J Math Imaging Vis 24: 279–293, 2006

DOI: 10.1007/s10851-005-3628-9

# Camera Calibration and Direct Reconstruction from Plane with Brackets

YIHONG WU AND ZHANYI HU
*National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, P.O. Box 2728, Beijing 100080, P.R. China*
yhwu@nlpr.ia.ac.cn
huzy@nlpr.ia.ac.cn
http://www.nlpr.ia.ac.cn/english/rv

**Published online:** 9 February 2006

**Abstract.** Camera calibration and 3D reconstruction are important issues in computer vision. Two applications of bracket algebra in these two issues are presented in this work. Firstly, a camera calibration method is proposed, which is from only distance ratios of object points. Thanks to the effective computations of brackets, this method does not need to set up any world coordinate system and thus can use the geometric information of irregular objects conveniently. Secondly, we represent the reconstruction solution of plane structure directly from four known control points and give some new and useful error analysis results. The solution based on brackets is concise and short, and the error analysis results can act as a theoretical guidance in practice. Simulations and experiments on real images validate our proposed camera calibration method, direct reconstruction solution and error analysis results.

**Keywords:** bracket algebra, camera calibration, direct reconstruction of plane structure, error analysis

## 1. Introduction

Projective geometric invariant plays a very important role in computer vision. Due to its effectiveness in the computation of projective invariants, bracket algebra, which is the subalgebra of geometric algebra composed of determinants only, has had some applications in the field of computer vision [3–5, 10, 14, 16, 17, 21, 25]. Currently in computer vision, the great potential of bracket algebra is not yet fully developed. This paper is to present two new applications of bracket algebra in camera calibration and 3D reconstruction.

Usually, for camera calibration using the geometric information of spatial points, the methods firstly set up an object coordinate system, then solve the camera projective matrix or the homography between the scene and the image, and finally establish the constraints on the camera intrinsic parameters from the obtained camera projective matrix or homography [1, 26]. It is straightforward to set up an object coordinate system if the spatial points distribute regularly. In order to calibrate camera flexibly using the geometric information of irregular object, here we propose a new camera calibration method from only distance ratios of spatial points without involving any object coordinate system. Bracket algebra makes easy the representation and computation of the scene distance ratios. Simulated and real experiments validate this method, and show its flexibility.

A rectangle pattern or a four-coplanar-control-point pattern is popular in many vision tasks such as camera calibration, camera pose determination, and metrology [2, 11–13, 15, 18, 19, 22, 23]. From a single view of a plane with a rectangle or at least four control points, the traditional way of metrology or the reconstruction is to determine the homography between the space plane and the image plane at first, then to reconstruct plane structure via this homography [6–9]. Some error analysis for the metrology via homography has been given [7, 9]. Here, we represent directly the reconstruction solution of plane structure from four control points (not necessarily the vertexes of a rectangle). The reconstruction solution is concise and short. We use it for error analysis, and find out some new and interesting results from a geometric standpoint. The error analyses of [7, 9] are also suitable for our direct reconstruction solution. Simulations and experiments on real data validate the direct reconstruction solution and our new error analysis results.

The organization of this paper is as follows. Some preliminaries are listed in Section 2. Section 3 gives the camera calibration method from the scene distance ratios. The direct reconstruction solution and analytical error analyses are detailed in Section 4. Simulations and experiments on real data are reported in Section 5, and Section 6 is some concluding remarks.

## 2. Preliminaries

A bold number or bold small letter denotes a 3-vector or 2D homogeneous coordinates, the symbol "| |" denotes the absolute value, the bracket "[ ]" denotes the determinant of vectors in it, and "≈" denotes the equality up to a scalar.

Bracket algebra is in fact the algebra on determinants, which is the basic algebra tool for computing projective geometric invariants [20, 24]. Exchanging two vectors in a bracket will change the sign of the bracket, for example: $[\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3] = -[\mathbf{a}_2\mathbf{a}_1\mathbf{a}_3]$. $[\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3] = 0$ means that $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ are collinear.

Let $S_{\mathbf{a}123}$ be the area of the triangle with $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ as the vertexes, then, $[\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3] = 2S_{\mathbf{a}123}$ when the order of $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ is counterclockwise; otherwise, $[\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3] = -2S_{\mathbf{a}123}$. Thus, $\frac{[\mathbf{a}_2\mathbf{a}_3\mathbf{a}_4]}{[\mathbf{a}_1\mathbf{a}_3\mathbf{a}_4]} = \frac{S_{\mathbf{a}234}}{S_{\mathbf{a}134}}$ when $\mathbf{a}_1, \mathbf{a}_2$ are in the same side of the line through $\mathbf{a}_3, \mathbf{a}_4$; and $\frac{[\mathbf{a}_2\mathbf{a}_3\mathbf{a}_4]}{[\mathbf{a}_1\mathbf{a}_3\mathbf{a}_4]} = -\frac{S_{\mathbf{a}234}}{S_{\mathbf{a}134}}$ when $\mathbf{a}_1, \mathbf{a}_2$ are in the different side of the line through $\mathbf{a}_3, \mathbf{a}_4$.

The area $S_{\mathbf{a}123}$ of the triangle $\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3$ also can be represented as:

$$\frac{1}{4}\sqrt{(d_{a12}+d_{a13}+d_{a23})(-d_{a12}+d_{a13}+d_{a23})(d_{a12}-d_{a13}+d_{a23})(d_{a12}+d_{a13}-d_{a23})}$$

where $d_{aij}$ is the distance between $\mathbf{a}_i, \mathbf{a}_j$.

$e_1x + e_2y + e_3 = 0$ is a line with $(x, y, 1)$ as variable vector in 2D plane. Let $(x_0, y_0, 1)$ be the coordinates of a point in this plane, then the distance from this point to this line is:

$$\frac{|e_1x_0 + e_2y_0 + e_3|}{\sqrt{e_1^2 + e_2^2}} \tag{1}$$

For two fixed points $\mathbf{a}_1, \mathbf{a}_2$, the line-coordinates of the line through them is $\mathbf{a}_1 \times \mathbf{a}_2$. It follows that by (1), the distance of $\mathbf{a}_3$ to the line $\mathbf{a}_1\mathbf{a}_2$ is $\frac{|[\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3]|}{\sqrt{(\mathbf{a}_1\times\mathbf{a}_2)_x^2+(\mathbf{a}_1\times\mathbf{a}_2)_y^2}}$, where $(\mathbf{a}_1\times\mathbf{a}_2)_x$, $(\mathbf{a}_1\times\mathbf{a}_2)_y$ are the first and second coordinates of $\mathbf{a}_1\times\mathbf{a}_2$. Denote this distance as $d$, and its denominator as $s$. It can be seen that $s$ is not related with $\mathbf{a}_3$. There is $|[\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3]| = sd$. Thus, when $\mathbf{a}_1, \mathbf{a}_2$ are fixed and $\mathbf{a}_3$ is varying, $|[\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3]|$ can also represent the varying distance of $\mathbf{a}_3$ to the line $\mathbf{a}_1\mathbf{a}_2$ in some sense.

Under the pinhole camera model, a spatial point $\mathbf{M}_i$ is projected to a point $\mathbf{m}_i$ in the image plane by:

$$s_i\mathbf{m}_i = \mathbf{K}(\mathbf{R}, \mathbf{t})\mathbf{M}_i, \quad i = 1..6 \tag{2}$$

where $\mathbf{R}$, $\mathbf{t}$ are a 3×3 rotation matrix and a 3-D translation vector, $s_i$ is a nonzero scalar, and

$$\mathbf{K} = \begin{pmatrix} f & s & u_0 \\ 0 & \alpha f & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

is the 3×3 matrix of camera intrinsic parameters, $f$ is the focal length, $\alpha$ the aspect ratio, $s$ the skew parameter, $(u_0, v_0)$ the principal point. In our subsequent discussions, we always assume the camera model is of the pinhole one. In other words, possible model distortions will not be considered.

In particular, when we consider only the spatial points in the $x$-$y$ plane (we can set up $x$-$y$ plane in the scene plane), (2) can be simplified as:

$$s_i\mathbf{m}_i = \mathbf{K}(\mathbf{R}\,\mathbf{t})\begin{pmatrix} x \\ y \\ 0 \\ w \end{pmatrix} = \mathbf{K}(\mathbf{r}_1\,\mathbf{r}_2\,\mathbf{t})\begin{pmatrix} x \\ y \\ w \end{pmatrix}, \tag{3}$$

where $\mathbf{r}_1$, $\mathbf{r}_2$ are the first and second columns of the rotation matrix $\mathbf{R}$.

The image of the line at infinity of a plane in space is called a vanishing line.

Let **1**, **2**, **3**, **4**, **5** be five points on a scene plane $\pi$, and $\mathbf{m}_i$, $i = 1..5$ their images under a view. They are all 2D homogeneous coordinates with the last element 1. We assume that no three of **1**, **2**, **3**, **4** are collinear, and no three of their images are collinear.

## 3. Camera Calibration Using only Distance Ratios without Object Coordinate System

Consider **1**, **2**, **3**, **4** in the scene plane $\pi$ and their images $\mathbf{m}_i$, $i = 1..4$ under a view, we have (see Appendix A):

$$\frac{s_2}{s_1} = \frac{[\mathbf{234}][\mathbf{m}_1\mathbf{m}_3\mathbf{m}_4]}{[\mathbf{134}][\mathbf{m}_2\mathbf{m}_3\mathbf{m}_4]}, \quad \frac{s_3}{s_1} = \frac{[\mathbf{234}][\mathbf{m}_1\mathbf{m}_2\mathbf{m}_4]}{[\mathbf{124}][\mathbf{m}_2\mathbf{m}_3\mathbf{m}_4]},$$
$$\frac{s_4}{s_1} = \frac{[\mathbf{234}][\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}{[\mathbf{123}][\mathbf{m}_2\mathbf{m}_3\mathbf{m}_4]}. \tag{4}$$

Denote them as $w_2$, $w_3$, $w_4$ respectively. Since $\mathbf{r}_1$, $\mathbf{r}_2$ are two orthogonal unitary vectors, we obtain (see Appendix A):

$$\begin{aligned}\frac{(w_2\mathbf{m}_2-\mathbf{m}_1)^\tau\mathbf{K}^{-\tau}\mathbf{K}^{-1}(w_2\mathbf{m}_2-\mathbf{m}_1)}{(w_3\mathbf{m}_3-\mathbf{m}_1)^\tau\mathbf{K}^{-\tau}\mathbf{K}^{-1}(w_3\mathbf{m}_3-\mathbf{m}_1)}&=\left(\frac{d_{12}}{d_{13}}\right)^2,\\ \frac{(w_2\mathbf{m}_2-\mathbf{m}_1)^\tau\mathbf{K}^{-\tau}\mathbf{K}^{-1}(w_2\mathbf{m}_2-\mathbf{m}_1)}{(w_4\mathbf{m}_4-\mathbf{m}_1)^\tau\mathbf{K}^{-\tau}\mathbf{K}^{-1}(w_4\mathbf{m}_4-\mathbf{m}_1)}&=\left(\frac{d_{12}}{d_{14}}\right)^2,\end{aligned} \tag{5}$$

where $d_{ij}$ is the distance between the scene points **i**,**j**. Note that both the right sides of the above two equations are distance ratios. In addition, each ratio of $\frac{[\mathbf{234}]}{[\mathbf{134}]}$, $\frac{[\mathbf{324}]}{[\mathbf{124}]}$, $\frac{[\mathbf{423}]}{[\mathbf{123}]}$ in $w_2$, $w_3$, $w_4$ can be expressed as distance ratios as shown in the third and fourth paragraphs of Section 2. For example,

$$\frac{[\mathbf{234}]}{[\mathbf{134}]}=\delta_\pm\frac{S_{234}}{S_{134}}=\delta_\pm\frac{\sqrt{(r_{23/34}+r_{24/34}+1)(r_{24/34}+1-r_{23/34})(r_{23/34}+1-r_{24/34})(r_{23/34}+r_{24/34}-1)}}{\sqrt{(r_{13/34}+r_{14/34}+1)(r_{14/34}+1-r_{13/34})(r_{13/34}+1-r_{14/34})(r_{13/34}+r_{14/34}-1)}},$$

where $\mathbf{r}_{ij/kl}=\frac{d_{ij}}{d_{kl}}$, and $\delta_\pm=+$ if **1**, **2** are in the same side of the line **34**; otherwise $\delta_\pm=-$. Hence, (5) are the equations on **K** from only distance ratios.

