$$

where $p'_{t_0+k}(i)$ is extended l pixels along the direction of the corresponding motion vector $v_{t_0+k}(i)$. Figure 4b shows an example of motion vector type 2 reduction where the left side white vehicle-related motion vectors are almost eliminated.

3.2.2. Reduction of Motion Vector Type 3

The motion vector type 3 comes from on-coming or decelerating vehicles. As both motion vector types 1 and 3 have similar directions and lengths, it is difficult to distinguish type 3 from type 1. For such reasons, the region where the type 3 motion occurs frequently is excluded in [34]. The angle between every motion vector and image horizon is used to exclude the region. Motion vectors having an angle in the interval $[0^0, 30^0]$ are excluded in [34]. Our motion vector type 3 reduction uses the angle criterion, but a smaller angle interval is applied to prevent from excluding type 1. Motion vectors with angles in the interval $[0^0, 10^0]$ are excluded from the current vector set. A vector length-based reduction is proposed to reduce the remaining type 3, based on the experimental observation that type 3 tends to be shorter than type 1. A certain percentage of the longest motion vectors are kept in the current vector set V_{t_0+k} and others are excluded. In summary, the vectors that have small angles with image horizon or short lengths are eliminated from the current vector set. Figure 4c shows that the right-side truck-related motion vectors are reduced.

3.3. Angle-Based RANSAC Voting Method

In this section, we propose an angle-based RANSAC voting method to overcome the disadvantage of the existing perpendicular distance-based metrics as mentioned in Section 2.3 (Figure 5a). As shown in Figure 5b, the intersection point (indicated by the blue dot) calculated from randomly selected two vectors (indicated by black lines) in the set V_{t_0+k} is hypothesis R-VP p_H. The angle $\theta(i)$ between two vectors $v_{t_0+k}(i)$ and $u_{t_0+k}(i)$ is calculated as given in Equation (3) and used for voting. $u_{t_0+k}(i)$ is a vector pointing in the direction from point $p'_{t_0+k}(i)$ to the hypothesis p_H.

$$\theta(i) = \arccos\left(\frac{u_{t_0+k}(i) \cdot v_{t_0+k}(i)}{\|u_{t_0+k}(i)\| \, \|v_{t_0+k}(i)\|}\right) \tag{3}$$

As the angle $\theta(i)$ is smaller when the vector $v_{t_0+k}(i)$ orients closer to the hypothesis, it can be used as a good metric for voting. As shown in Figure 5b, given two vectors $v_{t_0+k}(i)$ and $v_{t_0+k}(j)$ (indicated by the red arrows) have different angles $\theta(i)$ and $\theta(j)$, unlike the perpendicular distance metric described in Section 2.3 and Figure 5a. Even when two motion vectors move further along their respective directions, the vector $v_{t_0+k}(i)$ orienting closer to the hypothesis has still a smaller angle than another $v_{t_0+k}(j)$. This is the reason why the angle-based metric is helpful to achieve higher performance of the R-VP detection compared to the perpendicular distance-based one. The angle metric is used in the proposed angle-based RANSAC voting method.

A score function is designed by using the exponential function so that a vector with a smaller angle contributes more to the voting. The score $S(i)$ is calculated on every motion vector $v_{t_0+k}(i) \in V_{t_0+k}$ for the given hypothesis. A motion vector is regarded as an inlier if it has an angle $\theta(i)$ that is smaller than the predefined T_θ.

