

Article

Robust and Accurate Vision-Based Pose Estimation Algorithm Based on Four Coplanar Feature Points

Zimiao Zhang *, Shihai Zhang and Qiu Li

Tianjin Key Laboratory of High Speed Cutting & Precision Machining, Tianjin University of Technology and Education, Tianjin 300222, China; zshky77@163.com (S.Z.); liqiu@tute.edu.cn (Q.L.)

* Correspondence: zzm19850126@aliyun.com; Tel.: +86-22-2811-9031

Academic Editor: Joonki Paik

Received: 26 October 2016; Accepted: 14 December 2016; Published: 17 December 2016

Abstract: Vision-based pose estimation is an important application of machine vision. Currently, analytical and iterative methods are used to solve the object pose. The analytical solutions generally take less computation time. However, the analytical solutions are extremely susceptible to noise. The iterative solutions minimize the distance error between feature points based on 2D image pixel coordinates. However, the non-linear optimization needs a good initial estimate of the true solution, otherwise they are more time consuming than analytical solutions. Moreover, the image processing error grows rapidly with measurement range increase. This leads to pose estimation errors. All the reasons mentioned above will cause accuracy to decrease. To solve this problem, a novel pose estimation method based on four coplanar points is proposed. Firstly, the coordinates of feature points are determined according to the linear constraints formed by the four points. The initial coordinates of feature points acquired through the linear method are then optimized through an iterative method. Finally, the coordinate system of object motion is established and a method is introduced to solve the object pose. The growing image processing error causes pose estimation errors the measurement range increases. Through the coordinate system, the pose estimation errors could be decreased. The proposed method is compared with two other existing methods through experiments. Experimental results demonstrate that the proposed method works efficiently and stably.

Keywords: pose estimation; four coplanar points; analytical and iterative; linear constraints; the coordinate system of object motion

1. Introduction

With the development of modern industrial technology, quickly and accurately determining the position and orientation between objects is becoming more and more important. This process is called pose estimation. Vision-based pose measurement technology, also known as perspective-n-point (PNP) problem, is to determine the position and orientation of a camera and a target with n feature points on the condition of knowing their world coordinates and 2D image pixel coordinates. It has the advantages of non-contact and high efficiency. It can be widely applied in robotics [1], autonomous aerial refueling [2], electro-optic aiming systems [3,4], virtual reality [5], etc.

The existing approaches to solve object poses can be divided into two categories: analytical algorithms and iterative algorithms. Analytical algorithms apply linear methods to obtain algebraic solutions: Hu et al. [6] uses four points to achieve a linear solution. Lepetit et al. [7] proposed a non-iterative solution named EPnP algorithm. Tang et al. [8] presented a linear algorithm on the condition of five points. Ansar et al. [9] presented a general framework to directly recover the rotation and translation. Duan et al. [10] introduced a new affine invariant of trapezium to realize pose estimation and plane measurement.

As for iterative algorithms, pose estimation is formulated as a nonlinear problem with the constraints, and then it is solved using nonlinear optimization algorithms, most typically, Levenberg-Marquardt method. DeMenthon et al. [11–13] proposed POSIT algorithm. It gets the initial value of the solution using a scaled orthographic model to approximate the perspective projection model. Zhang et al. [14,15] proposed a two-stage iterative algorithm. The iterative algorithm is divided into a depth recovery stage and an absolute orientation stage. Peng et al. [16] achieved the object pose non-linearly on the basis of five points using a least squares approach. Liu et al. [17] gets the object pose based on the corresponding geometrical constraints formed by four non-coplanar points. Zhang et al. [18] proposed an efficient solution for vision-based pose determination of a parallel manipulator. Fun et al. [19] proposed a robust and high accurate pose estimation method to solve the PNP problem in real time.

The analytical solutions generally take less computation time. However, they are sensitive to observation noise and usually obtain lower accuracy. As for iterative solutions, the pose estimation is translated into a nonlinear least squares problem through the distance constraints. Then the distance error between feature points is minimized based on 2D image pixel coordinates. However, the iterative solutions also have drawbacks: (1) non-linear optimization needs a good initial estimate of the true solution, (2) they are more time consuming than analytical solutions.

During the pose estimation process, the image processing errors emerge mainly because the perspective projection point of the feature marker center and the perspective projection image center of feature markers do not coincide. The current papers focus on the reduction of image processing errors through better extraction of the image center [20–22]. However, those methods are not powerful enough to eliminate the inconsistency between the perspective projection point of the feature marker center and the center of the corresponding perspective projection image, especially when the measurement range increases.

Based on the discussion above, in this paper a robust and accurate pose estimation method based on four coplanar points is proposed: In the first step, by utilizing the linear constraints formed by points, the coordinates of points in the camera coordinate system are solved analytically. The results obtained in the first step are then set as the initial values of an iterative solving process to ensure the accuracy and convergence rate of non-linear algorithm. The Levenberg-Marquardt optimization method is utilized to refine the initial values. In the second step, the coordinate system of object motion is established and the object pose is finally solved. The growing image processing error causes greater pose estimation errors with an increasing measurement range. Through the coordinate system, the pose estimation errors could be decreased.

The rest of the paper is organized as follows: Sections 2 and 3 propose a robust and accurate pose estimation method. Section 4 provides some experiments to examine the method. Section 5 gives the conclusion.

2. The Solving of Feature Point Coordinates in the Camera Coordinate System

The target pattern with four coplanar points is designed for pose estimation as shown in Figure 1. P_0, P_1, P_2 and P_3 form a trapezium. P_0P_1 is parallel to P_2P_3 .

P_0 is set as the origin of coordinate system. The connecting line of P_0 and P_1 is Y axis. $\vec{P_0P_1} \times \vec{P_1P_3}$ is Z axis. Then X axis is $\vec{Z} \times \vec{Y}$. In this way, the world coordinate system (target coordinate system) is constructed.

To achieve the solution of target position, the coordinates of points in the camera coordinate system need to be solved first. $o_c - x_c y_c z_c$ is the camera coordinate system. $o_w - x_w y_w z_w$ is the world coordinate system. The coordinates of each point in the camera coordinate system is $P_{ci} = (x_{ci}, y_{ci}, z_{ci})^T$. The coordinates of each point in the world coordinate system are $P_{wi} = (x_{wi}, y_{wi}, z_{wi})^T$. P_{wi} is known before solving the pose. The corresponding ideal image coordinates are $I_{ui} = (x_{ui}, y_{ui}, 1)^T (i = 0, 1, 2, 3)$. The relationship of P_{ci} and I_{ui} could be described as

$$z_{ci} I_{ui} = \begin{bmatrix} s_x f / d_x & 0 & u_0 \\ 0 & f / d_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} P_{ci} = K P_{ci} \quad (1)$$

$$P_{ci} = \lambda_i K^{-1} I_{ui} (\lambda_i = z_{ci})$$

where (u_0, v_0) is the center pixel of the computer image, s_x is the uncertainty image factor, d_x and d_y are center to center distances between pixels in the row and column directions respectively, f is the focal length, and λ_i is the projection depth of P_i .

