IEICE TRANS. INF. & SYST., VOL.E96–D, NO.3 MARCH 2013

PAPER

A Texture-Based Local Soft Voting Method for Vanishing Point Detection from a Single Road Image

Trung Hieu BUI[†a)], ***Nonmember***, **Eitaku NOBUYAMA**[††b)], ***and*** **Takeshi SAITOH**[††c)], ***Members***

SUMMARY Estimating a proper location of vanishing point from a single road image without any prior known camera parameters is a challenging problem due to limited information from the input image. Most edge-based methods for vanishing point detection only work well for structured roads with clear painted lines or distinct boundaries, while they usually fail in unstructured roads lacking sharply defined, smoothly curving edges. In order to overcome this limitation, texture-based methods for vanishing point detection have been widely published. Authors of these methods often calculate the texture orientation at every pixel of the road image by using directional filter banks such as Gabor wavelet filter, and seek the vanishing point by a voting scheme. A local adaptive soft voting method for obtaining the vanishing point was proposed in a previous study. Although this method is more effective and faster than prior texture-based methods, the associated computational cost is still high due to a large number of scanning pixels. On the other hand, this method leads to an estimation error in some images, in which the radius of the proposed half-disk voting region is not large enough. The goal of this paper is to reduce the computational cost and improve the performance of the algorithm. Therefore, we propose a novel local soft voting method, in which the number of scanning pixels is much reduced, and a new vanishing point candidate region is introduced to improve the estimation accuracy. The proposed method has been implemented and tested on 1000 road images which contain large variations in color, texture, lighting condition and surrounding environment. The experimental results demonstrate that this new voting method is both efficient and effective in detecting the vanishing point from a single road image and requires much less computational cost when compared to the previous voting method.

key words: *vanishing point, texture-based, Gabor filter, soft voting*

1. Introduction

Using computer vision techniques to detect drivable road areas plays a very important role in navigating autonomous vehicle systems. Recently, numerous interesting lane and road detection algorithms have been widely published for urban and highway roads [1]–[3], structured roads [4]–[8] and unstructured roads [9]–[12]. In all of these studies, estimating the vanishing point (VP) is a key requirement because the correctly obtained VP provides a strong clue to the localization of the road region.

State-of-the-art vision-based VP detection methods can be mainly grouped into three categories: edge-based methods [1], [5], [6], prior-based methods [3], [8] and texture-based methods [9]–[11]. Most edge-based methods often include three steps: i) extract edge pixels by an edge detector ii) detect straight lines by a linear transformation and iii) obtain the VP by a voting algorithm. For instance, in [5], edge pixels are extracted by the Canny detector [13], and then straight lines are detected by the Hough transform, finally the intersections of any pair of lines vote for VPs on another Hough space. In general, these edge-based methods can be applied to real-time systems due to their computational efficiency. However, the disadvantage is that they only work well for structured roads with clear painted lines or distinct borders, while they usually fail in unstructured roads lacking sharply defined, smoothly curving edges.

In order to overcome the limitation of these edge-based methods, prior-based methods and texture-based methods for VP detection have been proposed recently. For instance, the prior-based method proposed in [8] is robust to varying imaging conditions, road types and scenarios by integrating contextual three-dimensional information with low-level cues. This contextual information includes horizon lines estimated by the method in [14], three-dimensional scene layout computed by the method in [15], three-dimensional road geometry inferred by the method in [16], and so on. From the viewpoint of computational cost, the method in [8] is time-consuming cause of integrating several different techniques. On the other hand, the global perspective structure matching method proposed by Wu et al. [3] requires a large-scale image or video training database and also manual works for labeling the VPs for the training stage. Therefore, such prior-based methods are inapplicable to detect the VP from a single road image. In contrast, texture-based methods for VP detection [9]–[11] are very effective for both structured and unstructured roads by utilizing the texture information from a single road image. All of these studies consist of three steps: i) calculate the texture orientations by applying a directional filter bank ii) determine the voters and VP candidates and iii) vote for obtaining the VP by a voting algorithm. For instance, Rasmussen [9], [10], as well as Kong et al. [11] uses the same Gabor wavelet filters introduced in [17] to compute dominant texture orientations before applying voting algorithms to obtain the VP. A global voting method for detecting VPs was proposed by Rasmussen [9], [10]. However, the computational cost of this method is very high due to a large number of scanning voters and VP candidates. Moreover, as pointed

Manuscript received July 25, 2012.
Manuscript revised November 9, 2012.
[†]The author is with the Department of Information Science, Graduate School of Computer Science and Systems Engineering, Kyushu Institute of Technology, Iizuka-shi, 820–8502 Japan.
[††]The authors are with the Department of Systems Design and Informatics, Kyushu Institute of Technology, Iizuka-shi, 820–8502 Japan.
a) E-mail: btrhieu@bach.ces.kyutech.ac.jp
b) E-mail: nobuyama@ces.kyutech.ac.jp
c) E-mail: saitoh@ces.kyutech.ac.jp
DOI: 10.1587/transinf.E96.D.690

(a) Proper detected VPs.

(b) Improper detected VPs.

Fig. 1 Examples of VP detection (pink crosses) by the LASV method.

out in [11], this global voting tends to favor VPs that are high in the image, leading sometimes to large estimation errors, especially when the true VP is in the lower part of the road image. In order to overcome these problems, Kong et al. [11] proposed an effective VP detection method, in which a "confidence level" function and a local adaptive soft voting (LASV) method were proposed. The confidence function is used to determine the voters by checking the reliability of the obtained texture orientations. Before applying the LASV method, a half-disk voting region is created for each VP candidate, and only voters within this half-disk vote for the VP candidate. The LASV method for VP detection performs well in general road images, especially in unstructured road images (see the images of Fig. 1 (a)). In addition, this method is more effective and faster than previous texture-based methods [9], [10]. However, the computational cost of the LASV method is still high due to a large number of scanning pixels. Besides, this method yields an estimation error in some images (see the images of Fig. 1 (b)), in which most voters in the lower part of the image cannot vote for the true VP because the radius of the proposed half-disk voting region is not large enough.

