UNSYS digital

Journal of Unmanned System Technology

Characterizing Indoor Vanishing Points by Local Dominant Orientation Signature from Omnidirectional Vision

Jun Yu Yang, Feng Li Lian, and Jiun Jau Lai

Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan

Abstract—For a mobile robot, it is necessary to have the ability to move accurately and autonomously in a known and/or unknown environment. The vanishing points existing in indoor environment are a key factor for the success of robot localization and navigation. To successfully detect these vanishing points, this paper proposes a novel approach of utilizing the distribution of orientation information from omnidirectional vision. The local orientation data are first computed by proper edge detection and then characterized by the local dominant orientation signature descriptors. These descriptors are further filtered based on the strength of its own orientation as well as its neighboring orientations. The final orientation field is constructed by using the interpolation of radial basis functions. The proposed approach of vanishing point detection has been extensively tested in various indoor environments such as narrow pathway library and multi-path corridor. Experimental results demonstrate good performance compared with existing approach.

Keywords—Omnidirectional camera, panoramic image, visual localization, vanishing point detection.

I. Introduction

The advances in robotic technology have implied that the robots are playing an important role in real-life scenarios, assisting people in their daily activities. For example, some office robots are commanded to fetch and deliver objects in an office environment [1]. To accomplish these tasks, the robots have to realize their self-position and destination. Besides, in order to reach their destination safely and efficiently, the robots must have the capability to navigate themselves among stationary walls or doors as well as other moving objects in a typical indoor environment. Furthermore, the ability of localization and navigation is also an important function for surveillance and security robots and home-care robots.

Localization refers to estimating the robot position in a known or unknown environment [2]. With the localization ability, a robot can realize its location and orientation in the space when moving around. For completing the task from one location to another location, the robot should be able to navigate itself based on available information in a real-time manner [3]. That is, the robot needs to compute its heading angle in terms of localized environmental information. Therefore, the understanding of vanishing points in the environment becomes an important task for robot localization and navigation [4].

Generally speaking, dense tracking and mapping (DTAM) and structure from motion (SfM) are two types of generic methods to understand the nearby environment of the host robot. The DTAM methods are for real-time camera tracking and scene reconstruction and do not rely on feature extraction from every image pixel. The key task of the DTAM methods is to precisely track the 6-degrees-of-freedom motion of the camera by the entire dense model. These methods then estimate the detailed texture of depth maps at selected keyframes [5]. Typical applications of the DTAM approaches include: augmented reality [6], SLAM [7], 3D Scanning and Printing [8].

On the other hand, the structure from motion (SfM) is a technique to simultaneously estimate the location of scene points (structure) and camera pose (motion) from a lot of images with only a given sparse set of correspondences between image features. A typical pipeline of SfM performs incrementally in several steps, including detecting image features, evaluating feature correspondence, estimating camera pose, localizing 3D points, and optimizing by bundle adjustment [9]. Typical applications of the DTAM approaches include: Image-based 3D modeling [10], hand-eye calibration [11], augmented reality [12], remote sensing [13].

A vanishing point can be defined as one point in an image where multiple parallel lines converge [14]. On the other hand, the vanishing point can also refer to a scene point at an infinite distance to the camera [15]. Understanding the vanishing point is very important for 3D interpretation, which is helpful for acquiring meaningful information about the real scene. The usage of the vanishing point is a technique, widely applied for indoor vision-based navigation. A lot of parallel lines exist in the indoor environment, such as the edges of floor boundary, windows, and railings. By utilizing the vanishing points identified from these lines, the directions which the robot can move into can be estimated [16].

The localization and navigation tasks of an autonomous robot conventionally depend upon range sensors, such as laser range finder, to provide distance information about surroundings. However, the range finders extract the range data by scanning only one single line along the surrounding objects or surfaces. Compared with a range sensor, a camera can

Corresponding author: Feng Li Lian (e-mail: fengli@ntu.edu.tw).
This paper was submitted on October 29, 2014; revised on July 9, 2015; and accepted on September 17, 2015.

provide more plentiful color information which is helpful to discriminate different objects in the environment [17].

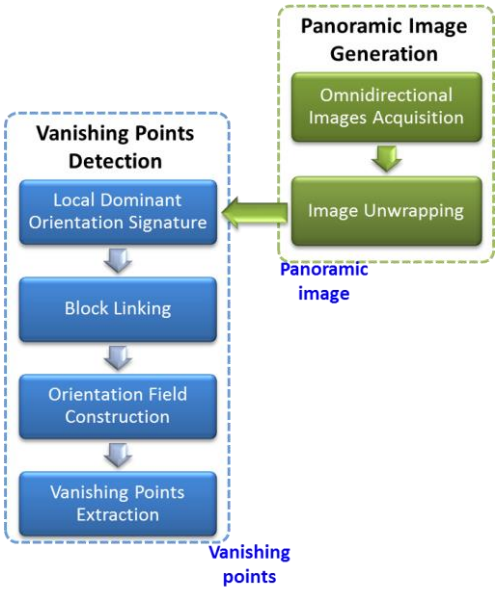


Figure 1 The flowchart of the vanishing point detection by LDOS descriptors of the image

Although many systems are developed based on monocular cameras, they are inappropriate for the robot with high speed motion due to the limited field of view of these cameras. To overcome this problem, omnidirectional cameras with full directional view of the surroundings are helpful tools for obtaining global information [18]. An omnidirectional camera can provide a 360° field of view along the horizontal plane. If an object is at a certain direction of omnidirectional image, it surely exists in the same direction of the real world. This means that the directional information of an omnidirectional image is reliable and ready to use. However, the range or distance information is undependable before proper characterization. Hence, the vanishing point is more difficult to detect than that in monocular images. Because these parallel lines in real world appear as radial straight lines in the captured images, the vanishing point can be obtained by analyzing these lines.

For example in [19], Manhattan world assumption is applied to a monocular image, and a Bayesian model is utilized to infer the orientation of the viewer which means the radial straight lines in a monocular image. By doing so, the vanishing point can be observed easily in such a normal image. Similarly, Martins et al. proposed a 3D orientation estimation method for urban environments by introducing Manhattan world assumption to a monocular image [20]. However, in omnidirectional images, these real-world parallel lines become different types of curves, which increase the difficulty of detecting vanishing points. Thus, the following two challenging tasks are required for correctly detecting vanishing points: One is to extract the curves from omnidirectional images which are parallel in the real world; and the other is to analyze the curves which are usefully and can be used to detect vanishing points.

To overcome these challenges, this paper proposes a novel systematic framework of using the local dominant orientation signature (LDOS) descriptors as the orientation feature for detecting vanishing points. These LDOS descriptors are proposed in [14] for monocular images. Here, the structural descriptors are applied in a panoramic image obtained from an omnidirectional camera. Next, the appropriateness and correctness of the local descriptor information is augmented and interconnected by the proposed edge-feature filter and chain-size filter. Finally, the orientation field is constructed by using the radial basis function and the vanishing points can be successfully extracted. The overall flowchart is illustrated in **Figure 1**.

The rest of the paper is organized as follows. Section II surveys related work. Section III briefly describes the definition and computation of the LDOS descriptors. Section IV presents the detailed procedure of the vanishing point detection. Section V analyzes the results of experimental tests and demonstrates the benefits of the proposed detection approach. Finally, Section VI summarizes this paper and discusses future work.

II. Related Work

Numerous vanishing point detection methods based on omnidirectional camera have been proposed. The existing approaches can be roughly categorized into two groups in terms of geometry and energy properties.

First, many vanishing point detection methods use the basis of unit sphere. In the beginning, the representation of sphere has been utilized in [21] for normal monocular images. In recent years, the spherical representation is used to deal with various kinds of central cameras such as omnidirectional camera by mapping the original image to the equivalent sphere. If the camera has more than one projection center, the sphere equivalence can help to produce an acceptable and reliable geometry approximation in numbers of conditions [22]. A projection model for sphere representations commonly used is proposed in [23]. Antone and Teller introduced a method combining the technique of unit sphere and the Hough transform among multiple cameras [24]. This method can be applied to single rectangular images as well, but using in an omnidirectional image can increase the accuracy of vanishing point detection. Therefore, more and more vanishing points extraction systems based on omnidirectional images are proposed. Bosse et al. designed a system based on omnidirectional vision for vanishing point detection and structure-from-motion [25]. Potential vanishing points are detected by finding clusters of line intersections. Then, the random sample consensus (RANSAC) based method is used to seed initial clusters for an EM-based (expectation-maximization) refinement.

In [26], and [27], the authors search for the vanishing points fast with the orthogonality characteristic because their systems are applied in urban environments. The geometric structure of the city makes the vanishing points orthogonal to each other.

Several methods such as [21] and [28] use the Hough transform to extract the line feature on the unit sphere and the vanishing point detection can be performed, but these methods are just for monocular images. These methods based on the

Hough transform are sensitive to the parameter sampling, that is, the angle resolution, and may induce the mistake of detecting the vanishing points, which have been confirmed in [29]. Consequently, the powerful RANSAC framework is applied to improve the limitation. For omnidirectional images, in [30], the Hough transform and RANSAC are also used for vanishing points detection, and the vanishing point information is used for plane extraction. In [31], the panoramic image is projected onto the unit sphere by a linear mapping. That is, the 2D image coordinate of the panoramic image is converted into the two spherical angles by using the linear relationship between them.

The second category in vanishing point detection is based on entropy. Bonev et al. proposed a system to navigate a mobile robot using an omnidirectional camera [32]. The directions of the corridors or roads, namely the directions of vanishing points, are estimated from the entropy analysis of a one-dimensional omnidirectional image. The density of features corresponding to more distant obstacles is higher in the omnidirectional image.

