Machine Vision and Applications (2010) 21:163–176
DOI 10.1007/s00138-008-0156-9

ORIGINAL PAPER

Automatic free parking space detection by using motion stereo-based 3D reconstruction

Jae Kyu Suhr · Ho Gi Jung · Kwanghyuk Bae · Jaihie Kim

Received: 19 October 2007 / Accepted: 4 June 2008 / Published online: 15 July 2008

Abstract This paper proposes a free parking space detection system by using motion stereo-based 3D reconstruction. An image sequence is acquired with a single rearview fisheye camera and the view behind the automobile is three-dimensionally reconstructed by using point correspondences. Metric information is recovered from the camera height ratio and free parking spaces are detected by estimating the positions of adjacent vehicles. Since adjacent vehicles are usually located near the epipole, their structures are seriously degraded. To solve this problem, we select point correspondences by using a de-rotation-based method and mosaic 3D structures by estimating a similarity transformation. Unlike in previous work, our system proposes an efficient way of locating free parking spaces in 3D point clouds. Odometry is not used because its accuracy depends largely on road conditions. In the experiments, the system was tested in 154 different parking situations and its success rate was 90% (139 successes in 154 cases). The detection accuracy was evaluated by using ground truth data that was acquired with a laser scanner.

J. K. Suhr · H. G. Jung · K. Bae · J. Kim (✉)
Biometrics Engineering Research Center,
School of Electrical and Electronic Engineering,
Yonsei University, 134 Shinchon-dong, Seodaemun-gu,
Seoul 120-749, Republic of Korea
e-mail: jhkim@yonsei.ac.kr

J. K. Suhr
e-mail: lfisbf@yonsei.ac.kr

K. Bae
e-mail: paero@yonsei.ac.kr

H. G. Jung
Global R&D H.Q, MANDO Corporation,
Yongin-Si 446-901, Republic of Korea
e-mail: hgjung@mando.com

Keywords Parking space detection · Motion stereo · Three-dimensional reconstruction · Three-dimensional mosaicing

1 Introduction

An automatic parking system provides convenience for drivers by automatically finding free parking spaces and steering automobiles toward them. Recently, there has been increased interest in automatic parking systems [3]. For instance, the 2003 Toyota Prius adopted the "Intelligent Parking Assist" feature as an option and about 80% of buyers have selected this option [20]. Due to customer interest and the Prius's success, many car manufacturers and component manufacturers are preparing to release self-parking products [25].

Automatic parking systems consist of three components: path planning including free parking space detection, an automatic steering and braking system used to implement the planned trajectory, and the HMI (human machine interface), which can be used to receive driver input and provide visual information of the ongoing parking process [15].

Free parking space detection has been implemented by using various methods: the ultrasonic sensor-based method [7,30], the laser scanner-based method [17,31], the short range radar network-based method [6,12], and the vision-based method [9,15,16,18,19,41,42]. Among these, the vision-based method is attractive to drivers because it visualizes parking procedures, which make drivers feel safer. The vision-based method can be categorized into four approaches: the parking space marking-based approach, the binocular stereo-based approach, the light stripe projection-based approach, and the motion stereo and odometry-based approach.

The first approach recognizes parking space markings. Xu et al. [42] developed color vision-based localization of

parking spaces. This method uses color segmentation based on neural networks, contour extraction based on the least square method, and inverse perspective transformation. Jung et al. [16] proposed the semi-automatic parking assist system which recognized marking lines by using the Hough transform in a bird's eye view edge image captured with a wide-angle camera. In this way, target spaces can be detected with a single image at a relatively low computational cost. Also, a general configuration of a rearview camera (a single fisheye camera) can be used. However, it cannot be used when parking space markings are not available. Also, performance can be degraded by poor visual conditions such as stains, shadows or occlusion.

The second approach recognizes adjacent vehicles by using a binocular stereo-based 3D reconstruction. Kaempchen et al. [19] developed the parking space estimation system which uses a feature-based stereo algorithm, a template matching on a depth map, and a 3D fitting to the planar surface model of the vehicle. This approach can easily recover metric information from the fixed length of the baseline and the camera extrinsic parameters need not be estimated every time. However, this requires extra costs and space for the equipment. Also, sub-pixel accuracy is required in case of short baseline, and point correspondences are difficult to find in case of long baseline.

Jung et al. [15] developed a method which combines the parking space marking-based approach and the binocular stereo-based approach. They used obstacle depth maps for establishing the search range and simple template matching for finding the exact location of free parking spaces. This method is robust to noise factors such as stains, trash and shadows when compared to the parking space marking-based method, but it can be only used when both obstacle depth and parking space markings are available.

The third approach recognizes adjacent vehicles by using a light plane projector and a single rearview camera. Jung et al. [18] developed a method which identified free parking spaces by analyzing the light stripe on objects to the rear of the vehicle produced by the light plane projector. This approach can be applied to dark underground parking lots and the algorithm for acquiring the 3D information is relatively simple. Also, a general configuration of a rearview camera can be used. However, this approach cannot be used during the day due to the presence of sunlight and the configuration of the camera and the light plane projector must be unchanged.

The fourth approach recognizes adjacent vehicles by using a motion stereo and odometry-based 3D reconstruction. Fintzel et al. [9] and Vestri et al. [41] proposed a system which provides a rendered image from a virtual viewpoint for better understanding of parking situations and procedures. This system obtains camera external parameters and metric information from odometry and reconstructs the 3D structure of the parking space by using point correspondences. This approach can easily reconstruct the Euclidean 3D structure by using odometry and a general configuration of rearview camera can be used. However, odometry information can be erroneous when road conditions are slippery due to rain or snow [21], and a free parking space detection method was not presented.

The proposed system is similar to the fourth approach. We three-dimensionally reconstruct the rearview structures by using a single rearview fisheye camera and find free parking spaces in the 3D point clouds. This system consists of six stages: image sequence acquisition, feature point tracking, 3D reconstruction, 3D structure mosaicing, metric recovery, and free parking space detection.

Compared to previous works [9,41], the proposed system makes three contributions. First, the degradation of the 3D structure near the epipole is solved by using de-rotation-based feature selection and 3D structure mosaicing. This is a serious problem when reconstructing 3D structures with an automobile rearview camera because the epipole is usually located on the image of an adjacent vehicle which must be precisely reconstructed. Although this problem was mentioned in [29,41], a solution was not presented. Second, an efficient method for detecting free parking spaces in 3D point clouds is proposed. For this task, the structure dimensions are reduced from 3D to 2D and the positions of adjacent vehicles are estimated. Third, odometry is not used because its accuracy largely depends on road conditions. The camera external parameters are estimated by using point correspondences and the metric information is recovered from the camera height ratio.

There has been some research into reconstructing 3D structures with similar configurations in the field of SLAM [26,27,29]. These studies used a single forward-looking wide angle camera for building maps and locating vehicles. Odometry was not used in these studies but the 3D structures were reconstructed only up to an unknown scale factor. The degradation near the epipole was mentioned but it was not considered.

In our experiments, the system was applied to 154 real parking situations. It succeeded in 139 cases and failed in 15 cases, producing a success rate of 90.3%. Detection accuracy was evaluated with 47 image sequences taken with a laser scanner data. This evaluation showed that the proposed system yielded acceptable accuracy levels.

The rest of this paper is organized as follows. In Sect. 2, we discuss the point correspondences and 3D reconstruction. In Sect. 3, we explain the problem of the epipole and offer a solution. In Sect. 4, we describe metric recovery and the free parking space detection. In Sect. 5, we present the experimental results including a comparison with the laser scanner data. Finally, in Sect. 6, we conclude the paper with a summary and some suggestions for future work.

2 Motion stereo-based 3D reconstruction

2.1 Point correspondences

Point correspondences in two different images have to be found in order to estimate the motion parameters and 3D structures. For this task, we considered three approaches. The first approach finds a small number of reliable point correspondences to estimate the fundamental matrix and matches many feature points by using the epipolar constraints. This is called guided matching [13], which requires the fisheye images to be undistorted and rectified but both are very time-consuming processes.

The second approach finds many point correspondences by using a naive algorithm, and rejects false matches by using an outlier rejection method designed for cameras on intelligent vehicles [33]. Even though this approach is fast and produces few mismatches, it is difficult to find point correspondences on automobile surfaces due to the lack of features.

The third approach detects feature points and tracks them through image sequences [22,34,35]. Since this method tracks the feature points between consecutive images, it can find many point correspondences on automobile surfaces. However, computational costs are high because the algorithm has to be applied to many images.

