Real-Time Security Monitoring Around a Video Surveillance Vehicle With a Pair of Two-Camera Omni-Imaging Devices

Pei-Hsuan Yuan, Kuo-Feng Yang, and Wen-Hsiang Tsai, *Senior Member, IEEE*

***Abstract*—A pair of two-camera omni-imaging devices is designed for use on the roof of a video surveillance vehicle, and corresponding 3-D vision-based techniques for real-time security surveillance around the vehicle are proposed, which may be used to monitor passing-by persons around the vehicle. First, the design of the pair of two-camera omni-imaging devices, each device consisting of two omnicameras, with their optical axes vertically aligned, is described. Then, a new analytic technique for fast 3-D space data acquisition that is based on a panomapping method for image-to-world space transformation, as well as the rotational invariance property of the omni-image, is proposed. Techniques for constructing top- and perspective-view images for the convenient observation of the monitored environment are also proposed. Finally, 3-D vision techniques for automatically detecting passing-by persons and computing their locations and body heights are proposed, followed by experimental results, which show the precision and feasibility of the proposed techniques.**

***Index Terms*—Omnicamera, omni-image, passing-by person detection, perspective-view image, security monitoring, top-view image, video surveillance, 3-D data acquisition.**

I. Introduction

VIDEO surveillance of environments has intensively been studied in recent years [1], [2]. To increase the mobility of video surveillance systems, vehicles are used as the carriers of such systems [3]–[5]. Applications of video surveillance vehicles include the dynamic monitoring of outdoor events, detection of passing-by persons, assistance for safe driving, warning of dangerous activities, and watching of environmental changes. Various types of cameras were used to capture environment images. Gandhi and Trivedi [3] made a good survey of vehicle surround capture techniques and proposed a novel omnivideo-based approach to synthesize dynamic panoramic surround maps using the stereo and motion analysis of video images from a pair of omnicameras on a vehicle. Micheloni *et al.* [4] used an autonomous vehicle to monitor moving objects in indoor environments, whereas Chen and Tsai [5] designed an autonomous vehicle to monitor planar objects on walls in buildings, and both works used projective cameras to capture environment images. Onoe *et al.* [6] and Mituyosi *et al.* [7] used omnicameras for tracking human body features. A video surveillance system for localizing objects using multiple omnicameras was proposed by Morita *et al.* [8]. Some related works that use pairs of omnicameras with hyperboloidal-shaped reflective mirrors can be found in [9] and [10]. In particular, Koyasu *et al.* [9] proposed an omnidirectional stereo system that consists of two vertically aligned omnicameras to detect and track obstacles. In addition, Ukida *et al.* [10] used a similar system and a space encoding scheme to acquire 3-D environment data for various applications. Furthermore, a method that reconstructs the 3-D data of static nearby vehicles by a mobile robot using a stereo omnicamera (a two-mirror omni-imaging system) was proposed by Meguro *et al.* [11]. Many more related techniques can be found in [15]–[27], which will be reviewed after presenting the method that is proposed in this paper.

Most of the aforementioned works are about indoor visual surveillances, and the speeds for computing the 3-D data were generally slow. Some of the works used autonomous vehicles to carry the camera system. Very few studies about security monitoring were reported to use a *video surveillance vehicle*—a car that is equipped with a camera system on its roof. In this paper, it is desired to design a video surveillance vehicle of this type, equipped with an omni-imaging system for real-time wide-area monitoring applications and has the following capabilities:

1) constructing a top-view image of the video surveillance vehicle's surrounding area for convenient observation of the environment security;
2) automatically constructing a perspective-view image of any direction or at any time instant specified by the user for real-time inspection;
3) automatically detecting and displaying any suspicious passing-by person;
4) measuring the 3-D features (i.e., location and height) of the passing-by person for security investigation purposes.

More specifically, for the specific purpose of security monitoring around a video surveillance vehicle, in this paper, we

Manuscript received October 21, 2010; revised February 27, 2011 and April 27, 2011; accepted June 4, 2011. Date of publication July 25, 2011; date of current version October 20, 2011. This work was supported in part by the National Science Council of Taiwan through Project 98-2221-E-009-116-MY3. The review of this paper was coordinated by Dr. M. S. Ahmed.

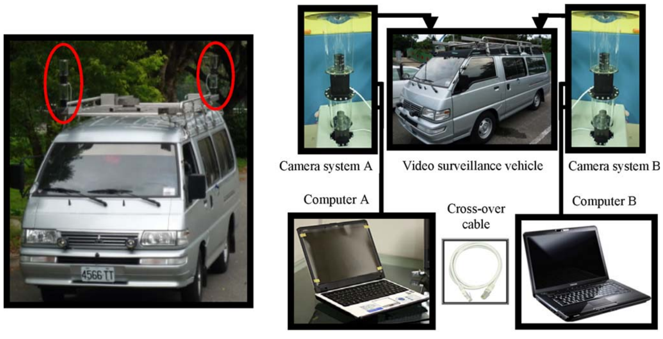
P. H. Yuan was with the Computer Vision Laboratory, Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan. She is now with the Taiwan Semiconductor Manufacturing Company, Hsinchu 300, Taiwan (e-mail: yuanegg.cs97g@nctu.edu.tw).

K. F. Yang is with the Institute of Computer Science and Engineering, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: castle7322.cs97g@nctu.edu.tw).

W. H. Tsai is with the Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: whtsai@cis.nctu.edu.tw).

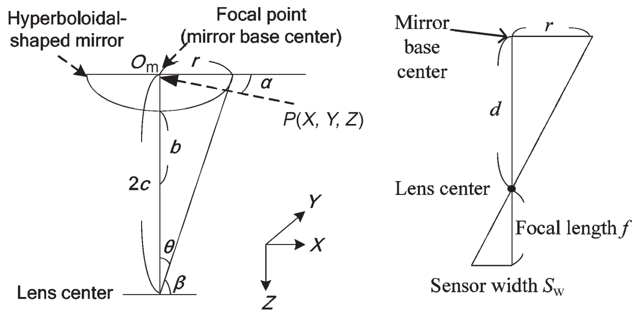
Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2011.2162862



(a) (b)

Fig. 1. Video surveillance vehicle and network used in this paper. (a) Vehicle with omnicameras (circled in red). (b) Network-connecting computers.



(a) (b)

Fig. 2. Omnicamera structure. (a) Omnicamera geometry. (b) Geometry between the hyperboloidal-shaped mirror and the projective camera.

design an *omni-imaging device* that is composed of two catadioptric cameras whose optical axes vertically aligned, and affix a pair of such omni-imaging devices to the roof of a vehicle to acquire the environment data. One device is affixed to the right-front corner of the vehicle's roof, and the other device is affixed to the left-rear corner [see Fig. 1(a)]. A new method that skillfully combines the uses of such a pair of omni-imaging devices, the so-called panomapping technique [12], as well as the rotational invariance property of the omni-image to compute the 3-D data of real-world points, is then proposed. The computation is based on *table lookup* and *analytic formulas* and can therefore be carried out in real time to construct both top- and perspective-view images for instantaneous observation of the vehicle's surrounding environment. In addition, a vision-based scheme that uses a vertical-line property in the omni-image for detecting passing-by persons, computes their locations and heights, and displays such feature data both from the top and the perspective views for convenient inspection is proposed, which is not found in previous works. A detailed comparison of the proposed method with related existing methods will also be given.

In the remainder of this paper, the configuration of the proposed system, the design of the omnicamera system, and the proposed technique for computing 3-D data are described in Section II. The proposed techniques for the detection and monitoring of a passing-by person are described in Section III. The schemes proposed for generating top- and perspective-view images are described in Section IV, followed by some experimental results and a comparison of the proposed method with related methods in Section V. Finally, the conclusions and descriptions of the contributions made in this paper are given in Section VI.

II. System Configuration and Design of the Proposed Omni-Imaging Devices

A. System Configuration

As illustrated in Fig. 1(b), the proposed system that is used in this paper includes a video surveillance vehicle, a pair of two-camera omni-imaging devices CS_A and CS_B that are affixed on the vehicle's roof, and two laptop computers COM_A and COM_B inside the vehicle. Each two-camera omni-imaging device is controlled by a laptop computer, and a local network was designed to connect the two computers. In particular, COM_A is used to display the top-view image of the surrounding area of the video surveillance vehicle, and COM_B is used to display the perspective-view images of specified directions outside the surveillance vehicle. The system process is divided into the following two phases: 1) the *learning* phase and 2) the *patrolling* phase. In the learning phase, the panomapping tables of the omnicameras used are constructed in advance, and in the patrolling phase, the video surveillance vehicle is driven in the outdoor environment for real security-monitoring applications.