Although numerous methods have been proposed for detecting vanishing points, they have some disadvantages. For the first type of methods, based on the unit sphere, they need to realize the accurate intrinsic and extrinsic parameters of the camera because the image information will be projected to various spaces according to the geometric relationships of the camera, image plane, and the world. Besides, these methods do not utilize the orientation information of the edges in the image. For the second type of methods, based on the entropy, they are not generic for various scenarios. The entropy of the far place is not necessary the highest in any environment.

Therefore, in this paper, a novel approach is proposed to detect vanishing points using the orientation information in a panorama. The omnidirectional camera does not need any calibration owing to the usage of the orientation information. Panoramic image is generated by unfolding an omnidirectional image. The parallel lines in the real world become curves in a panoramic image, and also converge in a vanishing point. Choi et al. used the LDOS descriptors to extract orientation feature in a monocular image. These LDOS descriptors, originally used in a monocular image in [14] is applied in the panoramic image to characterize key orientation information in each image block. On the other hand, some unreliable blocks are first filtered out and the left useful blocks are interpolated by the radial basis function for constructing the orientation field in the overall panoramic image. Finally, potential vanishing points are detected by analyzing the curve behavior of the orientation field.

III. Brief Note on Local Dominant Orient Signature

This section briefly describes key computational procedure of the local dominant orientation signature (LDOS), which is originally discussed in [14], and adopted here for characterizing particular structural information in a given image.

First of all, a given image is divided into M non-overlapping blocks, each of which is composed of $B \times B$ pixels. One of the most information features for characterizing structural information is the edge orientation and edge magnitude (i.e., the strength of the orientation) of every pixel. The edge orientation and magnitude can be easily computed by the Sobel operator. For easy computation, the computed edge orientation is quantized into K levels in the range of (−90°, 90°], and, due to the symmetry, the orientation ranges (−180°, −90°] and (90°, 180°] can be mapped into the K levels. Hence, the orientation level of the block is computed as follows:

$$\theta_L = \left\lceil \frac{\theta + 90^\circ}{180^\circ} \times K \right\rceil, \quad \theta \in (-90^\circ, 90^\circ] \tag{1}$$

where θ and θ_L represent the edge orientation and the quantized orientation level, respectively.

With each block, the magnitude of these orientations is accumulated together to form the histogram of oriented gradient (HOG) for the block [33]. The LDOS descriptor D_p^i is defined as the i-th-level dominant orientation of block m, (i=1,…,K, m=1,…,M), and corresponds the accumulated HOG value of the orientation level. Hence, the HOG pattern of the LDOS is an important index for characterizing the structural information embedded in the block and identifying the similarity between blocks.

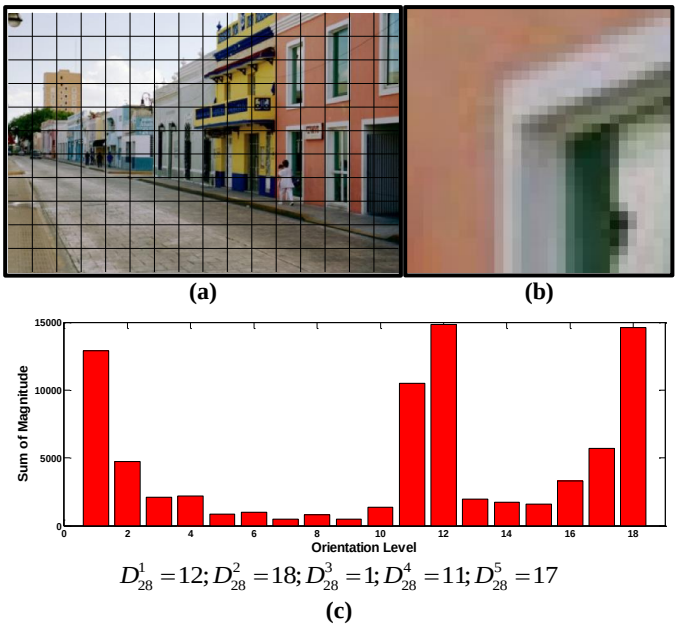


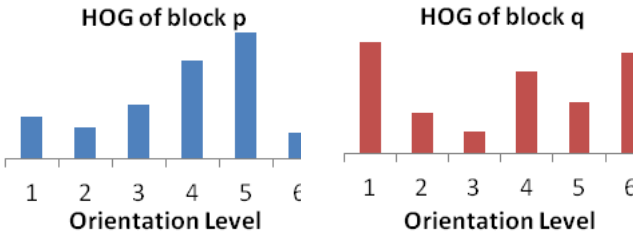
Figure 2 Illustrated example of the LDOS computation: (a) The original image of 9×16=144 blocks, (b) The 28-th block in the image, (c) The Histogram of Oriented Gradient and the LDOS of the 28-th block

Figure 2(a) shows an example of computing the LDOS of one image obtained from the wiki website [34]. The image is first divided into M=144 (i.e., 9×16) blocks, each of 30×30 pixels (i.e., B=30). The 28-th block of the image is shown in **Figure 2(b)**, and the orientation level of the block is computed by Equation (1). Also, the HOG of the orientation levels and the first five LDOS descriptors are summarized in **Figure 2(c)**. On the other hand, the first two LDOS descriptors are shown in **Figure 3**, marked as red lines and green lines, respectively. It can be observed that the LDOS descriptors greatly reflect the structural information in the image. In order to compare the similarity between block p and block q, the similarity index is defined as follows:

$$s(D_p, D_q, n) = \sum_{i=1}^{n} \omega_i \left| D_p^i - D_q^i \right| \quad (2)$$

where ω_i is the weight of the difference between the *i*-th LDOS descriptor. In general, $\omega_1 \geq \ldots \geq \omega_n$ and $\sum_i \omega_i = 1$.

Figure 3 The first two LDOS descriptors of the image



$$\Rightarrow D_p^1 = 5; D_p^2 = 4 \qquad \Rightarrow D_q^1 = 1; D_q^2 = 6$$

$$s(D_p, D_q, 2) = \sum_{i=1}^{2} \omega_i \times \min\left(\left|D_p^i - D_q^i\right|, K - \left|D_p^i - D_q^i\right|\right)$$

$$= 0.8 \times \min(4, 2) + 0.2 \times \min(2, 4) \qquad = 2$$

Figure 4 An example of similarity computation between two blocks

Because of the symmetry of edge orientation, Equation (2) is modified to equation (3) as follows.

$$s(D_p, D_q, n) = \sum_{i=1}^{n} \omega_i \times \min\left(\left|D_p^i - D_q^i\right|, K - \left|D_p^i - D_q^i\right|\right) \quad (3)$$

A simple example is provided in **Figure 4**. Here, only the first two dominant orientations are considered, and the weights are set as 0.8 and 0.2.

IV. Vanishing Point Detection using Local Dominant Orientation Signature

In the beginning, an omnidirectional image, such as the one shown in **Figure 5(a)**, is unfolded into a panoramic image, as shown in **Figure 5(b)**. The vanishing point detection algorithm includes the following four processes: (1) Computing the local dominant orientation signature (LDOS) descriptor from the structural features of the image; (2) Eliminating unreliable features through edge-feature filter and chain-size filter; (3) Constructing the orientation field by using the radial basis function (RBF); (4) Extracting the vanishing points. The detailed description of these four processes is discussed in the following.

A. Computing local dominant orientation signature

The whole panoramic image is first divided into M non-overlapping blocks whose size is $B \times B$ pixels. The LDOS descriptors of block m, $\{D_m^1, D_m^2, \ldots, D_m^K\}$, $m = 1, \ldots, M$, with K levels of orientation quantization, can be obtained as discussed in Section III. The first two descriptors of **Figure 6(a)**, namely, 1st-DO and 2nd-DO, are denoted by black and red lines, respectively, in **Figure 6(b)**.

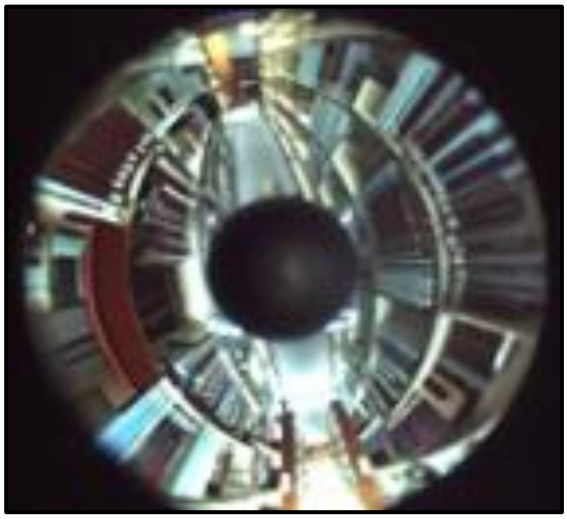
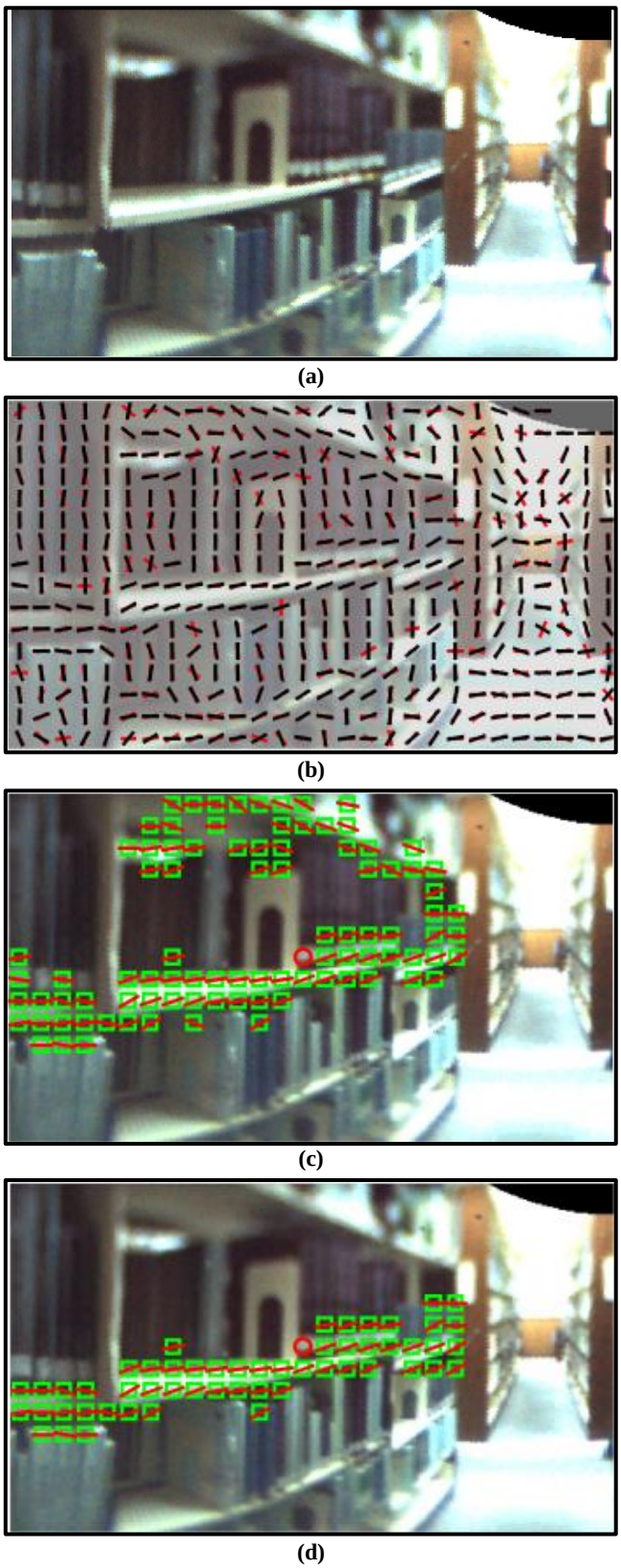


