

Research Article

Vision-Based Satellite Recognition and Pose Estimation Using Gaussian Process Regression

Haopeng Zhang^{1,2,3}, Cong Zhang^{1,2,3}, Zhiguo Jiang^{1,2,3}, Yuan Yao^{1,2,3} and Gang Meng⁴

¹Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China

²Key Laboratory of Spacecraft Design Optimization and Dynamic Simulation Technologies, Ministry of Education, Beijing 100191, China

³Beijing Key Laboratory of Digital Media, Beijing 100191, China

⁴Beijing Institute of Remote Sensing Information, Beijing 100192, China

Correspondence should be addressed to Haopeng Zhang; zhanghaopeng@buaa.edu.cn

Received 13 September 2018; Revised 14 March 2019; Accepted 26 March 2019; Published 9 June 2019

Academic Editor: Seid H. Pourtakdoust

Copyright © 2019 Haopeng Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we address the problem of vision-based satellite recognition and pose estimation, which is to recognize the satellite from multiviews and estimate the relative poses using imaging sensors. We propose a vision-based method to solve these two problems using Gaussian process regression (GPR). Assuming that the regression function mapping from the image (or feature) of the target satellite to its category or pose follows a Gaussian process (GP) properly parameterized by a mean function and a covariance function, the predictive equations can be easily obtained by a maximum-likelihood approach when training data are given. These explicit formulations can not only offer the category or estimated pose by the mean value of the predicted output but also give its uncertainty by the variance which makes the predicted result convincing and applicable in practice. Besides, we also introduce a manifold constraint to the output of the GPR model to improve its performance for satellite pose estimation. Extensive experiments are performed on two simulated image datasets containing satellite images of 1D and 2D pose variations, as well as different noises and lighting conditions. Experimental results validate the effectiveness and robustness of our approach.

1. Introduction

Optical imaging sensors have been widely used as the essential payloads of vision systems in aerospace applications: autonomous rendezvous and docking [1–4], vision-based landing [5], position and pose estimation [6–13], on-orbit serving [14, 15], space robotics [16], satellite recognition [12, 17–19], 3D structure reconstruction and component detection [20, 21], etc. Vision-based recognition and pose estimation of a target satellite are one of the key technologies to achieve these applications. The manufacturing technology and performance of the imaging sensors develop rapidly in the past decades. For example, the space-based visible (SBV) sensor [22] can provide images with spatial resolution by possibly reducing the range from the imaging sensor to the target satellite [23]. Recently improved sCMOS sensors [24] can perform affordable, wide-field, and rapid cadence

surveillance from the low Earth orbit (LEO) to out past the geosynchronous orbit (GEO). In addition, a next-generation chip scale imaging sensor SPIDER [25] has been designed to provide higher resolution by enabling a larger-aperture imager in a constrained volume. Owing to the improvement of imaging sensors, image data captured by space-based vision systems can be of higher quality. Such high-quality image data contain more detailed information of the target satellite, which thus could benefit satellite recognition and pose estimation.

Previous vision-based pose estimation methods for space objects can be broadly divided into two classes, i.e., 3D model-based methods and 2D image-based methods. 3D model-based pose estimation requires a prior 3D model of the target space object which contains rich information: structure, shape, textures, and so on. The 3D model could be a CAD model [1, 3, 4, 26, 27] or 3D point cloud [2, 13].

When estimating the poses of the target, 3D models are utilized directly to 2D images or to the 3D data reconstructed by 2D images using stereo vision [28]. In addition, the 3D model can also be used to generate projection images from sampled viewpoints for matching when an input 2D image is given [27]. However, such accurate prior 3D models are usually difficult to be obtained in practice. 2D image-based methods attempt to directly restore pose information from an image sequence or a single image based on binocular (or stereo) vision [8, 9] or monocular vision [6, 14, 29]. Although no prior 3D model is needed, most of existing image-based methods may need camera calibration [8, 29] or optical markers [14] on the target spacecraft. Meanwhile, due to the principle of binocular vision, binocular vision-based approaches may be invalid when the imaging sensors are far from the target spacecraft; i.e., the distance from the target spacecraft to the imaging sensors is much farther than the distance between the two imaging sensors. Recently, learning-based methods [10–12, 19] have been proposed without the above limitations. Zhang et al. [10] introduced Homeomorphic Manifold Analysis to the aerospace area to estimate relative poses of space objects. Zhang and Jiang [11] and Zhang and Jiang [19] handled spacecraft pose estimation by using kernel regression-based methods. These methods are fundamentally based on supervised learning technology (like kernel regression), and no other requirements but only training image data are needed, which thus can be also regarded as 2D image-based methods.

