International Journal of Automation and Computing 14(2), April 2017, 159-168
DOI: 10.1007/s11633-016-1047-6

A New Monocular Vision Measurement Method to Estimate 3D Positions of Objects on Floor

Ling-Yi Xu[1,2] Zhi-Qiang Cao[1,2] Peng Zhao[1,2] Chao Zhou[1,2]

[1]State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

[2]University of Chinese Academy of Sciences, Beijing 101408, China

Abstract: A new visual measurement method is proposed to estimate three-dimensional (3D) position of the object on the floor based on a single camera. The camera fixed on a robot is in an inclined position with respect to the floor. A measurement model with the camera′s extrinsic parameters such as the height and pitch angle is described. Single image of a chessboard pattern placed on the floor is enough to calibrate the camera′s extrinsic parameters after the camera′s intrinsic parameters are calibrated. Then the position of object on the floor can be computed with the measurement model. Furthermore, the height of object can be calculated with the paired-points in the vertical line sharing the same position on the floor. Compared to the conventional method used to estimate the positions on the plane, this method can obtain the 3D positions. The indoor experiment testifies the accuracy and validity of the proposed method.

Keywords: Visual measurement, calibration, localization, position estimation, monocular vision.

1 Introduction

Vision systems provide abundant information for mobile robots to realize localization and navigation[1, 2]. They can be classified into three categories: monocular, binocular, and multi-camera vision systems. In mobile robot community, monocular and binocular vision systems are the typical configurations for measurement. Binocular vision system consists of two cameras, which can be used to obtain the three-dimensional (3D) positions of targets. Se et al.[3] proposed a method to localize and build 3D maps of unmodified environments by using scale-invariant visual landmarks. In [4], a stereo vision system was employed for mobile robot navigation by building occupancy grid maps of the environment. A kinect 3D sensor was used to detect obstacles in a greenhouse environment[5]. 3D position information can be easily obtained by binocular vision system. Nevertheless, the feature matching from two images captured by two cameras is time-consuming, which affects the system′s real-time performance. Therefore, many researchers turn to monocular vision system to obtain a cheap and simple vision system as well as to reduce time cost[6−21]. In monocular vision systems, two kinds of cameras can be found. One is omnidirectional camera[6, 7] and another is normal one[8−18]. The field of view of an omnidirectional camera is large and the detection range can reach 360°. Goedemé et al.[6] used an omnidirectional camera to build topologically organized environment maps of a complex, natural environment for navigation. Do et al.[7] implemented the position estimation based on a maximum likelihood estimation with a map from optimally selected visual features using Gaussian process regression. One drawback of this kind of camera is that large distortions can reduce precision. Normal cameras are more popular than omnidirectional camera. Yin et al.[8] used a monocular camera to calculate the position of points on the ground. It has a good accuracy with tiny errors, however, it cannot estimate the height of objects above the floor. Zhao et al.[9] used a monocular camera to estimate the relative pose of the robot and the docking station with the premise of known identifier. Bresson et al.[10] proposed a monocular simultaneous localization and mapping (SLAM) method. In [11], a monocular SLAM system was presented, which employs a particle filter and top-down search to allow real-time performance while mapping large numbers of landmarks. A real-time algorithm based on a sparse but persistent map of natural landmarks was proposed to recover the 3D trajectory of a monocular camera[12]. It was pointed out in [13] that the accuracy of the reconstruction is increased with a calibrated camera. In the reconstruction, a set of key frames is extracted from the reference sequence to compute the camera pose by using interest points matched between key frames. A real-time image-based localization method was presented in [14], which estimates six-degree-of-freedom camera pose from input video by efficiently tracking natural features and matching them to the 3D points reconstructed offline. Campbell et al.[20] described a monocular robot vision system, which

Research Article
Manuscript received December 2, 2015; accepted June 7, 2016; published online February 21, 2017
This work was supported by National Natural Science Foundation of China (Nos. 61273352 and 61473295), National High Technology Research and Development Program of China (863 Program) (No. 2015AA042307), and Beijing Natural Science Foundation (No. 4161002).
Recommended by Associate Editor Xun Chen

accomplishes accurate dead-reckoning, closed-loop motion control, and precipice and obstacle detection. Zhou and Li[21] presented a homography-based approach to detect the ground plane from monocular sequences provided by a robot. It shall be noted that the methods with monocular vision estimate 3D positions of objects based on multiple images captured in sequence or realize 2D position measurement based on single image. If 3D information can be obtained from single image, the effectiveness of rebuilding will be increased. It is sure that the 3D measurement with single image for monocular vision is helpful to improve the performance of the system.

The monocular vision system can be used to measure the depth even the 3D position of the objects on the ground[22, 23] based on the height and pitch angle of the camera. In [22], the mapping relation between image row pixel values and the actual imaging angles is established with the image of a vertical target placed in the front of the camera. The obstacle′s depth information is extracted in real time by combining the projection geometry model. Based on the height and inclination angle of the camera, a three-dimensional position measurement model is established[23]. The distance from a point on the ground to the bottom of the camera is calculated. Then the height and area of the target objects on the ground are computed based on the pinhole imaging model and the measured distance. It is sure that the accurate height and inclination angle of the camera are helpful to improve the measurement accuracy[23].

In this paper, a novel monocular method is presented to estimate 3D positions of targets on the ground. Two parameters used in this method are the height h of the camera′s optical center and the pitch angle θ of camera′s optical axis. The 3D position of a point over the ground can be estimated in a single measurement combined with the image coordinate of its paired point on the ground. The paired points are the points in a vertical line sharing the same coordinate (x, y).

The rest of this paper is organized as follows. In Section 2, the visual measurement model is presented. Section 3 presents the calibration principle in theory and the calibration method in practice. Error analysis is given in Section 4. Experiments are provided in Section 5, and Section 6 concludes the paper.

2 Visual measurement model

The camera is located on the top of the robot, looking down from above[20], as shown in Fig. 1 (a). The origin O_c of camera coordinate system is at the optical center. The axis X_c is parallel to the horizontal axis U in the image. The axis Y_c is parallel to vertical axis V in the image. The optical axis pointing to the scene is selected as the axis Z_c. The vertical line passing the O_c intersects the ground at O_w, as the origin of robot coordinate system. The axis X_w is parallel to X_c. The axis Z_w is heading to O_c along the vertical line O_wO_c. The axis Y_w is assigned by right-hand rule. The relationship between the camera coordinates and the robot coordinates is shown in Figs. 1 (b) and 1 (c).