Figure 5 (a) An omnidirectional image

Figure 5 (b) The corresponding panoramic image



(a)

(b)

(c)

(d)

Figure 6 The LDOS and the results using different neighbor definitions: (a) the original image, (b) the first two LDOS descriptors on the brightened original image, (c) the block chain generated with the standard 8-neighbor approach, (d) the block chain generated with the orientation-neighbor approach

B. Feature Filtering, Block Linking, and Size Filtering

Carefully viewing the LDOS descriptors shown in **Figure 6(b)**, it can be observed that some of the descriptors do not correctly reflect the orientation of the corresponding block. Hence, it is required to remove these descriptors with less or inappropriate orientation information.

First of all, the blocks with a smaller magnitude or inappropriate orientation should be identified. It is done by verifying whether the magnitude, denoted as 1st-DM, of the first dominant orientation, 1st-DO, is too small or not. If 1st-DM is too small, then the block might contain few edge features and the block should be removed. On the other hand, because only parallel curves are the most important features for detecting vanishing points, the blocks with the feature of vertical lines should be removed. That is, the blocks with large 1st-DO should be removed. In the tested scenario, the following conditions are used to filter out these useless blocks:

1) 1st-DM > 500, and

2) |1st-DO| < 70°.

For different scenarios, other user-defined parameters can also be used to reflect the feature of the orientation information.

The next task is associating adjacent similar blocks to generate a useful block chain. Two blocks are considered as adjacent similar blocks if they are next to each other and the similarity index, defined in Equation (3), between them is good enough. For each of presentation, only 1st-DO and 2nd-DO are used to compute the similarity. Their weights are defined as follows: $\omega_1 = 0.8$, $\omega_2 = 0.2$, and the similarity threshold is set as 2. Therefore, the similarity value between Block p and Block q can be computed as follows:

$$\begin{aligned} s(D_p, D_q, 2) = {} & 0.8 \times \min(\left|D_p^1 - D_q^1\right|, 18 - \left|D_p^1 - D_q^1\right|) \\ & + 0.2 \times \min(\left|D_p^2 - D_q^2\right|, 18 - \left|D_p^2 - D_q^2\right|). \end{aligned} \quad (4)$$

To associate adjacent blocks, it requires proper definition of neighboring blocks. In standard image processing, either 4-neighbor or 8-neighbor approach is usually implemented. Because the 4-neighbor approach cannot associate the blocks at diagonal positions, the result might be too rough. With the 8-neighbor approach, the blocks can be associated in more directions. However, two groups of blocks with different dominant orientations could be connected if they are adjacent blocks and have certain level of similarity. This will further generate inappropriate association of blocks. One example is shown in **Figure 6(c)**. The (red) circle in the center denotes the starting point to associate blocks, and the (green) squares denote the blocks associated with the (red) circle. In this example, the blocks representing different parallel curves are linked together.

The standard 8-neighbor approach regards all the eight adjacent blocks as neighbors regardless of the dominant orientations of these blocks. Here, the orientation-neighbor is defined based on the extended line of the dominant orientation of the block as shown in **Figure 7**. The gray blocks represent the neighbors of the center block. The dominant orientation of

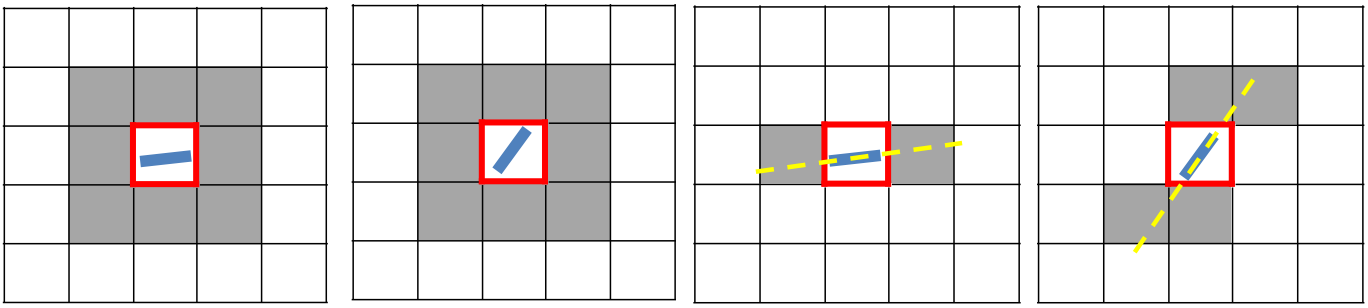
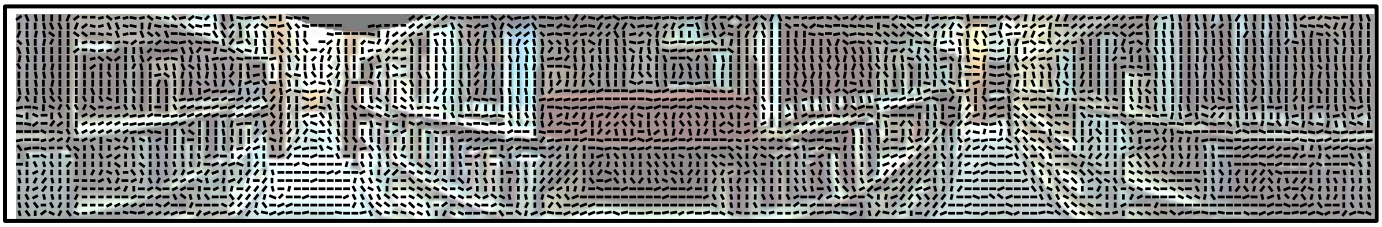
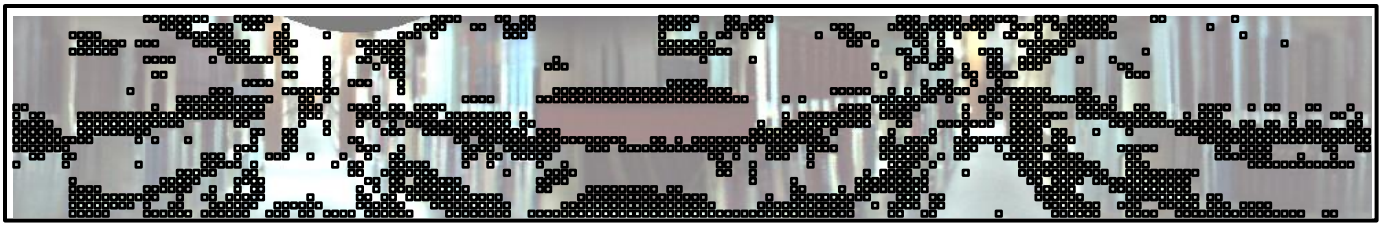


Figure 7 (a) 8-neighbor approach and (b) orientation-neighbor approach

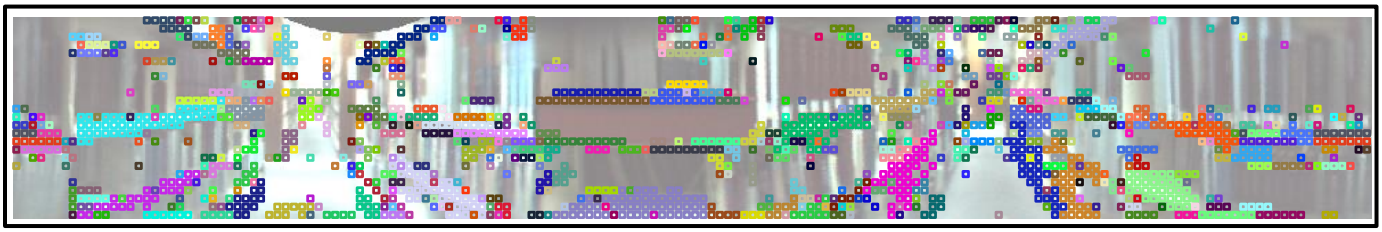
(a)



