第19卷第1期
2000年2月

红外与毫米波学报
J. Infrared Millim. Waves

Vol. 19, No. 1
February, 2000

A GENETIC ALGORITHM APPROACH TO ACCURATE CALIBRATION OF CAMERA *

HUANG Hai-Yun QI Fei-Hu

(Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200030, China)

Abstract A new two-stage calibration method based on the close form solution and genetic algorithm was presented. In the first stage, the distortion parameters were ignored. The internal and external parameters were estimated by least square algorithm. In the second stage, the parameters estimated in the first stage were used as initial values. Taking account of the camera distortions, all parameters were optimized by genetic algorithm to get exact solutions. Because the algorithm optimized the rotation angles directly, the orthonormal constraints were easily satisfied in the present method. Moreover, the approach simplified the calibration process.

Key words camera calibration, genetic algorithm, computer vision.

一种精确标定摄像机的遗传算法方案 *

黄海赟 戚飞虎

(上海交通大学计算机科学与工程系,上海,200030)

摘要 摄像机标定的一个重要问题是既要保证参数估计的精确性,又要满足旋转矩阵正交约束条件.本文在线性优化和遗传算法的基础上提出了一种新的双段标定方案.第一阶段,在忽略畸变参数的情况下,利用最小二乘法估计摄像机内外参数.为减少畸变影响,本阶段仅考虑图像中心区域的标定点;第二阶段,以第一阶段的估计值为初始值,并考虑摄像机畸变的影响,根据整个图像范围的标定点,利用遗传算法优化摄像机的所有参数以获得精确解.由于该算法直接优化摄像机的旋转角度,所以本文的标定方案能够容易地满足旋转矩阵正交约束条件.此外,采用遗传算法也简化了摄像机标定过程.

关键词 摄像机标定,遗传算法,计算机视觉.

Introduction

Camera calibration is an important issue in the field of machine vision. In recent years, the techniques of computer vision are widely used in many applications, and the need for accurate measurements is increasing. A well-calibrated camera can present not only accurate images for geometrical measurements but also a good foundation for the matching and reconstruction of stereo scenes[1].

The existing techniques for camera calibration can be classified into three categories: linear optimization approach[2], nonlinear optimization approach and two-stage approach[3~6]. The most widely used technique is the two-stage approach. It divides the calibration parameters into several groups, e. g. two groups. In the first stage, all external and major internal parameters are estimated by solving linear equations based on a distortion-free camera model. In the second stage, the remained[3-7] or all[6] parameters are obtained by nonlinera optimization, which takes account of distortions of camera. Compared with the linear optimization method, the two-stage approach can solve many distortion parameters of camera, and

* The project supported by the National Natural Science Foundation of China
Received 1999-07-01, revised 1999-09-17

* 国家自然科学基金(编号 69772002)资助项目
稿件收到日期 1999-07-01,修改稿收到日期 1999-09-17

its computational work is less than that of the nonlinear optimization method.

The afore-mentioned approaches have a common disadvantage, i. e. they can not guarantee the orthonormal constrains of rotation matrix. This paper presents a new calibration method on the basis of least square algorithm and genetic algorithm. It provides not only accurate soluitions but also orthonormal rotation matrix. Moreover, it simplifies the calibration process.

1 Camera Models and Parameters

Figure 1 illustrates a pinhole camera model and four coordinate systems that are frequently referred in computer vision.

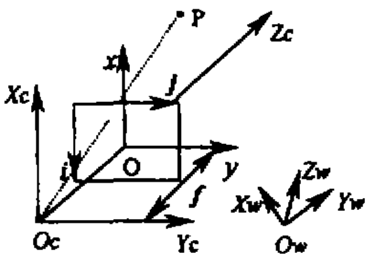

Fig. 1 A pinhole camera model
图 1 小孔摄像机模型

1. A world coordinate system Xw-Yw-Zw: a referenced 3D orthogonal coordinate system;

2. A camera coordinate system Xc-Yc-Zc: its origin coincides with the optical center of the camera, and the Zc axis coincides with the optical axis of the camera;

3. An image plane coordinate system x-y: a 2D orthogonal coordinate system. Its origin is the intersection of the optical axis with the image plane, and the x and y axes are parallel to the Xc and Yc axes, respectively.

