Pose Estimation of Space Objects Based on Hybrid Feature Matching of Contour Points

Xin Zhang, Haopeng Zhang, Quanmao Wei, and Zhiguo Jiang

Image Processing Center, School of Astronautics, Beihang University,
Beijing Key Laboratory of Digital Media,
Beijing, China
{zhangxxag,zhanghaopeng,weiqm,jiangzg}@buaa.edu.cn

Abstract. This paper presents an improved pose estimation algorithm for vision-based space objects. The major weakness of most existing methods is limited convergence radius. In most cases they ignore the influence of translation, only focusing on rotation parameters. To breakthrough these limits, we utilizes hybrid local image features to explicitly establish 2D-3D correspondences between the input image and 3D model of space objects, and then estimate rotation and translation parameters based on the correspondences. Experiments with simulated models are carried out, and the results show that our algorithm can successfully estimate the pose of space objects with large convergence radius and high accuracy.

Keywords: Pose estimation, space object, hybrid local image features, 2D-3D correspondences

1 Introduction

Determining the pose parameters of a space object is one of the fundamental issues in space missions. In recent years, optical imaging system has been widely used in aerospace for many applications such as automatic rendezvous and docking, on-orbit self-serving,etc[1]. There is an urgent demand for vision-based pose estimation algorithms which can be applied to space objects. On the other hand, with the rapid development of high quality imaging sensors, images which contain more details can serve as the input of the pose estimation process, which is critical for improving accuracy of pose parameters estimated by vision-based methods.

2D-3D pose estimation is a fundamental task for computer vision applications and many feasible approaches have been proposed to solve this problem[2–8]. The main difficulty of vision-based pose estimation lies in the establishment of certain correspondences between the image and 3D model. Focusing on this point, the existing methods can be generally divided into three categories: (1)

The first kind of methods separate the pose estimation process into two parts directly, i.e. first determining the correspondences through local feature extraction such as SIFT[9], stable region extraction[10], etc., and then estimating pose parameters base on these features. However, the performance of these methods largely depends on the capacity of certain feature, and the results are not always satisfying. (2) The second kind of methods aims at bypassing the problem of determining 2D-3D correspondences by the technique of image matching. A number of silhouette images of space object's 3D model are projected beforehand with different view angle and translation, and then the object's pose is regarded as the pose parameters of the most similar silhouette image measured by calculating the similarity between the input image and each of the silhouette images. This kind of methods can roughly determine the pose parameter, but the number of silhouette images grows exponentially when a more accurate estimation result is required. (3) The third category employs iterative mechanism to determine correspondence and estimates pose parameters simultaneously. Compared with methods of the first two categories, the advantages of these methods are that no silhouette image is needed to generate beforehand and are stabler than methods using certain feature extraction. However, they suffer from the limits of convergence speed and convergence radius.

Images of space objects often have relative small size and low resolution, and thus lack of texture information. These characteristics make it difficult to accurately estimate pose parameters for space objects using vision-based methods. In this paper, we propose a contour-based approach which does not need texture information and broadly belongs to the third category. This method is motivated by the work in [5] which employs distance map to establish tentative 2D-3D point correspondences, then estimates pose parameters and updates correspondences simultaneously and iteratively. Our improvements mainly lie in two aspects: (1) Instead of using distance map for point-matching, we exploit hybrid local feature of contour points to establish tentative 2D-3D correspondences, which is more accurate and robust to translation. (2) In order to avoid interpolation, we use the original points given in 3D model to establish 2D-3D correspondences, making our method computationally much more efficient.

The rest of the paper is arranged as follows. In Section 2 we detail our method by describing the establishment of 2D-3D correspondences and the orthogonal iteration algorithm. Section 3 presents the simulation experiments and the results. Finally the conclusion is given in Section 4.

2 Iterative Pose Estimation Based on Contour Points' Hybrid Feature

This section presents the details of our algorithm and analysis them theoretically. Given the 3D model of a space object which contains vertex and normal information, our goal is to find the 3D (Euclidean) transformation making the model coincide with the object in the referential coordinate attached to a calibrated camera.

2.1 Notation

Camera model and coordinate systems configuration are presented in Fig.1. Subscripts u, p and v respectively indicate camera coordinate system, image plane coordinate system and object self-centered coordinate system.