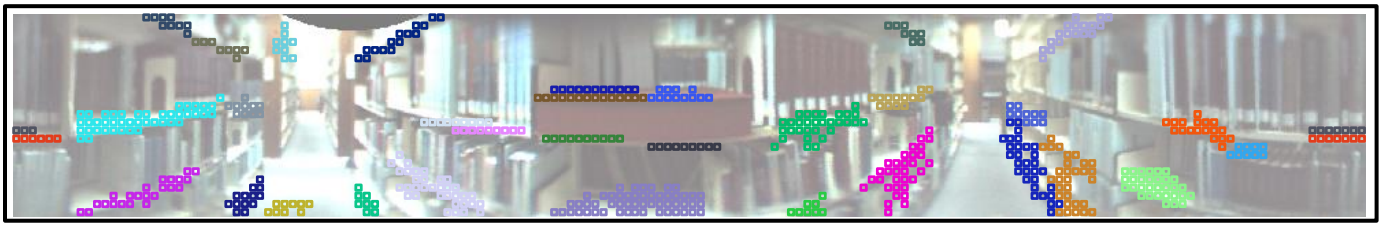
(b)



(c)



(d)



(e)

Figure 8 The result of performing feature filtering, block linking, and size filtering: (a) original panoramic image, (b) 1st-DO of the panoramic image, (c) after performing feature filtering. The blocks with strong and non-vertical 1st DO are preserved: (d) after the similar-block association. Different colors represent different groups of blocks: (e) after the chain-size filtering

each center block is plotted in the center. The (yellow) dash lines shown in **Figure 7(b)** denote the extended lines of the dominant orientation. The gray blocks passed by these (yellow) dash lines are defined as the orientation neighbors. **Figure 6(d)** shows the generated block chain by using the orientation-neighbor approach.

Finally, some chains with a small number of blocks are unreliable and could be noisy for generating a strong set of parallel lines. Hence, if the number of block chains is smaller than a constant *C*, says *C=7*, these blocks are also eliminated. That is, the block chains are preserved if the chain of blocks is strong enough to be considered as parallel lines.

Figure 8 displays the result of performing orientation-feature filtering, similar-block linking, and chain-size filtering. An original panoramic image is shown in **Figure 8(a)**. The 1st-DO in each block is displayed in **Figure 8(b)**. After performing the feature filtering, the blocks having a strong and non-vertical 1st-DO are maintained, and the perverted blocks are the black blocks shown in **Figure 8(c)**. Furthermore, after checking the similarity and the orientation-based adjacency, appropriate blocks are linked together to form block chains, as shown in **Figure 8(d)**, where different colors represent different sets of block chains. Finally, **Figure 8(e)** shows a small set of block chains after the chain-size filtering. It can be observed that all the identified block chains greatly map to the locations with strong parallel lines.

C. Orientation Field Construction

The orientations in the identified block chains are assumed to be the parallel curves in real-world scenario. The next task is to properly utilize the orientation information to construct a suitable parallel-curve trend by the interpolation technique with radial basis functions (RBFs). A radial basis function is typically used to build up an approximation of the following form [35]:

$$y(x) = \sum_{i=1}^{N} \omega_i \cdot \varphi(\|x - x_i\|) \tag{5}$$

where the approximating function $y(x)$ is represented as a sum of N radial basis functions, $\varphi(\cdot)$, each associated with a different center x_i, and weighted by an appropriate coefficient ω_i. Typical types of RBFs include Gaussian, multi-quadric, and inverse multi-quadric. In the tested scenario, an inverse multi-quadric of the following form is used:

$$\varphi(r) = \frac{1}{\sqrt{1+r^2}} \tag{6}$$

where r denotes the distance between x and x_i, that is, $r = \|x - x_i\|$, x_i denotes the location of the identified blocks, and x denotes the position of other blocks. That is, the orientation at x can be estimated by utilizing the approximation function of using the orientation at x_i.

On the other hand, the angular symmetry of panoramic should be further taken into account. Hence, during the interpolation, the identified block chains are copied to both left and right sides of the original panoramic image, as shown in **Figure 9**. Therefore, the leftmost and rightmost blocks of the original panoramic image can be correctly interpolated. **Figure 10(b)** shows the resulting interpolated orientation field by using the inverse multi-quadric RBF on the identified orientation in **Figure 10(a)**.

D.Vanishing Point Extraction

The vanishing points are the convergent points of sets of parallel lines on one vertical wall, horizontal floor or ceiling, viewed from one typical 2D image. Hence, the last task is to extract the location of vanishing points by observing the trend of parallel curves from the interpolated orientation field. For example, in a narrow environment such as corridor, vanishing points exists at the end of the aisle as shown in **Figure 11**, where these vanishing points are manually marked by (red) crosses.

From **Figure 10(b)**, it can be observed that the orientation of the blocks around the vanishing points exhibit either up-concave or down-concave patterns, as shown in **Figure 12(a)** and **(b)**, respectively. Therefore, the location of vanishing points can be identified by justifying the existence of concave pattern on the orientation field by executing the following four steps: (1) orientation classification; (2) concave extracting and grouping; (3) concave pairing and linking; and (4) finding vanishing points. The operation of these four steps is discussed in detail in the following.

First of all, the orientation of all the blocks are classified into three groups: small (either positive or negative) orientation, θ_S, large positive orientation, θ_{LP}, and large negative orientation, θ_{LN}. The classification decision is based on the following rule:

$$O_c(x,y) = \begin{cases} \theta_{LP}, & \text{if } O(x,y) > \phi_c \\ \theta_{LN}, & \text{if } O(x,y) < -\phi_c \\ \theta_S, & \text{otherwise} \end{cases} \tag{4}$$

where ϕ_c is a threshold constant, $O(\cdot,\cdot)$ denotes the original orientation, and $O_c(\cdot,\cdot)$ denotes the classified orientation field. After applying the rule to **Figure 10(b)**, these three groups of orientation blocks are marked as white, black, and gray blocks, respectively, as shown in **Figure 13(a)**.

Next, in order to correctly determine the up-concave and down-concave patterns, these small orientations located between the large positive and large negative orientations should be identified. That is, in an up-concave pattern, the locations of these orientations, horizontally from left to right, should be large negative, small, and large positive, and, in a down-concave pattern, the location should be large positive, small, and large negative. This phenomenon can be easily observed in **Figure 12(a)** and **(b)**. Hence, by utilizing these characteristics, the up-concave and down-concave patterns can be easily detected. It can be easily observed that the up-concave and down-concave patterns are usually existed above and below the so-called same-height line (SHL), which is the same horizontal line both in real world and on the panoramic image.

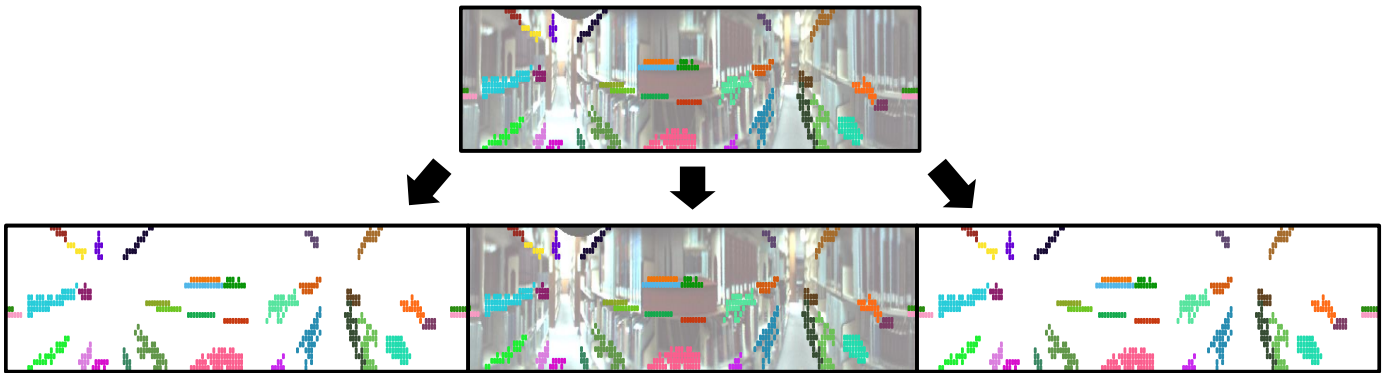
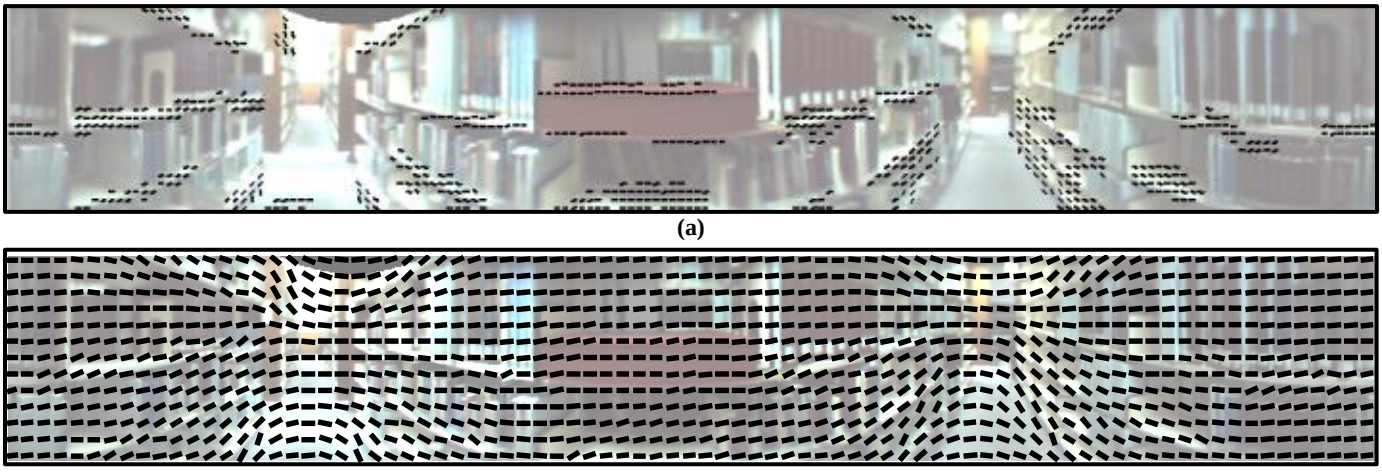


