Vision-based Pose Estimation for Textureless Space Objects by Contour Points Matching

Xin Zhang, Zhiguo Jiang, *Member, IEEE,* Haopeng Zhang, *Member, IEEE,* and Quanmao Wei

***Abstract*—This paper presents a novel vision-based method to solve the 6-degree-of-freedom pose estimation problem of textureless space objects from a single monocular image. Our approach follows a coarse-to-fine procedure, utilizing only shape and contour information of the input image. To achieve invariance to initialization, we select a series of projection images which are similar to the input image and establish many-to-one 2D-3D correspondences by contour feature matching. Intensive attention is focused on outlier rejection and we introduce an innovative strategy to fully utilize geometric matching information to guide pose calculation. Experiments based on simulated images are carried out, and the results manifest that pose estimation error of our approach is about 1% even in situations with heavy outlier correspondences.**

***Index Terms*—Pose estimation, textureless space object, contour feature matching, outlier rejection.**

I. Introduction

DETERMINING the pose parameters of space objects is one of the fundamental tasks in space-based space surveillance systems [1]. The United States, Russia and Canada have invested massive resources in developing their surveillance capabilities to achieve Space Situational Awareness (SSA) [2]. The Space Based Space Surveillance (SBSS) Satellite [3], launched in September 2010, is a significant stepping stone toward a functional space-based space surveillance constellation and the future of space superiority. In February 2013, the Near-Earth Object Surveillance Satellite (NEOSSat) [4] was launched, which is the first space telescope dedicated to detecting and tracking asteroids and satellites. With the rapid improvements of high resolution imaging sensors, e.g. sCMOS sensors [5] and the Segmented Planar Imaging Detector for Electro-optical Reconnaissance (SPIDER) [6], the optical imaging system has been widely used in space surveillance for many applications such as automatic rendezvous and docking [7], position and pose estimation [1], [8]–[14], on-orbit self-servicing [15], [16], etc. Therefore, it is practicable and promising to solve the pose estimation problem of space objects by means of vision-based methods.

Pose estimation aims at retrieving the 6-degree-of-freedom (6-DoF) transformation of the object coordinate frame with reference to the camera coordinate frame. It is an intensively discussed issue in computer vision, augmented reality and robotic navigation [17]–[20]. Typical pose estimation algorithms can be roughly classified into monocular approaches and binocular approaches. Due to the long distances from the space objects to the cameras, binocular vision-based approaches cannot generate accurate range information for further calculation. Whereas the monocular approaches have less restrictions on the input, although they may require some time-consuming processes like feature extraction. For monocular approaches, the pose estimation problem is commonly divided into two independent parts, i.e. determining 2D-3D point or line correspondences, and estimating pose parameters based on these correspondences. The latter can be solved by the well studied Perspective-n-Point (PnP) or Perspective-n-Line (PnL) algorithms [22]–[28], and the main difficulty lies in the establishment of certain correspondences between 2D images and 3D models. Focusing on this issue, we can further categorize the monocular methods into three groups: geometric methods, appearance-based methods and iterative methods. Geometric methods [17], [18] rely on optical markers on target spacecraft or local feature matching methods to achieve correspondences. While appearance-based methods aim at bypassing the sub-problem of determining 2D-3D correspondences using the techniques of template matching [19] or machine learning [1], [11], [12]. Another alternative, iterative methods [13], [14], [21], allows for mismatches in the 2D-3D correspondences at the start of the process, and attempts to refine the matching accuracy iteratively. The advantages of all the three kinds of methods are integrated in our approach to accomplish a robust and accurate pose estimation algorithm capable of dealing with textureless space objects.

In this work, we consider distinctive characteristics of the space object images: (i) lack of texture; (ii) change in scale due to the variance in imaging distance; (iii) simple background. Because of the first characteristic, most existing geometric approaches are not practicable. On the other hand, relative simple background of space object images makes it easy to segment or extract contours. Under this condition, we develop a vision-based pose estimation approach for textureless space objects that only utilizes shape and contour information of the input monocular images. Moreover, intensive attention is focused on outlier rejection which is essential to the accuracy and convergence of proposed algorithm. By introducing a novel strategy to utilize geometric matching information for rejecting outliers and measuring the relative correctness among inliers, we explore the inner connections between the two sub-processes of pose estimation and integrate them effectively, which has not been attempted to our knowledge.

The rest of the paper is organized as follow. Section II

Corresponding author: Haopeng Zhang (zhanghaopeng@buaa.edu.cn).

X. Zhang, Z. Jiang, H. Zhang, and Q. Wei are with Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China, and Beijing Key Laboratory of Digital Media, Beihang University, Beijing 100191, China (e-mail: {zhang_xin_by; jiangzg; zhanghaopeng; weiqm}@buaa.edu.cn)

Manuscript received May 13, 2017; revised.

investigates the valuable works in pose estimation for space objects and illustrates the originalities of our proposed method. Section III details each component of our method from problem formulation to the PnP solution. Section IV presents the simulation experiments and the results. Finally conclusions are summarized in Section V.

II. Related Work

Most previous studies on pose estimation for space objects are based on reasonable assumptions of priori knowledge and input forms. The priori knowledge generally includes calibrated cameras and available 3D models, and the input forms vary from 2D monocular images to 3D point clouds. Aghili et al. [8] purpose an Iterative Closest Point algorithm combined with Kalman filter to achieve fast convergence for pose estimation and tracking of space objects. Much recently, Opromolla [9] integrates template matching and the concepts of principal component analysis for pose acquisition, restricting both the computational cost and data storage to adapt to the on-board situation. These two works are both designed to process 3D point clouds provided by scanning or stereovision systems. However, due to the long imaging distances, the range information can not be fully reliable. It would be more considerate to take the error of 3D point clouds into account. Compared with 3D point clouds, 2D monocular images have less restriction on the data collecting facilities, and the methods based on 2D images are direct and complete. Liu [10] specializes in cylinder-shaped space objects and applies ellipse extraction to determine pose parameters. Nevertheless the generalization ability still remains to be improved. Zhang [1], [11], [12] introduces homeomorphic manifold analysis and kernel regression-based methods to the aerospace area to estimate relative poses of space objects. In his successive works, however, training and testing of the machine learning models are restricted to image sets containing only 1-DoF and 2-DoF rotation due to the explosion of possible combinations in 6-DoF pose space.

In [13], Leng et al. propose a contour-based approach which employs distance map to achieve correspondences and adopts the orthogonal iteration (OI) algorithm to calculate pose parameters iteratively for aircraft. His work validates the feasibility of retrieving pose parameters using image contours, although a reliable initialization which is relatively close to the ground truth of pose parameters is usually needed. This limitation is largely caused by the insufficient capacity of distance map for contour matching. As an improvement, previous work in [14] takes account of the curvature of contour points to establish 2D-3D correspondences. Like the approaches we have discussed above, however, [13] and [14] give little attention to the problem of outlier rejection. This task is only considered as a preliminary process based on the technique of Random Sample Consensus (RANSAC) [35], which is time-consuming and offers no optimality guarantees. Moreover, most PnP algorithms resorting to RANSAC for outlier rejection could be trapped in local minima. This issue is extensively discussed in [23]. Ferraz et al. explore the algebraic error of linear PnP formulation and progressively reject outliers with a simple loss function. His proposed algorithm, REPPnP [23], has remarkably accelerated the process of outlier rejection. Nonetheless it still relies on the abundant data of 2D-3D correspondences and only labels the correspondences in a binary mode.

Focusing on the limitations of previous studies, we seek to make original improvements to accomplish better accuracy and robustness. The main contributions of our proposed approach are threefolds:

(i) We propose a complete 2D image-based 6 DoF pose estimation approach for textureless space objects utilizing shape and contour information. Proposed approach can achieve high accuracy not relying on good initialization or massive training data.

(ii) We establish many-to-one 2D-3D correspondences and propose an innovative strategy to calculate confidence probabilities, which are demonstrated to be effective in rejecting outliers without inefficient RANSAC.

(iii) We measure the relative correctness among inliers utilizing the confidence probabilities to improve pose accuracy, which can neither be achieved by the PnP solutions combined with RANSAC nor REPPnP. To our knowledge, proposed algorithm is the first attempt at utilizing geometric matching information to guide pose calculation. In this way we integrate the two sub-processes of pose estimation effectively. This innovative strategy may also help to avoid local minima since we introduce strong priori of reliable correspondences to restrict the pose solution space.

