12th International CSI Computer Conference (CSICC'07)
Shahid Beheshti University, Tehran, Iran, 20-22 February 2007

An Iterative Geometrical Noise Cancellation Approach to Closed-form Camera Pose Estimation

Kambiz Rahbar[1], Hamid Reza Pourreza[2]

[1] SaShiraz Electro-optic and Laser Technology Research Center,
Shiraz, Iran
rahbar@sashiraz.co.ir

[2] Department of Computer Engineering, Faculty of Engineering,
Ferdowsi University, Mashad, Iran
hpourreza@um.ac.ir

Abstract

This research addresses a new iterative geometrical noise cancellation method for closed-form camera pose estimation based on collinearity theory. We first explain how to estimate camera position and it's orientation by employing extra nonsingular point of the edge line of the landmark's corner through a closed-form geometrical pose estimation algorithm. Then, we propose a new iterative noise cancellation algorithm to reduce the estimation error of camera transaction matrix, which has the most portion in camera pose estimation errors. To validate our proposed method, we test it as a computer simulation. The results show that this method is efficient, accurate and robustness. But the drawback is that the camera rotation matrix, which used in noise cancellation algorithm, may not always be inverseable.

Keywords

Iterative Geometric Noise Cancellation, Closed-Formed Camera Pose Estimation, Collinearity Theory

1. Introduction

During Recent decades, the camera pose estimation project stands out and attract itself special attentions. Generally, we can categorize camera pose estimation methods into two main trends: first, methods employing registered labels in database and try to find the position of the camera based on the comparison between capture features and database. For instance Santos et al [7] introduced an iterative geometric method for pose estimation from four co-planar points. They tried to identify possible labels composed of markers in a 2D post-processing by using a divide and conquer strategy to segment the camera's image space and attempted an iterative geometric 3D reconstruction of position and orientation in camera space, and finally they compared reconstructed labels to database for identification; And second, mathematical and geometrical methods which employing geometrical relations between captured images and camera position to solve the position and orientation of the camera; for instance Shi et al [2, 3] estimated the camera position and orientation from 2D-3D corner correspondence when vertex of the corner is occluded and Lee et al [6] integrated precise position and shape information of an object which is obtained by a pattern recognition procedure into the calibration process based upon a point correspondence scheme to estimate the position and orientation of a camera.

Anyhow, each category has its own advantages and disadvantages. For instance, also techniques based on the registered labels in database may be run faster, but they may lost pose estimation accuracy and are useful just for some applications in closed area.

Anyhow here, our work is based on Shi et al [2, 3] report. We first estimate camera position and it's orientation through mathematical and geometrical relations between captured image and camera position. Then, we try to improve the results accuracy by introducing new iterative noise cancellation algorithm which is based on collinearity theory. It should be noted that, here we assume that all of the intrinsic camera parameters are well known [4, 5, 8] and the image at hand which will be used for camera pose estimation is free from the any affect of radial distortion and slant.

This paper organized as follows: Section II describes geometrical closed-form pose estimation. Section III studies the collinearity theory. Section IV presents an iterative geometrical noise cancellation algorithm. Section V reports and analyzes the simulation results, and finally section VI gives the paper conclusions.

2. Closed-form geometrical pose estimation

Here, we are going to describe Shi et al [2, 3] method, which is based on mathematical and geometrical relations between captured image and camera position. It should be noted that, we assume all of the intrinsic camera parameters are well known and the image at hand which will be used for camera pose estimation is free from the any affect of radial distortion and slant.