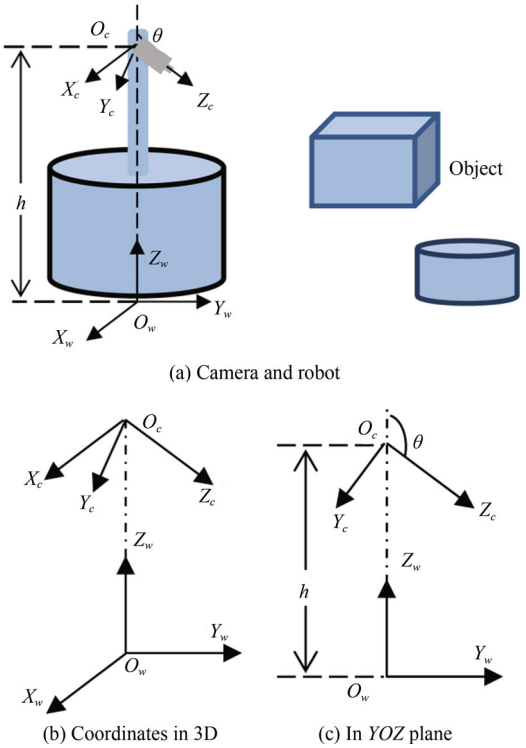


Fig. 1 Camera coordinates and robot coordinates

The camera system transforms to the world system by rotating θ around the axis X_c then translating–h along the axis Z_c. In this way, the extrinsic parameter matrix ${}^W_C T$ is

$$ {}^W_C T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & h\sin\theta \\ 0 & \sin\theta & \cos\theta & -h\cos\theta \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1} $$

where h is the distance from O_w to O_c, θ is the angle between axes Z_w and Z_c.

For a point P_i in the Cartesian space, its position in the normalized imaging plane can be computed as

$$ \begin{pmatrix} z_{ci}x_{1ci} \\ z_{ci}y_{1ci} \\ z_{ci} \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & h\sin\theta \\ 0 & \sin\theta & \cos\theta & -h\cos\theta \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{wi} \\ y_{wi} \\ z_{wi} \\ 1 \end{pmatrix} \tag{2} $$

where $(x_{wi},\ y_{wi},\ z_{wi})$ is the position of P_i in the robot system, $(x_{1ci},\ y_{1ci},\ 1)$ is the imaging position of P_i in the normalized imaging plane in the camera system, z_{ci} means the coordinate of P_i on the axis Z_c in the camera system.

Formula (2) can be rewritten as

$$ \begin{cases} z_{ci}x_{1ci} = x_{wi} \\ z_{ci}y_{1ci} = y_{wi}\cos\theta - z_{wi}\sin\theta + h\sin\theta \\ z_{ci} = y_{wi}\sin\theta + z_{wi}\cos\theta - h\cos\theta. \end{cases} \tag{3} $$

Substituting z_{ci} in the third equation to the other two equations in (3), we have

$$\begin{cases} x_{wi} - x_{1ci}y_{wi}\sin\theta - x_{1ci}z_{wi}\cos\theta = \\ \qquad -x_{1ci}h\cos\theta \\ (y_{1ci}\sin\theta - \cos\theta)y_{wi} + (\sin\theta + y_{1ci}\cos\theta)z_{wi} = \\ \qquad h\sin\theta + y_{1ci}h\cos\theta. \end{cases} \tag{4}$$

For the point on the plane $X_wO_wY_w$, we have Z_{wi}=0. In this case, formula (4) can be changed to (5).

$$\begin{cases} x_{wi} = x_{1ci}y_{wi}\sin\theta - x_{1ci}h\cos\theta \\ y_{wi} = \dfrac{h\sin\theta + y_{1ci}h\cos\theta}{y_{1ci}\sin\theta - \cos\theta}. \end{cases} \tag{5}$$

It is obvious that (5) can be used to estimate the positions of points in Cartesian space with the known parameters h and θ and the imaging coordinate $(x_{1ci},\ y_{1ci})$ in the condition $Z_{wi}=0$. $(x_{1ci},\ y_{1ci})$ can be calculated with (6) when the image coordinate $(u_i,\ v_i)$ and the intrinsic parameters are known.

$$\begin{bmatrix} x_{1ci} \\ y_{1ci} \\ 1 \end{bmatrix} = \begin{bmatrix} \dfrac{1}{k_x} & 0 & -\dfrac{u_0}{k_x} \\ 0 & \dfrac{1}{k_y} & -\dfrac{v_0}{k_y} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \tag{6}$$

where k_x and k_y are the magnification factors, $(u_0,\ v_0)$ is the image coordinate of the principal point of the camera.

In addition, the coordinate z_{wi} of the points in the vertical line sharing the same coordinate $(x_{wi},\ y_{wi})$ can be estimated using (7) with their imaging coordinate once the shared coordinate $(x_{wi},\ y_{wi})$ is calculated.

$$z_{wi} = \frac{h\sin\theta + y_{1ci}h\cos\theta - (y_{1ci}\sin\theta - \cos\theta)y_{wi}}{(\sin\theta + y_{1ci}\cos\theta)}. \tag{7}$$

3 Calibration

The parameters h and θ are needed for the visual measurement in (5) and (7). Formula (5) can be used as a way to calibrate h and θ, by known $(x_{wi},\ y_{wi})$ and the matched $(x_{1ci},\ y_{1ci})$. But there are larger errors in the positions of given points in the robot coordinates since they are manually measured and the origin is invisible. Fortunately, the relative positions of the points in a pattern are accurate. Hence, the parameters h and θ can be calculated using the relative position between P_i and P_{i-1}. From the second equation in (5), we have

$$\begin{aligned} &y_{wi} - y_{w(i-1)} = \\ &\quad h\frac{\sin\theta + y_{1ci}\cos\theta}{y_{1ci}\sin\theta - \cos\theta} - h\frac{\sin\theta + y_{1c(i-1)}\cos\theta}{y_{1c(i-1)}\sin\theta - \cos\theta} \end{aligned} \tag{8}$$

where y_{wi} and $y_{w(i-1)}$ represent the positions of P_i and P_{i-1} in the robot coordinates, y_{1ci} and $y_{1c(i-1)}$ are the imaging positions of P_i and P_{i-1} in the camera coordinates.

With trigonometric transform, formula (8) can be rewritten as

$$\begin{aligned} &\Delta y_{wi}(y_{1ci} + y_{1c(i-1)})\sin 2\theta - \\ &\quad \Delta y_{wi}(1 - y_{1ci}y_{1c(i-1)})\cos 2\theta - \\ &\quad 2(y_{1ci} - y_{1c(i-1)})h = \Delta y_{wi}(1 + y_{1ci}y_{1c(i-1)}) \end{aligned} \tag{9}$$

where Δy_{wi} is relative position, $\Delta y_{wi} = y_{wi} - y_{w(i-1)}$.

Obviously, (9) is a linear equation with three unknown variables $\sin 2\theta$, $\cos 2\theta$ and h. The variables can be solved with three equations at least. In other words, four points are enough to calibrate the parameters θ and h with linear equations. But plenty of points are helpful to improve the estimation accuracy of h and θ with the least square (LS) method. With n points, whose relative positions are known, $n-1$ equations as given in (9) can be obtained. The LS method is used to solve the variables $\sin 2\theta$, $\cos 2\theta$ and h, then the rotation angle θ can be obtained.

