International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.7, No.4 (2014), pp.1-12
http://dx.doi.org/10.14257/ijsip.2014.7.4.01

Self-calibration of Camera with Rotary Motion Based on SIFT Feature Matching

Canlin Li, Fubao Zhu and Ni Yao

School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450000, China
lcl_zju@aliyun.com

Abstract

In the field of computer vision and photogrammetry, it is constantly necessary to online or real-time acquire the camera parameters through self-calibration. This paper presents a method about self-calibration of camera with rotary motion based on SIFT feature matching. The proposed approach first shoots more than three images of the same scene by rotating the camera and keeping its position and internal parameters unchanging. After SIFT feature extraction and sequentially cycled matching for all images, the optimal reference image and effective images are determined by virtue of the algorithm on pose estimation. According to the coordinates of matched SIFT features, all 2D projection transformation matrices which transforms the reference into other effective images are calculated. With these matrices, relevant linear equations are established and the internal matrix of camera is solved. The proposed method can be online applied to quickly, accurately and stably obtain internal parameters of camera. Real data has been used to test the proposed approach, and very good results have been achieved.

Keywords: *camera calibration, SIFT, self-calibration, feature matching*

1. Introduction

The process of acquiring the parameters of camera geometric model is called camera calibration [1]. It is an essential step to extract three-dimensional space information from two-dimensional images in the applications of image processing and computer vision, and is widely used in three-dimensional reconstruction, navigation, visual surveillance and other fields. Under certain camera model, camera calibration strikes camera parameters of the model with a series of mathematical transformations and calculations by virtue of the image processing. At present, there have been a lot of camera calibration methods, which contains traditional calibration and self-calibration.

Traditional calibration is relatively mature and stable but based on specific experimental conditions, such as calibration references with known shape or size [2-5]. The traditional methods of using calibration references obtain a wide range of applications, which typically involve Tsai two-step method [2-3]. These traditional methods require using the calibration references in the process of shooting calibration images, which would bring great inconvenience into the camera calibration, since the use and position adjustment of calibration reference may result in the online task interrupted.

Camera self-calibration does not require the use of calibration reference, and estimates the camera internal parameters based solely on the corresponding relationship between feature pixels of images, which makes online and real time calibrating camera parameters possible.

ISSN: 2005-4254 IJSIP

However, many existing self-calibration methods still cannot get stable results and be mature enough for practical application [6-10], and therefore need some ways to improve the existing self-calibration techniques.

This paper presents a method about self-calibration of camera with rotary motion based on SIFT feature matching. The proposed approach requires fixing camera position, that is, camera not being displaced, but being rotated with unknown rotation angle. With the internal nature of the captured image sequence under the special shooting conditions, the complex problem solving nonlinear self-calibration is simplified, thereby solving the internal reference of camera based on SIFT feature extraction and matching. The proposed method provides a fast self-calibration solution with stable and accurate results. This makes online and real-time calibrating the internal parameters of camera possible in the case of no dependent calibration references, thus improving the existing camera self-calibration.

2. The Proposed Approach on Camera Self-Calibration

For the proposed approach, we make the camera fixed, and rotate it to shoot multiple images of the same scene under the condition of constant internal parameters. Then extract and match SIFT features of shooting images, and take advantage of the SIFT feature correspondence between the images to solve the self-calibration problem. The proposed approach is described as follows in details.

2.1. Shooting the Images of Self-Calibration

Camera is fixed at the same position in the 3D world space, and is rotated to different directions to capture images of the same scene. In the course of capturing image sequences, we remain internal reference matrix K unchanged. We assume that L images (L≥3) are captured as the input images of the proposed method.

K is an upper triangular matrix representing camera internal parameters.

$$K = \begin{bmatrix} k_u & s & p_u \\ 0 & k_v & p_v \\ 0 & 0 & 1 \end{bmatrix}, \tag{1}$$

The goal of this self-calibration method is to determine the internal reference matrix K of camera, where

k_u is the magnification in u direction (horizontal) of image, unit in pixels,

k_v is the magnification in v direction (vertical) of image, unit in pixels,

s is the distortion factor corresponding to the distortion of camera axis,

p_u and p_v are the principal point coordinates, unit in pixels.