In the past decade, vision-based satellite recognition or identification is attracting more and more interest [12, 17–19, 30–33]. Some of these methods focus on extracting *good* features to improve recognition performance of traditional classifiers like k-nearest-neighbors, support vector machines, etc. [17] represented the combined features of satellite images in a latent space generated by kernel locality preserving projections. Pan et al. [30] achieved satellite recognition by fusing infrared and visible image features. Ding et al. [18] proposed normalized affine moment invariants and illumination invariant multiscale autoconvolution for autonomous space object identification. Shi et al. [31] encoded satellite image features by elastic net sparse coding and [32] also used sparse coding-based probabilistic latent semantic analysis to get semantic features for satellite recognition. Meanwhile, the other approaches tend to use various machine learning models for a better recognition of the satellites. Zhang and Jiang [19] solved multiview space object recognition by kernel regression, [12] used homeomorphic manifold analysis for satellite recognition, and [33] built a nine-layer deep convolutional neural network DCNN to achieve space target recognition. In particular, [19] and [12] formulated vision-based satellite recognition and pose estimation in one framework.

In this paper, following previous works [12, 19], we also address the problem of vision-based satellite recognition and pose estimation, which is to estimate the relative pose of a target satellite and simultaneously recognize its category using imaging sensors. We develop a novel method based on monocular vision using Gaussian process regression (GPR), which is not only powerful for predicting continuous

quantities but also applicable for discrete values. We make assumption that the regression function mapping from the image (or feature) of the target satellite to its relative pose or category follows a Gaussian process (GP), and then, this GP can be properly parameterized by a mean function and a covariance function. Given the training data, we can easily obtain the explicit formulations of predictive equations by a maximum-likelihood approach, in which the mean value of the predicted output (i.e., the estimated pose or recognized category) and its variance (which indicate the uncertainty) can be computed. Considering the fact that we recognize the category of a target satellite priorly before estimating its pose using GPR, we can solve multiview satellite recognition and pose estimation in one framework via pose-after-category strategy. Besides, as shown in [10, 12, 19], images of one space object with different poses lie on intrinsic low-dimensional manifolds which are homeomorphic to each other, and such homeomorphic manifold can be beneficial for pose estimation. Therefore, we also use normalized n -sphere (i.e., the homeomorphic manifold) in a $(n + 1)$ -dimensional Euclidean space to represent n -degree-of-freedom (n -DoF) pose variation and then learn a multiple-output GPR model for satellite pose estimation. To validate the effectiveness and robustness of our approach, extensive experiments have been performed on a simulated image dataset called BUAA-SID dataset [19], considering multiple complex conditions including 1D and 2D pose variations, image noises, and lighting conditions.

Comparing with previous vision-based satellite recognition and pose estimation methods, the contributions of this paper can be divided into three aspects. Firstly, GPR, a more powerful regression model, is introduced for the task of satellite recognition and pose estimation. GPR has strong statistical foundation and many desirable properties which are suitable for classification and pose estimation. Secondly, a homeomorphic manifold constraint is employed to improve the pose estimation capability of the original GPR model. Our manifold constrained Gaussian process regression (MCGPR) can obviously improve the performance of pose estimation, leading a surprising increase of more than one order of magnitude in some cases. Thirdly, as the GPR model can offer a variance of each output, we defined and calculated the *uncertainty* of the predicted value by its variance. The uncertainty represents the credibility of the predicted value, which can help us choose more convincing results. The uncertainty represents the credibility of the predicted value, which can help us choose more convincing results. In the field of aerospace, identifying the certainty may be more important than the accuracy of the results, so the uncertainty has an important strategic significance. It should be noticed that this paper is an extended version of our paper [34] published in IEEE Aerospace Conference 2015 at Big Sky, MT, USA, where parts of a pose estimation work were previously represented.

The rest of the paper is organized as follows: we detailedly describe the framework of the GPR model for satellite recognition and pose estimation in Section 2. Experimental results and analyses are presented in Section 3. Section 4 concludes the paper.

