Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 23 (2005) 831–840
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Single view metrology from scene constraints

Guanghui Wang[a,c,*], Zhanyi Hu[b], Fuchao Wu[b], Hung-Tat Tsui[c]

[a]Department of Control Engineering, Aviation University of Airforce, Changchun, 130022, People's Republic of China
[b]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, People's Republic of China
[c]Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, N.T. Hong Kong, China

Received 2 June 2003; received in revised form 20 April 2005; accepted 20 April 2005

Abstract

The problem of how to retrieve Euclidean entities of a 3D scene from a single uncalibrated image is studied in this paper. We first present two methods to compute the camera projection matrix through the homography of a reference space plane and its vertical vanishing point. Then, we show how to use the projection matrix and some available scene constraints to retrieve geometrical entities of the scene, such as height of an object on the reference plane, measurements on a vertical or arbitrary plane with respect to the reference plane, distance from a point to a line, etc. In particular, the method is further employed to compute the volume and surface area of some regular and symmetric objects from a single image, the undertaking seems no similar report in the literature to our knowledge. In addition, all the algorithms are formulated in an explicit and linear geometric framework, and the involved computation is linear. Finally, extensive experiments on simulated data and real images as well as a comparative test with a closely related method in the literature validate our proposed methods.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Computer vision; Single view metrology; Projective matrix; Metric measurement; Geometrical constraints

1. Introduction

One of the main aims of Computer Vision is to take measurements of the environment and reconstruct its 3D model. The problem of using vision technique to measure geometrical entities in the world has attracted a lot of attention and received wide applications in recent years [1,3,5], such as architectural and indoor measurement, reconstruction from paintings, forensic measurement and traffic accident investigation. Traditional approach to measurement is to take all the distances manually by using metric tapes or rulers or by some special devices such as ultrasonic devices, laser range finder, etc. These approaches are time consuming, prone to errors and invasive. With computer vision based methods, what one needs to do is only to take several pictures, then all measurements can be done offline with more accuracy, flexibility and efficiency. There are several potential advantages for this kind of approaches [1]. First, it is user friendly. Once the images are taken, users can take measurements desktoply and store them in a database. Second, the data acquisition process is rapid, simple and minimally invasive, since it only involves a camera to take pictures of the environment to be measured. Third, the acquired data are stored digitally in a disk ready for reuse at any time without going back to the original scene when new measurement are needed.

Generally speaking, the methods of computer vision based measurements in the literature may be broadly divided into two categories. The classical method is to reconstruct the metric structure of the scene from two or more images by stereovision techniques [6–9]. If we can obtain the Euclidean reconstruction of a scene, then any geometrical information about the scene can be retrieved accordingly. However, the Euclidean reconstruction is a hard task due to the problem of seeking correspondences from different views. In addition, the errors introduced by matching and camera calibration may propagate along the computational chain and cause a loss of accuracy to the final results.

The other one is to directly use a single uncalibrated image [3,4,10]. It is well known that only one image cannot provide enough information for a complete 3D

* Corresponding author. Tel.: +852 26098457; fax: +852 26035558.
E-mail address: ghwang@ee.cuhk.edu.hk (G. Wang).

0262-8856/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2005.04.002

reconstruction. However, some metrical quantities can be inferred directly from one image under the knowledge of some geometrical scene constraints such as planarity of points and parallelism of lines and planes. In [2], a homography based approach to distance measurement on a world plane was proposed, where the homography was computed from four specified control points on the plane. The same problem was also discussed in [5], where several other approaches to recovering the homography and taking measurement on a space plane were proposed. In [3], the authors described another approach to compute 3D affine measurement from a single perspective image. It is assumed that the vanishing line of a reference plane in the scene as well as a vanishing point in a reference direction can be determined from the image, then three canonical types of measurements (i.e. distances between any plane which are parallel to the reference plane, area and length ratio on these planes and the camera's position) can be computed. In [11–14], some methods were also investigated for object reconstruction from measurements in a single view both in computer vision community and photogrammetric community. These methods were based on the constraints of the object to be reconstructed, such as edges, coplanarity, parallelism, perpendicularity, etc.

In this paper, we mainly investigate the problem on single view metrology and propose some novel approaches to the recovery of geometrical information in a 3D scene. We first propose two new methods to retrieve the projection matrix from certain available scene constraints, while the geometrical information is computed directly from the projection matrix and a prior knowledge of the scene. We show how to (i) compute height of an object on the reference plane; (ii) take measurements on a vertical and arbitrary plane with respect to the reference plane; (iii) compute other geometrical entities, such as distance from a point to a plane, angle formed by two lines or two planes, distance from a point to a line, etc. In particular, we show how to apply the method to compute volume and surface area of some regular and symmetric objects from a single image. To our knowledge, there is no similar study reported in the literature.

Compared with other methods, we try to make fully use of the available scene constraints and recover more metric measurements, rather than affine entities as shown in [1,3]. All the algorithms involved in the paper are organized in an explicit and linear geometric framework, which is easy to implement. We also carry out some comparative tests with Criminisi's method [3].

The remaining parts of this paper are organized as follows: In Section 2, some preliminaries on projection matrix and homography are briefed. Then the methods for projection matrix computation are presented in Section 3. In Section 4, the novel approaches to the recovery of geometrical entities in the scene are elaborated. The test results with simulated data and real images are presented in Sections 5 and 6, respectively. Some conclusions are drawn at the end of this paper.

2. Some preliminaries

In order to facilitate our discussions in the subsequent sections, some preliminaries on camera projection matrix and homography are presented here. Please refer to Hartley [8] for more details. In this paper, the following notations are used: a point in space or image is denoted by a bold lower case letter, e.g. $\boldsymbol{x}$, while its corresponding homogeneous vector is denoted by $\tilde{\boldsymbol{x}} = [\boldsymbol{x}^{\mathrm{T}}, w]^{\mathrm{T}}$, and $(\boldsymbol{x})_i$ stands for the ith element of the vector $\boldsymbol{x}$; A matrix is denoted by a bold upper case letter, e.g. $\boldsymbol{P}$, and $\boldsymbol{P_i}$ stands for the ith column vector of the matrix $\boldsymbol{P}$, while $P_{i,j}$ for an element in the ith row and jth column of $\boldsymbol{P}$; '$\approx$' means equality up to a nonzero scale.