The parameters k_u and k_v are closely linked to focal length of camera. If $k_u = k_v$ and $s = 0$, k_u and k_v are just focal length in pixels. $k_u = k_v$ means that the photosensitive array of camera contains square pixels. If $k_u \neq k_v$, k_u is the ratio of the focal length f and the pixel size in u direction, and k_v is the ratio of the focal length f and the pixel size in v direction. It means there are the following relationships.

$$f_u = k_u \cdot dpx \ , \tag{2}$$

$$f_v = k_v \cdot dpy \ , \tag{3}$$

where f_u and f_v are effective focal length, or distance from image plane to projective center, unit in millimeters, and *dpx* and *dpy* are the pixel size of image plane, respectively in u and v directions. $k_u \neq k_v$ means that the photosensitive array of camera contains non-square pixels, such as CCD camera situation.

2.2. SIFT Feature Extraction

SIFT is an image local feature based on scale space proposed by Lowe DG [11]. It remains invariant to rotation, scaling zoom, pan and even brightness changes, and also maintains a certain degree of stability to perspective changes, affine transformation, even noise. SIFT feature has good uniqueness and rich information, even though a handful of objects can also produce a lot of SIFT feature vectors [12]. Existing camera calibration methods are often based on some fixed geometric features such as corner points, lines, circles, etc in the images. Since these geometric features are assumed to exist in the scene, it confines the calibrating scene. As for the proposed self-calibration method, we extract SIFT features from the calibrating images, in order to ensure the extracted features invariant to rotation, scaling zoom, pan and even brightness changes. By the way, since it is not necessary to assume the existence of the fixed geometric features such as corner points, lines or circles in the scene, no restrictions or requirements are imposed on the shooting scenes.

2.3. SIFT Feature Matching

As for SIFT feature point matching between two images A and B, we adopt the Euclidean distance of two corresponding SIFT feature vectors as the similarity measure of two SIFT feature points. Here nearest neighbor ratio is applied in the specific determination of similarity. It is explained as follows. For a certain SIFT feature point a in image A, we find out the first nearest feature point b1 and the second nearest feature point b2 in the image B according to Euclidean distance. If the ratio of the first nearest Euclidean distance between a and b1 and the second nearest Euclidean distance between a and b2 is less than a certain threshold, for example 0.6, the pair of a and b1 is considered to be one pair of matched SIFT points. If this ratio threshold is reduced, the number of SIFT matched point pairs will be reduced, but the match is more stable.

For more images, after their SIFT feature points are extracted, the sequentially cycled matching is conducted among them, so as to determine the position of respective SIFT feature points in these images corresponding to same three-dimensional feature point. The sequentially cycled matching is described as follows.

Suppose there are L images I_0, I_1, ..., I_{L-1}, for each SIFT feature point p_i in I_0,we conduct the following round of SIFT feature matching.

1) Search SIFT feature points in the images $\mathrm{I}_1 \ldots \mathrm{I}_{L-1}$ which can match p_i in I_0. If the matched SIFT feature point corresponding to p_i can be found in every image of $\mathrm{I}_1 \ldots \mathrm{I}_{L-1}$, which are named as $\mathrm{p}_{0,1}$, $\mathrm{p}_{0,2}$, $\mathrm{p}_{0,3}$, …, $\mathrm{p}_{0,L-1}$, go to Step 2). If not, this round of SIFT feature matching is over.

2) Search SIFT feature points in the images $\mathrm{I}_2 \ldots \mathrm{I}_{L-1}$ which can match $\mathrm{p}_{0,1}$ in I_1. If the matched SIFT feature point corresponding to $\mathrm{p}_{0,1}$ can be found in every image of $\mathrm{I}_2 \ldots \mathrm{I}_{L-1}$, which are named as $\mathrm{p}_{1,2}$, $\mathrm{p}_{1,3}$, …, $\mathrm{p}_{1,L-1}$, go to Step 3). If not, this round of SIFT feature matching is over.

3) Search SIFT feature points in the images $\mathrm{I}_3 \ldots \mathrm{I}_{L-1}$ which can match $\mathrm{p}_{0,2}$ in I_2. If the matched SIFT feature point corresponding to $\mathrm{p}_{0,2}$ can be found in every image of $\mathrm{I}_3 \ldots \mathrm{I}_{L-1}$, which are named as $\mathrm{p}_{2,3}$, …, $\mathrm{p}_{2,L-1}$, go to Step 4). If not, this round of SIFT feature matching is over.

……

L-2) Search SIFT feature point in the image I_{L-1} which can match $p_{0,L-2}$ in I_{L-2}. If the matched SIFT feature point corresponding to $p_{0,L-2}$ can be found in the image I_{L-1}, which is named as $p_{L-2,L-1}$, go to step L-1). If not, this round of SIFT feature matching is over.