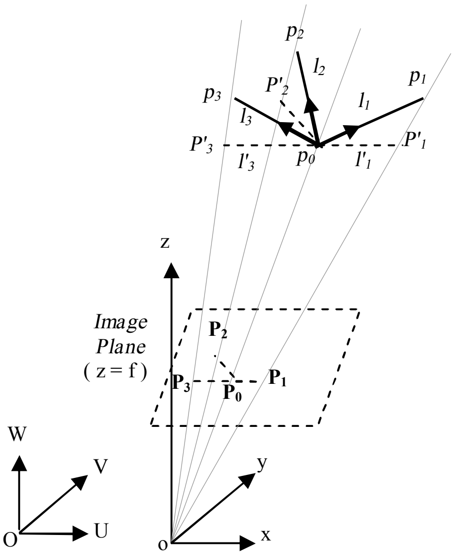


Fig.1: Basic imaging geometry of the corner

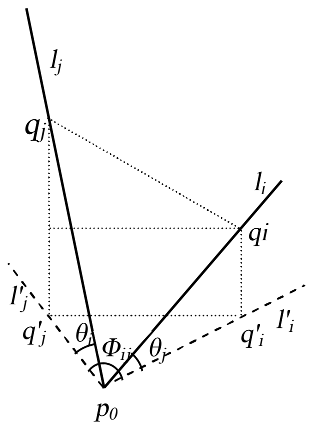


Fig.2: Edge lines l_i and l_j of the corner

Fig.1 shows the imagining geometry condition to the problem. Assume that, all of the landmarks as shown in fig.2, which are used for pose estimation have the orthogonal corners; And let $O-UVW$ be space coordinate system fixed on the ground, and $o-xyz$ be the camera coordinate system, which is chosen to be fixed on the camera with the origin coinciding with the center of the camera and the *z*-axis coinciding with the optical axis and pointing to the front of the camera. Suppose the focal length is f and the image plane is located at $z=f$ with its coordinated axes X and Y parallel to the axes x and y of the camera coordinates, respectively. Let us imagine that the coordinate system $o-xyz$ is obtained by first rotating the coordinate system $O-UVW$ with rotation matrix *R* then translating it with vector $\vec{T}$.

Let us represent a 3D corner in the camera coordinate system by the vertex and the edge directions of the corner. Because all these are 3D vectors, a 3D corner can be represented by a 3×4 matrix. Let vector $\vec{p}_0$ denote the 3D position of the vertex p_0 and $\vec{n}_i$, $i=1,2,3$, the unit vectors along 3D directions of the edge lines l_i of the corner, then the representation of the 3D corner $p_0-l_1l_2l_3$, is $c=[\vec{p}_0 \mid \vec{n}_1 \mid \vec{n}_2 \mid \vec{n}_3]$. Applying central projection to the image, we can get $\vec{P}_0$ and $\vec{L}_i$, $(i=1,2 \text{ and } 3)$, the images of vertex p_0 and edge lines l_i of the corner respectively. Thus,

$$\vec{P}_0=(X_0,Y_0,f)=f\frac{\vec{P}_0}{z_0}=(f\frac{x_0}{z_0},f\frac{y_0}{z_0},f) \tag{1}$$

where (X_0,Y_0), the first two components of $\vec{P}_0$, are the image coordinates of the vertex in the image coordinate system $O'-XY$, and the third component is f; (x_0,y_0,z_0) is the coordinate of $\vec{p}_0$ in the camera coordinate system $o-xyz$. Suppose the equation of the image line $\vec{L}_i$ in the image plane is

$$A_iX+B_iY+C_i=0 \tag{2}$$

The equation of the projecting plane of edge line l_i in the camera coordinate system is

$$A_ix+B_iy+\frac{C_i}{f}z=0 \tag{3}$$

Therefore, we can use the image line parameters to represent the normal vector $\vec{N}_i$ of the projecting plane of the corresponding edge line

$$\vec{N}_i=(A_i,B_i,\frac{C_i}{f})^T \tag{4}$$

In fact, since $\vec{N}_i$ is the normal of the projecting plane of the edge line l_i, $\vec{N}_i$ is orthogonal to $\vec{n}_i$ and the vertex vector $\vec{p}_0$. The image corner can be represented by a 3×4 matrix: $C=[\vec{P}_0 \mid \vec{N}_1 \mid \vec{N}_2 \mid \vec{N}_3]$ in camera coordinate system. Considering that the edge lines of a corner are rays, we should put a constraint on the vector $\vec{N}_i=(A_i,B_i,\frac{C_i}{f})^T$ to make the edge lines of an image corner go in one direction, i.e., in the same direction with the 3D edge lines.

To determine the directions of the edge lines for a 3D orthogonal corner from a single view, in the projecting plane of edge line l_i, $i=1,2,3$, make a ray l'_i start from vertex p_0 and be perpendicular to $\vec{p}_0$, such that $\vec{l}'_i=\vec{N}_i\times\vec{P}_0$. Suppose the angle between $\vec{l}_i$ and $\vec{l}'_i$ is θ_i, and the angle between $\vec{l}'_i$ and $\vec{l}'_j$ is φ_{ij}, $i\neq j$, $i,j=1,2,3.$ Obviously, we have

$$\cos\phi_{ij}=\frac{\vec{l}'_i.\vec{l}'_j}{|\vec{l}'_i||\vec{l}'_j|}=\frac{(\vec{N}_i\times\vec{P}_0).(\vec{N}_j\times\vec{P}_0)}{|\vec{N}_i\times\vec{P}_0||\vec{N}_j\times\vec{P}_0|} \tag{5}$$

In the case of orthogonal corner, the relationship between θ_i, θ_j and ϕ_{ij} can be easily found (fig.2). Imagine points q_i, q_j belongs to l_i, l_j respectively, and $|p_0 q_i| = |p_0 q_j| = 1$; points q'_i, q'_j belongs to l'_i, l'_j respectively, and $q_i q'_i \perp \vec{l}'_i$, $q_j q'_j \perp \vec{l}'_j$, where $\perp$ denotes ‘‘perpendicular to’’. Then, in echelon $q_i q'_i q'_j q_j$, we have

$$\begin{aligned} |q'_i q'_j|^2 &= |q_i q_j|^2 - (|q_j q'_j| - |q_i q'_i|)^2 \\ &= 2 - (\sin\theta_i - \sin\theta_j)^2 \end{aligned} \quad (6)$$