B. Camera Design Principle

The structure of each catadioptric omnicamera with a reflective mirror of the hyperboloidal shape used in this paper is illustrated in Fig. 2, where the *world coordinate system* (WCS) is specified by (X, Y, Z), with its origin O_m located at the mirror base center, which we assume to be coincident with one focal point of the hyperboloidal shape of the mirror. The shape of the mirror in the *camera coordinate system* (CCS), which is specified by (x, y, z), with its origin O_a located at the middle point between the point O_m and the lens center of the projective camera, may be described [12] as

$$\frac{s^2}{a^2} - \frac{z^2}{b^2} = -1, \quad s = \sqrt{x^2 + y^2} \tag{1}$$

where a and b are two parameters of the hyperboloidal shape. The parameter d, as shown in Fig. 2(b), is the distance between the camera lens center and the mirror base center, whose value can be obtained by a simple formula $d = 2c$, with $c^2 = a^2 + b^2$. In addition, the axis of the camera is aligned with the axis of the mirror, and the lens center is fixed at the other focal point of the mirror shape.

By the shape geometry of a hyperboloid described by (1), the value α that specifies the elevation angle of a real-world point P, as shown in Fig. 2(a), can be computed [10] by

$$\tan\alpha = \frac{(b^2 + c^2)\sin\beta - 2bc}{(b^2 - c^2)\cos\beta} \tag{2}$$

where the angle β, as shown in Fig. 2(a), can be computed as

$$\beta = \pi/2 - \theta \tag{3}$$

with

$$\theta = \tan^{-1}(r/2c) \tag{4}$$

where r is the distance of P to O_{m} on the XY plane and equals the radius of the circular mirror base when $\alpha = 0$. In Fig. 2(b), by the principle of similar triangles, we have

$$\frac{d}{r} = \frac{f}{S_{\mathrm{w}}} \tag{5}$$

where f is the focal length of the projective camera, and S_{w} is the width of the square CMOS sensor in the camera.

The goal of the omnicamera design in this paper is to construct a mirror of the hyperboloidal shape and determine the distance from the camera to the mirror under the constraint that f, S_{w}, and r are of given values, which fits the structure of the vehicle roof. The projective camera that we use has $f = 6$ mm and $S_{\mathrm{w}} = 2.4$ mm, and to affix the cameras to a steel rod on the vehicle roof, we chose the mirror to have a base with an appropriate radius r of 4 cm. Therefore, according to (5) and the aforementioned fact that $d = 2c$, we can derive d and c to be $d = 10$ cm and $c = 5$ cm, respectively. In addition, according to (3) and (4) and assuming that α is zero, the values of the angles θ and β can be computed to be $\theta = 21.8^\circ$ and $\beta = 68.2^\circ$, respectively. Finally, using β and c, we may reduce (2) to be an equation with only one variable b of the following form:

$$(b^2 + 25) \times 0.9285 - 10 \times b = 0$$

which may then be solved to get $b = 3.39$ cm. In addition, by $c^2 = 5^2 = a^2 + b^2$, a can be solved to be 3.68 cm. Thus, all of the following desired parameters of the hyperboloidal-shaped mirror are obtained, all in centimeters:

1) $a = 3.68$;
2) $b = 3.39$;
3) $c = 5$;
4) $d = 10$.

C. Three-Dimensional Data Acquisition

Each omni-imaging device that is used in this paper consists of two omnicameras that are designed in the aforementioned way. The two cameras—the *upper* and the *lower* cameras—are tied together, with their mirrors both facing down and their axes vertically aligned. An alternative illustration of the upper omnicamera configuration is shown in Fig. 3, where the *image coordinate system* (ICS) is specified by (u, v), with the image center being its origin, and a pixel p with image coordinates (u, v) corresponds to a real-world point P with coordinates (X, Y, Z).

In this paper, a new technique is proposed to compute the 3-D data (X, Y, Z) of each real-world point P in the WCS using the two elevation angles α_1 and α_2 of P with respect to the mirror bases of the two omnicameras in an omni-imaging device, as illustrated in Fig. 4, where the values of α_1 and α_2 may be obtained using the panomapping method proposed by Jeng and Tsai [12].

More specifically, as shown in Figs. 3 and 4(a), each image pixel p is the projection of a real-world point P whose light ray goes onto the mirror of the omnicamera and is then reflected onto the camera imaging plane, resulting in an azimuth angle θ (shown in Fig. 3) and an elevation angle α (shown in Fig. 4(a) as α_1 or α_2). Through the panomapping method proposed in [12], given the image coordinates (u, v) of an image pixel p that corresponds to a real-world point P, the elevation angle α and the azimuth angle θ of P can be obtained by a table lookup using a *panomapping table*. One example of the panomapping table is shown in Table I. In addition, according to the mirror surface geometry, which we assume to radially be symmetric, a relationship, called the *radial stretching function* and denoted as f_{s}, between the elevation angle α and the radial distance r of p in the image plane with respect to the image center is established to be

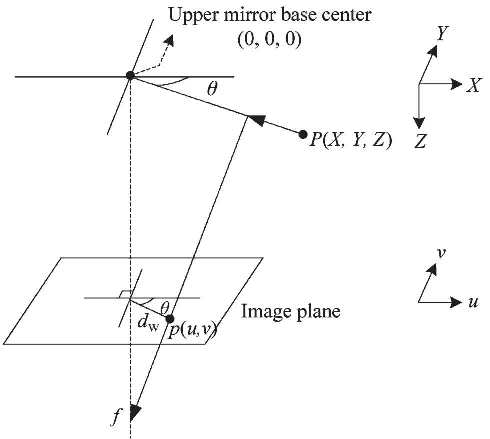


Fig. 3. Configuration of an upper omnicamera with a mirror.

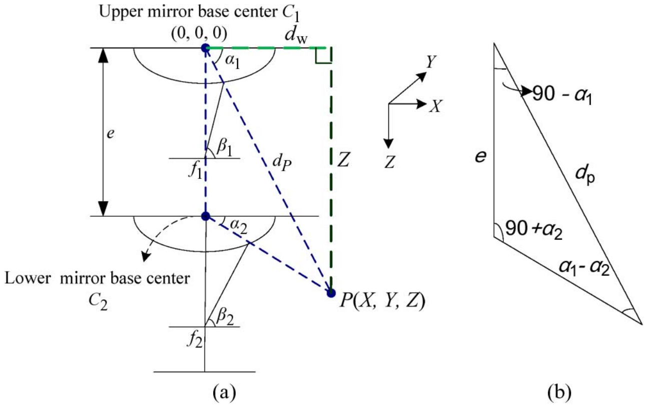


Fig. 4. Computing 3-D data. (a) Ray tracing of a real-world point P in an imaging device. (b) Blue triangle in (a) shown in detail.

$$r = f_{\mathrm{s}}(\alpha) = a_0 + a_1 \times \alpha^1 + a_2 \times \alpha^2 + \cdots + a_5 \times \alpha^5$$

where the coefficients a_0–a_5 are computed as follows using the *known* image coordinates (u_i, v_i) and the corresponding *known* world coordinates (X_i, Y_i, Z_i) of six real-world *landmark points* P_i that are manually selected in advance, where $i = 16$: 1) compute the radial distances r_i for each of the six points as $r_i = \sqrt{u_i^2 + v_i^2}$; 2) compute the elevation angle α_i for each of the six points as $\alpha_i = \tan^{-1}(Z_i/\sqrt{X_i^2 + Y_i^2})$; and 3) solve the following six simultaneous equations to get the values of a_0–a_5:

$$r_i = f_{\mathrm{s}}(\alpha_i) = a_0 + a_1 \times \alpha_i^1 + a_2 \times \alpha_i^2 + \cdots + a_5 \times \alpha_i^5$$

where $i = 1, 2, \ldots, 6$.

TABLE I
EXAMPLE OF A PANOMAPPING TABLE OF SIZE $M \times N$ [12]

	θ_1	θ_2	θ_3	θ_4	...	θ_M
α_1	(u_{11}, v_{11})	(u_{21}, v_{21})	(u_{31}, v_{31})	(u_{41}, v_{41})	...	(u_{M1}, v_{M1})
α_2	(u_{12}, v_{12})	(u_{22}, v_{22})	(u_{32}, v_{32})	(u_{42}, v_{42})	...	(u_{M2}, v_{M2})
α_3	(u_{13}, v_{13})	(u_{23}, v_{23})	(u_{33}, v_{33})	(u_{43}, v_{43})	...	(u_{M3}, v_{M3})
α_4	(u_{14}, v_{14})	(u_{24}, v_{24})	(u_{34}, v_{34})	(u_{44}, v_{44})	...	(u_{M4}, v_{M4})
...	...	...	...	...	...	...
α_N	(u_{1N}, v_{1N})	(u_{2N}, v_{2N})	(u_{3N}, v_{3N})	(u_{4N}, v_{4N})	...	(u_{MN}, v_{MN})

With a_0–a_5 derived, a panomapping table, as shown in Table I, is then constructed as follows. For each real-world point P_{ij} with azimuth-elevation angle pair (θ_i, α_j), compute the coordinates (u_{ij}, v_{ij}) of the corresponding image pixel p_{ij} as

$$u_{ij} = r_j \times \cos\theta_i, \quad v_{ij} = r_j \times \sin\theta_i$$

where $r_j = f_s(\alpha_j) = a_0 + a_1 \times \alpha_j^1 + a_2\alpha_j^2 + \cdots + a_5 \times \alpha_j^5$. After the table has been constructed, when an image pixel p with coordinates (u, v) in a given omni-image is given and checked by a table lookup to be located in entry E_{ij} with coordinates (u_{ij}, v_{ij}) in Table I, the azimuth–elevation angle pair of the corresponding real-world point P can be obtained to be (θ_i, α_j). Note that this azimuth–elevation angle pair only says that P is located on a light ray R with the azimuth direction θ_i and the elevation angle α_j; it does not sufficiently specify the 3-D position of P—any real-world point on light ray R will identically appear to be the pixel p in the image with the same coordinates (u_{ij}, v_{ij}). In addition, note that the panomapping table involves no camera parameter and is invariant in nature with respective to the camera position (i.e., it is not changed when the camera is moved around).