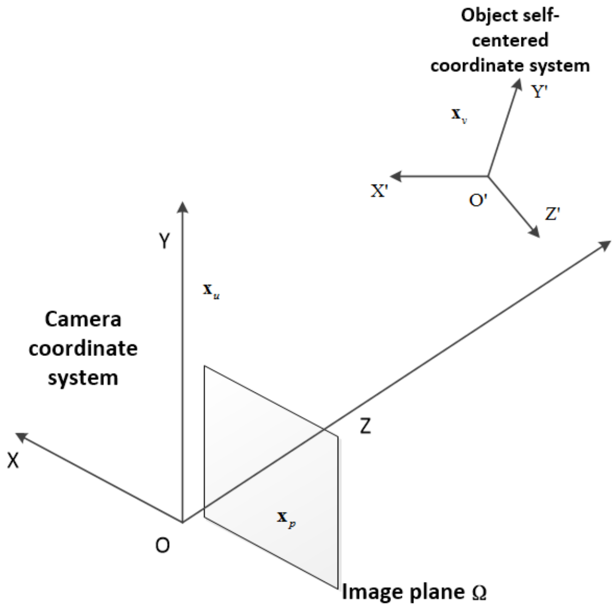

Fig. 1. Camera model and coordinate frames configuration

Let $\mathbf{x}_u = [X, Y, Z]^T$ denotes the coordinates of a point in R^3 measured with respect to a referential coordinate attached to the imaging camera. We denote by $\Omega \subset R^2$ the image plane, and assume that the camera is modeled as an ideal perspective projection: $\pi : R^3 \to \Omega; \mathbf{x}_u \to \mathbf{x}_p$, where $\mathbf{x}_p = [x, y]^T = [X/Z, Y/Z]^T$ denotes coordinates in Ω. Object self-centered coordinate system and camera coordinate system are related by the rigid transformation as

$$\mathbf{x}_u = \mathbf{R}\mathbf{x}_v + \mathbf{t} \tag{1}$$

where $\mathbf{R}$ is the rotation matrix and $\mathbf{t}$ is the translation vector. The image plane coordinate system and object self-centered coordinate system are related by the equation

$$\begin{pmatrix} \mathbf{x}_p \\ 1 \end{pmatrix} \sim \mathbf{K}(\mathbf{R}|\mathbf{t}) \begin{pmatrix} \mathbf{x}_v \\ 1 \end{pmatrix} \tag{2}$$

where symbol '$\sim$' means equal in homogeneous manner, and $\mathbf{K}$ is the inner camera parameter matrix which is known as a priori.

2.2 Establishing 2D-3D Point Correspondence

We firstly establish the 2D-2D correspondence between contour of the input image and contour of the projection image which is generated by projecting object's 3D model onto the image plane. Border following algorithm proposed in [11] is utilized to produce binary and single-pixel wide contours.

The contour point's hybrid feature extracted in our method consists of pixel's position and curvature in the image. We construct a vector which represents local feature by the following equation

$$\mathbf{f} = (f_1, f_2, f_3)^T = (x, y, \omega k)^T \tag{3}$$

where $(x, y)^T$ represents the position of the pixel, and k is the curvature of the contour at this point. For discrete contour image it is difficult to calculate curvature, so we calculate USAN [12] value as an alternative. USAN describes the target to the background ratio in a circle template, it is robust to noise and has scale invariant property. ω is a weighting coefficient, and we can rectify the relative effects between feature components to reflect the structure of the feature space accurately by adjusting the coefficient. The choice of ω has to guarantee that when the difference between the input image and the projection image is relative small, the curvature feature will not overwhelm the effect of x and y ; meanwhile when the difference between the input image and the projection image is conspicuous, the curvature feature has to play a major role. In our current implementation, the value of ω is set to 6, and tests show that the correspondence result is relative good. The curve of correspondence error between contour of input image and contour of projection image is presented in Fig.2.

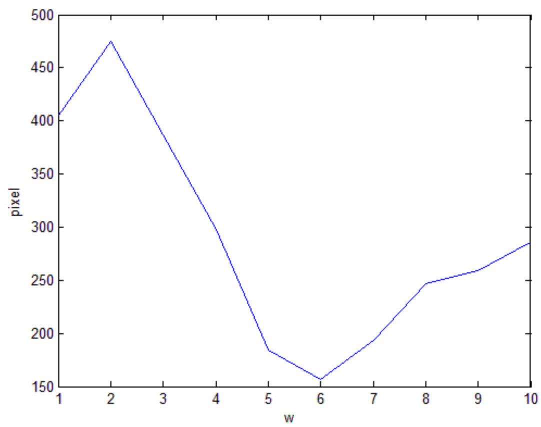

Fig. 2. Curve of correspondence error

Let $\mathbf{C}_i$ denotes the set of feature vector of the input image contour points and $\mathbf{C}_p$ denotes the set of feature vector of the projection image contour points. Given a point F_p in projection image contour with feature vector $\mathbf{f_p} = (f_{p1}, f_{p2}, f_{p3})^T$, the feature vector $\mathbf{f_i}$ of its correspondence point F_i in input image contour can be described as follow

$$\mathbf{f_i} = (f_{i1}, f_{i2}, f_{i3})^T = \min_{\mathbf{g}} \parallel \mathbf{f_p} - \mathbf{g} \parallel, \forall \mathbf{g} \in \mathbf{C}_i \tag{4}$$

where symbol '$\| \cdot \|$' means calculate two-norm of the vector.

