Moving Target Tracking and Measurement with a Binocular Vision System

T. W. Yang *, K. Zhu, and Q. Q. Ruan
School of Computer and Information Technology
Beijing Jiaotong University
Beijing 100044, China

J. D. Han
The State Key Laboratory of Robotics
Shenyang Institute of Automation, Chinese Academy of Sciences
Shenyang 110016, China

Abstract - visual tracking and measurement is one of the most important tasks in computer vision and finds applications in traffic surveillance, vision-guided mobile robots etc. In this paper, a binocular vision system is used to achieve the visual servoing, mimicking the movements of human eyes. First, a robust scheme, which combines adaptive background subtraction and Camshift algorithms, is proposed to detect and track a moving object. Then, we discuss the problems of camera parameter calibration and position measurement of moving objects. A new approach is introduced to realize the extrinsic parameter calibration of a pan-tilt camera, providing successful tracking and accurate measurement of a moving target, even as it is outside the view of camera. Numerous experiments have been done on the binocular vision system of a humanoid mobile platform, and the results show that the proposed approaches work very well.

I. INTRODUCTION

Autonomous mobile robots drew much passion in the past decades. Even now, the interest is still growing in the communities of robotics and artificial intelligence. Of all sorts of reasons, one is that there exist various potential applications for mobile robots, such as home or office service, industrial surveillance and inspection, intelligent transportation, planet exploration and military reconnaissance, where tasks to be carried out are tiring, dangerous, or inaccessible for people. A mobile robot is coupled with its moving environment dynamically, and its capabilities totally depend on equipped sensors. It is very important for the mobile robot to understand its surrounding from the exteroceptive sensing information and to move safely with no failure. Dead reckoning sensors (optical encoder and odometer) are widely used to derive the location of a mobile robot, and the tactile, infrared, ultrasonic sensors for collision avoidance. As we know, for human being, visual perception plays a privileged role in analysis and identification of the real world around us, while it is difficult for a mobile robot to have the visual capabilities that humans have.

Nowadays, visual devices are more powerful and cheaper, and image processing algorithms are not time-consuming as before. Visual applications can be found in robot vision [1, 4] and intersection surveillance [2], etc. In the past years, major attention has been paid to object detecting and tracking, and many papers were published in this area. Asada et al. [3] utilized a binocular stereo vision to realize unknown object tracking. An extension of adaptive visual servoing was proposed without using the knowledge of camera parameters and object motion, but stationary references are needed to predict object motion. Allen et al. [4] proposed an approach that uses image difference techniques to track and grab a moving object. In [5], Capellades et al. described a system for tracking humans and detecting human-object interaction in a video sequence. Humans and objects are modeled through color distributions, and detected with an appearance based approach while maintaining continuous labeling.

Popular methods of detecting and tracking an object include variations on the theme of background subtraction algorithm [6]. It requires a relatively small computation time and shows robust detection in good illumination conditions, while suffers from the problems of occlusion, the presence of shadow, and sudden illumination change, and so on. When a camera moves and the scene changes instantly, the algorithm is unsuitable to detect and track moving objects in robot vision. Therefore, many efforts have been made to improve the background subtraction algorithm. It is integrated in [2] with feature tracking and multi-level grouping algorithms, to directly group corner features into objects using proximity and motion history. Alsaqre and Yuan [7] used a shadow detection method to improve object detection, and two similarity functions are used to realize accurate matching of object features.