With regard to computing the 3-D coordinates for a real-world point P in the WCS, because two omnicameras are used in each imaging device, after the two image pixels p_1 and p_2 that correspond to a real-world point P have been identified in the two omni-images taken by the two cameras, two elevation angles α_1 and α_2, as shown in Fig. 4(a), can be obtained by the aforementioned table lookup process using the image coordinates (u_1, v_1) and (u_2, v_2) of p_1 and p_2, respectively. With the upper mirror base center being located at the WCS origin with world coordinates (0, 0, 0), the goal of 3-D data computation can be achieved using α_1 and α_2 to compute the world coordinates (X, Y, Z) of P. For this case, as shown in Fig. 4(b), by the law of sines in trigonometry, we have

$$\frac{d_P}{\sin(90^\circ + \alpha_2)} = \frac{e}{\sin(\alpha_1 - \alpha_2)} \tag{6}$$

where d_P is the distance between P and the base center of the upper mirror C_1, and e is the disparity between the two cameras in the omni-imaging device, which is manually measured in advance. Equation (6) may be reduced to get d_P as

$$d_P = \frac{1}{\cos\alpha_1} \times \frac{e}{\tan\alpha_1 - \tan\alpha_2}. \tag{7}$$

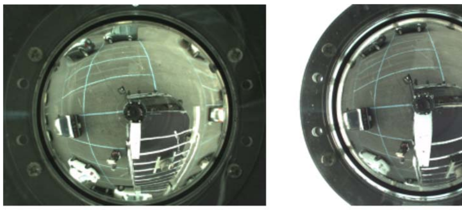

(a) (b)

Fig. 5. Background images taken by a two-camera imaging device. (a) Background taken by the upper camera. (b) Background taken by the lower camera.

Therefore, the horizontal distance d_w and the vertical distance Z of P, as shown in Fig. 4(a), may be computed, respectively, to be

$$\begin{aligned} d_\mathrm{w} &= d_P \cos\alpha_1 = \frac{e}{\tan\alpha_1 - \tan\alpha_2} \\ Z &= d_P \sin\alpha_1 = \frac{e \times \tan\alpha_1}{\tan\alpha_1 - \tan\alpha_2}. \end{aligned} \tag{8}$$

Furthermore, according to Fig. 3, the azimuth angle θ in the figure can be described in terms of the pixel coordinates (u, v) as follows:

$$\sin\theta = \frac{v}{\sqrt{u^2 + v^2}}, \quad \cos\theta = \frac{u}{\sqrt{u^2 + v^2}} \tag{9}$$

from which the value of θ in the ICS can be computed.

Now, according to the characteristic that the axes of the cameras are aligned with the axis of the mirror, the rotational invariance property of the omnicamera indicates that the azimuth angle of point P in the WCS and the azimuth angle of the corresponding pixel p in the ICS are identical [12]. Denoting both angles by θ, we can compute the values of X and Y in the WCS as

$$X = d_\mathrm{w} \times \cos\theta = \frac{e \times \cos\theta}{\tan\alpha_1 - \tan\alpha_2} \tag{10}$$

$$Y = d_\mathrm{w} \times \sin\theta = \frac{e \times \sin\theta}{\tan\alpha_1 - \tan\alpha_2}. \tag{11}$$

In summary, given a pair of matching image points that correspond to a real-world point P, we can compute the azimuth θ of P by (9) and obtain a pair of elevation angles α_1 and α_2 of P through a panomapping table lookup. Then, the world coordinates (X, Y, Z) that describe the unique 3-D position of P can *analytically* be computed by (8), (10) and (11).

III. AUTOMATIC DETECTION OF PASSING-BY PERSONS

A. Detection of Moving Objects in Omni-Images

Before extracting moving objects around a surveillance vehicle, *background images* without objects are captured in advance in the learning phase with the vehicle in a static state or, whenever necessary, in the patrolling phase with the vehicle also in a static state. One example is shown in Fig. 5. Then, *foreground images*, possibly with objects, are taken in real time in the patrolling phase. Both background and foreground images are color ones, which are transformed into grayscale

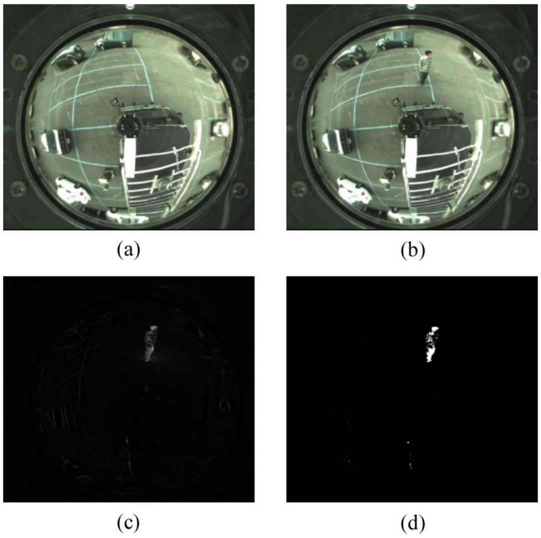


Fig. 6. Passing-by person detection. (a) Background image. (b) Foreground image. (c) Difference image obtained by subtracting (b) from (a). (d) Binary image obtained by bilevel thresholding.

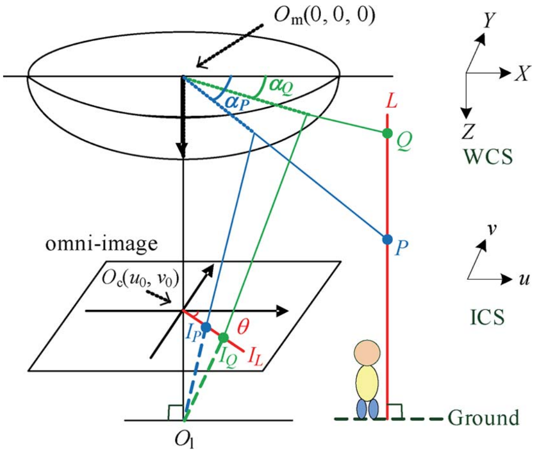


Fig. 7. Vertical-line property of omnicamera. The projection of a vertical line in the real world will result in a line in omni-image that goes through the image center.

images at the beginning of object detection. By subtracting the background image from the foreground image, a *difference image* is obtained. Because there usually exist noise pixels in the difference image, such as noise pixels that are caused by light variations, the difference image is thresholded next into a binary image to eliminate such noise pixels using the moment-preserving thresholding technique proposed by Tsai [13]. After these steps, we can obtain a bilevel image I_{bl}, with detected moving objects all labeled as black pixels. One example of the result is shown in Fig. 6.

B. Detection of a Passing-By Person's Head

As illustrated in Fig. 7, the proposed technique for detecting a passing-by person's head is based on a vertical-line property of the omni-image: Each line L, which is parallel to the Z-axis in the WCS and therefore vertical to the ground, is projected onto the image as a line I_L *that goes through the image center*. This case means that, if a passing-by person stands on the ground, the *midline* I_L of the person's silhouette will go through the image center O_c, as illustrated in Fig. 8(a). Therefore, we may find the top of each passing-by person's head through the following steps.