Let $\mathbf{C}=\mathbf{K}^{-T}\mathbf{K}^{-1}$, the two equations in (5) are linear on **C**. There are only two independent equations on the intrinsic parameters from parallel scene planes. Therefore, camera intrinsic parameters can be recovered linearly and completely from three views (with different camera orientations) of a plane, or from one view of three planes with different directions.

Based on (5), here is our algorithm to calibrate a camera from distance ratios of scene points possibly distributed irregulary in differenct planes (in each plane, there are at least four points).

*Step 1*. In each plane, choose four scene points, which distribute as evenly as possible, and whose image points distribute also as evenly as possible. We denote them as **1**, **2**, **3**, **4**. From their distance ratios and their images in each view, set up the equations on $\mathbf{C}=\mathbf{K}^{-T}\mathbf{K}^{-1}$ by (5).

*Step 2*. In each view, for each of the rest scene points **5**, **6** , . . . , **i**, **M** of each plane, set up the equations on $\mathbf{C}=\mathbf{K}^{-T}\mathbf{K}^{-1}$ by substituting **4** and $\mathbf{m}_4$ with **i** and $\mathbf{m}_i$ in the equations of (5).

*Step 3*. Solve the equations established in Steps 1 and 2 for **C** at first for example by SVD method, and then recover **K** from **C** by Cholesky decomposition.

## 4. Direct Reconstruction of Plane Structure from Control Points and Error Analysis

Let **1**, **2**, **3**, **4** be four control points with known coordinates. We use subscript $x$ to indicate the first element of a vector, subscript $y$ to indicate the second element of a vector.

### 4.1. *Direct Reconstruction Solution*

Divide each side of (3) by $s_1$, and let $\mathbf{H}=\mathbf{K}(\mathbf{r}_1,\mathbf{r}_2,t)/s_1$. Then, we have $(\mathbf{m}_1, w_2\mathbf{m}_2, w_3\mathbf{m}_3)=\mathbf{H}(\mathbf{1},\mathbf{2},\mathbf{3})$, where $w_2$, $w_3$ are as in (4). Thus,

$$\begin{aligned}\mathbf{H}&=(\mathbf{m}_1, w_2\mathbf{m}_2, w_3\mathbf{m}_3)(\mathbf{1},\mathbf{2},\mathbf{3})^{-1}\\&=\frac{1}{[1\,2\,3]}(\mathbf{m}_1, w_2\mathbf{m}_2, w_3\mathbf{m}_3)\\&\quad((\mathbf{2}\times\mathbf{3})^T,(\mathbf{3}\times\mathbf{1})^T,(\mathbf{1}\times\mathbf{2})^T)^T,\end{aligned}$$

$$\begin{aligned}\mathbf{H}^{-1}&=(\mathbf{1},\mathbf{2},\mathbf{3})(\mathbf{m}_1, w_2\mathbf{m}_2, w_3\mathbf{m}_3)^{-1}\\&=\frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}(\mathbf{1},\mathbf{2},\mathbf{3})\Big((\mathbf{m}_2\times\mathbf{m}_3)^T,\\&\quad\frac{1}{w_2}(\mathbf{m}_3\times\mathbf{m}_1)^T,\frac{1}{w_3}(\mathbf{m}_1\times\mathbf{m}_2)^T\Big)^T.\end{aligned}$$

So, the direct reconstruction of a scene point **5** from its image is:

$$\begin{aligned}\mathbf{5}\approx\mathbf{H}^{-1}\mathbf{m}_5&=\frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}(\mathbf{1},\mathbf{2},\mathbf{3})\Big([\mathbf{m}_2\mathbf{m}_3\mathbf{m}_5],\\&\quad\frac{1}{w_2}[\mathbf{m}_3\mathbf{m}_1\mathbf{m}_5],\frac{1}{w_3}[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_5]\Big)^T\\&=\frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\Big([\mathbf{m}_2\mathbf{m}_3\mathbf{m}_5]\mathbf{1}\\&\quad+\frac{1}{w_2}[\mathbf{m}_3\mathbf{m}_1\mathbf{m}_5]\mathbf{2}+\frac{1}{w_3}[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_5]\mathbf{3}\Big),\end{aligned}$$

where **1**, **2**, **3** are homogeneous with the last element 1. Then, the non-homogeneous coordinates of **5** is:

$$\mathbf{5}_x=\frac{w_2w_3[\mathbf{m}_2\mathbf{m}_3\mathbf{m}_5]\mathbf{1}_x+w_3[\mathbf{m}_3\mathbf{m}_1\mathbf{m}_5]\mathbf{2}_x+w_2[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_5]\mathbf{3}_x}{w_2w_3[\mathbf{m}_2\mathbf{m}_3\mathbf{m}_5]+w_3[\mathbf{m}_3\mathbf{m}_1\mathbf{m}_5]+w_2[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_5]}, \tag{6}$$

$$\mathbf{5}_y=\frac{w_2w_3[\mathbf{m}_2\mathbf{m}_3\mathbf{m}_5]\mathbf{1}_y+w_3[\mathbf{m}_3\mathbf{m}_1\mathbf{m}_5]\mathbf{2}_y+w_2[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_5]\mathbf{3}_y}{w_2w_3[\mathbf{m}_2\mathbf{m}_3\mathbf{m}_5]+w_3[\mathbf{m}_3\mathbf{m}_1\mathbf{m}_5]+w_2[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_5]}. \tag{7}$$

The line-coordinates of the line at infinity of the scene $x$-$y$ plane is $(0, 0, 1)^T$, so the line-coordinates of its image, i.e. the vanishing line, is:

$$\begin{aligned}\mathbf{l}_o = \mathbf{H}^{-T}(0,\ 0,\ 1)^T &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left((\mathbf{m}_2\times\mathbf{m}_3), \frac{1}{w_2}\right.\\ &\left.(\mathbf{m}_3\times\mathbf{m}_1), \frac{1}{w_3}(\mathbf{m}_1\times\mathbf{m}_2)\right)(\mathbf{1}^T, \mathbf{2}^T, \mathbf{3}^T)^T(0,\ 0,\ 1)^T\\ &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left((\mathbf{m}_2\times\mathbf{m}_3), \frac{1}{w_2}(\mathbf{m}_3\times\mathbf{m}_1),\right.\\ &\left.\frac{1}{w_3}(\mathbf{m}_1\times\mathbf{m}_2)\right)(1,\ 1,\ 1)^T\\ &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left((\mathbf{m}_2\times\mathbf{m}_3) + \frac{1}{w_2}(\mathbf{m}_3\times\mathbf{m}_1),\right.\\ &\left.+\ \frac{1}{w_3}(\mathbf{m}_1\times\mathbf{m}_2)\right). \qquad (8)\end{aligned}$$

The denominator in (6) and (7) is equal to $w_2w_3[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]$ $(\mathbf{l}_o\cdot\mathbf{m}_5)$, where $\mathbf{l}_o\cdot\mathbf{m}_5$ is the inner product of $\mathbf{l}_o$ and $\mathbf{m}_5$. By (1), we know that the absolute value of the denominator can be $sd_{5o}$, where $s$ is a scalar independent of $\mathbf{m}_5$ and $\mathbf{5}$, and $d_{5o}$ is the distance from $\mathbf{m}_5$ to the vanishing line $\mathbf{l}_o$. Therefore when $\mathbf{m}_5$ moves in the image plane, the closer it is to $\mathbf{l}_o$, the smaller is the absolute value of the denominator. The denominator is zero, if and only if $\mathbf{m}_5$ is on the vanishing line $\mathbf{l}_o$, or $\mathbf{5}$ is a point at infinity.

The line-coordinates of the scene $y$-axis $x = 0$ is $(1, 0, 0)^T$, so the line-coordinates of its image is:

$$\begin{aligned}\mathbf{l}_1 = \mathbf{H}^{-T}(1,\ 0,\ 0)^T &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\\ &\left((\mathbf{m}_2\times\mathbf{m}_3), \frac{1}{w_2}(\mathbf{m}_3\times\mathbf{m}_1), \frac{1}{w_3}(\mathbf{m}_1\times\mathbf{m}_2)\right)\\ &(\mathbf{1}^T, \mathbf{2}^T, \mathbf{3}^T)^T(1,\ 0,\ 0)^T = \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left(\mathbf{1}_x(\mathbf{m}_2\times\mathbf{m}_3)\right.\\ &\left.+\ \frac{1}{w_2}\mathbf{2}_x(\mathbf{m}_3\times\mathbf{m}_1) + \frac{1}{w_3}\mathbf{3}_x(\mathbf{m}_1\times\mathbf{m}_2)\right).\end{aligned}$$

Similarly, the line-coordinates of the image from $x$-axis $(0, 1, 0)^T$ is:

$$\mathbf{l}_2 = \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left(\mathbf{1}_y(\mathbf{m}_2\times\mathbf{m}_3) + \frac{1}{w_2}\mathbf{2}_y(\mathbf{m}_3\times\mathbf{m}_1) + \frac{1}{w_3}\mathbf{3}_y(\mathbf{m}_1\times\mathbf{m}_2)\right).$$

Therefore, (6) and (7) are actually:

$$\mathbf{5}_x = \frac{\mathbf{l}_1\cdot\mathbf{m}_5}{\mathbf{l}_o\cdot\mathbf{m}_5}, \quad \mathbf{5}_y = \frac{\mathbf{l}_2\cdot\mathbf{m}_5}{\mathbf{l}_o\cdot\mathbf{m}_5} \qquad (9)$$

Let $\mathbf{v}_1 = \mathbf{l}_o\times\mathbf{l}_1$ and $\mathbf{v}_2 = \mathbf{l}_o\times\mathbf{l}_2$. We have:

$$\begin{aligned}\mathbf{v}_1 = \mathbf{l}_o\times\mathbf{l}_1 &= \mathbf{H}^{-T}(0,\ 0,\ 1)^T\times\mathbf{H}^{-T}(1,\ 0,\ 0)^T\\ &= \frac{1}{\det(\mathbf{H})}\mathbf{H}((0,\ 0,\ 1)^T\times(1,\ 0,\ 0)^T) = \frac{1}{\det(\mathbf{H})}\\ &\mathbf{H}(0,\ 1,\ 0)^T = \frac{1}{w_2w_3[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\\ &(\mathbf{m}_1, w_2\mathbf{m}_2, w_3\mathbf{m}_3)\ (\mathbf{3}_x - \mathbf{2}_x, \mathbf{1}_x - \mathbf{3}_x, \mathbf{2}_x - \mathbf{1}_x)^T\\ &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left(\frac{(\mathbf{3}_x - \mathbf{2}_x)}{w_2w_3}\mathbf{m}_1 + \frac{(\mathbf{1}_x - \mathbf{3}_x)}{w_3}\mathbf{m}_2\right.\\ &\left.+\frac{(\mathbf{2}_x - \mathbf{1}_x)}{w_2}\mathbf{m}_3\right).\end{aligned}$$

Similarly, $\mathbf{v}_2 = \mathbf{l}_o\times\mathbf{l}_2 = \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}(\frac{(\mathbf{3}_y-\mathbf{2}_y)}{w_2w_3}\mathbf{m}_1 + \frac{(\mathbf{1}_y-\mathbf{3}_y)}{w_3}\mathbf{m}_2 + \frac{(\mathbf{2}_y-\mathbf{1}_y)}{w_2}\mathbf{m}_3)$.