2. Methodology

2.1. General Gaussian Process Regression Model. Gaussian processes (GPs) have received increasing attention in the field of machine learning in the past few years, for classification or regression [35–40]. Classification and regression are subproblems of supervised learning, which involve the prediction of discrete and continuous quantities, respectively. Since discrete quantities can be seen as samples of continuous ones, we can also perform classification the same as regression. Thus, we can use a regression model for both classification and regression problems. Gaussian process regression (GPR) has many desirable properties, such as ease of obtaining and expressing uncertainty in predictions, the ability to capture a wide variety of behaviour through a simple parametrisation, and a natural Bayesian interpretation [36]. So we try to employ it to solve the problem of classification and pose estimation of space objects. We will describe the framework of GPR briefly in this section.

$\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$ is set as the input training data, where $\mathbf{x}_i \in \mathcal{R}^d$ are the input data (i.e., different kinds of image representations, e.g., the six kinds of image representations used in Section 3 of this paper), \mathcal{R}^d is the d -dimensional Euclidean space, $\mathbf{y}_i \in \mathcal{R}^v$ are the corresponding target values represented in the v -dimensional Euclidean space, and N is the number of training data. For clarity and without loss of generality, we take 1D output for example, i.e., $y_i \in \mathcal{R}$ indicates a certain dimension of $\mathbf{y}_i \in \mathcal{R}^v$. Then, the regression model for the task with noise can be written as $y = f(\mathbf{x}) + \varepsilon$, where f is the regression function mapping from \mathcal{R}^d to \mathcal{R} and noise ε follows an independent, identically distributed Gaussian distribution with zero mean and variance σ_n^2 , i.e., $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$. According to [38], the Gaussian process can be written as

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \quad (1)$$

where the mean function $m(\mathbf{x})$ is taken to be zero and the covariance function $\text{cov}(f(\mathbf{x}), f(\mathbf{x}')) = k(\mathbf{x}, \mathbf{x}')$ is set to be the squared exponential covariance function with isotropic distance measure with unit magnitude [38],

$$k(\mathbf{x}, \mathbf{x}') = \exp \left\{ -(\mathbf{x} - \mathbf{x}')^T (\text{ell}^2 I)^{-1} (\mathbf{x} - \mathbf{x}') \right\}, \quad (2)$$

where \mathbf{x} and \mathbf{x}' are two datapoints, I is the d -order identity matrix, and parameter ell^2 is the characteristic length-scale. Then, in the GPR model, the Gaussian process prior over the function f is [38]

$$\mathbf{f} \sim \mathcal{N}(\mathbf{0}, K(\mathbf{X}, \mathbf{X})), \quad (3)$$

where \mathbf{f} denotes the regression function outputs of the entire training set $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ and $K(\mathbf{X}, \mathbf{X})$ denotes the $N \times N$

matrix of the covariances evaluated at all pairs of training points using equation (2).

Let the testing set be \mathbf{X}_* ; then, under prior equation (3), the joint distribution of the observed target values \mathbf{y} of \mathbf{X} and the regression function outputs \mathbf{f}_* of \mathbf{X}_* can be written as [38]

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N} \left(\mathbf{0}, \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I & K(\mathbf{X}, \mathbf{X}_*) \\ K(\mathbf{X}_*, \mathbf{X}) & K(\mathbf{X}_*, \mathbf{X}_*) \end{bmatrix} \right), \quad (4)$$

and the likelihood is a factorized Gaussian $\mathbf{y}|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma_n^2 I)$. Thus, the predictive equations for the GPR model can be written as [38]

$$\mathbf{f}_*|\mathbf{X}, \mathbf{y}, \mathbf{X}_* \sim \mathcal{N}(\bar{\mathbf{f}}_*, \text{cov}(\mathbf{f}_*)), \quad (5)$$

where

$$\bar{\mathbf{f}}_* = K(\mathbf{X}_*, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I]^{-1} \mathbf{y},$$

$$\text{cov}(\mathbf{f}_*) = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}_*, \mathbf{X}) [K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I]^{-1} K(\mathbf{X}, \mathbf{X}_*). \quad (6)$$

The target value can be represented by the mean values $\bar{\mathbf{f}}_*$, and the variances in $\text{cov}(\mathbf{f}_*)$ can indicate the uncertainties of the target values predicted by the model. The larger the variance is, the less convincing the corresponding estimated pose is. We may reject some estimation results in practice if their variances are too large.