Learning both advantages and disadvantages of current texture-based methods encourages us to propose a new lower-computational-cost and higher-accuracy local soft voting method to detect the VP from a single road image. The basic concepts of the proposed method are: i) to reduce the number of confidential voters and ii) to scan the voters instead of the VP candidates (note that the number of voters is much lower than the number of VP candidates).

In order to reduce the number of voters, the threshold for the confidence levels is set to be higher than that in the LASV method. In the voting process, a new VP candidate region is defined for each voter, and a new local soft voting function is proposed. Each voter votes for all the pixels in its VP candidate region, with the voting score calculated by a local soft voting function proposed in this paper. The proposed method has been implemented and tested on 1000 road images which contain large variations in color, texture, lighting condition and surrounding environment. The experimental results demonstrate that this new method is both efficient and effective in detecting the VP and requires less computational cost when compared to the LASV method [11].

The remainder of the present paper is organized as follows. Related research is reviewed in Sect. 2, and the Gabor filters and the confidence level function introduced in [11] are explained in Sect. 3. A VP candidate region for each voter and a new local soft voting method are proposed in Sect. 4, and the proposed method is summarized in Sect. 5. In Sect. 6, experimental results are demonstrated to show the effectiveness of the proposed method. Section 7 presents our conclusions.

2. Related Research

As stated above, previous texture-based methods [9]–[11] have attempted to detect the VP based on texture orientation calculation. In all of these studies, Gabor filters are applied in order to compute the texture orientation at each pixel of a road image. A VP is then detected using a voting algorithm.

A global hard voting method is proposed in [9], [10] as the voting algorithm. In this algorithm, all the pixels of the image can be VP candidates, and the voting region of a VP candidate is defined as the entire image below the VP candidate. The left-hand figure in Fig. 2 shows an example of the voting region, where V is a VP candidate and V_R is the voting region of V. (Note that the gray frame around the image is a region in which the convolution with Gabor filters cannot be calculated. In the present case, the width of the region is eight pixels.) A pixel P in V_R votes for V with a fixed voting score if the angle $\gamma = \angle(\overrightarrow{PV}, \overrightarrow{O_P})$ is below a certain threshold, where the vector $\overrightarrow{O_P}$ denotes the texture orientation at P and γ denotes the angle between the $\overrightarrow{PV}$ and $\overrightarrow{O_P}$ directions. A VP is detected as the pixel having the highest voting score. The disadvantages of this method are: i) the computational cost is very high because a large voting region is scanned for each VP candidate and ii) improper VPs are detected in several cases because a voting score is fixed irrespective of the distance between P and V.

In order to overcome these disadvantages, a half-disk voting region and a soft voting method, referred to as local adaptive soft voting (LASV), were introduced in [11]. In this previous paper, Gabor filters are also used to compute the texture orientation at every pixel of the road image. Moreover, in [11], a confidence level function is introduced in order to discard the pixels for which the estimated texture orientations are not reliable. If the confidence level exceeds a certain threshold the pixel is kept as a voter; otherwise the pixel is discarded. The remaining pixels are referred to as the remaining voters, which are used to detect a VP. In this method, a VP candidate V is searched for in the uppermost 90% of pixels of the entire image, and the voting region V_R of V is defined as a half-disk below V centered at V (see the right-hand figure of Fig. 2). The radius R of this half-disk is set to $0.35 \times H$, where H is the height of the image. Each remaining voter P inside V_R votes for V with a voting score calculated by a voting function

$$\mathrm{Vote(P, V)} = \begin{cases} \frac{1}{1+[\gamma \times d(\mathrm{P,V})]^2} & \text{if } \gamma \leq \frac{5}{1+2d(\mathrm{P,V})} \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

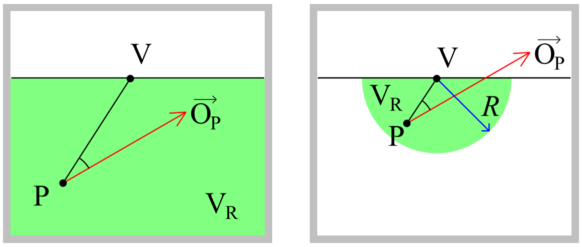


Fig. 2 Global voting method and LASV method.

where $d(\mathrm{P}, \mathrm{V})$ denotes the distance between P and V divided by the diagonal length of the input image. The pixel having the highest voting score is selected as a VP. As mentioned in Introduction, this method is more effective and faster than previous global voting methods [9], [10]. However, the computational cost of the LASV method is still high, and this method yields an estimation error in some images, in which the remaining voters far from the true VP cannot vote for the true VP because R is not large enough. Our experimental results demonstrate that, although using the half-disk voting region with a larger value of R may improve the estimation performance of the LASV method, it also increases the computational cost of the algorithm. These experimental results will be described in detail in Sect. 6.3.

3. Texture Orientation and Confidential Level

In this section, the texture orientation calculation method and the confidence level function introduced in [11] are briefly explained.