4. An image coordinate system i-j: Its origin locates on the upper left corner of the image plane. In the case of digital images, i is the row number, and j is the column number.

Considering a pinhole camera model, which does not involve distortions, the transformation to convert the world coordinates to image coordinates is expressed by:

$$\frac{z_c}{f}\begin{bmatrix} x \\ y \\ f \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \mathbf{R}\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \mathbf{T} \tag{1}$$

where $\mathbf{R}$ denotes the rotation matrix, $\mathbf{T}$ denotes the translation vector, and f represents the effective focal length. Formula (1) can be converted to a composite matrix form:

$$\begin{bmatrix} \frac{z_c}{f}x \\ \frac{z_c}{f}y \\ z_c \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}\begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{2}$$

Because of the imperfection of manufacture, the image plane coordinates of a point are often subject to camera distortions that can not be neglected in the case of accurate measurements. This paper considers three types of distortions: radial, decentering and thin prism distortions[3,6]. Let (x', y') be the estimated values of (x, y), the image plane coordinates are corrected by:

$$\begin{cases} x = x' + \delta'_x \\ \quad = x' + k_1 x'(x'^2 + y'^2) + \\ \quad\; p_1(3x'^2 + y'^2) + 2p_2 x' y' + \\ \quad\; s_1(x'^2 + y'^2) \\ y = y' + \delta'_y \\ \quad = y' + k_1 y'(x'^2 + y'^2) + \\ \quad\; 2p_1 x' y' + p_2(x'^2 + 3y'^2) + \\ \quad\; s_2(x'^2 + y'^2) \end{cases} \tag{3}$$

where δ'_x and δ'_y denote the distortion errores with respect to (x', y'). In practice, a digital image only presents image coordinates, (i, j), for a point. If (i_0, j_0) denote the image coordinates that coincide with the origin of image plane coordinate system, then (x', y') can be expressed as:

$$x' = (i - j_0)\omega_i \quad y' = (i - j_0)\omega_j \tag{4}$$

where ω_i and ω_j denote the coefficients that convert image coordinates to image plane coordinates, and $\omega_i < 0$, $\omega_j > 0$. From the manufacture specification, we can get the row and column distances, d_i and d_j, between adjacent photoelectric elements in CCD array. Translating signals from CCD to computer is a D-A-D process that is harmonized by line-synchronizing signals. Generally, ω_i and ω_j can be approximately expressed as Eq. (5). However, due to the frequency errores and small tilt of CCD array, etc., ω_i and ω_j should be re-estimated in practice.

$$\omega_i = -d_i \quad \omega_j = \frac{\textit{Num. of the CCD elements in column direction}}{\textit{Num. of the image pixels in column direction}} d_j, \quad (5)$$

Now, we can state the calibration problem as follows: Given a sufficient number of control points P_m with their world coordinates (x_{wm}, y_{wm}, z_{wm}) and corresponding image coordinates (i_m, j_m), evaluate camera's internal parameters $(i_0, j_0, \omega_i, \omega_j, f)$, external parameters $(\mathbf{R}, \mathbf{T})$ and distortion parameters $(k_1, p_1, p_2, s_1, s_2)$.

This paper adopts a two-stage approach to calibrate all parameters. First, use a distortion-free model e.g. pinhole camera model and least square algorithm to estimate all external and part of internal parameters; Second, taking account of camera distortions, a genetic algorithm is performed to optimize all parameters, and the solutions in the first stage are set as initial values.