In triangle $\Delta q'_i p_{0i} q'_j$, we have

$$\begin{aligned} |q'_i q'_j|^2 &= |p_0 q'_i|^2 + |p_0 q'_j|^2 \\ &\quad - 2|p_0 q_i|^2 |p_0 q'_j|^2 \cos\phi_{ij} \\ &= (\cos\theta_i)^2 + (\cos\theta_j)^2 \\ &\quad - 2\cos\theta_i \cos\theta_j \cos\phi_{ij} \end{aligned} \quad (7)$$

From equations (6) and (7), we can get:

$$\tan\theta_i \tan\theta_j + \cos\phi_{ij} = 0 \quad (8)$$

So,

$$\begin{aligned} \tan\theta_1 \tan\theta_2 + \cos\phi_{12} = 0 \\ \tan\theta_2 \tan\theta_3 + \cos\phi_{23} = 0 \\ \tan\theta_3 \tan\theta_1 + \cos\phi_{31} = 0 \end{aligned} \quad (9)$$

Equations (2)–(4) have a closed form solution

$$\begin{aligned} \tan\theta_1 &= \pm\sqrt{-\frac{\cos\phi_{12}\cos\phi_{31}}{\cos\phi_{23}}} \\ \tan\theta_2 &= -\frac{\cos\phi_{12}}{\tan\theta_1} \\ \tan\theta_3 &= -\frac{\cos\phi_{31}}{\tan\theta_1} \end{aligned} \quad (10)$$

From equations (5) and (10), θ_1, θ_2 and θ_3 can be solved. Then, the directions of the edge line l_i of the 3D corner can be viewed as a rotation of $\vec{l}'_i$ with an angle θ_i around axis N_i, so $\vec{n}_i$ can be easily computed.

Equation (10) has two sets of solutions differing by a sign. Geometrically, it means that one can usually find two 3D orthogonal corners to fit one image corner. One of the corners is the reflection of the other about the plane which determined by $\vec{l}'_1$, $\vec{l}'_2$ and $\vec{l}'_3$. So, we should leave out a set of solution according to the real scene.

- **Camera Orientation**

In order to represent the pose of the camera conveniently, we fix the space coordinate system $O-UVW$ on the corner $p_0 - l_1 l_2 l_3$, with the origin at the vertex p_0 of the 3D corner and UVW-axes coincide with the edge lines $l_1 l_2 l_3$, respectively. Suppose that the camera coordinate system $o-xyz$ is obtained from the space coordinate system $O-UVW$ by a rotation R followed by a transaction T. Obviously, R and T correspond to the orientation matrix and location vector of camera pose respectively. Therefore, we have

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = R.[\vec{n}_1 \,|\, \vec{n}_2 \,|\, \vec{n}_3] \quad (11)$$

Because the matrix $[\vec{n}_1 \,|\, \vec{n}_2 \,|\, \vec{n}_3]$ is orthogonal, it is easy to see that

$$R = [\vec{n}_1 \,|\, \vec{n}_2 \,|\, \vec{n}_3]^{T} \quad (12)$$

- **Camera Location**

In the following we show that the camera location (transaction) can be uniquely determined if an additional image point not lying in the vertex of the corner is given. This space point is called the nonsingular point of this corner. Without loss of generality, suppose a known space point p_1 lying in edge line l_1 and $|p_1 p_2| = d$, the image point of p_1 is P_1 which lying in image line L_3. The intersection between line l'_1 and line $op_1(P_1)$ is p'_1, see fig.1.

In $\Delta p'_1 p'_2 p'_3$, according to sine theorem we have

$$\frac{|p_0 p'_1|}{\sin(\angle p_0 p_1 p'_1)} = \frac{d}{\sin(\angle o p'_1 p_0)} \quad (13)$$

Where $\angle$ denotes a corner. Obviously

$$\angle p_0 p_1 p' = \cos^{-1}\left(\frac{\vec{n}_1 . \vec{P}_1}{|\vec{n}_1| . |\vec{P}_1|}\right) \quad (14)$$

$$\angle o p' p_0 = \cos^{-1}\left(\frac{\vec{P}_1 . \vec{l}'_1}{|\vec{P}_1| . |\vec{l}'_1|}\right) \quad (15)$$

From equations (13)–(15), we can get $\angle o p' p_0$ and $|p_0 p'_1|$. Then, substituting them into the following equation

$$\vec{p}_0 = |p_0 p'_1| . (\tan(o p'_1 p_0)) | . \frac{\vec{P}_0}{|\vec{P}_0|} \quad (16)$$

we can get $\vec{p}_0$ in camera coordinate system, thus transaction $\vec{T}$ can be determined by

$$R \vec{p}_0 + \vec{T} = 0 \quad (17)$$

Therefore, in this subsection, a closed form solution of the camera pose is obtained when a 3D orthogonal corner is observed.