In recent years, mean shift is regarded as one of the effective tracking algorithms. It is a statistical method for finding the mode of a probability distribution, initially introduced by Fukunaga and Hostetler [8]. The mode of the probability distribution is used to track an object in video sequences. But, the distribution of a moving object changes from frame to frame, and it is very challenging to track moving objects robustly. In [9], mean shift was employed to determine the target candidate that is the most similar to a given target model, and the prediction of the next target location is computed with a Kalman filter. As it is a color-based tracking method, searching is computationally intensive and probably fails due to color variation in irregular illumination conditions. Therefore, a combined color and color gradient histogram is used in [10], to represent the tracked target appearance model in feature space. But if an occlusion occurs, the color distribution of a target changes

* The corresponding author: Tel.: +86 10 5168 4902; fax: +86 10 5184 0526; E-mail address: twyang@bjtu.edu.cn.
abruptly, and target losing probably happens. Therefore, Deguchi et al. [12] used a particle filter algorithm to estimate the motion model of the target. Kernel profile and bandwidth play an important role in mean shift algorithm. Chen et al. [11] used the asymmetric object shape as the kernel to remove the background noise of the symmetric kernel and to maximize the Bhattacharyya distance. Here, Camshift algorithm [13], originated from mean shift, is used to obtain probability distributions that vary dynamically, and to track moving objects in video frame sequences.

In many visual applications, one is to establish the spatial relationship between a robot and obstacles. The robot needs to know the locations of the objects in order to realize safe navigation and tracking. Visual distance measurement was thus received considerable attention in the past years, and many techniques have been suggested in mobile robot applications, providing absolute or relative object location to mobile robot. A vision system may have one or two even more cameras, and camera may be stationary or moving. To measure an object position, all the camera’s intrinsic and extrinsic parameters should be calibrated. With monocular vision, it is very difficult to determine the position of an object in a 3-dimensional (3D) space, if no knowledge of reference scale is provided a priori. Juengel et al. [14] introduced a bearing-based distance measurement for objects with unknown size. Reference objects are used to determine the position and orientation of a Sony AIBO’s camera to the ground. However, reference information is not always accurate enough in practice. Instead, binocular vision is able to realize accurate 3D measurement and does not need any reference. The position of an object is directly computed from the projected pixel positions of the object in the cameras. Here, a binocular vision system is then employed to measure the position of a moving object in a 3D space, and as the tracked object is going to be out of view, both cameras are allowed to pan and tilt on the horizontal and vertical planes to secure the tracking.

In this paper, our aim is to detect and track a moving object through a video sequence, and simultaneously measure its position in a 3D space. In the following section, an approach combined background subtraction and Camshift algorithms is proposed to segment object model and track object motion in a video sequence. Section 3 introduces an efficient algorithm to measure the distance of moving objects, and a new method for camera calibration is presented as its pan and tilt angles vary. Finally, conclusions are made in Section 4.

II. MOVING TARGET TRACKING

Target tracking finds many potential applications such as visual navigation of mobile robots, environment exploration and surveillance etc. As it is an important problem, target detection must precede any tracking. To this end, a target model is first established as a template to match target candidates in a video frame sequence, and then Bhattacharyya distance is used to determine the position of a target candidate tracked. In this study, background subtraction (BS) is considered to segment the target model in the foreground of the region of interest, and a mean shift based algorithm, Camshift, is used to track the target moving in the scene. The algorithms are described below, respectively.

A. Background Subtraction Algorithm

Given a video sequence, the success of tracking an object depends on the correct detection of the object. For many years, BS algorithms have been widely used to obtain the tracked model. A BS algorithm first extracts a background hypothesis from a sequence of images, and then calculates the difference between the background hypothesis and the current image to find foreground objects.

Now, let us establish first a background hypothesis, namely, target model. As irregular illumination and shade may cause slight change in the background image, especially in a changing environment, the running average algorithm is used here to obtain a robust background, given by

$$\mathbf{F}_k(x, y) = (1 - \alpha) \mathbf{F}_{k-1}(x, y) + \alpha \mathbf{I}_k(x, y)$$

where $\mathbf{I}_k(x, y)$ is the $k$th image in a video sequence, $\mathbf{F}_k(x, y)$ represents the background of the $k$th image in the sequence, and $\alpha$ denotes the background updating ratio.

