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 3D vision-based techniques for real-time security surveillance around the vehicle are proposed, which may be used for monitoring passing-by persons around the vehicle. First, the design of the pair of two-camera omni-imaging devices, each device consisting of two omni-cameras with their optical axes vertically aligned, is described. Then, a new analytic technique for fast 3D space data acquisition based on a pano-mapping method for image-to-world space transformation and the rotational invariance property of the omni-image is proposed. Also proposed are techniques for constructing top-view and perspective-view images for convenient observation of the monitored environment. Finally, 3D vision techniques for detecting automatically passing-by persons and computing their locations and body heights are proposed, followed by experimental results showing the precision and feasibility of the proposed techniques.

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

---

I. INTRODUCTION

Video surveillance of environments has been studied intensively in recent years [1, 2]. In order to increase the mobility of the video surveillance system, vehicles are used as the carriers of such systems [3, 4, 5]. Applications of video surveillance vehicles include dynamic monitoring of outdoor events, detection of passing-by persons, assistance for safe driving, warning of dangerous activities, watching of environmental changes, etc. 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 omni-video based approach for synthesizing dynamic panoramic surround maps using stereo and motion analysis of video images from a pair of omni-cameras on a vehicle. Micheloni, et al. [4] used an autonomous vehicle to monitor moving objects in indoor environments. 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 omni-cameras for tracking human body features. A video surveillance system for localizing objects using multiple omni-cameras was proposed by Morita et al. [8]. Some related works using pairs of omni-cameras with hyperboloidal-shaped reflective mirrors can be found in [9, 10]. Specifically, Koyasu et al. [9] proposed an omni-directional stereo system consisting of two vertically aligned omni-cameras to detect and track obstacles. And Ukida et al. [10] used a similar system and a space encoding scheme to acquire 3D environment data for various applications. Furthermore, a method that reconstructs the 3D data of static nearby vehicles by a mobile robot using a stereo omni-camera (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 proposed in this study.

Most of the above-mentioned works are about indoor visual surveillances and the speeds for computing the 3D 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 equipped with a camera system on its roof. In this study, 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, which 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) constructing a perspective-view image of any direction automatically or at any time instant specified by the user for real-time inspection; (3) detecting and displaying any suspicious passing-by person automatically; and (4) measuring the 3D features (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, we design in this study an *omni-imaging device* which is composed of two...
catadioptric cameras with their 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 to the left-rear corner (see Fig. 1(a)). A new method, which combines skillfully the uses of such a pair of omni-imaging devices, a so-called pano-mapping technique [12], as well as the rotational invariance property of the omni-image to compute the 3D data of real-world points, is then proposed. The computation is based on table lookup and analytic formulas and so can be carried out in real-time to construct both top- and perspective-view images for instantaneous observation of the vehicle’s surrounding environment. Also, a vision-based scheme using a vertical-line property in the omni-image for detecting passing-by persons, computing their locations and heights, and displaying such feature data both from the top and the perspective views for convenient inspection is proposed as well, which are not found in previous works. A detailed comparison of the proposed method with related existing methods will be given later.

In the remainder of this paper, the configuration of the proposed system, the design of the omni-camera system, and the proposed technique for computing 3D data are described in Section II. The proposed techniques for detection and monitoring of a passing-by person are described in Section III. And 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, conclusions and descriptions of the contributions made in this study are given in Section VI.

II. SYSTEM CONFIGURATION AND DESIGN OF PROPOSED OMNI-IMAGING DEVICES

A. System Configuration

As illustrated in Fig. 1(b), the proposed system used in this study includes a video surveillance vehicle, a pair of two-camera omni-imaging devices $CS_A$ and $CS_B$ 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. Specifically, $COM_A$ is used to display the top-view image of the surrounding area of the video surveillance vehicle, and $COM_B$ to display the perspective-view images of specified directions outside the surveillance vehicle. The system process is divided into two phases, the learning phase and the patrolling phase. In the former, the pano-mapping tables of the used omni-cameras are constructed in advance, and in the latter, the video surveillance vehicle is driven in the outdoor environment for real security monitoring applications.

B. Camera Design Principle

The structure of each catadioptric omni-camera with a reflective mirror of the hyperboloidal shape used in this study 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 $O_s$ located at the middle point between the point $O_m$ and the lens center of the projective camera, may be described [12] as

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

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 $c^2 = a^2 + b^2$. Also, the axis of the camera is aligned with that of the mirror, and the lens center is fixed at the other focal point of the mirror shape.