Gabor filters are used to calculate the texture orientation at each pixel of a road image. For a scale ω and an orientation ϕ, the Gabor filter is defined as follows:

$$\Psi_{\omega,\phi}(x, y) = \frac{\omega}{\sqrt{2\pi c}} e^{-\omega^2(4a^2+b^2)/(8c^2)}(e^{ia\omega} - e^{-c^2/2}), \quad (2)$$

where $a = x\cos\phi + y\sin\phi$, $b = -x\sin\phi + y\cos\phi$, and c is a constant. As in [11], we will use the Gabor filters with 36 orientations, five scales, and $c = 2.2$, i.e., ϕ is chosen to be from 0° to 175° with an angle interval of 5°, and ω is chosen to be from 1 to 5 with a scale interval of 1. Figure 3 shows the real filters and the imaginary filters of the Gabor filters with 36 orientations and five scales, where each filter consists of 17×17 pixels.

For the gray level value of a road image $I(z)$ at $z = (x, y)$ the convolution of the image and a Gabor filter is defined by

$$G_{\omega,\phi}(z) = I(z) \otimes \Psi_{\omega,\phi}(z), \quad (3)$$

and the response image $R_\phi(z)$ for the orientation ϕ is calculated as the average of the square norm of $G_{\omega,\phi}$ at different scales, as follows:

$$R_\phi(z) = \text{average}_\omega \left\{ \left(\mathrm{Re}(G_{\omega,\phi})\right)^2 + \left(\mathrm{Im}(G_{\omega,\phi})\right)^2 \right\}. \quad (4)$$

Then the texture orientation angle $\theta(z)$ at z is defined in terms of the maximum average response as follows:

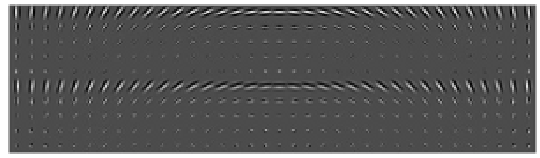
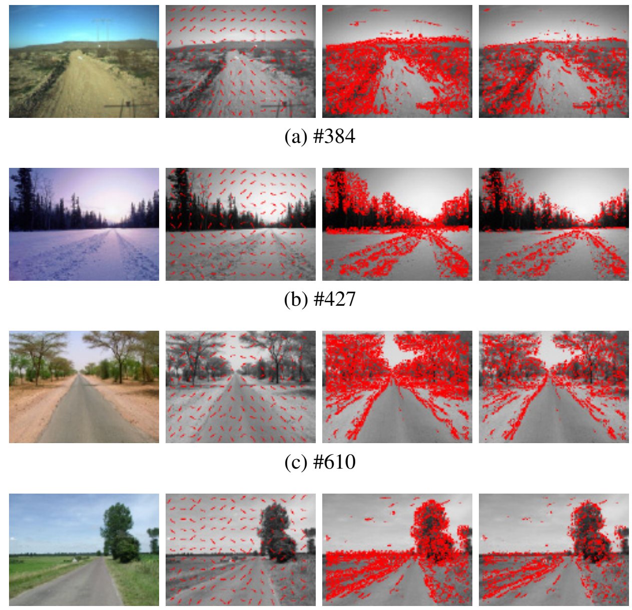


Fig. 3 Gabor filters with 36 orientations and five scales.



(a) #384

(b) #427

(c) #610

(d) #626

Fig. 4 Remaining voters for the LASV method and the proposed method (original images, orientations computed by Gabor filters, remaining voters determined by the confidence function of the LASV method, remaining voters determined by the confidence function of the proposed method).

$$\theta(z) = \text{argmax}_\phi R_\phi(z). \quad (5)$$

In order to define a confidence level, let $r_1(z) > \cdots > r_{36}(z)$ be the ordered values of the Gabor response for the 36 considered orientations (in particular, $r_1(z) = R_{\theta(z)}(z)$). Then, a confidence level function is defined by

$$\mathrm{Conf}(z) = 1 - \frac{\text{average}\{r_5(z), \ldots, r_{15}(z)\}}{r_1(z)}. \quad (6)$$

In [11], all the pixels that have confidence levels that are smaller than

$$\delta \times \left(\max_z \mathrm{Conf}(z) - \min_z \mathrm{Conf}(z)\right), \quad (7)$$

with $\delta = 0.3$ are discarded. The remaining pixels become voters that are referred to as the remaining voters in the voting process.

The experimental results of the present study demonstrate that too many off-road pixels remain when using the threshold $\delta = 0.3$. Since a high number of remaining voters increases the computational cost in the voting process, we use $\delta = 0.5$ in order to reduce the number of remaining voters. In the examples shown in Fig. 4, the red points in the third column indicate the remaining voters for $\delta = 0.3$

Table 1 Number of remaining voters.

	#384	#427	#610	#626
$\delta = 0.3$	13,332	10,118	14,595	10,346
$\delta = 0.5$	8,155	6,074	9,072	5,568

and those in the last columns indicate the remaining voters for $\delta = 0.5$. The numbers of the remaining voters are listed in Table 1. The number of remaining voters for $\delta = 0.5$ is reduced by approximately 38~46% compared to that for $\delta = 0.3$. The experimental results reveal that by reducing the number of remaining voters, the total computational time of the algorithm can be reduced approximately 11.48%.

4. Proposed Local Soft Voting Method

In this section, a VP candidate region is introduced, as well as a new local soft voting method is proposed in order to improve the estimation performance and reduce the computational cost of the algorithm. The LASV method [11] is performed as follows: i) scan the VP candidates as the uppermost 90% of the pixels in the image ii) create a half-disk voting region for each VP candidate iii) calculate the voting score received by each VP candidate from the remaining voters in its half-disk voting region and iv) obtain the VP as the VP candidate having the largest voting score. As stated in Introduction, the computational cost of the LASV method is high due to a large number of scanning pixels, and this method yields an estimation error in some images.