### 4.2. *Error Analysis for Direct Reconstruction Solution*

There are thorough error analyses on homography $\mathbf{H}$, on image point localization, and on both of them in [7, 9]. These results can be used straightforward for our direct reconstruction solution (6) and (7). In sequel, we are to have an error analysis on (6) and (7) from a new geometric standpoint.

If the control-point pattern is made appropriately the image processing for extracting the images of the control points is simple and gives accurate results, hence noise may be ignored. In the following ignoring the noise in the image points of the control points, we are to assess the reconstruction error of other points due to their image noise. Suppose the noise of $\mathbf{m}_5$ is $\Delta\mathbf{m}_5 = (\Delta u, \Delta v, 0)^T$. Let $\hat{x}_5, \hat{y}_5$ be the reconstructed results from $\mathbf{m}_5 + \Delta\mathbf{m}_5$, and let $\mathbf{m}_u = (1, 0, 0)^T, \mathbf{m}_v = (0, 1, 0)^T$, then, $\Delta\mathbf{m}_5 = (\Delta u, \Delta v, 0)^T = \Delta u\mathbf{m}_u + \Delta v\mathbf{m}_v$. Thus, by (9) we have:

$$\begin{aligned}|\hat{x}_5 - x_5| &= \left|\frac{\mathbf{l}_1\cdot(\mathbf{m}_5 + \Delta\mathbf{m}_5)}{\mathbf{l}_o\cdot(\mathbf{m}_5 + \Delta\mathbf{m}_5)} - \frac{\mathbf{l}_1\cdot\mathbf{m}_5}{\mathbf{l}_o\cdot\mathbf{m}_5}\right|\\ &= \left|\frac{(\mathbf{l}_o\cdot\mathbf{m}_5)(\mathbf{l}_1\cdot\Delta\mathbf{m}_5) - (\mathbf{l}_1\cdot\mathbf{m}_5)(\mathbf{l}_o\cdot\Delta\mathbf{m}_5)}{(\mathbf{l}_o\cdot\mathbf{m}_5)(\mathbf{l}_o\cdot\mathbf{m}_5 + \mathbf{l}_o\cdot\Delta\mathbf{m}_5)}\right|\\ &= \left|\frac{(\mathbf{l}_o\times\mathbf{l}_1)\cdot(\mathbf{m}_5\times\Delta\mathbf{m}_5)}{(\mathbf{l}_o\cdot\mathbf{m}_5)(\mathbf{l}_o\cdot\mathbf{m}_5 + \mathbf{l}_o\cdot\Delta\mathbf{m}_5)}\right|\\ &= \left|\frac{[\mathbf{v}_1\mathbf{m}_5\Delta\mathbf{m}_5]}{(\mathbf{l}_o\cdot\mathbf{m}_5)(\mathbf{l}_o\cdot\mathbf{m}_5 + \mathbf{l}_o\cdot\Delta\mathbf{m}_5)}\right|, \qquad (10)\end{aligned}$$

Similarly, there is:

$$|\hat{y}_5 - y_5| = \left| \frac{\mathbf{l}_2 \cdot (\mathbf{m}_5 + \Delta\mathbf{m}_5)}{\mathbf{l}_o \cdot (\mathbf{m}_5 + \Delta\mathbf{m}_5)} - \frac{\mathbf{l}_2 \cdot \mathbf{m}_5}{\mathbf{l}_o \cdot \mathbf{m}_5} \right| = \left| \frac{[\mathbf{v}_2 \mathbf{m}_5 \Delta\mathbf{m}_5]}{(\mathbf{l}_o \cdot \mathbf{m}_5)(\mathbf{l}_o \cdot \mathbf{m}_5 + \mathbf{l}_o \cdot \Delta\mathbf{m}_5)} \right|. \quad (11)$$

In the next, we are to analyze (10) and (11) when the scene plane and the image plane are either parallel or not parallel.

*When the Scene Plane and the Image Plane are Parallel*

If the scene plane $\pi$ is parallel to the image plane, the vanishing line $\mathbf{l}_o$ will be coincident with the line at infinity of the image plane. At the time, we have $\mathbf{l}_{ox} = 0, \mathbf{l}_{oy} = 0$ (the first and second elements of $\mathbf{l}_o$). Let $\mathbf{l}_{ow}$ be the last element of $\mathbf{l}_o$. By $\mathbf{l}_{ox}=0, \mathbf{l}_{oy}=0$, we can solve out $\frac{[124]}{[234]}$, $\frac{[134]}{[234]}$ from (8), then we substitute the results into $\mathbf{l}_{ow}$, simplify it, and obtain $\mathbf{l}_{ow} = 1$ (see Appendix B). Thus, $\mathbf{l}_o = (0, 0, 1)^T$. Since the last element of $\mathbf{m}_5$ is 1, there is $\mathbf{l}_o \cdot \mathbf{m}_5 = (0, 0, 1)^T \cdot \mathbf{m}_5 = 1$. Also, $\mathbf{l}_o \cdot \Delta\mathbf{m}_5 = (0, 0, 1)^T \cdot (\Delta u, \Delta v, 0)^T = 0$. Therefore, the absolute errors from (10) and (11) become:

$$|\hat{x}_5 - x_5| = |[\mathbf{v}_1 \mathbf{m}_5 \Delta\mathbf{m}_5]|,$$
$$|\hat{y}_5 - y_5| = |[\mathbf{v}_2 \mathbf{m}_5 \Delta\mathbf{m}_5]|.$$

Because $\mathbf{v}_i = \mathbf{l}_o \times \mathbf{l}_i, i = 1, 2$ are points on the line $\mathbf{l}_o$, thus the last elements of $\mathbf{v}_i$ are zero. Let $\overline{\mathbf{v}}_i$ be the 2-vector of the first and second elements of $\mathbf{v}_i$, $\Delta\overline{\mathbf{m}}_5$ be the 2-vector of the first and second elements of $\Delta\mathbf{m}_5$, i.e. $\Delta\bar{\mathbf{m}}_5 = (\Delta u, \Delta v)^T$, and let $\overline{\mathbf{m}}_i, i = 1..4$ be the non-homogeneous coordinates of $\mathbf{m}_i$. Then the above two equations are:

$$|\hat{x}_5 - x_5| = |[\mathbf{v}_1 \mathbf{m}_5 \Delta\mathbf{m}_5]| = \left| \begin{bmatrix} \overline{\mathbf{v}}_1, \overline{\mathbf{m}}_5, \Delta\overline{\mathbf{m}}_5 \\ 0, \quad 1, \quad 0 \end{bmatrix} \right| = |[\overline{\mathbf{v}}_1, \Delta\overline{\mathbf{m}}_5]|,$$
$$|\hat{y}_5 - y_5| = |[\mathbf{v}_2 \mathbf{m}_5 \Delta\mathbf{m}_5]| = \left| \begin{bmatrix} \overline{\mathbf{v}}_2, \overline{\mathbf{m}}_5, \Delta\overline{\mathbf{m}}_5 \\ 0, \quad 1, \quad 0 \end{bmatrix} \right| = |[\overline{\mathbf{v}}_2, \Delta\overline{\mathbf{m}}_5]|,$$

where $\overline{\mathbf{v}}_1 = \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}(\frac{(\mathbf{3}_x - \mathbf{2}_x)}{w_2 w_3}\overline{\mathbf{m}}_1 + \frac{(\mathbf{1}_x - \mathbf{3}_x)}{w_3}\overline{\mathbf{m}}_2 + \frac{(\mathbf{2}_x - \mathbf{1}_x)}{w_2}\overline{\mathbf{m}}_3)$, $\overline{\mathbf{v}}_2 = \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}(\frac{(\mathbf{3}_y - \mathbf{2}_y)}{w_2 w_3}\overline{\mathbf{m}}_1 + \frac{(\mathbf{1}_y - \mathbf{3}_y)}{w_3}\overline{\mathbf{m}}_2 + \frac{(\mathbf{2}_y - \mathbf{1}_y)}{w_2}\overline{\mathbf{m}}_3)$ (See the last paragraph of Section 4.1), they are not dependent on $\mathbf{m}_5$.

It follows that:

*Result 1.* If the scene plane $\pi$ is parallel to the image plane:

(1) The absolute errors of both $\hat{x}_5$ and $\hat{y}_5 X_5$ and $Y_5$ are linear and homogeneous on the noise $\Delta\bar{\mathbf{m}}_5 = (\Delta u, \Delta v)^T$.
(2) The values of the absolute errors are not dependent on the position of $\mathbf{m}_5$. They are only dependent on the four control points **1**, **2**, **3**, **4**, their images $\mathbf{m}_i$, $i = 1..4$, and the noise $\Delta u$, $\Delta v$.

In particularly, when **1**, **2**, **3**, **4** are the vertexes of a rectangle, we set up the $x$-$y$ world coordinate system in such a way that $\mathbf{1} = (0, 0, 1)^T, \mathbf{2} = (a, 0, 1)^T, \mathbf{3}(0, b, 1)^T, \mathbf{4} = (a, b, 1)^T$, where $a$, $b$ are the side lengths of the rectangle. At the time, the absolute errors are (Appendix C):

$$\begin{aligned} |\hat{x}_5 - x_5| &= |[\overline{\mathbf{v}}_1, \Delta\overline{\mathbf{m}}_5]| \\ &= \frac{|a|}{|[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]|} |[\overline{\mathbf{m}}_3 - \overline{\mathbf{m}}_1, \Delta\overline{\mathbf{m}}_5]| \\ &= \frac{|t_3|}{|f|} \left| \frac{s\cos\theta + f\sin\theta}{f\alpha} \Delta v - \cos\theta \Delta u \right|, \end{aligned} \quad (12)$$

$$\begin{aligned} |\hat{y}_5 - y_5| &= |[\overline{\mathbf{v}}_2, \Delta\overline{\mathbf{m}}_5]| \\ &= \frac{|b|}{|[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]|} |[\overline{\mathbf{m}}_1 - \overline{\mathbf{m}}_2, \Delta\overline{\mathbf{m}}_5]| \\ &= \frac{|t_3|}{|f|} \left| \frac{s\sin\theta + f\cos\theta}{f\alpha} \Delta v - \sin\theta \Delta u \right|, \end{aligned} \quad (13)$$

where $f$, $s$, $\alpha$ are the camera intrinsic parameters as in (2), $\theta$ is the rotation angle of the camera, and $t_3$ is the third element of the translation of the camera. When the image plane is parallel to the scene plane, $|t_3|$ is the distance of the camera to the scene plane.