![Figure 2. Omni-camera structure. (a) Omni-camera geometry. (b) Geometry between hyperboloidal-shaped mirror and projective camera.](image)

By the shape geometry of a hyperboloid described by (1), the value $\alpha$ specifying the elevation angle of a real-world point $P$ shown in Fig. 2(a) can be computed [10] by:

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

where the angle $\beta$ as shown in Fig. 2(a) can be computed as:

$$\beta = \pi/2 - \theta$$

with

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

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

$$\frac{d}{r} = \frac{f}{S_w}$$
where \( f \) is the focal length of the projective camera and \( S_n \) is the width of the square CMOS sensor in the camera.

The goal of omni-camera design in this study is to construct a mirror of the hyperboloidal shape and determine the distance from the camera to the mirror under the constraint that \( f, S_n, \) and \( r \) are of given values which fit the structure of the vehicle roof. The projective camera we use has \( f = 6 \) mm and \( S_n = 2.4 \) 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. So, according to (5) and the previously-mentioned fact that \( d = 2c \), we can derive \( d \) and \( c \) respectively to be \( d = 10 \) cm and \( c = 5 \) cm. Also, according to (3) and (4) and with \( \alpha \) assumed to be zero, the values of the angles \( \theta \) and \( \beta \) can be computed to be \( \theta = 21.8^\circ \) and \( \beta = 68.2^\circ \). Finally, using \( \beta \) and \( c \), we may reduce (2) to be an equation with only one variable \( b \) of the following form

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

which may then be solved to get \( b = 3.39 \) cm. And by \( c^2 = 5^2 = a^2 + b^2 \), \( a \) can be solved to be 3.68 cm. Thus, all of the desired parameters of the hyperboloidal-shaped mirror are obtained: \( a = 3.68, b = 3.39, c = 5, d = 10, \) all in the unit of cm.

C. 3D Data Acquisition

Each omni-imaging device used in this study consists of two omni-cameras designed in the way described above. The two cameras, one called the upper and the other the lower, are tied together, with their mirrors both facing down and their axes vertically aligned. An alternative illustration of the upper omni-camera configuration is shown in Fig. 3, where the image coordinate system (ICS) is specified by \((u, v)\) with the image center as 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 study, a new technique is proposed to compute the 3D data, \((X, Y, Z)\), of each real-world point \( P \) in the WCS by the use of the two elevation angles \( \alpha_1 \) and \( \alpha_2 \) of \( P \) with respect to the mirror bases of the two omni-cameras in an omni-imaging device, as illustrated in Fig. 4, where the values of \( \alpha_1 \) and \( \alpha_2 \) may be obtained by using a pano-mapping 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 omni-camera and is then reflected onto the camera imaging plane, resulting in an azimuth angle \( \theta \) (shown in Fig. 3) and an elevation angle \( \alpha \) (shown in Fig. 4(a)) as \( \alpha_1 \) or \( \alpha_2 \). By the pano-mapping method proposed in [12], given the image coordinates \((u, v)\) of an image pixel \( p \) corresponding to a real-world point \( P \), the elevation angle \( \alpha \) and the azimuth angle \( \theta \) of \( P \) can be obtained by table lookup using a pano-mapping table. An example of the pano-mapping table is shown in Table 1. Also, according to the mirror surface geometry which we assume as usual to be radially symmetric, a relationship, called the radial stretching function and denoted as \( f_r \), between the elevation angle \( \alpha \) and the radial distance \( r \) in the image plane with respect to the image center is established to be:

\[
r = f_r(\alpha) = a_0 + a_1\times\alpha^1 + a_2\times\alpha^2 + \ldots + a_5\times\alpha^5
\]

where the coefficients \( a_0 \) through \( a_5 \) are computed in the following way using the known image coordinates \((u, v)\) and the corresponding known world coordinates \((X, Y, Z)\) of six real-world landmark points \( P \), selected manually in advance, where \( i = 1 \) through 6: (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 \( \alpha_i \) for each of the six points as \( \alpha_i = \tan^{-1}(Z_i/\sqrt{X_i^2 + Y_i^2}) \); (3) solve the following six simultaneous equations to get the values of \( a_0 \) through \( a_5 \):

\[
r_i = f_r(\alpha_i) = a_0 + a_1\times\alpha_i^1 + a_2\times\alpha_i^2 + \ldots + a_5\times\alpha_i^5
\]

where \( i = 1, 2, \ldots, 6 \).