In our method, the basic idea for reducing the computational cost is to scan the remaining voters instead of the VP candidates. In order to construct the VP candidate region for each remaining voter, two conditions are considered: i) the VP candidates of a remaining voter are always above it in the image and ii) the angle between the direction from a remaining voter to a VP candidate and the texture orientation at that remaining voter is smaller than a certain threshold. The remaining voter votes for the pixels in its VP candidate region. Figure 5 shows examples of the VP candidate region: a circular sector in the left-hand figure and a triangular region in the right-hand figure, where P is a remaining voter, $\overrightarrow{O_P}$ is its texture orientation, and ϵ is the angle tolerance. Initially, the circular sector is more intuitive than the triangular region for use in the proposed method. Note that in the LASV method every remaining voter votes for the pixels whose distance from the voter is less than R with $\gamma \leq \frac{5}{1+2d(\mathrm{P},\mathrm{V})}$, which means the VP candidate region of a voter in the LASV method is included in the circular sector depicted in the left side of Fig. 5.

However, from the viewpoint of computational cost, scanning the inside of the triangular region is much simpler. Thus, we use the triangular region. Note that once R (:= |PQ|, the length of the line segment PQ) and ϵ are given, the vertices I and J are determined automatically. As a voting process, a local soft voting method is adopted in which the remaining voter votes for the VP candidate in its VP candidate region all the more as it is close to the VP candidate,

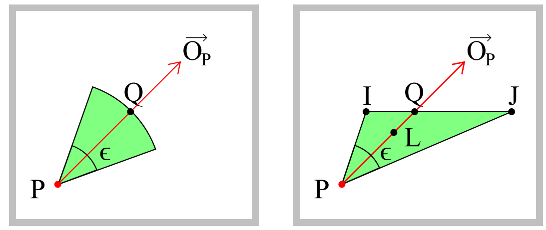


Fig. 5 VP candidate regions.

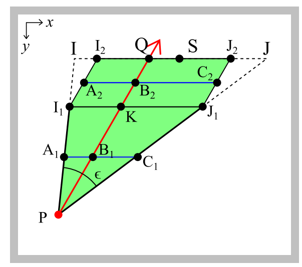


Fig. 6 Modified VP candidate region.

and the angle between the orientation of its texture and the direction from it to the VP candidate is close to zero. This indicates that the voting scores of Q should be smaller than that of L and larger than that of I or J (see the right-hand figure of Fig. 5). The proposed voting score function is defined as follows:

$$\mathrm{Vote}(\mathrm{P},\mathrm{V}) = \frac{\exp(-\alpha/\beta)}{1 + d(\mathrm{P},\mathrm{V})^2}, \tag{8}$$

where V is a pixel (a VP candidate) in the VP candidate region of P, α is the distance from V to the pixel on PQ with the same y coordinate of V, and β is a constant parameter. Note that the voting score decreases as the distance $d(\mathrm{P},\mathrm{V})$ or α increases, and the voting scores of pixels near I and J are approximately equal to zero. Hence, the pixels near I and J of the triangular region are useless for the soft voting process. Based on these observations, the VP candidate region is modified from the triangular region to a shape that is a combination of a triangle and a parallelogram, as shown in Fig. 6. The fundamental strategy to construct this modified VP candidate region is to reduce the computational time. This modified VP candidate region can be drawn if PK and PQ are determined. In our method, we use $|\mathrm{PK}| = 0.50 \times H$ and $|\mathrm{PQ}| = 0.65 \times H$, which yields the best performance in all our experiments. The constant parameter β is selected to satisfy that the voting score of the pixel very near Q is approximately equal to the voting score of Q, and the voting score of the pixel very near K is approximately equal to the voting score of K (see Fig. 6). For instance, the voting score of Q (with $\alpha = 0$) should be approximately equal to the voting score of S (with $\alpha = 1$) in Fig. 6 (note that Q and S have the same y coordinate).

Figure 7 shows an example of the voting score of Q (with $\alpha = 0$) and S (with $\alpha = 1$) in the case that the texture orientation at P is 45°. In the figure, the black dashed line indicates the voting score of Q, and the red line indicates the

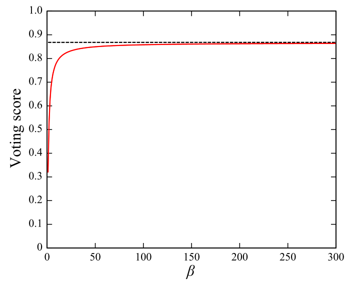


Fig. 7 An example of voting score for Q ($\alpha = 0$) and S ($\alpha = 1$).

voting score of S. We see that, when β increases, the voting score of S becomes approximately equal to that of Q. Our experimental results reveal that when the texture orientation at P varies from 0° to 175°, by using $\beta = 180$, we obtain the best performance of VP detection.

Next, we summarize the proposed local soft voting method.

Step 1 Let M be a two-dimensional matrix of the same size as the road image, and set all the elements of M to zero. Let $R_1 := 0.50 \times H$ and $R_2 := 0.65 \times H$.