$$S(i) = \begin{cases} e^{-|\theta(i)|} & \theta(i) < T_\theta \\ 0 & otherwise \end{cases} \tag{4}$$
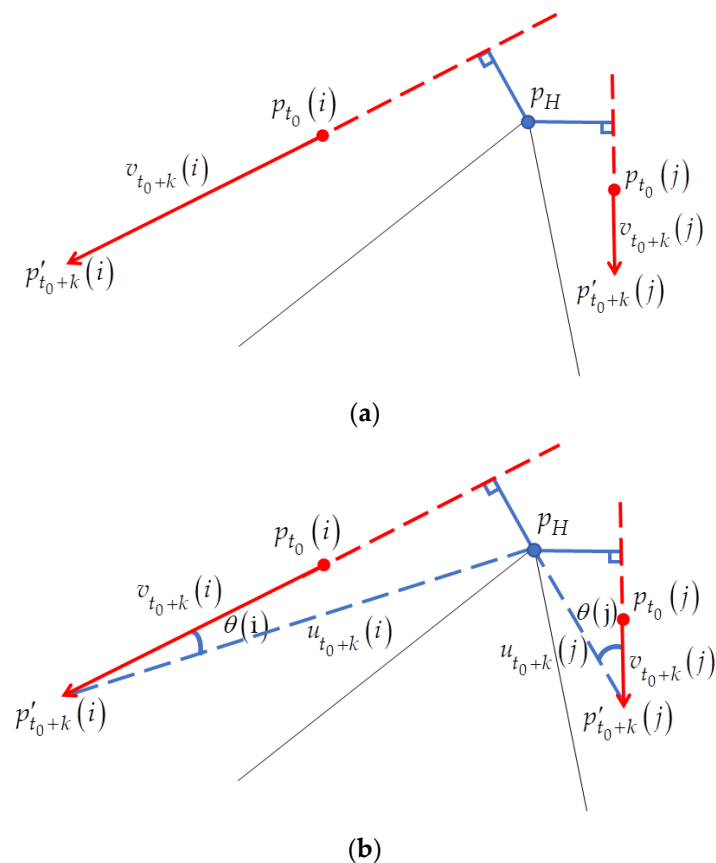
(a)

(b)

Figure 5. Examples of (**a**) perpendicular distance metric and (**b**) the proposed angle-based metric between vectors and hypothesis R-VP, assuming that hypothesis R-VP p_H is obtained from randomly selected two vectors.

4. Experiments

For the simulations, the proposed R-VP detection is tested on Jiqing Expressway dataset [56] that includes various driving conditions such as straight roads, slightly curved roads, shadows, illuminations, occlusions, moving objects, etc. The dataset is composed of 32 video clips with FHD (1920 × 1080) dimensions. Each video clip has 5393 frames, then the total number of frames is 172,576 frames (=32 clips × 5393 frames). The coordinates of points on the lanes are fully provided in the dataset. Intersection points of the lanes are calculated and used as ground-truth R-VPs.

Two existing R-VP detection methods are also tested for comparison with the proposed method. The motion-based method [34] is implemented as faithfully as possible, which consists mainly of two successive frames-based motion vector detection, region-based motion vector selection, and the perpendicular distance metric-based RANSAC voting. Maximum 500 motion vectors are detected from every two successive frames and used for the voting. The edge-based R-VP method [30] using Canny edge detector and probabilistic Hough Transform is also implemented as faithfully as possible. For the proposed method, 500 corner points are detected in the initial frame ($n_{t_0} = 500$) and tracked. Whenever the number of the tracked points falls under $T_N = 400$, new 500 corner points will be detected and appended to the initial set. In the stable motion vector detection step, a tracked point is kept when its displacement between two successive frames is bigger than $T_D = 2$. The parameter $T_\theta = 45^0$ is used to distinguish inliers and outliers.

The performances are evaluated by using the normalized Euclidean distance between the detected R-VP's coordinate (x_E, y_E) and the ground truth one (x_G, y_G). The normalized error *dist* is given by [34].

$$dist = \frac{\sqrt{(x_E - x_G)^2 + (y_E - y_G)^2}}{D} \tag{5}$$

Here D is the length of the diagonal of the image. The error *dist* ranges over $[0, 1]$. When *dist* is closer to zero, the position of the detected R-VP is closer to that of ground-truth R-VP.

Table 1 shows the accuracy of the proposed method and the other two methods, where the average of the normalized error with its standard deviation is shown. The processing time in milliseconds is also presented for the comparison of computational complexity. To evaluate the validity of the stationary point-based motion vector selection step, the proposed method is tested in four different cases; One is the case of not using motion vector type 2 reduction and type 3 reduction. Two others are the cases of applying either type 2 or type 3 reduction. The last one is applying both reductions.