L-1) If $p_{0,2} = p_{1,2}$ in I_2 ,and $p_{0,3} = p_{1,3} = p_{2,3}$ in I_3, ……, and $p_{0,L-1} = p_{1,L-1} = \dots = p_{L-2,L-1}$ in I_{L-1}, this round of SIFT feature matching is successfully over, and p_i in I_0, $p_{0,1}$ in I_1, $p_{0,2}$ in I_2, …, $p_{0,L-1}$ in I_{L-1} are saved as matched SIFT feature points corresponding to same three-dimensional feature point.

The sequentially cycled matching can ensure correctly matching SIFT feature points to a maximize degree. After SIFT feature extraction and matching, we get matched SIFT feature points in multiple images corresponding to same three-dimensional feature point. Actually, for the subsequent processing step, it is enough to just save the pixel location of every SIFT feature point extracted from SIFT feature vector.

2.4. Determine the Optimal Reference Image and Effective Images

We hope that the proposed self-calibration approach adopt the calibration images with bigger rotation angle as far as possible, in order to achieve better calibration results. So we design the approach to provide automatically filtering of the input images in the pre-processing stage. For every input image, this pre-processing stage focuses on automatically calculating the relatively rotational orientation between this image and all other images which includes the rotation angles of three coordinate axes. Those images whose relative rotation angle in any direction is greater than a predetermined angle (for example 30 degrees) are considered to be the effective candidate images about this input image. The input image with the largest number of effective candidate images is determined as the optimum reference image, and the image and its effective candidate images are considered as the effective images of the proposed self-calibration approach. If the number of the effective images is less than 3, all input images are considered as the effective images, so that the self-calibration can be performed due to this performing condition where the self-calibration method requires at least three input images. We name the optimum reference image as EI_0, and all images in the following steps means the effective images, which are named as $EI_0, EI_1, \dots, EI_{N-1}$.

Since we select images which are more suitable for rotational motion self-calibration method based on the relative orientation among images in the pre-processing stage, the better calibrating results can be achieved. If there is not enough effective images, applying all images to the proposed self-calibration method will make the calibrating results not so bad, especially in the case of a large number of images.

The algorithm about calculating the relatively rotational orientation between two images [14] is revealed in Section 3.

2.5. Calculating 2D Projection Transformation

After the pre-processing stage, we select EI_0 as the reference image. For every image EI_j (j=1…N-1) of effective images except for EI_0, we compute 2D projection transformation P_j which transforms EI_0 into EI_j. Determining all N-1 P_j mainly involves two steps, where one is calculating N-1 P_j using a linear method, and two is iteratively refining N-1 P_j using Levenberg-Marquardt optimization method.

1) Calculating N-1 P_j using a linear method

Suppose that $u_i^0 = (u_i^0, v_i^0, 1)^T$ and $u_i^j = (u_i^j, v_i^j, 1)^T$ are two pixel positions about the ith pair of matched SIFT features between the image EI_0 and the image EI_j, $u_i^0 = (u_i^0, v_i^0, 1)^T$ is in EI_0, and

$\mathrm{u}_i^j = (u_i^j, v_i^j, 1)^{\mathrm{T}}$ is in EI_j, the following equation between the pair of matching pixels is always established.

$$w_i^j (u_i^j, v_i^j, 1)^{\mathrm{T}} = P_j (u_i^0, v_i^0, 1)^{\mathrm{T}} \tag{4}$$

where w_i^j is unknown. Let

$$P_j = \begin{bmatrix} p_{j1} & p_{j2} & p_{j3} \\ p_{j4} & p_{j5} & p_{j6} \\ p_{j7} & p_{j8} & p_{j9} \end{bmatrix}. \tag{5}$$

After eliminating the unknown w_i^j from equation (4) and rearranging the equation, we can get

$$\begin{cases} u_i^0 p_{j1} + v_i^0 p_{j2} + p_{j3} + 0 \cdot p_{j4} + 0 \cdot p_{j5} + 0 \cdot p_{j6} + (-u_i^0 u_i^j) p_{j7} + (-v_i^0 u_i^j) p_{j8} + (-u_i^j) p_{j9} = 0 \\ 0 \cdot p_{j1} + 0 \cdot p_{j2} + 0 \cdot p_{j3} + u_i^0 p_{j4} + v_i^0 p_{j5} + p_{j6} + (-u_i^0 v_i^j) p_{j7} + (-v_i^0 v_i^j) p_{j8} + (-v_i^j) p_{j9} = 0 \end{cases}. \tag{6}$$