With \( a_0 \) through \( a_5 \) derived, a pano-mapping table like that of Table 1 is then constructed in the following way: for each real-world point \( P_{ij} \) with azimuth-elevation angle pair \((\theta_i, \alpha_i)\), 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 \( r_j = f_r(\alpha) = a_0 + a_1\times\alpha_j^1 + a_2\times\alpha_j^2 + \ldots + a_5\times\alpha_j^5 \). After the table is constructed, when an image pixel \( p \) with coordinates \((u, v)\) in a given omni-image is given and checked by table lookup
to be located in entry $E_{ij}$ with coordinates $(u_j, v_j)$ in Table 1, the azimuth-elevation angle pair of the corresponding real-world point $P$ can be obtained to be $(\theta, \alpha)$. Note that this azimuth-elevation angle pair only says that $P$ is located on a light ray $R$ with the azimuth direction $\theta$ and the elevation angle $\alpha$; it does not specify the 3D position of $P$ sufficiently — any real-world point on light ray $R$ will appear identically to be the pixel $p$ in the image with the same coordinates $(u_j, v_j)$. Also note that the pano-mapping table involves no camera parameter and is invariant in nature with respect to the camera position (i.e., it is not changed when the camera is moved around).

Back to the discussion about computing the 3D coordinates for a real-world point $P$ in the WCS. Since two omni-cameras are used in each imaging device, after the two image pixels $p_1$ and $p_2$ corresponding to real-world point $P$ are identified in the two omni-images taken by the two cameras, two elevation angles $\alpha_1$ and $\alpha_2$ as shown in Fig. 4(a) can be obtained by the above 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 3D data computation now can be achieved by using $\alpha_1$ and $\alpha_2$ to compute the world coordinates $(X, Y, Z)$ of $P$. For this, as shown in Fig. 4(b), by the law of sines in trigonometry, we have:

$$d_p = \frac{e}{\sin(90 + \alpha_1)} = \frac{e}{\sin(\alpha_1 - \alpha_2)}$$

(7)

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 measured manually in advance. Eq. (7) may be reduced to get $d_p$ as:

$$d_p = \frac{1}{\cos \alpha_1} \times \frac{e}{\tan \alpha_1 - \tan \alpha_2}.$$  

(8)

And so the horizontal distance $d_u$ and the vertical distance $Z$ of $P$ as shown in Fig. 4(a) may be computed respectively to be:

$$d_u = d_p \cos \alpha_1 = \frac{e}{\tan \alpha_1 - \tan \alpha_2};$$

$$Z = d_p \sin \alpha_1 = \frac{e \times \tan \alpha_1}{\tan \alpha_1 - \tan \alpha_2}.$$  

(9)

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

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

(10)

from which the value of $\theta$ 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 omni-camera says 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 of the angles by $\theta$, we can compute the values of $X$ and $Y$ in the WCS as:

$$X = d_u \times \cos \theta = \frac{e \times \cos \theta}{\tan \alpha_1 - \tan \alpha_2};$$

(11)

$$Y = d_u \times \sin \theta = \frac{e \times \sin \theta}{\tan \alpha_1 - \tan \alpha_2}.$$  

(12)

In summary, given a pair of matching image points corresponding to a real-world point $P$, we can compute the azimuth $\theta$ of $P$ by Eqs. (10), and obtain also a pair of elevation angles $\alpha_1$ and $\alpha_2$ of $P$ by pano-mapping table lookup. Then, the world coordinates $(X, Y, Z)$ describing the unique 3D position of $P$ can be computed by Eqs. (9), (11), and (12) analytically.