When we create a projection image, we dye each vertex of the 3D model with different color. Then by using color attribute as index, we can efficiently retrieve the required 3D coordinate of the vertex from object's 3D model, and carry the established 2D-2D point correspondence relationship forward to the final 2D-3D point correspondences.

The algorithm in [5] establishes 2D-2D correspondences for each point in projection image, then back-projects them to 3D model by interpolation to get 2D-3D correspondences. Such approach is not only a waste of computation, but may also bring in error in the process of interpolation. As an improvement, we choose the points in projection image that exactly correspond to vertices in 3D model to establish 2D-3D correspondences. In our implementation, the number of points in projection image contour is about 600, and after our filtration, the number of 2D-3D correspondences can be reduced to about 50. The decrease of calculation is remarkable.

2.3 Estimating Pose Parameters Based on 2D-3D Correspondence

After the establishment of 2D-3D point correspondences between the input image and object's 3D model, the next step is the process of point-based pose estimation. As in [5], we adopt the orthogonal iteration (OI) algorithm proposed in [2] which is fast and numerically precise.

We define a similarity function by employ the XNOR operation '$\odot$' to the binary input image contour and the binary projection image contour, to measure the fitness of the pose estimation result returned by our method.

$$Similarity = \frac{area(\mathbf{C}_{in} \odot \mathbf{C}_{pr})}{area(\mathbf{C}_{pr})} \tag{5}$$

in which $\mathbf{C}_{in}$ represents the contour extracted from the input image, and $\mathbf{C}_{pr}$ represents the contour extracted from the projection image. If the pose estimation result is close to object's actual pose, the similarity value after the XNOR operation would be close to 1. This measurement is utilized as termination criterion in our implementation.

3 Experiment

In this section, we test the performance of our pose estimation method in convergence speed and convergence radius. The method in [5] is the baseline for comparison. All the codes involved in our algorithm are implemented in C++ and run on a PC with 2.6 GHz CPU and 8 GB RAM.

Fig.3 presents twelve frames from two pose estimation runs to give a clear inspection about the convergence process of our method. We initiate the iteration with a conspicuous deviation for rotation and translation. The six frames in (a) and (b) are extracted from a total of 100 iterations in a pose estimation run.

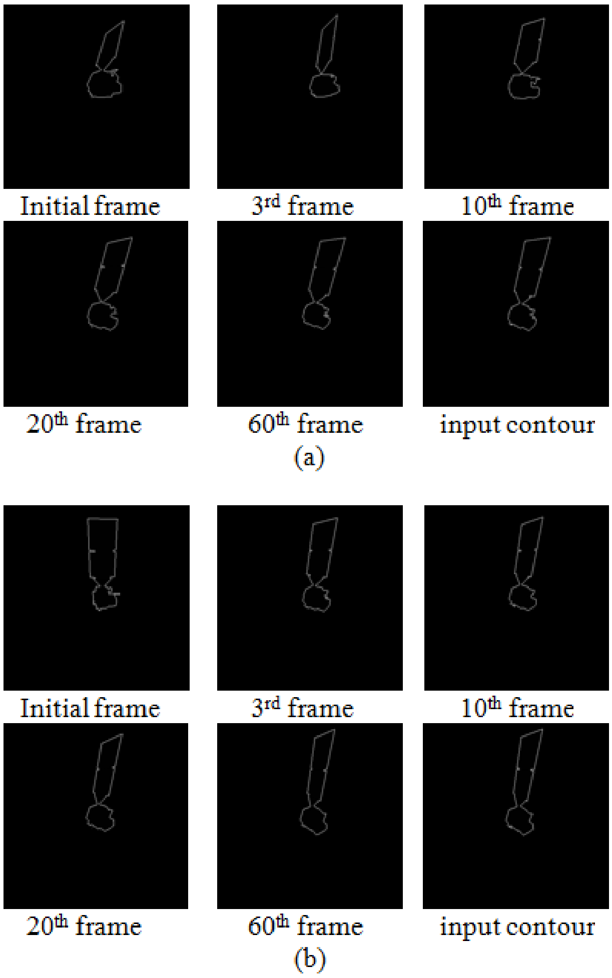

Fig. 3. Convergence illustration of our method

Table 1 summarizes the convergence speed and convergence radius of our methods and the method in [5]. The threshold of similarity is set to 0.96. For pose sample density, we set the intervals to 20° for all the three Euler angles (yaw angle ψ, pitch angle θ and roll angle ϕ). The initial pose solution required