The formula (6) is a group of equation including two equations involving nine unknown items of P_j, which means that there are two equations involving nine unknown items of P_j inferred from a pair of matched SIFT feature points. Thus, four pairs of matched SIFT feature points can bring about an equation group including eight equations involving nine unknown items of P_j, so P_j can be determined in the meaning of a scale factor difference when p_{j9} =1. When there are M (M≥5) pairs of matched SIFT feature points between EI_0 and EI_j, we can get an over-determined system of linear equations with the following form.

$$A\mathrm{p} = 0 , \tag{7}$$

where A is a 2M×9 matrix, and $\mathrm{p} = (p_{j1}, p_{j2}, p_{j3}, p_{j4}, p_{j5}, p_{j6}, p_{j7}, p_{j8}, p_{j9})^{\mathrm{T}}$ is a column vector from the items of P_j. The over-determined system of linear equations (7) can be solved to determine p by virtue of SVD decomposition or Jacobi method, which make $\| A\mathrm{p} \|$ minimize and $\|\mathrm{p}\| = 1$ where $\| \|$ means the magnitude of vector.

2) Iteratively refining N-1 P_j using Levenberg-Marquardt optimization method

From equation (4), we introduce the vector $\mathrm{y}_i = (y_{i1}, y_{i2}, y_{i3})^{\mathrm{T}}$, and let

$$\mathrm{u}_i^j = P_j \mathrm{y}_i , \tag{8}$$

where u_i^j is the adjusted value of u_i^j, and y_i is the adjusted value of y_i. Using Levenberg-Marquardt optimization method aims to calculate P_j and y_i by iterative refinement, and make the error indicated by formula (9) minimize.

$$\sum_{\substack{j=1 \ldots N-1 \\ i=0 \ldots M-1}} d(\mathrm{u}_i^j, \mathrm{u}_i^j)^2 , \tag{9}$$

The initial value of P_j can take the value linearly determined in the step 1), and the initial value of y_i can take u_i^0. In fact the vector y_i can be considered as the direction of corresponding 3D point in the scene projection reconstruction space, and calculating P_j and y_i is equal to the projection reconstruction of the scene.

3) Transforming P_j into a value with unit determinant

For every P_j, let $B_j = \alpha_j P_j$ so as to $\det(B_j) = 1$. It is easy to know that $\alpha_j = \sqrt[3]{1/\det(P_j)}$, thus P_j can be transformed into B_j with unit determinant as follows

$$B_j = \sqrt[3]{1/\det(P_j)} \cdot P_j , \tag{10}$$

2.6. Computing Internal Reference Matrix K of Camera

Search an upper triangular matrix K to make $K^{-1}B_jK = R_j$ (j=1, 2…N－1) become a rotation matrix where K is the internal reference matrix of camera, and R_j represents the rotation matrix of EI_j relative to EI_0. According to the property $R_j = R_j^{-T}$ of the rotation matrix, we get $K^{-1}B_jK = K^TB_j^{-T}K^{-T}$, *i.e.*,

$$B_jKK^TB_j^T = KK^T. \tag{11}$$

Let

$$C = KK^T = \begin{pmatrix} a & b & c \\ b & d & e \\ c & e & f \end{pmatrix}. \tag{12}$$

Since C is a symmetric matrix, the equation (11) is become as follows.

$$B_jCB_j^T = C. \tag{13}$$

For every B_j, after expanding matrix multiplication in the equation (13), and eliminating redundant equations, we can get six homogeneous linear equations involving six unknowns *a*, *b*, *c*, *d*, *e* and *f*. So, there are the following steps:

1) From N-1 B_j, we establish a homogeneous over-determined system of linear equations as follows.

$$XC' = 0, \tag{14}$$

where $C' = (a,b,c,d,e,f)^T$ is a vector consisted of the independent items of *C*, and *X* is a 6(N-1)× 6 matrix. The least-square solution of C' is the eigenvector corresponding to the minimum eigenvalue of X^TX [13]. By conducting SVD decomposition of *X* or using Jacobi method, we can find the smallest eigenvalue of symmetric matrix X^TX.

2) For the generated *C*, it is necessary to judge if it is a positive definite matrix. If not, *C* is abandoned. Otherwise, go to step 3).

3) According to $C=KK^T$, solve *K* by virtue of Cholesky decomposition. If the diagonal terms of *K* are required to be positive, the Cholesky decomposition is unique

3. The Algorithm on Pose Estimation between Two Images

The algorithm about calculating the relatively rotational orientation between two images is as follows.

1) Preprocess the image pixel coordinates.

The basic assumption of pre-processing is that the coordinate of main point is in the center of the image. As for the estimation algorithm, this assumption is reasonable. Although the main point of the actual image will have some offset relative to the center, the offset is very small. Using this assumption can estimate the positional relationship between the images in the case of the unknown focal length.