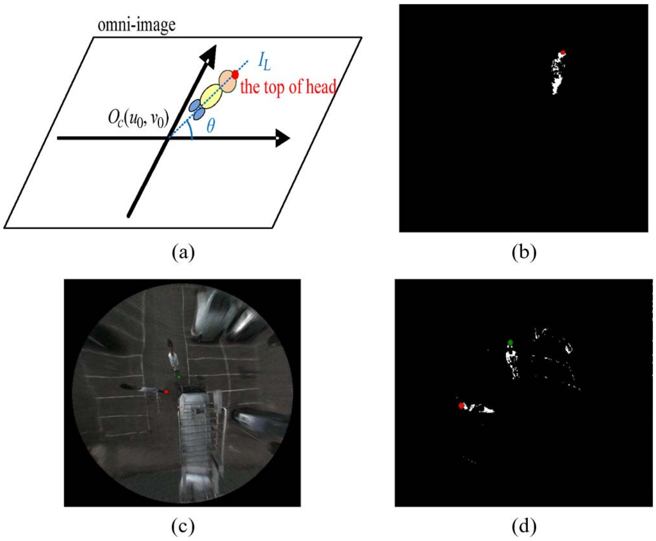


Fig. 8. Passing-by person's head detection. (a) Middle line of a person that goes through the image center. (b) Result of one-person detection in Fig. 6(d) (head marked in red). (c) and (d) Result of two-person detection (feet and heads marked in red and green, respectively).

1) Transform the rectangular image coordinates (u, v) of the bilevel difference image I_{bl} into polar images (θ, r), where θ is the azimuth angle, and r is the radial distance.
2) Scan each radial line inward from the image boundary to the image center based on the use of (θ, r).
3) Find each *sufficiently long* line segment in I_{bl}, and group neighboring segments found into a set L_{s}.
4) Find the farthest line-segment endpoint in L_{s} with respect to the image center as the passing-by person's head location.

If nonhuman objects will possibly appear in the omni-image, Step 3 should be refined using features such as the *width* of the set L_{s} (equal to the number of line segments in L_{s}, which refers to the width of the person) to differentiate and exclude them. The result of applying this process to Fig. 6 is shown in Fig. 8(b). Another result of detecting two persons is shown in Fig. 8(c) and (d).

C. Computation of a Passing-By Person's Location and Height

With the passing-by person's head location detected in the upper and lower images taken by an omni-imaging device, the formulas derived in Section II-C may be used to compute the location (X, Y) of the person and his/her body height h. In particular, the coordinates (X, Y) are only the coordinates described by (10) and (11). As can easily be figured out based on Figs. 4 and 7, the person's body height h may be computed as

$$h = d_{\mathrm{h}} - Z \tag{12}$$

where d_{h} is the height of the upper camera with respect to the ground, and Z is given as computed by (8), because P is the passing-by person's head point [shown as the red point in

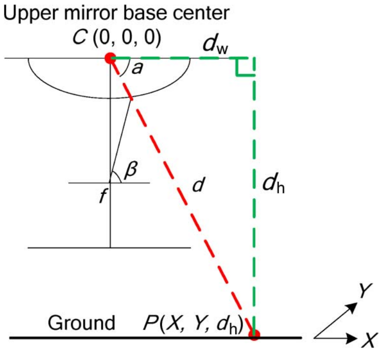


Fig. 9. Ray tracing of a real-world point P on the ground onto the mirror.

Fig. 8(b)]. Note that the origin of the WCS is located at the mirror base center of the upper omnicamera. In addition, by a back projection of the person's feet location at (X, Y, d_{h}) in the WCS into the omni-image, the person's feet can be marked for easier observation [for example, see the color points in Fig. 8(c)].

The detection of passing-by persons and the computation of their location and height features, as aforementioned, is based on the technique of background subtraction, which, by requiring background image updating whenever necessary, still has many applications with a video surveillance vehicle as the working platform, such as car driver identification, alert-region protection, car stealing prevention, passing-by person counting, and violent attack warming.

IV. Generation of Top- and Perspective-View Images

A. Construction of Top-View Images Using Upper Cameras

Because the field of view (FOV) of the upper camera in an imaging device is wider than that of the lower camera, we use the upper camera to construct a top-view image of the video surveillance vehicle's surrounding area. The height d_{h} of the camera that is affixed on the vehicle's roof is known in advance by manual measurement. In addition, for simplicity, it is assumed that all pixels around the vehicle in the image are projections of real-world points *on the ground*, as illustrated in Fig. 9.

We adopt a backward-mapping scheme based on the use of the panomapping table to compute the desired top-view image. First, according to Fig. 9, we may compute the horizontal distance d_{w} between a ground point P and the mirror base center C in terms of the coordinates (X, Y, d_{h}) of P in the WCS as follows:

$$d_{\mathrm{w}} = \sqrt{X^2 + Y^2}. \qquad (13)$$

Accordingly, the azimuth angle θ of P can be derived as

$$\theta = \cos^{-1}(X/d_{\mathrm{w}}) = \sin^{-1}(Y/d_{\mathrm{w}}). \qquad (14)$$

On the other hand, because $d_{\mathrm{w}} = d_{\mathrm{h}} \times \cot\alpha$, where d_{h} is the height of the upper camera's mirror base, we can compute the elevation angle α of P by

$$\alpha = \tan^{-1}(d_{\mathrm{h}}/d_{\mathrm{w}}). \qquad (15)$$

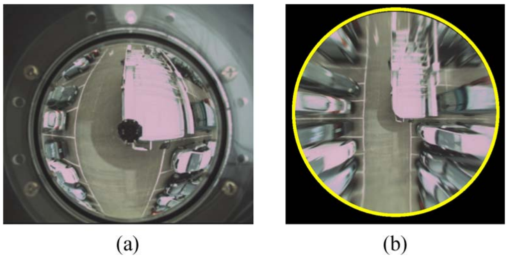


Fig. 10. Omni-image and its corresponding top-view images. (a) Omni-image. (b) Top-view image obtained from backward mapping.

Furthermore, using the aforementioned radial stretching function $r = f_{\mathrm{s}}(\alpha)$, the radial distance r that corresponds to α can be derived. Accordingly, by the rotational-invariance property of omni-imaging, the coordinates (u, v) of the image pixel p that corresponds to P can be obtained from (14) as

$$u = r\cos\theta, \quad v = r\sin\theta. \qquad (16)$$

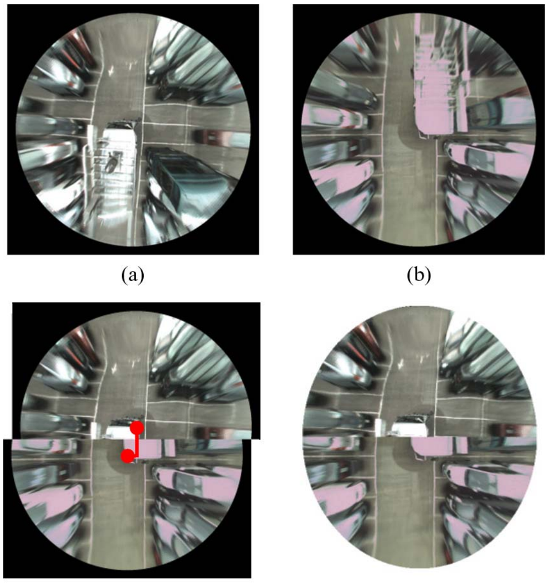
As a result, a complete top-view image can be computed. A top-view image of a parking area that is computed with the surveillance vehicle in the middle is shown in Fig. 10. Note that all the cars that appear in the figure are distorted and that this phenomenon comes from their violation of the assumption that the processed points are projections of real-world points *on the ground*, as aforementioned. However, the result is still good for visual inspection of the vehicle's surround.

B. Merging of Two Top-View Images Into a Single Image

As aforementioned, we can construct two top-view images I_{t1} and I_{t2} using the omni-images taken from the two upper cameras, as shown in Fig. 11(a) and (b). It is assumed that the relative position of the two upper cameras on the vehicle roof are manually measured in advance and described by a pair of offsets $(C_{\mathrm{W}}, C_{\mathrm{L}})$, with C_{W} and C_{L} being the horizontal and vertical distances between the two imaging devices, respectively. It is desired to merge the two top-view images I_{t1} and I_{t2} into a single image, denoted as I_{int}, to get a wider FOV of the vehicle's surround. To do this work, we first divide the FOV that is covered by the desired top-view image I_{int} into the following two parts: one part from the upper half of I_{t1} and the other part from the lower half of I_{t2}. All the pixels in the latter part are then shifted for the offset $(C_{\mathrm{W}}, C_{\mathrm{L}})$ before being merged into the former part. An example of a top-view image that is obtained is shown in Fig. 11(c). To obtain a *better looking* top-view image, an elliptical shape is further used as a *viewing window*, and all the pixels outside the window are discarded, yielding an elliptical-shaped image. For Fig. 11(c), the result of this process is shown in Fig. 11(d).