To predict multiple output variables, we can follow the simple approach mentioned in Section 9.1 of [38] in an intuitive way, i.e., modeling each output variable as independent of other variables and treating them separately.

2.2. GPR for Satellite Recognition. When GPR is used for satellite recognition, the target value should be class labels. For a C -class classification problem, we can define the target values as a label vector $\mathbf{y}_i = [0, 0, \dots, 1, \dots, 0]^T \in \mathcal{R}^C$, where the value '1' is located at the c -th dimension for class c ($c = 1, 2, \dots, C$). Then, we can train C -independent GPR models to predict the label vector $\mathbf{y}^* = [y_1^*, y_2^*, \dots, y_C^*]^T$ of a testing data \mathbf{x}_* , and \mathbf{x}_* 's predicted class label $y^* = c$ if y_c^* is maximal among all the C dimensions. At the same time, we can obtain the corresponding variance $\text{cov}(y_c)$ to indicate the uncertainty of the predicted class label. It should be noticed that we indeed treat C -class classification problem as C -independent 2-class classification problems using a multiple-output GPR model.

2.3. Manifold Constrained GPR for Pose Estimation. Homeomorphic view manifold has been applied for pose estimation recently. Zhang et al. [10] used normalized 1-sphere in \mathcal{R}^2 as a homeomorphic manifold representation of all view manifolds in the case of 1D pose variation, while

[19] expending such homeomorphic manifold constraint for a more general case. A similar thing happened in the case of 2D and 3D pose variations. For the case of 2D pose variation, a normalized 2-sphere in \mathcal{R}^3 can represent a 2D pose variation of one object, and for the 3D case, the homeomorphic view manifold becomes a normalized 3-sphere in \mathcal{R}^4 . Assuming the pose angles of the input \mathbf{x}_i are $(\theta_i, \beta_i, \zeta_i)$, where θ , β , and ζ are the yaw angle, pitch angle, and roll angle, respectively, then, the pose on the homeomorphic view manifold can be written as [10]

$$\mathbf{y}_i = \begin{cases} \begin{bmatrix} \cos \theta_i \\ \sin \theta_i \end{bmatrix} \in \mathcal{R}^2 (\text{in 1D case}), \\ \begin{bmatrix} \cos \theta_i \cos \beta_i \\ \sin \theta_i \cos \beta_i \\ \sin \beta_i \end{bmatrix} \in \mathcal{R}^3 (\text{in 2D case}), \\ \begin{bmatrix} \cos \theta_i \cos \beta_i \cos \zeta_i \\ \sin \theta_i \cos \beta_i \cos \zeta_i \\ \sin \beta_i \cos \zeta_i \\ \sin \zeta_i \end{bmatrix} \in \mathcal{R}^4 (\text{in 3D case}). \end{cases} \quad (7)$$

Instead of simply using the pose angles, the representation of the poses on the homeomorphic view manifold can preserve the topology of the view manifolds. Such representation has achieved significant improvement for pose estimation [10, 12, 19].

Applying the homeomorphic manifold constraint to the GPR framework, we expand the output of the GPR framework and we can obtain a multiple output even in the case of 1D pose estimation. So we call such multiple-output GPR model *manifold constrained Gaussian process regression* (MCGPR). For example, in a 1D case, the original GPR model can be learned by the training data $\{(\mathbf{x}_i, \theta_i)\}_{i=1}^N$, where θ_i is the corresponding yaw angle of data \mathbf{x}_i , and such a single-output GPR model can predict the yaw angle θ^* of an input \mathbf{x}_* directly. But when predicting the 1D pose of the input \mathbf{x}_* using the MCGPR model, the MCGPR model should be learned by the training data $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, where the pose of \mathbf{x}_i is presented as \mathbf{y}_i using equation (7) in the 1D case. Then, we can get the predicted output in \mathcal{R}^2 as $\mathbf{y}_* = [y1^*, y2^*]^T$, where $y1^*$ and $y2^*$ are the first and the second dimensions separately obtained by their respective GPR models. Then, the estimated yaw angle can be computed by the inverse trigonometric function as

$$\theta^* = g(y1^*, y2^*) = \arctan \left(\frac{y2^*}{y1^*} \right). \quad (8)$$