Table 1. Performance comparisons of the proposed, the motion-based [33] and the edge-based [29] methods in terms of the average distance between the detected and ground truth R-VPs, its standard deviation, and processing time.

Methods		RANSAC Iterations	Average of Error	Standard Deviationof Error	Average of Time Processing (ms)
Edge-based method		900	0.0191347	0.2454778	611.20
Motion-based method		900	0.0148898	0.2088319	147.92
	Reduction of type 2	Reduction of type 3			
The proposed method	✖	✖	900	0.0075473	0.0101119	446.42
	✔	✖	900	0.0049879	0.0080097	410.22
	✖	✔	900	0.0051795	0.0091563	339.21
	✔	✔	900	0.0038549	0.0073061	287.07
	✔	✔	45	0.0042509	0.0073760	62.57

As shown in Table 1, the proposed motion-based R-VP detection, which applies both motion vector type 2 reduction and type 3 reduction, has the highest accuracy (the smallest average error) and the smallest standard deviation. The proposed method has outstanding processing time. The proposed method achieves the average of error 0.0038549 (8.48 pixels in distance) and the standard deviation of 0.0073061 (16.08 pixels in distance). The achieved values are 3.86 and 28.58 times smaller than those of the existing motion-based method, respectively. In particular, the proposed method with a smaller RANSAC iteration number of 45 is the fastest (62.57 ms), maintaining high accuracy similar to the case of iteration number 900. Even in the case of not using the motion vector selection step, the proposed method shows higher accuracy than the two existing methods in terms of both average error and standard deviation. This means that the proposed angle-based RANSAC voting works very well. Simulation results show that the proposed stationary point-based motion vector selection is also efficient to improve the R-VP detection performance. In particular, motion vector type 2 reduction contributes to the improvements slightly more than type 3 reduction. Figure 6 shows the distribution of the normalized errors for the proposed and two existing methods. The normalized errors of the proposed method distribute closer to zero, compared to the others. The performances of the proposed method are also evaluated under three different illumination environments as in Table 2. The video dataset is divided into bright days (17 video clips), slightly dark days (5 video clips), and in-tunnel (10 video clips). The performances are evaluated separately. As shown in the table, the average

error and the standard deviation are very similar in three different illuminations. This demonstrates the robustness of the proposed method.
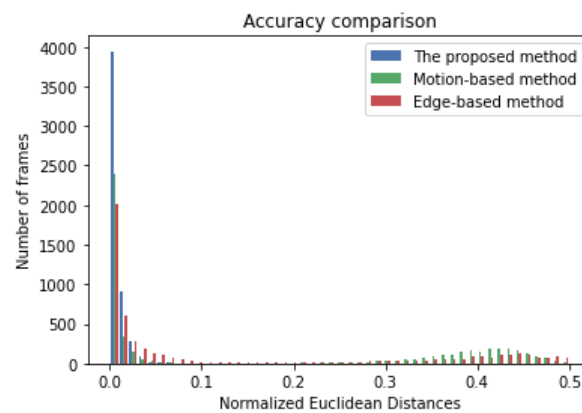

Figure 6. Distribution of normalized error of the proposed (blue), existing motion-based (green), and edge-based (red) R-VP detection methods.

Table 2. The performances of the proposed method with reduction of type 2 and type 3 under three different illumination conditions of driving scenes such as bright days, slightly dark days, and in-tunnels.

	Bright Days		Slightly Dark Days		In Tunnels	
	Average of Error	Standard Deviation	Average of Error	Standard Deviation	Average of Error	Standard Deviation
The proposed method with RANSAC iteration 900	0.0041823	0.0072681	0.0041252	0.0073249	0.0032571	0.0073250
The proposed method with RANSAC iteration 45	0.0045382	0.0073146	0.0046241	0.0074313	0.0035902	0.0073825

To evaluate the performances according to the feature extractions, the proposed R-VP detections using SIFT (scale-invariant feature transform) and SURF (speeded-up robust features) features instead of the Shi-Tomasi feature are tested and compared. Table 3 shows that both SIFT and SURF have higher accuracies compared to the two existing methods, but the accuracies are lower than those of using the Shi-Tomasi feature. As mentioned in [54], the Shi-Tomasi feature was mainly designed for tracking, whereas SIFT [57] was mainly designed for image matching and SURF [58] is just a speeded-up version of SIFT. Additionally, Table 3 also shows that SIFT and SURF require very high computational costs. Therefore, SIFT and SURF are not efficient features for motion vector estimation in the R-VP detection problem.