The first and second approaches require two images. The memory size for saving the images is small but it is difficult to select key frames without saving the whole sequence. Key frames determine the 3D reconstruction interval. The third approach also requires two images every moment if it is implemented in real-time. Since point correspondences for each frame are saved, it is convenient to select key frames by using the tracking results. By summarizing this comparison, the tracking approach was selected in our application.

For tracking, we chose the Lucas–Kanade method [22, 35] because it produces accurate results, offers affordable computational power [2,24], and there are some existing examples of real-time hardware implementations [5,8,23]. This method uses the least square solution of optical flows. If I and J are two consecutive images and $\mathbf{x}$ and Ω denote the feature position and the small spatial neighborhood of $\mathbf{x}$, respectively, then the goal is to find the optical flow vector,$\mathbf{v}$ which minimizes:

$$\min_{\mathbf{v}} \sum_{\mathbf{x}\in\Omega} \{I(\mathbf{x}) - J(\mathbf{x}+\mathbf{v})\}^2. \tag{1}$$

The solution of Eq. (1), $\mathbf{v}_{\text{opt}}$ is given by:

$$\mathbf{v}_{\text{opt}} = G^{-1}\mathbf{b}$$
$$G = \sum_{\mathbf{x}\in\Omega} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}, \quad \mathbf{b} = \sum_{\mathbf{x}\in\Omega} \begin{bmatrix} \delta I I_x \\ \delta I I_y \end{bmatrix}. \tag{2}$$

I_x and I_y are the image gradients in the horizontal and vertical directions, respectively, and δI is the image pixel difference. Since the matrix G is required to be non-singular, the image location where the minimum eigenvalue of G is larger than the threshold is selected as a feature point and tracked through the image sequence.

2.2 Three-dimensional reconstruction

Once the point correspondences are obtained, the structure of the parking space is three-dimensionally reconstructed by using the following three steps: key frame selection, motion parameter estimation, and triangulation. First of all, the key frames which determine the 3D reconstruction interval should be appropriately selected. If there is not enough camera motion between the two frames, the motion parameters is inaccurately estimated and in the opposite case, the number of point correspondences is decreased.

Some algorithms have been proposed to select key frames [28,29,37]. We used a simple but less general method which uses the average length of optical flow. This method works well because rotational motion is always induced by translational motion in our application. Since parking spaces are reconstructed at the driver's request, the latest frame is selected as the first key frame. The second key frame is selected when the average length of optical flow from the first key frame exceeds the threshold. The next key frame is selected in the same way. The threshold value was set to 50 pixels and this made the baseline length approximately 100–150 cm.

Once the key frames are selected, the fundamental matrix is estimated to extract the motion parameters. For this task, we used RANSAC followed by an M-Estimator. Torr et al. [36] found that fundamental matrix estimation performance could be improved by using this combination. We also performed experiments using the various methods in [1] by using automobile rearview fisheye images and the same combination was found to be the best.

The RANSAC is based on randomly selecting a set of points to compute the candidates of the fundamental matrix by using a linear method. This method calculates the number of inliers for each fundamental matrix and chooses the one which maximizes it. Once the fundamental matrix is determined, it is refined by using all the inliers. The M-Estimator reduces the effect of the outliers weighting the residual of each point correspondence. If $\mathbf{x}_i'$ and $\mathbf{x}_i$ are the coordinates of the point correspondences in two images and F is the fundamental matrix, then the M-Estimator is based on solving Eq. (3):

$$\min_{F} \sum_{i} w_i (\mathbf{x}_i'^{\mathrm{T}} F \mathbf{x}_i)^2 \tag{3}$$

w_i is a weight function and we used Huber's [14] function.

After estimating the fundamental matrix, we follow the method presented in [13]. The essential matrix is calculated by using the fundamental matrix and the camera intrinsic parameters matrix. The camera intrinsic parameters were pre-calibrated because they do not change in our application. The four combinations of the rotation matrix and the translation vector are extracted from the essential matrix. Since only the correct combination allows the 3D points to be located in front of both cameras, several randomly selected points are reconstructed to determine the correct combination. The projection matrices of the two cameras are produced by combining the camera intrinsic parameters matrices, the rotation matrix, and the translation vector. After that, the 3D points are calculated by using a linear triangulation method. If P and P' represent the projection matrices of the two cameras and $\mathbf{X}$ represents the 3D point of the point correspondence ($\mathbf{x}$ and $\mathbf{x}'$), they satisfy Eq. (4).

$$\begin{aligned} \mathbf{x} \times (P\mathbf{X}) &= \mathbf{0} \\ \mathbf{x}' \times \left(P'\mathbf{X}\right) &= \mathbf{0}. \end{aligned} \tag{4}$$

By combining the above two equations into the form $A\mathbf{X} = 0$, the 3D point ($\mathbf{X}$) is simply calculated by finding the unit singular vector corresponding to the smallest singular value of A. This is solved by using a SVD. The matrix A is expressed as:

$$A = \begin{bmatrix} x\mathbf{p}^{3\mathrm{T}} - \mathbf{p}^{1\mathrm{T}} \\ y\mathbf{p}^{3\mathrm{T}} - \mathbf{p}^{2\mathrm{T}} \\ x'\mathbf{p}'^{3\mathrm{T}} - \mathbf{p}'^{1\mathrm{T}} \\ y'\mathbf{p}'^{3\mathrm{T}} - \mathbf{p}'^{2\mathrm{T}} \end{bmatrix} \tag{5}$$

$\mathbf{p}^{i\mathrm{T}}$ and $\mathbf{p}'^{i\mathrm{T}}$ represent the ith rows of the projection matrices of the two cameras, and $[x, y]^{\mathrm{T}}$ and $[x', y']^{\mathrm{T}}$ represent the image coordinates of the point correspondences. For 3D reconstruction, we did not use a complex optimization algorithm such as a bundle adjustment [38] because its computational cost is too high for our application.

3 Feature selection and 3D structure mosaicing

3.1 Degradation of 3D structure near the epipole

When reconstructing 3D structures in our application, heavy degradation appears near the epipole. This is because triangulation has to be performed at a small angle in that area. With a small angle, the accuracy of the 3D points is degraded because of the relatively high portions of the point detection error and the image quantization error. This can be shown as a rank deficiency of the matrix A in Eq. (5). When the feature point nears the epipole, the rank of matrix A becomes closer to two. This causes unreliable estimation of the 3D points.

Even though this problem is very serious in 3D reconstruction as mentioned in [29,41], it has not been dealt with in previous works because of two reasons. First, the epipole is not located inside the image in many applications because of camera configurations. This happens when the 3D structures are reconstructed by using a stereo camera or a single moving camera whose translation in the optical axis is not dominant relative to the translations in the other axes [11,19]. Second, the epipole is located inside the image but it is not on the target objects. This happens when a single forward (or backward) looking camera moves along a road or corridor. In this case, the epipole is located inside the image but it is usually on objects far from the camera, so the region around the epipole is not interesting [26,27,29].

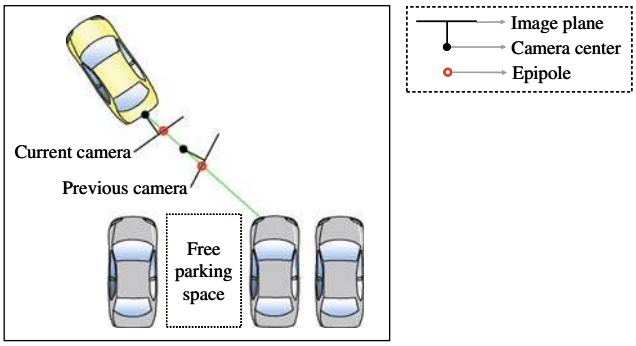

Fig. 1 Location of the epipole in a typical parking situation

In our application, the translation in the optical axis is quite dominant, so the epipole is always located inside the image. Also, the epipole is usually located on the image of an adjacent vehicle which is our target object used for locating free parking spaces. Figure 1 shows the epipole location in a typical parking situation. As shown in this figure, the epipole is usually located on the image of an adjacent vehicle due to the motion characteristics of the automobile rearview camera.

For this reason, the 3D structure of the adjacent vehicle is erroneously reconstructed in our application. Figure 2 shows the location of the epipole in the last frame of the image sequence and its reconstructed 3D structure. We depict the structure as seen from the top after removing the points near the ground plane. In this figure, the 3D points near the epipole on the adjacent vehicle appear quite erroneous, so the free parking space detection results will be degraded by those points.