Usually, for the intrinsic parameters, there is $|s| < |f|$, thus by it, and $|\cos(\theta)| \leq 1, |\sin(\theta)| \leq 1$, we have:

$$\begin{aligned} |\hat{x}_5 - x_5| &= \frac{|t_3|}{|f|} \left| \frac{s\cos\theta + f\sin\theta}{f\alpha} \Delta v - \cos\theta \Delta u \right| \\ &\leq \frac{|t_3|}{|f|} \left( \frac{2}{|\alpha|} |\Delta v| + |\Delta u| \right), \\ |\hat{y}_5 - y_5| &= \frac{|t_3|}{|f|} \left| \frac{s\sin\theta + f\cos\theta}{f\alpha} \Delta v - \sin\theta \Delta u \right| \\ &\leq \frac{|t_3|}{|f|} \left( \frac{2}{|\alpha|} |\Delta v| + |\Delta u| \right). \end{aligned}$$

If the skew is zero and the aspect ratio is 1, i.e. $s = 0$, $\alpha = 1$, then:

$$\begin{aligned}|\hat{x}_5 - x_5| &= \frac{|t_3|}{|f|}|\sin\theta\Delta v - \cos\theta\Delta u| \\ &\le \frac{|t_3|}{|f|}(|\Delta v| + |\Delta u|), \\ |\hat{y}_5 - y_5| &= \frac{|t_3|}{|f|}|\cos\theta\Delta v - \sin\theta\Delta u| \\ &\le \frac{|t_3|}{|f|}(|\Delta v| + |\Delta u|),\end{aligned}$$

By (12) and (13), the distance between the true point **5** and its reconstructed point is:

$$\begin{aligned}\Delta d &= \sqrt{(\hat{x}_5 - x_5)^2 + (\hat{y}_5 - y_5)^2} \\ &= \frac{|t_3|}{|f|}\sqrt{\frac{\Delta v^2}{\alpha^2} + \left(\frac{s}{f\alpha}\Delta v - \Delta u\right)^2} \\ &\le \frac{|t_3|}{|f|}\sqrt{\frac{\Delta v^2}{\alpha^2} + \left(\frac{|\Delta v|}{\alpha} + |\Delta u|\right)^2}.\end{aligned}$$

When $s = 0$, this distance becomes into $\Delta d = \frac{|t_3|}{|f|}\sqrt{\frac{\Delta v^2}{\alpha^2} + \Delta u^2}$. Furthermore if $\alpha = 1$, it is $\Delta d = \frac{|t_3|}{|f|}\sqrt{\Delta v^2 + \Delta u^2}$.

We have the following result:

*Result 2*. If the scene plane $\pi$ is parallel to the image plane, and the four control points are the vertexes of a rectangle:

(1) The absolute errors for both $x$ and $y$ are not larger than $\frac{|t_3|}{|f|}(\frac{2}{|\alpha|}|\Delta v| + |\Delta u|)$. If $s = 0$, $\alpha = 1$, they are not larger than $\frac{|t_3|}{f}(|\Delta v| + |\Delta u|)$. The geometric distance error $\Delta d$ is not larger than $\frac{|t_3|}{|f|}\sqrt{\frac{\Delta v^2}{\alpha^2} + (\frac{|\Delta v|}{\alpha} + |\Delta u|)^2}$. Furthermore, if $s = 0$ and $\alpha = 1$, $\Delta d$ is the product of $\frac{|t_3|}{|f|}$ and the noise norm $\sqrt{\Delta v^2 + \Delta u^2}$. When the distance between the scene plane and the camera, i.e. $|t_3|$, is small, and the focal length $f$ is large, then $\frac{|t_3|}{|f|}$ will be small, so do the absolute errors of each coordinate and the geometric error $\Delta d$.

(2) The absolute error $|\hat{x}_5 - x_5| = \frac{|a|}{|[\mathbf{m}_1, \mathbf{m}_2\mathbf{m}_3]|}|[\overline{\mathbf{m}}_3, -\overline{\mathbf{m}}_1, \Delta\overline{\mathbf{m}}_5]|$, so for the same $\Delta\overline{\mathbf{m}}_5$, $\mathbf{m}_1$, $\mathbf{m}_3$, and $a$, when the distance from $\mathbf{m}_2$ to the line $\mathbf{m}_1\mathbf{m}_3$ is large, the absolute error $|\hat{x}_5 - x_5|$ will be small. Similarly, for the same $\Delta\overline{\mathbf{m}}_5$, $\mathbf{m}_1$, $\mathbf{m}_2$, and $b$, when the distance from $\mathbf{m}_3$ to $\mathbf{m}_1\mathbf{m}_2$ is large, the error $|\hat{y}_5 - y_5|$ will be small.

(3) The geometric distance error $\Delta d$ is not related to the rotation angle $\theta$ showing that distances of two points are preserved under different rotation transformations. If $s = 0$, $\Delta u$ and $\Delta v$ are not correlated for $\Delta d^2$.

Simulations (see Section 5) show that the differences between the given upper error bounds and the errors are not large. When $\frac{|t_3|}{|f|}$ is small, the errors will be small. This is consistent with people's experiences in practice very much.

We also represent the relative error of the estimated distance between two reconstructed points and find that if the noise of the two image points is equal, the error is zero. This is because at the time the two reconstructed points and the two true points form a parallelogram. This relative error is also not related to the rotation angle.

*When the Scene Plane and the Image Plane are not Parallel*

When the scene plane and the image plane are not parallel, the vanishing line $\mathbf{l}_o$ is not coincident with the line at infinity in the image plane. Thus, it is a line not at infinity.

By the first order of Taylor expansion at (0,0) for $(\Delta u, \Delta v)$, (10) and (11) become into:

$$\begin{aligned}|\hat{x}_5 - x_5| &= \left|\frac{[\mathbf{v}_1\mathbf{m}_5\Delta\mathbf{m}_5]}{(\mathbf{l}_o\cdot\mathbf{m}_5)(\mathbf{l}_o\cdot\mathbf{m}_5 + \mathbf{l}_o\cdot\Delta\mathbf{m}_5)}\right| \\ &\approx \left|\frac{[\mathbf{v}_1\mathbf{m}_5\Delta\mathbf{m}_5]}{(\mathbf{l}_o\cdot\mathbf{m}_5)^2}\right| \\ &= \left|\frac{[\mathbf{v}_1\mathbf{m}_5\mathbf{m}_u]\Delta u + [v_1\mathbf{m}_5\mathbf{m}_v]\Delta v}{(\mathbf{l}_o\cdot\mathbf{m}_5)^2}\right|, \\ |\hat{y}_5 - y_5| &= \left|\frac{[\mathbf{v}_2\mathbf{m}_5\Delta\mathbf{m}_5]}{(\mathbf{l}_o\cdot\mathbf{m}_5)(\mathbf{l}_o\cdot\mathbf{m}_5 + \mathbf{l}_o\cdot\Delta\mathbf{m}_5)}\right| \\ &\approx \left|\frac{[\mathbf{v}_2\mathbf{m}_5\Delta\mathbf{m}_5]}{(\mathbf{l}_o\cdot\mathbf{m}_5)^2}\right| \\ &= \left|\frac{[\mathbf{v}_2\mathbf{m}_5\mathbf{m}_u]\Delta u + [v_2\mathbf{m}_5\mathbf{m}_v]\Delta v}{(\mathbf{l}_o\cdot\mathbf{m}_5)^2}\right|.\end{aligned}$$

where $\mathbf{m}_u = (1, 0, 0)^T$, $\mathbf{m}_v = (0, 1, 0)^T$, $\Delta\mathbf{m}_5 = (\Delta u, \Delta v, 0)^T = \Delta u\mathbf{m}_u + \Delta v m_v$.

The distance between the true point **5** and its reconstructed point is:

$$\begin{aligned}\Delta d &= \sqrt{(\hat{x}_5 - x_5)^2 + (\hat{y}_5 - y_5)^2} \\ &\approx \frac{\sqrt{[\mathbf{v}_1\mathbf{m}_5\Delta_5]^2 + [\mathbf{v}_2\mathbf{m}_5\Delta\mathbf{m}_5]^2}}{(\mathbf{l}_o\cdot\mathbf{m}_5)^2}\end{aligned}$$

Thus, we have:

*Result 3.*

(1) If $\mathbf{m}_5$ is moving on the line $\mathbf{v}_1\mathbf{m}_u$, i.e. the line through $\mathbf{v}_1$ and parallel to $u$ axis, then $[\mathbf{v}_1\mathbf{m}_5\mathbf{m}_u] = 0$, at the time, the absolute error $|\hat{x}_5 - x_5|$ is mainly from $\Delta v$. Similarly, if $\mathbf{m}_5$ is moving on the line through $\mathbf{v}_1$ and parallel to $v$ axis, $|\hat{x}_5 - x_5|$ is mainly from $\Delta u$.

(2) If $\mathbf{m}_5$ is moving on the line $\mathbf{v}_2\mathbf{m}_u$, i.e. the line through $\mathbf{v}_2$ and parallel to $u$ axis, then $[\mathbf{v}_2\ \mathbf{m}_5\ \mathbf{m}_u] = 0$, at the time, the absolute error $|\hat{y}_5 - y_5|$ is mainly from $\Delta v$. Similarly, if $\mathbf{m}_5$ is moving on the line through $\mathbf{v}_2$ and parallel to $v$ axis, $|\hat{y}_5 - y_5|$ is mainly from $\Delta u$.

(3) When the noise is small, we can assume $\Delta u = \Delta v = \delta$. Then, $|\hat{x}_5 - x_5| \approx |[\mathbf{v}_1, \mathbf{m}_5, \mathbf{m}_u + \mathbf{m}_v]|\delta/(\mathbf{l}_o \cdot \mathbf{m}_5)^2$, $|\hat{y}_5 - y_5| \approx |[\mathbf{v}_2, \mathbf{m}_5, \mathbf{m}_u + \mathbf{m}_v]|\delta/(\mathbf{1}_o \cdot \mathbf{m}_5)^2$. Let $d_1$, $d_2$, $d_o$ be the distances from $\mathbf{m}_5$ to the lines $\mathbf{v}_1(\mathbf{m}_u + \mathbf{m}_v)$, $\mathbf{v}_2(\mathbf{m}_u + \mathbf{m}_v)$, $\mathbf{l}_o$ respectively. Thus, by (1), there are: $|\hat{x}_5 - x_5| \approx s_1 d_1 \delta / s_o^2 d_o^2$, $|\hat{y}_5 - y_5| \approx s_2 d_2 \delta / s_o^2 d_o^2$, where $s_1, s_2, s_o$ are not dependent on $\mathbf{m}_5$. In the region $d_1 \le d_o$, $|\hat{x}_5 - x_5| \approx s_1 d_1 \delta / s_o^2 d_o^2 \le s_1 s_o^2 d_o$. So under the same noise, $|\hat{x}_5 - x_5|$ tends to decrease with $\mathbf{m}_5$ moving away from $\mathbf{l}_o$. Similarly, $|\hat{y}_5 - y_5|$ tends to decrease with $\mathbf{m}_5$ moving away from $\mathbf{l}_o$ in the region of $d_2 \le d_o$. Therefore, in the region satisfying both $d_1 \le d_o$ and $d_2 \le d_o$, $\Delta d \approx \frac{\sqrt{(s_1^2 d_1^2 + s_2^2 d_2^2)\delta^2}}{s_o^2 d_o^2} \le \frac{\sqrt{s_1^2 + s_2^2}|\delta|}{s_o^2 d_o}$. This error $\Delta d$ also tends to decrease with $\mathbf{m}_5$ moving away from $\mathbf{l}_o$ under the same noise.

*Remark.*

(1) We have done some simulations (see Section 5) showing that whether $d_1 \le d_o$, $d_2 \le d_o$ or not, the absolute errors tend to decrease with $\mathbf{m}_5$ moving away from $\mathbf{l}_o$. But if $d_1 \le d_o$, $d_2 \le d_o$, the errors decrease fast. Thus, in practice in order to improve the accuracy of vision task, we can let the image region be as far as possible from the vanishing line. When the image plane and scene plane are parallel, the vanishing line is coincident with the line at infinity of the image plane, at the time the distance from the image region to the vanishing line is maximum.