III. Proposed Algorithm

This section presents the details and theoretical explanations of proposed algorithm. Assuming calibrated cameras and available 3D models of space objects, our goal is to retrieve the transformation of the object coordinate frame with reference to the camera coordinate frame. Our approach follows a coarse-to-fine procedure. Specifically, an image gallery which contains about 3000 images projected from sampled viewpoints is constructed in advance. Given an input image, we select the most similar subset of the image gallery by taking Hu invariant moments [30] as similarity measurement in the coarse step. Then in the fine step, taking the projection images as intermediaries, we establish many-to-one 2D-3D correspondences between the input image points and vertices in the 3D model by ORB feature matching and color indexing. Generally, these correspondences may include mismatches. We seek to remove them by introducing coefficients to represent the confidence probabilities of the 2D-3D correspondences. All the coefficients form a diagonal weight matrix, which is combined with the OI algorithm to robustly calculate the rotation and translation parameters within an iterative framework. The overall scheme of our approach is presented in Fig. 1.

A. Problem Formulation

The configuration of the coordinate frames is presented in Fig. 2. Superscripts c, p and o indicate camera coordinate frame, image plane coordinate frame and object self-centered coordinate frame, respectively. The camera coordinate frame

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems

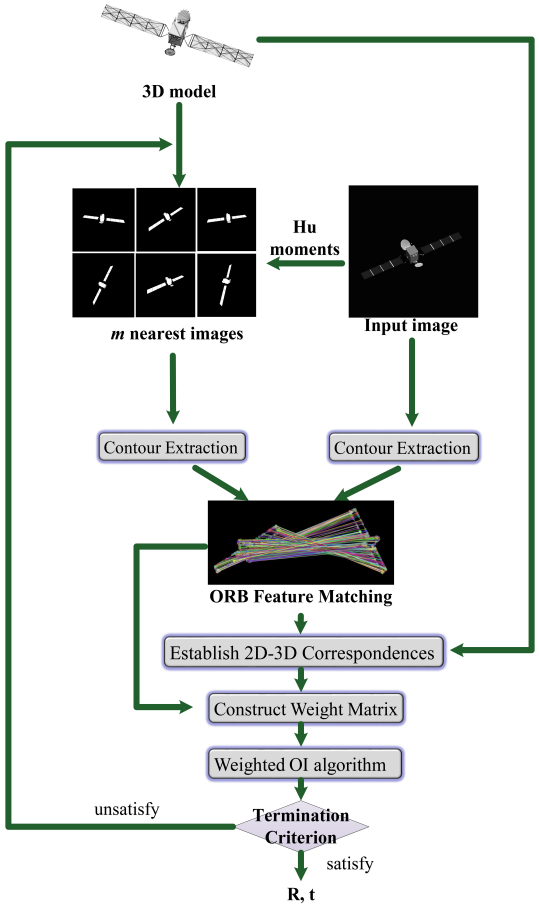


Fig. 1. Overall scheme of our proposed pose estimation method for textureless space objects.

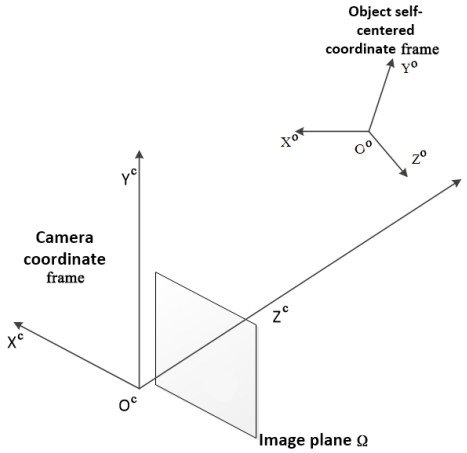


Fig. 2. Configuration of the coordinate frames.

$O^cX^cY^cZ^c$ and the object self-centered coordinate frame $O^oX^oY^oZ^o$ are 3D coordinate frames, while the image plane coordinate frame denoted by $\Omega \subset R^2$ is a 2D coordinate frame fixed at $Z^c = 1$ with its origin at the top left corner. The camera is modeled as an ideal perspective projection.

The object self-centered coordinate frame and the camera coordinate frame are related by the rigid transformation

$$\mathbf{x}^c = \mathbf{R}\mathbf{x}^o + \mathbf{t} \tag{1}$$

where $\mathbf{x}^c$ and $\mathbf{x}^o$ represent the coordinates of the same 3D vertex with reference to the camera frame and with reference to the object frame, respectively. $\mathbf{R}$ is a 3×3 rotation matrix which rotates the object frame to align with the camera frame and $\mathbf{t}$ is the translation vector equaling O^cO^o. The image plane frame and object self-centered frame are related by the equation

$$\begin{pmatrix} \mathbf{x}^p \\ 1 \end{pmatrix} \sim \mathbf{K}(\mathbf{R}|\mathbf{t}) \begin{pmatrix} \mathbf{x}^o \\ 1 \end{pmatrix} \tag{2}$$

where symbol '$\sim$' means equal in homogeneous manner, and $\mathbf{K}$ is the 3×3 inner calibration matrix known as a priori knowledge. Proposed pose estimation algorithm attempts to establish 2D-3D correspondences between the image points $\{\mathbf{x}^p\}$ and the 3D model vertices $\{\mathbf{x}^o\}$, so that we can retrieve an optimal solution of $\mathbf{R}$ and $\mathbf{t}$ by solving a set of equation (2).

B. Selecting Subset of Image Gallery

Proposed approach first generates an image gallery containing 3042 images projected from sampled viewpoints. Regardless of changes in translation and scale, we align the centers of the 3D models with Z axis of the camera frame and rotate the 3D models to obtain projection images. The rotation is formulated in the form of Euler angles, i.e. pitch angle θ, yaw angle ψ and roll angle ϕ , and rotate axes are O^oX^o, O^oY^o and O^oZ^o, respectively. Since there usually exists certain symmetry in the structure of space objects, we sample the yaw angle ψ and pitch angle θ in the range of $[-90°, 90°]$ at intervals of $15°$. The roll angle ϕ is sampled in the range of $[-180°, 180°)$ with a span of $20°$. OpenGL functions are used to draw 3D models and project images. Specifically, the R, G, B values of a vertex are determined as follows:

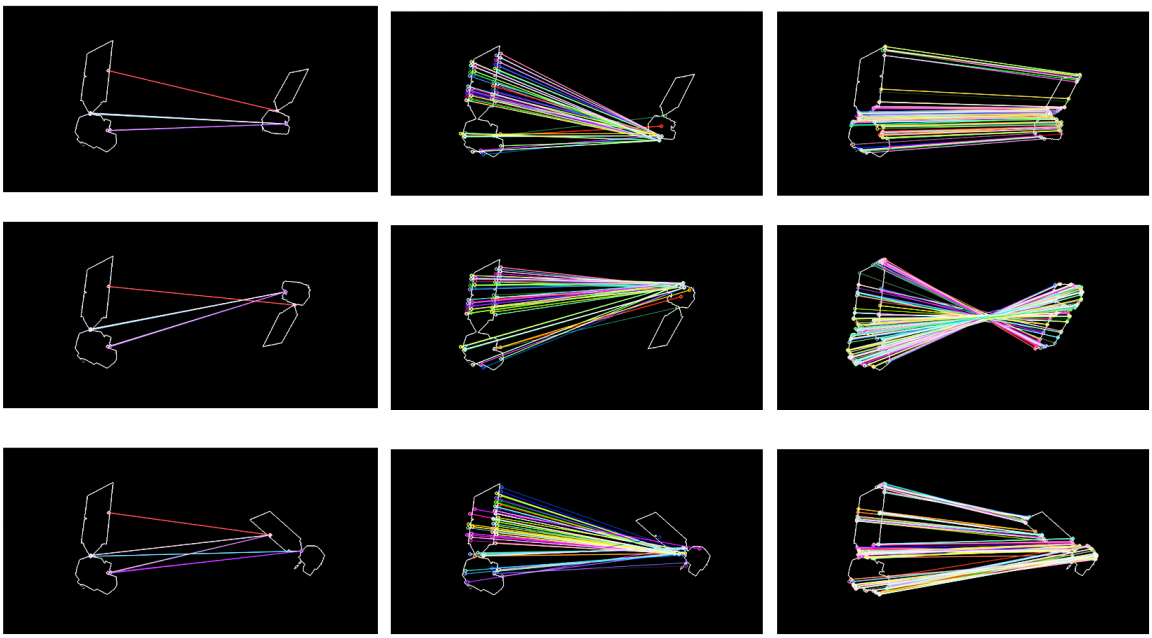
$$10 \times i = R \times 65536 + G \times 256 + B \tag{3}$$

where i refers to the index of the vertex in 3D models. In this way we are able to deal with the 3D models including one million vertices which is large enough for the usual cases. The color pattern of OpenGL is set to be smooth, so that the color of a point between two vertices is determined by the distances from the two ends. This color information generated will be useful in the process of establishing 2D-3D point correspondences. In the coarse step, however, we only utilize shape information, therefore we convert each colored projection image into a binary image. We also calculate in advance Hu moments [30] for each binary image to decrease the computational cost of the subsequent comparison process.