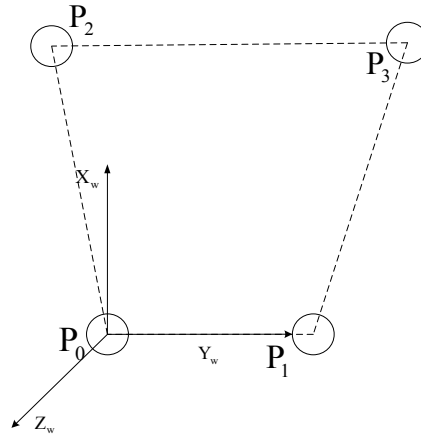


Figure 1. The measurement target with four points.

The coordinates of points in the camera coordinate system could be obtained by solving λ_i . Since P_0P_1 is parallel to P_2P_3 , two linear constraints are introduced. Then λ_i is solved (The solving process is in the Appendix A).

$$\begin{aligned} \lambda_0 &= \sigma_0 \lambda_1, \lambda_2 = \sigma_2 \lambda_1 / \varepsilon, \lambda_3 = \sigma_3 \lambda_1 / \varepsilon \\ \varepsilon &= \|P_{w1} - P_{w0}\| / \|P_{w3} - P_{w2}\| \\ \|P_{w3} - P_{w2}\| &= \|P_{c3} - P_{c2}\| = \lambda_1 \|\sigma_3 K^{-1} I_{u3} / \varepsilon - \sigma_2 K^{-1} I_{u2} / \varepsilon\| \\ \lambda_1 &= \|P_{w1} - P_{w0}\| / \|\sigma_3 K^{-1} I_{u3} - \sigma_2 K^{-1} I_{u2}\| \end{aligned} \quad (2)$$

The coordinates of points in the camera coordinates system is obtained by λ_i . So, the vector from the optical center O_c to each point could be calculated through Equation (3).

$$\begin{aligned} \vec{O_c P_{ci}} &= \sigma_i K^{-1} I_{ui} \|P_{w1} - P_{w0}\| / \|\sigma_3 K^{-1} I_{u3} - \sigma_2 K^{-1} I_{u2}\| (i = 0, 1) \\ \vec{O_c P_{ci}} &= \sigma_i K^{-1} I_{ui} \|P_{w3} - P_{w2}\| / \|\sigma_3 K^{-1} I_{u3} - \sigma_2 K^{-1} I_{u2}\| (i = 2, 3) \end{aligned} \quad (3)$$

The length between the feature points could be calculated through Equation (4).

$$D_{i,j}^2 = \|\vec{O_c P_{ci}}\|^2 + \|\vec{O_c P_{cj}}\|^2 - 2 \vec{O_c P_{ci}} \cdot \vec{O_c P_{cj}} (i, j = 0, 1, 2, 3) \quad (4)$$

In order to maintain the planarity of the four points, the coplanar constraint (CO) expressed as Equation (5) should also be considered.

$$CO = (\vec{P_{c0} P_{c1}} \times \vec{P_{c2} P_{c3}}) \cdot (\vec{P_{c0} P_{c2}} \times \vec{P_{c1} P_{c3}}) \quad (5)$$

Through Equation (6), the Levenberg-Marquardt optimization method is then used to solve λ_i . The value of λ_i obtained in Equation (2) is used as the initial value to ensure the accuracy and convergence speed of the algorithm.

$$\begin{aligned} D_{i,j}^2 - \|P_{wi}P_{wj}\| &= 0 \\ CO &= 0 \end{aligned} \quad (6)$$

3. The Solving of Object Pose

The feature points coordinates in the camera coordinate system $P_{c0}, P_{c1}, P_{c2}, P_{c3}$ are calculated through λ_i . The following step is to solve the object pose. As shown in Figure 2, $o_m - x_m y_m z_m$ is the rotation and translation coordinate system of target. The three free degrees in rotation are yaw (y_m axis and α angle), pitch (x_m axis and β angle), and roll (z_m axis and γ angle).

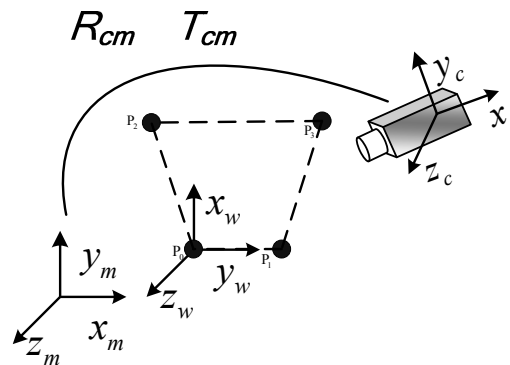


Figure 2. The rotation and translation coordinate system.

$o_m - x_m y_m z_m$ is established according to the following steps: (1) The target rotates around y_m axis. The images of the target are captured in the locations of different rotation angles. N_1 groups of spatial coordinates P_{c0} can be obtained. With the least squares method, the N_1 groups of spatial coordinates P_{c0} are used to fit a plane: $Ax + By + Cz + D = 0$. The vector of the y_m axis is,

$$v = \begin{bmatrix} r_4 & r_5 & r_6 \end{bmatrix}^T = \begin{bmatrix} \frac{A}{\sqrt{A^2+B^2+C^2}} & \frac{B}{\sqrt{A^2+B^2+C^2}} & \frac{C}{\sqrt{A^2+B^2+C^2}} \end{bmatrix}^T \quad (7)$$

(2) The target rotates around x_m axis. The images of target are captured in the locations of different rotation angles. N_2 groups of spatial coordinates P_{c0} can be obtained. With the least squares method, the N_2 groups of spatial coordinates P_{c0} are used to fit a plane: $Ex + Fy + Gz + H = 0$. The vector of the x_m axis is,

$$u = \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}^T = \begin{bmatrix} \frac{E}{\sqrt{E^2+F^2+G^2}} & \frac{F}{\sqrt{E^2+F^2+G^2}} & \frac{G}{\sqrt{E^2+F^2+G^2}} \end{bmatrix}^T \quad (8)$$

(3) The point sets P_{c0} obtained in steps (1) and (2) share one rotation center. A sphere-fitting is adopted to describe the center. According to the following sphere-fitting equation, the sphere center could be calculated. The sphere center is the rotation center (the origin of $o_m - x_m y_m z_m$).

$$\begin{aligned} \sum_{i=1}^{N_1} (P_{c0x} - x_{om})^2 + (P_{c0y} - y_{om})^2 + (P_{c0z} - z_{om})^2 - r^2 &= 0 \\ \sum_{i=1}^{N_2} (P_{c0x} - x_{om})^2 + (P_{c0y} - y_{om})^2 + (P_{c0z} - z_{om})^2 - r^2 &= 0 \end{aligned} \quad (9)$$

(x_{om}, y_{om}, z_{om}) is the sphere center and r is the sphere radius.