### 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. An 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 ones at the beginning of the object detection process. By subtracting the background image from the foreground one, a difference image is obtained. Because there usually exist noise pixels in the difference image, such as those caused by light variations, the difference image is thresholded next into a binary one to eliminate such noise pixels using the moment-preserving thresholding technique proposed by Tsai [13]. After these steps, we can obtain a bi-level image $I_{bl}$ with detected moving objects all labeled as black pixels. An 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 to detect 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 so vertical to the ground, is projected onto the image as a line \( L_2 \) going through the image center. This means that if a passing-by person stands on the ground, the midline \( L_2 \) of the person’s silhouette will go through the image center \( O_c \), as illustrated in Fig. 8(a). So, we may find the top of each passing-by-person’s head by the following steps: (1) transform the rectangular image coordinates \((u, v)\) of the bi-level difference image \( I_{b} \) into polar ones \((\theta, r)\) where \( \theta \) is the azimuth angle and \( r \) the radial distance; (2) scan each radial line inward from the image boundary to the image center based on the use of \((\theta, r)\); (3) find each sufficiently-long line segment in \( I_{b} \) and group neighboring segments so found into a set \( L_s \); (4) find the farthest line-segment end point in \( L_s \) with respect to the image center as the passing-by person’s head location. If non-human objects will appear possibly in the omni-image, Step (3) should be refined using features like the width of the set \( L_s \) (equal to the number of line segments in \( L_s \), which means 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 Figs. 8(c) and 8(d).

\[ h = d_h - Z \]  

where \( d_h \) is the height of the upper camera with respect to the ground and \( Z \) is as that computed by Eq. (9) because \( P \) is the passing-by person’s head point (shown as the red point in Fig. 8(b)). Note that the origin of the WCS is located at the mirror base center of the upper omni-camera. Also, 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 (e.g., see the color points in Fig. 8(c)).

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

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 the lower one, 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 affixed on the vehicle’s roof is known in advance by manual measurement. Also, 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.

\[ d_w = \sqrt{X^2 + Y^2} \]

We adopt a backward-mapping scheme based on the use of the pano-mapping 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:

\[ d_w = \sqrt{X^2 + Y^2} \]  

Accordingly, the azimuth angle \( \theta \) of \( P \) can be derived as:
\[ \theta = \cos^{-1}(X/d_w) = \sin^{-1}(Y/d_w). \]  

(15)

On the other hand, because \( d_w = d_h \times \cot \alpha \) where \( d_h \) is the height of the upper camera’s mirror base, we can compute the elevation angle \( \alpha \) of \( P \) by:

\[ \alpha = \tan^{-1}(d_h/d_w). \]  

(16)

Furthermore, using the radial stretching function \( r = f_r(\alpha) \) mentioned previously, the radial distance \( r \) corresponding to \( \alpha \) can be derived. Accordingly, by the rotational-invariance property of omni-imaging, the coordinates \((u, v)\) of the image pixel \( p \) corresponding to \( P \) can be obtained from (15) as:

\[ u = r \cos \theta; \quad v = r \sin \theta. \]  

(17)

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

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

### B. Merging of two top-view images into a single one

By the way describe above, we can construct two top-view images \( I_{t1} \) and \( I_{t2} \) using the omni-images taken from the two upper cameras, like those shown in Figs. 11(a) and 11(b). It is assumed that the relative position of the two upper cameras on the vehicle roof are measured manually in advance and assumed that the relative position of the two upper cameras on the roof, the vehicle shape in the top-view images always appears at fixed locations in the taken omni-images. Therefore, it is feasible to superimpose a real surveillance vehicle shape \( S_r \) on the elliptical-shaped top-view image like that in Fig. 11(d) as the central landmark of the image, making observation of the vehicle’s position in the surround more convenient. For this, 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 \( I_{t1} \) and \( I_{t2} \), 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 in yellow the surveillance vehicle’s shape area \( A \) in \( I_{t1} \) and \( I_{t2} \) before they are merged into one, resulting in a figure like that shown in Fig. 12(a); (3) apply a texture synthesis scheme to fill ground texture into the marked surveillance vehicle shape area \( A \); and (4) superimpose a pre-segmented real top-view vehicle shape \( S_r \) like that shown in Fig. 12(b) on the resulting image of the last step, yielding a better-looking output image like that shown in Fig. 12(c). Note that \( S_r \) is obtained in the learning phase by manual segmentation from a top-view image like that of Fig. 11(a).

Step (3) above is carried out by the following way: for each pixel \( p_n \) in the surveillance vehicle shape \( A \), find the pixel \( p_n \) which is outside \( A \) and closest to \( p_n \), and use the color of \( p_n \) to fill up \( p_n \). Also, the relations of pixels in \( A \) with those 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.

**Figure 11.** Top-view image construction. (a) Right-front top-view image. (b) Left-rear top-view image. (c) Integrated top-view image (red dots: centers of original images). (d) Top-view image viewed in an elliptical shape.