(2) $\mathbf{l}_o$ is the vanishing line of the scene plane usually not appearing in the image region and being far away from the image region. It is determined by $\mathbf{m}_i$, $= 1..4$. When $\mathbf{m}_5$ moves away from $\mathbf{l}_o$, relatively $\mathbf{m}_5$ moves close to $\mathbf{m}_i$, $i = 1..4$. Therefore, the points close to the control points are reconstructed more precisely.

(3) When considering the noise in the image points $\mathbf{m}_i$, $i = 1..4$ of the control points, the symbol error expressions can also be obtained from the given closed-form solutions (6) and (7). But, the expressions are complex. So we do error analyses by repeated experiments. We find that the corresponding item 3 of Result 3 is also right if the noise levels in the image points of the control points are all equal, and if the difference between these noise levels and the noise level in other image point is larger than a threshold.

## 5. Experiments

In this section, experiments are carried out to test the results of Sections 3 and 4.

### 5.1. *Simulations for Calibration Using Distance Ratios Without Object Coordinate System*

This section is to test the camera calibration method of Section 3 by simulations.

The simulated camera intrinsic parameters are:

$$\mathbf{K} = \begin{bmatrix} f_u & s & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1000 & 0 & 512 \\ 0 & 900 & 384 \\ 0 & 0 & 1 \end{bmatrix}$$

We take five coplanar scene points (distributed irregularly), and then project them to the simulated image planes at three different positions. The images are of size $500 \times 600$, $800 \times 900$, $600 \times 1200$ pixels, and shown in Fig. 1. Gaussian noise with mean 0 and standard deviation ranging from 0 to 2.0 pixels is directly added to the five image points, and then the intrinsic parameters are computed from distance ratios by the method in Section 3. For each noise level, we perform 100 times independent experiments, and the averaged results are shown in Table 1. The results validate this method and show its stability. Under noise level 2, the relative errors of $f_v$, $f_u$, $u_o$, $v_o$ are respectively 0.2695, 0.3714, 0.3707, and 0.2198, which are satisfactory.

### 5.2. *Real Experiments for Calibration using Distance Ratios without Object Coordinate System*

This section is to test the camera calibration method of Section 3 by real experiments.

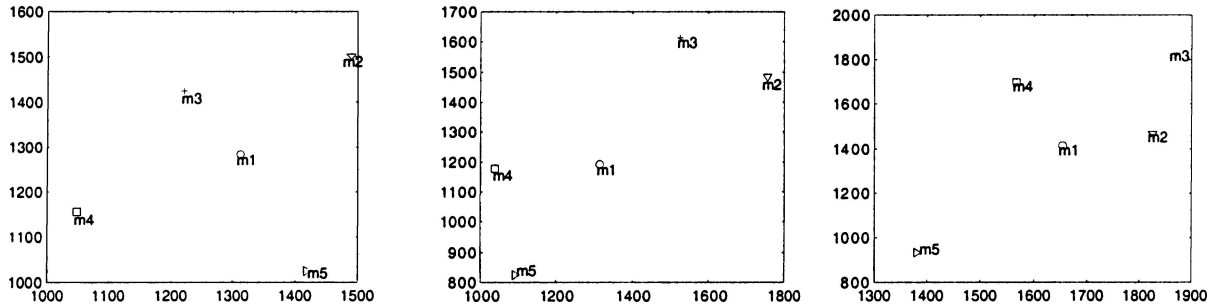



*Figure 1.* Three simulated images for camera calibration from distance ratios.

We take four images by a COOLPIX5700 camera as shown in Fig. 2. The image size is of 1024 × 768 pixels. Ten corners in each image are extracted by OpenCV, and denoted as $\mathbf{m}_i$, $i = 1..10$, as shown in the first image. The corresponding spatial points of these image points are denoted as **1**,. . .,**10**. They distribute irregularly. We use the distance ratios from the spatial points **1**,. . .,**6** to calibrate the camera by the method of Section 3. The result is:

$$\mathbf{K} = \begin{pmatrix} 568.91 & -39.05 & 432.89 \\ 0 & 416.57 & 368.80 \\ 0 & 0 & 1 \end{pmatrix}.$$

In order to verify the estimated **K**, we use it to compute some distances $d_{jk}$ from **7**, **8**, **9**, **10** by the equations (see (5)):

$$d_{1i}^2 = s_0(w_i\mathbf{m}_i - \mathbf{m}_1)^T\mathbf{K}^{-T}\mathbf{K}^{-1}(w_i\mathbf{m}_i - \mathbf{m}_1),$$
$$d_{1j}^2 = s_0(w_j\mathbf{m}_j - \mathbf{m}_1)^T\mathbf{K}^{-T}\mathbf{K}^{-1}(w_j\mathbf{m}_j - \mathbf{m}_1),$$
$$d_{ij}^2 = s_0(w_i\mathbf{m}_i - w_j\mathbf{m}_j)^T\mathbf{K}^{-T}\mathbf{K}^{-1}(w_i\mathbf{m}_i - w_j\mathbf{m}_j),$$

where $s_0 = d_{1q}^2/(w_q\mathbf{m}_q - \mathbf{m}_1)^T\mathbf{K}^{-T}\mathbf{K}^{-1}(w_q\mathbf{m}_q - \mathbf{m}_1)$, $q = 2..6$. Due to noise, we use the average of these five values of $s_0$ in the distance calculations. And $w_i$, $w_j$ are computed by some distance ratios. For example, $w_7 = 2.7712 * [\mathbf{m}_1\mathbf{m}_3\mathbf{m}_4]/[\mathbf{m}_3\mathbf{m}_4\mathbf{m}_7]$, where $2.7712 = d_{7,34}/d_{1,34}$, $d_{7,34}$ is the distance of **7** to the line **34**, and $d_{1,34}$ is the distance of **1** to the line **34**. Some calculated results are:

$$d_{17} = 22.5638\,\text{cm}, \quad d_{18} = 21.9373\,\text{cm},$$
$$d_{78} = 15.8947\,\text{cm}, \quad d_{19} = 25.9094\,\text{cm},$$
$$d_{1,10} = 20.5116\,\text{cm}, \quad d_{9,10} = 18.1777\,\text{cm}.$$

The ground truths for them are:

$$d_{17} = 23.3\,\text{cm}, \quad d_{18} = 22.2\,\text{cm}, \quad d_{78} = 16\,\text{cm},$$
$$d_{19} = 25.6\,\text{cm}, \quad d_{1,10} = 21.2\,\text{cm}, \quad d_{9,10} = 18.3\,\text{cm}.$$

We can see that the estimated distances are all quite close to their ground truths. These indirectly validate the estimated intrinsic parameters **K**.

*Table 1.* The estimated camera intrinsic parameters from distance ratios under different noise levels (pixel).

| Noise levels | $f_u$ | $f_v$ | s | $u_0$ | $v_0$ |
|---|---|---|---|---|---|
| 0 | 1000 | 900 | 0 | 512 | 384 |
| 0.4 | 1016.52 | 914.66 | −2.97 | 520.77 | 384.84 |
| 0.8 | 1065.49 | 965.65 | −10.33 | 551.04 | 395.20 |
| 1 | 1125.16 | 1017.65 | −23.26 | 558.84 | 379.02 |
| 1.5 | 1234.93 | 1158.20 | −42.20 | 644.95 | 402.08 |
| 2.0 | 1269.53 | 1234.25 | −58.83 | 701.82 | 468.41 |

### 5.3. *Simulations for Error Analysis of Direct Reconstruction Solution*

In this section, simulations are performed to test Results 1, 2 and 3 in Section 4.

*For Result 1 (the Scene Plane and the Image Plane are Parallel)*

The simulated camera intrinsic parameters are:

$$\mathbf{K} = \begin{pmatrix} 1000 & 0.1 & 512 \\ 0 & 900 & 384 \\ 0 & 0 & 1 \end{pmatrix}.$$

The *x*-*y* plane of the world coordinates system is set up as the scene plane. And, the rotation axis, rotation angle, and the translation vector of the camera are:

$$\mathbf{r} = (0, 0, 1)^T, \quad \theta = 60^\circ, \quad \mathbf{T} = (-150, -130, -180)^T.$$

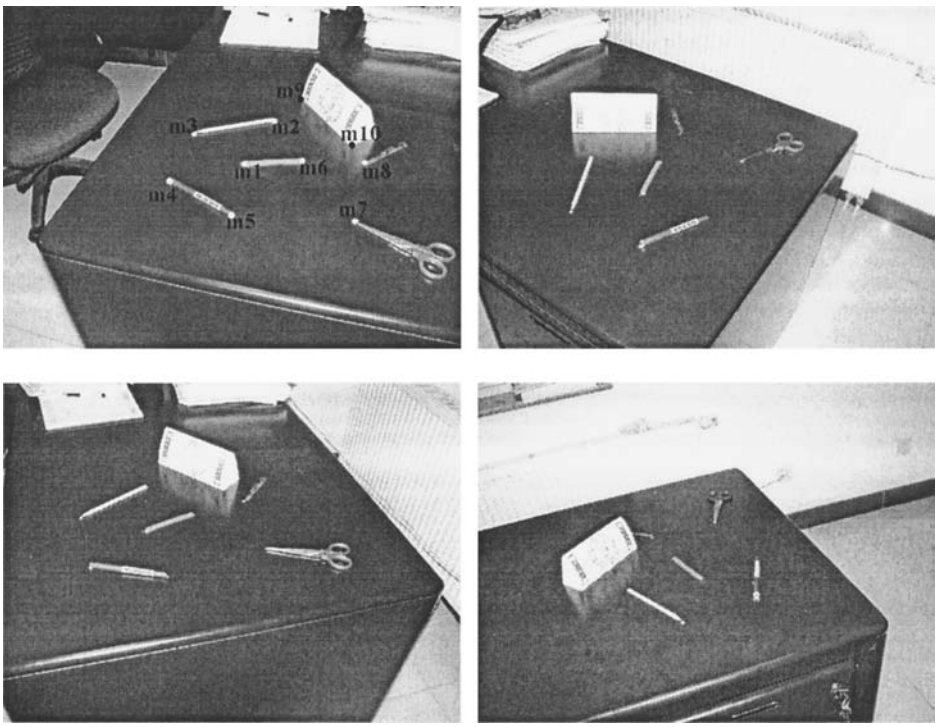



*Figure 2*. The real images for camera calibration.

The points $(0, 0, 1)^T, (40, 0, 1)^T, (0, 30, 1)^T, (40, 30, 1)^T$ on the $x$-$y$ plane are taken as the control points. An image of size $1000 \times 1000$ shown in Fig. 3 is generated from the four control points as well as regularly distributed 144 test points, where the points '+' are from the four control points and the points '$\circ$' are from the 144 test points.

The noise levels of 0, 0.4, 0.8, 1.2, 1.6, 2 pixels are directly added to each coordinate of each of the 144 image points, then we perform the reconstruction from them by (6) and (7), and compute the absolute error of each coordinates. We find that, under each noise level, all of the errors ($<1$) for the 144 points have the same first 12 digits, see columns 2 and 3 of Table 2. So, the absolute errors of reconstruction are independent of the position of reconstructed points, this validates the second item of Result 1. Thus, in the following simulations for Results 1 and 2, we use the mean of the absolute errors of the 144 points as the absolute error under each noise level.

We compute the absolute error under each noise level and plot them with varying noise. The result is shown in Fig. 4. As it can be seen, the relations of the absolute errors and noise are linear, which is consistent with the first item of Result 1.