(4) The vector of the z_m axis is $w = u \times v = \begin{bmatrix} r_7 & r_8 & r_9 \end{bmatrix}^T$. The vector of the y_m axis is then adjusted by $v = w \times u = \begin{bmatrix} r_4 & r_5 & r_6 \end{bmatrix}^T$.

As $o_m - x_m y_m z_m$ is established, the rotation and translation matrix from $o_c - x_c y_c z_c$ to $o_m - x_m y_m z_m$ is obtained. The coordinates of feature points in $o_m - x_m y_m z_m$ are shown below.

$$R_{cm} = \begin{bmatrix} u & v & w \end{bmatrix} = \begin{bmatrix} r_1 & r_4 & r_7 \\ r_2 & r_5 & r_8 \\ r_3 & r_6 & r_9 \end{bmatrix} \quad T_{cm} = \begin{bmatrix} x_{om} & y_{om} & z_{om} \end{bmatrix}^T \quad (10)$$

$$P_{mi} = R_{cm}P_{ci} + T_{cm}$$

The coordinates of feature points in $o_m - x_m y_m z_m$ are represented with P_{mi} .

The positioning accuracy of the feature points during image processing have a direct impact on the pose estimation accuracy. The automatic identification of circular markers is more convenient. There are many available algorithms for the center positioning of circular markers. So in this paper, the center point of the circular markers is selected as the feature point. The nature of imaging is the perspective projection. It has the characteristic that the object is big when near and small when far. This causes the perspective projection point of the circular marker's center and the center of the corresponding perspective projection image (usually an ellipse) to not coincide, especially when the measurement range is greater [23]. This inconsistency causes image processing errors. Then image processing errors emerge, which may result in the pose estimation errors (The ideal image coordinates $I_{ui} = (x_{ui}, y_{ui}, 1)^T$ in Equation (1) are obtained through these image processing steps). In order to improve measurement accuracy, the following method is introduced.

The point sets P_{c0} obtained in Section 2 represent the known typical position of a target in the moving space. It means N typical positions of a target. The pose of a target in the typical position is known. According to Equation (10), N P_{m0} are obtained. Then the coordinates of P_0 in $o_m - x_m y_m z_m$ in the typical position j are represented with Q_{mj_0} . P_{mk_i} represents the coordinates of feature points in $o_m - x_m y_m z_m$ in the location k. The nearest Q_{mj_0} to P_{mk_0} is Q_{mj_0} as shown in Figure 3. P_{mk_i} is calculated as

$$O_m \vec{P}_{mk_i} = O_m \vec{Q}_{mj_i} + Q_{mj_i} \vec{P}_{mk_i} \quad (i = 0, 1, 2, 3) \quad (11)$$

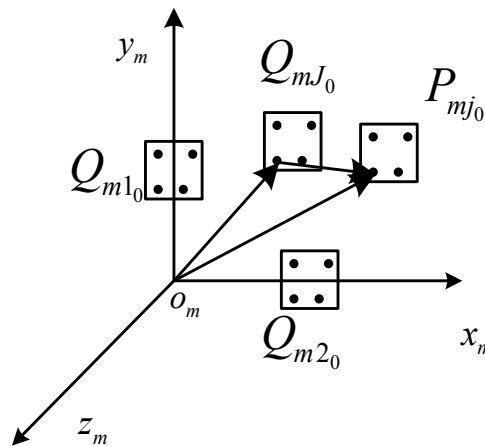


Figure 3. The solving of the object pose.

Each pose vector in the moving space corresponds to three angles. They are α (yaw angle), β (pitch angle), and γ (roll angle). They could be calculated as shown in Equation (12).

$$\begin{aligned} \alpha_{O_m P_{mk_i}} &= \alpha_{O_m Q_{mj_i}} + \alpha_{Q_{mj_i} P_{mk_i}} \\ \beta_{O_m P_{mk_i}} &= \beta_{O_m Q_{mj_i}} + \beta_{Q_{mj_i} P_{mk_i}} \\ \gamma_{O_m P_{mk_i}} &= \gamma_{O_m Q_{mj_i}} + \gamma_{Q_{mj_i} P_{mk_i}} \end{aligned} \quad (12)$$

$\alpha_{O_m Q_{mj_i}}$, $\beta_{O_m Q_{mj_i}}$ and $\gamma_{O_m Q_{mj_i}}$ are known quantities. $\alpha_{Q_{mj_i} P_{mk_i}}$, $\beta_{Q_{mj_i} P_{mk_i}}$ and $\gamma_{Q_{mj_i} P_{mk_i}}$ could be calculated according to Equation (13). In this way, image processing error caused by the inconsistency mentioned above is significantly reduced, especially when the measurement range is greater. The pose

measurement accuracy is then improved. The target pose could be represented with $\alpha_{O_m Q_{mk_i}}$, $\beta_{O_m Q_{mk_i}}$, and $\gamma_{O_m Q_{mk_i}}$.

$$\begin{aligned}
 R_Q &= \begin{bmatrix} g_1 & g_2 & g_3 \end{bmatrix} \\
 g_2 &= \frac{\overrightarrow{Q_{mj0}Q_{mj1}}}{|\overrightarrow{Q_{mj0}Q_{mj1}}|} \\
 l_1 &= \frac{\overrightarrow{Q_{mj1}Q_{mj3}}}{|\overrightarrow{Q_{mj1}Q_{mj3}}|} \\
 g_3 &= g_2 \times l_1 \\
 g_1 &= g_3 \times g_2 \\
 R_P &= \begin{bmatrix} g_4 & g_5 & g_6 \end{bmatrix} \\
 g_5 &= \frac{\overrightarrow{P_{mk0}P_{mk1}}}{|\overrightarrow{P_{mk0}P_{mk1}}|} \\
 l_2 &= \frac{\overrightarrow{P_{mk1}P_{mk3}}}{|\overrightarrow{P_{mk1}P_{mk3}}|} \\
 g_6 &= g_5 \times l_2 \\
 g_4 &= g_6 \times g_5 \\
 R_{QP} &= R_P^{-1} R_Q \\
 R_{QP} &= \begin{bmatrix} \sin \alpha \cos \gamma & \cos \alpha \sin \gamma & -\sin \alpha \\ \sin \beta \sin \alpha \cos \gamma - \cos \beta \sin \gamma & \sin \beta \sin \alpha \sin \gamma + \cos \beta \cos \gamma & \sin \beta \cos \alpha \\ \cos \beta \sin \alpha \cos \gamma + \sin \beta \sin \gamma & \cos \beta \sin \alpha \sin \gamma - \sin \beta \cos \gamma & \cos \beta \cos \alpha \end{bmatrix}
 \end{aligned} \tag{13}$$

4. Experiment Results

The experimental system that consists of a target, a three-axis rotation stage, and two CCD cameras is shown in Figure 4. The rotation range of stage in yaw axis, pitch axis, roll axis are $\pm 160^\circ$, $\pm 90^\circ$, $\pm 60^\circ$ respectively.