### C. Superimposition of Surveillance Vehicle Shape on 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 taken omni-images. Therefore, it is feasible to superimpose a real surveillance vehicle shape \( S_r \) on the
diagram.

**Figure 12.** Superimposing real surveillance vehicle shape onto top-view image. (a) Top-view image with vehicle shape marked in yellow. (b) Pre-segmented real shape of video surveillance vehicle. (c) Result of ground texture filling and superimposition of real surveillance vehicle shape.

### 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 pano-mapping table [12]. The major steps include: (1) map each pixel \(p\) in the desired perspective-image \(I_p\) at coordinates \((k, l)\) to a pair of elevation and azimuth angles \((\alpha, \theta)\) in the pano-mapping table according to the geometry of the desired perspective transformation; (2) find the image coordinates \((u, v)\) in the table entry corresponding to \((\alpha, \theta)\); and (3) assign the color channel values of pixel \(p\) in \(I_o\) to be those of the pixel at coordinates \((u, v)\) in \(I_o\). The detail of mapping \((k, l)\) to \((\alpha, \theta)\) in Step (1) is described as follows.

Assume that the perspective-view image \(I_p\) we want to generate from \(I_o\) is at a distance \(D\) to the mirror base center \(O_m\) and has \(M_x \times N_y\) pixels. Also assume that \(I_p\) is the image of a planar rectangular region \(A_p\) with width \(W \times H\) and is perpendicular to the floor in the real-world space, as illustrated from the top view by Fig. 13.

(a) Computing the azimuth angle \(\theta\) —

The angle \(\phi\) 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 \(\phi\) may be computed as \(\phi = \cos^{-1}[1 - W^2/(2D^2)]\). Also, the angle \(\beta\) in the figure is just \(\beta = (\pi - \phi)/2\). Next, let \(P_j\) denote the intersection point of the light ray \(R_p\) projected onto the image pixel \(p\) and the planar region \(A_p\) mentioned previously. Then, we may compute the distance \(d\) between point \(P_j\) and the border point \(P_r\) shown in Fig. 13(b) by linear proportionality to be \(d = k \times W/M_p\) 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 \(P_j\) and the mirror base center \(O_m\) as shown in Fig. 13(b) satisfies the equality \(L^2 = D^2 + d^2 - 2 \times D \times d \times \cos \beta\). Also, the distance \(h\) from point \(P_j\) to the line segment \(O_mP_r\) may be computed as \(h = d \times \sin \beta\). Finally, the azimuth angle \(\theta\) of point \(P_j\) with respect to \(O_mP_r\) satisfies \(\sin \theta = h/L\), which, by the equalities derived above, leads to the following desired value:

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

(b) Computing the elevation angle —

An illustration of the involved imaging configuration for computing the elevation angle \(\alpha\) 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. And so, by linear proportionality again, the height of \(P_j\) may be computed to be \(H_p = (i \times H)/N_p\) where \(i\) is the coordinate of pixel \(p\) in the vertical direction. Finally, by trigonometry, the desired elevation angle \(\alpha\) may be derived to be:

\[
\alpha = \tan^{-1}(H_p/L).
\]  

This completes the derivations of the azimuth-elevation angle pair \((\theta, \alpha)\) conducted in Step (1) mentioned above. Note that in these derivations, the start direction (specified by the line segment \(O_mP_r\)) of the angle \(\phi\) spanned by the width \(W\) of \(I_p\), as shown in Fig. 14, coincides with the horizontal direction \(\theta^o\), resulting in the azimuth angle \(\theta\). Of course, perspective-view images for other azimuth angles may also be generated. A convenient scheme to do this is described next.

E. Generation of Perspective-view Images Specified with Mouse Clicks or Panel Touch

In this study, a technique is proposed to enable a user to change the view direction of the perspective-view image conveniently by mouse clicking, or equivalently, by panel touching). Fig. 15(a) shows a perspective-view image 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. Specifically, the horizontal motion of a mouse specified by its horizontal location \(M_x\) may be used to define the azimuth angle \(\theta\) of the space point \(P_c\) corresponding to the image center \(p_c\); and similarly the vertical motion of the mouse specified by its vertical location \(M_y\) may be used to define the elevation angle \(\alpha\) of \(P_c\).