C. Superimposition of a Surveillance Vehicle Shape on a Top-View Image

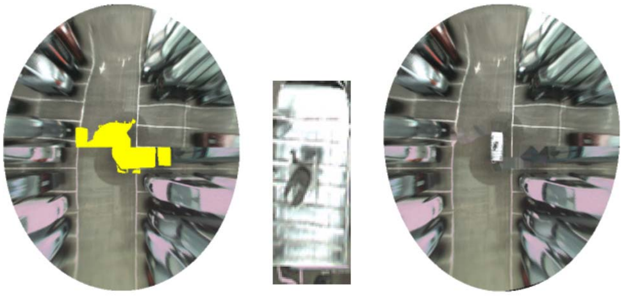
Because the imaging devices are affixed on the vehicle's roof, the vehicle shape in the top-view images always appears at *fixed* locations in the omni-images taken. Therefore, it is feasible to superimpose a *real surveillance vehicle shape* S_{rv} on the elliptical-shaped top-view image similar to Fig. 11(d)



(a) (b)

(c) (d)

Fig. 11. Top-view image construction. (a) Right-front top-view image. (b) Left-rear top-view image. (c) Integrated top-view image (red dots refer to the center of the original images). (d) Top-view image viewed in an elliptical shape.



(a) (b) (c)

Fig. 12. Superimposing real surveillance vehicle shape onto a top-view image. (a) Top-view image with a vehicle shape marked yellow. (b) Presegmented real shape of a video surveillance vehicle. (c) Result of the ground texture filling and superimposition of a real surveillance vehicle shape.

as the *central landmark* of the image, making observation of the vehicle's position in the surround more convenient. For this aim, the following steps are conducted.

1) Trace and segment out manually the real top-view shapes S_1 and S_2 of the surveillance vehicle in the two top-view images $I^{\mathrm{o}}_{\mathrm{t1}}$ and $I^{\mathrm{o}}_{\mathrm{t2}}$, respectively, in the learning phase.
2) Superimpose S_1 and S_2, respectively, on I_{t1} and I_{t2} acquired in each patrolling phase to segment out by image matching and mark yellow the surveillance vehicle's shape area A in I_{t1} and I_{t2} before they are merged into one, resulting in a figure similar to Fig. 12(a).
3) Apply a texture synthesis scheme to fill ground texture into the marked surveillance vehicle shape area A.
4) Superimpose a presegmented real top-view vehicle shape S_{rv} as shown in Fig. 12(b) on the resulting image of the last step, yielding a *better looking* output image, as shown in Fig. 12(c).

Note that S_{rv} is obtained in the learning phase by manual segmentation from a top-view image, as shown in Fig. 11(a).

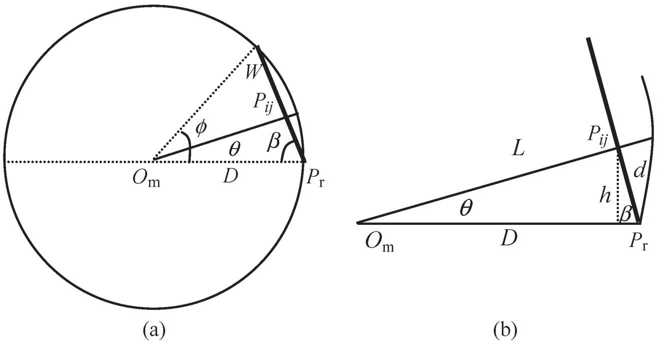


Fig. 13. Top view of a generated perspective-view image.

Step 3 is carried out as follows. For each pixel p_c in the surveillance vehicle shape A, find the pixel p_n that is outside A and closest to p_c, and use the color of p_n to fill up p_c. In addition, the relations of pixels in A with the pixels outside A are kept in a table for use in the later top-view image generation process to improve the top-view image display speed.

D. Generation of Perspective-View Images

The construction of a perspective-view image I_{P} from an omni-image I_{o} is based on the use of the panomapping table [12]. The major steps are described as follows: 1) Map each pixel p in the desired perspective-view image I_{P} at coordinates (k, l) to a pair of elevation and azimuth angles (α, θ) in the panomapping table according to the geometry of the desired perspective transformation; 2) find the image coordinates (u, v) in the table entry that corresponds to (α, θ); and 3) assign the color channel values of pixel p in I_{P} to be the values of the pixel at coordinates (u, v) in I_{o}. The detail of mapping (k, l) to (α, θ) in Step 1 is described as follows.

Assume that the perspective-view image I_{p} that we want to generate from I_{o} is at a distance D to the mirror base center O_{m} and has $M_{\mathrm{p}} \times N_{\mathrm{p}}$ pixels. In addition, assume that I_{p} is the image of a planar rectangular region A_{P} with size $W \times H$ and is perpendicular to the floor in the real-world space, as illustrated from the top view in Fig. 13.

a) Computing the Azimuth Angle θ: The angle ϕ that is spanned by the width W of I_{p}, as shown in Fig. 13(a), may be derived by the law of cosines to satisfy the equality $W^2 = D^2 + D^2 - 2 \times D \times D \times \cos\phi$ so that ϕ may be computed as $\phi = \cos^{-1}[1 - W^2/(2D^2)]$. In addition, the angle β in the figure is just $\beta = (\pi - \phi)/2$. Next, let P_{ij} denote the intersection point of the light ray R_{P} that is projected onto the image pixel p and the aforementioned planar region A_{P}. Then, we may compute the distance d between point P_{ij} and the border point P_r shown in Fig. 13(b) by linear proportionality to be $d = k \times W/M_{\mathrm{P}}$, because A_{P} has a width of W, I_{p} has a width of M_{p} pixels, and pixel p has the coordinate k in the horizontal direction.

Furthermore, by the law of cosines, again, the distance L between point P_{ij} and the mirror base center O_{m}, as shown in Fig. 13(b), satisfies the equality $L^2 = D^2 + d^2 - 2 \times d \times D \times \cos\beta$. In addition, the distance h from point P_{ij} to line segment $O_{\mathrm{m}}P_r$ may be computed as $h = d \times \sin\beta$. Finally, the azimuth angle θ of point P_{ij} with respect to $O_m P_{\mathrm{r}}$ satisfies

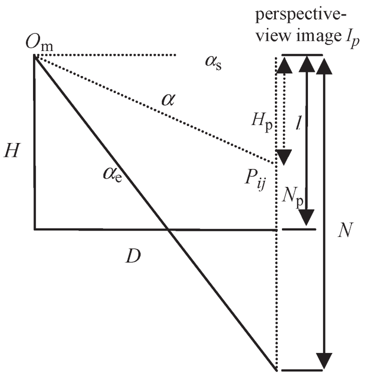


Fig. 14. Lateral configuration for generating a perspective-view image.

$\sin\theta = h/L$, which, by the previously derived equalities, leads to the following desired value:

$$\theta=\sin^{-1}(h/L)=\sin^{-1}\left[\frac{d\times\sin\beta}{\sqrt{D^2+d^2-2\times d\times D\times\cos\beta}}\right]. \tag{17}$$

b) Computing the Elevation Angle: An illustration of the involved imaging configuration for computing the elevation angle α from a lateral view is shown in Fig. 14. The height of region A_P is H and image I_p is divided into N_p intervals in the vertical direction. Therefore, by linear proportionality, again, the height of P_{ij} may be computed to be $H_\mathrm{p} = (l \times H)/N_\mathrm{p}$, where l is the coordinate of pixel p in the vertical direction. Finally, by trigonometry, the desired elevation angle α may be derived to be

$$\alpha = \tan^{-1}(H_\mathrm{p}/L). \tag{18}$$

This completes the derivations of the azimuth–elevation angle pair (θ, α) conducted in Step 1. Note that, in these derivations, the start direction (specified by the line segment $O_\mathrm{m}P_r$) of the angle ϕ spanned by the width W of I_p, as shown in Fig. 14, coincides with the horizontal direction 0°, resulting in the azimuth angle θ. Of course, perspective-view images for other azimuth angles may also be generated. A convenient scheme to do this step is described next.

E. Generation of Perspective-View Images Specified With Mouse Clicks or Panel Touch

In this paper, a technique is proposed to enable a user to conveniently change the view direction of the perspective-view image by mouse clicking or, equivalently, by panel touching. Fig. 15(a) shows a perspective-view image that is generated from the omni-image shown in Fig. 15(b). According to the previous derivations and an observation of the images in the figure, we can see that there exists a relation between the mouse motion direction and the viewing direction of the generated perspective-view image. In particular, the horizontal motion of a mouse specified by its horizontal location M_x may be used to define the azimuth angle θ of the space point P_c that corresponds to the image center p_c. Similarly, the vertical motion of the mouse specified by its vertical location M_y may be used to define the elevation angle α of P_c.