In practice, however, it is difficult to let the directions of chessboard coordinates same as the directions of axes X_w and Y_w. Suppose the angle between the horizontal direction of chessboard and the axis X_w is α. Then

$$\begin{cases} x_w = x_p\cos\alpha + y_p\sin\alpha \\ y_w = -x_p\sin\alpha + y_p\cos\alpha \end{cases} \tag{10}$$

where $(x_p,\ y_p)$ is the corner′s coordinate in the chessboard coordinates.

The differential of second equation in (10) gives

$$\Delta y_w = -\Delta x_p\sin\alpha + \Delta y_p\cos\alpha. \tag{11}$$

Substituting (11) into (9), we have

$$\begin{aligned} &-\Delta x_p(y_{1ci} + y_{1c(i-1)})\tan\alpha\sin 2\theta + \\ &\quad \Delta y_p(y_{1ci} + y_{1c(i-1)})\sin 2\theta + \\ &\quad \Delta x_p(1 - y_{1ci}y_{1c(i-1)})\tan\alpha\cos 2\theta - \\ &\quad \Delta y_p(1 - y_{1ci}y_{1c(i-1)})\cos 2\theta - \\ &\quad \frac{2(y_{1ci} - y_{1c(i-1)})h}{\cos\alpha} + \\ &\quad \Delta x_p(1 + y_{1ci}y_{1c(i-1)})\tan\alpha = \\ &\quad \Delta y_p(1 + y_{1ci}y_{1c(i-1)}). \end{aligned} \tag{12}$$

For a pair points in a column on the chessboard, Δx_p=0. In this case, (12) is rewritten as (13).

$$\begin{aligned} &\Delta y_p(y_{1ci} + y_{1c(i-1)})\sin 2\theta - \\ &\quad \Delta y_p(1 - y_{1ci}y_{1c(i-1)})\cos 2\theta - \\ &\quad \frac{2(y_{1ci} - y_{1c(i-1)})h}{\cos\alpha} = \\ &\quad \Delta y_p(1 + y_{1ci}y_{1c(i-1)}). \end{aligned} \tag{13}$$

For a pair points in a row on the chessboard, Δy_p=0. In this case, (12) is rewritten as (14). Furthermore, (15) is obtained from (14).

$$\begin{aligned} &-\Delta x_p(y_{1ci} + y_{1c(i-1)})\tan\alpha\sin 2\theta + \\ &\quad \Delta x_p(1 - y_{1ci}y_{1c(i-1)})\tan\alpha\cos 2\theta + \\ &\quad \Delta x_p(1 + y_{1ci}y_{1c(i-1)})\tan\alpha = \\ &\quad \frac{2(y_{1ci} - y_{1c(i-1)})h}{\cos\alpha}. \end{aligned} \tag{14}$$

It can be found that an equation as given in (13) can be obtained with a pair points in a column on the chessboard. n equations are formed with n pairs of points, in which each pair of points is in the same column on the chessboard. The three unknown variables $\sin 2\theta$, $\cos 2\theta$ and $\frac{h}{\cos\alpha}$ in (13) can be solved via the LS method with at least three equations as given in (13). After $\sin 2\theta$, $\cos 2\theta$ and $\frac{h}{\cos\alpha}$ are obtained, $\tan\alpha$ is computed with (15) by using a pair of points in a row on the chessboard. Then α is gotten and h is computed from $\frac{h}{\cos\alpha}$. Of course, more pairs of points are helpful to improve the accuracy of θ, α and h.

4 Error analysis

Here, we analyze the influence of variables h, θ, x_{1ci} and y_{1ci} on errors of x_{wi} and y_{wi} in the condition z_{wi}=0. The total differential $\mathrm{d}y_{wi}$ is derived from the second equation in (5), and then the total differential $\mathrm{d}x_{wi}$ is obtained from the first equation in (5) according to $\mathrm{d}y_{wi}$. The expressions of $\mathrm{d}x_{wi}$ and $\mathrm{d}y_{wi}$ are given in (16). Substituting (6) to (16), we have (17).

The error's departure can be estimated from (17). Generally, the pitch angle is 100° approximately. Since the feature points are below the center, the image coordinate $u-u_0$ <300 pixels and $v-v_0$ <500 pixels. Assume k_x = 1 600, k_y = 1 600. The unit variation of h, θ, u and v cost different to $\mathrm{d}x_{wi}$ and $\mathrm{d}y_{wi}$. In the conditions above, the error $\mathrm{d}x_{wi}$ is −6.6 mm and $\mathrm{d}y_{wi}$ is −42.7 mm if $\mathrm{d}h$ = 0.1 mm and $\mathrm{d}\theta$=0.5°(about 0.009 rad) with sub-pixel variations of u and v. In the errors $\mathrm{d}x_{wi}$ and $\mathrm{d}y_{wi}$, the main part is from the error $\mathrm{d}\theta$. In detail, 0.5° departure of θ (around 100°) contributes about −6.6 mm of $\mathrm{d}x_{wi}$ and −42.6 mm of $\mathrm{d}y_{wi}$, more than 99% in total error. It is obvious that a tiny variation of θ will result in large shift of image coordinate v. In the case that θ and h are accurate, the errors $\mathrm{d}u$ and $\mathrm{d}v$ are the main source of the errors $\mathrm{d}x_{wi}$ and $\mathrm{d}y_{wi}$ for the points whose $v-v_0$ is small.

$$\tan\alpha=\frac{\dfrac{2(y_{1ci}-y_{1c(i-1)})h}{\cos\alpha}}{\Delta x_p[-(y_{1ci}+y_{1c(i-1)})\sin 2\theta+(1-y_{1ci}y_{1c(i-1)})\cos 2\theta+(1+y_{1ci}y_{1c(i-1)})]} \tag{15}$$

$$\begin{cases}\mathrm{d}y_{wi}=\dfrac{\sin\theta+y_{1ci}\cos\theta}{y_{1ci}\sin\theta-\cos\theta}\mathrm{d}h+\dfrac{-h}{(y_{1ci}\sin\theta-\cos\theta)^2}\mathrm{d}y_{1ci}+\dfrac{-y_{1ci}^2h-h}{(y_{1ci}\sin\theta-\cos\theta)^2}\mathrm{d}\theta\\ \mathrm{d}x_{wi}=\left[\dfrac{h-h\cos 2\theta+y_{1ci}h\sin 2\theta}{2(y_{1ci}\sin\theta-\cos\theta)}-h\cos\theta\right]\mathrm{d}x_{1ci}+\left[\dfrac{x_{1ci}-x_{1ci}\cos 2\theta+y_{1ci}x_{1ci}\sin 2\theta}{2(y_{1ci}\sin\theta-\cos\theta)}-x_{1ci}\cos\theta\right]\mathrm{d}h+\\ \quad\left[x_{1ci}h\sin\theta+\dfrac{x_{1ci}y_{1ci}h\sin 2\theta\sin\theta-x_{1ci}y_{1ci}h\cos 2\theta\cos\theta+\left(2y_{1ci}^2+1\right)x_{1ci}h\cos 2\theta\sin\theta}{(y_{1ci}^2+1)+(1-y_{1ci}^2)\cos 2\theta-2y_{1ci}\sin 2\theta}-\right.\\ \quad\left.\dfrac{\left(y_{1ci}^2+2\right)x_{1ci}h\sin 2\theta\cos\theta+(y_{1ci}\cos\theta+\sin\theta)x_{1ci}h}{(y_{1ci}^2+1)+(1-y_{1ci}^2)\cos 2\theta-2y_{1ci}\sin 2\theta}\right]\mathrm{d}\theta\end{cases} \tag{16}$$