Step 2 For each remaining voter $\mathrm{P}(x_0, y_0)$, calculate the coordinates of the points $\mathrm{K}(x_1, y_1)$ and $\mathrm{Q}(x_2, y_2)$ as follows:

$$x_1 := x_0 - \sin(\theta) \times R_1, \tag{9}$$

$$y_1 := y_0 + \cos(\theta) \times R_1, \tag{10}$$

$$x_2 := x_0 - \sin(\theta) \times R_2, \tag{11}$$

$$y_2 := y_0 + \cos(\theta) \times R_2, \tag{12}$$

where θ is the texture orientation angle at P, and repeat procedures (a) and (b) below:

(a) (Calculation of voting scores in the triangle $\mathrm{PI_1J_1}$) For $x := x_1$ to x_0, calculate

$$y_{10} := y_0 + (x - x_0)/\tan(\theta), \tag{13}$$

$$y_{11} := y_0 + (x - x_0)/\tan(\theta + \epsilon), \tag{14}$$

$$y_{12} := y_0 + (x - x_0)/\tan(\theta - \epsilon), \tag{15}$$

where $\mathrm{B}_1(x, y_{10})$, $\mathrm{A}_1(x, y_{11})$, and $\mathrm{C}_1(x, y_{12})$ (see Fig. 6), and calculate the voting score repeatedly for $y := y_{11}$ to y_{12}

$$M(x, y) = M(x, y) + \frac{\exp(-|y - y_{10}|/\beta)}{1 + d(\mathrm{P}, \mathrm{V})^2}, \tag{16}$$

$$d(\mathrm{P}, \mathrm{V})^2 = ((x - x_0)^2 + (y - y_0)^2)/\mathrm{Diag}^2, \tag{17}$$

where $\mathrm{V}(x, y)$, $\beta = 180$, and Diag denotes the diagonal length of the input image.

(b) (Calculation of voting scores in the parallelogram $\mathrm{I_1J_1J_2I_2}$) For $x := x_2$ to $x_1 - 1$, calculate

$$y_{20} := y_0 + (x - x_0)/\tan(\theta), \tag{18}$$

$$y_{21} := y_{20} - |\mathrm{KI_1}|, \tag{19}$$

$$y_{22} := y_{20} + |\mathrm{KJ_1}|, \tag{20}$$

where $\mathrm{B}_2(x, y_{20})$, $\mathrm{A}_2(x, y_{21})$, and $\mathrm{C}_2(x, y_{22})$ (see Fig. 6), and calculate the voting score repeatedly for $y = y_{21}$ to y_{22}

$$M(x, y) := M(x, y) + \frac{\exp(-|y - y_{20}|/\beta)}{1 + d(\mathrm{P}, \mathrm{V})^2}. \tag{21}$$

Step 3 Find the element of M that has the largest value, and let its index be the coordinate of the VP.

5. Algorithm Summary

To obtain the VP, the proposed method is performed as follows:

Step 1 Calculate the texture orientation at every pixel of the road image using Gabor filters with five scales and 36 orientations (Sect. 3).

Step 2 Keep the pixels having confidence levels that exceed the threshold (7), with $\delta = 0.5$ as remaining voters (Sect. 3).

Step 3 Perform the proposed local soft voting method to obtain a VP (Sect. 4).

6. Experimental Results

6.1 Image Dataset

The proposed method is compared to the LASV method using numerical examples. Most of these images have been used by Kong et al. in [11]; the remainder were downloaded from the Internet by using Google Image. Among them, about 600 images are unstructured roads, and about 400 images are structured roads. These road images contain large variations in color, texture, lighting condition and surrounding environment without any prior known camera parameters, some of them are shown in Fig. 8. Since these images are of very different size, all images are normalized to the same size (height: 180 pixels, width: 240 pixels) by using the bicubic image interpolation method [18]. To assess the algorithm's performance versus human perception of the VP location, we invited 7 persons to manually mark the VP location in each image in this collection after they are trained to know the vanishing point concept.

Since the marked VPs in each image are very close, we defined the center of these marked locations as the ground truth VP location. In order to measure the accuracy of VP estimation algorithm, we use the normalized Euclidean distance, where the Euclidean distance between the estimated VP and the ground truth is normalized by the diagonal length of the road image as follows:

$$\mathrm{NormDist} = \frac{\sqrt{(x_e - x_g)^2 + (y_e - y_g)^2}}{\mathrm{Diag}}, \tag{22}$$

where (x_e, y_e) is the estimated VP position, and (x_g, y_g) is the marked ground truth position. Since the input image is normalized to 180×240 pixels, the normalized Euclidean

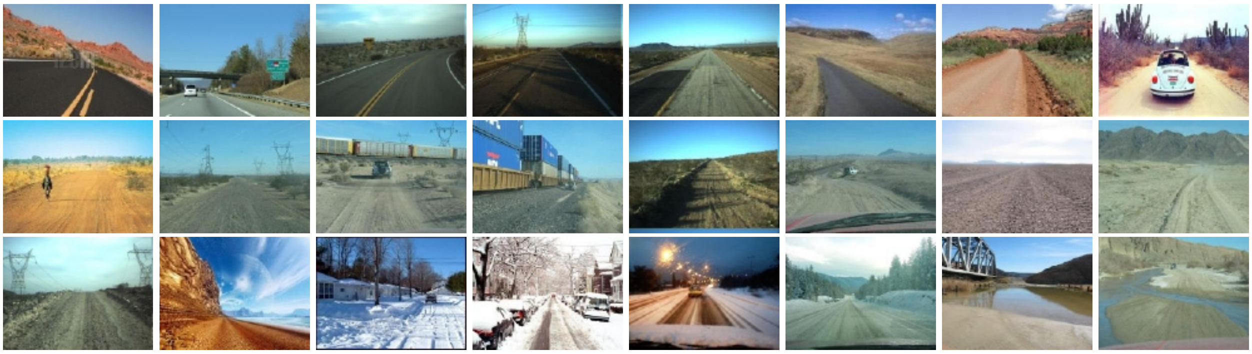


Fig. 8 Different road types with varying colors, textures, and illumination conditions.

Table 2 Performance of different VP estimation algorithms for 1000 tested images.

Average	Methods					
	Soft+Modified	Soft+Triangle	Hard+Modified	Hard+Triangle	LASV0	LASV1
NormDist	0.0729	0.0737	0.0739	0.0735	0.0948	0.0709
Total time (s)	5.213	5.245	4.388	4.400	33.820	24.815
Prep. time (s)	0.012	0.012	0.012	0.012	0.117	0.117
Voter determination time (s)	3.966	3.966	3.966	3.966	3.966	3.966
Voting time (s)	1.235	1.267	0.410	0.422	29.737	20.732

distance of 0.1 in (22) means that the location of the estimated VP is about 30 pixels away from that of the marked ground truth.