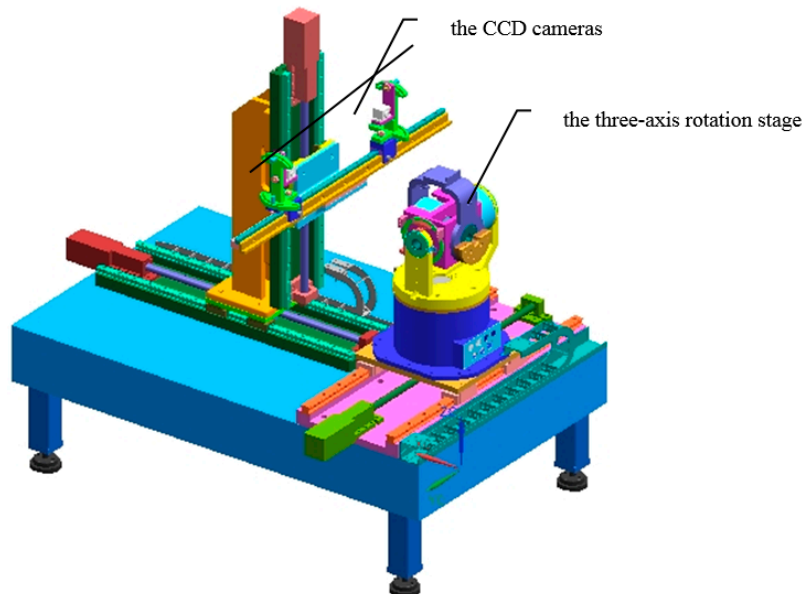


Figure 4. The measurement system.

Camera used in this paper is Teli CSB4000F-20 (Teli, Tokyo, Japan) with the resolution $2008(h) \times 2044(v)$, pixel size $0.006 \times 0.006 \text{ mm}^2$. The lens is Pentax (Tokyo, Japan) 25 mm. The positioning accuracy of rotation stage is less than $20''$. The calibration results of camera intrinsic parameters are shown in Table 1 [24]. k_1, k_2 are the radial distortion coefficients. p_1, p_2 are the tangential distortion coefficients.

Table 1. Calibration results.

Parameters	f_x	f_y	c_x	c_y	k_1	k_2	p_1	p_2
Camera 1	4341.501	4341.918	1034.667	1033.926	−0.361	0.140	−0.00024	0.00008
Camera 2	4373.530	4373.659	1000.997	1020.977	−0.373	0.342	0.00033	−0.00040

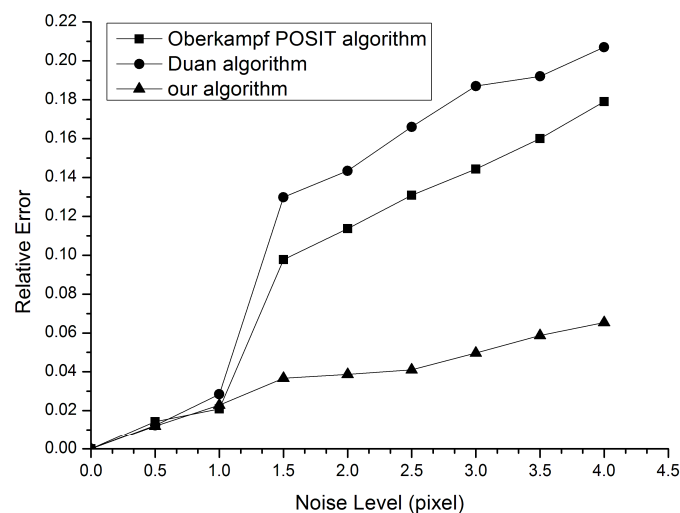
The images are captured with a single CCD camera in different locations. The feature point coordinates in the camera coordinate system are obtained in the current location. Then the rotation and translation coordinate system of target $o_m - x_m y_m z_m$ is established. $\alpha_{O_m Q_{mk_i}}$, $\beta_{O_m Q_{mk_i}}$ and $\gamma_{O_m Q_{mk_i}}$ which could be used to represent the target pose are solved.

The target pose measurement experiment includes three parts: computer simulation experiments, real image experiments for accuracy, and real image comparative experiments with other methods.

4.1. Computer Simulation Experiments

To validate accuracy and noise immunity of the algorithm in this paper, the algorithm proposed is compared with the Oberkampf POSIT algorithm [11–13] and the Duan algorithm [10] during the computer simulation experiment process. In this process, the pinhole imaging model of a camera is simulated, thus the points are transformed with perspective projection and the simulated image coordinates of points are acquired. Gaussian noise from 0 pixels to 4 pixels are added to the point's coordinates of images. The relative errors of estimated poses using the proposed algorithm, Duan algorithm, and Oberkampf POSIT algorithm are shown in Figure 5.

It can be seen that the errors are reduced after optimization and they increase with the noise level. It is noticeable that the method proposed produces better results than the other two methods, especially when the noise level is greater than one pixel. In addition, it is noted that the errors keep almost the same level under lower noise disturbance. This phenomenon can be explained as follows: owing to the fact that the Duan algorithm does not have an iterative solving process, this leads to a relatively high error. The Oberkampf POSIT method takes no account of both the coplanarity constraint and initial value of iteration process. With the noise level increasing, the factor of error changes from initial value of the iteration process to coplanarity. This results in a higher error than the proposed algorithm.

**Figure 5.** The simulation experiment results.

4.2. The Measurement Experiments for Accuracy

The measurement range of yaw angle, pitch angle, and roll angle is set to $-35^{\circ} \sim 35^{\circ}$. The measurement errors are shown in Figure 6. The measurement error is the absolute difference of the estimated angle to the true angle.