3.2 De-rotation-based feature selection and 3D structure mosaicing

To solve the problem of the epipole and obtain a precise 3D rearview structure, we propose a two-step method. First the unreliable point correspondences are removed by using a de-rotation-based method, and then the removed part of the structure is substituted by mosaicing several 3D structures. In the first step, we eliminate the rotational effect from the optical flow. Since the optical flow length is proportional to the 3D point accuracy in a pure translation [39], we simply

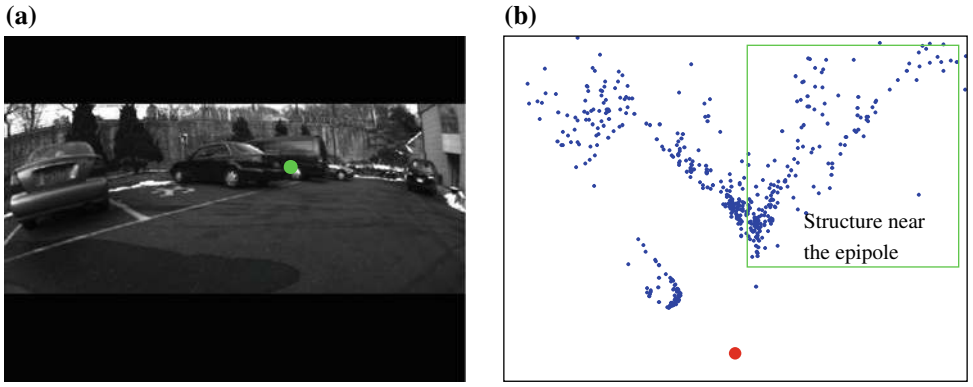

Fig. 2 **a** A typical location of the epipole. **b** Its reconstructed 3D structure as seen from the top after removing the points near the ground plane

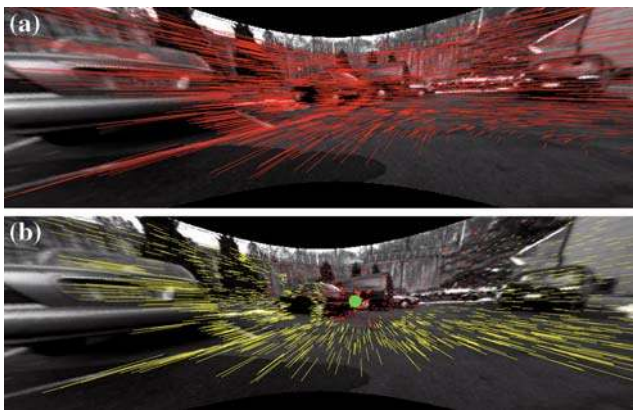

Fig. 3 **a** Undistorted optical flows. **b** De-rotated optical flows

throw away the point correspondences whose optical flow lengths are shorter than the threshold. This prevents the 3D structure from including erroneously reconstructed points. For eliminating the rotational effect, a conjugate rotation is used [13]. If $\mathbf{x}$ and $\mathbf{x}'$ are the images of a 3D point ($\mathbf{X}$) before and after the pure rotation:

$$\begin{aligned} \mathbf{x} &= K[I|0]\mathbf{X} \\ \mathbf{x}' &= K[R|0]\mathbf{X} = KRK^{-1}\mathbf{x} \end{aligned} \tag{6}$$

so that $\mathbf{x}' = H\mathbf{x}$ with $H = KRK^{-1}$. K, I, and R represent a 3×3 camera intrinsic parameters matrix, a 3×3 identity matrix, and a 3×3 rotation matrix, respectively.

Figure 3 describes the de-rotated-based feature selection procedure. The optical flows found in the fisheye images are undistorted as shown in Fig. 3a. After that, the undistorted optical flows are de-rotated by using a conjugate rotation as shown in Fig. 3b. All the optical flows in Fig. 3b point toward the epipole because the rotational effect is totally eliminated. In this case, the epipole is known as the focus of expansion. In Fig. 3b, the red lines indicate the unreliable optical flows classified by the de-rotation-based method. The unreliable optical flows include the features near the epipole and far from the camera. The threshold for the optical flow length was set to ten pixels.

In the second step, we reconstruct several 3D structures by using the reliable point correspondences and mosaic them into one structure by estimating the similarity transformation. This process substitutes the removed part of the rearview structure. The similarity transformation parameters consist of $R(3 \times 3$ rotation matrix), $\mathbf{t}(3 \times 1$ translation vector), and c (scaling) and we use the least-square fitting method [40] with the 3D point correspondences known from the tracking results. Since the reconstructed 3D points may be erroneous and include outliers, the RANSAC approach [10] is used for parameter estimation. The least-square fitting method can be explained as follows. We are given two sets of 3D point correspondences $\mathbf{X}_i$ and $\mathbf{Y}_i$; $i = 1, 2, \ldots, n$ in the 3D space. $\mathbf{X}_i$ and $\mathbf{Y}_i$ are considered as 3×1 column vectors, and n is equal to or larger than three. The relationship between $\mathbf{X}_i$ and $\mathbf{Y}_i$ can be described as:

$$\mathbf{Y}_i = cR\mathbf{X}_i + \mathbf{t}. \tag{7}$$

The mean squared error of two sets of points can be written as:

$$e^2(R, \mathbf{t}, c) = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{Y}_i - (cR\mathbf{X}_i + \mathbf{t})\|^2. \tag{8}$$

If A and B are the $3 \times n$ matrices of $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ and $\{\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n\}$, respectively, and $\mathrm{UDV}^{\mathrm{T}}$ is a SVD of AB^{T} ($\mathrm{UU}^{\mathrm{T}} = \mathrm{VV}^{\mathrm{T}} = \mathrm{I}$, $D = \mathrm{diag}(d_i)$, $d_1 \geq d_2 \geq \cdots \geq 0$), the transformation parameters which minimize the mean squared error can be calculated by:

$$R = \mathrm{UV}^{\mathrm{T}}, \mathbf{t} = \boldsymbol{\mu}_Y - cR\boldsymbol{\mu}_X, c = \frac{1}{\sigma_X^2} trace(D) \tag{9}$$

$\boldsymbol{\mu}_Y$, $\boldsymbol{\mu}_X$ and σ_X^2 can be defined as:

$$\boldsymbol{\mu}_Y = \frac{1}{n}\sum_{i=1}^{n} \mathbf{Y}_i, \boldsymbol{\mu}_X = \frac{1}{n}\sum_{i=1}^{n} \mathbf{X}_i, \sigma_X^2 = \frac{1}{n}\sum_{i=1}^{n} \left\|\mathbf{X}_i - \boldsymbol{\mu}_X\right\|^2. \tag{10}$$

Figure 4 shows the key frame images and the reconstructed 3D structures when using and without using the de-rotation-based feature selection. The 3D structures are shown as seen

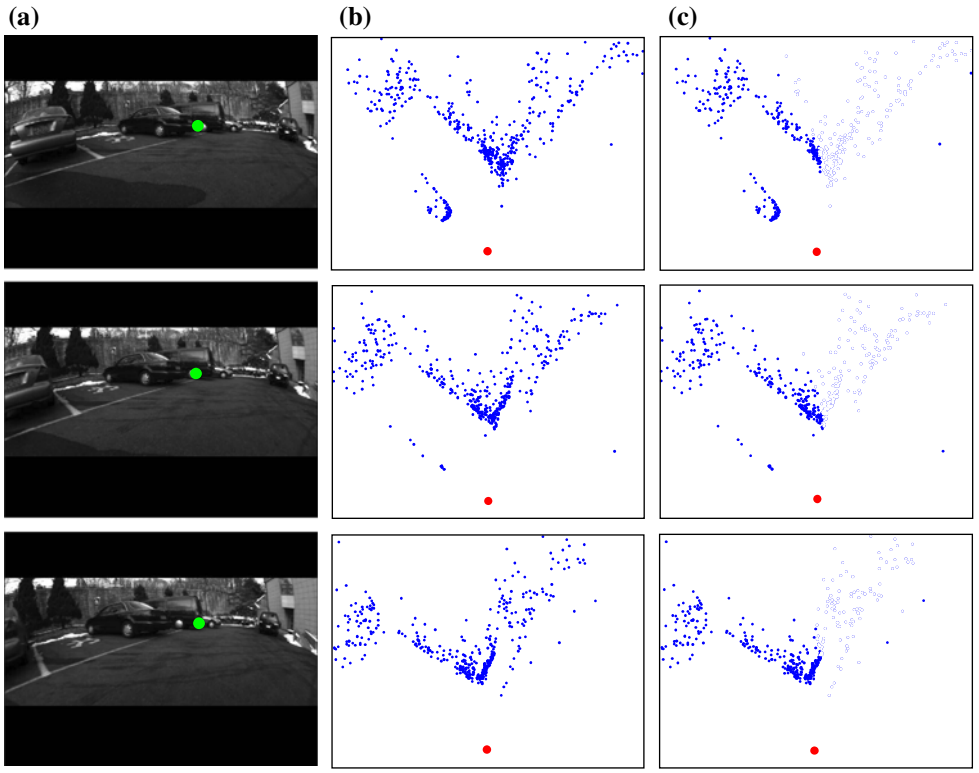

Fig. 4 **a** Key frame images. **b** 3D structures without using the de-rotation-based feature selection. **c** 3D structures when using the de-rotation-based feature selection