Our approach takes Hu moments as similarity measurement since they are invariant to rotation, translation and changing in scale. Meanwhile, the computational cost is decreased sharply owing to the binary form of images. In the coarse step we first convert a gray-level input image into a binary image, and calculate the Hu moments for comparison. The difference between the binary input image I and each binary projection image G in the image gallery is measured as follows:

$$diff(I, G) = \sum_{k=1}^{3} |h_k^I - h_k^G| \tag{4}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems



(a) SIFT (b) SURF (c) ORB

Fig. 3. Contour matching results for three local features. In each subfigure, the left contour is the contour of the input image, and the right one is the contour of a projection image in the similar subset. We could see that SIFT can hardly extract keypoints in the contours, while the matching results for SURF are obviously wrong. ORB performs best in the three local features, although we can still find some mismatching points when there is a mirror transformation between the input image and the projection image.

where $h_k, k = 1, 2, 3$ refers to the first three components of the Hu moments. We select the images corresponding to the least m value of $diff(I, G)$ as the most similar subset of the image gallery.

We have noticed that in many relative works [17] the initial similar images are selected by means of machine learning methods which perform excellently in terms of feature extraction, especially for complex images. However, machine learning methods may not suit our condition due to the simplicity of the binary images.

C. Establishing 2D-3D Point Correspondences

Taking the similar subset images as intermediaries, we attempt to establish 2D-3D correspondences between the points in input images and the vertices of 3D models. Firstly, we extract contours of the input images and the similar subset images using the Border following algorithm [32]. This process is an easy task and the computational cost is negligible because all the images have been converted into binary form. As mentioned in subsection B, the colored projection images corresponding to the m binary subset images are useful here for establishing 2D-3D correspondences. For each subset image, we traverse all the contour pixels to check if the R,G,B values in the corresponding colored projection image satisfy equation (3). By color indexing, we achieve accurate intermediate 2D-3D correspondences between the contours of subset images and the vertices of the 3D models. These intermediate correspondences are sparse because only 3D vertices of triangle patches on the contour can be retrieved. Since all the images in the subset are similar to the input images, we can deduce that most vertices should appear in more than one projection image. Therefore, there exist many-to-one correspondences in the intermediate 2D-3D correspondences. Then we map the contour points of subset images to input images points. In this work we try three popular local features for contour matching: SIFT [36], SURF [37] and ORB [29]. We extract keypoints from the contours of input images and subset images to establish 2D-2D point matchings. Fig. 3 presents matching points extracted by the three features, indicating that ORB is more suited for contour matching than the other two features in our condition due to its binary pattern.

The ORB feature matching is a sparse point-to-point matching containing mismatches. Some contour points in subset images corresponding to 3D vertices may not be matched in the primitive ORB feature matching results. So, as an approximation in [33], we calculate the homography matrices between the input images and the subset images using a least-square algorithm. Generally, the pixel coordinate errors caused by homography is acceptable when the translation between the input image and the subset image is far smaller than the imaging distance. So in our implementation, we set the imaging distance to be more than ten times the size of 3D models, which accords with the long imaging distance characteristic of space object images. Using each homography matrix $H_j, j = 1, 2, ..., m$, we can calculate the input image points corresponding to the contour points of subset images by the following equation

$$\begin{pmatrix} x_{jq}{}' \\ y_{jq}{}' \\ 1 \end{pmatrix} \sim \mathbf{H}_j \begin{pmatrix} x_{jq} \\ y_{jq} \\ 1 \end{pmatrix}, j = 1, 2, ..., m \tag{5}$$

where $[x_{jq}, y_{jq}]^T$ refers to the qth contour point in the jth

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems

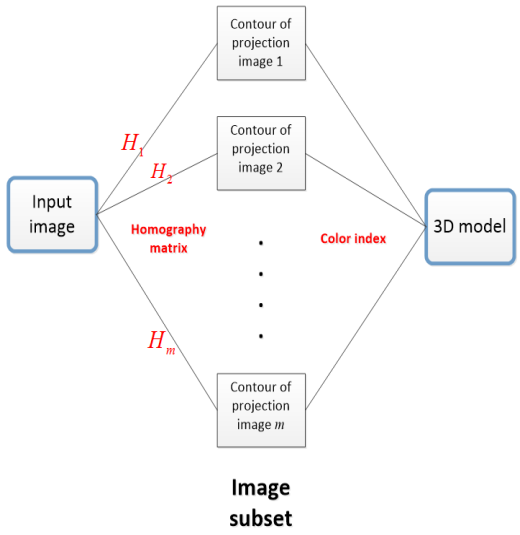


Fig. 4. Illustration of establishing 2D-3D point correspondences

subset image, and $[x_{jq}{}', y_{jq}{}']^T$ is the corresponding point in the input image. Combining the intermediate 2D-3D correspondences and equation (5), we achieve many-to-one 2D-3D correspondences between the input image points and the vertices of 3D models. Our process of establishing 2D-3D correspondences is demonstrated in Fig. 4.

Establishing 2D-3D correspondences is a crucial process in pose estimation algorithms, directly influencing the accuracy of pose parameters. [17] supposes that 3D models contain feature descriptors which are used in 3D reconstruction. This prerequisite increases the complicity of the 3D models and limits the application of the method. Assuming simple 3D models containing only vertices and structure information, our process of establishing 2D-3D correspondences is mostly motivated by [13]. However, the algorithm in [13] utilizes only one projection image as initialization and establishes 2D-2D correspondences for all the contour points in the projection image using distance map, and then back-projects them to the 3D models by interpolation to achieve 2D-3D correspondences. Such an strategy is not only a waste of computation, but may also bring in errors in interpolation. As in [14], our approach selects contour points of projection images which directly correspond to the 3D vertices by color indexing, and resorts to homography matrices to achieve many-to-one 2D-3D correspondences between the input images and the 3D models.

Fig. 5 presents typical many-to-one correspondences achieved by our approach. It can be seen that there exist mismatches in the 2D-3D correspondences. Thus, we seek to reduce their impact on the accuracy of pose parameters by constructing a weight matrix to reject outliers and measure the relative correctness among inliers.

D. Constructing Weight Matrix

The weight matrix W is a $n \times n$ diagonal matrix where n is the number of 2D-3D correspondences. The diagonal element $\omega_k, k = 1, 2, \cdots, n$ is the confidence probability of the kth correspondence.

$$\mathbf{W} = \begin{pmatrix} \omega_1 & & & \\ & \omega_2 & & \\ & & \ddots & \\ & & & \omega_n \end{pmatrix} \tag{6}$$

Since we have obtained several 2D corresponding points in the input image for each 3D vertex, the tasks mainly lie in selecting the exact 2D point and determining the confidence probability for this 2D-3D correspondence. First, for each 3D vertex, we cluster the corresponding 2D points by the method in [31] which can adaptively determine the number of categories and the centers. Then we propose a formula to calculate the confidence probability of each cluster. In this work, we consider three factors that influence the confidence probability of a 2D point cluster:

(i) the correctness of the homography matrices for 2D points;

(ii) the concentration of the 2D points in the cluster;

(iii) the number of 2D points in the cluster.

In terms of the first factor, we define a coefficient α for each subset image. This coefficient is inversely proportional to the reprojection error to measure the correctness of the homography matrix. If we obtain $Q_j, j = 1, 2, \cdots, m$ matching points between the input image and the jth subset image, α_j will be formulated as follows:

$$\alpha_j = Q_j \Big/ \sum_{q=1}^{Q_j} \sqrt{(\widehat{x}_{jq} - x_{jq}{}')^2 + (\widehat{y}_{jq} - y_{jq}{}')^2}, \quad j = 1, 2, \cdots, m \tag{7}$$

where $(\widehat{x}_{jq}, \widehat{y}_{jq})^T$ is the matching point in the input image, and $(x_{jq}{}', y_{jq}{}')^T$ is calculated by equation (5). Normalization is indispensable to make α_j play the role of relative correctness.

$$\alpha_j = \frac{\alpha_j}{\max\limits_{1 \le j \le m} \alpha_j}, j = 1, 2, \cdots, m \tag{8}$$