Table 3. The performances of the proposed method with reduction of type 2 and type 3 using scale-invariant feature transform (SIFT) and speeded-up robust features (SURF) features.

Methods	RANSAC Iterations	Average of Error	Standard Deviationof Error	Average of Time Processing (ms)
The proposed method using SIFT	900	0.0091880	0.0100742	1500.37
	45	0.0106574	0.0113037	1335.65
The proposed method using SURF	900	0.0132818	0.0159866	728.19
	45	0.0146137	0.0162612	632.57

We also test the proposed method using the perpendicular distance metric-based RANSAC [59] instead of the proposed angle-based RANSAC. Results in Table 4 show

that RANSAC using the angle-based metric is better than that of using the perpendicular distance metric.

Table 4. The performances of the proposed method with reduction of type 2 and type 3 using perpendicular distance metric-based RANSAC.

Methods	RANSAC Iterations	Average of Error	Standard Deviation of Error	Average of Time Processing (ms)
The proposed method using distance-based RANSAC	900	0.0093751	0.0136126	283.61
	45	0.0112964	0.0138230	61.82

Some examples of the proposed motion-based R-VP detection are shown in Figure 7, where ground-truth and detected R-VPs are indicated as green and red dots, respectively. The motion vectors are also represented by red lines. Examples show that the proposed method works well in various illumination conditions such as bright and slightly dark daytime, tunnel, and nighttime. An extra test is performed for a nighttime scene as the nighttime video is not included in the ground-truth dataset. Note that ground-truth R-VP is not represented in Figure 7d. We observe that the detected R-VP overlaps visually the intersection point of the road boundaries. The red and green dots in Figure 7a–c almost coincide. Two examples of nighttime R-VP detection are shown in Figure 8, where video frames extracted from "ZJU Day and Night Driving Dataset" [60,61] are used without ground-truth R-VPs. White lines are road boundaries drawing by hand and the red circle is detected R-VP. The red circle and the intersection of road boundaries are also almost coincided. Such coincidences demonstrate the accuracy of the proposed method.

(**a**)

(**b**)

(**c**)

(**d**)

Figure 7. Some examples of the proposed motion-based R-VP detection in various illumination conditions such as (**a**) bright and (**b**) slightly dark daytime, (**c**) tunnel, and (**d**) nighttime. Note that ground-truth R-VP is not indicated in (**d**).

(**a**)   (**b**)
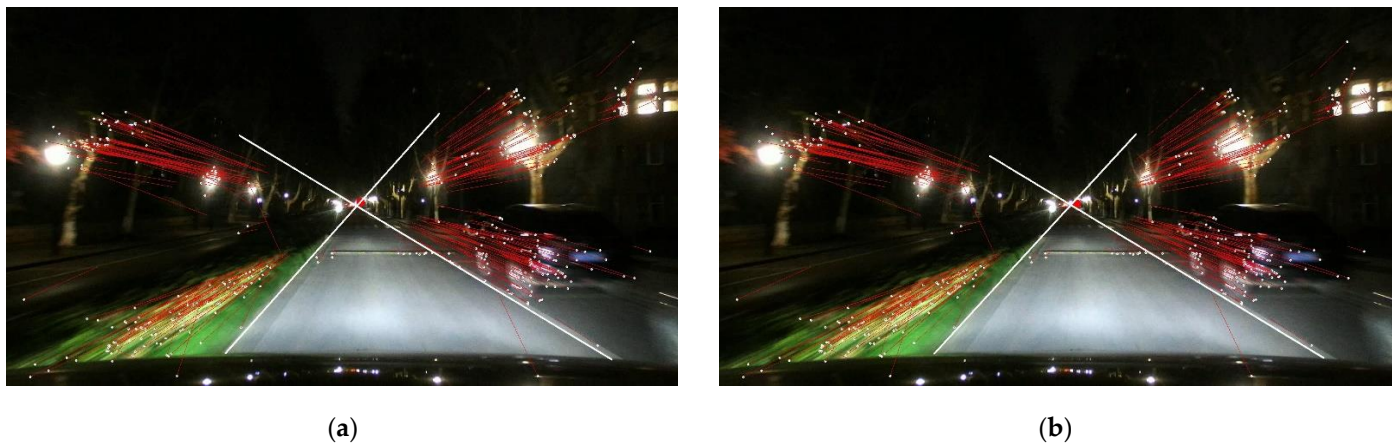
Figure 8. Detected R-VPs (indicated by red circles) and road boundaries drawing by hand (by white lines) under nighttime conditions using the proposed method applying reduction of type 2 and type 3, using (**a**) RANSAC iteration 900 and (**b**) iteration 45.