Then it is comparatively easy to obtain the target model by subtracting the background from the current image, $\mathbf{F}_k(x, y)$, that is,

$$I^b_k(x, y) = |\mathbf{F}_k(x, y) - \mathbf{F}_k(x, y)|$$

where $I^b_k(x, y)$ is the model of the target.

Additionally, this target model can be rewritten to be a binary image, where non-zero pixel values represent the target to be tracked. The binary image is given by

$$I^b_k(x, y) = \begin{cases} 1 & I^b_k(x, y) > T \\ 0 & \text{else} \end{cases}$$

where $I^b_k(x, y)$ is the binary model, namely, the segmentation of the object, and $T$ is the threshold value specified.

As can be seen, the target model is updated with time in the video sequence. The algorithm is simple, especially suitable for scene background change and moving targets. Moreover, a single detection failure due to occlusion and illumination does not cause track losing on the whole.

B. Camshift Algorithm

Real-time tracking algorithm must be fast and efficient, and simultaneously consume less computation resources. To this end, a robust tracking algorithm based on dynamic probability distribution model is taken into consideration here, and developed from mean shift. The mean shift tracking algorithm is quite well-known and efficient for blob tracking where the target and candidate models are represented by their color histograms. The position of the candidate blob is determined by virtue of the Bhattacharyya distance between the candidate and target models. Mean shift (MS) is a gradient ascent algorithm, which climbs the gradient of a probability distribution to find the nearest dominant mode of a target object.
A mean shift algorithm is based on the definition of a Bhattacharyya distance. According to this definition, we have the density estimation \( \rho(x) \) at \( x \) in the current image as

\[
\rho(x) = C_h \frac{1}{h} \left[ \sum_{i=1}^{n} k\left( \frac{x - x_i}{h} \right) \right],
\]

where \( C_h \) is a normalization constant, \( \{x_i\}_{i=1,...,n} \) is the pixels of the target window, \( k \) is called the kernel profile function, \( h \) is the kernel bandwidth.

To find the mode of a target probability distribution, the problem turns to find the maximum value of the density estimation of (4), namely, the location of a target candidate.

Differentiating (4), we have the density estimation gradient at \( x \), with the form of

\[
\nabla \rho(x) = \frac{2C_p k}{h^2} \left( \sum_{i=1}^{n} \left( g\left( \frac{x - x_i}{h} \right) - g\left( \frac{x - x}{h} \right) \right) \right) \frac{x - x_i}{h^2} \frac{x - x}{h^2},
\]

where \( g(x) = k'(x) \).

Set

\[
m_{h,g}(x) = \left( \sum_{i=1}^{n} \frac{x - x_i}{h} \right) \frac{x - x}{h^2}.
\]

To obtain the maximum value of (4), we search the mode of the probability distribution of a target candidate in the current image with (6). The MS algorithm is computationally fast and performs satisfactorily for various sequences. But, if the target model is extracted from a color histogram without spatial information, a drift is often observed when the target undergoes partial occlusion. Several improvements to the algorithm have been suggested. Of them, one is Camshift, which is often adopted to track moving objects in a complex scene.

The advantage of Camshift is that there is no need to calculate projection of color histogram of all the pixels. Only a little larger than the searching window is enough, so that the algorithm, avoiding complex computation, is suitable for the real-time target searching and tracking. In addition, Camshift allows the background to change abruptly, so that it is applicable in open fields. The algorithm is described as follows,

1. Choose a search window and specify its initial size and location.
2. Compute the color histogram of the searching window in the current image.
3. Compute the mean location in the search window with the following steps.
   1. Compute the zero moment with
   \[
   Z_{00} = \sum_{x} \sum_{y} I(x, y)
   \]
   2. Compute the first moment for \( x \) and \( y \)
   \[
   Z_{10} = \sum_{x} \sum_{y} x I(x, y) \quad Z_{01} = \sum_{x} \sum_{y} y I(x, y)
   \]
   where \( I(x, y) \) is the pixel value at the position of \( (x, y) \) in the image.
4. Update the size of the search window.
5. Repeat (3) and (4), until the centroid converges to some point specified.