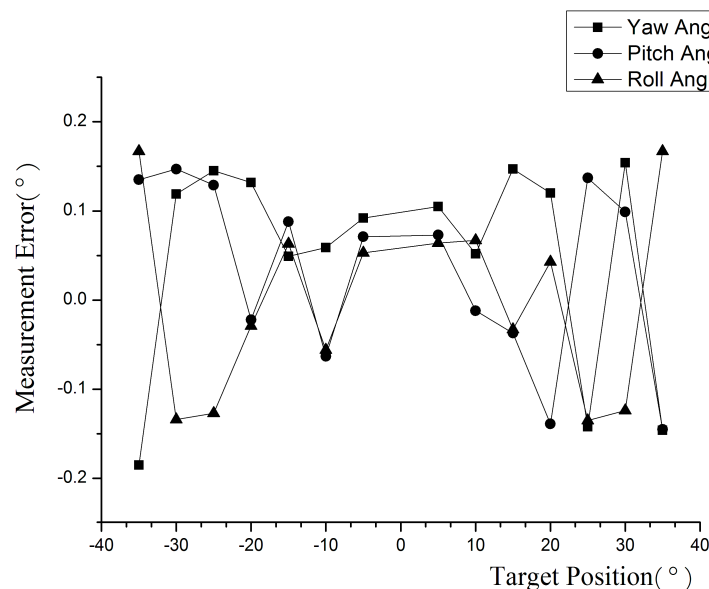


Figure 6. The measurement error of yaw, pitch, and roll angle.

4.3. The Comparative Experiments for Accuracy

The algorithm proposed in this paper, Duan algorithm, and Oberkampfs POSIT algorithm are used to calculate target poses. The root mean square (RMS) errors are displayed in Figure 7. Each point in the figure is the RMS error over multiple measurements. As to the Duan algorithm, when the rotation angle is greater, the RMS error of our method is less than that of Duan algorithm. The experiment results demonstrate that the analytical pose measurement process is sensitive to the observation noise. This error could be reduced by the iterative process in our method. At the same time, in our algorithm a good initial value is provided at the beginning of iteration to ensure pose estimation accuracy. As to the Oberkampfs POSIT algorithm, experiment results demonstrate that the image processing error mentioned in Section 3 exists in the pose measurement process. Through the method in Section 3, the error could be eliminated successfully, especially when the measurement range is greater.

By comparing the results of our method and those of the Oberkampfs POSIT and Duan algorithms, it is obvious that the measurement accuracy of our method is higher than those in the whole moving space.

As to the real applications, we first take images for the calibration board of the camera, as shown in Figure 8. The four corner circular markers (marked in red) of the calibration board are used to form a trapezium. The center points of the four circular markers are extracted from the images. Then the lengths of the four sides of the trapezium are calculated with the algorithm proposed in this paper, Duan algorithm, and Oberkampfs POSIT algorithm. The measurement errors are shown in Table 2. Figure 9 shows the image sets of the calibration board. D_{01} is the distance between P_0 and P_1 . D_{02} is the distance between P_0 and P_2 . D_{23} is the distance between P_2 and P_3 . D_{13} is the distance between P_1 and P_3 . The measurement error is the absolute difference of the estimated length to the true length. It can be seen from Table 2 that the results of our algorithm are more close to the real value. The measurement error of our method is less than 0.2 mm. The algorithm proposed in this paper is more effective.

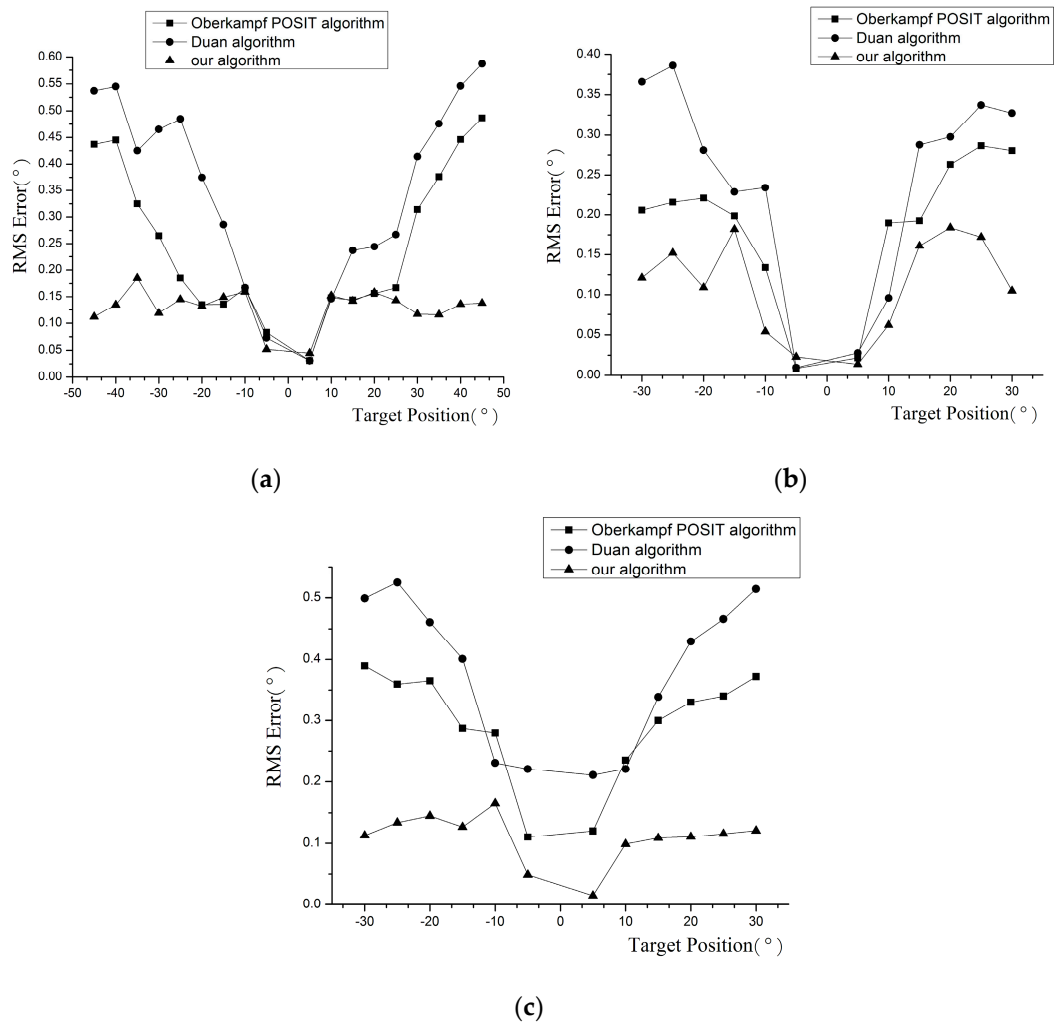


Figure 7. The RMS error of rotation angles: (a) Yaw angle; (b) Pitch angle; (c) Roll angle.