6.2 Comparing Experimental Results

In order to assert the effectiveness of the proposed local soft voting method and the new VP candidate region, we compare the performances of six VP detection algorithms. Table 2 shows the experimental results for 1000 tested images. In this table, the first four methods (with $\delta = 0.5$) refer to the method described in Sect. 4. In particular, the "Soft+Modified" denotes the proposed method, and the "Soft+Triangle" denotes the method using a triangular VP candidate region. The "Hard+Modified" denotes the method using a hard voting strategy, and the "Hard+Triangle" denotes the method using a triangular VP candidate region and a hard voting strategy. (Note that the hard voting strategy is performed by replacing (8) with $\mathrm{Vote(P, V)} = 1$.) The "LASV0" denotes the LASV method with the radius R of the half-disk voting region being set to $0.35 \times H$ and $\delta = 0.3$ as proposed in [11]. The "LASV1" denotes the LASV method with R being set to $0.65 \times H$ and $\delta = 0.6$ (note that this "LASV1" method yields the best estimation performance for the LASV method in all our experiments). The "NormDist" denotes the average normalized Euclidean distance (note that a smaller value means the estimated VP location is closer to the location of the ground truth), and the "Total time" denotes the average computational time for each method. The "Prep. time" denotes the average preprocessing time, the "Voter determination time" denotes the average time for calculating the Gabor convolution and confidence level estimation, and the "Voting time" denotes the average time for the voting process for each method.

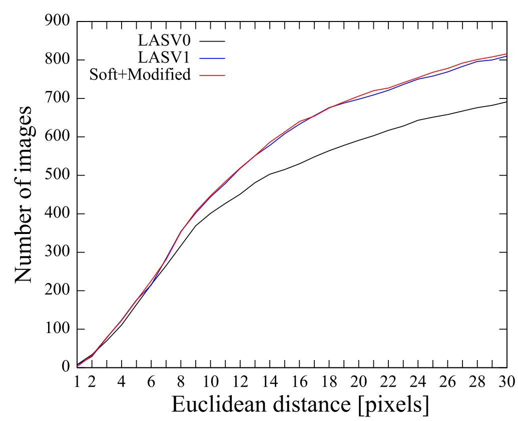
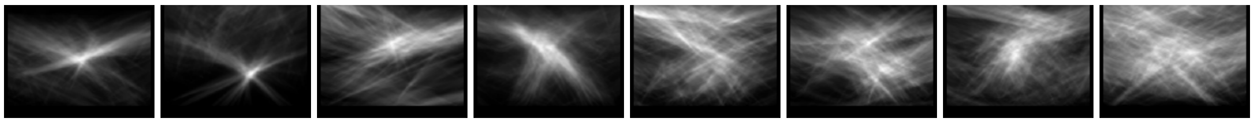


Fig. 9 Comparison of VP estimation performance.

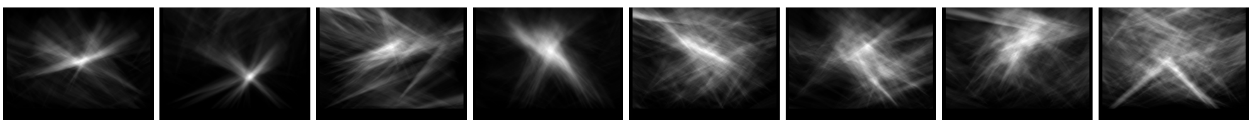
Firstly, we compare the estimation performances of the proposed method (Soft+Modified) and the other three methods (Soft+Triangle, Hard+Modified, Hard+Triangle). From Table 2, it can be seen that the proposed method yields the best estimation result among the four methods. Secondly, we compare the estimation performances of the proposed method, the LASV0 method, and the LASV1 method. The experimental results in Table 2 reveal that the estimation performance of the proposed method is considerably better than that of the LASV0 method, and almost the same as the LASV1 method (the difference between the NormDist values of the proposed method and the LASV1 method is 0.0020 which represents a value less than one pixel). In order to investigate the details of these experimental results, we evaluate the VP estimation performance while changing the threshold for Euclidean distances. Figure 9 shows a comparison of VP estimation performance between the

(a) Original images and ground truth locations of VPs (pink crosses).



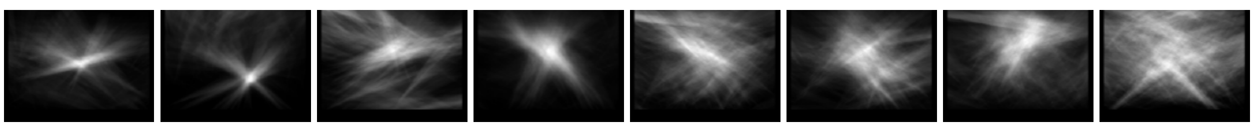
(b) Voting maps by the LASV0 method.

(c) Detected VPs by the LASV0 method (pink crosses).



(d) Voting maps by the LASV1 method.

(e) Detected VPs by the LASV1 method (pink crosses).



(f) Voting maps by the proposed method.

(g) Detected VPs by the proposed method (pink crosses).

Fig. 10 Examples of VP detection by the LASV method and the proposed method.

three methods. In the figure, the horizontal axis represents the Euclidean distances in pixels, while the vertical axis represents the number of images whose VP estimation error is less than the corresponding Euclidean distance. From this figure, it can be seen that the estimation performance of the proposed method is better than that of the LASV0 method, and almost the same as the LASV1 method.