Usually, the model of a tracked target is needed to give a priori in traditional Camshift algorithms, while here the model is provided by the previous BS algorithm, and updated with the current image in real time. Thereby, the combined BS and Camshift approach is robust for real-time tracking, especially for moving objects, and avoids track losing caused by inaccurate target models. Moreover, the cameras of our vision system are allowed to move, that is to say, the proposed algorithms can work in a changing background scene.

### C. Tracking experiments

The performance of the proposed approaches is examined through tracking a walking person. Figure 1 shows several pieces of experimental results. Here only the images from the left camera of a binocular vision system, which will be introduced in next section, are presented. The pan-tilt camera is controlled to rotate horizontally and vertically and search for the walking person when he is out of view of the camera. Figure 1(a) gives the tracked target model, and (b), (c), (d) and (e) are the images taken from a video sequence. As can be seen, the background of the scene is changing with the camera motion, and the size of the search window is various as the tracked target walks in front of the camera. Hence, the combined algorithms are quite robust, and work very well to track a moving target even as the camera is panned and tilted.

(a) target model, (b) frame 1, (c) frame 20, (d) frame 67, (e) frame 122.

Figure 1. Tracking a walking person.

### III. Binocular Measurement

In visual-guided applications, a mobile robot is often controlled to follow a moving object, as discussed above. So,
a safe distance should meet to avoid collisions while the mobile robot tracks the target, as humans do in driving. That is to say, a mobile robot should always know exactly the location of a tracked target, as it moves around and achieves successful tracking and navigation. A binocular vision system, which is used on a humanoid mobile platform shown in Figure 2, is considered here to realize the accurate distance measurement in a 3D space. In the following, the measuring system is introduced.

![Figure 2. A humanoid mobile platform.](image)

### A. A Binocular Vision System

Figure 3 shows the distance measurement model of a binocular vision system. The two cameras used are identical, and the pan and tilt angles of each camera can be controlled. The projection of an object point \( P(x, y, z) \) in the world coordinate system \( \Sigma_w \) is denoted as \( p_i(u_i, v_i) \) in the image plane \( \pi_i \), \( i = 1, 2 \). Essentially, \( p_i \) is the intersection point of the line \( PO_i \) with the image plane \( \pi_i \) and \( O_i \) is the camera’s optical center, and at the same time it is the origin of the camera coordinate system \( \Sigma_i \). Assuming that, the position of the object point is \( p_{oi} (x_{ci}, y_{ci}, z_{ci}) \) in \( \Sigma_{ci} \), from the pinhole camera model, we have

\[
\begin{bmatrix}
    u_i \\
    v_i \\
    1
\end{bmatrix} =
\begin{bmatrix}
    a_{xi} & 0 & u_{oi} & 0 \\
    0 & a_{yi} & v_{oi} & 0 \\
    0 & 0 & 1 & 1
\end{bmatrix} \begin{bmatrix}
    x_{ci} \\
    y_{ci} \\
    z_{ci} \\
    1
\end{bmatrix},
\]

(7)

where \( a_{xi}, a_{yi} \), \( u_{oi} \) and \( v_{oi} \) are the intrinsic parameters of the \( i \)th camera, and \( M_{i}^{in} \) is called intrinsic parameter matrix.

At the same time a point in the world coordinate system can be transformed into the camera coordinate system, that is

\[
\begin{bmatrix}
    x_{ci} \\
    y_{ci} \\
    z_{ci}
\end{bmatrix} =
\begin{bmatrix}
    R_i & t_i \\
    0 & 1
\end{bmatrix} \begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix} = M_{i}^{ex} \begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix}
\]

(8)

where \( R_i \in \mathbb{R}^{3 \times 3} \) and \( t_i \in \mathbb{R}^{3 \times 1} \) are respectively the extrinsic rotational matrix and translational vector of the \( i \)th camera coordinate system \( \Sigma_i \) with respect to \( \Sigma_w \). \( M_i^{ex} \) is called extrinsic parameter matrix. In fact, \( t_i \) is also the vector of the camera optical center in the world coordinate system.