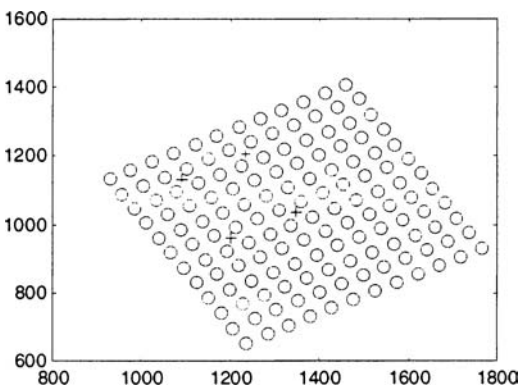



*Figure 3*. An image of 4 control points and 144 image points.

*Table 2*. The same first 13,12 digits of absolute errors for $x$, $y$, and the same first 13 digits of the distances $\Delta d$ from the 144 points.

| Noise level (pixel) | The absolute error of $x$ | The absolute error of $y$ | The distance error $\Delta d$ |
|---|---|---|---|
| 0 | 0.000000000000 | 0.000000000000 | 0.000000000000 |
| 0.4 | 0.033286032302 | 0.102346900869 | 0.107623640823 |
| 0.8 | 0.066572064605 | 0.204693801738 | 0.215247281646 |
| 1.2 | 0.099858096908 | 0.307040702607 | 0.322870922469 |
| 1.6 | 0.13314412921 | 0.40938760347 | 0.430494563292 |
| 2 | 0.166430161513 | 0.511734504346 | 0.538118204115 |

*Table 3.* The given upper error bound under each nonzero noise level.

| Noise level (pixel) | 0.4 | 0.8 | 1.2 | 1.6 | 2 |
|---|---|---|---|---|---|
| Upper error bound for $x$ and $y$ | 0.232 | 0.464 | 0.696 | 0.928 | 1.16 |
| Upper error bound for $\Delta d$ | 0.172 | 0.344 | 0.515 | 0.687 | 0.859 |

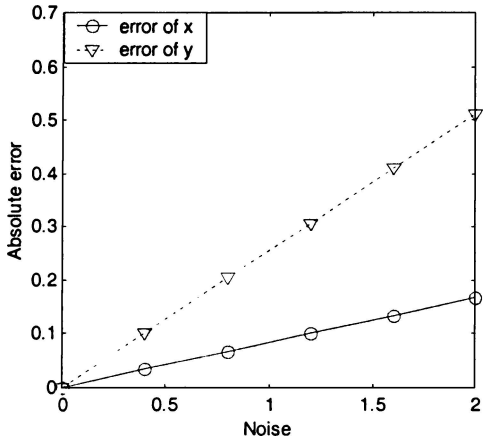



*Figure 4.* The absolute error vs. noise.

*For Result 2 (the Scene Plane and the Image Plane are Parallel)*

We compute the upper error bound $\frac{|t_3|}{|f|}(\frac{2}{|\alpha|}|\Delta v|+|\Delta u|)$ for both $x$ and $y$ and compute the upper error bound $\frac{|t_3|}{|f|}\sqrt{\frac{\Delta v^2}{\alpha^2}+(\frac{|\Delta u|}{\alpha}+|\Delta u|)}$ for $\Delta d$ given in Result 2 by using the same simulated data for Result 1 under each nonzero noise level. The result is shown in Table 3. By comparing Table 3 with Table 2, we can see that the given upper error bounds are always larger than the errors for $x$, $y$ and $\Delta d$ under each noise level, but the differences between them are not large.

We also have done the corresponding simulations to test the given error analyses when $s = 0$ and $\alpha = 1$, similar results are obtained.

The performance of the absolute errors and the distance between the true point and its reconstruction with varying focal length is assessed next. The simulated camera intrinsic parameters are:

$$\mathbf{K} = \begin{pmatrix} f & 0.1 & 512 \\ 0 & 0.9^* f & 384 \\ 0 & 0 & 1 \end{pmatrix}$$

with $f$ varying from 900 to 1260 with a step of 40.

The images with $f = 900$, $f = 1260$ are shown in Fig. 5. Their sizes are not larger than $1200 \times 1200$ pixels.

The performance of the absolute errors and the distance between the true point and its reconstruction with varying $f$ under noise level 1.2 pixels is shown in Fig. 6. As it can be seen, the errors are decreasing with $f$ increasing. Under other nonzero noise levels, we have the similar results.

Similarly, we can verify that the errors are decreasing with $t_3$ decreasing.

All of the above are consistent with the first item of Result 2. The second item of Result 2 is also validated by the similar simulation as for Result 3. The third item of Result 3 holds clearly, so we omit the test for it.

*For Result 3 (the Scene Plane and the Image Plane are not Parallel)*

It is clear that the first and second items of Result 3 hold. We are to test the third item of Result 3. An image of size $1200 \times 700$ is generated as shown in Fig. 7.

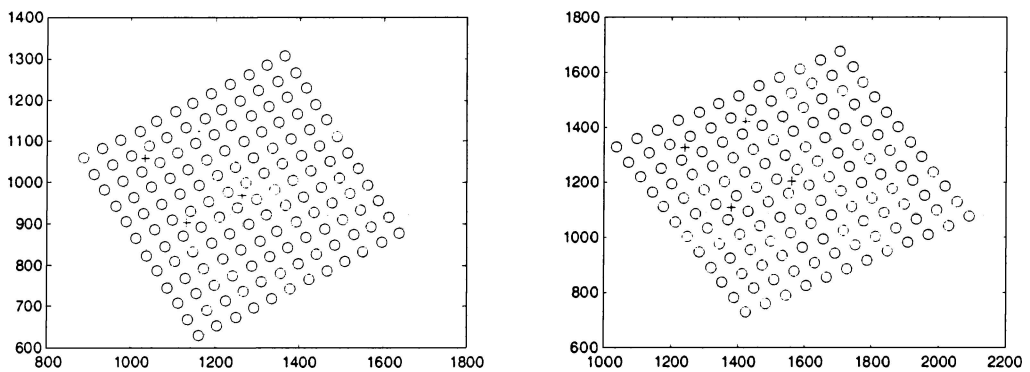



*Figure 5.* Two simulated images for testing the performance of the errors with varying focal length $f$, where the left one is the image with $f = 900$, the right one is the image with $f = 1260$.

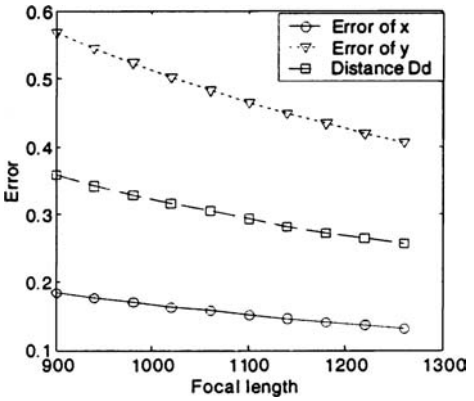



*Figure 6.* The absolute errors of $x$, $y$, and the distance $\Delta d$ between the true point and its reconstruction vs. the focal length, where Dd denotes $\Delta d$.

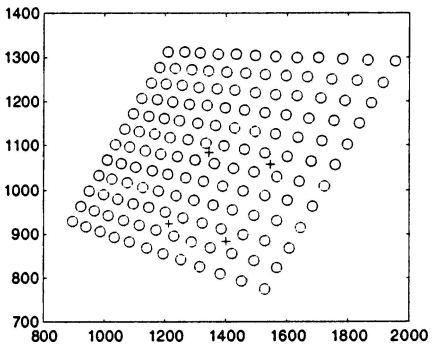


*Figure 7.* A simulated image for Result 3.

The same noise levels as before are added into each of the 144 image points. Under each noise level, we compute the distances $d_1$, $d_2$, $d_o$ from each image point to the lines $\mathbf{v}_1(\mathbf{m}_u + \mathbf{m}_v)$, $\mathbf{v}_2(\mathbf{m}_u + \mathbf{m}_v)$, and the vanishing line $\mathrm{I}_o$ (with notations as before). For the 144 different image points and different noise, we find there are always $d_1 < d_o$, $d_2 > d_o$. We plot the absolute errors with varying $d_o$ under each nonzero noise level, the results are shown in Fig. 8. Although $d_2 > d_o$, the absolute error of the second coordinate $y$ still tends to decrease with $d_o$ increasing. It is clear that the absolute error of the first coordinate $x$ tends to decrease fast with $d_o$ increasing. There is the same result for the distance $\Delta d$ between the true point and its reconstruction. This result is not plotted due to its nearly coinciding with the result for the absolute error of $x$. We can see that the absolute errors of $y$ are less than those of $x$ so it is natural that the distances $\Delta d$ for different points are mainly from those of $x$.

We also add noise to the image points of the control points, and find that there are the same results as Fig. 8 if the noise levels in the image points from the control points are all equal, and if the difference between these noise levels and the noise level in other image point is larger than a threshold.

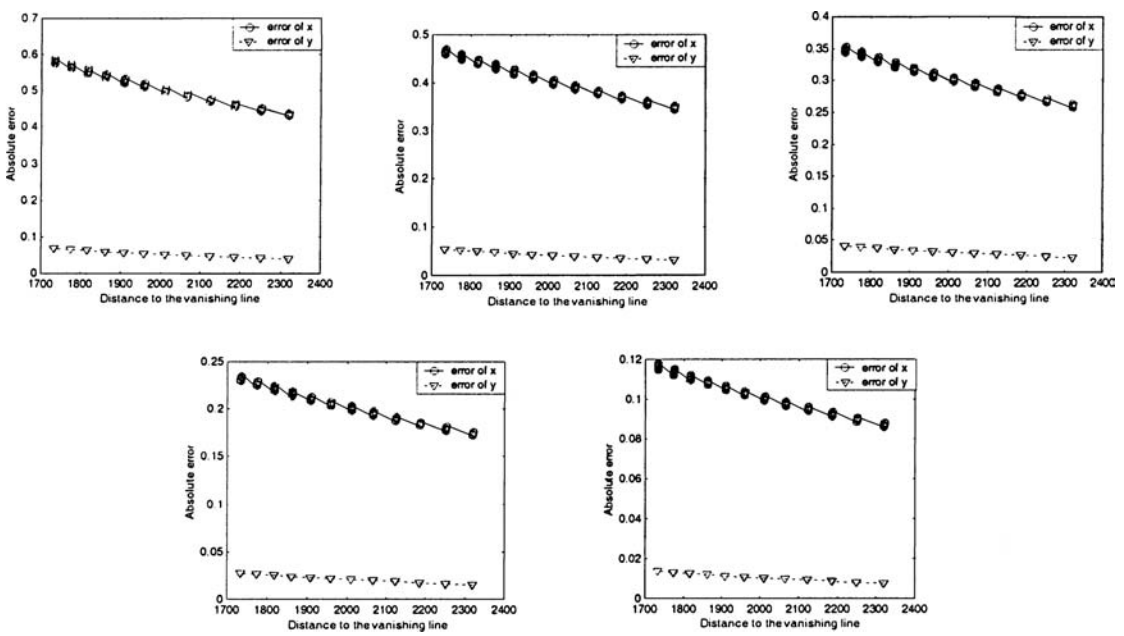
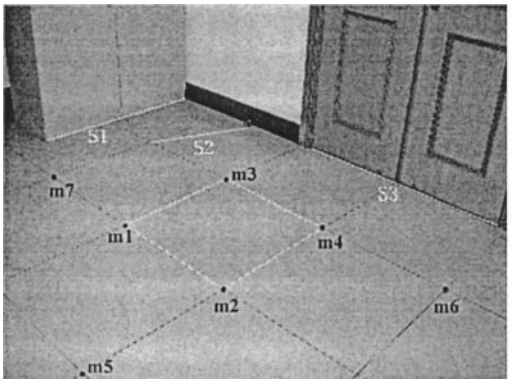



*Figure 8.* The absolute errors with the image point moving, where the horizontal axis indicates the distance from the image point to the vanishing line, the vertical axis indicates the absolute errors. These five figures are respectively under noise level of 2, 1.6, 1.2, 0.8, 0.4 pixels.