Next, the computational times of the three methods are discussed. From Table 2, it can be seen that the average computational time of the proposed method is considerably less than that of the LASV0 method and the LASV1 method. In particular, the "Total time" and the "Voting time" of the proposed method are approximately five times and 17 times less than those of the LASV1 method, respectively. The "Total time" and the "Voting time" of the proposed method are approximately 6.5 times and 24 times less than those of the LASV0 method, respectively. These results show that the proposed voting strategy requires much less computational cost than the LASV method. Note that the "Prep. time" of the proposed method is slightly less than that of the LASV method due to the difference in the number of preprocessing steps in which a median filter is applied in both methods and in addition a vertical edge elimination method is applied in the LASV method (our experimental results turn out that the proposed method without using the vertical edge elimination method is better and faster than the proposed method with using the vertical edge elimination method, in which the "NormDist" and the "Total time" are 0.0732 and 5.311(s), respectively. Hence, the vertical edge elimination method is not used in the proposed method). These numerical examples are performed using Matlab, running on a Core 2 Duo (3.5-GB RAM) machine, and the Matlab m-files for the LASV method were provided by the author of [11].

Figure 10 visually gives a comparison of VP detection on some sample images. In the figure, the VPs detected by

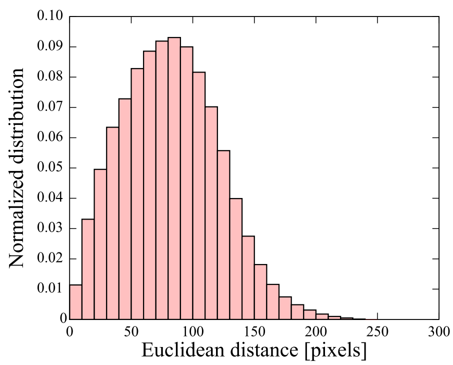


Fig. 11 Average distribution of distances from the remaining voters to the VP for 1000 road images.

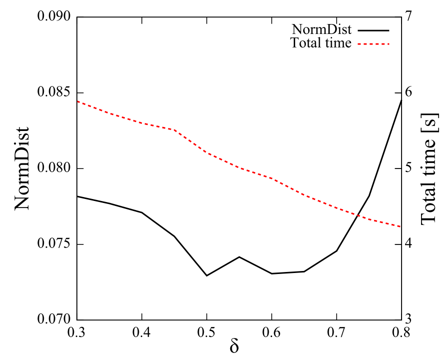


Fig. 12 The effect of δ on the VP estimation performance and the computational cost for the proposed method.

the LASV0 method, the LASV1 method and the proposed method are shown in (c), (e) and (g), respectively. Their first four columns show image examples in which all the three methods detect the VPs which almost coincide with the corresponding ground truth. Their last four columns show image examples in which both estimation performances of the LASV1 method and the proposed method are better than that of the LASV0 method.

6.3 Effect of Radius on Performance

In this section and the next section, we discuss about the two important parameters R and δ. In this section, we focus on the effect of radius R on the estimation performance. The LASV1 method (with $R = 0.65 \times H$ and $\delta = 0.6$) yields a better estimation accuracy than the LASV0 method (with $R = 0.35 \times H$ and $\delta = 0.3$). In order to investigate these results from the viewpoint of radius R, an additional experiment for the LASV method is carried out with $R = 0.35 \times H$ and $\delta = 0.6$ (R is smaller than that of the LASV1 method while δ is the same), and this is denoted by "LASV2". As a result, the average NormDist of the LASV2 method is 0.0999, which is worse than that of the LASV1 method. This result indicates that a small radius of a small voting region yields a worse estimation performance than a larger one. This can be confirmed by the average distribution graph shown in Fig. 11. In the figure, the horizontal axis represents the distances from the ground truth VPs in pixels, while the vertical axis represents the normalized distributions of remaining voters. The normalized distribution value for $\mathrm{d} \leq \mathrm{x} < \mathrm{d} + 10$ ($\mathrm{d} = 0, 10, 20, \ldots, 240$; x: Euclidean distance from the ground truth VP) is defined as the number of the remaining voters satisfying $\mathrm{d} \leq \mathrm{x} < \mathrm{d} + 10$ divided by the total number of the remaining voters. The graph in Fig. 11 shows the average normalized distributions of 1000 images. It can be seen from the graph that the number of remaining voters near the VP is small, which indicates that a small radius of a voting region cannot cover the many voters which possibly vote for the VP. This leads to the fact that the performance of the LASV0 method is worse than that of the LASV1 method. Also in our proposed method, the graph implies that a short $|PQ|$ (i.e., a small VP candidate region) is not suitable for most images because many remaining voters which possibly vote for the VP cannot vote for it when $|PQ|$ is short. Note that in general the pixels near the VP correspond to the road area far from the camera in the real world, and hence these pixels tend to be blurred. This explains that the number of remaining voters near the VP is small, because most of the blurred pixels cannot be remaining voters since their confidence levels of orientations are generally low.

6.4 Effect of δ on Performance

In this section, we discuss about δ. In order to evaluate how the threshold δ affects the estimation performance and the computational cost, we vary δ of the proposed method from 0.30 to 0.80 with an interval of 0.05, and the result is shown in Fig. 12. From the figure, it can be seen that the computational cost almost monotonously decreases with respect to the value of δ. On the other hand, a large value of δ and a small value of δ decrease the estimation performance. When δ is varied from 0.50 to 0.70, the proposed method yields almost the same estimation performance (with less than one pixel difference). From the viewpoint of estimation performance, we use $\delta = 0.50$ which yields the best result in our method.