The preprocessing mainly focuses on move the image pixel coordinates to the vicinity of predicted main point. As far as SIFT feature extraction is concerned, the origin of the image pixel coordinates is located in the upper left corner of the image. The translational motion of the preprocessing aims to move the origin to the center of the image.

2) Calculate the basis matrix *Q* between two images, according to the preprocessed image pixel coordinates and their corresponding relationship.

3) Execute SVD decomposition of *Q*, so as to $Q \approx UDW^T$, and det(*U*)=det(*V*)=1, where *V*=*WE*, and

$$E = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{15}$$

Let *r* and *s* represent two largest singular values which comes from two diagonal elements of matrix *D*.

4) Construct two matrices M_1 and M_x [14].

$$M_1 = \begin{pmatrix} \mathrm{U}_{11}V_{13} & \mathrm{U}_{12}V_{13} & \mathrm{U}_{13}V_{13} & r\mathrm{U}_{11}V_{11} + s\mathrm{U}_{12}V_{12} \\ \mathrm{U}_{11}V_{23} & \mathrm{U}_{12}V_{23} & \mathrm{U}_{11}V_{23} & r\mathrm{U}_{11}V_{21} + s\mathrm{U}_{12}V_{22} \\ \mathrm{U}_{21}V_{13} & \mathrm{U}_{22}V_{13} & \mathrm{U}_{23}V_{13} & r\mathrm{U}_{21}V_{11} + s\mathrm{U}_{22}V_{12} \\ \mathrm{U}_{11}V_{23} & \mathrm{U}_{22}V_{23} & \mathrm{U}_{23}V_{23} & r\mathrm{U}_{21}V_{21} + s\mathrm{U}_{22}V_{22} \end{pmatrix} \tag{16}$$

$$M_x = \begin{pmatrix} \text{-sU}_{13}V_{11} & \text{-rU}_{13}V_{12} & \text{rU}_{12}V_{12} + \text{sU}_{11}V_{11} & \text{rsU}_{13}V_{13} \\ \text{-sU}_{13}V_{21} & \text{-rU}_{13}V_{22} & \text{rU}_{12}V_{22} + \text{sU}_{11}V_{21} & \text{rsU}_{13}V_{13} \\ \text{-sU}_{23}V_{11} & \text{-rU}_{23}V_{12} & \text{rU}_{22}V_{12} + \text{sU}_{11}V_{11} & \text{rsU}_{13}V_{13} \\ \text{-sU}_{23}V_{21} & \text{-rU}_{23}V_{22} & \text{rU}_{12}V_{12} + \text{sU}_{11}V_{11} & \text{rsU}_{13}V_{13} \end{pmatrix} \tag{17}$$

Then calculate the determinant $p(x) = \det(M_1 - xM_x) = a_1 x + a_3 x^3$. Determine the values of constants a_1 and a_3 using the following algebraic methods. That is, let *x*=1 and *x*=2 respectively, we can get the following group of equations.

$$\begin{cases} a_1 + a_3 = \det(M_1 - M_x) \\ 2a_1 + 8a_3 = \det(M_1 - 2M_x) \end{cases} \tag{18}$$

We can acquire the values of constants a_1 and a_3 by solving equation group (18). If $-a_1 / a_3 < 0$, the pose problem has no solution and the algorithm terminates. Otherwise, let $x = \sqrt{-a_1 / a_3}$.

5) Solve the equation $(M_1 - xM_x)(\alpha, \beta, \gamma, 1)^T = 0$ to obtain the values of α , β and γ , and construct two matrices $X_{\alpha,\beta,\gamma}$ and $X^*_{\alpha,\beta,\gamma}$ as follows [14],

$$X_{\alpha,\beta,\gamma} = \begin{pmatrix} r & 0 & \alpha \\ 0 & s & \beta \\ 0 & 0 & \gamma \end{pmatrix}, \ X^*_{\alpha,\beta,\gamma} = \begin{pmatrix} s\gamma & 0 & 0 \\ 0 & r\gamma & 0 \\ -s\alpha & -r\beta & rs \end{pmatrix} \tag{19}$$

where *r* and *s* are two largest singular values of matrix *D* in step 3).

6) Construct the products $UX_{\alpha,\beta,\gamma}V^T$ and $UX^*_{\alpha,\beta,\gamma}V^T$, and calculate k_1 and k_2 by the following equation,

$$\begin{cases} k_2^2 = xg_{31} / f_{31} \\ k_1^2 = f_{13} / xg_{13} \end{cases} \tag{20}$$

Where f_{ij} and g_{ij} represent the corresponding items in the matrix $UX_{\alpha,\beta,\gamma}V^T$ and $UX^*_{\alpha,\beta,\gamma}V^T$. If the solved k_1 and k_2 are imaginary numbers, the pose problem has no solution and the algorithm terminates.