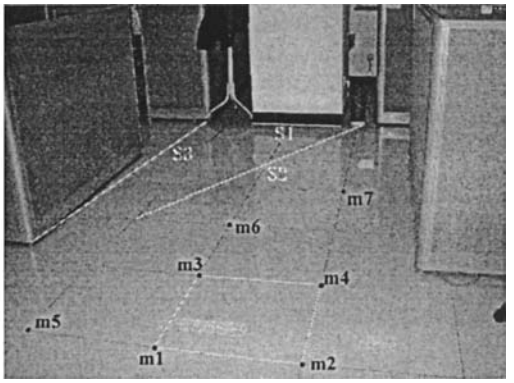




*Figure 9.* Two real indoor images for direct reconstruction solution.

### 5.4. *Real Experiments for the Direct Reconstruction Solution*

We use a Nikon COOLPIX990 camera to take two images of our lab floor as shown in Fig. 9. The images are of size 1024 × 768 pixels. We perform two independent experiments from each of the two images as follows.

The points from a rectangle marked by white color are extracted by the Canny edge detector and then are fitted by the least squares method as four lines. The intersections of the four lines are computed as the images of four control points $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3, \mathbf{m}_4$. We reconstruct the end points of three line segments S1, S2, S3 (see Fig. 9) by the direct reconstruction solution (6) and (7) and calculate the lengths of these line segments from the reconstructed points. The results are shown in Table 4. In addition, we reconstruct $\mathbf{m}_5, \mathbf{m}_6, \mathbf{m}_7$ of the left image and the right image in Fig. 9 by (6) and (7). The results are shown in Table 5. From Tables 4 and 5, we can see the results are satisfactory.

Outdoor scene is also used to test our direct reconstruction solution. We take an image of our institute's main building by a Nikon COOLPIX5700 camera, as shown in Fig. 10. The image is of size 2560 × 1920 pixels. We extract 12 corner image points by hand as shown in Fig. 10, then use the spatial points of $\mathbf{m}_i, i = 1..4$ as the control points to recover the spatial

*Table 4.* The test results for measurement from Fig. 9.

| | The left image | | | The right image | | |
|---|---|---|---|---|---|---|
| Line segment | S1 | S2 | S3 | S1 | S2 | S3 |
| True distance (mm) | 980 | 600 | 1080 | 815 | 2550 | 2700 |
| Estimated distance (mm) | 974.8 | 592.8 | 1068.5 | 811.6 | 2486.9 | 2665.9 |


*Figure 10.* A real outdoor image for direct reconstruction solution.

points of $\mathbf{m}_i, i = 5..12$ by (6) and (7). The distances $d_{ij}$ between pairs of the recovered spatial points from $\mathbf{m}_i, \mathbf{m}_j$ are computed as:

$$d_{35} = 205.2258\,\text{cm}, \quad d_{67} = 203.9955\,\text{cm},$$
$$d_{37} = 68.9452\,\text{cm}, \quad d_{56} = 67.6797\,\text{cm},$$
$$d_{78} = 128.7998\,\text{cm}, \quad d_{89} = 91.5342\,\text{cm},$$
$$d_{10,11} = 1653.6\,\text{cm}, \quad d_{12,13} = 815.6833\,\text{cm}.$$

Their corresponding ground truths are:

$$d_{35} = d_{67} = 202\,\text{cm}, \quad d_{37} = d_{56} = 64.5\,\text{cm},$$
$$d_{78} = 124.5\,\text{cm}, \quad d_{89} = 84.5\,\text{cm},$$
$$d_{10,11} = 1756.8\,\text{cm}, \quad d_{12,13} = 881.6\,\text{cm}.$$

We can see that the estimations and their ground truths are rather close.

*Table 5.* The test results for reconstruction from Fig. 9.

| Image point | The left image | | | The right image | | |
|---|---|---|---|---|---|---|
| | $\mathbf{m}_5$ | $\mathbf{m}_6$ | $\mathbf{m}_7$ | $\mathbf{m}_5$ | $\mathbf{m}_6$ | $\mathbf{m}_7$ |
| True space coordinates (mm) | (600, −600) | (1200, 600) | (−600, 0) | (−600, 0) | (0, 1200) | (600, 1800) |
| Estimated space coordinates (mm) | (596, −590) | (1179, 602) | (−591, 4.4) | (−586, 5.5) | (−1.7, 1202) | (594, 1800) |

## 6. Conclusions

With bracket computations, we have presented a new camera calibration method from only distance ratios of scene points. This method does not need to set up any world coordinate system and thus can use the geometric information of irregular objects conveniently. Moreover, a direct reconstruction solution of plane structure is provided. Thanks to the bracket-based computations, the solution representation becomes concise and short, from which some new and useful error analysis results are obtained. Simulated and real experiments confirm the validity of the camera calibration method from distance ratios, the direct reconstruction solution and the error analysis results.

## Appendix A

We are to prove (4) and (5).

Let $\mathbf{H} = \mathbf{K}(\mathbf{r}_1\mathbf{r}_2\mathbf{t})$. Then, by (3) we have: $s_i\mathbf{m}_i = \mathbf{H}\mathbf{i}$, where $i = 1, 2, 3, 4$ and $\mathbf{i} = \mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}$. Thus, $s_i s_j s_n[\mathbf{m}_i\mathbf{m}_j\mathbf{m}_n] = \det(\mathbf{H})[\mathbf{ijn}]$. So, $\frac{s_i[\mathbf{m}_i\mathbf{m}_j\mathbf{m}_n]}{s_k[\mathbf{m}_k\mathbf{m}_j\mathbf{m}_n]} = \frac{s_i s_j s_n[\mathbf{m}_i\mathbf{m}_j\mathbf{m}_n]}{s_k s_j s_n[\mathbf{m}_k\mathbf{m}_j\mathbf{m}_n]} = \frac{\det(\mathbf{H})[\mathbf{ijn}]}{\det(\mathbf{H})[\mathbf{kjn}]} = \frac{[\mathbf{ijn}]}{[\mathbf{kjn}]}$. Hence, $\frac{s_i}{s_k} = \frac{[\mathbf{ijn}][\mathbf{m}_k\mathbf{m}_j\mathbf{m}_n]}{[\mathbf{kjn}][\mathbf{m}_i\mathbf{m}_j\mathbf{m}_n]}$ and (4) is proved.

Denote $\bar{\mathbf{i}}$ as the non-homogeneous coordinates of $\bar{\mathbf{i}}$, then $\mathbf{i} = (\bar{\mathbf{i}}^T, 1)^T$. Thus, (3) can be changed into:

$$s_i\mathbf{K}^{-1}\mathbf{m}_i = (\mathbf{r}_1\mathbf{r}_2)\bar{\mathbf{i}} + \mathbf{t}\ .$$

Subtract the equation for $i = 1$ from the equations for $i = 2, 3, 4$, we obtain: $\mathbf{K}^{-1}(s_i\mathbf{m}_i - s_1\mathbf{m}_1) = (\mathbf{r}_1\mathbf{r}_2)(\bar{\mathbf{i}} - \bar{\mathbf{1}})$, $i = 2, 3, 4$ and $\bar{\mathbf{i}} = \bar{\mathbf{2}}, \bar{\mathbf{3}}, \bar{\mathbf{4}}$. Since $\mathbf{r}_1, \mathbf{r}_2$ are two orthogonal unitary vectors, we have: $(s_i\mathbf{m}_i - s_1\mathbf{m}_1)^T\mathbf{K}^{-T}\mathbf{K}^{-1}(s_i\mathbf{m}_i - s_1\mathbf{m}_1) = (\bar{\mathbf{i}} - \bar{1})^T(\bar{\mathbf{i}} - \bar{\mathbf{1}}) = d_{1i}^2$. Then, there is:

$$\frac{(s_i\mathbf{m}_i - s_1\mathbf{m}_1)^T\mathbf{K}^{-T}\mathbf{K}^{-1}(s_i\mathbf{m}_i - s_1\mathbf{m}_1)}{(s_j\mathbf{m}_j - s_1\mathbf{m}_1)^T\mathbf{K}^{-T}\mathbf{K}^{-1}(s_j\mathbf{m}_j - s_1\mathbf{m}_1)} = \frac{d_{1i}^2}{d_{1j}^2}\ .$$

Divide $s_1$ from the numerator and denominator, we have (5).

## Appendix B

We are to prove that the last element of the vanishing line $\mathbf{l}_o$ of (8) is 1 when the scene plane and image plane are parallel. Denote the first, second, last element of $\mathbf{l}_o$ as $\mathbf{l}_{ox}, \mathbf{l}_{oy}, \mathbf{l}_{ow}$.

Since the scene plane and image plane are parallel, the vanishing line is coincident to the line at infinity of the image plane. So, $\mathbf{l}_{ox} = 0, \mathbf{l}_{oy} = 0$. By (8), we have:

$$\begin{aligned}\mathbf{l}_{ox} &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}((\mathbf{m}_2 \times \mathbf{m}_3)_x + \frac{1}{w_2}(\mathbf{m}_3 \times \mathbf{m}_1)_x \\ &\quad + \frac{1}{w_3}(\mathbf{m}_1 \times \mathbf{m}_2)_x) = 0, \\ \mathbf{l}_{oy} &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}((\mathbf{m}_2 \times \mathbf{m}_3)_y + \frac{1}{w_2}(\mathbf{m}_3 \times \mathbf{m}_1)_y \\ &\quad + \frac{1}{w_3}(\mathbf{m}_1 \times \mathbf{m}_2)_y) = 0\ .\end{aligned}$$

By Cramer's rule from these two equations, we can solve out:

$$\begin{aligned}\frac{1}{w_2} &= \frac{\begin{bmatrix} -(\mathbf{m}_2 \times \mathbf{m}_3)_x & (\mathbf{m}_1 \times \mathbf{m}_2)_x \\ -(\mathbf{m}_2 \times \mathbf{m}_3)_y & (\mathbf{m}_1 \times \mathbf{m}_2)_y \end{bmatrix}}{\begin{bmatrix} (\mathbf{m}_3 \times \mathbf{m}_1)_x & (\mathbf{m}_1 \times \mathbf{m}_2)_x \\ (\mathbf{m}_3 \times \mathbf{m}_1)_y & (\mathbf{m}_1 \times \mathbf{m}_2)_y \end{bmatrix}} \\ &= \frac{((\mathbf{m}_1 \times \mathbf{m}_2) \times (\mathbf{m}_2 \times \mathbf{m}_3)) \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}}{((\mathbf{m}_3 \times \mathbf{m}_1) \times (\mathbf{m}_1 \times \mathbf{m}_2)) \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}} \\ &= \frac{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]\mathbf{m}_2 \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]\mathbf{m}_1 \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}} = 1\ ,\end{aligned}$$

where the last elements of $\mathbf{m}_1, \mathbf{m}_2$ are 1. Similarly, we can obtain $\frac{1}{w_3} = 1$. Thus, the last element of $\mathbf{l}_o$ is:

$$\begin{aligned}\mathbf{l}_{ow} &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left((\mathbf{m}_2 \times \mathbf{m}_3)_w + \frac{1}{w_2}(\mathbf{m}_3 \times \mathbf{m}_1)_w \right.\\ &\quad \left. + \frac{1}{w_3}(\mathbf{m}_1 \times \mathbf{m}_2)_w\right)\\ &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}((\mathbf{m}_2 \times \mathbf{m}_3)_w + (\mathbf{m}_3 \times \mathbf{m}_1)_w\\ &\quad + (\mathbf{m}_1 \times \mathbf{m}_2)_w)\\ &= \frac{(\mathbf{m}_2 \times \mathbf{m}_3 + \mathbf{m}_3 \times \mathbf{m}_1 + \mathbf{m}_1 \times \mathbf{m}_2) \cdot \begin{pmatrix}0\\0\\1\end{pmatrix}}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\\ &= \frac{((\mathbf{m}_2 - \mathbf{m}_1) \times (\mathbf{m}_3 - \mathbf{m}_1)) \cdot \begin{pmatrix}0\\0\\1\end{pmatrix}}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\\ &= \frac{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]} = 1\ .\end{aligned}$$

## Appendix C

When **1, 2, 3, 4** are the vertexes of a rectangle, we set up the x-y world coordinate system such that $\mathbf{1} = (0, 0, 1)^T, \mathbf{2} = (a, 0, 1)^T, \mathbf{3} = (0, b, 1)^T, \mathbf{4} = (a, b, 1)^T$, where $a, b$ are the two sides of the rectangle. Furthermore if the scene plane is parallel to the image plane, we are to prove (12) and (13).