	Method of [5]	Our Method
Average iteration number	25.8	22.7
Time per iteration(s)	0.1840	0.07357
Min radius(°)	19.7	18.5
Max radius(°)	41.2	41.7
Mean radius(°)	30.6	31.2
Error of rotation(°)	1.542	1.243
Error of translation(pixel)	3.1	2.8

Table 1. Convergence speed and convergence radius

by the iterative methods is obtained randomly within the range

$$\Delta\psi \times \Delta\theta \times \Delta\phi = [-45°, 45°] \times [-45°, 45°] \times [-45°, 45°] \tag{6}$$

For each pose we run 100 tests to explore its neighborhoods. We use the same formula as [5] to calculate convergence radius for the convenience of comparison.

$$R_{convergence} = \sqrt{\frac{N_{sucess}}{N_{total}}} \cdot R_{total} \tag{7}$$

where N_{total} represents the total number of tests for convergence radius, and $N_{success}$ represents the number of tests in which the pose parameters are calculated correctly. R_{total} represents the largest deviation of rotation angle in the tests. In our implementation, we have $N_{total} = 100$, $R_{total} = 45°$. The error of rotation and translation are the average of the deviation between estimated pose parameters and the true value in 100 tests in which the algorithm converges successfully. From the statistics in Table 1 we can see that for our method, it usually takes tens of iterations before successfully converging, and the time cost per iteration is less than 0.1s. The convergence radius is close to that of the method in [5] which is wide enough for practical application.

4 Conclusion

In this article, we focus on vision-based pose estimation of space objects.An improved contour-based iterative method is proposed which runs fast and has wide convergence radius. Our method solves the feature correspondence problem and pose estimation problem simultaneously and iteratively, and no priori feature correspondences is needed between the input monocular image and object's 3D model. Experimental results show that our method can successfully achieve space object pose estimation and meet the requirement of practical application.

Acknowledgments. This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61501009, 61371134 and 61071137), and the Fundamental Research Funds for the Central Universities.

References

1. Aghili, F., Kuryllo, M., Okouneva, G., English, C.: Robust vision-based pose estimation of moving objects for automated rendezvous and docking. International Conference on Mechatronics and Automation, 2010, pp. 305-311
2. Lu, C.P., Hager, G.D., Mjolsness, E.: Fast and globally convergent pose estimation from video images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22, (6), pp. 610-622
3. Dambrevile, S., Sandhu, R., Yezzi, A., Tannenbaum, A.: A geometric approach to joint 2D region-based segmentation and 3D pose estimation using a 3D shape prior. SIAM Journal of Imaging Sciences, 2010, 3, (1), pp. 110-132
4. Iwashita, Y., Kurazume, R., Konishi, K., Nakamoto, M., Hashizume, M., Hasegawa, T.: Fast alignment of 3D geometrical models and 2D grayscale images using 2D distance maps. Systems and Computers in Japan, 2007, 38, (14), pp. 1889-1899
5. Leng, D.W., Sun, W.D.: Contour-based iterative pose estimation of 3D rigid object. IET Computer Vision, 2011, 5, (5), pp. 291-300
6. Ansar, A., Daniilidis, K.: Linear pose estimation from points or lines. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25, (5), pp. 578-589
7. Rosenhahn, B., Perwass, C., Sommer, G.: Pose estimation of free-form contours. International Journal of Computer Vision, 2005, 62, pp. 267-289
8. Dambreville, S., Sandhu, R., Yezzi, A., Tannenbaum, A.: Robust 3D pose estimation and efficient 2D region-based segmentation from a 3D shape prior. Proceedings of the European Conference on Computer Vision, 2008, pp. 169-182.
9. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 2004, 60, (2), pp. 91-110
10. Donoser, M., Bischof, H.: Efficient maximally stable extremal region (MSER) tracking. Computer Vision and Pattern Recognition, 2006, 1, pp. 553-560
11. Suzuki, S., Abe, K.: Topological Structural Analysis of Digitized Binary Images by Border Following. Computer Vision, Graphics, and Image Processing, 1985, 30, 1, pp. 32-46
12. Smith, S.M., Brady, J.M.: Susan-a new approach to low level image processing. International Journal of Computer Vision, 1997, 23, (1),pp. 45-78