$$\begin{cases}\mathrm{d}y_{wi}=\dfrac{k_y\sin\theta+(v-v_0)\cos\theta}{(v-v_0)\sin\theta-k_y\cos\theta}\mathrm{d}h+\dfrac{-k_yh}{[(v-v_0)\sin\theta-k_y\cos\theta]^2}\mathrm{d}v+\dfrac{-\left[(v-v_0)^2+k_y^2\right]h}{[(v-v_0)\sin\theta-k_y\cos\theta]^2}\mathrm{d}\theta\\ \mathrm{d}x_{wi}=\left[\dfrac{k_yh-k_yh\cos 2\theta+(v-v_0)h\sin 2\theta}{2k_x[(v-v_0)\sin\theta-k_y\cos\theta]}-\dfrac{h\cos\theta}{k_x}\right]\mathrm{d}u+\\ \quad\left[\dfrac{k_y(u-u_0)-(u-u_0)k_y\cos 2\theta+(u-u_0)(v-v_0)\sin 2\theta}{2k_x[(v-v_0)\sin\theta-k_y\cos\theta]}-\dfrac{u-u_0}{k_x}\cos\theta\right]\mathrm{d}h+\\ \quad\left[\dfrac{u-u_0}{k_x}h\sin\theta+\dfrac{-(u-u_0)(v-v_0)k_yh\cos 3\theta}{k_x\left[(v-v_0)^2+k_y^2\right]+k_x\left[k_y^2-(v-v_0)^2\right]\cos 2\theta-2k_xk_y(v-v_0)\sin 2\theta}+\right.\\ \quad\dfrac{(u-u_0)h\left\{(v-v_0)^2\cos 2\theta\sin\theta-k_y^2\sin 2\theta\cos\theta-\left[(v-v_0)^2+k_y^2\right]\sin\theta\right\}}{k_x\left[(v-v_0)^2+k_y^2\right]+k_x\left[k_y^2-(v-v_0)^2\right]\cos 2\theta-2k_xk_y(v-v_0)\sin 2\theta}-\\ \quad\left.\dfrac{\left[(v-v_0)k_y\cos\theta+k_y^2\sin\theta\right](u-u_0)h}{k_x\left[(v-v_0)^2+k_y^2\right]+k_x\left[k_y^2-(v-v_0)^2\right]\cos 2\theta-2k_xk_y(v-v_0)\sin 2\theta}\right]\mathrm{d}\theta\end{cases} \tag{17}$$

5 Experiment

The camera used in the experiments was Daheng MER-130-30UC, whose image size was 1 280×1 024 pixels. The monocular camera was fixed on the top of the robot, looking down to the ground. The black and white chessboard with 15×15 blocks was put on the ground, in front of the robot, offering plenty of points in Cartesian space. Each block in the chessboard was with the size of 50 mm×50 mm. There were few desks and other objects placed randomly in front of and beside the chessboard. The points on the edge of desks and objects were used to test the validity of the 3D position estimation in space.

5.1 Calibration

Firstly, the camera's intrinsic parameters were calibrated with the chessboard using Matlab calibration tool box. The results of the intrinsic parameters were as follows.

k_x=1 624.339 59, k_y=1 623.036 60, u_0=634.769 07,

v_0=499.017 88.

The lens distortion parameters were [−0.070 21, 0.073 48, −0.001 12, 0.002 67].

There were 225 points on the chessboard in camera view to estimate h and θ. The calibration scene and the image captured by the camera were given in Fig. 2. It can be seen from Fig. 2 that the image blocks on the chessboard become smaller and smaller from bottom to top. Since the blocks from line 11 to 15 were far from robot, the block length in Y direction was about 10 pixels. In other words, a pixel mismatch of corners was about 10% departure. Therefore, the estimation of h and θ was based on 150 corner points from the blocks from line 1 to 10. The corner points were extracted in sub-pixel resolution with Matlab function.

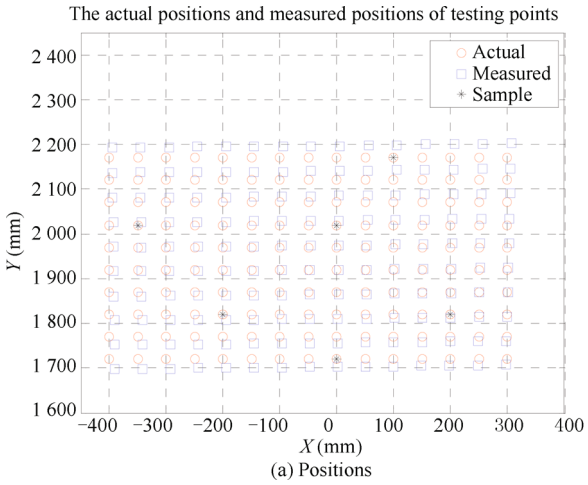
(a) Scene

(b) Image captured by the camera

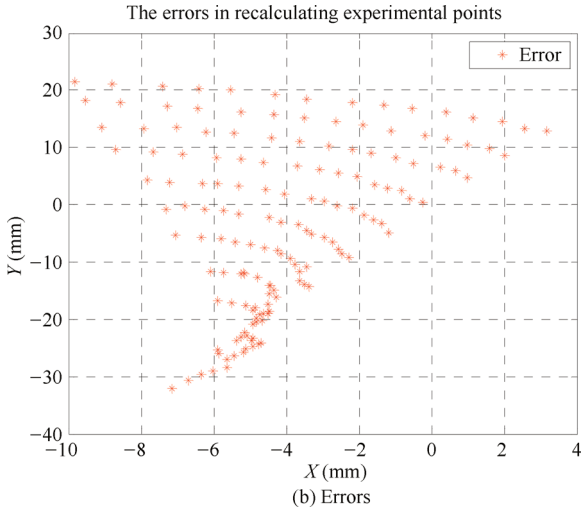
Fig. 2 Calibration scene

1) Calibration using (9) with 6 points

Assume the directions of chessboard coordinates were as same as the directions of axes X_w and Y_w. In other words, assume $x_w = x_p$ and $y_w = y_p$. Equation (9) was used to calibrate θ and h. Considering the minor change of θ causing major fluctuation of h, the partial pivoting was used to estimate θ. The estimation of h was based on θ with (5).