from the top after removing the points near the ground plane. Figure 4a shows the key frame images and their epipole locations. We can see that the epipoles are located on different positions of the adjacent vehicle. Figure 4b shows the reconstructed 3D structures of each key frame without using the de-rotation-based feature selection. The structures near the epipoles are badly reconstructed. However, the erroneously reconstructed part in one structure is correctly reconstructed in another structure. Figure 4c shows the reconstructed 3D structures of each key frame when using the de-rotation-based feature selection. Most of the erroneous 3D points in Fig. 4b are deleted.

Figure 5 shows the reconstructed structures when using and without using the proposed feature selection and 3D mosaicing methods. The red point indicates the camera center. By using the proposed two-step method, we obtained more precise structure near the epipole. In the experimental results, the advantages of this method are presented in detail by comparing the reconstructed structures with the laser scanner data.

4 Free parking space detection

4.1 Metric recovery

For locating free parking spaces in terms of centimeters, the metric information of the 3D structure has to be recovered. This is usually achieved by using a known baseline length or prior knowledge of the 3D structure. Since the camera height in the real world is known in our application, we estimate the camera height in the reconstructed world and use the ratio for metric recovery. The camera height in the real world is assumed as fixed in this paper. A height sensor can be used with camera height variations that may occur due to changing cargos or passengers.

To calculate the camera height ratio, the ground plane in the reconstructed world has to be estimated because the camera location is set to the origin. The estimation procedure consists of three steps: tilting angle compensation, density estimation-based ground plane detection, and 3D plane estimation-based ground plane refinement. The tilting angle is calculated and the 3D structure is rotated according to the calculated angle. This procedure forces the ground plane parallel to the XZ-plane. In our camera configuration (shown in Fig. 6), the tilting angle (θ) can be calculated by [4]:

$$\theta = \arctan\left(\frac{e_y - y_0}{f}\right) \tag{11}$$

e_y and y_0 are the y-axis coordinates of the epipole and the principal point, respectively. f is the focal length of the camera.

Since there is usually only one plane (the ground plane) parallel to the XZ-plane after compensating the tilting angle, the density of the y-axis coordinate of the 3D points has the maximum peak at the location of the ground plane. Figure 7 shows the density of the y-axis coordinate of the 3D points.

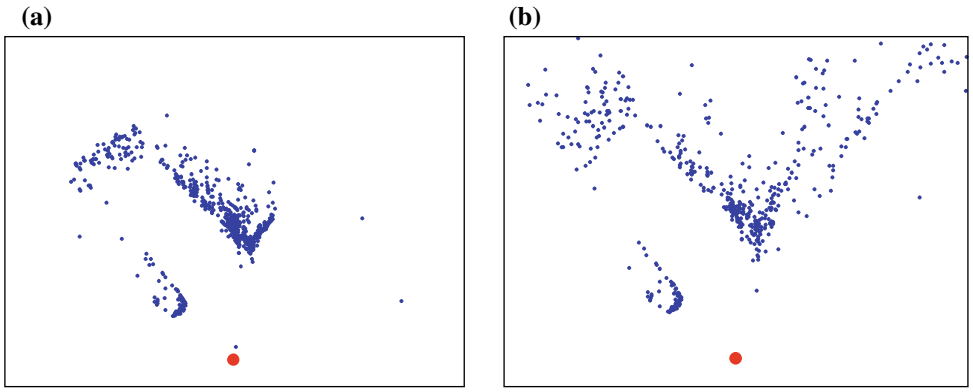

Fig. 5 Comparison of the 3D structures **a** when using the proposed two-step method **b** without using the proposed two-step method

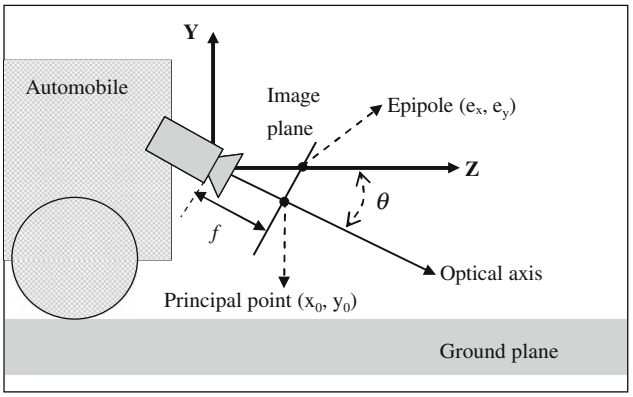

Fig. 6 Configuration of rearview camera

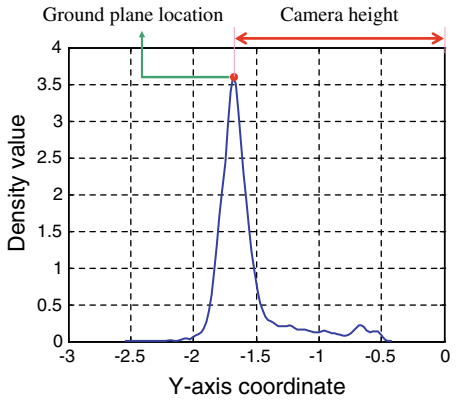

Fig. 7 Density of the y-axis coordinate of the 3D points

In this figure, the peak location is recognized as the location of the ground plane and the distance from the peak location to the origin is recognized as the camera height in the 3D structure.

After that, the location and the orientation of the ground plane are refined by 3D plane estimation. The 3D points near the initially detected ground plane are selected and the RANSAC approach is used for estimating the 3D plane. The camera height is refined by calculating the perpendicular distance between the camera center and the estimated 3D plane.

The 3D structure is scaled in centimeters by using the camera height ratio between the estimated camera height and the known camera height. After that, the 3D points far from the camera center are deleted and the remaining points are rotated according to the 3D plane orientation to make the ground plane parallel to the XZ-plane. Figure 8a, b shows the final result of the metric recovery in the camera-view and the top-view, respectively.

4.2 Free parking space detection

Once the Euclidean 3D structure is reconstructed, the free parking spaces are detected in the 3D point clouds. For this task, we estimate the positions of the adjacent vehicles and locate free parking spaces accordingly. Because position estimation in 3D space can be complicated and time-consuming, we reduce the dimensions of the structure from 3D to 2D. The 3D points, whose heights from the ground plane are between 30–160 cm, are selected and the height information is removed to reduce the dimensions. After that, we delete the isolated points by counting the number of neighbors. Figure 9a shows the dimension reduction result. Since all points in Fig. 9a do not belong to the outermost surface of the automobile, we select the outline points by using the relationship between the incoming angle and the distance from the camera center. This procedure is performed for better estimation of the position of the adjacent vehicle. The incoming angle is the angle between the horizontal axis and the line joining the camera center and a 2D point. Figure 9a is re-depicted in Fig. 9b by using the incoming angle and the distance from the camera center. Since the points on the same vertical line comes from the same incoming angle in Fig. 9b, the nearest point from the camera center among the points on the same vertical line is recognized as the outline point. Figure 9c shows the result of outline point selection.