2.1. Camera projection matrix

Under the pinhole camera model, a 3D point $\boldsymbol{x}$ in space is projected to an image point $\boldsymbol{m}$ via a 3×4 projection matrix $\boldsymbol{P}$ as

$$s\tilde{\boldsymbol{m}} = \boldsymbol{P}\tilde{\boldsymbol{x}} = \boldsymbol{K}[\boldsymbol{R}, \boldsymbol{t}]\tilde{\boldsymbol{x}} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4]\tilde{\boldsymbol{x}} \tag{1}$$

where, s is an unknown nonzero scalar; $\boldsymbol{K}$ is the camera's intrinsic matrix; $\boldsymbol{R}$ and $\boldsymbol{t}$ are the rotation matrix and translation vector of the camera coordinate system to the world system.

Lemma 1. *The first three columns of projection matrix $\boldsymbol{P}$ are images of the vanishing points corresponding to the world coordinate axes X, Y and Z, respectively, and the last column $\boldsymbol{P}_4$ is the image of the origin of the world system.*

Proof. In the world coordinate system, the direction of X-, Y- and Z-axes can be expressed as $\tilde{\boldsymbol{x}}_w = [1,0,0,0]^{\mathrm{T}}$, $\tilde{\boldsymbol{y}}_w = [0,1,0,0]^{\mathrm{T}}$, $\tilde{\boldsymbol{z}}_w = [0,0,1,0]^{\mathrm{T}}$, respectively, the homogeneous form of the world origin is $\tilde{\boldsymbol{o}}_w = [0,0,0,1]^{\mathrm{T}}$. Thus, it is easy to verify that:

$$\tilde{\boldsymbol{v}}_x \approx \boldsymbol{P}\tilde{\boldsymbol{x}}_w = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4][1,0,0,0]^{\mathrm{T}} = \boldsymbol{P}_1$$

$$\tilde{\boldsymbol{v}}_y \approx \boldsymbol{P}\tilde{\boldsymbol{y}}_w = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4][0,1,0,0]^{\mathrm{T}} = \boldsymbol{P}_2$$

$$\tilde{\boldsymbol{v}}_z \approx \boldsymbol{P}\tilde{\boldsymbol{z}}_w = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4][0,0,1,0]^{\mathrm{T}} = \boldsymbol{P}_3$$

$$\tilde{\boldsymbol{o}}_x \approx \boldsymbol{P}\tilde{\boldsymbol{o}}_w = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4][0,0,1,0]^{\mathrm{T}} = \boldsymbol{P}_4$$

□

Lemma 2. *Given a point $\boldsymbol{m}$ in an image, its back-projection is a ray in space passing through the camera center and the point $\boldsymbol{m}$, i.e. $\boldsymbol{L}_{\mathrm{b}} = \{\boldsymbol{x} | s\tilde{\boldsymbol{m}} = \boldsymbol{P}\tilde{\boldsymbol{x}}, \boldsymbol{x} \in \Re^3\}$.*

Lemma 3. *Given a line $\boldsymbol{l}$ in an image, its back-projection spans a plane in space passing through the camera center and the line $\boldsymbol{l}$, which is defined as $\Pi = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{l}$.*

One of the major advantages of using the full 3×4 projection matrix is that it is not necessary to think of

the image as occupying a particular position in space with respect to the object. The image can be a thousand miles away from the object, while the projection matrix can still be used.

2.2. Plane to plane homography

Suppose there is a plane in the scene, without loss of generality, we set X- and Y-axes of the world coordinate system on the plane, then for a point on the plane, Eq. (1) becomes:

$$s\begin{bmatrix} u \\ v \\ t \end{bmatrix} = \boldsymbol{P}\begin{bmatrix} x \\ y \\ 0 \\ w \end{bmatrix} = \underbrace{[\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_4]}_{\boldsymbol{H}}\begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}\begin{bmatrix} x \\ y \\ w \end{bmatrix} \tag{2}$$

Hence, the projection from a point on the plane to its image is simplified as

$$s\tilde{\boldsymbol{m}} = \boldsymbol{H}\tilde{\boldsymbol{x}} \tag{3}$$

where, $\boldsymbol{H} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_4]$ is called plane to plane homography. Usually, $\boldsymbol{H}$ is a non-singular 3×3 homogeneous matrix (degeneracy occurs if and only if camera center is on the space plane) with 8 degrees of freedom because it can only be defined meaningfully up to a scaling factor. According to Eq. (2), each image-to-world point correspondence can give rise to two independent linear constraints on the nine elements of $\boldsymbol{H}$. Thus, the homography can be uniquely determined form four general coplanar space points (no three points are collinear) and their corresponding image points. If more than four pairs of such correspondences are given, $\boldsymbol{H}$ may be estimated by a suitable minimization scheme [1,8].

Lemma 4. *The homography can also be defined by line correspondences, i.e.* $s\boldsymbol{L} = \boldsymbol{H}^{\mathrm{T}}\boldsymbol{l}$, *with* $\boldsymbol{L}$ *a line in space and* $\boldsymbol{l}$ *the corresponding image. Hence, given four coplanar line correspondences in general position (i.e. no three lines are concurrent), the homography can be determined uniquely.*

Once the homography matrix between the world and image plane is determined, an image point can be back-projected to a point on the world plane via $\boldsymbol{H}^{-1}$. Hence the real distance between two image points on the plane can then be simply computed from the Euclidean distance between their back-projected 3D points. This is the basic principle of plane metrology [2].

3. Projection matrix recovery from scene constraints

In the following discussion, we assume that homography $\boldsymbol{H}$ of a reference plane, together with the vanishing point $\tilde{\boldsymbol{v}}_z$ of the direction perpendicular to the reference plane (called vertical vanishing point later), may be inferred directly from the image of the scene to be measured. This assumption is feasible in practice as illustrated in Section 6 and our early study in [5]. Since $\boldsymbol{H} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_4]$, $\boldsymbol{P}_3 = \lambda\tilde{\boldsymbol{v}}_z$, the projection matrix $\boldsymbol{P} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4]$ can be recovered easily once the nonzero scalar λ is determined.

3.1. Case for a simplified camera model

Generally speaking, there are five intrinsic parameters for a pinhole camera model, and the camera matrix is in the form of

$$\boldsymbol{K} = \begin{bmatrix} f_u & s & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix},$$

where, $f_{\mathrm{u}}, f_{\mathrm{v}}$ represent the camera's focal length corresponding to the u- and v-axes of camera coordinates, $[u_0, v_0]^{\mathrm{T}}$ is the coordinates of camera's principal point, $s = f_{\mathrm{u}}ctg\theta$ refers to the skew factor with θ the included angle of the two axes of the image system. For some high quality CCD camera, the two axes are perpendicular to each other, i.e. $\theta \to 90°$, thus the skew may be set to zero and the camera is simplified to have only four intrinsic parameters.