In general, we observed that the proposed method can detect R-VP efficiently when host vehicles move in parallel with road boundaries and road surfaces are almost flat as mentioned in Section 1. The proposed method sometimes fails to detect correctly R-VP in cases of uneven (or bumpy) and curved roads.

5. Conclusions

In this paper, we proposed a motion-based R-VP detection method. Three main processing steps often used in the existing R-VP detection methods are carefully examined. Based on the analysis, we propose stable motion detection, stationary point-based motion vector selection, and angle-based RANSAC voting, which are successfully applied to improve the performances of the main three steps.

Through the simulations, we show that the proposed stable motion detection and angle-based RANSAC voting contribute considerably to the improvement of R-VP detection performance. In particular, the proposed angle-based RANSAC voting works much better than conventional perpendicular distance-based voting. The proposed motion-based R-VP detection that combines all three proposed three-step algorithms outperforms the existing motion-based and edge-based R-VP detection methods. The simulation results show that the proposed method is applicable for nighttime, as well as various illumination conditioned driving scenes. It is also shown that the standard deviation of the detected R-VP position is the smallest, compared to others. The performances are very similar in three different illumination conditions. This means the proposed method is robust as well as accurate compared to the existing R-VP detection methods. From the viewpoint of computational complexity, the proposed can be realized in real-time. It is possible to apply selectively the proposed three-step algorithms depending on the system resources. Even though the proposed method is not evaluated on the unstructured road driving video because of the lack of the ground truth datasets, it is expected that it works in a stable on the unstructured road driving.