3. Collinearity theory

Given a set of 3D–2D correspondences $(p_i = (x_i, y_i, z_i)^T, P'_i = (X'_i, Y'_i)^T) \quad (i = 1,2,...,n \,|\, n \geq 3)$ between a 3Dscene and its 2D projective image, the relationship between the 3D–2D correspondences (p_i, P'_i) can then be represented [8], without loss of generality, as

$$z_i' \begin{pmatrix} P_i' \\ 1 \end{pmatrix} = Rp_i + t \tag{18}$$

where the standard pinhole camera model is adopted assuming that the focal length of the camera is equal to 1 for computational convenience, R and t represent the orientation and position of the camera respectively in the scene centered coordinate frame, and z_i' represents the depth of point p_i in the camera centered coordinate frame.

Given any two 3D–2D correspondences (p_1, P_1') and (p_2, P_2'), we have from equation (18):

$$z_1' \begin{pmatrix} X_1' \\ Y_1' \\ 1 \end{pmatrix} = R \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} + t \tag{19}$$

and

$$z_2' \begin{pmatrix} X_2' \\ Y_2' \\ 1 \end{pmatrix} = R \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} + t \tag{20}$$

Taking equation (19) away from equation (20) gives

$$z_1' \begin{pmatrix} X_1' \\ Y_1' \\ 1 \end{pmatrix} - z_2' \begin{pmatrix} X_2' \\ Y_2' \\ 1 \end{pmatrix} = R(p_1 - p_2) \tag{21}$$

Expanding this equation leads to

$$z_1'X_1' - z_2'X_1' = R_1(p_1 - p_2) \tag{22}$$

$$z_1'Y_1' - z_2'Y_1' = R_2(p_1 - p_2) \tag{23}$$

and

$$z_1' - z_2' = R_3(p_1 - p_2) \tag{24}$$

where R_1, R_2, and R_3 are the rows of rotation matrix R. Equation (24) is equivalent to

$$z_1' = R_3(p_1 - p_2) + z_2' \tag{25}$$

Substituting equation (24) into equations (22) and (23) results in

$$z_2'(X_2' - X_1') + X_1'R_3(p_1 - p_2) = R_2(p_1 - p_2) \tag{26}$$

and

$$z_2'(Y_2' - Y_1') + Y_1'R_3(p_1 - p_2) = R_2(p_1 - p_2) \tag{27}$$

Eliminating z_2' from these two equations yields:

$$\frac{X_2' - X_1'}{Y_2' - Y_1'} = \frac{\dfrac{R_1(p_1 - p_2)}{R_3(p_1 - p_2)} - X_1'}{\dfrac{R_2(p_1 - p_2)}{R_3(p_1 - p_2)} - Y_1'} \tag{28}$$

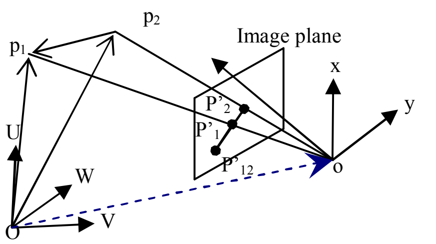


Fig.3: The collinearity constraint: image P_1, and P_2 and the projection P_{12} of their correspondence rotated point difference $R(p_1 - p_2)$ on the image plane are collinear.

This equation defines a collinearity constraint, as it shows that distinct image points P_1', P_2', and $\left(\frac{R_1(p_1 - p_2)}{R_3(p_1 - p_2)}, \frac{R_2(p_1 - p_2)}{R_3(p_1 - p_2)}\right)^T$ are collinear (Fig.3). This constraint says that the projection of the rotated difference $R(p_i - p_j)$ of any two 3D points p_i and p_j on the image plane must lie on the line connecting their corresponding image points P_i' and P_j' in the image plane. Thus, for any two 3D–2D correspondences (p_i, P_i') and (p_j, P_j') $(i, j = 1, 2, ..., n \mid i \neq j)$ we have a constraint similar to equation (26). For a number n of 3D–2D correspondences, we will have $C_n^2 = \frac{n(n-1)}{2}$ such constraints. With the number n of 3D–2D correspondences increasing, the number of such constraints will geometrically increase as $O(n^2)$. All these lines are mutually interconnected in the image plane and thus, in the general case, it is virtually impossible for different images to have the same mutually interconnected structure, enabling model based object recognition from its projective image.