Proposition 1. *The camera projection matrix* $\boldsymbol{P}$ *can be uniquely determined from the homography* $\boldsymbol{H}$ *and the vertical vanishing point* $\tilde{\boldsymbol{v}}_z$ *in case of a simplified camera model with zero skew.*

Proof. Decompose the projection matrix into a 3×3 submatrix $\boldsymbol{M}$ and a 3-vector $\boldsymbol{P}_4$, i.e. $\boldsymbol{P} = [\boldsymbol{M}, \boldsymbol{P}_4]$ with $\boldsymbol{M} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3] = [\boldsymbol{P}_1, \boldsymbol{P}_2, \lambda\tilde{\boldsymbol{v}}_z]$, then we have

$$\boldsymbol{C} = \boldsymbol{M}\boldsymbol{M}^{\mathrm{T}} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \lambda\tilde{\boldsymbol{v}}_z]\begin{bmatrix} \boldsymbol{P}_1^{\mathrm{T}} \\ \boldsymbol{P}_2^{T} \\ \lambda\boldsymbol{v}_z^{T} \end{bmatrix}$$

$$= \boldsymbol{P}_1\boldsymbol{P}_1^{\mathrm{T}} + \boldsymbol{P}_2\boldsymbol{P}_2^{\mathrm{T}} + \lambda^2\tilde{\boldsymbol{v}}_z\tilde{\boldsymbol{v}}_z^{\mathrm{T}} = [C_{i,j}]_{3\times3} \tag{4}$$

where, $C_{i,j} = P_{i,1}P_{j,1} + P_{i,2}P_{j,2} + \lambda^2 v_i v_j$, v_i is the ith element of $\tilde{\boldsymbol{v}}_z$. On the other hand, $\boldsymbol{M} = \alpha\boldsymbol{K}\boldsymbol{R}$, with α a nonzero scalar and $\boldsymbol{R}$ the orthonormal rotation matrix, i.e. $\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}} = \mathrm{diag}(1,1,1) = \boldsymbol{I}$. Hence, Eq. (4) is equivalent to

the following form:

$$\boldsymbol{C} = \boldsymbol{M}\boldsymbol{M}^{\mathrm{T}} = \alpha^2 \boldsymbol{K}\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}}\boldsymbol{K}^{\mathrm{T}} = \alpha^2 \boldsymbol{K}\boldsymbol{K}^{\mathrm{T}}$$

$$= \alpha^2 \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_u & 0 & 0 \\ 0 & f_v & 0 \\ u_0 & v_0 & 1 \end{bmatrix}$$

$$= \alpha^2 \begin{bmatrix} f_u^2 + u_0^2 & u_0 v_0 & u_0 \\ u_0 v_0 & f_v^2 + v_0^2 & v_0 \\ u_0 & v_0 & 1 \end{bmatrix} \tag{5}$$

Equation $C_{1,2}C_{3,3} = C_{1,3}C_{2,3}$ holds by comparing the above two equations, from which the scalar can be solved as $\lambda = \pm\sqrt{b/a}$, where:

$$\begin{cases} a = P_{1,1}P_{2,1}v_3^2 + P_{1,2}P_{2,2}v_3^2 + v_1v_2P_{3,1}^2 + v_1v_2P_{3,2}^2 - P_{1,1}P_{3,1}v_2v_3 - P_{1,2}P_{3,2}v_2v_3 - P_{2,1}P_{3,1}v_1v_3 - P_{2,2}P_{3,2}v_1v_3 \\ b = P_{1,1}P_{2,2}P_{3,1}P_{3,2} + P_{1,2}P_{2,1}P_{3,1}P_{3,2} - P_{1,1}P_{2,1}P_{3,2}^2 - P_{1,2}P_{2,1}P_{3,1}^2 \end{cases}.$$

Note that only one of the two solutions of the scalar is reasonable. Since, $\det(\boldsymbol{M}) > 0$ as the objects must lie in the front of camera, the true solution can be determined from this constraint and the projection matrix is finally recovered.

3.2. Case for a general camera model

Numerical test in Section 5 shows that the influence of the skew factor to the distance measurement is significant when s is not near to zero. In this case, a general camera model with five intrinsic parameters should be adopted.

Proposition 2. *The camera projection matrix $\boldsymbol{P}$ can be uniquely determined from the homography $\boldsymbol{H}$, the vertical vanishing point $\tilde{\boldsymbol{v}}_z$ and a vertical reference length z_0 in case of a general camera model.*

Proof. Without loss of generality, select the world coordinate system as shown in Fig. 1, with the *O–XY* plane lying on the reference plane. Suppose the two end points of the reference length are $\boldsymbol{x}$ and $\boldsymbol{x}'$ with $\boldsymbol{x}$ on the reference plane, the corresponding points in the image are $\tilde{\boldsymbol{m}}_x = [u_x, v_x, 1]^{\mathrm{T}}$, $\tilde{\boldsymbol{m}}'_x = [u'_x, v'_x, 1]^{\mathrm{T}}$, respectively. Then from $\tilde{\boldsymbol{x}} = s\boldsymbol{H}^{-1}\tilde{\boldsymbol{m}}_x$, it is easy to retrieve the coordinates of point $\boldsymbol{x} = [x_0, y_0, 0]^{\mathrm{T}}$, so the coordinates of point $\boldsymbol{x}'$ must be $\boldsymbol{x}' = [x_0, y_0, z_0]^{\mathrm{T}}$. From

$$s\tilde{\boldsymbol{m}}'_x = \boldsymbol{P}\tilde{\boldsymbol{x}}' = [\boldsymbol{P}_1, \boldsymbol{P}_2, \lambda\tilde{\boldsymbol{v}}_z, \boldsymbol{P}_4]\tilde{\boldsymbol{x}}' \tag{6}$$

we can obtain the following equation system by eliminating the nonzero scalar s

$$\boldsymbol{A}\lambda = \boldsymbol{b} \tag{7}$$

where

$$\boldsymbol{A} = \begin{bmatrix} v_1 z_0 - v_3 u'_x z_0 \\ v_2 z_0 - v_3 v'_x z_0 \end{bmatrix},$$

$$\boldsymbol{b} = \begin{bmatrix} P_{3,1}u'_x x_0 + P_{3,2}u'_x y_0 + P_{3,4}u'_x - P_{1,1}x_0 - P_{1,2}y_0 - P_{1,4} \\ P_{3,1}v'_x x_0 + P_{3,2}v'_x y_0 + P_{3,4}v'_x - P_{2,1}x_0 - P_{2,2}y_0 - P_{2,4} \end{bmatrix}.$$

So λ may be solved by means of least-square technique as:

$$\lambda = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{b} \tag{8}$$

□

The projection matrix obtained here represents a general projective camera, so it is straightforward to retrieve the camera parameters by decomposition of $\boldsymbol{P}$. For example, let $\boldsymbol{P} = [\boldsymbol{M}, \boldsymbol{P}_4]$, then the camera matrix $\boldsymbol{K}$ and the rotation matrix $\boldsymbol{R}$ can be recovered by decomposing $\boldsymbol{M}$ into $\boldsymbol{M} = \boldsymbol{K}\boldsymbol{R}$ using RQ-decomposition; the translation vector $\boldsymbol{t} \approx \boldsymbol{K}^{-1}\boldsymbol{P}_4$; and the camera center is at $\boldsymbol{C} = \boldsymbol{M}^{-1}\boldsymbol{P}_4$.