The proposed R-VP detection method plays an important role to improve the performances of various applications such as online camera calibration, detection of drivable areas, and driving direction of the host vehicle.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hearn, D.; Baker, M.P. Chapter 12: Three-Dimensional Viewing. In *Computer Graphics C Version*, 2nd ed.; Pearson: Noida, India, 1997; pp. 446–447.
2. Guo, X.; Li, Q.; Sun, C. A new road tracking method based on heading direction detection. *J. Automob. Eng.* **2018**, *233*, 232–248. [CrossRef]
3. Orghidan, R.; Salvi, J.; Gordan, M.; Orza, B. Camera calibration using two or three vanishing points. In Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS), Wrocław, Poland, 9–12 September 2012; pp. 123–130.
4. Chang, C.; Zhao, J.; Itti, L. DeepVP: Deep Learning for Vanishing Point Detection on 1 Million Street View Images. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 4496–4503.
5. Xingxin, L.; Liqiang, Z.; Zujun, Y.; Yanqin, W. Adaptive auxiliary input extraction based on vanishing point detection for distant object detection in high-resolution railway scene. In Proceedings of the 14th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), Changsha, China, 1–3 November 2019; pp. 522–527.
6. Zeng, Z.; Wu, M.; Zeng, W.; Fu, C.W. Deep Recognition of Vanishing-Point-Constrained Building Planes in Urban Street Views. *IEEE Trans. Image Process.* **2020**, *29*, 5912–5923. [CrossRef]
7. Han-Soo, C.; Keunhoi, A.; Myungjoo, K. Regression with residual neural network for vanishing point detection. *Image Vis. Comput. J. Books* **2019**, *91*, 103797.
8. Liu, Y.B.; Zeng, M.; Meng, Q.H. Unstructured Road Vanishing Point Detection Using Convolutional Neural Networks and Heatmap Regression. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 1–8. [CrossRef]
9. Wang, Y.; Lu, X.; Ling, Z.; Yang, Y.; Zhang, Z.; Wang, K. A Method to Calibrate Vehicle-Mounted Cameras under Urban Traffic Scenes. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 3270–3279. [CrossRef]
10. Yoo, J.H.; Lee, S.; Park, S.; Kim, D.H. A Robust Lane Detection Method Based on Vanishing Point Estimation Using the Relevance of Line Segments. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 3254–3266. [CrossRef]
11. Ding, W.; Li, Y. Efficient vanishing point detection method in complex urban road environments. *IET Comput. Vis.* **2015**, *9*, 549–558. [CrossRef]
12. Wang, E.; Sun, A.; Li, Y.; Hou, X.; Zhu, Y. Fast vanishing point detection method based on road border region estimation. *IET Image Process.* **2018**, *12*, 361–373. [CrossRef]
13. Zhang, Y.; Su, Y.; Yang, J.; Ponce, J.; Kong, H. When Dijkstra Meets Vanishing Point: A Stereo Vision Approach for Road Detection. *IEEE Trans. Image Process.* **2018**, *27*, 2176–2188. [CrossRef]
14. Ding, W.; Li, Y.; Liu, H. Efficient vanishing point detection method in unstructured road environments based on dark channel prior. *IET Comput. Vis.* **2016**, *10*, 852–860. [CrossRef]
15. Hwang, H.J.; Yoon, G.; Yoon, S.M. Optimized Clustering Scheme-Based Robust Vanishing Point Detection. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 199–208. [CrossRef]
16. VanQuang, N.; Heungsuk, K.; SeoChang, J.; Kwangsuck, B. A Study on Real-Time Detection Method of Lane and Vehicle for Lane Change Assistant System Using Vision System on Highway. *Eng. Sci. Technol. Int. J.* **2018**, *21*, 822–833.
17. Kim, J. Efficient Vanishing Point Detection for Driving Assistance Based on Visual Saliency Map and Image Segmentation from a Vehicle Black-Box Camera. *Symmetry* **2019**, *11*, 1492. [CrossRef]
18. Pei, L.; Liu, K.; Zou, D.; Li, T.; Wu, Q.; Zhu, Y.; Li, Y.; He, Z.; Chen, Y.; Sartori, D. IVPR: An Instant Visual Place Recognition Approach Based on Structural Lines in Manhattan World. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 4173–4187. [CrossRef]
19. Li, B.; Guo, Y.; Zhou, J.; Cai, Y.; Xiao, J.; Zeng, W. Lane Detection and Road Surface Reconstruction Based on Multiple Vanishing Point & Symposia. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Changshu, Suzhou, 30 June–1 July 2018; pp. 209–214.
20. Suttorp, T.; Bucher, T. Robust vanishing point estimation for driver assistance. In Proceedings of the IEEE Intelligent Transportation Systems Conference, Toronto, ON, Canada, 17–20 September 2006; pp. 1550–1555.