Accordingly, as an interaction with the proposed system via mouse clicking on the computer screen which shows a perspective-view image like that of Fig. 15(a), we define in this study an angle value \(\theta_{\text{mouse}}\) according to the mouse click location in the horizontal direction so that \(\theta_{\text{mouse}}\) becomes larger and larger as the mouse moves from right to left in the image of Fig. 15(a); and then modify the equation for computing the azimuth angle, namely, Eq. (18), derived previously to be:

\[
\theta = \sin^{-1}(h/L) + \theta_{\text{mouse}}.
\]  

Similarly, we define another angle value \(\alpha_{\text{mouse}}\) according to the mouse click location in the vertical direction such that \(\alpha_{\text{mouse}}\).
increases gradually as the mouse moves from top to bottom in Fig. 15(a); and then modify Eq. (19) to be:

\[ \alpha = \tan^{-1} \left( \frac{H_p}{L} \right) + \alpha_{\text{mouse}}. \]  

The user interface becomes friendlier after adding the two variables \( \theta_{\text{mouse}} \) and \( \alpha_{\text{mouse}} \) and a user of the surveillance vehicle can now choose any view direction conveniently 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 generated in this way are shown in Figs. 15(c) and 15(d).

Figure 15. Corresponding omni-image and perspective-view image. (a) A perspective-view image. (b) Omni-image part (enclosed roughly 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).

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

A. Experimental Results of 3D Data Computation

At first, we conducted an experiment to test the precision of the computed 3D data by computing the positions of some real-world points and compared them with those obtained manually. 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 5cm, and the bigger width between every two crosses is 25 cm. We picked out 15 pairs of corresponding pixels from these grids and shapes in the two taken images. An example is shown in Fig. 17. We then calculated, by Eqs. (9) in Section II.C, the horizontal distance \( d_w \) and the height \( Z \) of the real-world points.

In addition, to increase the computation accuracy, we used four radial stretching functions instead of just one in the pano-mapping process, each function dealing with a quarter of the omni-image, as shown in Fig. 18. This measure is especially necessary when the assumption of radial symmetry of the mirror surface mentioned previously is not valid, which happens to be the case encountered in this study because of imperfect manufacturing of the mirrors.

The results of calculations of the 3D data, \( d_w \) and \( Z \), of 15 selected landmark points on the calibration pattern using the taken images are shown in Tables 2 and 3, which resulted respectively from the use of the pano-mapping tables of Tables A and B depicted in Fig. 18. As can be seen, the differences between the measured data and the computed ones are all small, resulting in small RMSE values (each defined to be the square root of the average of all the difference values). All the error rates (each defined to be the ratio of the RMSE over the average measured data value) can be seen to be about 5%, which are good for practical applications. 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 3D computation results. The computed correlation coefficient values 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.

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

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

Fig. 18. Four image regions corresponding to four radial stretching functions used in constructing pano-mapping tables.

<table>
<thead>
<tr>
<th>Point No.</th>
<th>Azimuth angle (°)</th>
<th>Elevation angle (°)</th>
<th>(1) Measured 3D real-world point data (cm)</th>
<th>(2) Computed 3D real-world point data (cm)</th>
<th>(3) Differences (1) – (2) (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \theta )</td>
<td>( \alpha )</td>
<td>( d_w )</td>
<td>( Z )</td>
<td>( d_w )</td>
</tr>
<tr>
<td>1</td>
<td>0.00</td>
<td>54.95</td>
<td>249</td>
<td>355</td>
<td>250.2</td>
</tr>
<tr>
<td>2</td>
<td>0.00</td>
<td>52.96</td>
<td>249</td>
<td>330</td>
<td>251.4</td>
</tr>
<tr>
<td>3</td>
<td>0.00</td>
<td>50.77</td>
<td>249</td>
<td>305</td>
<td>255</td>
</tr>
<tr>
<td>4</td>
<td>0.00</td>
<td>48.35</td>
<td>249</td>
<td>280</td>
<td>260.6</td>
</tr>
<tr>
<td>5</td>
<td>0.00</td>
<td>45.68</td>
<td>249</td>
<td>255</td>
<td>247.6</td>
</tr>
<tr>
<td>6</td>
<td>0.00</td>
<td>42.73</td>
<td>249</td>
<td>230</td>
<td>257.7</td>
</tr>
<tr>
<td>7</td>
<td>0.00</td>
<td>39.46</td>
<td>249</td>
<td>205</td>
<td>236.6</td>
</tr>
<tr>
<td>8</td>
<td>0.00</td>
<td>37.35</td>
<td>249</td>
<td>190</td>
<td>227.7</td>
</tr>
<tr>
<td>9</td>
<td>0.00</td>
<td>34.32</td>
<td>249</td>
<td>170</td>
<td>223</td>
</tr>
<tr>
<td>10</td>
<td>0.00</td>
<td>31.07</td>
<td>249</td>
<td>150</td>
<td>231.8</td>
</tr>
<tr>
<td>11</td>
<td>0.00</td>
<td>27.57</td>
<td>249</td>
<td>130</td>
<td>238</td>
</tr>
<tr>
<td>12</td>
<td>10.69</td>
<td>54.45</td>
<td>227.96</td>
<td>319</td>
<td>219.5</td>
</tr>
<tr>
<td>13</td>
<td>12.00</td>
<td>57.47</td>
<td>203.45</td>
<td>319</td>
<td>185.5</td>
</tr>
<tr>
<td>14</td>
<td>15.85</td>
<td>64.10</td>
<td>154.89</td>
<td>319</td>
<td>156.1</td>
</tr>
<tr>
<td>15</td>
<td>18.84</td>
<td>67.67</td>
<td>130.5</td>
<td>319</td>
<td>131.02</td>
</tr>
</tbody>
</table>