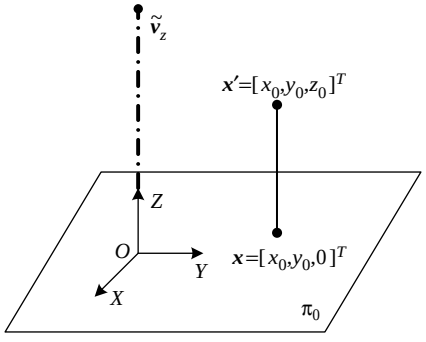

Fig. 1. Reference plane π_0, vertical vanishing point v_z and a vertical reference segment on the plane.

4. Methods for geometrical information recovery

In this section, we will present some novel methods for scene measurements directly from the recovered projection matrix and some scene constraints.

4.1. Height measurement

Suppose $\boldsymbol{x}$ and $\boldsymbol{x}'$ are the two end points of the object to be measured, with $\boldsymbol{x}$ on the reference plane, refer to Fig. 1. Let the image points of $\boldsymbol{x}$ and $\boldsymbol{x}'$ be $\tilde{\boldsymbol{m}}_x = [u_x, v_x, 1]^{\mathrm{T}}$ and $\tilde{\boldsymbol{m}}'_x = [u'_x, v'_x, 1]^{\mathrm{T}}$, respectively. Suppose the coordinates of point $\boldsymbol{x}$ in homogeneous form is $\tilde{\boldsymbol{x}} = [x_0, y_0, 0, 1]^{\mathrm{T}}$, the height of the object is z_0, then we have $\tilde{\boldsymbol{x}}' = [x_0, y_0, z_0, 1]^{\mathrm{T}}$, with x_0, y_0, z_0 three unknowns. Our goal is to compute the height z_0. Since

$$\begin{cases} s_1\tilde{\boldsymbol{m}}_x = \boldsymbol{P}\tilde{\boldsymbol{x}} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4]\tilde{\boldsymbol{x}} \\ s_2\tilde{\boldsymbol{m}}'_x = \boldsymbol{P}\tilde{\boldsymbol{x}}' = [\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4]\tilde{\boldsymbol{x}}' \end{cases} \tag{9}$$

expand the above equations and eliminate the two scalars s_1, s_2, we have

$$\boldsymbol{A}\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \boldsymbol{b} \tag{10}$$

where

$$\boldsymbol{A} = \begin{bmatrix} P_{1,1} - P_{3,1}u_x & P_{1,2} - P_{3,2}u_x & 0 \\ P_{2,1} - P_{3,1}v_x & P_{2,2} - P_{3,2}v_x & 0 \\ P_{1,1} - P_{3,1}u'_x & P_{1,2} - P_{3,2}u'_x & P_{1,3} - P_{3,3}u'_x \\ P_{2,1} - P_{3,1}v'_x & P_{2,2} - P_{3,2}v'_x & P_{2,3} - P_{3,3}v'_x \end{bmatrix},$$

$$\boldsymbol{b} = \begin{bmatrix} P_{3,4}u_x - P_{1,4} \\ P_{3,4}v_x - P_{2,4} \\ P_{3,4}u'_x - P_{1,4} \\ P_{3,4}v'_x - P_{2,4} \end{bmatrix}.$$

Thus, height z_0 can be obtained via least squares as:

$$z_0 = ((\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{b})_3 \tag{11}$$

4.2. Measurement on vertical plane

As shown in Fig. 2, π_0 is the reference plane, π_1 is the vertical plane perpendicular to π_0 and intersects π_0 at line $\boldsymbol{L}$. Denote the coordinates of π_1 as $\Pi_1 = [a_1, b_1, c_1, d_1]^{\mathrm{T}}$, we say a point $\tilde{\boldsymbol{v}} = [x, y, z, 1]^{\mathrm{T}}$ is in π_1 if and only if $\Pi_1^{\mathrm{T}}\tilde{v} = 0$.

Proposition 3. *For the configuration of Fig. 2, let* $\Pi_1 = [a_1, b_1, c_1, d_1]^{\mathrm{T}}$, *and suppose* $\boldsymbol{l}$ *is the corresponding image of* $\boldsymbol{L}$, *then* $a_1 = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_1$, $b_1 = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_2$, $c_1 = 0$, $d_1 = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_3$.

Proof. Since π_0 is coincident with the *O–XY*-plane, and the vertical plane π_1 is parallel to *OZ*-axis, it is clear that the coordinates of plane π_0 is $\Pi_0 = [0, 0, 1, 0]^{\mathrm{T}}$, and $c_1 = 0$. From Lemma 4, we have $\boldsymbol{l} = s\boldsymbol{H}^{-\mathrm{T}}\boldsymbol{L}$, thus the following equations hold true up to a common scalar factor.

$$a_1 = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_1, \qquad b_1 = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_2, \qquad d_1 = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_3 \tag{12}$$

i.e. the coordinates of plane π_1 is: $\Pi_1 = [(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_1, (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_2, 0, (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_3]^{\mathrm{T}}$. □

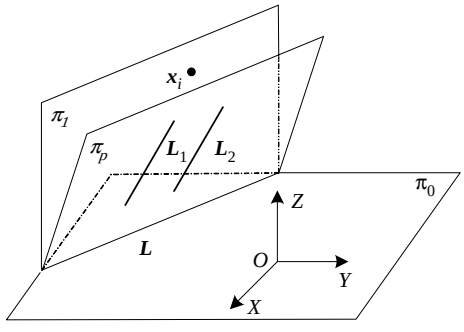

Fig. 2. Measurement on vertical plane π_1 and arbitrary plane π_p.