(a) Positions



(b) Errors

Fig. 3 The positions and errors after counterbalance with the parameters calibrated from 6 points

It is sure that fewer points can be used to calibrate the model. In fact, 4 points are enough in theory. Firstly, the parameters h and θ were calibrated based on 6 points. The calibration results are as follows. The estimation value of θ was 1.777 4 rad (about 101.8°). The height h was about 874.6 mm. Then, all of 150 points were used to test the accuracy of results after the estimation of h and θ with 6 points. The recalculated positions of these points were considered measured positions, they are shown with the actual positions in Fig. 3 (a). The red circles "o" are the actual positions, the blue squares "□" are the recalculated positions, and the black stars "∗" are sample points used in calibration. The distribution of errors between the measured and actual positions is given in Fig. 3 (b). The value range was [1 720 mm, 2 170 mm] and the error range was [−32.1 mm, 21.5 mm] for y_{wi}. For x_{wi}, its value range was [−400 mm, 300 mm] and the error range was [−9.8 mm, 3.2 mm]. The error mean in X_w-axis and Y_w-axis was (−4.1 mm, −2.7 mm). The variance of the errors was (6.401 8, 210.990 6). The relative errors in X_w-axis and Y_w-axis were defined as the rate of the errors with the average of y_{wi}, 1 945 mm. The relative errors in Y_w-axis and X_w-axis were

less than 1.65% and 0.5%, respectively.

2) Calibration using (9) with 50 points

In theory, more points are helpful to improve the calibration accuracy. In this experiment, 50 points including the 6 points in last experiment selected from 150 corner points were used to estimate h and θ in order to reduce the effects of data saturation. After the estimation of h and θ, similar to above experiment, all of 150 points were used to test the accuracy of results.

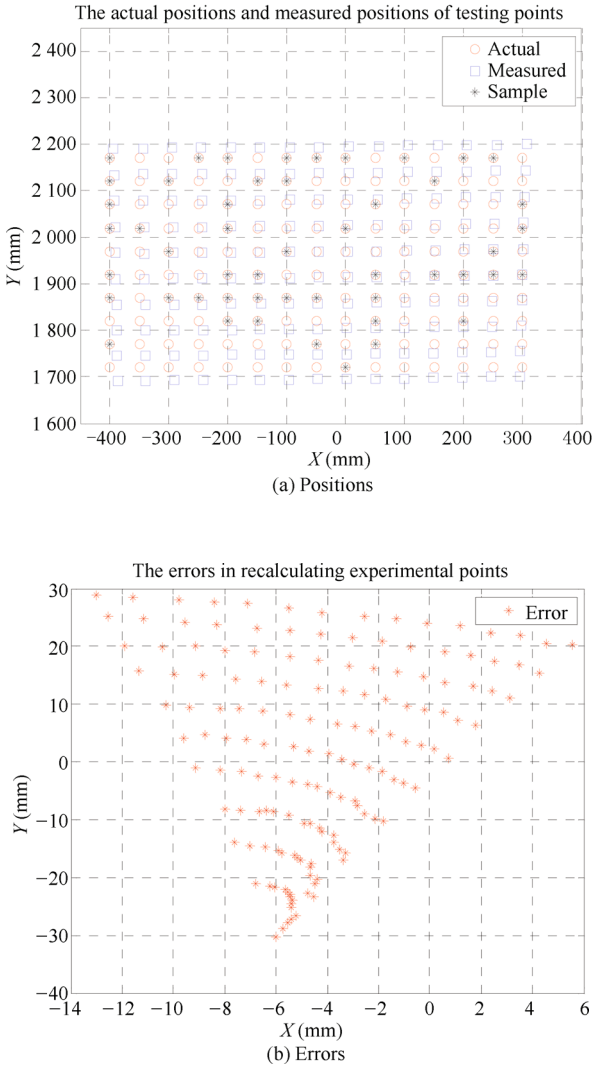


Fig. 4 The positions and errors after counterbalance with the parameters calibrated from 50 points

The calibration results are as follows. The value of θ was 1.769 8 rad (about 101.4°). The height h was about 854.6 mm. Then, the positions of 150 points were recalculated to calculate the errors between the actual positions and measured positions. The results can be found in Fig. 4 (a). The red circles "○" are the actual positions, the blue squares "□" are the recalculated positions, the black stars "∗" are sample points used in calibration. Fig. 4 (b) shows the distribution of these errors. The errors of actual positions and measured positions were in the range of [−30.3 mm, 28.9 mm] for y_{wi} and [−13.0 mm, 5.6 mm] for x_{wi}. The error mean in X_w- and Y_w-axis was (−4.4 mm, 2.3 mm). The variance of the errors was (13.502 1, 264.661 1). The relative errors in Y_w-axis and X_w-axis were less than 1.56% and 0.7%, respectively.

It can be seen that more sample points have little effect to improve the calculation accuracy in the case that the directions of chessboard coordinates were assumed to be as same as the directions of axes X_w and Y_w.

3) Calibration with the proposed method

In this experiment, the angle α between the horizontal direction of chessboard and the axis X_w was taken into account in calibration, that is, equations (13), (5) and (15) were used to calibrate θ, h and α. Six pairs of points were selected to solve $\sin 2\theta$, $\cos 2\theta$ and $\frac{h}{\cos\alpha}$ with (13), in which each pair of points was in the same column on the chessboard. $\frac{h}{\cos\alpha}$ was finely estimated with (5) after θ was obtained. 3 pairs of points were selected to solve $\tan\alpha$ with (15), in which each pair of points was in the same row on the chessboard. The estimation value of θ was 1.835 4 rad (about 105.2°). The height h was about 1 013.0 mm. The angle α was −0.011 9 rad (about −0.68°). The recalculated positions were considered measured positions, which are shown with the actual positions in Fig. 5 (a). The red circles "○" are the actual positions, the blue squares "□" are the recalculated positions, the black stars "∗" are paired-point samples A used in calibration with (13), and the black triangles "Δ" are point samples B used in calibration with (15). The distribution of errors between the measured and actual positions is given in Fig. 5 (b). Compared with former experiments, the results′ accuracy was higher than before. The error range was [−1.6 mm, 1.0 mm] for y_{wi} and [−36.8 mm, −11.6 mm] for x_{wi}. The error mean in X_w- and Y_w-axis was (26.0 mm, 0.0 mm). This error mean can be considered as determinate error. It can also be found from Fig. 5 (a) that there is determinate error in X_w-axis. The determinate error 26.0 mm in X_w-axis was from the offset of manual measurement of benchmark, it can be eliminated by adding it to the actual positions. After the determinate error was corrected, the error range was [−1.6 mm, 1.0 mm] for y_{wi} and [−10.8 mm, 12.4 mm] for x_{wi}. The error mean in X_w-axis and Y_w-axis was (0.03 mm, 0.01 mm). The variance of the errors was (26.811 8, 0.324 8). The relative errors in Y_w-axis and X_w-axis were less than 0.1% and 0.6%, respectively.

Compared with Fig. 3 (b) and Fig. 4 (b), the error distribution in Fig. 5 (d) was more precise and reasonable. Compared with former experiment, the results′ accuracy in this experiment was much higher than before. All of the relative errors were less than 0.6%, which indicates the proposed method is accurate.