7) Calculate the matrix $Q' = K_2 Q K_1$, where k_1 and k_2 are the following diagonal matrix.

$$K_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & k_1 \end{pmatrix}, \ K_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & k_2 \end{pmatrix} \tag{21}$$

8) Execute SVD decomposition of Q' , so as to $Q' = U'D'V'^T$.

Let the camera projection matrix $P_1 = (K_1 | 0)$, then P_2 should be set to one of the following four matrix[14]:

$$\begin{array}{l} (K_2 U' E V'^T \mid K_2 U'(0,0,1)^T) \\ (K_2 U' E^T V'^T \mid K_2 U'(0,0,1)^T) \\ (K_2 U' E V'^T \mid -K_2 U'(0,0,1)^T) \\ (K_2 U' E^T V'^T \mid -K_2 U'(0,0,1)^T) \end{array} \tag{22}$$

According to the fact that the matching points must be located in front of the camera, we can make a correct select of P_2. The specific process is as follows.

(a) Select any pair of matched SIFT feature points, and let them be $x = (x_1, x_2, 1)^T$ and $x' = (x_1', x_2', 1)^T$, and $X = (X_1, X_2, X_3)$ and $X' = (X_1', X_2', X_3')$ represent their 3D coordinates. According to the epipolar geometry, there is a relationship between X and X' as equation (23).

$$X' = R(X - T), \tag{23}$$

where *R* represents rotation matrix, and *T* represents translation matrix, and they are respectively set to the left and right part in one of four matrix in (22). According to equation (23), we can get

$$x_1' = \frac{X_1'}{X_3'} = \frac{R_1(X - T)}{R_3(X - T)} = \frac{R_1(x - T / X_3)}{R_3(x - T / X_3)}, \tag{24}$$

where R_i means the ith row of matrix *R*. After the deformation on equation (24), the equation (25) is available.

$$X_3 = \frac{(R_1 - x_1' R_3)T}{(R_1 - x_1' R_3)x} \tag{25}$$

Thus, the three-dimensional coordinates of *X* are achieved as follows.

$$X_1 = x_1 X_3, X_2 = x_2 X_3 \tag{26}$$

According to equation (23), the coordinates $X' = (X_1', X_2', X_3')$ can be similarly determined.

(b) In accordance with Step (a), when we select one of four matrix in (22) and set *R* and *T* respectively to be the left and right part in this matrix, if the solved X_3 and X_3' are all positive, the selected matrix is the final result, and the rotation matrix *R* between two images is finally determined.

9) Determining the relatively rotational angles between two images.

According to the rotation matrix *R*, the angles of rotation can be obtained. The rotation angles are commonly expressed with the Euler angles, *i.e.*, precession angle ψ, nutation angle θ, and spin angle φ. According to the relevant knowledge of the rotation matrix, we can get the following results.

$$\begin{array}{l} \sin(\varphi) = -R_{31} \\ \tan(\psi) = R_{32} / R_{33} \\ \tan(\theta) = R_{21} / R_{11} \end{array} \tag{27}$$

where R_{ij} represents an item of rotation matrix *R*. By the knowledge of trigonometric functions, it is easy to solve three rotational angles between two images.

4. Experimental results

We took real images to confront our proposed framework with the real world. The camera to be calibrated is an off-the-shelf Panasonic AW-E300 CCD camera. The image resolution is 720×576. Only three groups of real images which come from three difference scenes are shown in the paper due to space limitation, as can be seen in Figures 1-3.