For a point $\boldsymbol{x_i}$ on the vertical plane π_1, suppose $\boldsymbol{m}_i = [u_i, v_i]^{\mathrm{T}}$ is its corresponding image, then the coordinates of $\boldsymbol{x}_i$ can be retrieved by the intersection of the back-projected ray of image point $\boldsymbol{m}_i$ and the plane $\boldsymbol{\pi}_1$, i.e.

$$\begin{cases} s_i\tilde{\boldsymbol{m}}_i = \boldsymbol{P}\tilde{\boldsymbol{x}}_i \\ \Pi_1^{\mathrm{T}}\tilde{\boldsymbol{x}}_i = 0 \end{cases} \tag{13}$$

eliminate scalar $\boldsymbol{s}_i$ and reorganize the above equation into a linear system in form of

$$\boldsymbol{A}\boldsymbol{x}_i = \boldsymbol{b} \tag{14}$$

where

$$\boldsymbol{A} = \begin{bmatrix} P_{1,1} - P_{3,1}u_i & P_{1,2} - P_{3,2}u_i & P_{1,3} - P_{3,3}u_i \\ P_{2,1} - P_{3,1}v_i & P_{2,2} - P_{3,2}v_i & P_{2,3} - P_{3,3}v_i \\ a_1 & b_1 & c_1 \end{bmatrix},$$

$$\boldsymbol{b} = \begin{bmatrix} P_{3,4}u_i - P_{1,4} \\ P_{3,4}v_i - P_{2,4} \\ -d_1 \end{bmatrix}.$$

Thus, the coordinates of the point $\boldsymbol{x}_i$ can be computed directly from the above equation.

4.3. Measurement on arbitrary plane

Suppose an arbitrary plane $\boldsymbol{\pi}_p$ intersect the reference plane $\boldsymbol{\pi}_0$ at line $\boldsymbol{L}$ (still refer to Fig. 2), then all the space planes passing through line $\boldsymbol{L}$ form a pencil, and the pencil of planes may be expressed as $\Pi_p = \Pi_1 + \lambda\Pi_0 = [(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_1, (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_2, \lambda, (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_3]^{\mathrm{T}}$, where $\Pi_0 = [0, 0, 1, 0]^{\mathrm{T}}$ is the coordinates of the reference plane, $\Pi_1 = [(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_1, (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_2, 0, (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_3]^{\boldsymbol{T}}$ is that of the vertical plane passing through $\boldsymbol{L}$. Thus, the plane π_{p} is defined up to only one unknown parameter λ.

Proposition 4. *The coordinates of plane* π_p *can be determined from the image of a pair of parallel lines in the plane*.

Proof. Suppose $\Pi_p = [a_p, b_p, c_p, d_p]^{\mathrm{T}}$, then $a_p = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_1$, $b_p = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_2$, $c_{\mathrm{p}} = \lambda$, $d_p = (\boldsymbol{H}^{\mathrm{T}}\boldsymbol{l})_3$. Denote the parallel lines in space and their corresponding images as $\boldsymbol{L}_1, \boldsymbol{L}_2$ and $\boldsymbol{l}_1, \boldsymbol{l}_2$, respectively, denote the intersection of $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$ as v_{p}, which is located at infinity, then its image (i.e. the intersection of $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$) can be computed as

$\tilde{\boldsymbol{v}}_i = \boldsymbol{l}_1 \times \boldsymbol{l}_2 = [a_i, b_i, c_i]^{\mathrm{T}}$, From equation (1), we have:

$$s\tilde{\boldsymbol{v}}_i = \boldsymbol{P}\tilde{\boldsymbol{v}}_p = [\boldsymbol{M}, \boldsymbol{P}_4]\begin{bmatrix} v_p \\ 0 \end{bmatrix} = \boldsymbol{M}\boldsymbol{v}_p \tag{15}$$

On the other hand, v_{p} lies on the plane π_{p}, i.e.

$$\Pi_p^{\mathrm{T}}\tilde{\boldsymbol{v}}_p \approx [a_p, b_p, \lambda]\boldsymbol{M}^{-1}[a_i, b_i, c_i]^{\mathrm{T}} = 0 \tag{16}$$

Thus, λ can be easily computed from the above equation. After retrieving the coordinates of the plane π_{p}, we can take measurements on the plane in a similar way as shown in Section 4.2. □

Remark 1. If some other prior information in the arbitrary plane, such as a reference distance, two orthogonal lines, etc. can be retrieved from the image, the scalar λ, as well as the coordinates of the plane, can also be computed in a similar way.

Remark 2. For a point in the reference plane, its coordinates can be retrieved directly from the homography, this allow us to compute the distance between any pair of two points, with each lying on one of the two planes.

4.4. Volume and surface area measurement

Based on the above discussion, we can also retrieve the volume and surface area of some regular and symmetric objects, such as a cylinder, circular cone, cube, pyramid, sphere, etc. Fig. 3 shows an image of a truncated circular cone in the reference plane, we will take it as an example to outline the algorithm.

(i) Fit the two image conics $\boldsymbol{C}_{\mathrm{u}}$ and $\boldsymbol{C}_{\mathrm{l}}$. The upper and lower surfaces of a truncated cone are imaged as conics, and the conics can be fitted by the visible contour of the cone by Hough transform and other robust algorithm.

(ii) Compute the vanishing line $\boldsymbol{l}_\infty$ of the reference plane: The vanishing line of the reference plane is: $\boldsymbol{l}_\infty = s\boldsymbol{H}^{-\mathrm{T}}\boldsymbol{L}_\infty$, with $\boldsymbol{L}_\infty = [0, 0, 1]^{\mathrm{T}}$.

(iii) Retrieve the two conic centers $\boldsymbol{o}_{\mathrm{u}}$ and $\boldsymbol{o}_{\mathrm{l}}$. Since the upper and lower surfaces of the truncated cone are parallel, they share the same vanishing line $\boldsymbol{l}_\infty$ in the image, while the conic center is the pole of the vanishing line with respect to the conic. Hence, we have $\tilde{\boldsymbol{o}}_u = \boldsymbol{C}_u^{-1}\boldsymbol{l}_\infty$, $\tilde{\boldsymbol{o}}_l = \boldsymbol{C}_l^{-1}\boldsymbol{l}_\infty$

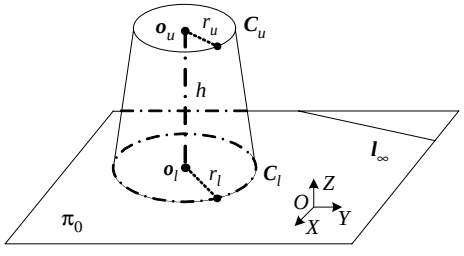

Fig. 3. A truncated cone on the reference plane in space.

(iv) Compute the height of the truncated cone h. The height of the truncated cone is the distance between the two centers, which can be computed by the method proposed in Section 4.1.

(v) Compute the upper and lower radius r_{u} and r_{l}. The radius refers to the corresponding Euclidean distance between the center and any point on the conic (which corresponds to a circle in space). A more faithful result can be obtained by using more points on the conic.