When predicting the 2D pose of the input \mathbf{x}_* using the MCGPR model, the training data become $\{(\mathbf{x}_i, \theta_i, \beta_i)\}_{i=1}^N$,

where β_i is the corresponding pitch angle of data \mathbf{x}_i . Then, the predicted output in \mathcal{R}^3 is $\mathbf{y}_* = [y1^*, y2^*, y3^*]^T$, getting a third dimension more than the 1D case. According to equation (7), the estimated yaw angle θ^* can still be computed using equation (8), and the estimated pitch angle can be computed by the inverse sine function as

$$\beta^* = h(y3^*) = \arcsin(y3^*). \quad (9)$$

Similar solutions can be solved for the 3D case. It should be noticed that, unlike original GPR-based pose estimation that directly predicts the pose angles, such solutions are specially designed for our MCGPR model because the outputs of MCGPR are constrained on the homeomorphic manifold.

2.4. Uncertainty of Predicted Values. Let Δy represent the uncertainty of the target value y , then if we use the GPR model to predict y , we can get the variance of y , i.e., $\text{cov}(y)$ directly, and we define $\Delta y = \sqrt{\text{cov}(y)}$, where $\sqrt{\text{cov}(y)}$ is the standard deviation. By such definition, the uncertainty Δy will share the same physical meaning and unit as the target value y . To make it clear, we describe our process for calculating the uncertainty of the predicted class label and estimated pose angles in detail.

2.4.1. Uncertainty of Predicted Class Label. As mentioned in Section 2.2, the uncertainty of the predicted class label can be defined as

$$\Delta y = \sqrt{\text{cov}(yc)}, \quad \text{if } y_* = c, \quad (10)$$

where yc is the c -th dimension output of the predicted label vector \mathbf{y} .

2.4.2. Uncertainty of Predicted 1D Pose. In the 1D case, if we use the original GPR, the target value y will be the yaw angle θ , i.e., $y = \theta$, $\Delta \theta = \sqrt{\text{cov}(\theta)} = \sqrt{\text{cov}(y1)}$ just as we defined it. However, in the MCGPR model where $y1 = \cos \theta$ and $y2 = \sin \theta$, the relationship between the estimated yaw angle θ and the target values $y1$ and $y2$ follows the inverse trigonometric function as in equation (8). As $\Delta y1 = \sqrt{\text{cov}(y1)}$ and $\Delta y2 = \sqrt{\text{cov}(y2)}$ can be obtained directly by the model, to compute $\Delta \theta$, we consider the total differential of θ as

$$d\theta = \frac{\partial g}{\partial y1} dy1 + \frac{\partial g}{\partial y2} dy2, \quad (11)$$

where $\partial g / \partial (\cdot)$ stands for the partial differential. From one perspective, a differential dy can represent the variation of y , while from another perspective, dy can also indicate the uncertainty of y . So we can regard dy as Δy . Then, noticing that $y1^2 + y2^2 = \cos^2 \theta + \sin^2 \theta = 1$, the uncertainty

of θ can be computed according to the total differential in equation (11) as

$$\begin{aligned}\Delta\theta &= \sqrt{\left(\frac{\partial g}{\partial y_1}\right)^2 (\Delta y_1)^2 + \left(\frac{\partial g}{\partial y_2}\right)^2 (\Delta y_2)^2} \\ &= \sqrt{\left(\frac{y_2}{y_1^2 + y_2^2}\right)^2 (\Delta y_1)^2 + \left(\frac{y_1}{y_1^2 + y_2^2}\right)^2 (\Delta y_2)^2} \\ &= \sqrt{\left(\frac{y_2}{y_1^2 + y_2^2}\right)^2 \text{cov}(y_1) + \left(\frac{y_1}{y_1^2 + y_2^2}\right)^2 \text{cov}(y_2)} \\ &= \sqrt{(y_2)^2 \text{cov}(y_1) + (y_1)^2 \text{cov}(y_2)}.\end{aligned}\quad (12)$$