Figure 9 Three copies of identified block chains are used for interpolation



(a)

(b)

Figure 10 The construction of the orientation field: (a) the orientations of identified block chains, (b) the interpolated orientation field of the whole image

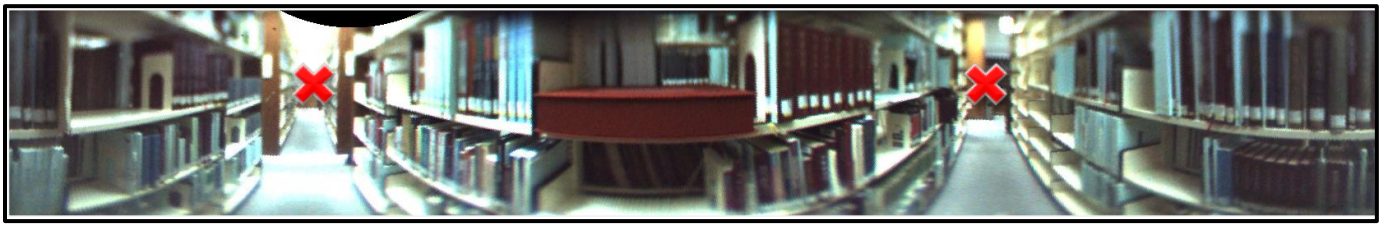
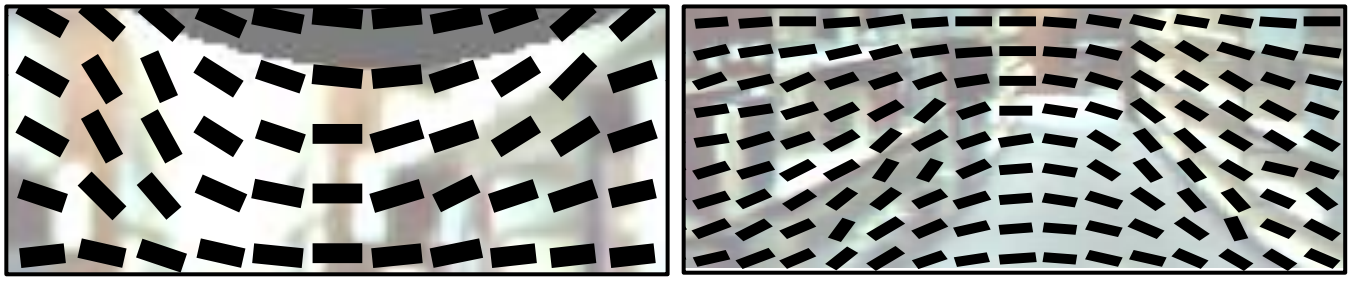


Figure 11 The manual vanishing points on the panoramic image



(a) **(b)**

Figure 12 The two types of concaves in orientation field: (a) up-concave and (b) down-concave

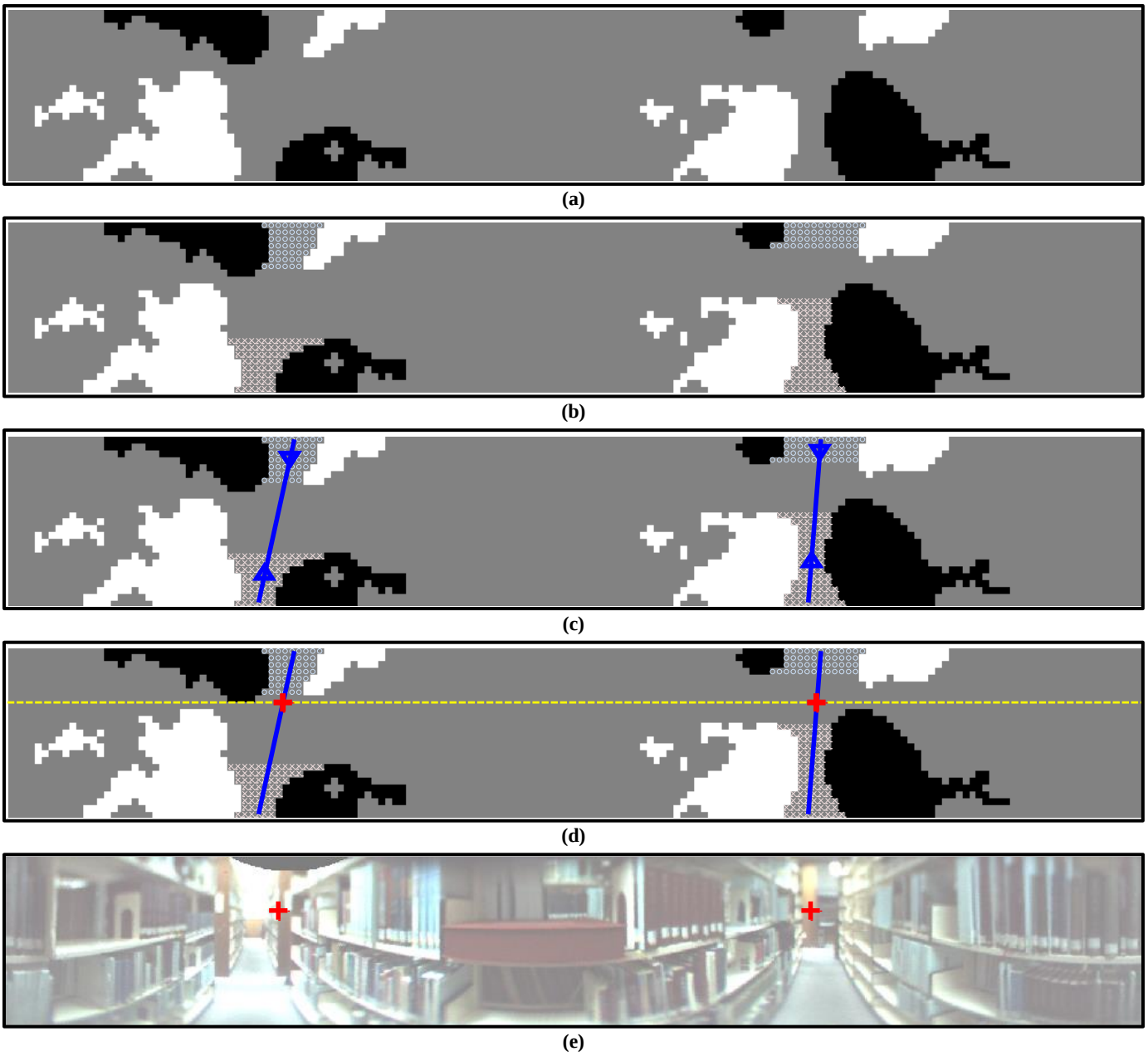
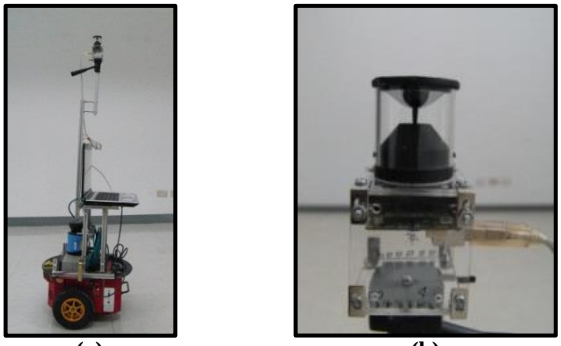


Figure 13 The result of vanishing point extraction: (a) three classes of orientation field, (b) small orientation for either up-concave or down-concave pattern, (c) connecting the centers of concave pairs, (d) identified vanishing points, i.e., the intersection of the paired lines and the same-height line

As shown in **Figure 13(b)**, the blocks marked by (blue) circles and (red) crosses between white and black regions are the blocks of small orientation and belong to up-concave and down-concave patterns, respectively. Hence, in **Figure 13(b)**, two separate pairs of up-concave and down-concave are identified.

The third step is to compute the geometric center of every up-concave or down-concave area, and, then, to pair these up-concave and down-concave together. As shown in **Figure 13(c)**, the inverted triangles are the geometric centers of up-concave, and the regular triangles are the geometric centers of down-concave. Furthermore, the down-concave center and the up-concave center are paired together if they are close to each other, as the two (blue) lines shown in **Figure 13(c)**.

Finally, the intersection of the paired lines and SHL are the vanishing points. As shown in **Figure 13(d)**, the (blue) solid lines are the paired lines connecting up-concaves and down-concaves and the (yellow) dash line is SHL. The vanishing points are the intersection of these two sets of lines and denoted as (red) plus signs. The identified vanishing points are marked into the original panoramic image as shown in **Figure 13(e)**.



(a) (b)

Figure 14 The apparatus used in the experiment. (a) The mobile robot equipped with an omnidirectional camera in (b).

V. Experimental Results and Analysis

A. *Experimental Setup*

The apparatus used in the experiment is displayed in **Figure 14**, where an omnidirectional camera (VS-C14U-80-ST) shown in **Figure 14(b)** is moutned on top of the mobile robot (Active Media Pioneer III) with on-board encoders. The camera is used to capture omnidirectional images. Four typical indoor environments are used to test the proposed vanishing point detection algorithm. They are: (I) a crossroad of bookshelves at our campus library; (II) a T-junction corridor outside the office, MD258; (III) a corridor space outside the office, MD604; and (IV) a corridor space outside the office, MD705. The panoramic views of these spaces are shown in **Figure 15**, and their geometric specifications are listed in **Table I**.