21. Young-Woo, S.; Ragnunathan, R. Detection and tracking of the vanishing point on a horizon for automotive applications. In Proceedings of the 6th IEEE IROS Workshop on Planning, Perception and Navigation for Intelligent Vehicles (PPNIV-14), Chicago, IL, USA, 14–18 September 2014.
22. Kroeger, T.; Dai, D.; Timofte, R.; Van Gool, L. Discovery of Sets of Mutually Orthogonal Vanishing Points in Videos. In Proceedings of the 2015 IEEE Winter Applications and Computer Vision Workshops, Waikoloa, HI, USA, 6–9 January 2015; pp. 63–70.
23. Ju-Han, Y.; Dong-Hwan, K.; Sung-Kee, P. A new lane detection method based on vanishing point estimation with probabilistic voting. In Proceedings of the IEEE International Conference on Consumer Electronics (ICCE), Berlin, Germany, 6–9 September 2015; pp. 204–205.
24. Young-woo, S.; Ragunathan, R. Use of a monocular camera to analyze a ground vehicle's lateral movements for reliable autonomous city driving. In Proceedings of the IEEE IROS Workshop on Planning, Perception and Navigation for Intelligent Vehicles, Tokyo, Japan, 3–7 November 2013.
25. Tian, Y.; Cui, W.; Li, X.; Chen, L. A Robust Lane Detection Method Based on Vanishing Point Estimation. *Procedia Comput. Sci.* **2018**, *131*, 354–360.
26. Piao, J.; Shin, H. Robust hypothesis generation method using binary blob analysis for multi-lane detection. *IET Image Process.* **2017**, *11*, 1210–1218. [CrossRef]
27. Yong, L.; Weili, D.; XuGuang, Z.; Zhaojie, J. Road detection algorithm for Autonomous Navigation Systems based on dark channel prior and vanishing point in complex road scenes. *Robot. Auton. Syst.* **2016**, *85*, 1–11.
28. Jiaming, H.; Zhong, Y.; Guoxiong, H.; Ting, F.; Hao, X. Robust and efficient vanishing point detection in unstructured road scenes for assistive navigation. *Sens. Rev.* **2018**, *39*, 137–146.
29. Bazin, J.; Pollefeys, M. 3-line RANSAC for orthogonal vanishing point detection. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 4282–4287.
30. Yoon-Young, M.; Zong-Woo, G.; Gi-Tae, H. Vanishing point detection for self-driving car using harmony search algorithm. *Swarm Evol. Comput.* **2018**, *41*, 111–119.
31. Su, Y.; Zhang, Y.; Alvarez, J.M.; Kong, H. An illumination-invariant nonparametric model for urban road detection using monocular camera and single-line lidar. In Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO), Macau, China, 5–8 December 2017; pp. 68–73.
32. Mistry, V.H.; Makwana, R.M. Computationally efficient vanishing point detection algorithm based road segmentation in road images. In Proceedings of the IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT), New York, NY, USA, 2–3 December 2016; pp. 103–110.
33. Wang, Z.; Qu, W.; Kamata, S. Vehicle detection from onboard camera using patch decided vanishing point. In Proceedings of the 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Shanghai, China, 14–16 October 2017; pp. 1–7.
34. Yu, Z.; Zhu, L. Roust Vanishing Point Detection Based on the Combination of Edge and Optical Flow. In Proceedings of the 4th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS), Nagoya, Japan, 13–15 July 2019; pp. 184–188.
35. Yu, Z.; Zhu, L.; Lu, G. An Improved Phase Correlation Method for Stop Detection of Autonomous Driving. *IEEE Access* **2020**, *8*, 77972–77986. [CrossRef]
36. Jang, J.; Jo, Y.; Shin, M.; Paik, J. Camera Orientation Estimation Using Motion-Based Vanishing Point Detection for Advanced Driver-Assistance Systems. *IEEE Trans. Intell. Transp. Syst.* **2020**, 1–11. [CrossRef]
37. Jo, Y.; Jang, J.; Paik, J. Camera orientation estimation using motion based vanishing point detection for automatic driving assistance system. In Proceedings of the IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 12–14 January 2018; pp. 1–2.
38. Yang, W.; Fang, B.; Tang, Y.Y. Fast and Accurate Vanishing Point Detection and Its Application in Inverse Perspective Mapping of Structured Road. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *48*, 755–766. [CrossRef]
39. Yang, G.; Wang, Y.; Yang, J.; Lu, Z. Fast and Robust Vanishing Point Detection Using Contourlet Texture Detector for Unstructured Road. *IEEE Access* **2019**, *7*, 139358–139367. [CrossRef]
40. Moghadam, P.; Dong, J.F. Road direction detection based on vanishing-point tracking. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Algarve, Portugal, 7–12 October 2012; pp. 1553–1560.
41. Hui, K.; Jean-Yves, A.; Jean, P. Vanishing point detection for road detection. In Proceedings of the CVPR 2009—IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–21 June 2009; pp. 96–103.
42. Min, Z.; Yisha, L.; Yan, Z.; Huosheng, H. Visual Campus Road Detection for an UGV using Fast Scene Segmentation and Rapid Vanishing Point Estimation. *IFAC Proc. Vol.* **2014**, *47*, 11898–11903.
43. Deng, T.; Yan, H.; Li, Y. Learning to Boost Bottom-Up Fixation Prediction in Driving Environments via Random Forest. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 3059–3067. [CrossRef]
44. Moghadam, P.; Starzyk, J.A.; Wijesoma, W.S. Fast Vanishing-Point Detection in Unstructured Environments. *IEEE Trans. Image Process.* **2012**, *21*, 425–430. [CrossRef]
45. Shi, J.; Wang, J.; Fu, F. Fast and Robust Vanishing Point Detection for Unstructured Road Following. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 970–979. [CrossRef]
46. Fan, X.; Shin, H. Road vanishing point detection using weber adaptive local filter and salient-block-wise weighted soft voting. *IET Comput. Vis.* **2016**, *10*, 503–512. [CrossRef]