If the automobile shape is assumed to be a rectangle as seen from the top, the position of the adjacent vehicle can be represented by a corner point and orientation. Therefore, we estimate the corner point and orientation of the adjacent

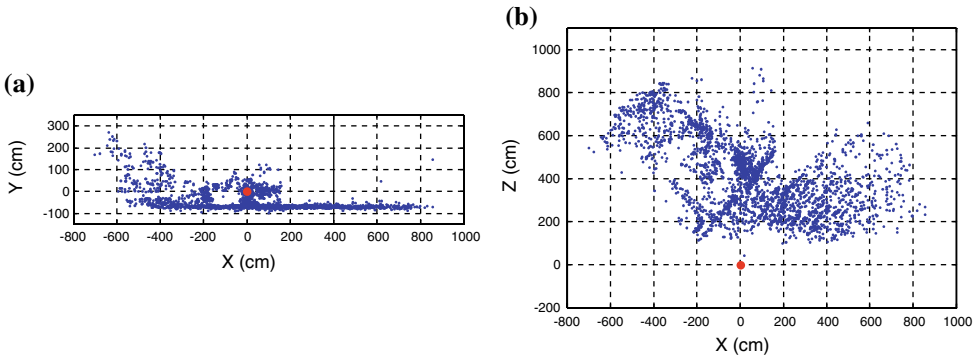

Fig. 8 Result of metric recovery. **a** Camera-view, **b** top-view. (*0, 0, 0*) indicates the camera center

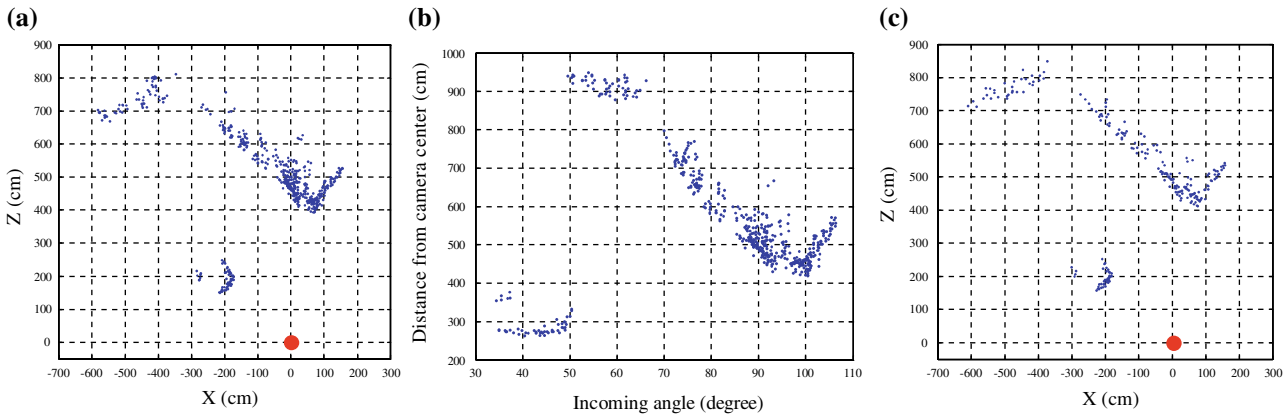

Fig. 9 **a** Dimension reduction result. **b** Re-depicted 2D points with the incoming angle and the distance from the camera center. **c** Result of outline point selection. (*0, 0*) indicates the camera center

vehicle and use these values to locate free parking spaces. Since the reconstructed structure is noisy and includes not only adjacent vehicles but also other obstacles, we use a projection-based method. This method rotates the 2D points and projects them onto the x and z-axes. It discovers the rotation angle which maximizes the sum of the maximum peak values of the two projection results. The rotation angle and the locations of the two maximum peak values are recognized as the orientation and the corner point, respectively. This method estimates the corner point and orientation at the same time, and it is robust to noisy data.

However, when using this method, we cannot know whether the estimated orientation is longitudinal or lateral. To determine this, it is assumed that a driver turns right when a free parking space is located on the left, and vice versa. This assumption helps us to determine the orientation by using the turning direction of automobile estimated from the rotation matrix.

After estimating the corner point and orientation, the points on the longitudinal side of the adjacent vehicle are selected and used for refining the orientation by using RANSAC-based line estimation. This procedure is needed because the lateral side of automobiles is usually curved so the longitudinal side gives more precise orientation information. The corner point is also refined according to the refined orientation.

To locate the most appropriate free parking spaces, other adjacent vehicles located opposite the estimated vehicle are also searched. The search range is set as Fig. 10a by using the estimated corner point and orientation. We set a circle with a radius of 150 cm and its center is located 300 cm away from the corner point in the lateral direction. If there are point clouds inside the search range, the other vehicle is considered to be found and the free parking space is located in the middle of two vehicles in the lateral direction. The corner points of two adjacent vehicles are projected in a longitudinal direction and the outer one is used to locate free parking spaces. This is described in Fig. 10b. In this figure, corner point 1 is selected because this is the outer one. If the other vehicle is not found, the free parking space is located beside the detected adjacent vehicle with a 50 cm interval in the lateral direction. Figure 11 shows the final result of the detection process. The width and length of the free parking space were set as 180 and 480 cm, respectively, since this is the size of the vehicle used for the experiment.

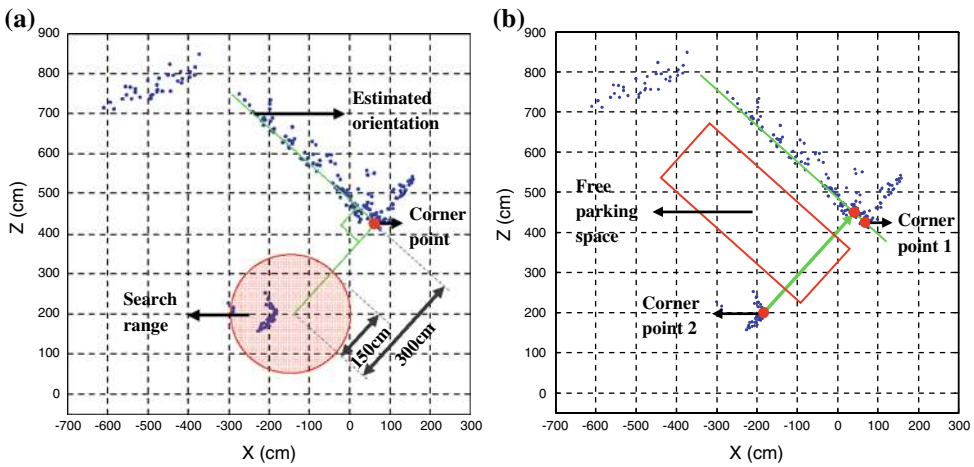

Fig. 10 **a** Search range of other adjacent vehicles. **b** Free parking space localization

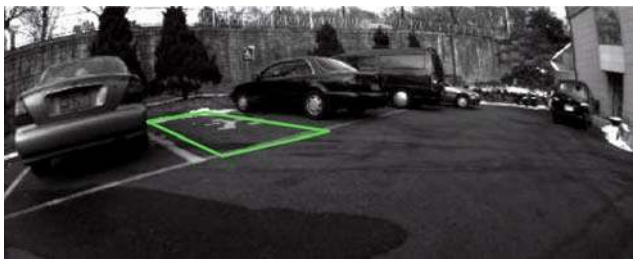

Fig. 11 Detected free parking space depicted on the last frame of the image sequence

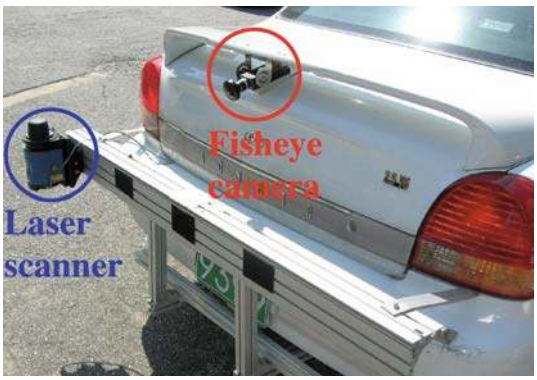

Fig. 12 Fisheye camera and laser scanner mounted on the automobile

5 Experimental results

The proposed system was tested in 154 different parking situations. From the database, 53 sequences were taken with the laser scanner data and 101 sequences were taken without it. The image sequences were acquired with a fisheye camera. Its horizontal and vertical fields of view were about 154° and 115°, respectively. The image resolution and frame rate were 1,024 × 768 pixels and 15 fps, respectively. The processing time of the overall procedure (except the feature tracking process) was 5.1 s on average when running it in MATLAB using a 2.4 GHz Intel Core2 Quad CPU. We analyzed the results in terms of success rate and detection accuracy. For success rate, a manual check was performed to determine whether the detected space was located inside the free space. For detection accuracy, the errors of the estimated corner point and orientation of the adjacent vehicle were measured by using laser scanner data. The experimental results consist of three parts. First, we compare the reconstructed structures when using and without using the proposed feature selection and 3D mosaicing methods. Second, the successes and failures of the system are discussed. Third, the accuracies of the estimated corner point and orientation are presented.