Table I The specification of the tested environments

	Location	Space	Size (m)	Characteristics
I	Campus Library	Crossroad	1/1.17	Noise form books and bookshelves
II	MD258	T-junction corridor	1.8/2.4	Three corridor directions
III	MD604	Corridor	2.5	Noise form railings and windows
IV	MD705	Corridor	2.5	Poor and non-uniform features, noise from boards

B. *Vanishing Points Detection Based on LDOS*

One snapshot of the experimental results of these four tested environments are shown in **Figure 16**- **Figure 19**. The links to the videos of these four tested scenarios can be found in the Appendix. In each figure, four intermediate results generated by the proposed vanishing point detection algorithm are shown, and they are: (a) the available orientations of the reliable block chains, (b) the interpolation result of RBF, i.e., the orientation field, (c) the identification of vanishing points shown as (red) plus signs, and (d) the vanishing points shown in the panoramic view. In (c), the white and black regions indicate the large positive and negative, respectively, orientations. The circles and crosses between the white and black regions denote the up-concaves and the down-concaves, respectively. The (yellow) dash line denotes the same-height line and the (blue) solid line denotes the pairing line linking the paired concaves.

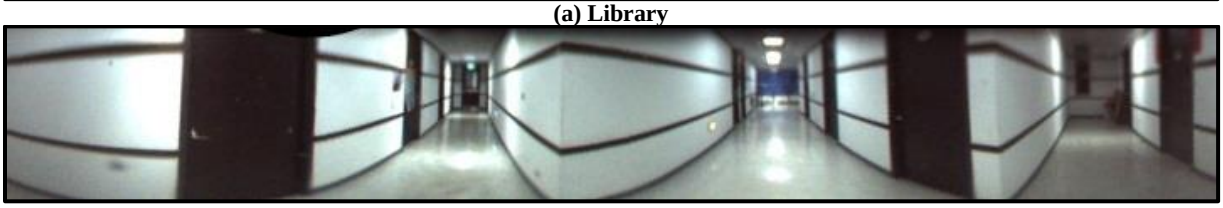
In **Figure 16(a)**, the reliable orientations of parallel curves in the environment are extracted and preserved. These parallel curves are generated from the boards of the bookshelves. Then, the orientation field in **Figure 16(b)** is constructed from the orientations shown in **Figure 16(a)** by the RBF interpolation. The concave extraction and paired result can be observed in **Figure 16(c)**. There are two up-concaves and two down-concaves in this figure, and the two couples of the concaves are generated according to their horizontal distance. In **Figure 16(d)**, these two vanishing points are detected correctly in spite of the complex surroundings. It shows that the proposed vanishing point detection approach is able to work in the complex indoor environment, and the vanishing point result is accurate.

In **Figure 17(a)**, the reliable orientations of the parallel curves in the environment are extracted and preserved. The parallel curves are generated from the textures on the wall. Unfortunately, the light reflection on floor also generates some orientations, which is unexpected. Then, the orientation field in **Figure 17(b)** is constructed from the orientations shown in **Figure 17(a)** by the RBF interpolation. The negative effect of light reflection can be reduced by interpolation. The concave extraction and paired result can be observed in **Figure 17(c)**. Although there are two down-concaves un-paired in this figure, the vanishing points can be detected with the vertical pairing lines. In **Figure 17(d)**, all the vanishing points are detected completely and correctly even though there are three floor directions appearing in the panoramic image. Moreover, the light reflection on the floor does not affect the task to detect the vanishing point.

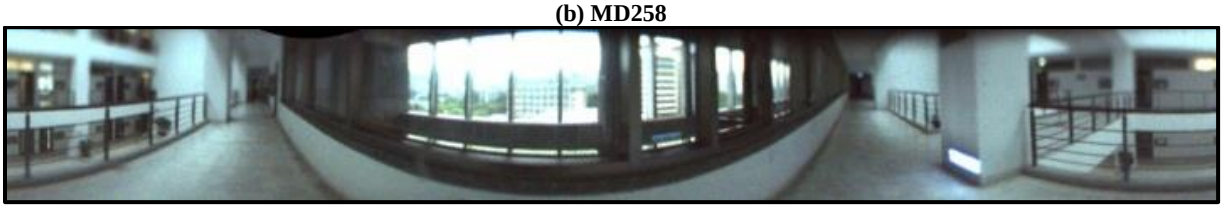
In the third environment, as shown in **Figure 18**, although the orientation information of railings and windows generate the parallel curves to help vanishing points detection, the objects behind the railings and windows may generate some noise. Many orientations of the opposite building are preserved as shown in **Figure 18(a)**. Then, the orientation field in **Figure 18(b)** is constructed from the orientations shown in **Figure 18 (a)** by the RBF interpolation. In **Figure 18(b)**, the regularity of the orientation field can be preserved in spite of the unexpected orientation information of the opposite building. The concave extraction and paired result can be observed in **Figure 18(c)**. Although there are two down-concaves un-paired in this figure, the vanishing points can be detected with the vertical pairing lines. In **Figure 18(d)**, the two vanishing points are detected correctly because the orientation information of the floor boundary is correct and adequate.

In the fourth and simple environment, as shown in **Figure 19**, the feature sources are only the floor boundary and the notice boards. In **Figure 19**, the reliable orientations of the parallel curves in the environment are extracted and preserved. In this figure, most of the orientation information of the floor boundary is extracted. **Figure 19(b)** is constructed from the orientations shown in **Figure 19(a)** by the RBF interpolation.

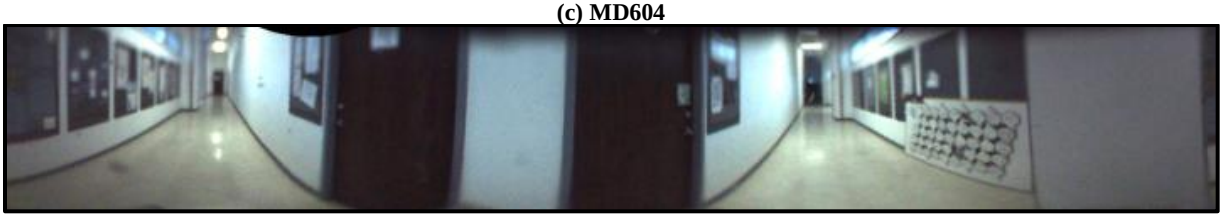
(a) Library



(b) MD258



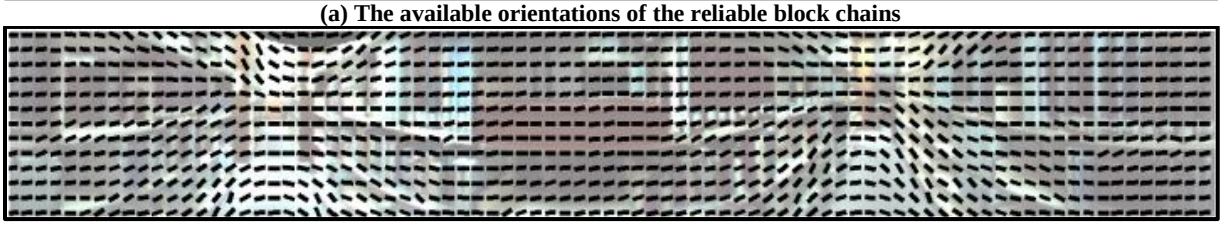
(c) MD604



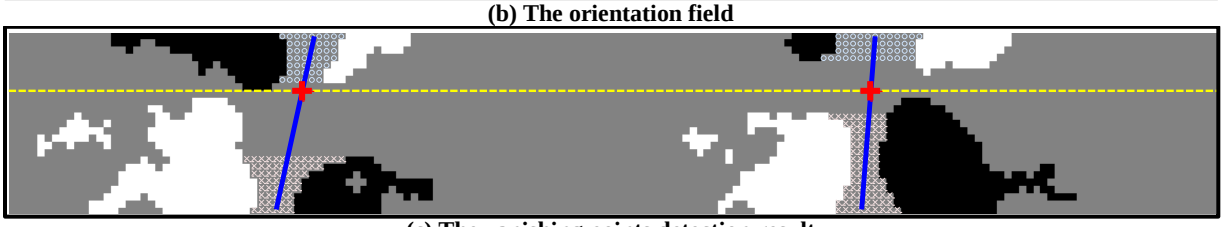
(d) MD705

Figure 15 The panoramic view of the tested environments

(a) The available orientations of the reliable block chains



(b) The orientation field

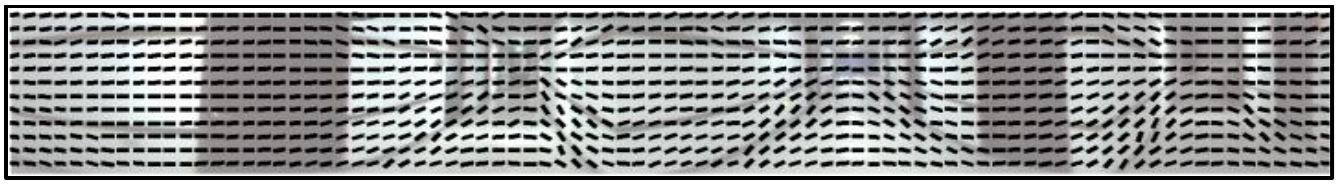


(c) The vanishing points detection result

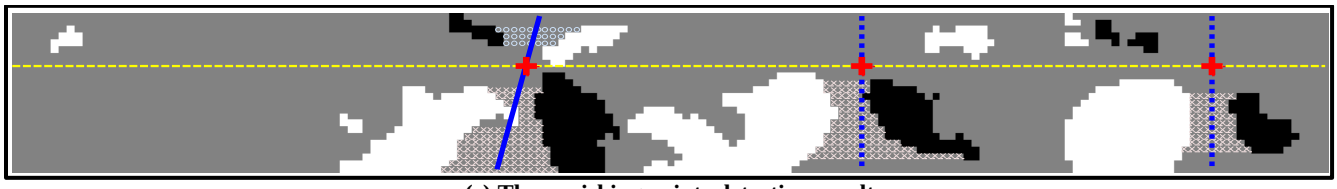
(d) The detected vanishing points on the original panoramic view