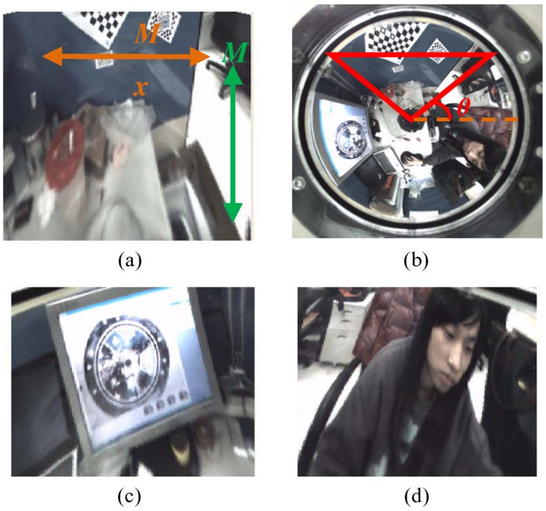


Fig. 15. Corresponding omni-image and perspective-view image. (a) Perspective-view image. (b) Omni-image part (roughly enclosed by the red triangle) from which (a) was generated. (c) and (d) Two other perspective-view images generated from (b) by mouse clicking on (a).

Accordingly, as an interaction with the proposed system through mouse clicking on the computer screen, which shows a perspective-view image as shown in Fig. 15(a), we define in this paper an angle value θ_mouse according to the mouse click location in the horizontal direction so that θ_mouse becomes larger gradually as the mouse moves from right to left in the image in Fig. 15(a). Then, we modify the equation for computing the azimuth angle, i.e., (17), which is previously derived to be

$$\theta = \sin^{-1}(h/L) + \theta_\mathrm{mouse}. \tag{19}$$

Similarly, we define another angle value α_mouse according to the mouse click location in the vertical direction such that α_mouse gradually increases as the mouse moves from top to bottom, as shown in Fig. 15(a). Then, we modify (18) to be

$$\alpha = \tan^{-1}(H_\mathrm{p}/L) + \alpha_\mathrm{mouse}. \tag{20}$$

The user interface becomes friendlier after adding the two variables θ_mouse and α_mouse, and a user of the surveillance vehicle can now conveniently choose any view direction by mouse clicking (or panel touching) to observe the corresponding perspective-view image of a scene of his/her interest. Two more experimental results of perspective-view images that are generated this way are shown in Fig. 15(c) and (d).

V. Experimental Results and Discussions

A. Experimental Results of 3-D Data Computation

At first, we conducted an experiment to test the precision of the computed 3-D data by computing the positions of some real-world points and compared them with manually obtained positions. We laid an omni-imaging device on the ground and took images of a calibration pattern with alternating black and white grids and some black cross shapes, as shown in Fig. 16. The width between every two consecutive grids on the pattern is 5 cm, and the bigger width between every two crosses is

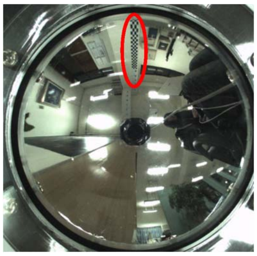


Fig. 16. Omni-image of a calibration pattern with black and white grids and black cross shapes on a wall taken by an omnicamera laid on the ground.

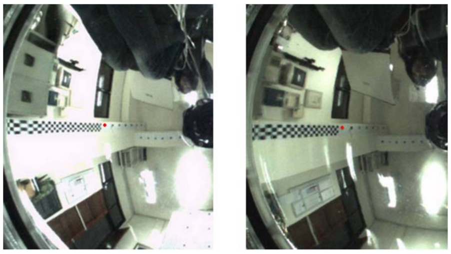


Fig. 17. Picked-out pairs of corresponding pixels in two omni-images (marked by two red dots).

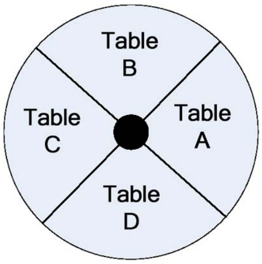


Fig. 18. Four image regions that correspond to four radial stretching functions used in constructing panomapping tables.

25 cm. We picked out 15 pairs of corresponding pixels from these grids and shapes in the two images taken. One example is shown in Fig. 17. We then calculated, by (8) in Section II-C, the horizontal distance d_{w} and the height Z of the real-world point that corresponds to each pair of corresponding image pixels. In addition, to increase the computation accuracy, we used *four*, instead of only one, radial stretching functions in the panomapping process, each function dealing with a quarter of the omni-image, as shown in Fig. 18. This measure is particularly necessary when the assumption of radial symmetry of the aforementioned mirror surface is *not* valid, which happens to be the case encountered in this study because of the imperfect manufacturing of the mirrors.

The results of calculations of the 3-D data d_{w} and Z of 15 selected landmark points on the calibration pattern using the images taken are shown in Tables II and III, which resulted, respectively, from the use of the panomapping tables in Fig. 18, Tables A and B. As shown, the differences between the measured and the computed data are all small, resulting in small root mean-square error (RMSE) values (each value defined to be the square root of the average of all the difference values). All the error rates (each rate defined to be the ratio of the RMSE over the average measured data value) are shown to be about 5%, which are good for practical applications.

TABLE II
STATISTICS OF THE COMPUTED 3-D DATA USING PANOMAPPING TABLE A

Point No.	Azimuth angle (°)	Elevation angle (°)	(1) Measured 3D real-world point data (cm)		(2) Computed 3D real-world point data (cm)		(3) Differences (1) – (2) (cm)	
	θ	α	d_{w}	Z	d_{w}	Z	d_{w}	Z
1	0.00	54.95	249	355	250.2	355.4	-1.20	-0.40
2	0.00	52.96	249	330	251.4	331.4	-2.40	-1.40
3	0.00	50.77	249	305	255	310.2	-6.00	-5.20
4	0.00	48.35	249	280	260.6	290.7	-11.60	-10.70
5	0.00	45.68	249	255	247.6	253.6	1.40	1.40
6	0.00	42.73	249	230	257.7	237.2	-8.70	-7.20
7	0.00	39.46	249	205	236.6	197	12.40	8.00
8	0.00	37.35	249	190	227.7	175	21.30	15.00
9	0.00	34.32	249	170	223	154.1	26.00	15.90
10	0.00	31.07	249	150	231.8	140.5	17.20	9.50
11	0.00	27.57	249	130	238	124.2	11.00	5.80
12	10.69	54.45	227.96	319	219.5	300.2	8.46	18.80
13	12.00	57.47	203.45	319	185.5	286.7	17.95	32.30
14	15.85	64.10	154.89	319	156.1	323.3	-1.21	-4.30
15	18.84	67.67	131.02	319	130.5	321.2	0.51	-2.20
Average			230.42	258.4				
RMSE							12.52	12.336
Error rate = RMSE / Average							5.43%	4.77%
Correlation coefficient	Azimuth angles						-0.03	0.17
	Elevation angle						-0.48	-0.18

TABLE III
STATISTICS OF THE COMPUTED 3-D DATA USING PANOMAPPING TABLE B

Point No.	Azimuth angle (°)	Elevation angle (°)	(1) Measured 3D space point data (cm)		(2) Computed 3D space point data (cm)		(3) Differences (1) – (2) (cm)	
	θ	α	d_{w}	Z	d_{w}	Z	d_{w}	Z
1	90.00	54.95	249	355	249	355.2	0.00	-0.20
2	90.00	52.96	249	330	251.4	331.4	-2.40	-1.40
3	90.00	50.77	249	305	255	310.2	-6.00	-5.20
4	90.00	48.35	249	280	260.6	290.7	-11.60	-10.70
5	90.00	45.68	249	255	247.6	253.6	1.40	1.40
6	90.00	42.73	249	230	257.7	237.2	-8.70	-7.20
7	90.00	39.46	249	205	236.6	197	12.40	8.00
8	90.00	37.35	249	190	231.3	178.4	17.70	11.60
9	90.00	34.32	249	170	232.8	160.8	16.20	9.20
10	90.00	31.07	249	150	237.3	143.2	11.70	6.80
11	90.00	27.57	249	130	238	124.2	11.00	5.80
12	100.69	54.45	227.96	319	219.7	303.9	8.26	15.10
13	102.00	57.47	203.45	319	185.5	286.7	17.95	32.30
14	105.85	64.10	154.89	319	156.1	323.3	-1.21	-4.30
15	108.84	67.67	131.02	319	130.5	321.2	0.52	-2.20
Average			230.42	258.4				
RMSE							10.436	11.121
Error rate = RMSE / Average							4.53%	4.30%
Correlation coefficient	Azimuth angles						0.04	0.21
	Elevation angle						-0.41	-0.12

We also computed the values of the correlation coefficient between the difference values and the azimuth/elevation angles, respectively, to see the affections of such angles to the 3-D computation results. The correlation coefficient values computed show that only the variations of the elevation angle have medium influences (with correlation coefficient values of -0.48 and -0.41) on the precisions of the computed location values (d_{w}) of the real-world points.