(vi) Compute the volume V, lateral area S_{l} and whole area S_{w}. After retrieve the height h, upper radius r_{u} and lower radius r_{l}, it is very easy to compute the volume and surface area of the truncated cone by the following formulas:

$$V = \frac{\pi}{3}h(r_{\mathrm{u}}^2 + r_{\mathrm{u}}r_{\mathrm{l}} + r_{\mathrm{l}}^2)$$

$$S_{\mathrm{l}} = \pi(r_{\mathrm{u}} + r_{\mathrm{l}})\sqrt{h^2 + (r_{\mathrm{l}} - r_{\mathrm{u}})^2}$$

$$S_{\mathrm{w}} = S_{\mathrm{s}} + \pi(r_{\mathrm{u}}^2 + r_{\mathrm{l}}^2).$$

4.5. Retrieval of other geometrical entities

In this section, we will present an approach to measure the distance from a point to a line in space. While other geometrical entities, such as distance between two lines or two parallel planes, distance from a point to a plane, angle formed by two lines or two planes and that formed by a line and a plane, may also be recovered in a similar way.

Suppose π_1 and π_2 are two planes in the scene, given a point $\boldsymbol{x} \in \pi_1$ and a line $\boldsymbol{L} \in \pi_2$, with $\boldsymbol{m}$ and $\boldsymbol{l}$ the corresponding images.

(i) Compute the coordinates of point $\boldsymbol{x} = [x, y, z]^{\mathrm{T}}$ from $\boldsymbol{m}$ by the method stated in Section 4.3.

(ii) Compute the direction vector of line $\boldsymbol{L}$. First, retrieve the coordinates of plane π_2 by Proposition 3, let $\Pi_2 = [\boldsymbol{n}_2^{\mathrm{T}}, d_2]^{\mathrm{T}}$ with $\boldsymbol{n}_2$ the normal vector of π_2; then retrieve the back-projected plane π_{b} of $\boldsymbol{l}$, let $\Pi_{\mathrm{b}} = [\boldsymbol{n}_{\mathrm{b}}^{\mathrm{T}}, d_{\mathrm{b}}]^{\mathrm{T}}$ with $\boldsymbol{n}_{\mathrm{b}}$ the normal vector of the back-projected plane; so the direction vector of line $\boldsymbol{L}$ is $\boldsymbol{n}_{\mathrm{L}} = \boldsymbol{n}_2 \times \boldsymbol{n}_{\mathrm{b}}$.

(iii) Compute the distance from $\boldsymbol{x}$ to $\boldsymbol{L}$. Randomly select a point $\boldsymbol{x}_0$ on $\boldsymbol{L}$, i.e.

$$\begin{bmatrix} n_2^{\mathrm{T}}, d_2 \\ n_{\mathrm{b}}^{\mathrm{T}}, d_{\mathrm{b}} \end{bmatrix}\begin{bmatrix} x_0 \\ 1 \end{bmatrix} = 0,$$

then the distance can be computed as

$$d(\boldsymbol{x}, \boldsymbol{L}) = \frac{||(\boldsymbol{x} - \boldsymbol{x}_0) \times \boldsymbol{n}_{\mathrm{L}}||}{||\boldsymbol{n}_{\mathrm{L}}||}.$$

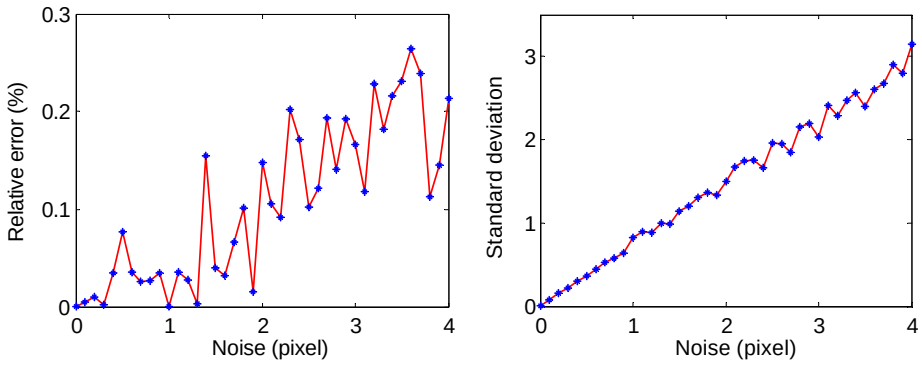

Fig. 4. Test result of height measurement under simplified camera model.

5. Experiments with simulated data

During the simulations, the camera's setup is: $f_u=1200$, $f_v=1000$, $s=0$, $u_0=512$, $v_0=384$. The image resolution is 1024×768 pixels. The camera extrinsic parameters are: rotation axis $r=[2,1,4]^T$, rotation angle $\alpha=\pi/7$ and translation $t=[10,5,250]^T$. We randomly select 10 points in the horizontal reference plane to compute homography and evenly select 100 points on a pair of vertical parallel lines $\boldsymbol{L}_1$, $\boldsymbol{L}_2$. The Gaussian image noise (with mean zero) is added on each image point, and the corresponding image lines $\boldsymbol{l}_1$, $\boldsymbol{l}_2$ are fitted via least-squares, then the vertical vanishing point can be computed as $\tilde{\boldsymbol{v}}_z=\boldsymbol{l}_1\times\boldsymbol{l}_2$. In order to provide more statistically meaningful results, we vary the noise level from 0 to 4 with a step of 0.1 during the test, and take 200 independent tests at each noise level.

5.1. Height measurement

We first randomly select a vertical reference length on the reference plane, retrieve the projection matrix according to the method for the simplified camera model, and then randomly select some objects on the reference plane and compute their height by the method stated in Section 4.1. The test results are shown in Fig. 4, where the left figure shows the mean of relative error of the estimated height at different noise level, while the right one is the corresponding standard deviations. From the results we can learn that the proposed approach is of high accuracy when the skew is zero.

5.2. The influence of skew factor

According to the camera setup in the test, $f_u=1200$, if the angle formed by the image axes changes from 89.5 to 90.5°, the skew factor s will change from -10.47 to 10.47 accordingly, so it is necessary to analyze its influence to the measurement. First, we vary the skew factor from -5 to 5 in a step of 0.2 without added noise in the image, then take the height measurement by the method for the simplified camera model, the results are shown in the left of Fig. 5, the value at each step is the mean of 200 independent tests. Next, we add 2 pixels noise in the image and do the same test again, the results are shown in the right of Fig. 5. We can conclude from these results that the variation of skew factor has a significant influence to the accuracy of measurement, and such an influence is far more pronounced than that of noise. Therefore, we should not use the simplified camera model to take measurement when the skew factor is not close to zero.