2.4.3. Uncertainty of Predicted 2D Pose. The output of the original GPR for 2D pose estimation will be $\mathbf{y} = [y_1, y_2]^T$, where $y_1 = \theta$ and $y_2 = \beta$ are the yaw angle and pitch angle, respectively, with the uncertainties $\Delta\theta = \sqrt{\text{cov}(\theta)} = \sqrt{\text{cov}(y_1)}$ and $\Delta\beta = \sqrt{\text{cov}(\beta)} = \sqrt{\text{cov}(y_2)}$. If we use the MCGPR model with output $\mathbf{y} = [y_1, y_2, y_3]^T$, where $y_1 = \cos\theta \cos\beta$, $y_2 = \sin\theta \cos\beta$, and $y_3 = \sin\beta$ according to equation (7), the relationship between the estimated pitch angle β and the target value y_3 follows the inverse trigonometric function as in equation (9), and the relationship between the estimated yaw angle θ and the target values y_1 and y_2 follows the inverse trigonometric function as in equation (8). Similar with the 1D case, we can compute the uncertainty of β according to its differential as

$$\Delta\beta = d\beta = \frac{dh}{dy_3} dy_3 = \frac{1}{\sqrt{1 - (y_3)^2}} \Delta y_3 = \sqrt{\frac{\text{cov}(y_3)}{1 - (y_3)^2}}. \quad (13)$$

Since the relationship between y_1 , y_2 , and y_3 changes to $y_1^2 + y_2^2 + y_3^2 = \cos^2\theta \cos^2\beta + \sin^2\theta \cos^2\beta + \sin^2\beta = 1$ in the 2D case, according to equations (11) and (12), the uncertainty of the yaw angle θ becomes

$$\begin{aligned}\Delta\theta &= \sqrt{\left(\frac{y_2}{y_1^2 + y_2^2}\right)^2 \text{cov}(y_1) + \left(\frac{y_1}{y_1^2 + y_2^2}\right)^2 \text{cov}(y_2)} \\ &= \sqrt{\left(\frac{y_2}{1 - (y_3)^2}\right)^2 \text{cov}(y_1) + \left(\frac{y_1}{1 - (y_3)^2}\right)^2 \text{cov}(y_2)}.\end{aligned}\quad (14)$$

By calculating the uncertainty, we can learn the relationship between the absolute error of the prediction result and the uncertainty and determine the corresponding threshold based on the existing results. Then, the result with uncertainty greater than the threshold can be regarded as an unreliable result, and a second interpretation could be performed to ensure the credibility of the results.

3. Experiments and Analyses

3.1. Dataset and Image Representation. We performed experiments on two simulated satellite image datasets BUAA-SID 1.0 [17, 41] and BUAA-SID 1.5 [10, 12, 19] to evaluate the proposed method. BUAA-SID 1.0 is a publicly available satellite image dataset containing 4600 gray images of 20 satellites together with 4600 corresponding binary images. These noise-free images were sampled on a viewing sphere from 230 viewpoints. The BUAA-SID 1.0 dataset can be used for testing the performance of GPR for multiview satellite recognition. Images in BUAA-SID 1.5 were simulated via the simulation method introduced in [42] using ten 3D satellite models selected from BUAA-SID 1.0. It contains four subsets. In the 1D subset of BUAA-SID 1.5, 3600 gray images and their corresponding binary images were simulated from 360 viewpoints uniformly sampled on a circle with the yaw angle $\theta \in [0^\circ, 360^\circ)$ and the pitch angle $\beta = 0^\circ$. Images in the 2D subset of BUAA-SID 1.5 were captured from 2042 viewpoints on a viewing sphere with the yaw angle $\theta \in [0^\circ, 360^\circ)$ and the pitch angle $\beta \in [-90^\circ, 90^\circ]$; the lighting subset of BUAA-SID 1.5 contains 10800 gray images of one satellite from the 1D subset, simulated in different lighting conditions, i.e., the phase angle of the light ranging from 0° to 90° in steps of 10° while the altitude angle of the light is 0° , 90° , and 180° , respectively; the noise subset of BUAA-SID 1.5 was obtained by adding Gaussian white noise to images in the 1D subset, and the variance varies from 0.001 to 0.01 in steps of 0.001. Compared with BUAA-SID 1.0, BUAA-SID 1.5 has a simpler posture change but more abundant data types, such as lighting and noise subsets. So the BUAA-SID 1.5 dataset can be used for better validation of GPR for both recognition and pose estimation. It should be noticed that the data, including BUAA-SID 1.0 and BUAA-SID 1.5 datasets, used to support the findings of this study are available from the corresponding author upon request.