Average RMSE 230.42 258.4 12.52 12.338 Error rate = RMSE / Average 0.43% 0.47%

Table 2. Statistics of computed 3D data using pano-mapping table Table A.
B. Experimental Results of passing-by person detection

We have also tested the technique 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 is found in the image, the body point which 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 image sequence with a person walking around the video surveillance vehicle, and the person was detected successfully 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 feet positions.

C. Experimental Results of Integration of System Functions

A system integrating the proposed techniques has been implemented, including the functions of (1) construction of a top-view image of the surveillance vehicle’s surrounding area; (2) detection of passing-by persons around the vehicle; (3) generation of a perspective-view image whose view direction may be specified by mouse clicking or is determined automatically to show the detected person. Each camera of model Artcam-200MI used in the proposed system takes 0.1 sec. to acquire a 1600×1200 image, and the processing cycle time of the system is 0.2 sec. (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 that of the higher ones 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

(a) 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. While good theories can be found in these papers, we tried in this study to design an omni-camera with suitable parameters to fit the roof structure of a vehicle used in this study.

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

(c) About object detection on moving vehicles — Gandhi and Trivedi [3] used images of two omni-cameras affixed to car side mirrors to synthesize perspective-view images, and detect in-front vehicles by binocular stereo and lateral ones by motion stereo. They also used parametric ego-motion compensation for an omni-directional vision sensor to detect surrounding events [19] [20]. The proposed method instead emphasizes 3D detection of both passing-by persons’ locations and heights, using omni-images directly without generating perspective views to speed up the detection process. This is made possible by the uses of the designed two-camera omni-imaging devices and the single pano-mapping technique.

(d) About top-view image generation — Ehlgen and Pajdla [21][22] installed two omni-cameras on the side mirrors of a truck to generate a bird’s-eye-view image covering the frontal scene. The top-view image generated in this study instead
covers the entire vehicle surrounding. Liu et al. [23] affixed six fisheye cameras around a car to generate a bird’s-eye-view image of the surrounding for driving assistance. No 3D data computation was considered, contrasting with our method which computes 3D feature point data.

(e) About human detection — Liu et al. [24] detected human motion with an omni-camera on a mobile robot using ego-motion compensation and temporal differencing after unwarping omni-images. The proposed method of this study instead detects passing-by persons directly from omni-images. Ng et al. [25] used multiple omni-vision sensors to synthesize perspective views and track human activities using N-ocular stereo techniques without acquiring range information. In contrast, our method detects human beings in vehicle surroundings and computes their 3D features (location and height) for various applications.

(f) About perspective-view image generation — Huang et al. [26] presented an in-car omni-imaging system which 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, while our method uses the pano-mapping approach. Kawasaki et al. [27] obtained 3D information by spatio-temporal analysis of omni-images using calibrated camera parameters, contrasting again with the tabular pano-mapping technique used by our method, involving no camera parameter.

(g) Comparisons about implemented functions — The result of a more detailed comparison of the previously-mentioned methods (except [17] which is a survey paper) with the proposed one in terms of a set of implemented functions is listed in Table 4, from which it can be seen that the proposed method has integrated more functions than the others.