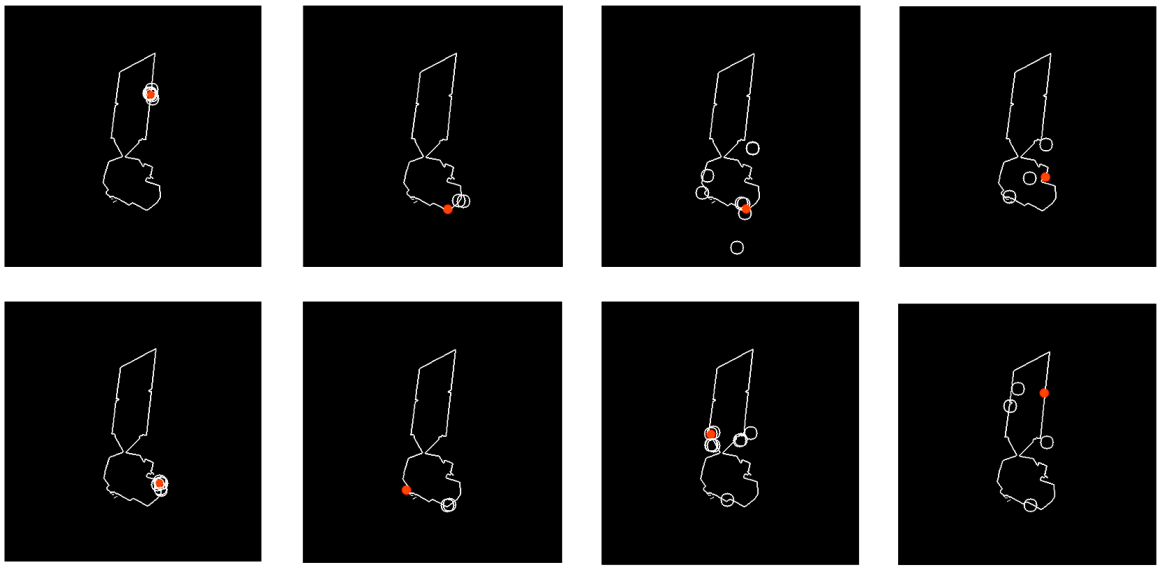
Each corresponding 2D point in the input image shares the same relative correctness with the projection image through which the point is calculated. The maximal α_j of the 2D points in a cluster is selected as the correctness of the cluster.

The concentration of a cluster is measured by the average distance from the 2D points to the cluster center:

$$\beta_j = \begin{cases} 0, & if\ n_j = 1 \\ 1 - \dfrac{\sum\limits_{i=1}^{n_j} ||\mathbf{p}_i - \mathbf{p}_{center}||_2}{n_j} \Big/ radius, & n_j > 1 \end{cases} \quad j = 1, 2, \cdots, N_{cluster} \tag{9}$$

where $n_j, j = 1, 2, \cdots, N_{cluster}$ is the number of 2D points in the jth cluster, and $N_{cluster}$ is the number of clusters. $radius$ is a hyperparameter representing the maximal range of a cluster, which is set to 20 pixels in our implementation. We set the concentration value of isolated points to zero, which can be interpreted in the way that the closest 2D point lies beyond the cluster range.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems



(a) concentrated and right (b) concentrated but wrong (c) scattered and including right matchings (d) scattered but no right matchings

Fig. 5. Four typical distributions of 2D points corresponding to one vertex. The white circles represent calculated corresponding 2D points in the input image, and the red points are the ground truth positions of the 3D vertices.

The third factor representing the number of points in the cluster is directly calculated as follows:

$$\gamma_j = \frac{n_j}{\sum_{j=1}^{N_{cluster}} n_j}, j = 1, 2, \cdots, N_{cluster} \tag{10}$$

Now the three factors α, β and γ are all in the range of $[0, 1]$, and we multiply them to represent the confidence probabilities of the clusters. Logarithmic function is utilized to balance the effects of the three factors.

$$\begin{aligned} &\omega_j = \alpha_j \times \ln(e - 1 + \beta_j) \times \gamma_j, \\ &j = 1, 2, \cdots, N_{cluster} \end{aligned} \tag{11}$$

For each cluster we calculate a confidence probability ω_j, then we select the maximal ω_j as the diagonal element in the weight matrix W.

$$\widehat{\omega}_k = \max_{1 \le j \le N_{cluster}} \omega_j, k = 1, 2, \cdots, n \tag{12}$$

The input image point corresponding to the 3D vertex is calculated as the weighted average of 2D points in the cluster with the maximal ω. Then we adjust the calculated 2D point to the nearest contour point measured by Euclidean distance.

The confidence probability defined above contributes to distinguish the typical distributions of 2D points manifested in Fig. 5. For cases (a) and (c), the values of α should be much larger than those in cases (b) and (d). Depending on the second and the third factors, we can select the cluster including the right corresponding 2D point in case (c). To reject outliers, we set a threshold δ which defines the minimal confidence probability for inlier correspondences.

$$\widehat{\omega}_k = \begin{cases} \widehat{\omega}_k, & \widehat{\omega}_k \ge \delta \\ 0, & \widehat{\omega}_k < \delta \end{cases} \tag{13}$$

For each many-to-one 2D-3D correspondence we repeat the process above, and finally we achieve one-to-one 2D-3D correspondences and a weight matrix W representing confidence probabilities which will be used to guide subsequent PnP algorithm.

It should be noticed that most monocular pose estimation algorithms divide the whole process into two independent parts, i.e. establishing 2D-3D correspondences and calculating pose parameters. They simply rely on RANSAC and the robustness of PnP algorithms to reject outlier correspondences. In contrast, we seek to explore geometric factors in estalishing 2D-3D correspondences to guide pose calculation. We propose an innovative strategy of calculating confidence probabilities to reject outliers without inefficient RANSAC. Further, we measure the relative correctness among inliers to improve pose accuracy, which can neither be achieved by the PnP solutions combined with RANSAC nor REPPnP.

E. Iterative Framework

Proposed approach relies on a iterative framework to improve correspondence correctness and pose accuracy gradually. We refer to the OI algorithm [22] which is fast and numerically precise, and integrate it with the weight matrix W to solve the PnP problem. Due to the introduction of W, the minimization objective function has become:

$$E(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^{n} \omega_i ||(\mathbf{I} - \widehat{\mathbf{V}}_i)(\mathbf{R}\mathbf{x}_i^o + \mathbf{t})||_2^2 \tag{14}$$

where

$$\widehat{\mathbf{V}}_i = \frac{\begin{pmatrix} \mathbf{x}_i^p \\ 1 \end{pmatrix} \begin{pmatrix} \mathbf{x}_i^p \\ 1 \end{pmatrix}^T}{\begin{pmatrix} \mathbf{x}_i^p \\ 1 \end{pmatrix}^T \begin{pmatrix} \mathbf{x}_i^p \\ 1 \end{pmatrix}} \tag{15}$$

represents the line-of-sight projection matrix.

As a result, the translation vector is formulated as follows:

$$\mathbf{t}(\mathbf{R}) = (\sum_{i=1}^{n} \omega_i(\mathbf{I} - \widehat{\mathbf{V}}_i))^{-1} \sum_{i=1}^{n} \omega_i(\widehat{\mathbf{V}}_i - \mathbf{I})\mathbf{R}\mathbf{x}_i^o \tag{16}$$

while $\mathbf{R}$ could be calculated iteratively. Assuming that $\mathbf{R}^{(k)}$ has been achieved, we define

$$\mathbf{y}_i(\mathbf{R}^{(k)}) \stackrel{def}{=} \widehat{\mathbf{V}}_i(\mathbf{R}^{(k)}\mathbf{x}_i^o + \mathbf{t}(\mathbf{R}^{(k)})) \tag{17}$$

$$\overline{\mathbf{y}}(\mathbf{R}^{(k)}) \stackrel{def}{=} \frac{1}{\sum_{i=1}^{n} \omega_i} \sum_{i=1}^{n} \omega_i \mathbf{y}_i(\mathbf{R}^{(k)}) \tag{18}$$

$$\overline{\mathbf{x}}^o \stackrel{def}{=} \frac{1}{\sum_{i=1}^{n} \omega_i} \sum_{i=1}^{n} \omega_i \mathbf{x}_i^o \tag{19}$$

and $\mathbf{M}$ plays the role of the cross-covariance matrix:

$$\mathbf{M}(\mathbf{R}^{(k)}) \stackrel{def}{=} \sum_{i=1}^{n} \omega_i(\mathbf{y}_i - \overline{\mathbf{y}})(\mathbf{x}_i^o - \overline{\mathbf{x}}^o)^T \tag{20}$$

$\mathbf{R}^{(k+1)}$ is formulated as follow according to [22]:

$$svd(\mathbf{M}(\mathbf{R}^{(k)})) = \mathbf{U}\sum\mathbf{V}^T \tag{21}$$

$$\mathbf{R}^{(k+1)} = \mathbf{U}\mathbf{V}^T \tag{22}$$

In proposed algorithm we combine the confidence probabilities of the 2D-3D correspondences and the OI algorithm to estimate pose parameters without RANSAC. It is worth mentioning that the original OI algorithm is sensitive to local minima. In practical applications, it always demands an exploration of the solution space to find optimal solutions. By weighting the 2D-3D correspondences in the original OI algorithm, we introduce strong priori of reliable correspondences to restrict the pose solution space, which may help to avoid local minima.

We project the 3D model from the estimated pose to obtain a new projection image and calculate Intersection over Union (IoU) score between the projection image and the input image. If it is closer to the input image than the subset images, we will replace the least similar subset image with the new projection image. In next iteration, we update both the 2D-3D correspondences and the weight matrix, then repeat the calculations above. Otherwise we just output $\mathbf{R}$ and $\mathbf{t}$ which project the most similar image as final pose results. The detailed termination strategy is shown in Algorithm 1.