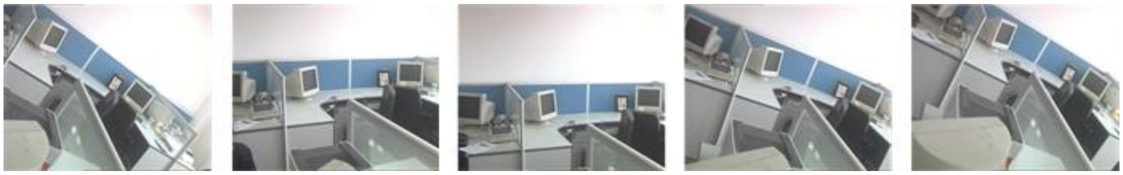

Figure 1. Group 1 of Real Images for Camera Self-Calibration

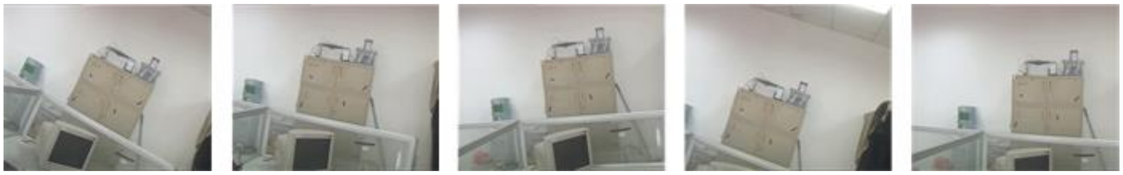

Figure 2. Group 2 of Real Images for Camera Self-Calibration

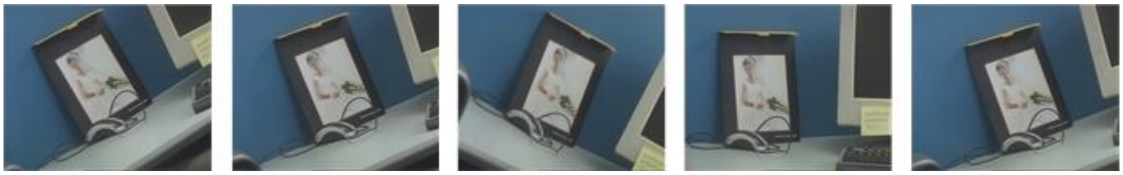

Figure 3. Group 3 of Real Images for Camera Self-Calibration

We adopted the proposed self-calibration to calibrate each image group, and presented the calibrated results of intrinsic parameters in Table 1, where the second, third and last rows are for Group 1 of image in Figure 1, Group 2 of image in Figure 2 and Group 3 of image in Figure 3, respectively. In Table 1, f is effective focal length, or distance from image plane to projective center, unit in millimeters. Here we let $f = f_u$. q is the image size factor about internal parameters and determined by $q = f_u / f_v$.

Table 1. Calibration Results of Intrinsic Parameters Applying the Proposed Self-Calibration Approach Respectively to Three Image Groups Shown in Figures 1-3

images	k_u	k_v	f	p_u	p_v	s	q
Group 1	**1066.16**	**1128.66**	**10.70**	**351.20**	**283.64**	**1.26**	**1.07**
Group 2	**1045.83**	**1115.64**	**10.49**	**368.64**	**295.84**	**1.31**	**1.07**
Group 3	**2781.13**	**2993.26**	**27.91**	**351.15**	**282.83**	**1.10**	**1.06**

During the self-calibration process, we detected a set of 2D feature points from every input image using SIFT detector respectively and matched the feature points using the sequentially cycled matching algorithm in Section 2.3. As we can see from the calibration results of intrinsic parameters, the calibrated coordinate (p_u, p_v) is very close to the ideal computer image coordinate (360, 288) of the origin in the image plane, and the calibrated distortion factor s is very close to 0, so our proposed self-calibration approach actually yields good calibration results.

We further evaluated the self-calibration accuracy by how well it can sense the 3D world. For each group of image, we applied the calibrated camera parameters to back project every 3D feature point into the image plane and computed the Euclidean distance between back-projected 2D feature point and real SIFT feature point. Then we averaged the Euclidean distances about all back-projected 2D feature points to obtain the mean back-projected error. In addition, we computed the standard deviation of the back-projected error. The mean back-projected error and the standard deviation are presented in the second and third columns in Table 2, respectively. As we can see from Table 2, the mean back-projected error and the standard deviation for the self-calibration are all lower than 1 pixel, which is satisfactory in many real computer vision and photogrammetry tasks.

Table 2. The Mean Back-Projected Error and the Standard Deviation for Three Image Groups

images	the mean back-projected error	the standard deviation
Group 1	0.641	0.560
Group 2	0.208	0.432
Group 3	0.586	0.525

For our experiments， we used a PC platform with 2.5GHz Intel Core i5-2450M processor, 4G RAM and Windows 7 to run the proposed self-calibration method implemented in C++. We recorded the computation time of obtaining the calibrated results as shown in Table 6. The computation time for the calibration method includes the time t1 for detecting and matching SIFT feature points, the time t2 for self-calibration and the time t3 for reconstructing and back projecting 3D feature point. As we can see from Table 3, except that the time of detecting and matching SIFT feature points is nearly or less than two second, the computation time for self-calibration and reconstructing 3D feature point is far less than one second, which has been very pleasing and encouraging without interrupting the execution of computer vision and photogrammetry tasks.