Several methods have been proposed in the field of multiview space object recognition (e.g., [12, 17, 19, 31]). In order to compare the performance between the methods more fairly and objectively, we use the above two datasets to do experiments. We also used two groups of image representations to represent the satellite images: the first group consists of four representations to represent the shape information of space objects, including original binary image (BI), distance transform (DT) obtained by applying a signed distance function [10] to binary images, Hu's moment invariants (HU) [43], and Fourier descriptor (FD) [44]; the second group includes original gray image (GI) and histogram of oriented gradients (HOG) [45] which can describe the appearance variations of space objects. All these representations were used in a vector form (i.e., data \mathbf{x} in Section 2) for training and testing GPR for classification and pose estimation or the MCGPR model for pose estimation. It should be noticed that when solving a 1D or 2D pose estimation problem using the GPR model, the corresponding pose \mathbf{y} in Section 2.1 was represented as the pose angle θ or (θ, β) directly, while for the MCGPR model, \mathbf{y} was represented using equation (7) in 1D or 2D case, respectively.

TABLE 1: Satellite recognition results on BUAA-SID 1.0 and comparison with [17, 19]. Results are in bold if the proposed method performs the best.

Representation	Method	Accuracy (%)
HU	Kernel method [19]	54.13
	Our GPR method	71.00
FD	Kernel method [19]	57.52
	Our GPR method	62.30
HOG	Kernel method [19]	98.00
	Our GPR method	86.61
MI+FD+region covariance [17]+HOG	KLPP [17]	95.87
	LPP [17]	88.22
	KPCA [17]	69.70
	PCA [17]	68.57

TABLE 2: Satellite recognition results on BUAA-SID 1.0 and comparison with [19, 31]. Results are in bold if the proposed method performs the best and in italic if the second best.

Number of training image	Representation	Method	Accuracy (%)
80	HU	SVM [31]	48.93
		Kernel method [19]	55.50
		Our GPR method	68.91
80	FD	SVM [31]	34.73
		Kernel method [19]	53.86
		Our GPR method	59.25
80	Elastic net sparse coding [31]	SVM [31]	82.07
	HOG	Kernel method [19]	97.13
		Our GPR method	84.98
90	HU	SVM [31]	50.50
		Kernel method [19]	56.04
		Our GPR method	69.67
90	FD	SVM [31]	34.36
		Kernel method [19]	54.25
		Our GPR method	59.52
90	Elastic net sparse coding [31]	SVM [31]	83.25
	HOG	Kernel method [19]	96.60
		Our GPR method	85.78
100	HU	SVM [31]	53.08
		Kernel method [19]	55.11
		Our GPR method	70.22
100	FD	SVM [31]	35.92
		Kernel method [19]	54.90
		Our GPR method	60.57
100	Elastic net sparse coding [31]	SVM [31]	87.54
	HOG	Kernel method [19]	97.65
		Our GPR method	85.40

3.2. Results of Satellite Recognition and Pose Estimation

3.2.1. *Satellite Recognition.* We performed recognition experiments on the BUAA-SID 1.0 dataset and 1D subset of

BUAA-SID 1.5, respectively. As mentioned in Section 2.2, we learned one GPR model for each dimension of the output label vector, resulting 20 GPR models for satellite recognition on the BUAA-SID 1.0 dataset and 10 on the 1D subset of

BUAA-SID 1.5. To compare with [12, 17, 19], half of the images were used for training, and the rest were used for testing. Specially, for comparison with [19, 31] on BUAA-SID 1.0, 80, 90, or 100, training images of each satellite were randomly selected to learn the classification models, as in [31], and the rest were used for testing. Results are shown in Tables 1, 2, and 3. The results of our GPR method are in bold if it achieves the best and in italics if it achieves the second best. For the 20-class classification problem on BUAA-SID 1.0, we can significantly improve the recognition accuracy when using 12-dimensional Hu’s moment invariants (HU) and 20-dimensional Fourier descriptors (FD), as seen in Tables 1 and 2. It is promising since lower dimension of features means lower computation and less storage. In addition, because binary images of space objects can be captured more easily and completely than high-resolution gray images, shape representations like HU and FD are more suitable for real aerospace applications. It can be seen from Table 3 that, on the 1D subset of BUAA-SID 1.5, our GPR method performs better than those in [19] and [12] in most conditions except the second best for FD representation. This shows the good recognition capability of our method.