4. Iterative geometrical noise cancellation

The error of transaction matrix, which has a more ration in pose estimation error, is arias more from the error of landmarks length (d). Additionally, in one hand, the corners that are detected by Harris corner detector [1] on the image plane appears from the system nature that means that they do not calculated from camera intrinsic and extrinsic parameters; they are the image of real corners of landmarks in $O-UVW$ space to $O-XY$ space. In other hand, collinearity theory exposes a simple mapping between $O-UVW$ and $o-xyz$ for some special points.

We could calculate the co-linear point, based on the image of two neighborhood corners like p_1 and p_2 on the image plane and then transformed it to the $O-UVW$ space by using extrinsic camera parameters. We know that

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = R \begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} + T \tag{29}$$

So,

$$\begin{bmatrix} u_i \\ v_i \\ w_i \end{bmatrix} = R^{-1}\left(\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} - T\right) \tag{30}$$

That means:

$$u_i = R_{11}^{-1}x_i + R_{12}^{-1}y_i + R_{13}^{-1}z_i - R_{11}^{-1}t_x - R_{12}^{-1}t_y - R_{13}^{-1}t_z \tag{31}$$

$$v_i = R_{21}^{-1}x_i + R_{22}^{-1}y_i + R_{23}^{-1}z_i - R_{21}^{-1}t_x - R_{22}^{-1}t_y - R_{23}^{-1}t_z \tag{32}$$

$$w_i = R_{31}^{-1}x_i + R_{32}^{-1}y_i + R_{33}^{-1}z_i - R_{31}^{-1}t_x - R_{32}^{-1}t_y - R_{33}^{-1}t_z \tag{33}$$

Hence, the image of co-linear point on $O-UVW$ space can be calculated as bellow:

$$\begin{aligned} u_1 - u_2 &= R_{11}^{-1} x_1 + R_{12}^{-1} y_1 + R_{13}^{-1} z_1 \\ &\quad - R_{11}^{-1} x_2 + R_{12}^{-1} y_2 + R_{13}^{-1} z_2 \\ &= R_{11}^{-1}(x_1 - x_2) + R_{12}^{-1}(y_1 - y_2) + R_{13}^{-1}(z_1 - z_2) \end{aligned} \tag{34}$$

$$\begin{aligned} v_1 - v_2 &= R_{21}^{-1} x_1 + R_{22}^{-1} y_1 + R_{23}^{-1} z_1 \\ &\quad - R_{21}^{-1} x_2 + R_{22}^{-1} y_2 + R_{23}^{-1} z_2 \\ &= R_{21}^{-1}(x_1 - x_2) + R_{22}^{-1}(y_1 - y_2) + R_{23}^{-1}(z_1 - z_2) \end{aligned} \tag{35}$$

$$\begin{aligned} w_1 - w_2 &= R_{31}^{-1} x_1 + R_{32}^{-1} y_1 + R_{33}^{-1} z_1 \\ &\quad - R_{31}^{-1} x_2 + R_{32}^{-1} y_2 + R_{33}^{-1} z_2 \\ &= R_{31}^{-1}(x_1 - x_2) + R_{32}^{-1}(y_1 - y_2) + R_{33}^{-1}(z_1 - z_2) \end{aligned} \tag{36}$$

By substituting the coordinate of P_1 and P_2 (the image of p_1 and p_2 correspondingly on the image plane) we have:

$$p_1 - p_2 = \begin{bmatrix} R_{11}^{-1}(X_1 - X_2) + R_{12}^{-1}(Y_1 - Y_1) \\ R_{21}^{-1}(X_1 - X_2) + R_{22}^{-1}(Y_1 - Y_1) \\ R_{31}^{-1}(X_1 - X_2) + R_{32}^{-1}(Y_1 - Y_1) \end{bmatrix} \tag{37}$$

As the image of co-linear point on $O-UVW$ space respondent to the collinearity theory is $p_1 - p_2$ and we know that

$$d = \sqrt{(p_1 - p_2)^2} = \sqrt{(u_1 - u_2)^2 + (v_1 - v_2)^2 + (w_1 - w_2)^2} \tag{38}$$

So, we could calculate d' (the estimation of d which is pre-given in calculating T) from co-liner point. Anyway because of CCD digitization d and d' may not be equal and have a little differences.

If d and d' had a poor differences, we could concluded that, the pre-given d is accurate enough, and so, the accuracy of the camera transaction parameter (T) is admissible; else the pre-given d is not accurate and should be corrected in the way that the differences between d and d' becomes poor. To this end, we can define the bellow function to find the real and original d by some common method like downhill simplex search, which finally leads to the accurate T.

$$|F - d| = 0 \tag{39}$$

It should be noted that, F is the process of calculating d' by using collinearity theory and camera extrinsic parameters.