Algorithm 1 Termination strategy.

Notation: θ: 6-DoF pose parameters; v_{min}: the minimal IoU score between the input image and the subset images; p: "patience", the number of times to observe lower IoU before stop; s_{max}: The maximal IoU of the calculated projection images;

Output: θ^*: Estimated pose parameters;

```
initial i = 0 and s_max = 0;
while i < p do
    calculate θ using weighted OI algorithm;
    s ← IoU(new projection image of θ)
    if s < v_min then
        stop;
    else
        replace the least similar subset image with the new projection image;
        update v_min;
        if s > s_max then
            i ← 0;
            s_max ← s;
            θ* ← θ;
        else
            i ← i + 1;
        end if
    end if
end while
```

IV. Experiment

In this section, we evaluate the performance of proposed pose estimation method in terms of accuracy and robustness. All the codes involved in our algorithm are implemented in C++ and run on a PC with 3.4 GHz CPU and 16 GB RAM. We use three typical textureless space object models for test: A2100, Earth Observing-1 (EO1) and TianGong, which are shown in Fig. 6. An ideal perspective camera is simulated by OpenGL with image size 400×400 pixels and focal length 200 pixels. As mentioned in $Problem\ Formulation$, the image plane locates at $Z^c = 1$, and we set the origins of the space object models at $(0, 0, 20)$ in the camera coordinate frame. The input images in our experiments are generated by simulation with random pose parameters. The three Euler angles are in the range of $[-90°, 90°]$, and the deviations of translation in x, y, z directions are in the range of $[-5, 5]$. The simulated images may not be exactly the same as those acquired by optical sensors in space. However, the experimental results on simulated data can still validate the capability of proposed approach on real-world data to some extent since we only utilize shape and contour information.

The rotation error and translation error are defined as in [17]:

$$E_r = ||quat(\mathbf{R}) - quat(\mathbf{R}_{true})||/||quat(\mathbf{R}_{true})|| \tag{23}$$

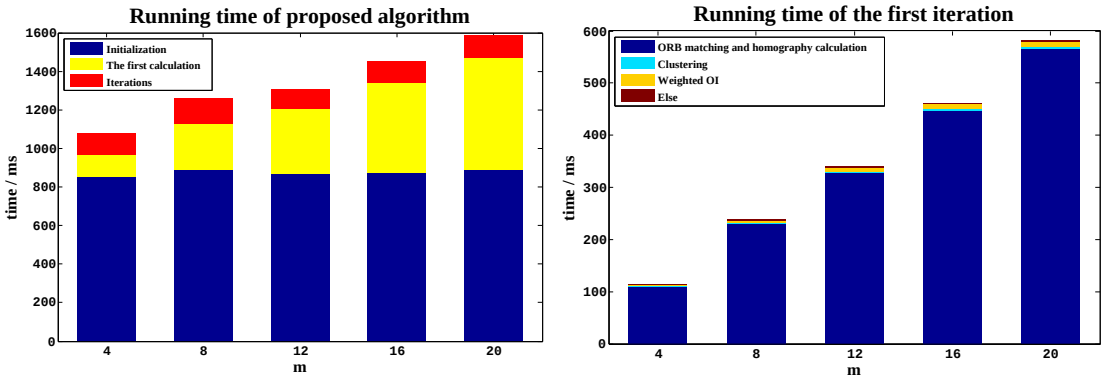
$$E_t = ||\mathbf{t} - \mathbf{t}_{true}||/||\mathbf{t}_{true}|| \tag{24}$$

in which the rotations are presented in the form of quaternions for concision. $\mathbf{R}_{true}$ and $\mathbf{t}_{true}$ denote the ground truth rotation and translation, respectively, and $\mathbf{R}$, $\mathbf{t}$ the estimated pose parameters.

A. Selecting Similar Image Subset

The number of images in the similar subset is an important parameter in the coarse step. If m is too small, we could not guarantee that the subset contains at least one image which is close enough to the input image. Meanwhile, a too large m may introduce unreliable projection images to the subset and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems



(a) Running time of proposed algorithm (b) Running time of the first iteration

Fig. 7. Running time of proposed algorithm and the first iteration at different m.

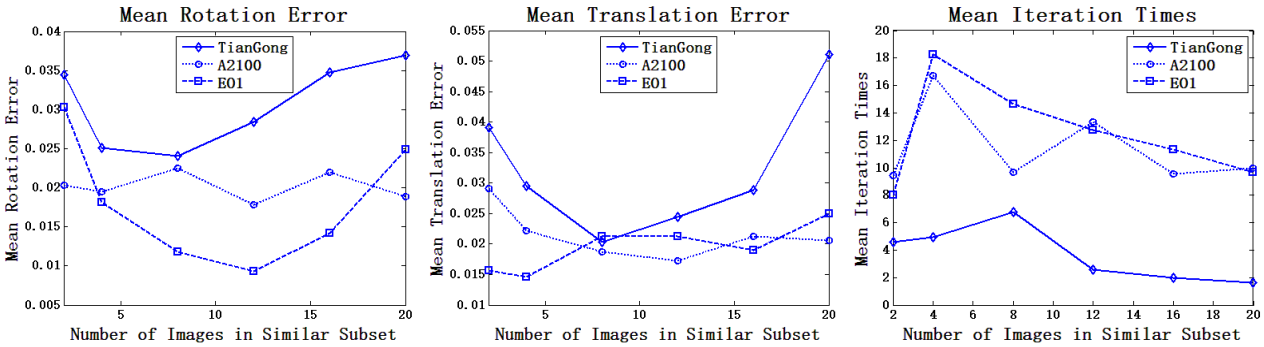
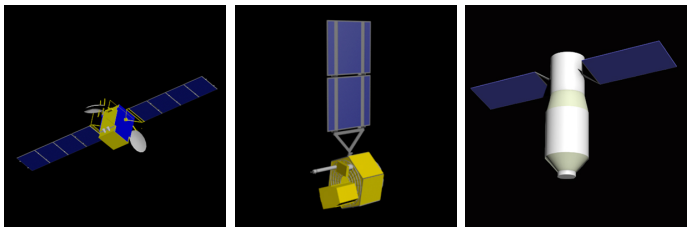


Fig. 8. The impact of m on the accuracy of pose parameters



(a) A2100 (b) EO1 (c) TianGong

Fig. 6. 3D models used in our experiments. The number of vertices are 2749, 6536 and 5736 respectively.

increase computational load. The running time of proposed algorithm is analyzed in Fig. 7. In initialization, we calculate Hu moments for 3042 binary projection images, preprocess the input image, and select the similar subset containing m projection images. In the first calculation, we preform ORB matching between the input image and m subset images. In latter iterations, we just need to update the ORB matching results because only one subset image is replaced in each iteration. As shown in Fig. 7a, the running time of proposed algorithm increases as m grows, which is mainly caused by the first calculation. In Fig. 7b, we can see that ORB matching takes most of the running time of the first calculation.

We identify the impact of m on the accuracy of pose parameters in Fig. 8. The value of m is sampled from 2 to 20, and each point is an average of 50 independent tests. We can see in Fig. 8 that when the value of m is between 8 and 12, the mean rotation and translation errors of the three models remain at a relative low level. Hence, considering the computational cost and pose accuracy, we fix the value of m at 12 in subsequent experiments.

B. Accuracy

We compare proposed approach with Leng [13], Zhang [14] and state-of-the-art PnP solutions combined with RANSAC. The idea of using contour feature is inspired by [13], although we have improved much based on it. Leng et.al resort to distance map to establish 2D-3D correspondences which is heavily dependent on a good initialization. Therefore, we select the most similar subset image as the initialization for [13] and [14]. The comparison results are presented in Table I. Our approach converges to the ground truth of pose parameters faster owing to the capability of ORB feature for contour matching and the strategy utilizing geometric factors to reject outliers. The errors of three Euler angles achieved by our approach are less than 1° and the translation error is about 1%. The average iteration times of proposed approach is almost half of those of [13] and [14], nevertheless the average running time is only slightly reduced, which implies that the efficiency of our approach still needs to be improved. In addition, Leng et.al uses all the contour pixels to establish correspondences, while we select just some accurate ones as in [14], hence the number of correspondences of our algorithm is much smaller.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems

TABLE I
COMPARISON AMONG OUR ALGORITHM, [13] AND [14] IN ACCURACY AND EFFICIENCY

	Leng [13]	Zhang [14]	Ours
Average iteration times	33.714	44.714	**19.429**
Running time(s)	1.467	1.311	**1.115**
Error of pitch($degree$)	0.693	**0.352**	0.411
Error of yaw($degree$)	1.033	0.886	**0.653**
Error of roll($degree$)	1.540	0.301	**0.244**
Er	0.02062	0.00944	**0.00788**
Et	0.02378	**0.00541**	0.01327
Number of correspondences	613.33	60.45	60.45

TABLE II
COMPARISON OF THREE STRATEGIES FOR WEIGHTING 2D-3D CORRESPONDENCES.