B. Experimental Results of Passing-By Person Detection

We have also tested the technique that we proposed for passing-by person detection and localization. The environment for this experiment is an open space in a parking area. Because of the property of imaging projection, after the region of a passing-by person has been found in the image, the body point that is farthest to the center of the image is located as the position of the person's head, as mentioned previously and shown in Figs. 6 and 8. In this experiment, we processed an

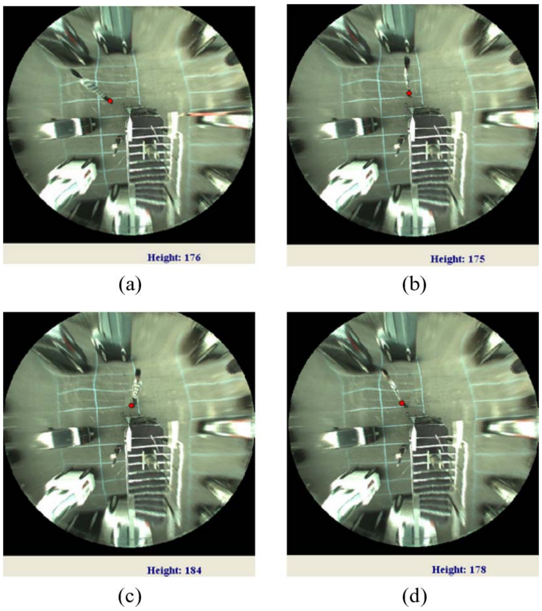


Fig. 19. Examples of passing-by person detection. (a)–(d) Detection results shown as top-view images with red points used to mark a detected person's feet.

image sequence with a person who was walking around the video surveillance vehicle, and the person was successfully detected in the image frames. A sequence of top-view images was constructed, and the locations of the person were correctly computed. Some results are shown in Fig. 19, in which the red points are used to mark the detected person's foot positions.

C. Experimental Results of the Integration of System Functions

A system that integrates the proposed techniques has been implemented, including the following functions: 1) the construction of a top-view image of the surveillance vehicle's surrounding area; 2) the detection of passing-by persons around the vehicle; and 3) the generation of a perspective-view image whose view direction may be specified by mouse clicking or is automatically determined to show the detected person. Each camera of model Artcam-200MI that was used in the proposed system takes 0.1 s to acquire a 1600 × 1200 image, and the processing cycle time of the system is 0.2 s (including person detection and top- and perspective-view image generation). The height of the lower cameras, when affixed to the surveillance vehicle, is 231.8 cm, and the height of the higher cameras is 256 cm. Two examples of the experimental results using the system are shown in Fig. 20.

D. Further Survey of Existing Methods and Comparisons

We conducted a survey of existing methods and compared our proposed technique with these methods, the results of which are given as follows.

1) *Survey about catadioptric optics*. Baker and Nayar [15] derived the class of single-lens single-mirror catadioptric sensors that have single viewpoints. Geyer and Daniilidis [16] proposed a unifying model for the projective geometry induced by catadioptric sensors. Although good theories can be found in these papers, we tried to design an omnicamera with suitable parameters to *fit* the roof structure of a vehicle used in this study.

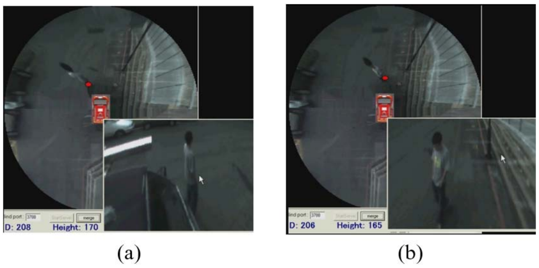


Fig. 20. Examples of generated images, with the right lower corner in each image showing the generated perspective-view image of a detected person.

2) *Survey about catadioptric stereo*. Xiong *et al.* [17] described various ways of designing omnidirectional stereo vision systems and related techniques for image unwarping, stereo matching, and moving object detection. Su *et al.* [18] omnidirectionally obtained obstacle information by an omnidirectional stereo vision system with a perspective camera coupled with two hyperbolic mirrors. Differently, the method proposed in this paper combines, in a *novel* way, the panomapping technique [12] and the use of a two-camera omni-imaging device for 3-D data computation and fast top- and perspective-view image generation.

3) *Survey about object detection on moving vehicles*. Gandhi and Trivedi [3] used images of two omnicameras that are affixed to car side mirrors to synthesize perspective-view images and detect in-front vehicles by binocular stereo and lateral vehicles by motion stereo. They also used parametric ego-motion compensation for an omnidirectional vision sensor to detect surrounding events [19], [20]. The method proposed in this paper emphasizes the 3-D detection of both passing-by persons' locations and heights, *directly* using omni-images without generating perspective views to speed up the detection. This approach is made possible by the use of the designed two-camera omni-imaging devices and the *single* panomapping technique.

4) *Survey about top-view image generation*. Ehlgen and Pajdla [21], [22] installed two omnicameras on the side mirrors of a truck to generate a bird's-eye-view image that covers the *frontal* scene. The top-view image generated in this study, instead, covers the *entire* vehicle surroundings. Liu *et al.* [23] affixed six fisheye cameras around a car to generate a bird's-eye-view image of the surroundings for driving assistance. No 3-D data computation was considered, in contrast with our method, which computes 3-D feature point data.

5) *Survey about human detection*. Liu *et al.* [24] detected human motion with an omnicamera on a mobile robot using ego-motion compensation and temporal differencing *after* unwarping omni-images, whereas the method that is proposed of this paper *directly* detects passing-by persons from omni-images. Ng *et al.* [25] used multiple omnivision sensors to synthesize perspective views and track human activities using N-ocular stereo techniques without

TABLE IV
COMPARISON OF EXISTING METHODS (O: YES; X: NO; AND ***: USING A PANOMAPPING TABLE)

	omni-image unwarping	Top view generation	Integration of multiple images	Real-time processing	Applied in outdoor environment	Passing-by people detection	Object distance estimation	Object height estimation
Our method	O	O	O	O	O	O	O	O
Su et al. [18]	O	X	X	O	X	X	O	X
Gandhi & Trivedi [3]	O	X	X	O	O	X	O	X
Gandhi & Trivedi [20]	X	O	X	O	O	X	O	X
Ehlgen & Pajdla [21, 22]	X	O	O	O	O	X	X	X
Liu, et al. [23]	O	O	O	O	O	X	X	X
Liu et al. [24]	O	X	X	O	X	O	X	X
Ng, et al. [25]	O	X	X	O	X	O	O	X
Huang et al. [26]	O	X	X	O	O	O	X	X
Kawasaki et al. [27]	O	X	X	X	O	X	O	O

acquiring range information. In contrast, our method detects human beings in vehicle surroundings and computes their 3-D features (i.e., location and height) for various applications.

6) *Survey about perspective-view image generation*. Huang *et al.* [26] presented an in-car omni-imaging system that processes acquired videos to obtain *direction-fixed* virtual perspective views on the driver, passengers, and frontal scenes using pan, tilt, and zoom parameters, in contrast with our method, which generates *direction-changing* perspective views on passing-by persons without using camera parameters. Ng *et al.* [25] generated images of both a walking person's view and an observer's view using a *range-space search technique*, whereas our method uses the *panomapping approach*. Kawasaki *et al.* [27] obtained 3-D information by the spatiotemporal analysis of omni-images using calibrated camera parameters, again contrasting with the tabular panomapping technique used by our method, involving no camera parameter.

7) *Comparisons about implemented functions*. The result of a more detailed comparison of the aforementioned methods (except for [17], which is a survey paper) with the proposed method in terms of a set of implemented functions is listed in Table IV, which shows that the proposed method has integrated more functions than the other methods.

VI. CONCLUSION

A pair of two-camera omni-imaging devices has properly been designed for use on the top of a video surveillance vehicle to monitor passing-by persons. The two devices are efficiently affixed to the right-front and left-rear of the vehicle roof to facilitate the generation of a top-view image that covers the vehicle surrounding area. A new 3-D data computation technique based on the panomapping concept and the rotational invariance property of the omni-image has been proposed. Because of the use of table lookup and analytic computation formulas, the technique can be implemented to satisfy real-time applications. Passing-by persons who appear around the surveillance vehicle can automatically be detected using an omni-image property about upright objects, with their location and body height computed by the proposed 3-D data acquisition technique. A top-view image of the vehicle's surrounding area with a real vehicle shape that is properly inserted in the middle is generated by registering two omni-images that are taken by two upper cameras. Perspective-view images that cover the detected person or any interesting scene spot can also automatically be generated in real time or by mouse clicking for convenient inspection. The system is useful for several security-monitoring applications around the video surveillance vehicle. The experimental results show the feasibility and precision of the proposed method for practical applications.

REFERENCES

[1] C. Regazzoni, V. Ramesh, and G. Foresti, "Special issue on video communications, processing, and understanding for third-generation surveillance systems," *Proc. IEEE*, vol. 89, no. 10, pp. 1355–1539, Oct. 2001.

[2] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Syst., Man, Cybern. C: Appl. Rev.*, vol. 34, no. 3, pp. 334–352, Aug. 2004.