In Appendix B, we have proved that $w_2 = 1, w_3 = 1$. Thus by the coordinates of **1, 2, 3, 4**, there are (See the last paragraph of Section 4.1):

$$\begin{aligned}\bar{\mathbf{v}}_1 &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left(\frac{(\mathbf{3}_x - \mathbf{2}_x)}{w_2 w_3}\bar{\mathbf{m}}_1 + \frac{(\mathbf{1}_x - \mathbf{3}_x)}{w_3}\bar{\mathbf{m}}_2\right.\\ &\quad \left. + \frac{(\mathbf{2}_x - \mathbf{1}_x)}{w_2}\bar{\mathbf{m}}_3\right) = \frac{a(\bar{\mathbf{m}}_3 - \bar{\mathbf{m}}_1)}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]},\\ \bar{\mathbf{v}}_2 &= \frac{1}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\left(\frac{(\mathbf{3}_y - \mathbf{2}_y)}{w_2 w_3}\bar{\mathbf{m}}_1 + \frac{(\mathbf{1}_y - \mathbf{3}_y)}{w_3}\bar{\mathbf{m}}_2\right.\\ &\quad \left. + \frac{(\mathbf{2}_y - \mathbf{1}_y)}{w_2}\bar{\mathbf{m}}_3\right) = \frac{b(\bar{\mathbf{m}}_1 - \bar{\mathbf{m}}_2)}{[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]}\ .\end{aligned}$$

So,

$$\begin{aligned}|\hat{x}_5 - x_5| &= |[\bar{\mathbf{v}}_1, \Delta\bar{\mathbf{m}}_5]\\ &= \frac{|a|}{|[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]|}|[\bar{\mathbf{m}}_3 - \bar{\mathbf{m}}_1, \Delta\bar{\mathbf{m}}_5]|, \quad \text{(c1)}\end{aligned}$$

$$\begin{aligned}|\hat{y}_5 - y_5| &= |[\bar{\mathbf{v}}_2, \Delta\bar{\mathbf{m}}_5]\\ &= \frac{|b|}{|[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3]|}|[\bar{\mathbf{m}}_1 - \bar{\mathbf{m}}_2, \Delta\bar{\mathbf{m}}_5]|. \quad \text{(c2)}\end{aligned}$$

Because the scene plane is parallel to the image plane, then $\mathbf{H}_0 = \mathbf{K}(\mathbf{r}_1\mathbf{r}_2\mathbf{t})$ has the form:

$$\begin{aligned}\mathbf{H}_0 &= \mathbf{K}(\mathbf{r}_1\mathbf{r}_2\mathbf{t})\\ &= \begin{pmatrix}f & s & u_0\\ 0 & \alpha f & v_0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\cos(\theta) & \sin(\theta) & t_1\\ -\sin(\theta) & \cos(\theta) & t_2\\ 0 & 0 & t_3\end{pmatrix}\end{aligned}$$

where $\theta$ is the rotation angle of the camera, $\mathbf{t} = (t_1\, t_2\, t_3)^T$. Substitute $\mathbf{H}_0$ and the coordinates of **1, 2, 3, 4** into (3) and expand each of the equations, we find: $s_i = t_3, i = 1..4$ and

$$\begin{aligned}\mathbf{m}_3 - \mathbf{m}_1 &= \frac{1}{t_3}\mathbf{H}(\mathbf{3} - \mathbf{1}) = \frac{b}{t_3}\begin{pmatrix}f\sin(\theta) + s\cos(\theta)\\ \alpha f\cos(\theta)\\ 0\end{pmatrix},\\ \mathbf{m}_1 - \mathbf{m}_2 &= \frac{1}{t_3}\mathbf{H}(\mathbf{1} - \mathbf{2}) = \frac{-a}{t_3}\begin{pmatrix}f\cos(\theta) - s\sin(\theta)\\ -\alpha f\sin(\theta)\\ 0\end{pmatrix}.\end{aligned} \quad \text{(c3)}$$

Also, there is:

$$[\mathbf{m}_1\mathbf{m}_2\mathbf{m}_3] = \frac{\det(\mathbf{H})}{t_3^3}[\mathbf{123}] = \frac{ab\alpha f^2}{t_3^2}\ . \quad \text{(c4)}$$

By substituting (c3), (c4) into (c1) and (c2), we obtain (12) and (13).

## Acknowledgments

We would like to thank the anonymous reviewers for their helpful suggestions. This work was supported by the National Key Basic Research and Development Program (973) under grant No. 2002CB312104 and National Natural Science Foundation of China under grant No. 60475009.

## References

1. Y.I. Abdel-Aziz and H.M. Karara, "Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry," in *Proc. ASP/UI Symp. on Close-Range Photogrammetry*, Urbana, Illinois, Jan. 1971, pp. 1–18.
2. M.A. Abidi and T. Chandra, "A new efficient and direct solution for pose estimation using quadrangular targets: Algorithm and evaluation," *IEEE Trans, on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 5, pp. 534–538, 1995.

3. E. Bayro-Corrochano and V. Banarer, "A geometric approach for the theory and applications of 3D projective invariants," *Journal of Mathematical Imaging and Vision*, Vol. 16, No. 2, pp. 131–154, 2002.
4. E. Bayro-Corrochano, J. Lasenby, and G. Sommer, "Geometric algebra: A framework for computing point and line correspondences and projective structure using n uncalibrated cameras," in *the 13th Int. Conf. on Pattern Recognition*, Vienna, Austria, Aug. 1996, pp. 334–338.
5. S. Carlsson, "Symmetry in perspective" in *ECCV*, Freiburg, Germany, June 1998, pp. 249–263.
6. A. Criminisi, "Single-view metrology: Algorithms and applications," in *DAGM-Symposium*, Zurich, Sept. 2002, pp. 224–239.
7. A. Criminisi, *Accurate Visual Metrology from Single and Multiply Images*. Distingiushed Dissertation Series. Springer-Verlag London Ltd., Sept. 2001.
8. A. Criminisi, I. Reid, and A. Zisserman, "Single view metrology," in *ICCV*, Corfu, Greece, Sept. 1999, pp. 434–442.
9. A. Criminisi, I. Reid, and A. Zisserman, "A plane measuring device," *Image and Vision Computing*, Vol. 17, No. 8, pp. 625–634, 1999.
10. G. Csurka and O. Faugeras, "Algebraic and geometric tools to compute projective and permutation invariants," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 1, pp. 58–65, 1999.
11. R.M. Haralick, "Determining camera parameters from the perspective projection of a rectangle," *Pattern Recognition*, Vol. 22, No. 3, pp. 225–230, 1989.
12. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.
13. FA van den Heuvel, "Exterior orientation using coplanar parallel lines," in *the 10th Scandinavian Conference on Image Analysis*, Lappeenranta, Finland, June 1997, pp. 71–78.
14. J. Lasenby, E. Bayro-Corrochano, A. Lasenby, and G. Sommer, "A new methodology for computing invariants in computer vision," in *the 13th Int. Conf. on Pattern Recognition*, Vienna, Austria, Aug. 1996, pp. 393–397.
15. R. Lee, P.C. Lu, and W.H. Tsai, "Robot location using single views of rectangular shapes," *PE&RS*, Vol. 56, No. 2, pp. 231–238, 1990.
16. S. Maybank, "Relation between 3D and 2D invariants," *Image and Vision Computing*, Vol. 16, pp. 13–20, 1998.
17. L. Mundy and A. Zisserman (Eds.), *Geometric Invariance in Computer Vision*, The MIT Press: Cambridge, Mass., 1992.
18. I. Reid and A. Zisserman, "Goal-directed video metrology," in *ECCV*, Cambridge, UK, April 1996, pp. 647–658.
19. P. Sturm, "Algorithms for plane-based pose estimation," in *CVPR*, Hilton Head Island, South Carolina, June 2000, pp. 706–711.
20. B. Sturmfels, *Algorithms in Invariant Theory*, Springer-Verlag: Wien New York, 1993.
21. N.L. White (Ed.), "Invariant methods in discrete and computational geometry," in *Proc. Curacao Conference*, Kluwer Academic Publishers, June 1994.
22. M. Wilczkowiak, E. Boyer, and P. Sturm, "3D modeling using geometric constraints: A parallelepiped based approach," in *ECCV*, Copenhagen, Denmark, May 2002, pp. 221–236.
23. M. Wilczkowiak, E. Boyer, and P. Sturm, "Camera calibration and 3D reconstruction from single images using parallelepipeds," in *ICCV*, Vancouver, Canada, July 2001, pp. 142–148.
24. Y.H. Wu, *Bracket Algebra, Affine Bracket Algebra and Automated Geometric Theorem Proving*. Ph.D. Dissertation, Institute of Systems Science, Chinese Academy of Sciences, Beijing, June 2001.
25. Y.H. Wu and Z.Y. Hu, "Invariant representations of a quadric cone and a twisted cubic," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 10, pp. 1329–1332, 2003.
26. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 11, pp. 1330–1334, 2000.

**Yihong Wu** received her Doctor of Science degree in Geometric Invariants and Applications from MMRC, Institute of Systems Science, Chinese Academy of Sciences, in 2001. From June 2001 to July 2003, she did her postdoctoral research in NLPR, Institute of Automation, Chinese Academy of Sciences. After then, she joined NLPR as an associate professor. Her research interests include polynomial elimination and applications, geometric invariant and applications, automated geometric theorem proving, camera calibration, camera pose determination, and 3D reconstruction etc.

**Zhanyi Hu** received the B.S. Degree in Automation from the North China University of Technology in 1985, the Ph.D. Degree (Docteur d'Etat) in Computer Science from the University of Liege, Belgium, in Jan. 1993. Since 1993, he has been with the Institute of Automation, Chinese Academy of Sciences. From May 1997 to May 1998, he also acted as a visiting scholar of Chinese University of Hong Kong on invitation. Dr. Hu now is a Research Professor of Computer Vision, a member of the Executive Expert Committee of the Chinese National High Technology R&D Program, a deputy editor-in-chief for Chinese Journal of CAD and CG, and an associate editor for Journal of Computer Science and Technology. His current research interests include Camera Calibration, 3D Reconstruction, Feature Extraction, and Vision Guided Robot Navigation etc.