Figure 16 Vanishing point detection result at the campus library

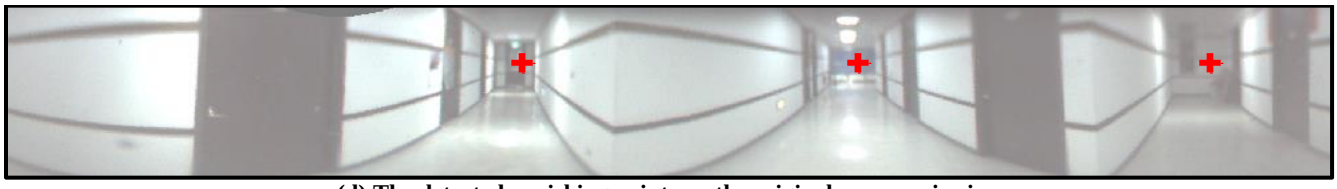
(a) The available orientations of the reliable block chains



(b) The orientation field

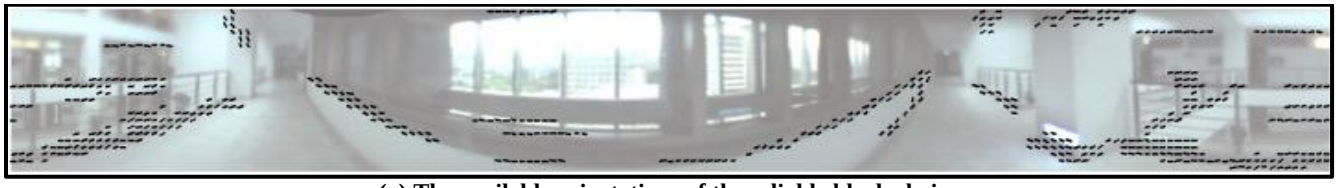


(c) The vanishing points detection result

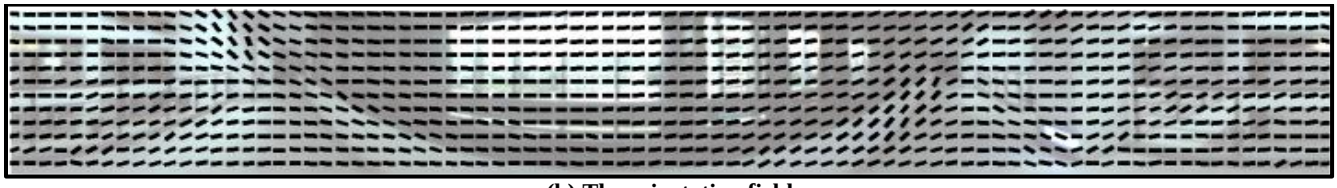


(d) The detected vanishing points on the original panoramic view

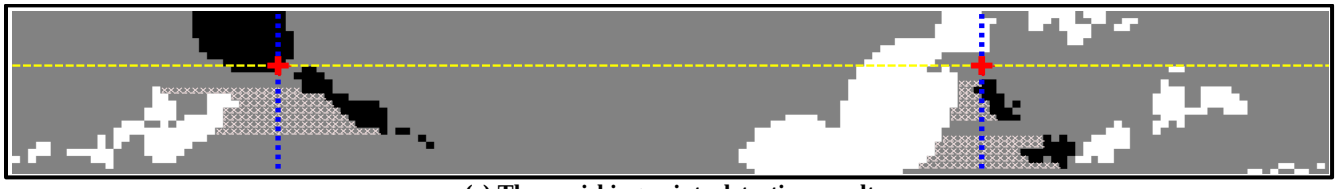
Figure 17 Vanishing point detection result at MD258



(a) The available orientations of the reliable block chains



(b) The orientation field

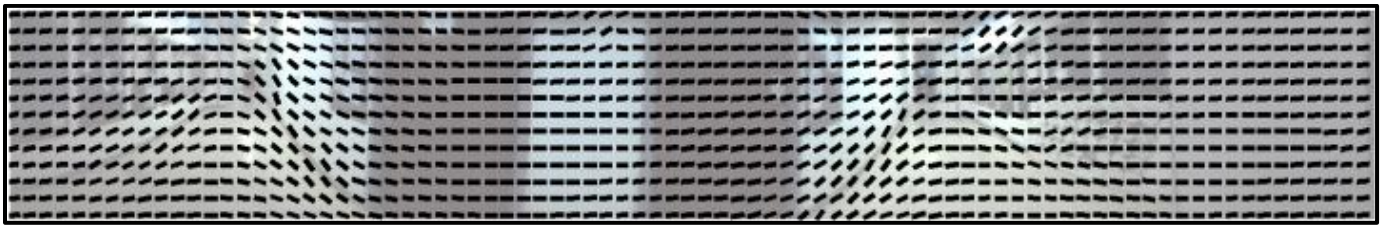


(c) The vanishing points detection result

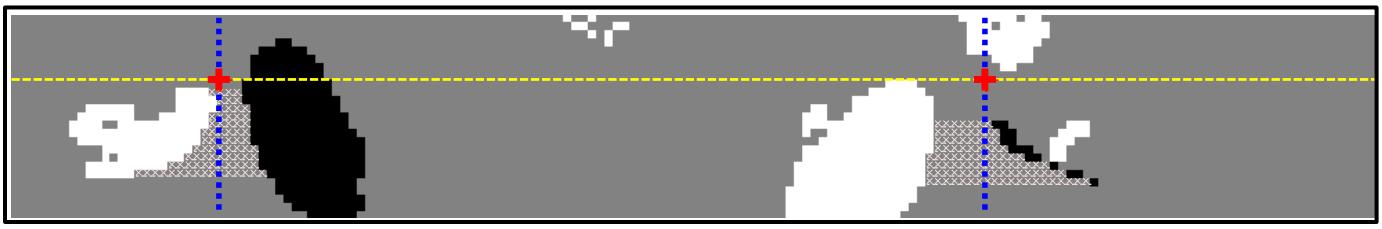
(d) The detected vanishing points on the original panoramic view

Figure 18 Vanishing point detection result at MD604

(a) The available orientations of the reliable block chains



(b) The orientation field



(c) The vanishing points detection result

(d) The detected vanishing points on the original panoramic view

Figure 19 Vanishing point detection result at MD705

The concave extraction and pairing result can be observed in **Figure 19(c)**. Although there are two down-concaves un-paired in this figure, the vanishing points can be detected with the vertical paired lines. In **Figure 19(d)**, the vanishing points are detected correctly owing to the sufficient orientations of the floor boundary which can be observed in **Figure 19(a)**.

C. Comparison with the Entropy Approach

In this section, the proposed vanishing point detection algorithm is compared with the existing technology based on the entropy of 1-D panorama, presented in [32][23], along with the vanishing points manually identified, denoted as the ground truth data. The data are shown in **TABLE II-TABLE V**, for these four tested environments.

As shown in **TABLE II**, the two vanishing points detected by the entropy method are acceptable although the right one is rather inaccurate. Using the entropy method, only one vanishing point can be correctly detected in the crossroad, and the other vanishing point is detected at a wrong location. It can be observed that the vanishing points detected by the proposed approach are very close to those manually identified points.

As shown in **TABLE III**, by using the entropy method, only one of three vanishing points can be detected in the T-junction corridor. However, the detected point is mapped at a wrong location. By using the proposed algorithm, all the vanishing points are correctly detected and the detected locations are very close to those manually identified points.

As shown in **TABLE IV**, although two vanishing points are detected by the entropy method, their locations are inaccurate. On the contrary, all the vanishing points can be correctly detected by the proposed algorithm, and the detected locations are close to those manually identified points. It shows that the locations with high entropy are not necessarily the vanishing points, and the features from parallel curves are more dependable for detecting vanishing points.

As shown in **TABLE V**, the two vanishing points detected by the entropy method are inaccurate. The result of using the entropy method in this environment is better than that in the previous three environments because the number of the vanishing points is only two and the vanishing points are located at the places with high entropy. On the contrary, all the vanishing points can be correctly detected by the proposed algorithm, and the detected locations are close to those manually identified points.

TABLE II COMPARISON OF VANISHING POINT DETECTION AT THE CAMPUS LIBRARY

Location / Method	VP-1	VP-2	Average Error
Ground Truth	-100.91°	77.14°	
LDOS	-93.57°	74.55°	4.97°
Entropy	-96.37°	95.29°	11.34°

TABLE III COMPARISON OF VANISHING POINT DETECTION AT MD258

Location / Method	VP-1	VP-2	VP-3	Average Error
Ground Truth	-42.11°	49.11°	143.94°	
LDOS	-41.05°	50.60°	145.21°	1.27°
Entropy	N/A	24.71°	N/A	24.71°

TABLE IV COMPARISON OF VANISHING POINT DETECTION AT MD604

Location / Method	VP-1	VP-2	Average Error
Ground Truth	-105.75°	73.51°	
LDOS	-107.45°	81.99°	5.09°
Entropy	-177.45°	-3.50°	74.35°

TABLE V COMPARISON OF VANISHING POINT DETECTION AT MD705

Location / Method	VP-1	VP-2	Average Error
Ground Truth	-117.34°	60.94°	
LDOS	-125.55°	74.33°	10.80°
Entropy	-90.76°	90.76°	28.20°

VI. CONCLUDING REMARKS

For improving the capability of localizing and navigating the mobile robots in an indoor environment, this paper presents a novel approach of identifying one of the key structural features, the vanishing points. In a 2D image, the vanishing points are the convergent points of parallel lines in the real world and can be detected by the proposed approach of using the orientation information. The orientation information is computed locally based on the Sobel edge detection and then characterized by the local dominant orientation signature descriptors. Useful and useless descriptors are further distinguished by the edge-feature filter and the chain-size filter. The rest of useful descriptors are then interpolated by the radial basis function for constructing the overall orientation field. Finally, the vanishing points can be identified based on the curve changing behavior. The proposed approach of vanishing point detection has been extensively tested in four typical indoor environments such as narrow pathway library and multi-path corridor. Experimental results demonstrate good performance compared with existing approach.