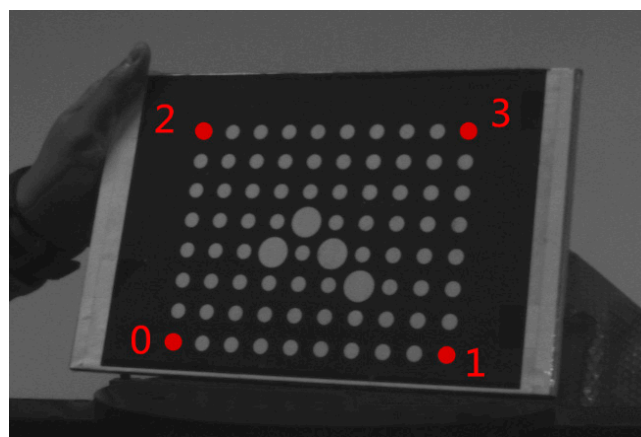
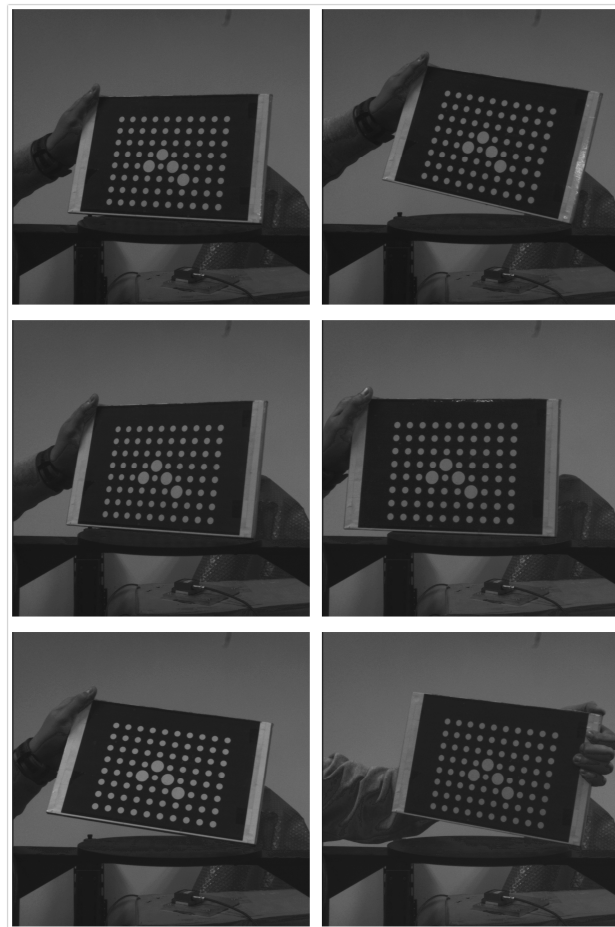


Figure 8. The image of P_0 , P_1 , P_2 , and P_3 on the calibration board.

Table 2. The measurement errors of the calibration board.

	Frame 1	Frame 2	Frame 3	Frame 4	Frame 5	Frame 6
D₀₁ (mm)						
Duan	−0.455	0.456	−0.352	0.568	−0.672	−0.433
Oberk	0.338	−0.209	0.189	−0.209	0.335	−0.326
Our	−0.047	0.148	−0.078	−0.041	0.039	−0.086
D₀₂ (mm)						
Duan	0.532	−0.622	−0.535	0.659	−0.553	0.436
Oberk	0.248	−0.295	0.306	−0.331	0.359	−0.389
Our	0.040	−0.138	0.025	0.114	0.052	0.088
D₂₃ (mm)						
Duan	0.555	−0.636	−0.418	−0.659	−0.559	0.736
Oberk	−0.323	0.203	−0.353	0.268	−0.177	0.384
Our	0.063	0.055	0.156	−0.125	−0.029	−0.139
D₁₃ (mm)						
Duan	−0.708	−0.654	−0.555	0.715	−0.453	−0.359
Oberk	0.448	−0.407	−0.348	0.207	−0.248	0.307
Our	−0.027	−0.072	0.076	−0.165	−0.089	−0.134

**Figure 9.** The image sets of the calibration board.

Then we test the proposed algorithm by experimentally detecting the pose information of a rotation stage in yaw angle, pitch angle, and roll angle. Figure 10 shows the image sets of the

rotation stage with feature points. Through the target on the rotation stage as shown in Figure 10, its pose is calculated using the algorithm proposed in this paper, Duan algorithm, and Oberkamp POSIT algorithm. The measurement errors are shown in Table 3. The measurement error is the absolute difference of the estimated angle to the true angle. It can be seen from Table 3 that the results of our algorithm are more close to the real value. The measurement error of our method is less than 0.2° . Using the same information extracted from the images, the proposed algorithm has an advantage to identify the pose of the rotation stage.

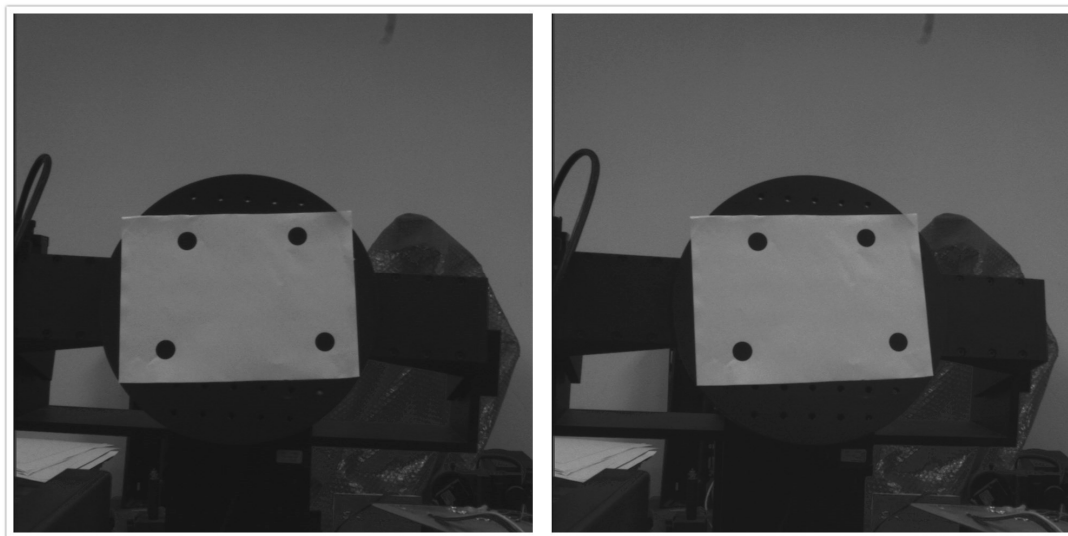


Figure 10. The image of the rotation stage with feature points.