5.2 3D measurement experiment

The testing experiment scenery was part of normal indoor office, like Fig. 6. Few corner points in this scenery were used to testify the accuracy of the measurement method. 10 pairs of points listed in Table 1 are shown with

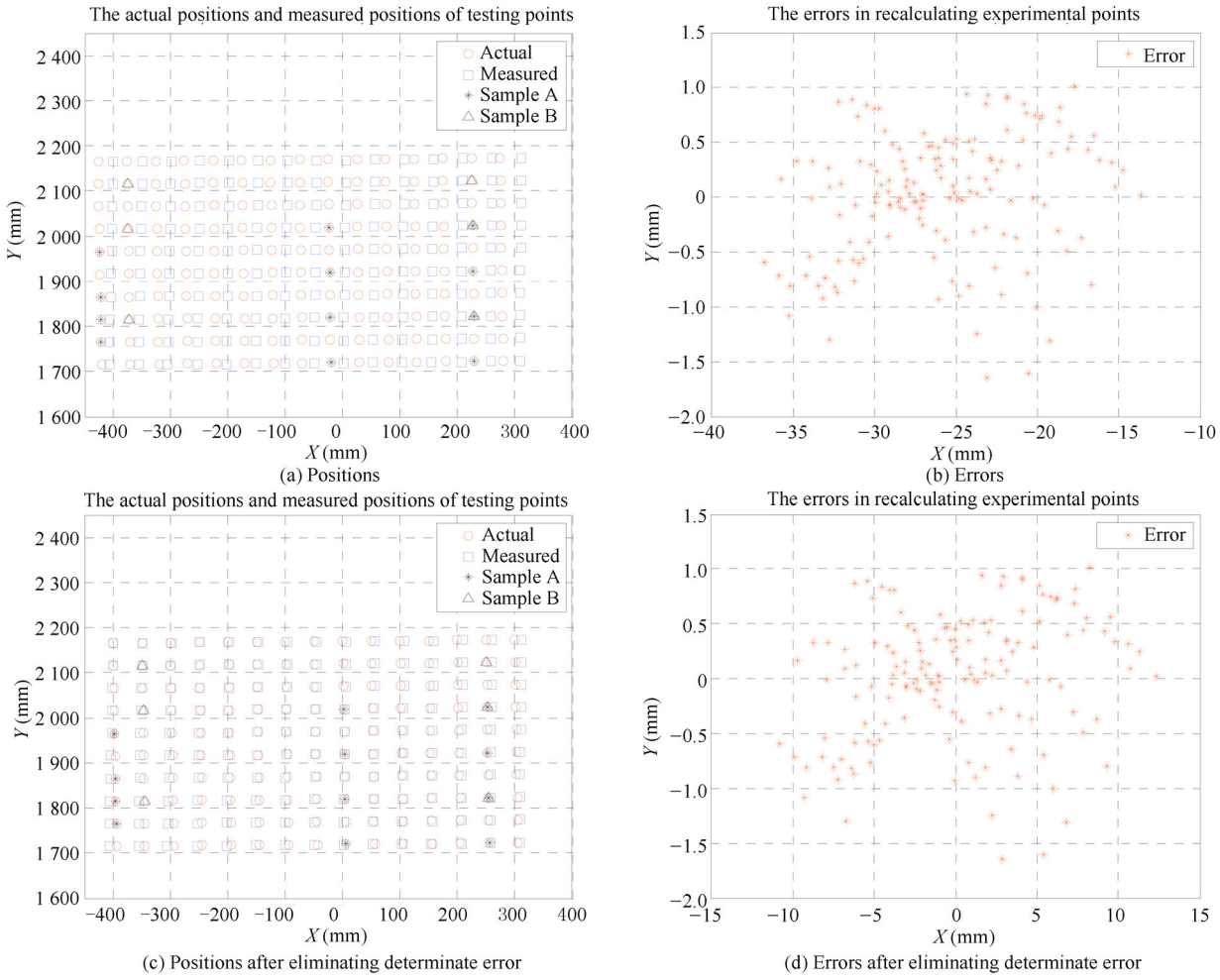


Fig. 5 The positions and errors after counterbalance with the consideration of the yaw angle of chessboard

yellow star "*" to state their positions in Fig. 6. The values of h, θ and α were the calibration results with the proposed method, that is, $\theta = 1.835\,4$ rad and $h = 1\,013.0$ mm, $\alpha = -0.011\,9$ rad. The equation used to calculate height Z_w was (7).

The points on the ground were used to calculate the values of x_w and y_w, which were essential to compute z_w of their paired points. The actual and measured positions of these points are shown in Fig. 7. Fig. 7 (a) shows the results in 3D space, Fig. 7 (b) shows the errors, and Fig. 7 (c) shows the errors in Z_w-axis. The error range of z_w was [−13.2 mm, 1.0 mm]. The relative error in Z_w-axis was defined as the rate of the errors with the height of objects. The relative error in Z_w-axis was less than 1.9%. The measurement results and the related paired points are also listed in Table 1. (x_{wim}, y_{wim}, z_{wim}) is the measurement position. (x_{wia}, y_{wia}, z_{wia}) is the actual position. (x_{wia}, y_{wia}) was transformed from the manual measured position with (10) using the angle α.

It is sure that the errors in height are acceptable to estimate the status of obstacles.

5.3 Comparison experiment

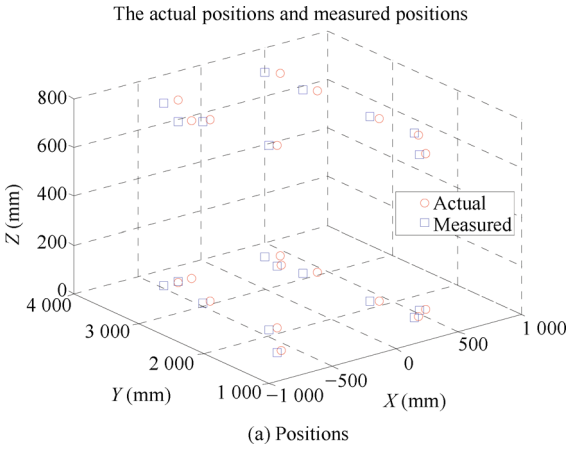
In comparison experiment, the compared method was from [8]. The data used in comparison experiment were the same 150 points sharing the same sequence as in Section 5.1. Firstly, the parameter $\boldsymbol{m}$ in the model in [8] was calibrated with the 50 points same as in Section 5.1. Its result is given in (18). Fig. 8 shows the measurement results of the block corners on the chessboard with the compared method in comparison experiment. The error range was [−4.0 mm, 3.0 mm] for y_{wi} and [−1.0 mm, 0.5 mm] for x_{wi}.