	Weighted OI	Binary weighted OI	OI
Error of pitch($degree$)	**0.644**	1.545	21.025
Error of yaw($degree$)	**0.963**	1.465	14.641
Error of roll($degree$)	**0.471**	0.683	6.926
Er	**0.01196**	0.02197	0.24805
Et	**0.01466**	0.02326	0.20249

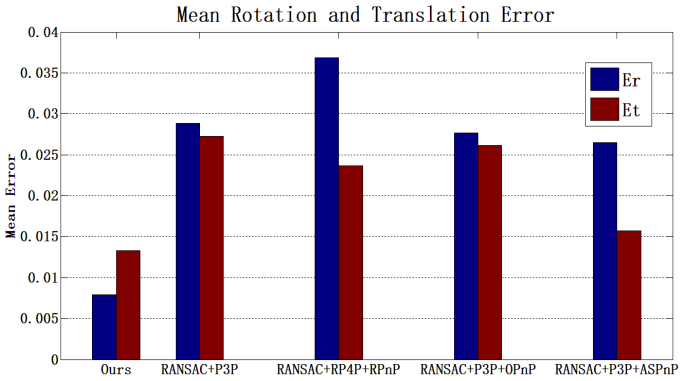


Fig. 9. Comparison in terms of accuracy of our algorithm with state-of-the-art PnP solutions combined with RANSAC

One main contribution of proposed approach is that we explore the geometric factors in establishing 2D-3D correspondences to guide pose calculation. We design experiments to validate the effectiveness of the weight matrix in rejecting outliers and improving accuracy. In Table II, we compare three different strategies for constructing the weight matrix, namely weighted OI (proposed), binary weighted OI (proposed), and the original OI algorithm. For the original OI, the weights are all set to 1. For the binary weighted OI, the weights for inliers are all set to 1. The pose errors of the original OI are far larger than those of weighted OI and binary weighted OI, which demonstrates the effectiveness of proposed weight matrix in outlier rejection. By setting a threshold to the confidence probabilities of 2D-3D correspondences, we can reject most outliers without inefficient RANSAC. Further, the weighted OI strategy performs better than the binary weighted OI strategy, indicating that measuring the relative correctness among inliers also helps to improve pose accuracy.

We select several state-of-the-art PnP algorithms combined with RANSAC: (RANSAC+P3P [24]); (RANSAC + RP4P + RPnP [25]); (RANSAC + P3P [24] + ASPnP [26]); and (RANSAC + P3P [24] + OPnP [27]) , to deal with the 2D-3D

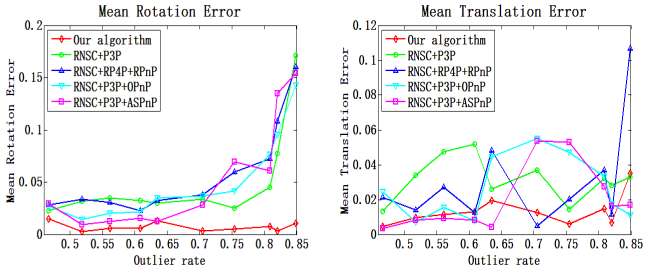


Fig. 10. The rotation and translation error in different conditions of outlier rate.

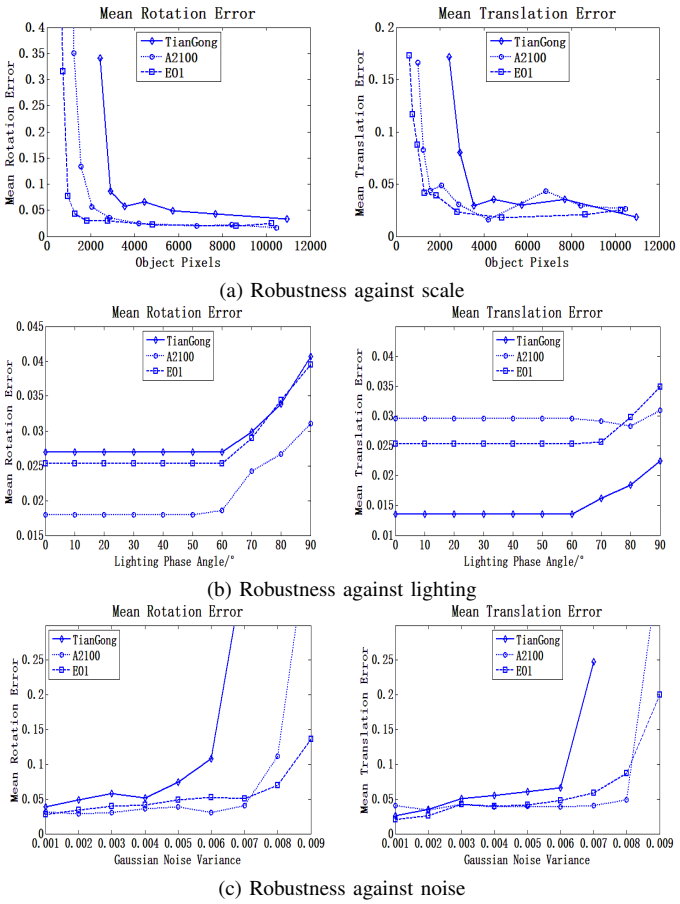
correspondences extracted by our algorithm. As illustrated in Fig. 9, proposed algorithm performs best in accuracy. Since all the methods deal with the same 2D-3D correspondences, we can reasonably attribute the capability to the weight matrix constructed with the geometric factors in establishing 2D-3D correspondences. Proposed algorithm has two advantages over the RANSAC-based PnP solutions. Firstly, we reject outliers efficiently without RANSAC. Secondly, we measure the relative correctness among inliers to improve pose accuracy, which cannot be achieved by the PnP solutions combined with RANSAC. We have noticed REPPnP [23] which embeds the outlier rejection scheme within the pose estimation pipeline, nevertheless it does not perform well on our data. We suspect that the reason mainly lies in that REPPnP is designed for a large number of correspondences and the outlier rate of our 2D-3D correspondences is beyond its breakdown point.

In Fig. 10 we analyze the robustness of proposed algorithm against outliers. We plot the rotation and translation errors with reference to the outlier rate of the 2D-3D correspondences. The outlier rate of our data varies from 46.67% to 84.91%, and it is far higher than the breakdown point of REPPnP, which is no more than 60% according to [23]. We can see in the figure that the state-of-the-art PnP solutions combined with RANSAC begin to fail at the outlier rate of 75 percent, while proposed algorithm is still stable and accurate. These results convincingly demonstrate that the confidence probabilities are effective in distinguishing inliers and outliers. It is worth mentioning that the confidence probabilities are continuous coefficients other than binary values, which can further indicate the relative correctness of the inlier 2D-3D correspondences. Hence, by measuring the relative correctness among inliers, we introduce strong priori of reliable correspondences to restrict pose solution space, which may help to avoid local minima and achieve satisfactory pose results than RANSAC-based methods do.

C. Robustness against Imaging Degradation

Aiming at the practical application to solve the pose estimation of space objects, we test the robustness of proposed approach against some typical imaging degradation factors including noise, scale and lighting changes in space environment. First we test the performance of proposed approach in different scales. The imaging scale is measured by the pixels of space objects in the input images. The number of pixels occupied by space objects varies from about 10000 to less

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems



(a) Robustness against scale

(b) Robustness against lighting

(c) Robustness against noise

Fig. 11. Robustness tests of our approach against imaging degradation.

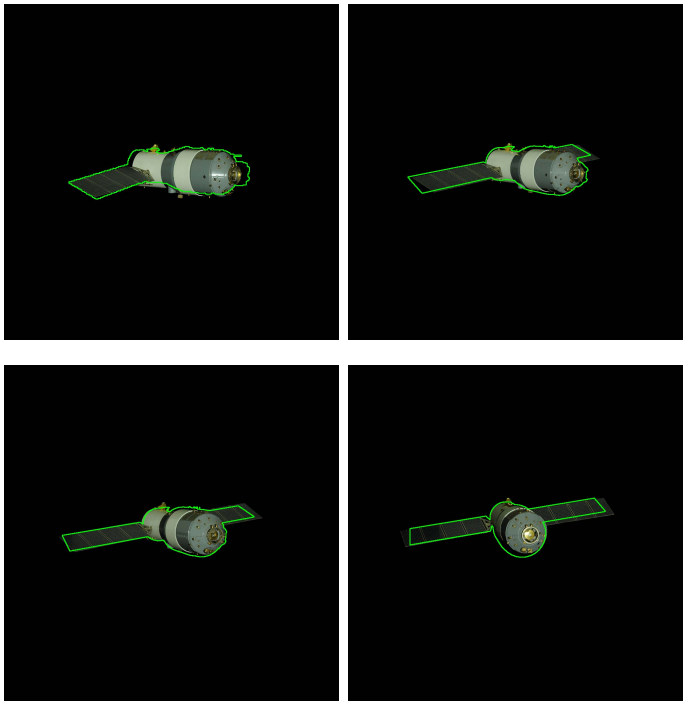


Fig. 12. Validation of proposed algorithm on real-world images of TianGong. The green curves are contours of calculated projection images.

than 2000 in our experiments. Results in Fig. 11a indicate that our approach is capable of dealing with input images with no less than 4000 pixels occupied by space objects. When the number of space object pixels descends to less than 3000, the error of pose parameters increases sharply. The reason could be that small objects reduce the precision in contour extraction and feature matching. A preliminary process involving super resolution reconstruction may be helpful; however, that is not the focus of this paper.