5.1 Comparison of the reconstructed structures

In this experiment, we reconstructed the 3D rearview structures when using and without using the proposed feature selection and 3D mosaicing methods and compared them to the laser scanner data. The laser scanner was the SICK LD-OEM1000 [32]. Its angular resolution and depth resolution are 0.125° and 3.9 mm, respectively and the systematic error is ±25 mm. Figure 12 shows the fisheye camera and the laser scanner mounted on the automobile. These two sensors were pre-calibrated.

Two comparison results are shown in Fig. 13. The reconstructed structures are depicted as seen from the top after removing the points near the ground plane. Figure 13a shows the last frames of two image sequences and the points on the vehicle indicate the locations of the epipoles. Figure 13b,c show the reconstructed rearview structures when using and without using the proposed method, respectively and the blue and red points indicate the reconstructed points and the laser scanner data, respectively.

By using this comparison, we can observe three advantages of the proposed feature selection and 3D mosaicing methods. First, it reduces the number of erroneously

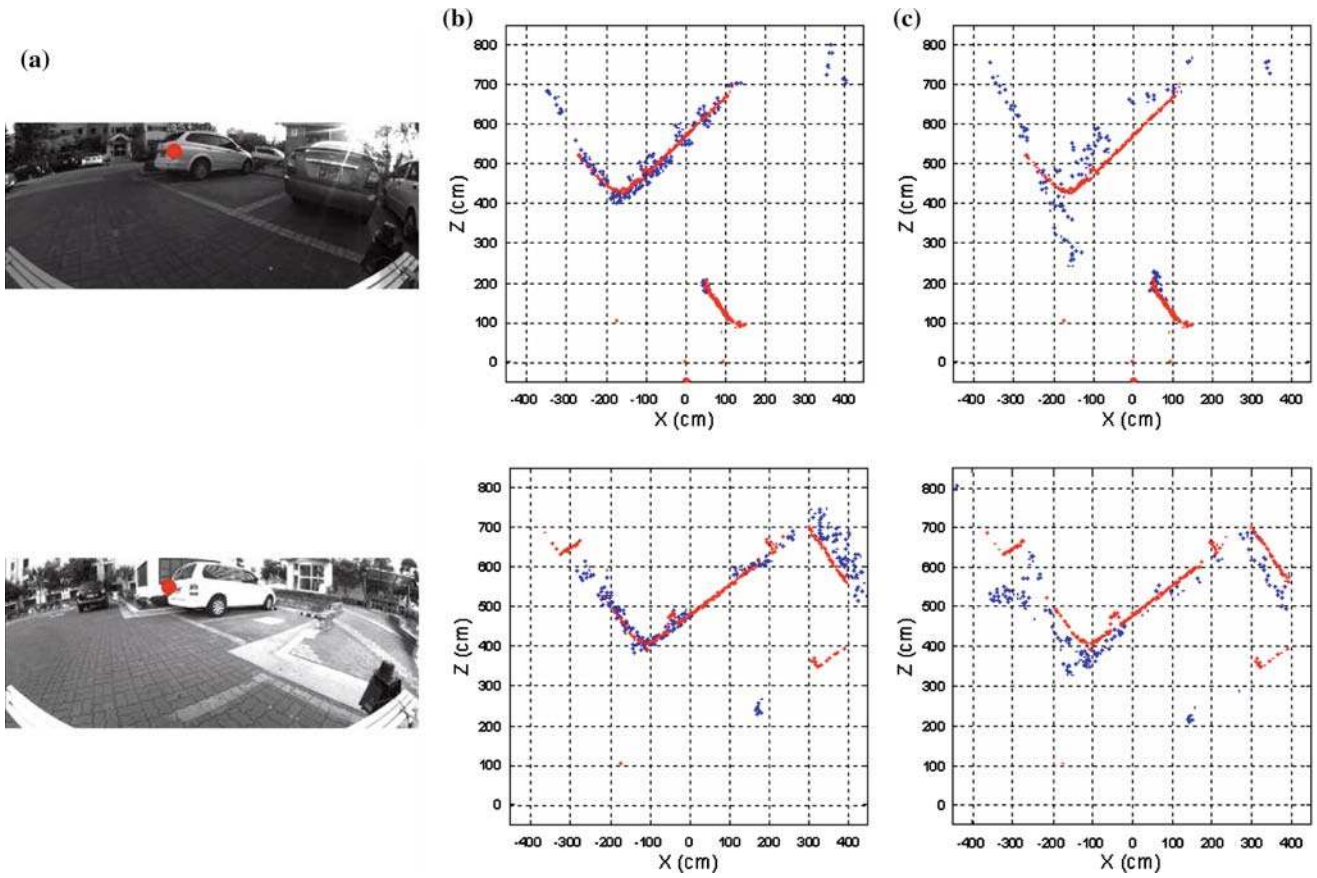

Fig. 13 **a** The last frames of two image sequences. **b** Reconstructed structures when using the proposed method. **c** Reconstructed structures without using the proposed method

reconstructed points. The structures in Fig. 13c shows more erroneous points outside the ground truth data than those in Fig. 13b because the proposed method removes the point correspondences near the epipole and far from the camera center. Second, it increases the amount of information about adjacent vehicles. The structures in Fig. 13b are more detailed than those in Fig. 13c because the density of the points on the adjacent vehicles is increased by mosaicing several 3D structures. Third, it enhances metric recovery results. In Fig. 13c, the scales of the reconstructed structures differ from the ground truth, since the proposed method produces more points on the ground plane, so it makes the ground plane estimation more accurate.

5.2 Free parking space detection results

The proposed system was applied to 154 real image sequences taken in the various situations. The ground planes were covered with asphalt, soil, snow, standing water, parking markers, etc. The automobiles varied in color from dark to bright and they included sedans, SUVs, trucks, vans, buses, etc. The environment included various types of buildings, vehicles, trees, etc. Figure 14 shows six successful examples. In this figure, the detected parking spaces are depicted on the last frames of the image sequences and corresponding rearview structures. To decide whether the system succeeded, we displayed the detected free parking space on the last frame of the image sequence. If it was located inside the free space between two adjacent vehicles, the result was considered to be a success. In this way, the system succeeded in 139 situations and failed in 15 situations, so the success rate was 90.3%.

Figure 15 shows four types of failures. In Fig. 15a, the sun was strongly reflected on the surface of the adjacent vehicle and the ground plane, so feature point tracking failed. In Fig. 15b, the adjacent vehicle was very dark and it was located in a shadowy region, so few feature points were detected and tracked on the automobile surface. In Fig. 15c, the free parking space was very far from the camera, so the structure of the white car was more precisely reconstructed than that of the silver van. This caused false detection. In Fig. 15d, part of the ground plane (darker region) on the parking space was repaved with asphalt, so the ground plane was not flat. This made the ground plane estimation erroneous. Out of fifteen failures, three could be depicted by Fig. 15a, nine could be depicted by Fig. 15b, two could be depicted by Fig. 15c, and one could be depicted by Fig. 15d.