5. Simulation results

Now we are going to study and analyze the effective parameters on camera poses estimation process for virtual studio through computer simulation situation using single landmark. To this end a 3D calibration rig, which is shown in Fig.4(a), is employed. Fig.4(b) shows the corners which are detected by Harris corner detector [1] as features points. From these feature points and by using an extra nonsingular point of the edge line of the landmark's corner, we can extract camera extrinsic parameters, which are camera position and its orientation.

We studied effective parameters on camera pose estimation process independently through computer simulation. The simulation results after 1000 iteration are summarized in the following tables.

(a) (b)

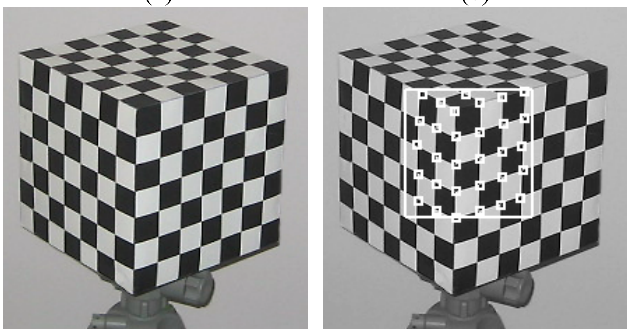


Fig.4: (a) 3D calibration rig. (b) Corners detected using Harris corner detector

Table 1 studies the error appears through changing CCD resolution. Table 2 shows the estimation's errors by adding some Gaussian noise to the landmark's length. Table 3 reports the rotation and translation RMS causes from adding Gaussian noise landmark's angles (β) for 100 iterations. It should be noted that when some noises from some sources like cubic manufacturing errors change parameter β from its original value, which is 90 degree, then Equation (40) is no longer valid. So we need to calculate θ_i from different equations as mentioned in [2, 3] which are summarized as bellow:

$$\begin{aligned} &\tan\theta_1 \tan\theta_2 + \cos\phi_{12} = \\ &\qquad \cos\beta_1 . \sqrt{(1+\tan^2\theta_1)(1+\tan^2\theta_2)} \\ &\tan\theta_2 \tan\theta_3 + \cos\phi_{23} = \\ &\qquad \cos\beta_2 . \sqrt{(1+\tan^2\theta_2)(1+\tan^2\theta_3)} \\ &\tan\theta_3 \tan\theta_1 + \cos\phi_{31} = \\ &\qquad \cos\beta_3 . \sqrt{(1+\tan^2\theta_3)(1+\tan^2\theta_1)} \end{aligned} \tag{40}$$

We solve Equation set (40) by Genetic Algorithm (GA). We reported the solution error in Table 3. Additionally we calculate the RMS of camera pan, tilt and roll motions which are reported in Table 3 too. Table 4 shows the camera pose estimation's errors when some Gaussian noise presented in camera focal length. And finally Table 5 studies the camera rotation and translation RMS causes from adding Gaussian noise to landmarks coordination. It should be noted that all of the values studied in mentioned tables are set or measured in millimeter.

Table 1 Camera rotation and translation RMS causes from changing resolution.

Resolution	RMS	
	Rotation (deg)	Translation (mm)
640×480	1.40e-06	0.012619
800×600	1.13e-06	0.004790
1024×768	6.37e-07	0.010955
1280×1024	6.90e-08	0.001731
1600×1200	6.89e-07	0.005727

Table 2 Camera rotation and translation RMS causes from adding Gaussian noise to landmark's length.

Gaussian Noise		RMS	
μ	σ	Rotation (deg)	Translation (mm)
0	0.25	0	0.028645
0	0.5	0	0.056275
0	0.75	0	0.085479
0	1	0	0.113160
0	1.25	0	0.143300
0	1.5	0	0.167350
0	1.75	0	0.196430
0	2	0	0.226360

Table 3 Camera rotation and translation RMS causes from adding Gaussian noise landmark's angles (β) for 100 iterations.

Gaussian Noise		RMS		
σ	μ	GA	Translation (mm)	Rotation (deg)
0.25	0	8.51e-04	0.3365	3.44e-05
0.5	0	7.38e-04	190.4391	0.0038
0.75	0	1.0172	8.18e-04	9.93e-05
1	0	8.45e-04	1.9106	1.82e-04
1.25	0	8.18e-04	2.2612	2.15e-04
1.5	0	8.34e-04	186.5566	0.0039
1.75	0	7.48e-04	192.3898	6.06e-38
2	0	7.97e-04	3.6022	3.51e-04

Table 4 Camera rotation and translation RMS when some Gaussian noise presented in camera focal length.