47. Yong, L.; Guofeng, T.; Anan, S.; Weili, D. Road extraction algorithm based on intrinsic image and vanishing point for unstructured road image. *Robot. Auton. Syst.* **2018**, *109*, 86–96.

48. Bui, T.H.; Saitoh, T.; Nobuyama, E. Road area detection based on texture orientations estimation and vanishing point detection. In Proceedings of the The SICE Annual Conference, Nagoya, Japan, 14–17 September 2013; pp. 1138–1143.

49. Neethu, J.; Anusha, B.; Krishnan, K. A Reliable Method for Detecting Road Regions from a Single Image Based on Color Distribution and Vanishing Point Location. *Procedia Comput. Sci.* **2015**, *58*, 2–9.

50. Zhaozi, Z.; Yingtuan, H.; Dixiao, C.; Jianru, X. Real-time road detection with image texture analysis-based vanishing point estimation. In Proceedings of the IEEE International Conference on Progress in Informatics and Computing (PIC), Nanjing, China, 18–20 December 2015; pp. 454–457.

51. Le, M.C.; Phung, S.L.; Bouzerdoum, A. Pedestrian Lane Detection in Unstructured Environments for Assistive Navigation. In Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA), Wollongong, Australia, 25–27 November 2014; pp. 1–8.

52. Nguyen, L.; Phung, S.L.; Bouzerdoum, A. Efficient Vanishing Point Estimation for Unstructured Road Scenes. In Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, Australia, 30 November–2 December 2016; pp. 1–6.

53. Negahdaripour, S.; Horn, B.K.P. A direct method for locating the focus of expansion. *Comput. Vis. Graph. Image Process. J.* **1989**, *46*, 303–326. [CrossRef]

54. Shi, J.; Tomasi, C. Good features to track. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 21–23 June 1994; pp. 593–600.

55. Lucas, B.D.; Kanade, T. An Iterative Image Registration Technique with an Application to Stereo Vision. In Proceedings of the 7th international joint conference on Artificial intelligence (IJCAI'81), Vancouver, BC, Canada, 24–28 August 1981; pp. 674–679.

56. Jiqing Expressway Dataset. Available online: https://github.com/vonsj0210/Multi-Lane-Detection-Dataset-with-Ground-Truth (accessed on 8 February 2021).

57. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]

58. Bay, H.; Tuytelaars, T.; Van Gool, L. SURF: Speeded-Up Robust Features. *Comput. Vis. Image Underst.* **2008**, 404–417. [CrossRef]

59. David, A.F.; Jean, P. Chapter 17: Segmentation and fitting using probabilistic methods. In *Computer Vision: A Modern Approach*; Prentice Hall Professional Technical Reference; Prentice Hall: Upper Saddle River, NJ, USA, 2002; p. 483. ISBN 978-0-13-085198-7.

60. Zhou, K.; Wang, K.; Yang, K. A Robust Monocular Depth Estimation Framework Based on Light-Weight ERF-PSPNet for Day-Night Driving Scenes. In Proceedings of the International Conference on Machine Vision and Information Technology (CMVIT), Sanya, China, 20–22 February 2020.

61. ZJU Day and Night Driving Dataset. Available online: https://github.com/elnino9ykl/ZJU-Dataset (accessed on 1 March 2021).