2 Initial Estimation Based on Least Square Algorithm

If camera distortions are not considered, image plane coordinates (x, y) can be obtained directly from (4). Substituting (4) into (2), we can get the relationship between the world and image coordinates.

$$\begin{bmatrix} iz_c \\ jz_c \\ z_c \end{bmatrix} = \begin{bmatrix} f_i r_{11} + i_0 r_{31} & f_i r_{12} + i_0 r_{32} & f_i r_{13} + i_0 r_{33} & f_i t_1 + i_0 t_3 \\ f_j r_{21} + j_0 r_{31} & f_j r_{22} + j_0 r_{32} & f_j r_{23} + j_0 r_{33} & f_j t_2 + j_0 t_3 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} \mathbf{x}_w \\ \mathbf{y}_w \\ \mathbf{z}_w \\ 1 \end{bmatrix} \quad (6)$$

where $f_i = f/\omega_i < 0$, $f_j = f/\omega_j > 0$. Without loss of generalization, let $t_3 > 0$. In other words, We put the world coordinate system in front of camera, then divide the first and the second lines of Eq. (6) by t_3. Thus, we can derive (7) from (6).

$$\begin{cases} i = x_w(f_i r_{11} + i_0 r_{31})/t_3 + \\ \quad y_w(f_i r_{12} + i_0 r_{32})/t_3 + \\ \quad z_w(f_i r_{13} + i_0 r_{33})/t_3 - \\ \quad r_{31} i x_w/t_3 - r_{32} i y_w/t_3 - \\ \quad r_{33} i z_w/t_3 + (f_i t_1 + i_0 t_3)/t_3 \\ j = x_w(f_j r_{21} + j_0 r_{31})/t_3 + \\ \quad y_w(f_j r_{22} + j_0 r_{32})/t_3 + \\ \quad z_w(f_j r_{23} + j_0 r_{33})/t_3 - \\ \quad r_{31} j x_w/t_3 - r_{32} j y_w/t_3 - \\ \quad r_{33} j z_w/t_3 + (f_j t_1 + j_0 t_3)/t_3 \end{cases} \quad (7)$$

Let

$$\begin{aligned} \mathbf{A}_1 &= [(f_i r_{11} + i_0 r_{31})/t_3 \quad (f_i r_{12} + i_0 r_{32})/t_3 \quad (f_i r_{13} + i_0 r_{33})/t_3]^T \\ &= (f_i \mathbf{r}_1 + i_0 \mathbf{r}_3)/t_3 \\ \mathbf{A}_2 &= [(f_j r_{21} + j_0 r_{31})/t_3 \quad (f_j r_{22} + j_0 r_{32})/t_3 \quad (f_j r_{23} + j_0 r_{33})/t_3]^T \\ &= (f_j \mathbf{r}_2 + j_0 \mathbf{r}_3)/t_3 \\ \mathbf{A}_3 &= [r_{31}/t_3 \quad r_{32}/t_3 \quad r_{33}/t_3]^T = \mathbf{r}_3/t_3 \\ \mathbf{A}_4 &= [(f_i t_1 + i_0 t_3)/t_3 \quad (f_j t_2 + j_0 t_3)/t_3]^T \\ \mathbf{A} &= [\mathbf{A}_1^T \quad \mathbf{A}_2^T \quad \mathbf{A}_3^T \quad \mathbf{A}_4^T]^T \end{aligned} \quad (8)$$

where $\mathbf{r}_1 = [r_{11} \quad r_{12} \quad r_{13}]^T$,
$\mathbf{r}_2 = [r_{21} \quad r_{22} \quad r_{23}]^T$,
$\mathbf{r}_3 = [r_{31} \quad r_{32} \quad r_{33}]^T$,
$\mathbf{R} = [\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{r}_3]^T$.

If the number of control points is n, a composite matrix form of (7) and (8) can be expressed as:

$$\mathbf{BA} = \mathbf{C} \quad (9)$$

where $\mathbf{C} = [i_1 \; j_1 \ldots \; i_n \; j_n]^T$

$$\mathbf{B} = \begin{bmatrix} x_{w1} & y_{w1} & z_{w1} & 0 & 0 & 0 & -i_1 x_{w1} & -i_1 y_{w1} & -i_1 z_{w1} & 1 & 0 \\ 0 & 0 & 0 & x_{w1} & y_{w1} & z_{w1} & -j_1 x_{w1} & -j_1 y_{w1} & -j_1 z_{w1} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{wn} & y_{wn} & z_{wn} & 0 & 0 & 0 & -i_n x_{wn} & -i_n y_{wn} & -i_n z_{wn} & 1 & 0 \\ 0 & 0 & 0 & x_{wn} & y_{wn} & z_{wn} & -j_n x_{wn} & -j_n y_{wn} & -j_n z_{wn} & 0 & 1 \end{bmatrix}$$