Fig. 6 Indoor testing experiment scene and measurement points

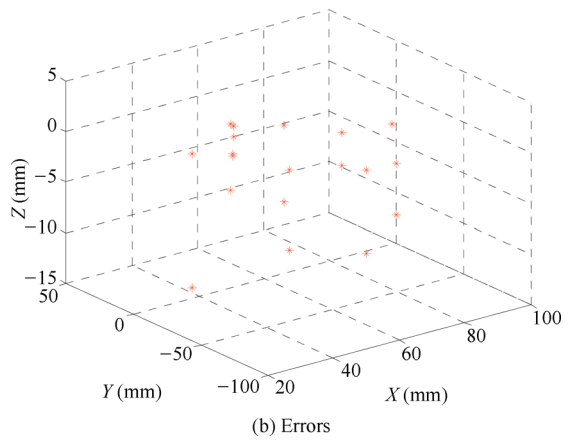
Table 1 Measurement results and the related paired points

Point	u	v	x_{wim} (mm)	y_{wim} (mm)	z_{wim} (mm)	x_{wia} (mm)	y_{wia} (mm)	z_{wia} (mm)	Errors(x, y, z) mm
1	74	996	−632.9	1 594.2	0.0	−578.2	1 606.9	0.0	54.7, 12.8, 0.0
2	34	702	−632.9	1 594.2	356.5	−578.2	1 606.9	350.0	54.7, 12.8, −6.5
3	335	796	−428.5	2 114.9	0.0	−364.2	2 111.5	0.0	64.4, −3.4, 0.0
4	302	264	−428.5	2 114.9	757.6	−364.2	2 111.5	750.0	64.4, −3.4, −7.6
5	402	587	−463.7	3 067.9	0.0	−413.5	3 067.0	0.0	50.1, −0.9, 0.0
6	388	209	−463.7	3067.9	745.0	−413.5	3 067.0	746.0	50.1, −0.9, 1.0
7	435	507	−468.8	3 666.7	0.0	−383.3	3 634.4	0.0	85.5, −32.3, 0.0
8	421	183	−468.8	3 666.7	749.9	−383.3	3 634.4	750.0	85.5, −32.3, 0.1
9	488	506	−345.0	3 675.7	0.0	−272.4	3 648.7	0.0	72.5, −27.0, 0.0
10	475	226	−345.0	3 675.7	653.3	−272.4	3648.7	650.0	72.5, −27.0, −3.3
11	782	503	347.9	3 702.5	0.0	417.8	3 628.9	0.0	70.0, −73.5, 0.0
12	789	179	347.9	3 702.5	755.1	417.8	3 628.9	750.0	70.0, −73.5, −5.1
13	814	577	363.0	3 133.7	0.0	424.6	3 062.0	0.0	61.6, −71.8, 0.0
14	823	201	363.0	3 133.7	753.3	424.6	3 062.0	745.0	61.6, −71.8, −8.3
15	903	783	389.0	2 160.4	0.0	435.9	2 110.0	0.0	46.9, −50.4, 0.0
16	924	259	389.0	2 160.4	757.9	435.9	2 110.0	750.0	46.9, −50.4, −7.9
17	1 030	1 019	432.7	1 554.8	0.0	467.6	1 547.4	0.0	34.9, −7.4, 0.0
18	1 037	333	432.7	1 554.8	758.2	467.6	1 547.4	745.0	34.9, −7.4, −13.2
19	1 094	957	537.7	1 682.1	0.0	579.3	1 660.7	0.0	41.6, −21.4, 0.0
20	1 094	424	537.7	1 682.1	639.8	579.3	1 660.7	640.0	41.6, −21.4, 0.2

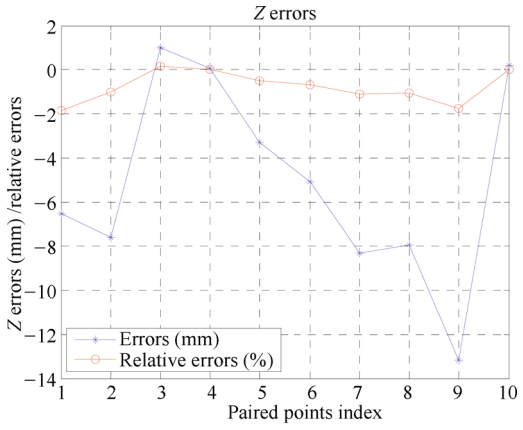
$$m^{\mathrm{T}} = \begin{bmatrix} m_1 & m_2 & m_3 & m_4 & m_5 & m_6 & m_7 & m_8 \end{bmatrix} = \begin{bmatrix} 6.406\,5 & 2.453\,6 & 561.732\,7 & 0.068\,9 & 0.280\,1 & 6\,588.936\,1 & -5.030\,2\times10^{-5} & 0.003\,8 \end{bmatrix} \quad (18)$$



(a) Positions



(b) Errors



(c) Z errors

Fig. 7 The measured positions and errors of the objects on the floor with the proposed method

In contrast to Figs. 4 and 5, the results in Fig. 8 measured with the compared method are with good accuracy. The parameter $\boldsymbol{m}$ in the compared method includes the intrinsic and extrinsic parameters, which ensure the tiny deviation of errors in re-calculation. Nevertheless, the compared method can only measure the 2D positions of the points on the floor. It cannot measure the heights of objects over the floor. Meanwhile, the proposed method in this paper has the ability to calculate the height of objects. The measured distance is up to 5 m. The errors in the proposed method are in the same level as ones in the compared method even

if the proposed method neglects the roll angle of the camera relative to the ground while the compared method includes all the intrinsic parameters and the extrinsic parameters of 6 degree-of-freedoms.

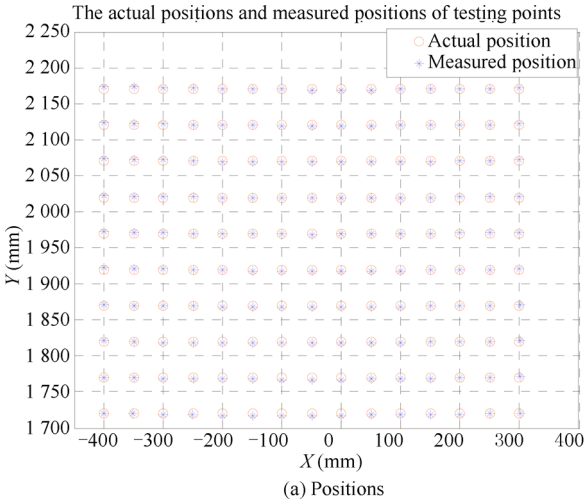


Fig. 8 The positions and errors after counterbalance with the compared method

Generally, there is a device to adjust the tilt of camera on robots. The relative tilt angle can be easily measured. Once the initial value of θ is estimated in calibration, the current value of θ can be obtained with the initial value and the adjusted relative tilt angle. The height of the camera is still the same. Then, the proposed method can be directly used to measure the objects′ positions on the floor and their heights without recalibration.

The proposed method can be used to measure 3D positions of objects. The paired points used in the proposed method can be found from edge-detection with the help of operators. In other words, the proposed method is useful to build the map for a mobile robot to realize localization and navigation. It is also helpful for a mobile manipulator to operate the object.