In the paper, only the directional information from the omnidirectional images is utilized. The color information for identifying different types of objects should be another useful index for robot localization and navigation. Also, other than the vanishing points, vertical lines and floor regions are two important features for the mobile robots to localize and navigate themselves in an indoor environment. The integration or fusion of these features can greatly improve the robustness of robot capability.

ACKNOWLEDGMENT

This work was supported in part by the Ministry of Education, Taiwan, and the Ministry of Science and Technology, Taiwan, under Grants, NSC-102-2221-E-002-247-MY3, and MOST 104-2627-M-002-016.

APPENDIX

Video links to the tested scenarios:

[1] MD258 at http://youtu.be/TDJ4UlNsRnE

[2] MD604 at http://youtu.be/griV5gSwG2M

[3] MD705 at http://youtu.be/Sshh1FWhEXI

[4] NTULib at http://youtu.be/geIPjnON7Xc

REFERENCES

[1] H. Huttenrauch, A. Green, M. Norman, L. Oestreicher, and K. Severinson Eklundh, “Involving Users in the Design of a Mobile Office Robot,” IEEE Transactions on Systems, Man, and Cybernetics - part C: Applications and Reviews, Vol. 34, No. 2, pp. 113-124, May. 2004.

[2] E. Menegatti, A. Pretto, A. Scarpa, and E. Pagello, “Omnidirectional Vision Scan Matching for Robot Localization in Dynamic Environments,” IEEE Transactions on Robotics, Vol. 22, No. 3, pp. 523-535, Jun. 2006. CrossRef

[3] C. Silpa-Anan and R. Hartley, "Visual localization and loop-back detection with a high resolution omnidirectional camera," The 6th Workshop on Omnidirectional Vision, Camera Networks and Non-classical cameras, Oct. 21, 2005.

[4] Z. Ma and H. Kim, “Heading Direction Computation of Golf-Ball Collecting Robot Using Vanishing Points,” in Proceedings of the IEEE International Conference on Information and Automation, Shenzhen, China, pp. 792-796, Jun. 6-8, 2011.

[5] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, “DTAM: Dense tracking and mapping in real-time,” in Proceedings of the International Conference on Computer Vision (ICCV), 2011. CrossRef

[6] F. Zheng, R. Schubert, and G. Welch, “A General Approach for Closed-Loop Registration in AR,” in Proceedings of the IEEE Conference on Virtual Reality, Orlando, FL, USA, pp. 47-50, Mar. 16-20, 2013. CrossRef

[7] J. Engel, T. Schops, and D. Cremers, "LSD-SLAM: Large-Scale Direct Monocular SLAM," European Conference on Computer Vision, Zurich, Switzerland, pp. 1-16, Sep. 6-12, 2014. CrossRef

[8] J. Sturm, E. Bylow, F. Kahl, and D. Cremers, “CopyMe3D: Scanning and printing persons in 3D,” in Pattern Recognition, Editor: J. Weicker, M. Hein, and B. Schiele, Berlin Heidelberg: Springer, pp. 405-414, 2013.

[9] Y. Gao, J. Luo, H. Qiu, and B. Wu, "Survey of structure from motion," in Proceedins of International Conference on Cloud Computing and Internet of Things, Changchun, China, pp. 72-76, Dec. 13-14, 2014. CrossRef

[10] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo Tourism: Exploring Photo Collections in 3D," ACM Transactions on Graphics, Vol. 25, No. 3, pp. 835-846, Jul. 2006. CrossRef

[11] J. Heller, M. Havlena, A. Sugimoto, and T. Pajdla, "Structure-from-Motion Based Hand-Eye Calibration using L_∞ Minimization," in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, Colorado, USA, pp. 3497-3503, Jun. 20-25, 2011

[12] M.-D. Yang, C.-F. Chao, K.-S. Huang, L.-Y. Lu, and Y.-P. Chen, "Image-based 3D scene reconstruction and exploration in augmented reality," Automation in Construction, vol. 33, pp. 48-60, Aug. 2013. CrossRef

[13] D. Turne, A. Lucieer, and C. Watson, "An Automated Technique for Generating Georectified Mosaics from Ultra-High Resolution Unmanned Aerial Vehicle (UAV) Imagery, Based on Structure from Motion (SfM) Point Clouds," Remote Sensing, vol. 4, no. 5, pp. 1392-1410, May 2012. CrossRef

[14] J. Choi, W. Kim, H. Kong, and C. Kim, "Real-time vanishing point detection using the Local Dominant Orientation Signature," 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video, Antalya, Turkey, May 16-18, 2011.

[15] J. A. Gaspar, "Omnidirectional Vision for Mobile Robot Navigation," PhD thesis, Technical Superior Institute, Technical University of Lisbon, Dec. 2002.

[16] R. Ebrahimpour, R. Rasoolinezhad, Z. Hajiabolhasani, and M. Ebrahimi, "Vanishing Point Detection in Corridors: Using Hough Transform and K-means Clustering," IET Computer Vision, Vol. 6, No. 1, pp. 40-51, Jan. 2012. CrossRef

[17] L. F. Posada, K. K. Narayanan, F. Hoffmann and T. Bertram, "Floor Segmentation of Omnidirectional Images for Mobile Robot Visual Navigation," in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, pp. 804-809, Oct. 18-22, 2010. CrossRef

[18] S. Kim, S.-Y. Oh, "SLAM in Indoor Environments Using Omni-directional Vertical and Horizontal Line Features," Journal of Intelligent and Robotic Systems, Vol. 51, No. 1, pp. 31-43, Jan. 2008. CrossRef

[19] J. M. Coughlan and A. L. Yuille, "Manhattan World: Orientation and Outlier Detection by Bayesian Inference," Neural Computation, Vol. 15, No. 5, pp. 1063-1088, May 2003. CrossRef

[20] A. T. Martins, P. M. Q. Aguiar, and M. A. T. Figueiredo, "Orientation in Manhattan: Equiprojective Classes and Sequential Estimation," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No.5, pp. 822-827, May 2005. CrossRef

[21] S. T. Barnard, "Interpreting Perspective Image," Artificial Intelligence Journal, Vol. 21, No. 4, pp. 435-462, Nov. 1983. CrossRef

[22] J. Kim, M. Hwangbo, and T. Kanade, "Spherical Approximation for Multiple Cameras in Motion Estimation: Its Applicability and Advantages," Computer Vision and Image Understanding, Vol. 114, No. 10, pp. 1068-1083, Oct. 2010. CrossRef

[23] J. P. Barreto and H. Araujo, "Geometric Properties of Central Catadioptric Line Images and Their Application in Calibration," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 8, pp. 1327-1333, Aug. 2005. CrossRef

[24] M. E. Antone and S. Teller, "Automatic Recovery of Relative Camera Rotations for Urban Scenes," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA, Vol. 2, pp. 282-289, Jun. 13-15, 2000. CrossRef

[25] M. Bosse, R. Rikoski, J. Leonard, S. Teller, "Vanishing Points and 3D Lines From Omnidirectional Video," in Proceedings of the IEEE International Conference on Image Processing, Rochester, New York, Vol.3, pp. 513- 516, Jun. 24-28, 2002. CrossRef

[26] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon, "Rotation Estimation and Vanishing Point Extraction by Omnidirectional Vision in Urban Environment," the International Journal of Robotics Research, Vol. 31, No. 1, pp. 63-81, Jan. 2012.

[27] S.-B. Kim, J.-C. Bazin, H.-K. Lee, K.-H. Choi, S.-Y. Park, "Ground Vehicle Navigation in Harsh Urban Conditions by Integrating Inertial Navigation System, Global Positioning System, Odometer and Vision Data," IET Radar, Sonar and Navigation, Vol. 5, No. 8, pp. 814–823, Oct. 2011. CrossRef

[28] L . Quan and R. Mohr, "Determining Perspective Structures Using Hierarchical Hough Transform," Pattern Recognition Letters, Vol. 9, No. 4, pp. 279-286, May 1989. CrossRef

[29] J. A. Shufelt, "Performance Evaluation and Analysis of Vanishing Point Detection Techniques," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 3, pp. 282-288, Mar. 1999. CrossRef

[30] J.-C. Bazin, P.-Y. Laffont, I. Kweon, C. Demonceaux, and P. Vasseur, "An Original Approach for Automatic Plane Extraction by Omnidirectional Vision," in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, pp. 752-758, Oct. 18-22, 2010. CrossRef

[31] A. Banno and K. Ikeuchi, "Omnidirectional Texturing Based on Robust 3D Registration through Euclidean Reconstruction from Two Spherical Images," Computer Vision and Image Understanding, Vol. 114, No. 4, pp. 491-499, Apr. 2010. CrossRef

[32] B. Bonev, M. Cazorla and F. Escolano, "Robot Navigation Behaviors Based on Omnidirectional Vision and Information Theory," Journal of Physical Agents, Vol. 1, No. 1, pp. 27-36, July 2007.

[33] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, Vol. 1, pp. 886–893, Jun. 2005. CrossRef

[34] J. Barillari. (2005, May 22). Merida, Mexico. Wikipedia, [Online]. VIEW ITEM

[35] (2012, Apr. 27). Radial basis function. Wikipedia, [Online]. VIEW ITEM