[3] T. Gandhi and M. M. Trivedi, "Vehicle surround capture: Survey of techniques and a novel omnivideo-based approach for dynamic panoramic surround maps," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 3, pp. 293–308, Sep. 2006.

[4] C. Micheloni, G. L. Foresti, C. Piciarelli, and L. Cinque, "An autonomous vehicle for video surveillance of indoor environments," *IEEE Trans. Veh. Technol.*, vol. 56, no. 2, pp. 487–498, Mar. 2007.

[5] K. C. Chen and W. H. Tsai, "Guidance of indoor vision-based autonomous vehicles for security patrolling by a SIFT-based vehicle localization technique using along-path object information," *IEEE Trans. Veh. Technol.*, vol. 59, no. 7, pp. 3261–3271, Sep. 2010.

[6] Y. Onoe, N. Yokoya, K. Yamazawa, and H. Takemura, "Visual surveillance and monitoring system using an omnidirectional video camera," in *Proc. Int. Conf. Pattern Recog.*, Brisbane, Australia, Aug. 16–20, 1998, vol. 1, pp. 588–592.

[7] T. Mituyosi, Y. Yagi, and M. Yachida, "Real-time human feature acquisition and human tracking by omnidirectional image sensor," in *Proc. IEEE Int. Conf. Multisens. Fusion Integr. Intell. Syst.*, Tokyo, Japan, Jul. 30–Aug. 1, 2003, pp. 258–263.

[8] S. Morita, K. Yamazawa, and N. Yokoya, "Networked video surveillance using multiple omnidirectional cameras," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom.*, Kobe, Japan, Jul. 16–20, 2003, vol. 3, pp. 1245–1250.

[9] H. Koyasu, J. Miura, and Y. Shirai, "Real-time omnidirectional stereo for obstacle detection and tracking in dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Maui, HI, Oct./Nov. 2001, vol. 1, pp. 31–36.

[10] H. Ukida, N. Yamato, Y. Tanimoto, T. Sano, and H. Yamamoto, "Omnidirectional 3-D measurement by hyperbolic mirror cameras and pattern projection," in *Proc. IEEE Conf. Instrum. Meas. Technol.*, Victoria, BC, Canada, 2008, pp. 365–370.

[11] J. I. Meguro, J. I. Takiguchi, Y. Amano, and T. Hashizume, “3-D reconstruction using multibaseline omnidirectional motion stereo based on GPS/DR compound navigation system,” *Int. J. Robot. Res.*, vol. 26, no. 6, pp. 625–636, Jun. 2007.

[12] S. W. Jeng and W. H. Tsai, “Using panomapping tables for unwarping of omni-images into panoramic- and perspective-view images,” *J. IET Imag. Process.*, vol. 1, no. 2, pp. 149–155, Jun. 2007.

[13] W. H. Tsai, “Moment-preserving thresholding: A new approach,” *Comput. Vis. Graph. Image Process.*, vol. 29, no. 3, pp. 377–393, Mar. 1985.

[14] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[15] S. Baker and S. Nayar, “A theory of single-viewpoint catadioptric image formation,” *Int. J. Comput. Vis.*, vol. 35, no. 2, pp. 175–196, Nov./Dec. 1999.

[16] C. Geyer and K. Daniilidis, “Catadioptric projective geometry,” *Int. J. Comput. Vis.*, vol. 45, no. 3, pp. 223–243, Dec. 2001.

[17] Z. Xiong, W. Chen, and M. J. Zhang, “Catadioptric omnidirectional stereo vision and its applications in moving objects detection,” in *Computer Vision*, Z. Xiong, Ed. Rijeka, Croatia: InTech, Nov. 2008, pp. 493–518.

[18] L. C. Su, C. J. Luo, and F. Zhu, “Obtaining obstacle information by an omnidirectional stereo vision system,” in *Proc. IEEE Int. Conf. Inf. Acquisition*, Shandong, China, Aug. 20–23, 2006, pp. 48–52.

[19] T. Gandhi and M. M. Trivedi, “Motion analysis for event detection and tracking with a mobile omnidirectional camera,” *Multimedia Syst. J.*, vol. 10, no. 2, pp. 131–143, Aug. 2004.

[20] T. Gandhi and M. Trivedi, “Parametric ego-motion estimation for vehicle surround analysis using an omnidirectional camera,” *Mach. Vis. Appl.*, vol. 16, no. 2, pp. 85–95, Feb. 2005.

[21] T. Ehlgen and T. Pajdla, “Maneuvering aid for large vehicle using omnidirectional cameras,” in *Proc. 8th IEEE Workshop Appl. Comput. Vis.*, Austin, TX, Feb. 2007, p. 17.

[22] T. Ehlgen and T. Pajdla, “Monitoring surrounding areas of truck–trailer combinations,” in *Proc. 5th Int. Conf. Comput. Vis. Syst.*, Bielefeld, Germany, Mar. 2007, pp. 21–24.

[23] Y. C. Liu, K. Y. Lin, and Y. S. Chen, “Bird’s-eye-view vision system for vehicle surrounding monitoring,” in *Proc. Conf. Robot Vis.*, Berlin, Germany, Feb. 2008, pp. 207–218.

[24] H. Liu, N. Dong, and H. Zha, “Omnidirectional-vision-based human motion detection for autonomous mobile robots,” in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2005, pp. 2236–2241.

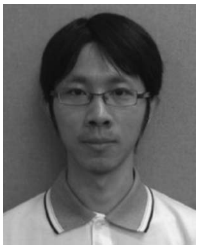
[25] K. C. Ng, H. Ishiguro, M. Trivedi, and T. Sogo, “An integrated surveillance system—Human tracking and view synthesis using multiple omnidirectional vision sensors,” *Image Vis. Comput.*, vol. 22, no. 7, pp. 551–561, Jul. 2004.

[26] K. S. Huang, M. M. Trivedi, and T. Gandhi, “Driver’s view and vehicle surround estimation using omnidirectional video stream,” in *Proc. IEEE Intell. Veh. Symp.*, Columbus, OH, Jun. 2003, pp. 444–449.

[27] H. Kawasaki, K. Ikeuchi, and M. Sakauchi, “Spatiotemporal analysis of omni-images,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Hilton Head Island, SC, Jun. 2000, pp. 577–584.

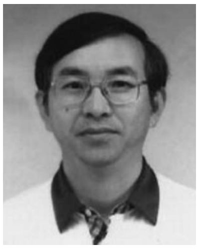
Pei-Hsuan Yuan received the B.S. degree in computer science and information engineering from the National Chi Nan University, Nantou City, Taiwan, in 2008 and the M.S. degree in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 2010.

From August 2008 to July 2010, she was a Research Assistant with the Computer Vision Laboratory, Department of Computer Science, National Chiao Tung University. She is currently with the Taiwan Semiconductor Manufacturing Company, Hsinchu. Her research interests include computer vision, image processing, video surveillance, and their applications.



Kuo-Feng Yang received the B.S. degree in mathematics from the National Chung Hsing University, Taichung, Taiwan, in 2006 and the M.S. degree in computer science and information engineering from Yuan Ze University, Jhongli City, Taiwan, in 2008. He is currently working toward the Ph.D. degree with the Institute of Computer Science and Engineering, National Chiao Tung University, Hsinchu, Taiwan.

Since August 2008, he has been a Research Assistant with the Computer Vision Laboratory, Department of Computer Science, National Chiao Tung University. His research interests include image and vision computing and autonomous vehicle applications.



Wen-Hsiang Tsai (SM’91) received the B.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, in 1973, the M.S. degree in electrical engineering from Brown University, Providence, RI, in 1977, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, in 1979.

Since 1979, he has been with the National Chiao Tung University (NCTU), Hsinchu, Taiwan, where he has served as the Head of the Department of Computer Science, the Dean of General Affairs, the Dean of Academic Affairs, and a Vice President. From 2004 to 2007, he was the President of Asia University, Taichung, Taiwan. He is currently a Chair Professor of Computer Science with the Department of Computer Science, NCTU. He has been an Editor or the Editor-in-Chief of several international journals, including *Pattern Recognition*, the *International Journal of Pattern Recognition and Artificial Intelligence*, and the *Journal of Information Science and Engineering*. He has published 144 journal papers and 227 conference proceedings. His research interests include computer vision, information security, video surveillance, and autonomous vehicle applications.

Dr. Tsai is a Life Member of the Chinese Pattern Recognition and Image Processing Society in Taiwan. He was the Chair of the Chinese Image Processing and Pattern Recognition Society of Taiwan from 1999 to 2000 and the IEEE Computer Society Taipei Section from 2004 to 2008. He has received several awards, including the Annual Paper Award from the Pattern Recognition Society, the Academic Award from the Ministry of Education, Taiwan, the Outstanding Research Award from the National Science Council of Taiwan, the ISI Citation Classic Award from Thomson Scientific, and more than 40 other academic paper awards from various academic societies.