Table 3. The Computation Time of Calibration for Three Image Groups

images	t1 (s)	t2 (s)	t3(s)
Group 1	2.012	0.018	0.110
Group 2	1.685	0.016	0.031
Group 3	1.794	0.031	0.047

5. Conclusion

This paper presents a method about self-calibration of camera with rotary motion based on SIFT feature matching. The proposed approach requires fixing camera position, that is, camera not being displaced, but being rotated with unknown rotation angle. The proposed self-calibration approach extracts and sequentially cycled matches SIFT features for all images. Then determines the optimal reference image and effective images and calculates 2D projection transformation matrices for effective images. Finally establishes relevant linear equations so as to acquire the internal reference matrix of camera. The proposed method can be applied to quickly, accurately and stably obtain internal parameters of camera. This makes online and real-time calibrating the internal parameters of camera possible in the case of no dependent calibration references, thus improving the existing camera self-calibration.

Acknowledgements

The work is supported by the National Nature Science Foundation of China (No.61303093), the Natural Science Foundation of Henan Educational Committee

(No.12B520068) and the PHD Foundation of Zhengzhou University of Light Industry (No.2011BSJJ002)

References

[1] O. Faugeras, "Three Dimensional Computer Vision: A Geometric Viewpoint", MIT Press, Massachusetts **(1993).**

[2] R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses", IEEE J. Robotics and Automation, vol. 3, no. 4, **(1987)**, pp. 323–344.

[3] R. Lenz and R. Y. Tsai, "Techniques for calibration of the scale factor and image center for high accuracy 3D machine vision metrology", IEEE Trans. on Pattern Anal. and Machine Intell., vol. 10, no. 5, **(1988)**, pp. 713–720.

[4] P. Sturm and S. Maybank, "On Plane-Based Camera Calibration: A General Algorithm, Singularities, Applications", Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, **(1999)** June 23-25, Ft. Collins, CO, USA.

[5] Z. Zhang, "A Flexible New Techniques for Camera Calibration", IEEE Trans. on Pattern Anal and Machine Intell., vol. 22, no. 11, **(2000)**, pp. 1330–1334.

[6] O. Faugeras, T. Luong and S. Maybank, "Camera Self-Calibration: Theory and Experiments", Proc. of the Second European Conf. on Computer Vision, **(1992)** May 19-22, Santa Margherita Ligure, Italy.

[7] R. I. Hartley, "Self-Calibration of Stationary Cameras", Int. J. of Computer Vision, vol. 22, no. 1, **(1997)**, pp. 5–23.

[8] M. Pollefeys and L.Van Gool, "Stratified Self-Calibration with the Modulus Constraint", IEEE Trans. on Pattern Anal and Machine Intell., vol. 21, no. 8, **(1999)**, pp. 707-724.

[9] Q. Ji and S. Dai, "Self-calibration of a Rotating Camera with a Translational Offset", IEEE Transactions on Robotics and Automation, vol. 20, no. 1, **(2004)**, pp. 1-14.

[10] I. Junejo, X. Cao and H. Foroosh, "Calibrating Freely Moving Cameras", The 18th Int. Conf. on Pattern Recognition **(2006)** August 20-24, Hong Kong, China.

[11] D. G. Lowe, "Distinctive image features from scale-invariant key points", Int. J. of Computer Vision, vol. 60, no. 2, **(2004)**, pp. 91–110.

[12] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors", Proc. IEEE Conf. on Computer Vision and Pattern Recognition, **(2003)** June 16-22, Wisconsin, USA.

[13] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, "Numerical Recipes in C: The Art of Scientific Computing (Second Edition)", Cambridge University Press, Cambridge **(1992).**

[14] R. I. Hartley, "Estimation of Relative Camera Positions for Uncalibrated Cameras", Proc. of the Second European Conf. on Computer Vision, **(1992)** May 19-22, Santa Margherita Ligure, Italy.

Authors

Li Canlin, received PhD degree in computer science from Shanghai Jiaotong University in 2010. Since 2010 he has been in the School of Computer and Communication Engineering at Zhengzhou University of Light Industry. His research interests include image processing, multimedia, digital entertainment and software engineering. Dr. Li is a member of IEEE as well as ACM.

Zhu Fubao, received PhD degree in computer science from Wuhan University in 2008. He is an associate professor of School of Computer and Communication Engineering at Zhengzhou University of Light Industry. His research interests include image processing, Geographic Information System and computer software.

Yao Ni, received the M.S. degree from Huazhong University of Science & Technology in 2011. He is a lecturer of School of Computer and Communication Engineering at Zhengzhou University of Light Industry. His research interests include image processing and software engineering.