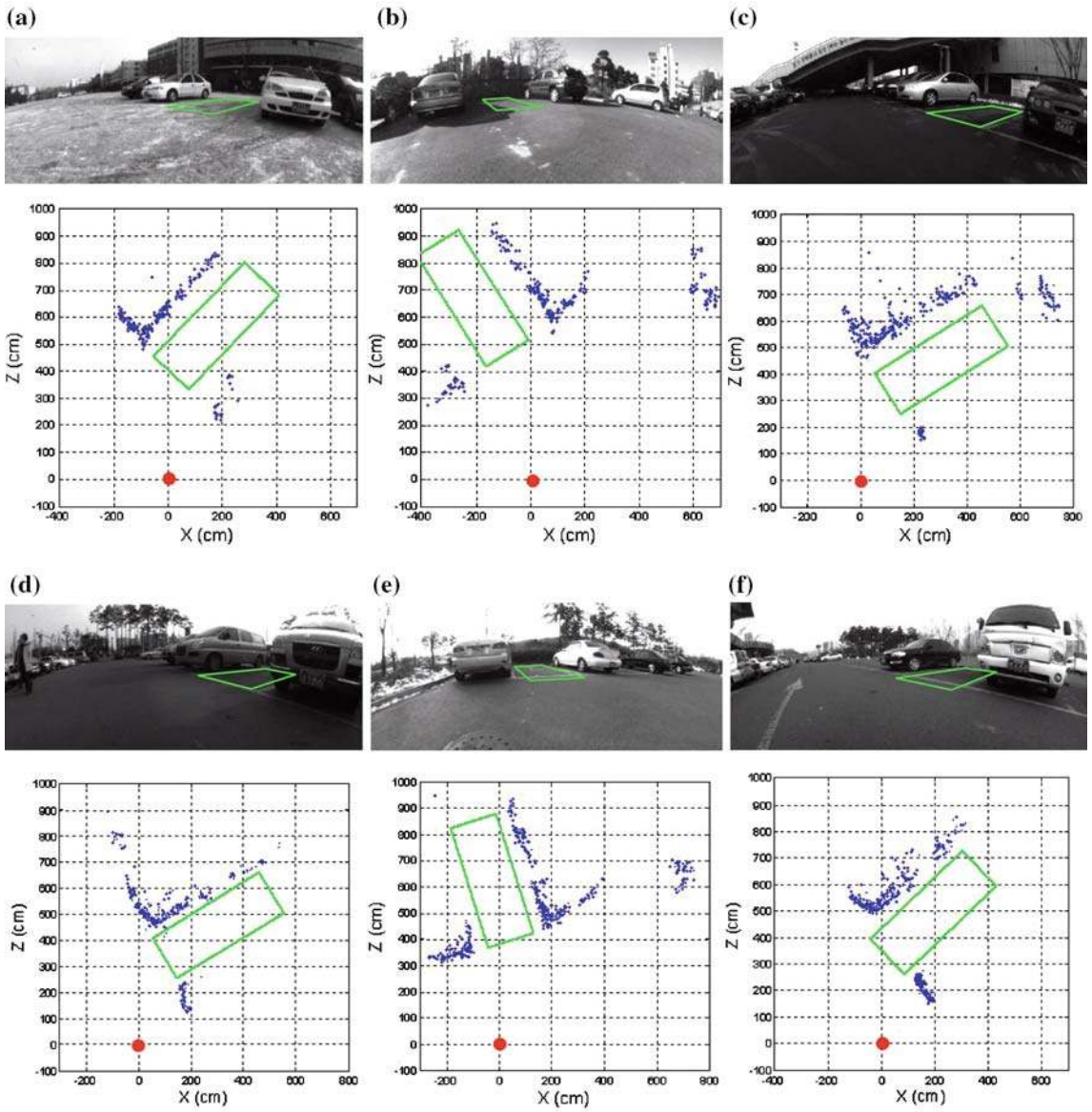

Fig. 14 Six successful detections. **a–f** The free parking spaces on the last frames of the image sequences and corresponding rearview structures. (*0, 0*) indicates the camera center

5.3 Accuracy of adjacent vehicle detection

Since the free parking space detection result depends on the estimation of the corner point and orientation, we calculated the errors of these two values for accuracy evaluation. The ground truth of the corner point and orientation were manually obtained by using laser scanner data. The error of the corner point is the Euclidean distance from the estimated point to the measured point and the error of the orientation is the absolute difference between the estimated angle and the measured angle. For this evaluation, 47 image sequences and the corresponding laser scanner data were used because 6 image pairs among 53 failed to detect free parking spaces due to the reasons mentioned in Sect. 5.2. The corner point and the orientation of the adjacent vehicle were estimated ten times for each image sequence. This is because the reconstructed structure can differ slightly every time due to the parameter estimation results.

In Fig. 16, the corner point error and the orientation error are depicted as histograms. The average and maximum errors of the corner point were 14.9 and 42.7 cm, respectively. The distance between the corner point and the camera center was between 281.4 and 529.2 cm. Since the lateral distance between two adjacent vehicles is approximately between 280 and 300 cm in a usual parking situation, there are about 50 cm extra room on each side of the vehicle. This means that even

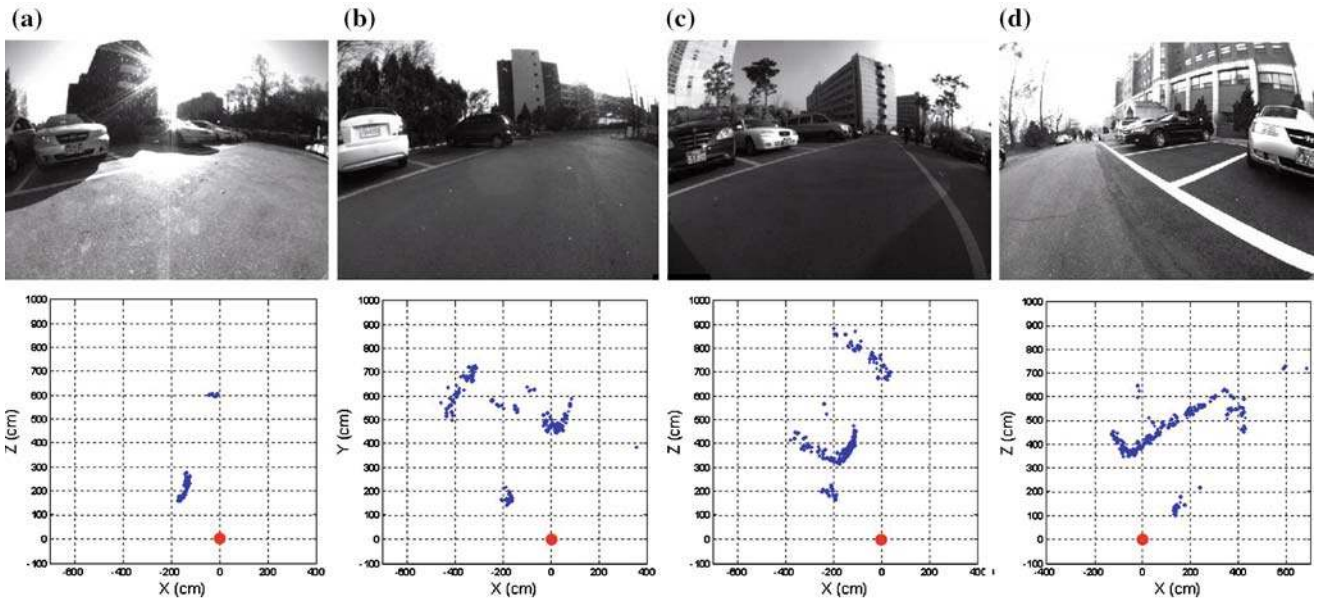

Fig. 15 Four types of failures. **a** Reflected sunlight. **b** Dark vehicle under a shadowy region. **c** Far parking space. **d** Uneven ground plane. The upper and the lower rows show the last frames of the image sequences and corresponding rearview structures, respectively. (*0, 0*) indicates the camera center

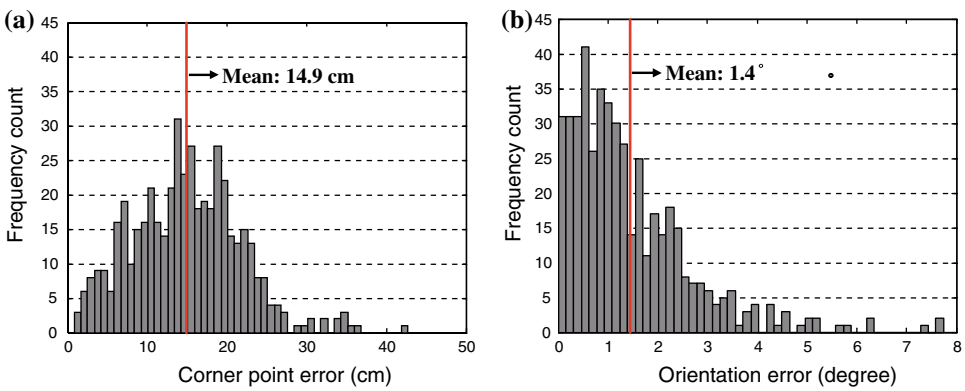

Fig. 16 Accuracy evaluation results. **a** Corner point error. **b** Orientation error

the maximum error of the corner point is acceptable for the free parking space localization. The average and maximum errors of the orientation were 1.4° and 7.7°, respectively. The average error of the orientation was acceptable but the maximum error of the orientation was somewhat large. This is because the side surfaces of automobiles sometimes show few corresponding points due to featurelessness and this makes the orientation estimation difficult. This evaluation shows that the proposed system produces acceptable results for detecting free parking spaces. For the worse cases, we are planning to refine the detection results when automobiles move backward into parking spaces.