Table 3. The measurement errors of the rotation stage with feature points.

		Position			
Yaw	5°	10°	15°	20°	25°
Duan	−0.132	0.252	0.245	0.485	−0.532
Oberk	−0.056	−0.154	−0.147	0.284	−0.334
Our	0.032	−0.057	0.045	−0.085	−0.132
Pitch (error)	5°	10°	15°	20°	25°
Duan	−0.127	−0.196	−0.201	0.386	−0.477
Oberk	0.046	0.177	−0.186	0.295	0.402
Our	0.027	0.043	−0.019	−0.102	−0.137
Roll	5°	10°	15°	20°	25°
Duan	0.129	−0.267	0.245	−0.399	−0.489
Oberk	0.036	−0.200	−0.196	0.247	−0.374
Our	−0.026	−0.052	0.049	0.093	0.140

Finally, we estimated the head pose. The pose measurement system based on inertial technology is mounted on a helmet to calculate the head pose as shown in Figure 11. The results are taken as the real values. The corresponding images of the head are captured and shown in Figure 12. The vertices of the trapezium are the outer corners of the two eyes and the mouth. The image points of the four vertices on each image are located in a manual way. The head pose is calculated using the algorithm proposed in this paper, Duan algorithm, and Oberkamp POSIT algorithm. The first image (Frame 0) in Figure 12 is set as the initial position (zero position). The angle between the current positions (Frame 1, Frame 2, Frame 3, and Frame 4) and the initial position (Frame 0) are calculated. The measurement errors are shown in Table 4. The measurement error is the absolute difference of the estimated angle to the real angle.



Figure 11. The motion capture system based on inertial technology.



Figure 12. The image sets of head pose.

Table 4. The measurement errors of head pose.

	Frame 1	Frame 2	Frame 3	Frame 4
Duan (°)				
Yaw	−0.845	−0.831	−0.776	0.766
Pitch	0.476	−0.552	0.544	0.422
Roll	−0.577	0.430	0.402	−0.427
Oberk (°)				
Yaw	−0.560	−0.620	−0.688	−0.644
Pitch	0.410	−0.374	−0.397	0.294
Roll	−0.355	−0.288	0.362	−0.403
Our (°)				
Yaw	−0.345	0.344	0.286	−0.374
Pitch	−0.177	−0.186	−0.211	0.204
Roll	−0.222	−0.199	0.175	0.232

It can be seen from Table 4 that the results of our algorithm are closer to the real value. The rough localization of the control points (the trapezium) in the image lead to the promotion of the measurement errors. However, our method could decrease these kinds of measurement errors compared with the other two methods. The errors of yaw, pitch, and roll are not independent. The errors of yaw, pitch, and roll influence each other for they constitute the rotation matrix together. When the head rotated in all the axes, as the errors of yaw, pitch, and roll influence each other, measurement errors are promoted.

The distance between object and camera is limited by the field of view. The corresponding experiment results (RMS errors) are shown in Table 5. When the camera is in different locations, the pose of the target is calculated. In Position 1 and Position 2, the camera is near to the target and the target is out of the field of view of the camera. In Position 5 and Position 6, the camera is far from the target and the target is out of the field of view of the camera. In Position 3 and Position 4, the target is in of the field of view of the camera. It can be seen as follows: if the object is out of the range of the

field of view, the measurement accuracy is lower. If the object is in the range of the field of view, the measurement accuracy is higher and little influenced by the distance between object and camera. So in order to obtain a higher accuracy, it is needed to ensure that the object is in the range of the field of view of the camera.

Table 5. The RMS error of cameras in different locations.

	Position 1	Position 2	Position 3	Position 4	Position 5	Position 6
Yaw (°)	0.411	0.384	0.192	0.189	0.477	0.512
Pitch (°)	0.400	0.372	0.186	0.177	0.444	0.504
Roll (°)	0.394	0.346	0.174	0.185	0.500	0.523

5. Conclusions

A robust and accurate vision-based pose estimation algorithm based on four coplanar feature points is presented in this paper. By combining the advantages of the analytical methods and the iterative methods, the iterative process is given a preferable initial value acquired independently by an analytical method. At the same time, the anti-noise ability is strengthened and the result is more stable. The pose estimation errors depend on the feature extraction accuracy. When the measurement range is greater, the image processing errors become greater. In the algorithm proposed, the coordinate system of object motion is established to solve the object pose. In this way, the image processing error which may result in the pose estimation errors could be reduced.

Acknowledgments: This research was supported by the National Natural Science Foundation of China (No. 51605332). This research was supported by the National Natural Science Foundation of China (No. 11302149). This research was supported by the Innovation Team Training Plan of Tianjin Universities and Colleges (Grant No. TD12-5043). This research was also supported by the Tianjin application foundation and advanced technology research program (the youth fund project), No. 15JCQNJC02400.

Author Contributions: Zimiao Zhang conceived and designed the experiments. Shihai Zhang and Qiu Li analyzed the data.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Proof of the Solving of λ_i

ε is the distance ratio of P_1P_0 and P_3P_2 .

$$\begin{aligned}\varepsilon &= \|P_{w1} - P_{w0}\| / \|P_{w3} - P_{w2}\| \\ P_{w1} &= \varepsilon(P_{w3} - P_{w2}) + P_{w0} = [P_{w0}, P_{w2}, P_{w3}][1, -\varepsilon, \varepsilon]^T\end{aligned}\quad (A1)$$

The transformation matrix from the world coordinates system to the camera coordinates system is \mathbf{R} and \mathbf{T} . P_{ci} could be expressed as $P_{ci} = \mathbf{R} \cdot P_{wi} + \mathbf{T}$, and then P_{c1} could be represented with P_{c0} , P_{c2} , P_{c3} . The first linear constraint is shown in Equation (A2).

$$\begin{aligned}P_{c1} &= \mathbf{R} \cdot P_{c4} + \mathbf{T} \\ &= \mathbf{R} \cdot (\varepsilon(P_{c3} - P_{c2}) + P_{c0}) + \mathbf{T} \\ &= \mathbf{R} \cdot (\varepsilon(P_{c3} - P_{c2} + \mathbf{T} - \mathbf{T}) + P_{c0}) + \mathbf{T} \\ &= \varepsilon(\mathbf{R} \cdot P_{c3} - \mathbf{R} \cdot P_{c2} + \mathbf{T} - \mathbf{T}) + \mathbf{R} \cdot P_{c0} + \mathbf{T} \\ &= \varepsilon(P_{c3} - P_{c2}) + P_{c0}\end{aligned}$$

$$P_{c1} = \varepsilon(P_{c3} - P_{c2}) + P_{c0} = [P_{c0}, P_{c2}, P_{c3}][1, -\varepsilon, \varepsilon]^T \quad (A2)$$