6 Conclusions

In this paper, a new monocular visual measurement method is presented for mobile robots. The method consists of height and pitch angle calibration and 3D position calculation. A single image with chessboard captured by monocular camera is used to estimate the height and pitch angle. Then for the normal indoor scenery, the position of object on the floor can be computed. Furthermore, the height of object can be calculated with the paired-points in the vertical line sharing the same position on the floor.

The error analysis provides the evidence that pitch angle is the critical part for accuracy. A minor change in pitch angle results in a larger error. Hence, there is no doubt that the positioning precision can be improved by increasing the accuracy of pitch angle estimation. The experiment results verify the effectiveness of the proposed method.

In future, we will focus on the automated measurement of the 3D positions of objects on floor combining the features extraction algorithm.

References

[1] G. N. Desouza, A. C. Kak. Vision for mobile robot navigation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 237–267, 2002.

[2] Q. Zhan, S. R. Huang, J. Wu. Automatic navigation for a mobile robot with monocular vision. In *Proceedings of IEEE Conference on Robotics, Automation and Mechatronics*, IEEE, Chengdu, China, pp. 1005–1010, 2008.

[3] S. Se, D. G. Lowe, J. J. Little. Vision-based global localization and mapping for mobile robots. *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 364–375, 2005.

[4] D. Murray, J. J. Little. Using real-time stereo vision for mobile robot navigation. *Autonomous Robots*, vol. 8, no. 2, pp. 161–171, 2000.

[5] S. Nissimov, J. Goldberger, V. Alchanatis. Obstacle detection in a greenhouse environment using the Kinect sensor. *Computers and Electronics in Agriculture*, vol. 113, pp. 104–115, 2015.

[6] T. Goedemé, M. Nuttin, T. Tuytelaars, L. Van Gool. Omnidirectional vision based topological navigation. *International Journal of Computer Vision*, vol. 74, no. 3, pp. 219–236, 2007.

[7] H. N. Do, M. Jadaliha, J. Choi, C. Y. Lim. Feature selection for position estimation using an omnidirectional camera. *Image and Vision Computing*, vol. 39, pp. 1–9, 2015.

[8] Y. J. Yin, D. Xu, Z. T. Zhang, X. G. Wang, W. T. Qu. Plane measurement based on monocular vision. *Journal of Electronic Measurement and Instrument*, vol. 27, no. 4, pp. 347–352, 2013. (in Chinese)

[9] P. Zhao, Z. Q. Cao, N. Gu, C. Zhou, D. Xu, M. Tan. A coordinated docking approach based on embedded vision. *International Journal of Robotics & Automation*, vol. 31, no. 1, pp. 52–62, 2016.

[10] G. Bresson, T. Féraud, R. Aufrère, P. Checchin, R. Chapuis. Real-time monocular SLAM with low memory requirements. *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1827–1839, 2015.

[11] E. Eade, T. Drummond. Scalable monocular SLAM. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, New York, USA, pp. 469–476, 2006.

[12] A. J. Davison, I. D. Reid, N. D. Molton, O. Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.

[13] E. Royer, M. Lhuillier, M. Dhome, J. M. Lavest. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, vol. 74, no. 3, pp. 237–260, 2007.

[14] H. Lim, S. N. Sinha, M. F. Cohen, M. Uyttendaele, H. J. Kim. Real-time monocular image-based 6-DoF localization. *The International Journal of Robotics Research*, vol. 34, no. 4–5, pp. 476–492, 2015.

[15] Y. Lu, D. Z. Song. Visual navigation using heterogeneous landmarks and unsupervised geometric constraints. *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 736–749, 2015.

[16] A. Milella, G. Reina, J. Underwood. A self-learning framework for statistical ground classification using radar and monocular vision. *Journal of Field Robotics*, vol. 32, no. 1, pp. 20–41, 2015.

[17] Y. Cao, L. C. Ma, W. G. Ma. Mobile target tracking based on hybrid open-loop monocular vision motion control strategy. *Discrete Dynamics in Nature and Society*, vol. 2015, Article number 690576, 2015.

[18] P. De Cristóforis, M. Nitsche, T. Krajnik, T. Pire, M. Mejail. Hybrid vision-based navigation for mobile robots in mixed in-door/outdoor environments. *Pattern Recognition Letters*, vol. 53, pp. 118–128, 2015.

[19] X. H. Chen, Y. M. Jia. Adaptive leader-follower formation control of non-holonomic mobile robots using active vision. *IET Control Theory & Applications*, vol. 9, no. 8, pp. 1302–1311, 2015.

[20] J. Campbell, R. Sukthankar, I. Nourbakhsh, A. Pahwa. A robust visual odometry and precipice detection system using consumer-grade monocular vision. In *Proceedings of the IEEE International Conference on Robotics and Automation*, IEEE, Barcelona, Spain, pp. 3421–3427, 2005.

[21] J. Zhou, B. X. Li. Robust ground plane detection with normalized homography in monocular sequences from a robot platform. In *Proceedings of the IEEE International Conference on Image Processing*, IEEE, Atlanta, USA, pp. 3017–3020, 2006.

[22] X. Y. Huang, F. Gao, G. Y. Xu, N. G. Ding, L. L. Xing. Depth information extraction of on-board monocular vision based on a single vertical target image. *Journal of Beijing University of Aeronautics and Astronautics*, vol. 41, no. 4, pp. 649–655, 2015. (in Chinese)

[23] S. S. Qu, X. Chen, X. H. Wu, Q. Yang. A method for measuring the height and area based on distance estimation of monocular vision. *Science Technology and Engineering*, vol. 16, no. 2, pp. 224–228, 2016. (in Chinese)

Ling-Yi Xu received the B. Sc. degree in control theory and control engineering from the University of Science and Technology Beijing, China in 2010. Currently, she is a master student in control theory and control engineering at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, China.

Her research interest is visual measurement for robots.

E-mail: xulingyi2014@ia.ac.cn

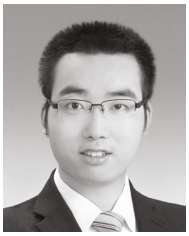
ORCID iD: 0000-0003-2984-7849

Zhi-Qiang Cao received the B. Sc.M. S. degrees from Shandong University of Technology, China in 1996 and 1999, respectively. In 2002, he received the Ph. D. degree in control theory and control engineering from Institute of Automation, Chinese Academy of Sciences, China. He is currently a professor in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, China.

His research interests include environmental cognition, robot control and multi-robot coordination.

E-mail: zhiqiang.cao@ia.ac.cn (Corresponding author)

ORCID iD: 0000-0003-1801-3363



Peng Zhao received the B. Sc. degree in mechanical design and automation science from Beijing Information Science and Technology University, China in 2010. He received the Ph. D. degree in control theory and control engineering at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, China in 2015.

His research interests include multi-robot system and visual servoing.

E-mail: peng.zhao@ia.ac.cn

Chao Zhou received the B. Sc. degree (Hons.) in automation from southeast University, China in 2003, and the Ph. D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, China in 2008. He is currently an associate professor in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, China.

His research interests include motion control of robots and bio-inspired robotic fish.

E-mail: chao.zhou@ia.ac.cn

ORCID iD: 0000-0003-4461-8075