Combining (7) and (8), we can obtain the relationship of any spatial point with its corresponding pixel in the image plane, given by

\[
\begin{bmatrix}
    u_i \\
    v_i \\
    1
\end{bmatrix} =
\begin{bmatrix}
    a_{xi} & 0 & u_{oi} & 0 \\
    0 & a_{yi} & v_{oi} & 0 \\
    0 & 0 & 1 & 1
\end{bmatrix} \begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix} = M_{i}^{in} M_{i}^{ex} \begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix}
\]

(9)

Rewriting (9) in a compact form of

\[
\begin{bmatrix}
    u_i \\
    v_i \\
    1
\end{bmatrix} = M_{i} \begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix}
\]

(10)

where \( M_i \in \mathbb{R}^{3 \times 4} \) is called the perspective matrix of the \( i \)th camera, defined as

\[
M_i = M_i^{in} M_i^{ex} =
\begin{bmatrix}
    a_{xi} & 0 & u_{oi} & 0 \\
    0 & a_{yi} & v_{oi} & 0 \\
    0 & 0 & 1 & 0
\end{bmatrix} \begin{bmatrix}
    R_i & t_i \\
    0 & 1
\end{bmatrix}.
\]

Therefore, these cameras’ intrinsic and extrinsic parameters should be determined to compute the position of a point in a 3D space from the images of the binocular vision system. There have been extensive papers on the parameter calibration of stereo cameras. Generally, camera calibration may be classified under two techniques: photogrammetric calibration and self-calibration. The former usually employs specially devised objects whose dimensions are known exactly in advance, and the calibration can be done very efficient. The later instead does not use any calibration object. It only uses image information at several camera locations in a static scene to compute camera parameters in real time. This method involves too many parameters to estimate, and thus is not very reliable. As mentioned earlier, the pan and tilt angles of each binocular camera are changeable, therefore, the photo-
grammetric calibration method cannot be applied here directly since it is used only for the stationary camera calibration. On the other side, self-calibration is time-consuming and not suitable to our vision system as well. So, more efficient and applicable algorithms are needed here, and a calibration approach based on Zhang’s method [15] will be studied in the following subsection.

Once the intrinsic and extrinsic parameters are known, we can compute the position of any object with its corresponding pixel positions in the two camera image planes. As can be seen in (9), there are four linear equations to find three position parameters. To solve the equations, a nonlinear minimum method is used here.

As discussed above, the point position computed from (9) is obtained in the world coordinate system. However, for certain robot vision applications, e.g., target track, the position of a target relative to robot is more helpful to achieve a successful operation. Therefore, the position of an object in the world coordinate system should be transformed into the robot coordinate system \( \Sigma_R \). Assuming that the homogeneous transformation from \( \Sigma_R \) to \( \Sigma_W \) is \( T \), we have

\[
\begin{bmatrix}
x_r \\
y_r \\
z_r \\
1
\end{bmatrix} =
T^{-1}
\begin{bmatrix}
x \\
y \\
z \\
1
\end{bmatrix}
\]

(11)

where \([x_r, y_r, z_r]^T\) is the position vector of the target in \( \Sigma_R \), and the transformation matrix \( T \) is varying as the mobile platform is moving, and can be determined from the sensing data of the platform, such as odometer, GPS, etc.