VI. CONCLUSIONS

A pair of two-camera omni-imaging devices has been designed properly for use on the top of a video surveillance vehicle to monitor passing-by persons. The two devices are affixed to the right-front and left-rear of the vehicle roof efficiently to facilitate generation of a top-view image which covers the vehicle surrounding area. A new 3D data computation technique based on the pano-mapping 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. A passing-by person appearing around the surveillance vehicle can be detected automatically using an omni-image property about upright objects, with his/her location and body height computed by the proposed 3D data acquisition technique. A top-view image of the vehicle’s surrounding area with a real vehicle shape inserted properly in the middle is generated by registering two omni-images taken by two upper cameras. Perspective-view images covering the detected person or any interesting scene spot can also be generated in real-time automatically or by mouse clicking for convenient inspection. The system is useful for many security monitoring applications around the video surveillance vehicle. The experimental results show the feasibility and precision of the proposed method for practical applications.

REFERENCES

Table 4. Comparison of existing methods (O: yes, X: no; ***: using pano-mapping table).

<table>
<thead>
<tr>
<th>Method</th>
<th>Omni-image unwarping</th>
<th>Top view generation</th>
<th>Integration of multiple images</th>
<th>Real-time processing</th>
<th>Applied in outdoor environment</th>
<th>Passing-by people detection</th>
<th>Object distance estimation</th>
<th>Object height estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our method</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>Su et al. [18]</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>O</td>
<td>O</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Gandhi &amp; Trivedi [3]</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>O</td>
<td>X</td>
<td>O</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Gandhi &amp; Trivedi [20]</td>
<td>X</td>
<td>O</td>
<td>X</td>
<td>O</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Ehligen &amp; Pajdla [21, 22]</td>
<td>X</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Liu, et al. [23]</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Liu et al. [24]</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Ng, et al. [25]</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>O</td>
<td>X</td>
<td>O</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Huang et al. [26]</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Kawasaki et al. [27]</td>
<td>O</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>O</td>
<td>X</td>
<td>O</td>
<td>O</td>
</tr>
</tbody>
</table>


Pei-Hsuan Yuan received the B.S. degree in computer science and information engineering from National Chi Nan University, Taiwan in 2008 and the M.S. degree in computer science from National Chiao Tung University, Taiwan in 2010. She was a research assistant at the Computer Vision Laboratory in the Department of Computer Science at National Chiao Tung University from August 2008 to July 2010. She is now with Taiwan Semiconductor Manufacturing Co. in Taiwan. Her research interests include computer vision, image processing, video surveillance and applications.

Kuo-Feng Yang received the B.S. degree in mathematics from National Chung Hsing University, Taiwan in 2006 and the M.S. degree in computer science and information engineering from Yuan Ze University, Taiwan in 2008. He has been a research assistant at the Computer Vision Laboratory in the Department of Computer Science at National Chiao Tung University since August 2008. He is also working toward his Ph. D. degree there. His current research interests include image and vision computing, and autonomous vehicle applications.

Wen-Hsiang Tsai received the B.S. degree in EE from National Taiwan University, Taiwan, in 1973, the M.S. degree in EE from Brown University, USA in 1977, and the Ph.D. degree in EE from Purdue University, USA in 1979. Since 1979, he has been with National Chiao Tung University (NCTU), Taiwan, where he is now a Chair Professor of Computer Science. At NCTU, 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 1999 to 2000, he was the Chair of the Chinese Image Processing and Pattern Recognition Society of Taiwan, and from 2004 to 2008, the Chair of the Computer Society of the IEEE Taipei Section in Taiwan. From 2004 to 2007, he was the President of Asia University, Taiwan.

Dr. Tsai 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 papers. Dr. Tsai has received many awards, including the Annual Paper Award from the Pattern Recognition Society of the USA; the Academic Award of the Ministry of Education, Taiwan; the Outstanding Research Award of the National Science Council, Taiwan; the ISI Citation Classic Award from Thomson Scientific, and more than 40 other academic paper awards from various academic societies.

Dr. Tsai’s current research interests include computer vision, information security, video surveillance, and autonomous vehicle applications. He is a Life Member of the Chinese Pattern Recognition and Image Processing Society in Taiwan and a Senior Member of the IEEE.