The unknown variable $\mathbf{A}$ is an 11-dimensional vector in (9). In order to get good estimation, the number of control points should not be less than 6. Thus, a least square solution to (9) is (10).

$$\mathbf{A} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{C} \tag{10}$$

Because $\mathbf{R}$ is an orthonormal matrix, we can derive most of the parameters from $\mathbf{A}$:

$$\begin{aligned}
t_3 &= \frac{1}{\|\mathbf{A}_3\|},\\
\mathbf{r}_3 &= t_3\mathbf{A}_3,\\
i_0 &= t_3^2\mathbf{A}_1^T\mathbf{A}_3,\\
j_0 &= t_3^2\mathbf{A}_2^T\mathbf{A}_3,\\
f_i &= -t_3\|\mathbf{A}_1 - i_0\mathbf{A}_3\|,\\
f_j &= t_3\|\mathbf{A}_2 - j_0\mathbf{A}_3\|,\\
\mathbf{r}_1 &= (\mathbf{A}_1 - i_0\mathbf{A}_3)t_3/f_i,\\
\mathbf{r}_2 &= (\mathbf{A}_2 - j_0\mathbf{A}_3)t_3/f_j,\\
\begin{bmatrix} t_1 \\ t_2 \end{bmatrix} &= \begin{bmatrix} \frac{t_3}{f_i} & 0 \\ 0 & \frac{t_3}{f_j} \end{bmatrix}\left(\mathbf{A}_4 - \begin{bmatrix} i_0 \\ j_0 \end{bmatrix}\right)
\end{aligned} \tag{11}$$

So far, most of the internal and external parameters are estimated by (11). But f, ω_i and ω_j are still unknown. Generally, ω_i and ω_j can be evaluated by (5), then f can be worked out further. Reference [7] also introduced some reasonable mehtods to calculate ω_i and ω_j. As mentined before, we ignored camera distortions in this section. In order to obtain good solutions, we used central points, which locate around image center and involve few distortions in this stage, and added more outer points in the next stage[6].

3 A Genetic Algorithm to Calibration

The existing calibration methods can not guarantee the strict orthonormal constrains of rotation matrix $\mathbf{R}$. These approaches either orthonormalize $\mathbf{R}$ at the end of calibration[6,8], or impose parts of constrains in the main algorithm[1,5]. However, the computing complexity is increased, and the veracity of solutions is reduced. In recent years, the GA (Genetic Algorithm) is applied as an effective optimization method to complex problem, and it is independent of the problem itself. Therefore, we use a GA in the second stage to hold the orthonormal constrains.

Let θ_x, θ_y and θ_z denote the rotation angles around the X_w, Y_w and Z_w axes, respectively, then $\mathbf{R}$ can be expressed as:

$$\mathbf{R} = \begin{bmatrix} \cos\theta_y\cos\theta_z & \cos\theta_x\sin\theta_z + \sin\theta_x\sin\theta_y\cos\theta_z & \sin\theta_x\sin\theta_z - \cos\theta_x\sin\theta_y\cos\theta_z \\ -\cos\theta_y\sin\theta_z & \cos\theta_x\cos\theta_z - \sin\theta_x\sin\theta_y\sin\theta_z & \sin\theta_x\cos\theta_z + \cos\theta_x\sin\theta_y\sin\theta_z \\ \sin\theta_y & -\sin\theta_x\cos\theta_y & \cos\theta_x\cos\theta_y \end{bmatrix} \tag{12}$$

We put θ_x, θ_y and θ_z directly into the chromosome vector of GA to hold the strict orthonormal constrains. The initial values of θ_x, θ_y and θ_z are given by:

$$\begin{aligned}
\theta_x &= \arctan\left(-\frac{r_{32}}{r_{33}}\right),\\
\theta_y &= \arcsin(r_{31}),\\
\theta_z &= \arctan\left(-\frac{r_{21}}{r_{11}}\right)
\end{aligned} \tag{13}$$

The chromosome vector b is expressed as (14), and it is easy to add new parameters (genes) to b.