5.3. Measurement on vertical plane

Here, the skew factor is set to $s=10$, and a reference length in the vertical plane is given so as to retrieve

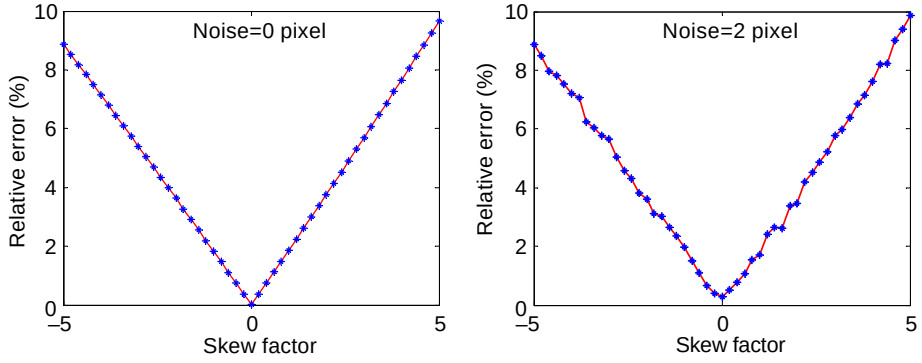

Fig. 5. The influence of skew factor to measurement.

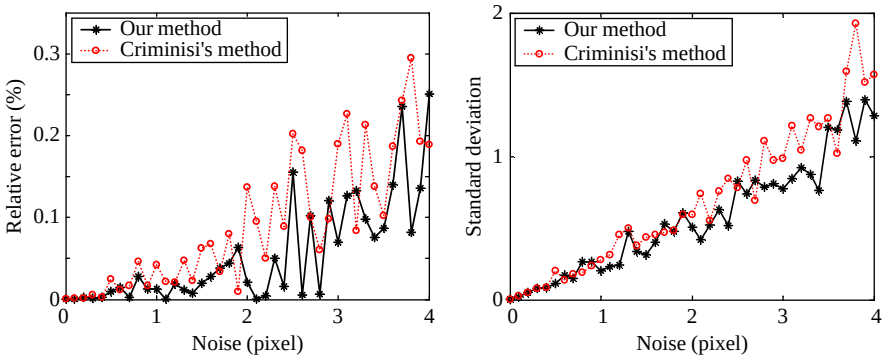

Fig. 6. Comparative test result of distance measurement on vertical plane under general camera model.

the projection matrix by the method in Section 3.2. Then randomly select some points on the vertical plane, compute their coordinates via the method in Section 4.2, and finally the distance between any two points. Fig. 6 shows the relative error and standard deviations of the estimated distances (in real lines with asterisk markers). We also carry out a comparative study with the method proposed by Criminisi [3]. It should be noted that the measurements in [3] are some affine entities, so less assumption is needed. However, the same assumption as proposed in this paper should be made if we want to rectify the affine measurements into the Euclidean ones. The test results are shown in Fig. 6 (in dashed lines with circular markers). We can see from the results that the accuracy of the two methods is comparative, while the proposed method performs a little bit better in the test.

6. Experiments with real images

In real image test, images are taken by a Nikon Coolpix 990 digital camera with resolution of 1024×768. Fig. 7 shows three images taken in our lab with the world coordinate system chosen on the ground. During the test, we take the ground floor as the reference plane and use the information of the floor tiles to compute homography with the ground truth that the side length of each tile is 61 mm (where, we select eight control points on the floor of Fig. 7(a) and 12 points on Fig. 7(b) and (c)). We use Canny edge detector to detect the vertical straight-line segments and use the detected lines to compute the vertical vanishing point via maximum likelihood estimator [8]. The vertical reference length is selected as $R_1 = 121.6$ cm, $R_2 = 234.2$ cm, $R_3 = 171.5$ cm, as shown in Fig. 7. The test results are

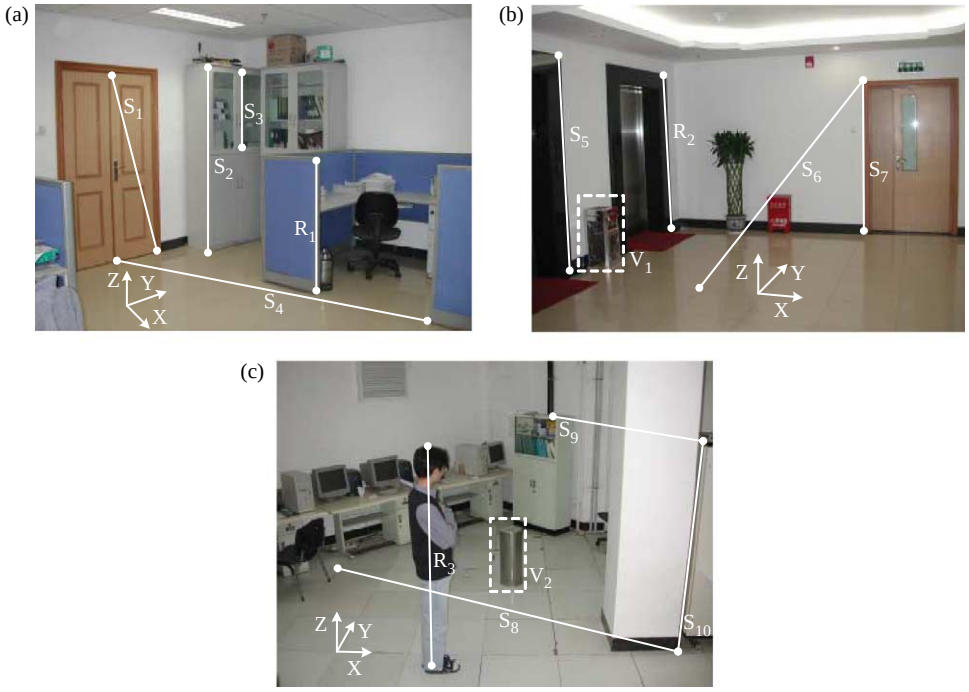

Fig. 7. Three images of indoor scene used in the tests.

Table 1
Comparative test results of real images

Line segments		S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}
True distance (cm)		212.7	219.8	89.8	334.2	234.2	380.0	212.8	389.2	302.7	170.4
Proposed method	Estimated value	213.20	219.83	89.07	333.96	235.41	381.78	214.14	388.46	301.25	169.33
	Relative error (%)	0.24	0.01	0.81	0.07	0.52	0.47	0.54	0.19	0.48	0.63
Criminisi's method	Estimated value	213.26	220.20	89.22	333.96	235.68	382.43	213.76	388.46	304.72	169.41
	Relative error (%)	0.26	0.18	0.65	0.07	0.63	0.64	0.45	0.19	0.67	0.58

given in Table 1, where, the true value is taken manually on the spot. Note that the distances of S_8 and S_9 are really difficult to take manually in practice. A comparative results with Criminisi's method [3] are also shown in Table 1. The two methods are of similar accuracy.