The lighting condition is formulated as in [1], and the lighting phase angle should not exceed $90°$ so that the space objects can be visible. The lighting phase angle is sampled in the range of $[0°, 90°]$ at intervals of $10°$. The lighting altitude angle is set to be $0°$, $60°$, $120°$ and $180°$. Since our approach only utilizes the shape and contour information of the input images, the changes in lighting condition have little impact on the performance of our approach as long as the complete contours can be extracted. In Fig. 11b, our approach is illustrated to be stable when the lighting phase angle is less than $60°$.

Zero-mean Gaussian white noise is added to the input images for noise tests. The variance is sampled from 0.001 to 0.009 at intervals of 0.001. Theoretically, the introduction of noise could largely influence the performance of contour extraction and feature matching. As a consequence, the errors of rotation parameters achieved by proposed approach increase remarkably when the variance of Gaussian white noise is bigger than 0.004. The breakdown point of translation against noise is at about 0.006. Generally speaking, noise in the input data, no matter 2D images or 3D point clouds, is a problem for most existing pose estimation algorithms, and a preliminary process of noise reduction is indispensable for practical application.

D. Validation on Real-World Data

As far as we know, there hardly exist public datasets of real space object images. Most relative researches are based on simulated images or point clouds. Limited by the access to real space object images, we evaluate our proposed algorithm on simulated data. However, we add noise, scale and illumination variations to the simulated images and validate the robustness of proposed algorithm to some extent. In addition, we take pictures of a solid model of TianGong spacecraft in a darkroom. We use a parallel light to simulate the sun, and we calibrate the camera intrinsic matrix by ourselves. We validate the effectiveness of proposed algorithm on these real-world images as shown in Fig. 12. Note that there exist some differences between the elaborate solid model and the simplified CAD model, both in terms of shape and size. This could lead to an increase of pose errors in our real-world experiments.

V. Conclusion

In this paper, we have proposed a monocular vision-based method to solve the 6-DoF pose estimation problem of textureless space objects. Proposed approach is a complete process independent of initialization, and has no constraints on the shape of space objects. We explore geometric factors

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2018.2815879, IEEE Transactions on Aerospace and Electronic Systems

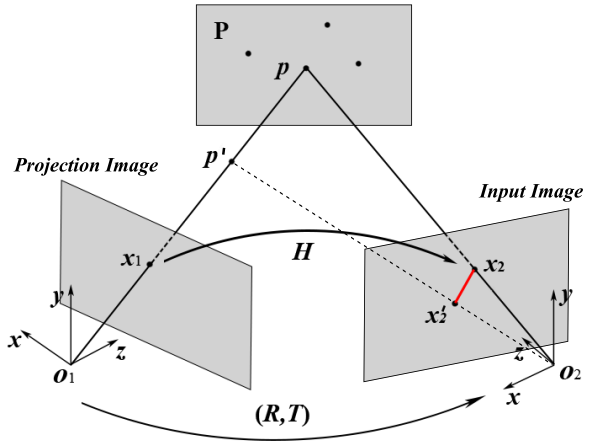


Fig. 13. Illustration of pixel coordinate errors (the red line) caused by homography matrix when 3D vertices are non-coplanar.

in establishing 2D-3D correspondences by constructing the weight matrix to guide pose calculation, which has not been attempted to our knowledge. Apart from rejecting outliers efficiently, we can further measure the relativeness correctness among inliers which cannot be achieved by RANSAC-based PnP solutions. We have demonstrated the effectiveness of proposed approach for pose estimation on simulated data and real-world data. Experimental results verify that our method can accurately estimate the pose of textureless space objects within a few iterations even in the heavy outlier situations. Proposed method can be applied to fundamental missions in space surveillance systems such as satellite tracking and on-orbit servicing.

APPENDIX
PIXEL COORDINATE ERRORS CAUSED BY HOMOGRAPHY MATRIX

As illustrated in Fig.13, we can calculate the homography matrix H between two images of plane P. x_1 is a projection image point corresponding to p', a 3D vertex outside plane P. The true projection of p' on the input image is x_2', and the calculated point using homography matrix is x_2. The 3D coordinates of p and p' in the camera frame $o_1 - xyz$ are:

$$\mathbf{p_1} = d\mathbf{x_1} = \begin{bmatrix} dx_{1x} \\ dx_{1y} \\ d \end{bmatrix}, \quad \mathbf{p_1'} = d'\mathbf{x_1} = \begin{bmatrix} d'x_{1x} \\ d'x_{1y} \\ d' \end{bmatrix} \tag{25}$$

where $\mathbf{x_1}$ is the homogeneous coordinate of image point x_1, d and d' are constants representing the imaging depths. The 3D coordinates of p and p' in the camera frame $o_2 - xyz$ are:

$$\begin{aligned} \mathbf{p_2} &= \mathbf{R}\mathbf{p_1} + \mathbf{T} = d\mathbf{R}\mathbf{x_1} + \mathbf{T} \\ &= d \begin{bmatrix} r_{11}\ r_{12}\ r_{13} \\ r_{21}\ r_{22}\ r_{23} \\ r_{31}\ r_{32}\ r_{33} \end{bmatrix} \begin{bmatrix} x_{1x} \\ x_{1y} \\ 1 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \\ &= \begin{bmatrix} d(r_{11}x_{1x} + r_{12}x_{1y} + r_{13}) + t_1 \\ d(r_{21}x_{1x} + r_{22}x_{1y} + r_{23}) + t_2 \\ d(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3 \end{bmatrix} \end{aligned} \tag{26}$$

$$\begin{aligned} \mathbf{p_2'} &= \mathbf{R}\mathbf{p_1'} + \mathbf{T} = d'\mathbf{R}\mathbf{x_1} + \mathbf{T} \\ &= d' \begin{bmatrix} r_{11}\ r_{12}\ r_{13} \\ r_{21}\ r_{22}\ r_{23} \\ r_{31}\ r_{32}\ r_{33} \end{bmatrix} \begin{bmatrix} x_{1x} \\ x_{1y} \\ 1 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \\ &= \begin{bmatrix} d'(r_{11}x_{1x} + r_{12}x_{1y} + r_{13}) + t_1 \\ d'(r_{21}x_{1x} + r_{22}x_{1y} + r_{23}) + t_2 \\ d'(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3 \end{bmatrix} \end{aligned} \tag{27}$$

The homogeneous coordinates of x_2 and x_2' are:

$$\begin{aligned} \mathbf{x_2} &= \begin{bmatrix} \frac{d(r_{11}x_{1x} + r_{12}x_{1y} + r_{13}) + t_1}{d(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} \\ \frac{d(r_{21}x_{1x} + r_{22}x_{1y} + r_{23}) + t_2}{d(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} \\ 1 \end{bmatrix}, \\ \mathbf{x_2'} &= \begin{bmatrix} \frac{d'(r_{11}x_{1x} + r_{12}x_{1y} + r_{13}) + t_1}{d'(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} \\ \frac{d'(r_{21}x_{1x} + r_{22}x_{1y} + r_{23}) + t_2}{d'(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} \\ 1 \end{bmatrix} \end{aligned} \tag{28}$$