$$b = \{\theta_x \quad \theta_y \quad \theta_z \quad t_1 \quad t_2 \quad t_3 \quad f \quad i_0 \quad j_0 \quad \omega_i \quad \omega_j \quad k_1 \quad p_1 \quad p_2 \quad s_1 \quad s_2\} \tag{14}$$

As mentioned before, the starting values of b are generated randomly on the basis of the solutions in the first stage. The initial values of distortion parameters are determined by experience, and generally, the magnitude is $10^{-4}\sim10^{-1}$. Furthermore, we use a parallel algorithm to expand the searching area. The cost function $E()$ of GA is defined by:

$$E(b_r) = \sum_{m=1}^{C}[(x'_m(b_r) - x_m(b_r))^2 + (y'_m(b_r) - y_m(b_r))^2]^{1/2} \tag{15}$$

where $(x'_m(b_r), y'_m(b_r))$ are the estimated image coordinates of control point P_m with respect to chromosome b_r, and they are obtained by (4) on the basis of (i_m, j_m). Then, $(x_m(b_r), y_m(b_r))$ denote the corresponding expectation, which are worked out by (2) on the basis of the world coordinates (x_{wm}, y_{wm}, z_{wm}). Our GA approach is described as follows.

1. Generate a collection of N groups, and each group includes M units. The chromosome of each unit is determined randomly on the basis of initial estimation or in a wide range. Let S denote the collection, G denote a group, and a superscript denote the generation number t, and subscripts denote the group and the unit numbers. Set $t=0$, we obtain the initial generation units.

$$G_i^0 \in S; G_i^0 = \{b_{i1}^0 \ldots b_{ij}^0 \ldots b_{iM}^0\};$$
$$i = 1, 2, \ldots N \tag{16}$$

2. Work out the cost function of each unit, $E(b_{ij}^t)$, and sort them in ascent order.

$$G_i^t = \{b_{i1}^t \ldots b_{ij}^t \ldots b_{iM}^t\} \ and$$
$$E(b_{ij}^t) \leqslant E(b_{i(j+1)}^t) \tag{17}$$

3. Perform the replication operator to each group. A few best and randomly selected units are transferred from G_i^t to the next generation. The number of replications is k.

$$G_i^{t+1} = \{b_{i1}^t \ldots b_{ik}^t \ldots \ldots\}$$
$$= b_{i1}^{t+1} \ldots b_{ik}^{t+1} \ldots \ldots\} \tag{18}$$

4. Perform the creation operator to each group. Use random operator $\Psi()$ to create p new units.

$$G_i^{t+1} = \{b_{i1}^{t+1} \ldots b_{ik}^{t+1} \ldots b_{i(k+p)}^{t+1} \ldots\} \tag{19}$$

5. Perform the mutation operator to each group. Select q units from the afore-mentioned $(k+p)$ units, then change parts of their genes randomly.

$$G_i^{t+1} = \{b_{i1}^{t+1} \ldots b_{ik}^{t+1} \ldots$$
$$b_{i(k+p)}^{t+1} \ldots b_{i(k+p+q)}^{t+1} \ldots\} \tag{20}$$

6. Perform the crossover operator to each group. If t is a multiple of a constant integer, randomly select two different units among all groups. Otherwise, select them in the same group. Then randomly choose a section of chromosome, and swap the section between these two different units. At the same time, few genes are changed randomly. Repeat the crossover operator $[M-(k+p+q)]$ times.

$$G_i^{t+1} = \{b_{i1}^{t+1} \ldots b_{ik}^{t+1} \ldots$$
$$b_{i(k+p)}^{t+1} \ldots b_{i(k+p+q)}^{t+1} \ldots b_{iM}^{t+1}\} \tag{21}$$

7. Increase the generation t by one, and select the best unit among all groups as current solution.

$$t = t + 1$$
$$b_{best} = \{b_{ij}^t | E(b_{ij}^t)\}$$
$$= \mathop{\mathrm{MIN}}_{i=1}^{N}(\mathop{\mathrm{MIN}}_{j=1}^{M}(E(b_{ij}^t)))\} \tag{22}$$

8. Examine the termination criteria. If $t > max$ or $E(b_{best}) < \varepsilon$, then halt the procedure, and output b_{best}. Otherwise loop to the second step.