Gaussian Noise		RMS	
μ	σ	Rotation (deg)	Translation (mm)
0	0.0001	0	0.0058
0	0.00025	0	0.0146
0	0.0005	0	0.0277
0	0.00075	0	0.0383
0	0.001	0	0.0561
0	0.0025	0	0.1318
0	0.005	0	0.2758
0	0.0075	0	0.3938

By studying Tables 1 – 4 we can draw the following outcomes: in Table 1 by increasing the CCD resolution, RMS of estimation decreases, in Table 2 and 3 shows that noise in landmarks' length and angles affect translation matrix more than rotation matrix, in Table 4, it is observable that adding some Gaussian noise to the camera focal length may affect translation matrix more than rotation matrix; i.e. the camera translation matrix is infected more than other camera extrinsic parameters by the noise of effective pose estimation parameters. We can reduce the estimation error by using our noise cancellation algorithm.

Fig.8 shows the convergence of iterative noise cancellation algorithm for a pre-given landmark's length equal to 300 mm. As this figure shows, the algorithm is converged to 301.14 mm, which is the true value of landmark's length. In real condition, as other parameters like CCD resolution affects the global minimum of Equation (39), noise cancellation algorithm may converge to the different value instead of true value of landmark's length. But, fortunately the primitive steps of the convergence are similar (see Fig.9), so by limiting the maximum iterative, we may found some values near to the true value that leads to the better estimation of translation matrix.

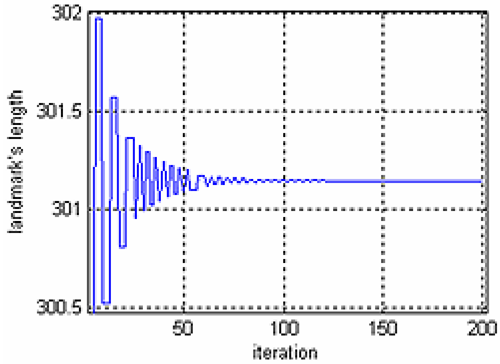


Fig8 Noise cancellation algorithm that is converging to the true value of landmark's length

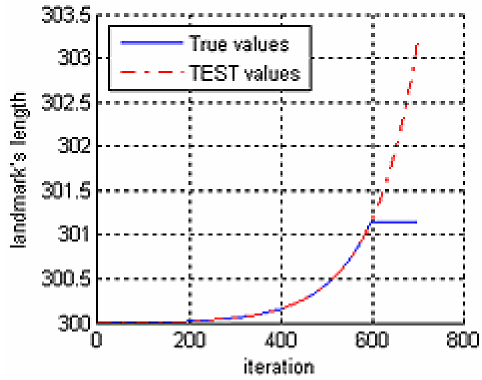


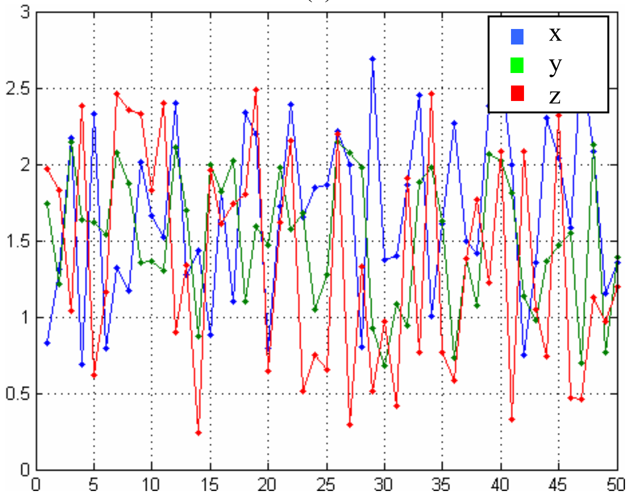
Fig. 9 Noise cancellation algorithm that is converging to the true value of landmark's length (resolution: 800×600)

Fig.10 shows the absolute error of pose estimation process for 100 sampled camera positions, while the camera is moving on a circular trajectory around the landmark by radius equal 2.3 meter. Meanwhile, Table 5 summarizes the RMS, minimum and maximum absolute error of the simulation results while using mentioned trajectory. Table 6 shows more details on a position which causes the biggest error in our simulation, while comparing it with the values obtained without noise cancellation algorithm. It is observable that employing noise cancellation algorithm increases the accuracies of results.

Table 5 Minimum, maximum, and RMS of pose estimation error (resolution: 800×600)

	x (mm)	y (mm)	z (mm)	Pan (deg)	Tilt (deg)	Roll (deg)
Min	6.71e-1	6.69e-1	2.04e-1	9.73e-4	2.73e-3	0 00e-0
MAX	2.75e-0	2.15 e-0	2.51 e-0	3.28e-2	3.58e-2	4.00e-6
RMS	1.55 e-0	1.23 e-0	1.33 e-0	1.43e-2	1.71e-2	2.16e-6

(a)



(b)