During the tests, a general camera model is employed to compute the projection matrix. After recovering the projection matrixes, the camera intrinsic and extrinsic parameters corresponding to each image are decomposed. Here, we just give the parameters of Fig. 7(a) and (b), from which we can see that the results are largely consistent with the real situation.

For Fig. 7(a):

$$\boldsymbol{K}_1 = \begin{bmatrix} 1209.0 & -9.3 & 503.2 \\ 0 & 1215.8 & 438.8 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{R}_1 = \begin{bmatrix} 0.7457 & 0.6662 & -0.0029 \\ 0.0875 & -0.1023 & -0.9909 \\ -0.6605 & 0.7387 & -0.1346 \end{bmatrix},$$

$$\boldsymbol{T}_1 = \begin{bmatrix} -96.01 \\ 94.25 \\ 434.77 \end{bmatrix}, \quad \boldsymbol{C}_1 = \begin{bmatrix} 350.50 \\ -247.55 \\ 151.63 \end{bmatrix}.$$

For Fig. 7(b):

$$\mathbf{K}_2 = \begin{bmatrix} 1210.8 & 0.2 & 541.3 \\ 0 & 1228.4 & 383.8 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{R}_2 = \begin{bmatrix} 0.9552 & 0.2952 & -0.0215 \\ 0.0036 & -0.0843 & -0.9964 \\ -0.2960 & 0.9517 & -0.0816 \end{bmatrix},$$

$$\mathbf{T}_2 = \begin{bmatrix} -16.69 \\ 119.79 \\ 416.67 \end{bmatrix}, \quad \mathbf{C}_2 = \begin{bmatrix} 138.84 \\ -381.51 \\ 153.01 \end{bmatrix}$$

We also compute the volume of the cuboid shape dustbin in Fig. 7(b) and the cylindrical one in Fig. 7(c), and the estimated value are $V_1 = 110{,}018.9\ \text{cm}^3$, $V_2 = 26{,}628.2\ \text{cm}^3$ with 0.69 and 0.74% relative error to the ground truth (taken manually on the spot) of 109,265.0 and 26,826.7 cm^3, respectively. The results are satisfactory.

7. Conclusions

In this paper, we mainly focus on geometrical information recovery from a single uncalibrated image and propose some novel approaches. Simulations and real image tests, as well as the comparison with the method in [3], validate the proposed methods and show that the proposed approaches are of high accuracy and strong robustness. In some structured environment, the geometrical constraints about the scene are not rare, such as the outline of a building, the frame of a window, a door, floor board, etc. the applicability of the proposed approaches seems not too limited.

It is clear that all the proposed approaches are based on some known specific geometrical information about the scene, and the precision of the measurement depends greatly on that of the image pretreatment, such as edge detection, line fitting and vanishing point determination. Hence, it is crucial to select a robust edge detection and line fitting technique so as to better the accuracy of measurement [15,16]. We should note that lens distortion has not been considered in our methods, since for the camera used in our experiments, the lens distortion is negligible. However, the image should be rectified firstly if the lens distortion does play a significant role to the accuracy of measurement.

Acknowledgements

The authors would like to thank the reviewers for their valuable comments and suggestions. The work is supported by the National Natural Science Foundation of China under grant No. 60175009 and 60121302, the Hong Kong RGC grant program under grant No. CUHK 4378/02E, the Scientific Development Project of Jilin Province under grant No. 20010320.

References

[1] A. Criminisi, Accurate Visual Metrology from Single and Multiple Uncalibrated Images-Distinguished Dissertations, Springer, Berlin, 2001.

[2] A. Criminisi, I. Reid, A. Zisserman, A plane measuring device, Image and Vision Computing 17 (8) (1999) 625–634.

[3] A. Criminisi, I. Reid, A. Zisserman, Single view metrology, International Conference on Computer Vision. Kekyrn, Greece, September, 1999 pp. 434–442.

[4] G.H. Wang, H.T. Tsui, Z.Y. Hu, F.C. Wu, Camera calibration and 3D reconstruction from a single view based on scene constraints, Image and Vision Computing 23 (3) (2005) 311–323.

[5] G.H. Wang, Z.Y. Hu, F.C. Wu, Single view based measurement on space planes, Journal of Computer Science and Technology. 19 (3) (2004) 374–382.

[6] G.H. Wang, H.T. Tsui, Z.Y. Hu, Reconstruction of structured scenes from two uncalibrated images, Pattern Recognition Letters 26 (2) (2005) 207–220.

[7] D. Liebowitz, A. Zisserman, A Metric rectification for perspective images of planes, Proceedings of IEEE International Conference on Computer Vision & Pattern Recognition, Santa Barbara, CA, June, 1998 pp. 482–488.

[8] R. Hartley, A. Zisserman, Multiple View Geometry in COMPUTER Vision, Cambridge University Press, Cambridge, MA, 2000.

[9] I. Reid, A. Zisserman, Goal-directed video metrology, Proceedings of European Conference of Computer Vision, Cambridge, UK, April, vol. II, 1996 pp. 647–658.

[10] F.A. van den Heuvel, 3D reconstruction from a single image using geometric constraints, ISPRS Journal of Photogrammetry & Remote Sensing 53 (6) (1998) 354–368.

[11] T. Kim, Y. Seo, K. Hong, Physics-based 3D position analysis of a soccer ball from monocular image sequences, Proceedings of International Conference on Computer Vision, Bombay, India, Jan., 1998 pp. 721–726.

[12] D. Liebowitz, A. Criminisi, A. Zisserman, Creating architectural models from images, Proceedings of Eurographics, Milan, Italy, 1999 pp. 39–50.

[13] M. Wilczkowiak, E. Boyer, P. Sturm, Camera calibration and 3D reconstruction from single images using parallelepipeds, Proceedings of International Conference on Computer Vision, Vancouver, Canada, July, vol. I, 2001 pp. 142–148.

[14] P. Gurdjos, R. Payrissat, About conditions for recovering the metric structure of perpendicular planes from the single ground plane to image homography, Proceedings of International Conference on Pattern Recognition, Barcelona, Spain, Sept., vol. I, 2000 pp. 358–361.

[15] G. McLean, D. Kotturi, Vanishing point detection by line clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (11) (1995) 1090–1095.

[16] A. Almansa, A. Desolneux, S. Vamech, Vanishing point detection without any a priori information, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (4) (2003) 502–507.