4 Experiments and Results

We define NCE (Normalized Calibration Error), which is similar to [6], as an evaluative criterion.

$$NCE = \frac{1}{n}\sum_{m}^{n}\left[\frac{(x'_{cm} - x_{cm})^2 + (y'_{cm} - y_{cm})^2}{z_{cm}^2(f_r^{-2} + f_c^{-2})/12}\right]^{1/2} \tag{23}$$

where (x_{cm}, y_{cm}, z_{cm}) denote the expectation of camera coordinates, which are calculated on the basis of (x_{wm}, y_{wm}, z_{wm}) by (1). Then (x'_{cm}, y'_{cm}) denote the corresponding estimated values, which are worked out on the basis of (i_m, j_m) by (4), (3) and (1). If $NCE \approx 1$, the performance of the calibration approach is good. On the contrary, if $NCE \geqslant 1$, the performance is bad. In the experiments, we used an interactive program to pick points. The hardware platform is P Ⅱ 266+64M RAM and the programming language is C++. The results are given in Table 1.

In Table 1, NCE becomes a little bigger while the number of control points is increased. This is because more outer points are used in the latter experiment, and these points introduce more distortions and errors to the calibration. Since the genetic algorithm is a kind of long-term computing work, our approach is suitable for off-line calibration. However, these problems do not influence good performance of the calibration method. Obviously, it holds the strict orthonormal constrains of rotation matrix, and it is universal and effective to different camera models.

Table 1 Calibration results and performance

表 1 标定结果和性能

Points num. central/outer	10 6/4	20 10/10	30 10/20	Points num. central/outer	10 6/4	20 10/10	30 10/20
θ_x	3.0243°	2.9774°	2.9824°	p_1	0.0006	0.0021	0.0030
θ_y	−1.8593°	−1.9276°	−1.9337°	p_2	0.0005	0.0039	0.0042
θ_z	−5.3386°	−5.4015°	−5.4263°	s_1	0.0031	0.0065	0.0070
t_1(mm)	36.7	35.4	34.9	s_2	0.0028	0.0048	0.0052
t_2(mm)	48.9	50.6	52.1	NCE	1.0847	1.1003	1.1065
t_3(mm)	1745.3	1750.4	1747.0	Time(s)	21	67	93
x_0(pixel)	117	120	118			Control parameters of GA	
y_0(pixel)	175	172	172	N		10	
ω_x(μm/pixel)	−14.995	−14.950	−14.924	M		50	
ω_y(μm/pixel)	13.583	13.547	13.539	k		8	
f(mm)	27.7	28.3	28.1	p		8	
k_1	0.0833	0.1216	0.1290	q		6	

REFERENCES

[1] Dan Zetu, Banerjee P, Data P S. Input model for virtual reality-aided facility layout. *I E Transaction*, 1998, **30** (7):597—620

[2] Grosky William I, Tamburino L A. A unified approach to the linear camera calibration problem. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1990, **12** (7):663—671

[3] Chatterjee Chanchal, Roy V P, CHONG E K P, *et al*. A nonlinear Gauss-Seidel algorithm for noncoplanar and coplanar camera calibration with convergence analysis. *Computer Vision and Image Understanding*, 1997, **67** (1): 58—80

[4] ZHOU Guo-Qing UZI Ethrog, FENG Wen-Hao, *et al*. CCD camera calibration based on natural landmarks. *Pattern Recognition*, 1998, **31**(11):1715—1724

[5] Bacakoglu Hakan, Kamel M S. A three-step camera calibration method. *IEEE Trans. on Instrumentation and Measurement*, 1997, **46**(5):1165—1172

[6] WENG Ju-Yang, Cohen P, Herniou M. Camera calibration with distortion models and accuracy evaluation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1992, **14**(10):965—980

[7] Lenz Reimar K, Tsai R. Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1988, **10**(5),:713—720

[8] WENG Ju-Yang, HUANG Thomas, Ahuja N. Motion and structure from two perspective views: algorithms, error, analysis, and error estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1989, **11**(5): 451—464