The image coordinate error caused by using homography matrix is:

$$\begin{aligned} x_2x_2' &= \begin{bmatrix} \frac{d'(r_{11}x_{1x} + r_{12}x_{1y} + r_{13}) + t_1}{d'(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} - \frac{d(r_{11}x_{1x} + r_{12}x_{1y} + r_{13}) + t_1}{d(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} \\ \frac{d'(r_{21}x_{1x} + r_{22}x_{1y} + r_{23}) + t_2}{d'(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} - \frac{d(r_{21}x_{1x} + r_{22}x_{1y} + r_{23}) + t_2}{d(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3} \end{bmatrix} \\ &= \begin{bmatrix} \frac{(d'-d)[(r_{11}x_{1x} + r_{12}x_{1y} + r_{13})t_3 + (r_{31}x_{1x} + r_{32}x_{1y} + r_{33})t_1]}{[d(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3][d'(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3]} \\ \frac{(d'-d)[(r_{21}x_{1x} + r_{22}x_{1y} + r_{23})t_3 + (r_{31}x_{1x} + r_{32}x_{1y} + r_{33})t_2]}{[d(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3][d'(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + t_3]} \end{bmatrix} \\ &= \begin{bmatrix} \frac{(1 - \frac{d}{d'})[(r_{11}x_{1x} + r_{12}x_{1y} + r_{13})\frac{t_3}{d} + (r_{31}x_{1x} + r_{32}x_{1y} + r_{33})\frac{t_1}{d}]}{[(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + \frac{t_3}{d}][(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + \frac{t_3}{d'}]} \\ \frac{(1 - \frac{d}{d'})[(r_{21}x_{1x} + r_{22}x_{1y} + r_{23})\frac{t_3}{d} + (r_{31}x_{1x} + r_{32}x_{1y} + r_{33})\frac{t_2}{d}]}{[(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + \frac{t_3}{d}][(r_{31}x_{1x} + r_{32}x_{1y} + r_{33}) + \frac{t_3}{d'}]} \end{bmatrix} \end{aligned} \tag{29}$$

Note that the rows of R are unit vectors and will not change the length of $\mathbf{x_1}$. Therefore, when the elements of T are far smaller than the imaging distance d, the pixel coordinate errors caused by homography matrix will be close to zero.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61501009, 61771031 and 61371134), the National Key Research and Development Program of China (2016YFB0501300, 2016YFB0501302) and the Aerospace Science and Technology Innovation Fund of CASC (China Aerospace Science and Technology Corporation).

REFERENCES

[1] Zhang H, Jiang Z, Elgammal A. Satellite recognition and pose estimation using homeomorphic manifold analysis. IEEE Transactions on Aerospace and Electronic Systems, 2015, 51(1):785-792.
[2] Kennewell J, Vo B. An overview of space situational awareness. International Conference on Information Fusion. 2013:1029-1036.
[3] Space based space surveillance (SBSS). [Online]. Available: http://www.globalsecurity.org/space/systems/sbss.htm.
[4] NEOSSat: Canada's Sentinel in the Sky. http://www.asc-csa.gc.ca/eng/satellites/neossat.
[5] Zimmer P, McGraw J, Ackermann M. Affordable Wide-field Optical Space Surveillance using sCMOS and GPUs. Advanced Maui Optical and Space Surveillance Technologies Conference, 2016.
[6] Duncan A, Kendrick R, Thurman S, et al. SPIDER: Next Generation Chip Scale Imaging Sensor. Advanced Maui Optical and Space Surveillance Technologies Conference, 2015.

[7] Petit A, Marchand E, Kanani K. Vision-based space autonomous rendezvous: A case study. IEEE International Conference on Intelligent Robots and Systems, 2011:619-624.
[8] Aghili F, Kuryllo M, Okouneva G, et al. Fault-tolerant pose estimation of space objects. IEEE International Conference on Advanced Intelligent Mechatronics, 2010:947-954.
[9] Opromolla R, Fasano G, Rufino G, et al. Pose Estimation for Spacecraft Relative Navigation Using Model-based Algorithms. IEEE Transactions on Aerospace and Electronic Systems, 2017.
[10] Liu, C, Hu, W. Relative pose estimation for cylinder-shaped spacecrafts using single image. IEEE Transaction on Aerospace and Electronics Systems, 50(4), pp. 3036-3056, Oct. 2014.
[11] Zhang H, Jiang Z, Yao Y, et al. Vision-based pose estimation for space objects by Gaussian process regression. Aerospace Conference, 2015:1-9.
[12] Zhang H, Jiang Z, Elgammal A. Vision-based pose estimation for cooperative space objects. Acta Astronautica, 2013, 91(10):115C122.
[13] Leng D, Sun W. Contour-based iterative pose estimation of 3D rigid object. IET Computer Vision, 2011, 5, (5), pp. 291-300.
[14] Zhang X, Zhang H, Wei Q, et al. Pose Estimation of Space Objects Based on Hybrid Feature Matching of Contour Points. Advances in Image and Graphics Technologies. 2016.
[15] Flores-Abad A, Ma O, Pham K, et al. A review of space robotics technologies for on-orbit servicing. Progress in Aerospace Sciences, 2014, 68(8):1-26.
[16] Zhang G, Liu H, Wang J, et al. Vision-Based System for Satellite On-Orbit Self-Servicing. 2008:296-301.
[17] Rubio A, Villamizar M, Ferraz L, et al. Efficient monocular pose estimation for complex 3D models. IEEE International Conference on Robotics and Automation, 2015, 2015:1397-1402.
[18] Li S. Absolute pose estimation using multiple forms of correspondence from RGB-D frames. IEEE Intl. Conf. on Robotics and Automation, 2016.
[19] Cao Z, Sheikh Y, Banerjee N K. Real-time scalable 6DOF pose estimation for textureless objects. IEEE International Conference on Robotics and Automation, 2016:2441-2448.
[20] Hu W, Zhu S C. Learning a probabilistic model mixing 3D and 2D primitives for view invariant object recognition. IEEE Conference on Computer Vision and Pattern Recognition, 2010:2273-2280.
[21] Iwashita Y, et al. 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.
[22] Lu C, Hager G, 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.
[23] Ferraz L, Binefa X, Morenonoguer F. Very Fast Solution to the PnP Problem with Algebraic Outlier Rejection. IEEE Conference on Computer Vision and Pattern Recognition, 2014:501-508.
[24] Kneip L, Scaramuzza D, Siegwart R. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. IEEE Conference on Computer Vision and Pattern Recognition, 2011, 42(7):2969-2976.
[25] Li S, Xu C, Xie M. A Robust O(n) Solution to the Perspective-n-Point Problem. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(7):1444-1450.
[26] Zheng Y, Sugimoto S, Okutomi M. ASPnP: An Accurate and Scalable Solution to the Perspective-n-Point Problem. IEICE Transactions on Information and Systems, 2013, E96.D(7):1525-1535.
[27] Zheng Y, Kuang Y, Sugimoto S, et al. Revisiting the PnP Problem: A Fast, General and Optimal Solution. IEEE International Conference on Computer Vision, 2013:2344-2351.
[28] Xu C, Zhang L, Cheng L, et al. Pose Estimation from Line Correspondences: A Complete Analysis and A Series of Solutions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016:1-1.
[29] Rublee E, Rabaud V, Konolige K, et al. ORB: An efficient alternative to SIFT or SURF. IEEE International Conference on Computer Vision, 2011:2564-2571.
[30] Hu M. Visual pattern recognition by moment invariants. Ire Transactions on Information Theory, 1962, 8(2):179-187.
[31] Rodriguez A, Laio A. Clustering by fast search and find of density peaks. Science, 2014, 334(6191):1492-1496.
[32] 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.
[33] Liu H, Zhang G, Bao H. Robust Keyframe-based Monocular SLAM for Augmented Reality. IEEE International Symposium on Mixed and Augmented Reality. 2016:1-10.
[34] Mur-Artal R, Montiel M, Juan T. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. IEEE Transactions on Robotics, 2015, 31,5, pp.1147-1163.
[35] Fischler M, Bolles R. Random Sample Consensus: A Paradigm for Model Fitting with Applications To Image Analysis and Automated Cartography. Communications of the Acm, 1980, 24(6):381-395.
[36] Lowe D. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 2004, 60, (2), pp. 91-110.
[37] Bay H, Tuytelaars T, Gool L V. SURF: Speeded Up Robust Features. Computer Vision and Image Understanding, 2006, 110(3):404-417.

Xin Zhang received his B.Eng. degree from Beihang University, Beijing, China, in 2016. He is currently pursuing his Ph.D. degree in the Image Processing Center, School of Astronautics, Beihang University. His research interests include computer vision, pattern recognition, and machine learning.

Zhiguo Jiang is a professor at Beihang University, and has been the Vice Dean of the School of Astronautics at Beihang University since 2006. He currently serves as a standing member of the Executive Council of China Society of Image and Graphics and also serves as a member of the Executive Council of Chinese Society of Astronautics. He is an Editor for the Chinese Journal of Stereology and Image Analysis. His current research interests include remote sensing image analysis, target detection, tracking and recognition, and medical image processing.

Haopeng Zhang received his B.Eng. and Ph.D. degrees from Beihang University in 2008 and 2014, respectively. He is currently an assistant professor at the Image Processing Center, School of Astronautics, Beihang University, China. He is a member of IEEE. His main research interests include multi-view object recognition, 3D object recognition and pose estimation, and other related areas in pattern recognition, computer vision, and machine learning.

Quanmao Wei received his B.Eng. degree from Beihang University, Beijing, China, in 2015. He is currently pursuing his Master degree in the Image Processing Center, School of Astronautics, Beihang University. His research interests include computer vision, pattern recognition, and machine learning.