7. Conclusions

Estimating a proper location of VP from a single road image without any prior known camera parameters is a challenging problem due to limited information from the input image. The LASV method [11] for VP detection is very effective for both structured and unstructured roads, and faster than previous texture-based method. However, the computational cost is still high due to a large number of scanning pixels. In addition, an estimation error is obtained in some images, in which the radius of the proposed half-disk voting region is not large enough. In this paper, a new local soft voting method has been proposed to overcome the limitations of the LASV method. In order to reduce the computational cost: i) the number of remaining voters is reduced by introducing a new threshold and ii) the number of scanning pixels is reduced by scanning the remaining voters instead of the VP candidates. On the other hand, to improve the estimation performance, a new VP candidate region and a new soft voting function are introduced. In order to assert the effectiveness of the proposed algorithm, the proposed method and the LASV method have been implemented and tested on 1000 road images which contain large variations in color, texture, lighting condition and surrounding environment. The experimental results reveal that: i) the proposed method outperforms better than the LASV method which uses a small voting region, especially in some images in which most remaining voters are far from the VP and ii) the computational cost of the proposed method is considerably less than that of the LASV method, the computational time for the Gabor convolution and confidence level estimation which accounts for most of the computational time in our proposed method is the same for the LASV method, whereas, the computational time of the proposed method for the voting process is much less than that of the LASV method.

Acknowledgement

The present study was supported in part by the Regional Innovation Cluster Program (Global Type [2nd Stage]).

References

[1] G. Zhang, N. Zheng, C. Cui, Y. Yan, and Z. Yuan, "An efficient road detection method in noisy urban environment," Proc. Intelligent Vehicles Symposium, pp.556–561, 2009.
[2] Y. He, H. Wang, and B. Zhang, "Color-based road detection in urban traffic scenes," IEEE Trans. Intelligent Transportation Systems, vol.5, no.4, pp.309–318, 2004.
[3] Q. Wu, W. Zhang, T. Chen, and B.V.K.V. Kumar, "Prior-based vanishing point estimation through global perspective structure matching," Proc. IEEE International Conference on Acoustics Speech and Signal Processing, pp.2110–2113, 2010.
[4] T. Suttorp and T. Bucher, "Robust vanishing point estimation for driver assistance," Proc. IEEE Intelligent Transportation Systems Conference, pp.1550–1555, 2006.
[5] 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.
[6] J.-P. Tardif, "Non-iterative approach for fast and accurate vanishing point detection," Proc. IEEE 12th International Conference on Computer Vision, pp.1250–1257, 2009.
[7] Y. Wang, L. Bai, and M. Fairhurst, "Robust road modeling and tracking using condensation," IEEE Trans. Intelligent Transportation Systems, vol.9, no.4, pp.570–579, 2008.
[8] J.M. Alvarez, T. Gevers, and A.M. Lopez, "3D scene priors for road detection," Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.57–64, 2010.
[9] C. Rasmussen, "Texture-based vanishing point voting for road shape estimation," Proc. British Machine Vision Conference, pp.7.1–7.10, 2009.
[10] C. Rasmussen, "Grouping dominant orientations for ill-structured road following," Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol.1, pp.470–477, 2004.
[11] H. Kong, J.-Y. Audibert, and J. Ponce, "Vanishing point detection for road detection," Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.96–103, 2009.
[12] Y. Alon, A. Ferencz, and A. Shashua, "Off-road path following using region classification and geometric projection constraints," Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol.1, pp.689–696, 2006.
[13] J. Canny, "A computational approach to edge detection," IEEE Trans. Pattern Anal. Mach. Intell., vol.PAMI-8, no.6, pp.679–698, 1986.
[14] A. Torralba and P. Sinha, "Statistical context priming for object detection," Proc. Eighth IEEE International Conference on Computer Vision, vol.1, pp.763–770, 2001.
[15] D. Hoiem, A.A. Efros, and M. Hebert, "Recovering surface layout from an Image," Int. J. Comput. Vis., vol.75, no.1, pp.151–172, 2007.
[16] J. Alvarez, T. Gevers, and A.M. Lopez, "Vision-based road detection using road models," Proc. 16th IEEE International Conference on Image Processing, pp.2073–2076, 2009.
[17] T.S. Lee, "Image representation using 2D Gabor wavelets," IEEE Trans. Pattern Anal. Mach. Intell., vol.18, no.10, pp.959–971, 1996.
[18] R.G. Keys, "Cubic convolution interpolation for digital image processing," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-29, no.6, pp.1153–1160, 1981.

Trung Hieu Bui He received his B.E. and M.E. degrees in automation engineering from Hanoi University of Technology, Vietnam, in 2007 and 2009, respectively. Since 2010, he has been with Kyushu Institute of Technology, where he is currently a third-year doctoral student in the Department of Information Science. His research interests include computer vision.

Eitaku Nobuyama He received his B.E., M.E., and Dr.E. degrees in mathematical engineering and instrumental physics from the University of Tokyo, Japan, in 1983, 1985, and 1988, respectively. In 1988, he joined the Department of Mathematical Engineering and Instrumental Physics at the University of Tokyo. Since 1991, he has been with Kyushu Institute of Technology, Japan, where he is currently a professor in the Department of Systems Design and Informatics. His research interests include control engineering and computer vision.

Takeshi Saitoh He received his B.E., M.E., and Dr.E. degrees in the Information and Computer Sciences from Toyohashi University of Technology in 1999, 2001, and 2004, respectively. In 2004, he joined the Information and Electronics at Tottori University. Since 2010, he has been with Kyushu Institute of Technology, where he is currently an associate professor in the Department of Systems Design and Informatics. His research interests include image processing, pattern recognition. He received the FIT young researcher's award in 2008. He is a member of IEEE and the Society of Instrument and Control Engineers.