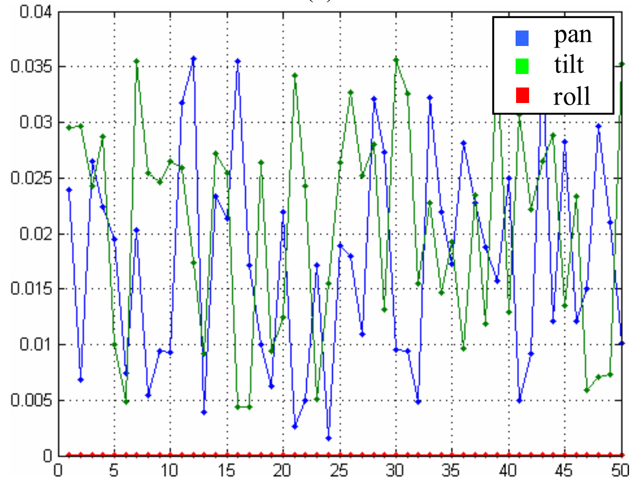


Fig.10 Pose estimation absolute error - (resolution: 800×600). (a) Absolute error of translation parameters. (b) Absolute error of orientation parameters.

Table 6 Efficiency of iterative geometrical noise cancellation algorithm

True Values		Without Noise Cancellation Algorithm		Using Noise Cancellation Algorithm	
Translation (mm)	Rotation (deg)	Translation (mm)	Rotation (deg)	Translation (mm)	Rotation (deg)
x	pan	x	pan	x	pan
-309.582	4.4392	-320.376	4.2886	-311.492	4.2886
y	tilt	y	tilt	y	tilt
-135.864	36.2833	-142.768	34.7044	-138.81	34.7044
z	roll	z	roll	z	roll
-336.542	0.0114	-352.988	0.0114	-342.627	0.0114

6. Conclusions

This paper studies a closed-form camera pose estimation algorithm with a new iterative geometrical noise cancellation approach for a virtual studio landmark based single camera system by using 2D-3D corner correspondence. The pose estimation algorithm includes two main units: Feature Extraction, and Pose Estimation unit. The first unit tries to extract the pattern corners of cube like landmark by employing Harris corner detector. Then by using these points, we try to driving the unit vectors along 3D directions of the edge lines of the Landmark's corner for pose estimation unit. By using these feature points and employing nonsingular point of this corner, the second unit tries to accurately determine camera position and orientation parameters. Our simulation shows that this method is efficient and accurate. But camera transaction matrix is too sensitive to noise. In fact it may be infected by noise more than other camera extrinsic parameters. So, we have developed an iterative geometrical noise cancellation to restore accurate camera transaction matrix. To this end, as the ration of landmark length is more than other given parameters in estimating camera transaction matrix, we recalculate the length of the landmark based on the collinearity theory and camera rotation parameters and then compare it with the given one. If these two are the same, we will conclude that the given landmark length parameter is accurate enough, so the camera transaction matrix is right; else we will make some small changes in landmark given length by non-derivative based methods like downhill simplex search and assume it as a new given landmark length parameter to recalculate camera transaction matrix. This process will be repeated again and again until the difference between given landmark length and recalculated one become poor. It should be noted that this deference because some other properties of camera system like CCD digitization may not be zero. To validate our new method, we test through computer simulation. The results show that it is efficient, accurate and robustness. But the drawback is that the camera rotation matrix may not always be inverseable.

References

[1] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," Alvey Vision Conference, pp. 147-151, 1988.

[2] F. Shi, X. Zhang and Y. Liu, "A new method of camera pose estimation using 2D-3D corner correspondence," Pattern Recognition Letters, vol. 25, pp. 1155-1163, 2004.

[3] F. Shi, X. Zhang, Y. Liu, and H. Liu, "Estimation of Camera Pose Using 2D-3D Occluded Corner Correspondence," IEEE ICSP International Conference, 2004.

[4] H. Luo, L. Zhu and H. Ding, "Camera calibration with coplanar calibration board near parallel to the imaging plane," Sensors and Actuators, 2006.

[5] J.Y. Bouguet and P.Perona, "Closed-form camera calibration in dual-space geometry," European Conference on Computer Vision (ECCV), 1998.

[6] J.M. Lee, B.H. Kim, M.H. Lee, K. Son, M.C. Lee, J.W. Choi, and S.H. Han, "Fine Active Calibration of Camera Position/Orientation through Pattern Recognition," IEEE, 1998.

[7] P. Santos, A. Stork, A. Buaes, and J. Jorge, "PTrack: Introducing a Novel Iterative Geometric Pose Estimation for a Market-based Single Camera Tracking System," IEEE Virtual Reality, Mar 25-29, Alexandria, Virginia, USA, 2006.

[8] Y.Liu and H. Holstein, "Pseudo-linearizing collinearity constraint for accurate pose estimation from single image," Pattern Recognition Letters, vol. 25 pp. 955-965, 2004.

[9] Z. Zhang, "A Flexible New Technique for Camera Calibration," Microsoft Technical report, http://research.microsoft.com/˜zhang, 1998.