B. Camera Parameter Estimation

To measure the distance of an object with a vision system, it is necessary to explicitly estimate the intrinsic and extrinsic camera parameters. For a camera with fixed focus, its intrinsic parameters do not change, while the extrinsic parameters are variable with the pan and tilt angles. The extrinsic parameters relate the absolute world coordinate system with the relative camera coordinate system. Therefore, if the camera is controlled to pan or tilt on the horizontal or vertical plane, \( R_1 \) and \( t \) will change, so will the perspective matrix \( M_1 \). Among many calibration techniques, Zhang’s method is flexible and reliable, especially very easy to implement with simple planar patterns. It also models the radial lens distortion problem. But, unfortunately it is only used for stationary camera calibration. Here, we propose a new calibration approach on the basis of Zhang’s method, where the two cameras we used are allowed to move horizontally and vertically.

The intrinsic and extrinsic parameters are calibrated as camera is located at the initial pan and tilt angles, using Zhang’s method. Both cameras are placed parallel and their optical axes are perpendicular to the pattern plane, and the axes of the two camera coordinate systems are parallel correspondingly with those of the world coordinate system. Hence, \( R_1 \) is an identity matrix for the initial camera position. Then, as the camera starts moving, the intrinsic parameter matrix \( M_1^{ip} \) does not change according to the previous analysis, while the extrinsic rotational matrix \( R_1 \) and the translational vector \( t_1 \) are varying with the camera panning angle \( \theta_p \) and tilting angle \( \theta_t \).

When we establish the camera coordinate system, generally the optical axis is defined to be \( z_{ci} \), shown in Figure 3, and the horizontal direction \( x_{ci} \) and the vertical direction \( y_{ci} \). Since camera motion causes the variation of the rotational matrix, it is necessary to investigate their relationship. If the camera is controlled to rotate with the \( x_{ci} \) axis, that is to say, the camera moves on the horizontal plane \( X_iOZ_i \), \( \theta_p \) is varying, and \( R_1 \) can be given by

\[
R_1(\theta_p) =
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos(\theta_p) & -\sin(\theta_p) \\
0 & \sin(\theta_p) & \cos(\theta_p)
\end{bmatrix}
\]

(12)

Similarly, as the camera is tilted with the \( y_{ci} \) axis, the rotational matrix is

\[
R_1(\theta_p, \theta_t) =
\begin{bmatrix}
\cos(\theta_t) & 0 & \sin(\theta_t) \\
0 & 1 & 0 \\
-\sin(\theta_t) & 0 & \cos(\theta_t)
\end{bmatrix}
\]

(13)

In (12) and (13), the pan and tilt angles can be measured directly with the shaft encoders. Now, a generic rotational matrix for the pan-tilt camera is thus obtained as

\[
R_1(\theta_p, \theta_t) = R_1(\theta_p)R_1(\theta_t)
\]

(14)

As the camera is in motion, the translational vector \( t \) varies as well, and this vector reflects the position variation of the camera’s optical center. In the pinhole camera model, the optical centre is essentially an imaginary point on the optical axis, and it can be obtained from camera calibration. Now, we attempt to fit the optical centre curve with its experimental data calibrated as the camera is panned or tilted at different locations. Once the curve expression is obtained, it can be then used to compute the optical centre at any camera pose, and we only need to know the camera pan and tilt angles. In [16], we give the curve expression with

\[
\begin{bmatrix}
y_{ci} \\
x_{ci} \\
z_{ci}
\end{bmatrix} =
\begin{bmatrix}
0 \\
k_i\theta_{pi} + b_i \\
\left( x_{ci} - x_{i0} \right)^2 + \left( z_{ci} - z_{i0} \right)^2 = r_i^2
\end{bmatrix}
\]

(15)

where \( k_i, b_i, x_{i0}, z_{i0} \) and \( r_i \) are constants under determination.

Then, it can be seen that \( x_{ci} \) is a linear function of \( \theta_{pi} \), and the third equation of (15) defines an ellipse trajectory.