Given $P_c = [P_{c0}, P_{c2}, P_{c3}]$, according to Equations (1) and (A2), Equation (A3) is deduced.

$$P_c = [P_{c0}, P_{c2}, P_{c3}] = \mathbf{K}^{-1}[I_{u0}, I_{u2}, I_{u3}] \text{diag}[\lambda_0, \lambda_2, \lambda_3] \quad (A3)$$

The both sides of Equation (A3) are multiplied with P_{c1} . Given $[\sigma_0, \sigma_2, \sigma_3] = [I_{u0}, -I_{u2}, I_{u3}]^{-1} I_{u1}$, finally Equation (A4) is obtained which represents the second linear constraint.

$$P_c^{-1} P_{c1} = \text{diag}(\lambda_1 / \lambda_0, \lambda_1 / \lambda_2, \lambda_1 / \lambda_3) [\sigma_0, -\sigma_2, \sigma_3]^T \quad (\text{A4})$$

$P_c^{-1} P_{c1}$ could also be expressed as Equation (A5) according to Equation (A2).

$$P_c^{-1} P_{c1} = [1, -\varepsilon, \varepsilon]^T \quad (\text{A5})$$

According to both Equations (A4) and (A5), λ_i is solved as shown in Equation (2) in the paper.

References

- Kim, S.J.; Kim, B.K. Dynamic ultrasonic hybrid localization system for indoor mobile robots. *IEEE Trans. Ind. Electron.* **2013**, *60*, 4562–4573. [\[CrossRef\]](#)
- Mao, W.; Eke, F. A survey of the dynamics and control of aircraft during aerial refueling. *Nonlinear Dyn. Syst. Theory* **2008**, *8*, 375–388.
- Valenti, R.; Sebe, N.; Gevers, T. Combining head pose and eye location information for gaze estimation. *IEEE Trans. Image Process.* **2012**, *21*, 802–815. [\[CrossRef\]](#) [\[PubMed\]](#)
- Murphy-Chutorian, E.; Trivedi, M.M. Head pose estimation in computer vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 607–626. [\[CrossRef\]](#) [\[PubMed\]](#)
- Schall, G.; Wagner, D.; Reitmayr, G.; Taichmann, E.; Wieser, M.; Schmalstieg, D.; Hofmann-Wellenhof, B. Global pose estimation using multi-sensor fusion for outdoor augmented reality. In Proceedings of the 8th International Symposium on Mixed and Augmented Reality (ISMAR 2009), Santa Barbara, CA, USA, 19–22 October 2009; pp. 153–162.
- Hu, Z.Y.; Wu, F.C. A note on the number solution of the non-coplanar P4P problem. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 550–555. [\[CrossRef\]](#)
- Lepetit, V.; Moreno-Noguer, F.; Fua, P. EPnP: An accurate O(n) solution to the PnP problem. *Int. J. Comput. Vis.* **2009**, *81*, 155–166. [\[CrossRef\]](#)
- Tang, J.; Chen, W.; Wang, J. A novel linear algorithm for P5P problem. *Appl. Math. Comput.* **2008**, *205*, 628–634. [\[CrossRef\]](#)
- Ansar, A.; Daniilidis, K. Linear pose estimation from points or lines. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 578–589. [\[CrossRef\]](#)
- Duan, F.Q.; Wu, F.C.; Hu, Z.Y. Pose determination and plane measurement using a trapezium. *Pattern Recognit. Lett.* **2008**, *29*, 223–231. [\[CrossRef\]](#)
- DeMenthon, D.F.; Davis, L.S. Model-based object pose in 25 lines of code. *Int. J. Comput. Vis.* **1995**, *15*, 123–141. [\[CrossRef\]](#)
- Gramegna, T.; Venturino, L.; Cicirelli, G.; Attolico, G.; Distanto, A. Optimization of the POSIT algorithm for indoor autonomous navigation. *Robot. Auton. Syst.* **2004**, *48*, 145–162. [\[CrossRef\]](#)
- Oberkampff, D.; DeMenthon, D.F.; Davis, L.S. Iterative pose estimation using coplanar points. In Proceedings of the 1993 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'93), New York, NY, USA, 15–17 June 1993; pp. 626–627.
- Zhang, S.; Liu, F.; Cao, B.; He, L. Monocular vision-based two-stage iterative algorithm for relative position and attitude estimation of docking spacecraft. *Chin. J. Aeronaut.* **2010**, *23*, 204–210.
- Zhang, S.; Cao, X.; Zhang, F.; He, L. Monocular vision-based iterative pose estimation algorithm from corresponding feature points. *Sci. Chin. Inf. Sci.* **2010**, *53*, 1682–1696. [\[CrossRef\]](#)
- Wang, P.; Xiao, X.; Zhang, Z.; Sun, C. Study on the position and orientation measurement method with monocular vision system. *Chin. Opt. Lett.* **2010**, *8*, 55–58. [\[CrossRef\]](#)
- Liu, M.L.; Wong, K.H. Pose estimation using four corresponding points. *Pattern Recognit. Lett.* **1999**, *20*, 69–74. [\[CrossRef\]](#)
- Zhang, S.; Ding, Y.; Hao, K.; Zhang, D. An efficient two-step solution for vision-based pose determination of a parallel manipulator. *Robot. Comput. Integr. Manuf.* **2012**, *28*, 182–189. [\[CrossRef\]](#)

19. Fan, B.; Du, Y.; Cong, Y. Robust and accurate online pose estimation algorithm via efficient three-dimensional collinearity model. *IET Comput. Vis.* **2013**, *7*, 382–393. [[CrossRef](#)]
20. Pan, H.; Huang, J.; Qin, S. High accurate estimation of relative pose of cooperative space targets based on measurement of monocular vision imaging. *Optik* **2014**, *125*, 3127–3133. [[CrossRef](#)]
21. Hmam, H.; Kim, J. Optimal non-iterative pose estimation via convex relaxation. *Image Vis. Comput.* **2010**, *28*, 1515–1523. [[CrossRef](#)]
22. Ons, B.; Verstraelen, L.; Wagemans, J. A computational model of visual anisotropy. *PLoS ONE* **2011**, *6*, e21091. [[CrossRef](#)] [[PubMed](#)]
23. Heikkila, J.; Silvén, O. A four-step camera calibration procedure with implicit image correction. In Proceedings of the 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, PR, USA, 17–19 June 1997; pp. 1106–1112.
24. Zhang, Z. Flexible camera calibration by viewing a plane from unknown orientations. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; Volume 1, pp. 666–673.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).