6 Conclusions

This paper proposed a free parking space detection system. This system acquires an image sequence with a rearview camera and reconstructs the 3D structure by using point correspondences. The metric information is recovered from the camera height ratio and free parking spaces are detected by estimating the positions of adjacent vehicles. Compared to previous work, this paper makes three contributions. First, we solved the serious degradation of 3D structures near the epipole. Second, we presented an efficient method for detecting free parking spaces in 3D point clouds. Third, our system did not use odometry due to its unreliability. In the experiments, the proposed system showed a 90% success rate (139 successes in 154 cases) and the accuracy evaluation showed that the proposed system produced acceptable results. In our opinion, the reason that previous works only visualized parking situations without detecting the exact target space is mostly because of the 3D structure degradation near the epipole and the uncertainty of odometry. Since those problems have been solved in this paper, we expect that our work may

increase the possibility for the practical realization of motion stereo-based automatic free parking space detection systems. In the future, we plan to improve the feature selection method by assigning confidence values to the feature points that were simply rejected in the proposed method. For more accurate parking space detection, we will update and refine the rear-view structures while automobiles move backward into initially detected free parking spaces and use a height sensor to cope with camera height variations.

Acknowledgments This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center (BERC) at Yonsei University. R112002105070020 (2008).

References

1. Armangue, X., Salvi, J.: Overall view regarding fundamental matrix estimation. Image Vis. Comput. **21**(2), 205–220 (2003)
2. Barron, J., Fleet, D., Beauchemin, S.: Performance of optical flow techniques. Int. J. Comput. Vis. **12**(1), 43–77 (1994)
3. Bishop, R.: Intelligent Vehicle Technology and Trends. Artech House Publishers, Norwood (685 Canton Street, Norwood, MA 02062) (2005)
4. Burger, W., Bhanu, B.: Estimating 3-D egomotion from perspective image sequences. IEEE Trans. Pattern Anal. Mach. Intell. **12**(11), 1040–1058 (1990)
5. Correia, M.V., Campilho, A.C.: Real-time implementation of an optical flow algorithm. In: Proceedings of the 16th International Conference on Pattern Recognition, vol. 4, pp. 247–250 (2002)
6. Deacon, C.: New Mercedes CL Class: Parking Assistance Technology. http://www.worldcarfans.com (2006)
7. Degerman, P., Pohl, J., Sethson, M.: Hough transform for parking space estimation using long range ultrasonic sensors. SAE Paper. Document Number: 2006-01-0810
8. Díaz, J., Ros, E., Ortigosa, E.M., Mota, S.: FPGA-based real-time optical flow system. IEEE Trans. Circuits Syst. Video Technol. **16**(2), 274–279 (2006)
9. Fintzel, K., Bendahan, R., Vestri, C., Bougnoux, S.: 3D parking assistant system. In: Proceedings of IEEE Intelligent Vehicle Symposium, pp. 881–886 (2004)
10. Fischler, M., Bolles, R.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)
11. Fitzgibbon, A., Cross, G., Zisserman, A.: Automatic 3D model construction for turn-table sequences. In: Proceedings of European Workshop on 3D Structure from Multiple Images of Large-Scale Environments, pp. 155–170 (1998)
12. Gorner, S., Rohling, H.: Parking lot detection with 24 GHz radar sensor. In: The 3rd International Workshop on Intelligent Transportation (2006)
13. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision, 2nd edn., pp. 204–205, 257–260, 290–292. Cambridge University Press, London (2003)
14. Huber, P.: Robust Statistics. Wiley, New York (1981)
15. Jung, H.G., Kim, D.S., Yoon, P.J., Kim, J.: 3D vision system for the recognition of free parking site location. Int. J. Automot. Technol. **7**(3), 351–357 (2006)
16. Jung, H.G., Kim, D.S., Yoon, P.J., Kim, J.: Parking slot markings recognition for automatic parking assist system. In: Proceedings of IEEE Intelligent Vehicle Symposium, pp. 106–113 (2006)
17. Jung, H.G., Cho, Y.H., Yoon, P.J., Kim J.: Integrated side/rear safety system. In: The 11th European Automotive Congress (2007)
18. Jung, H.G., Kim, D.S., Yoon, P.J., Kim, J.: Light stripe projection based parking space detection for intelligent parking assist system. In: Proceedings of IEEE Intelligent Vehicle Symposium (2007)
19. Kaempchen, N., Franke, U., Ott, R.: Stereo vision based estimation of parking lots using 3d vehicle models. In: Proceedings of IEEE Intelligent Vehicle Symposium, pp. 459–464 (2002)
20. Kageyama, Y.: Look, no hand! New Toyota parks itself. http://www.cnn.com 14 January (2004)
21. Kelly, A.: Linearized error propagation in odometry. Int. J. Robot. Res. **23**(2), 179–218 (2004)
22. 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, pp. 674–679 (1981)
23. Maya-Rueda, S., Arias-Estrada, M.: FPGA processor for real-time optical flow computation. Lect. Notes Comput. Sci. **2778**, 1103–1016 (2003)
24. McCane, B., Novins, K., Crannitch, D., Galvin, B.: On benchmarking optical flow. Comput. Vis. Image Underst. **84**(1), 126–143 (2001)
25. Moran, T.: Self-parking technology hits the market. Automot. News **8**(6227), 22–22 (2006)
26. Mouragnon, E., Dekeyser, F., Sayd, P., Lhuillier, M., Dhome, M.: Real time localization and 3D reconstruction. In: Proceedings of Computer Vision and Pattern Recognition, vol. 1, pp. 17–22 (2006)
27. Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., Sayd, P.: Monocular vision based SLAM for mobile robots. In: Proceedings of the 18th International Conference on Pattern Recognition, vol. 3, pp. 20–24 (2006)
28. Nister, D.: Frame decimation for structure and motion. Lect. Notes Comput. Sci. **2018**, 17–34 (2001)
29. Royer, E., Lhuillier, M., Dhome, M., Lavest, J.: Monocular vision for mobile robot localization and autonomous navigation. Int. J. Comput. Vis. **74**(3), 237–260 (2007)
30. Satonaka, H., Okuda, M., Hayasaka, S., Endo, T., Tanaka, Y., Yoshida, T.: Development of parking space detection using an ultrasonic sensor. In: The 13th World Congress on Intelligent Transportation Systems and Services (2006)
31. Schanz, A., Spieker, A., Kuhnert, D.: Autonomous parking in subterranean garages—a look at the position estimation. In: Proceedings of 2003 IEEE Intelligent Vehicle Symposium, pp. 253–258 (2003)
32. SICK, LD-OEM1000/Laser Measurement Sensors. http://www.sick.com (2007)
33. Suhr, J.K., Jung, H.G., Bae, K., Kim, J.: Outlier rejection for cameras on intelligent vehicles. Pattern Recognit. Lett. **29**(6), 828–840 (2008)
34. Tissainayagam, P., Suter, D.: Assessing the performance of corner detectors for point feature tracking applications. Image Vis. Comput. **22**(8), 663–679 (2004)
35. Tomasi, C., Shi, J.: Good features to track. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600 (1994)
36. Torr, P., Murray, D.: The development and comparison of robust methods for estimating the fundamental matrix. Int. J. Comput. Vis. **24**(3), 271–300 (1997)
37. Torr, P., Fitzgibbon, A., Zisserman, A.: The problem of degeneracy in structure and motion. recovery from uncalibrated image sequences. Int. J. Comput. Vis. **32**(1), 27–44 (1999)
38. Triggs, B., McLauchlan, P., Hartley, R., Fitzgibbon, A.: Bundle adjustment—a modern synthesis. Lect. Notes Comput. Sci. **1883**, 298–372 (2000)

39. Trucco, E., Verri, A.: Introductory techniques for 3-D computer vision. Prentice Hall, Englewood Cliffs (1998)
40. Umeyama, S.: Least-square estimation of transformation parameters between two point patterns. IEEE Trans. Pattern Anal. Mach. Intell. **13**(4), 376–380 (1991)
41. Vestri, C., Bougnoux, S., Bendahan, R., Fintzel, K., Wybo, S., Abad, F., Kakinami T.: Evaluation of a vision-based parking assistance system. In: Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems, pp. 56–60 (2005)
42. Xu, J., Chen G., Xie, M.: Vision-guided automatic parking for smart car. In: Proceedings of IEEE Intelligent Vehicle Symposium, pp. 725–730 (2000)