Now, we calibrate the camera parameters at an interval of \( \Delta \theta_p \) and obtain a set of the optical centre data \( \{x_{ci}, y_{ci}, z_{ci}\} \), \( j = 1, 2, \ldots, n \).

Using the experimental data, \( x_{i0}, z_{i0} \) and \( r_i \) can be determined with the following objective function,

\[
Q(\gamma, \zeta, \xi) = \min \left( \sum_{j=1}^{n} \left( x_{ci}^2 + y_{ci}^2 + x_{ci} + y_{ci} + z_{ci} + \xi_i \right)^2 \right)
\]

where \( \gamma = 2x_{i0}, \zeta = 2z_{i0}, \xi = x_{i0}^2 + z_{i0}^2 + r_i^2 \).
In the same way, from the data set of the optical centre and the pan angle \( \theta_p \), \( k_1 \) and \( b_1 \) can be obtained from the least square criterion below:

\[
O(k_i, b_i) = \min \sum (x_i - \left(k_i \cdot \theta_p + b_i\right))^2
\]

When these constants of \( k_i, b_i, x_0, z_0 \) and \( r_i \) are determined, we can estimate the extrinsic parameter matrix of a camera with (15) for any pan angle. Similarly, the expression of the optical centre can be given as the tilt angle varies. Now, we are able to provide a complete estimation of the camera parameters for any camera pan and tilt angle.

C. Experimental Results

The binocular vision system has two pan-tilt SONY CCD cameras attached on the head of the humanoid mobile platform. The range of the camera’s pan angle is from -100° to 100°, and the tilt angle of each camera is in [-25°, 25°]. The intrinsic and extrinsic parameters at different camera poses are calibrated with Zhang’s method. As we know, the intrinsic parameters of a camera depend on its optical and electronic components. Since the focus of the camera used is constant, the intrinsic parameters do not change as the camera pose is varying, while the extrinsic parameters are changeable with the camera motion. In the following, the calibrated intrinsic parameter matrices are given,

\[
M_{1}^{in} = \begin{bmatrix}
273.6410 & 0.0000 & 167.7659 \\
0.0000 & 272.0501 & 116.7224 \\
0.0000 & 0.0000 & 1.0000 \\
\end{bmatrix}
\]

and

\[
M_{2}^{in} = \begin{bmatrix}
272.6244 & 0.0000 & 176.9527 \\
0.0000 & 267.5410 & 108.8058 \\
0.0000 & 0.0000 & 1.0000 \\
\end{bmatrix}
\]

Although the two cameras of the binocular vision system are identical, the calibrated results are slight different. This may be caused by the calibration inaccuracies and the camera lenses’ deflections.

As discussed earlier, \( R \) can be computed at the different pan and tilt angles with their measured values. Here, we analyze the camera’s translational vector \( t \), and only the results of the left camera of the binocular system are given below. Figure 4 shows the calibrated data and the fitted curve of the optical centre as the left camera rotates horizontally. The expression constants of the optical centre trajectory are \( k=-9.77, b=35.79, x_0=-36.36, z_0=187.02, r=742.15 \). Figure 5 gives the experimental results as the left camera is controlled to tilt on the vertical plane, and the constants of the optical centre expression are \( k'=10.08, b'=25.69, y_0=-14.70, z_0'=129.92, r'=681.13 \).

Figure 4. The experimental results as camera rotating horizontally.

Figure 5. The experimental results as camera titling vertically.

Now, we can measure the distance of an object or target moving in front of the humanoid mobile platform, using the calibrated cameras. Even as the object is out of the view of the two cameras, their pan and tilt angles are adjusted adaptively. In our experiments, we place an object at different points whose positions are known a priori in a 3D space, and then measure its distance with the binocular vision system. To evaluate the precision of the position measurement, numerous comparison experiments have been done, and the mean-root-square error of the actual and measured values is analyzed, and we find it is less than 0.05 cm.

IV. CONCLUSIONS

In this paper, new approaches to track and measure a moving object in a 3D space are proposed with a binocular vision system. The adaptive BS algorithm is used here to segment and update the object model in real time, and is integrated with a Camshift searching algorithm to track the moving object successfully in a video sequence. To precisely measure the object distance with the binocular vision system,
an effective estimation method, based on Zhang’s camera calibration method, is proposed to calibrate the intrinsic and extrinsic parameters of a pan-tilt camera, which is allowed to pan and tilt on the horizontal and vertical planes. Hence, a moving object can be successfully tracked and precisely measured in the video sequence as it is outside the view of the cameras. Experimental results show the proposed approaches work very well on a humanoid mobile platform.

ACKNOWLEDGMENT

This work is supported by the funds from and Beijing Jiaotong University and the State Key Laboratory of Robotics, Shenyang Institute of Automation, CAS.

REFERENCES

[1] C. A. Richards and N. P. Papanikolopoulos, “The automatic detection and visual tracking of moving objects by eye-in-hand robotic systems,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 1995, pp.228-235.

[2] Z. W. Kim, “Real time object tracking based on dynamic feature grouping with background subtraction,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, AK, 2008, pp.1-8.

[3] M. Asada, T. Tanaka and K. Hosoda, “Adaptive binocular visual servoing for independently moving tracking,” in Proceedings of the IEEE International Conference on Robotics and Automation, CA, 2000, pp. 2076-2081.

[4] P. K. Allen, A. Timcenko, B. Yoshimi and P. Michelman, “Automated tracking and grasping of a moving object with a robot hand-eye system,” IEEE Transactions on Robotics and Automation, 9(2), 1993, pp. 152-165.

[5] M. B. Capellides, D. Doermann and R. Challappa, “An appearance based approach for human and object tracking,” in Proceedings of the IEEE International Conference on Image Processing, 2003, pp.85-88.

[6] L. Makkalena and A. Petrosino, “A self-organizing approach to background subtraction for visual surveillance applications,” IEEE Transactions on Image Processing, 17(7), 2008, pp. 1168-1177.

[7] F. E. Alsagre and B. Z. Yuan, “Multiple moving objects tracking for video surveillance systems,” in Proceedings of International Conference on Signal Processing, 2004, pp.1301-1305.

[8] K. Fukunaga and L. Hostetler, “The estimation of the gradient of a density function, with application in pattern recognition,” IEEE Transactions on Information Theory, 21(1), 1975, pp.32-40.

[9] D. Comaniciu and V. Ramesh, “Mean shift and optimal prediction for efficient object tracking,” in Proceedings of the International Conference on Image Processing, 2000, pp. 70-73.

[10] A. Zhao, M. J. Brooks and A. R. Dick, “Enhanced appearance models for object tracking,” in Proceedings of the 4th International Conference on Image and Graphics, 2007, pp. 513-518.

[11] X. P. Chen, S. S. Yu and Z. L. Ma, “An improved mean shift algorithm for moving object tracking,” in Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqi, China, 2008, pp.5111-5114.

[12] K. Deguchi, O. Kawamura and T. Okatani, “Object tracking by the mean-shift of regional color distribution combined with the particle-filter algorithm,” in Proceedings of the 17th International Conference on Pattern Recognition, 2004, pp. 506-509.

[13] G. R. Bradski, “Real time face and object tracking as a component of a perceptual user interface,” in Proceedings of IEEE Workshop on Applications of Computer Vision, 1998, pp. 214-219.

[14] M. Joengel, H. Mellmann and M. Spranger, “Improving vision-based distance measurements using reference objects,” in RoboCup 2007, Lecture Notes in Artificial Intelligence, Springer, 2007.

[15] Z. Y., Zhang, “A flexible new technique for camera calibration,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11), 2000, pp.1330-1334.

[16] K. Zhu, “Tracking and measuring a moving object with a binocular vision system,” Master Thesis, Beijing Jiaotong University, 2008.