Recently, deep learning methods are widely used for general object recognition. Zeng et al. [33] introduced such technology for space target recognition. We reproduced the nine-layer deep convolutional neural network in [33] to achieve satellite recognition on our datasets. We chosen rotation and crop as the way of data augmentation and also fixed the magnitude of training data at 8-fold. The training data is the same as above. We also trained other popular state-of-the-art deep learning networks including ResNet [46] and DenseNet [47], to make our comparison more comprehensive. The results are shown in Table 4. We can see that our GPR model gets similar results as DCNN [33] on BUAA-SID 1.0, not better than ResNet [46] and DenseNet [47]. As for the BUAA-SID 1.5 dataset, our GPR model achieves better satellite recognition accuracy than DCNN [33], the same 100% performance as ResNet [46] and DenseNet [47] on 1D the and noise subsets, and slightly worse than ResNet [46] and DenseNet [47] on the lighting subset. This means that deeper networks like ResNet [46] and DenseNet [47] can achieve better recognition results, especially when the classes to be classified enlarge, i.e., the recognition problem becomes harder. By contrast, our GPR model can be learned more easily using limited computing resources and give uncertainty of the outputting results.

3.2.2. Pose Estimation. In order to evaluate the performance of pose estimation, we performed 1D and 2D pose estimation experiments on the 1D subset and 2D subset of BUAA-SID 1.5, respectively. In each subset, we chose half of the images for training and the rest for testing. It should be noticed that on the 1D subset, we used two training strategies: in one strategy, we trained GPR/MCGPR models for each satellite individually with the assumption that the categories were known, while in the other strategy, we trained GPR/MCGPR models (indicated as GPR-ALL and MCGPR-ALL in Table 5) for all the ten satellites without prior knowledge of categories. For the quantitative evaluation and comparison, we follow

TABLE 3: Satellite recognition results on the 1D subset of BUAA-SID 1.5 and comparison with [12, 19]. Results are in bold if the proposed method performs the best and in italics if the second best.

Representation	Method	Accuracy (%)
BI	Homeomorphic manifold [12]	100
	SVM [12]	100
	Kernel method [19]	100
	Our GPR method	100
DT	Homeomorphic manifold [12]	100
	SVM [12]	100
	Kernel method [19]	100
	Our GPR method	100
HU	Homeomorphic manifold [12]	98.9
	SVM [12]	99.6
	Kernel method [19]	94.00
	Our GPR method	99.71
FD	Homeomorphic manifold [12]	95.2
	SVM [12]	98.2
	Kernel method [19]	94.56
	Our GPR method	96
GI	Homeomorphic manifold [12]	100
	SVM [12]	100
	Kernel method [19]	100
	Our GPR method	100
HOG	Homeomorphic manifold [12]	99.9
	SVM [12]	100
	Kernel method [19]	100
	Our GPR method	100

TABLE 4: Satellite recognition results and comparison with deep learning methods.

Dataset	Method	Accuracy (%)
BUAA-SID 1.0	DCNN [33]	86.78
	ResNet [46]	94.32
	DenseNet [47]	96.17
	Our GPR method	86.61
BUAA-SID 1.5 1D subset	DCNN [33]	100
	ResNet [46]	100
	DenseNet [47]	100
	Our GPR method	100
BUAA-SID 1.5 noise subset	DCNN [33]	99.58
	ResNet [46]	100
	DenseNet [47]	100
	Our GPR method	100
BUAA-SID 1.5 lighting subset	DCNN [33]	88.69
	ResNet [46]	100
	DenseNet [47]	100
	Our GPR method	97.593

TABLE 5: Pose estimation results on the 1D subset of BUAA-SID 1.5. Results are in bold if the proposed method performs the best and in italic if the second best.

Representation	Method	MAE (°)	AE < 1° (%)	AE < 2° (%)	AE < 5° (%)
BI	MCGPR	4.83	53.44	<i>70.11</i>	<i>88.06</i>
	MCGPR-ALL	6.13	33.67	53.56	78.72
	GPR	11.4	22.00	38.33	61.00
	GPR-ALL	14.46	10.28	21.33	45.61
	Kernel method [19]	4.84	53.33	70.17	88.22
DT	MCGPR	6.78	<i>48.94</i>	<i>63.22</i>	<i>78.72</i>
	MCGPR-ALL	8.82	20.67	39.22	67.50
	GPR	14.8	18.39	32.50	52.33
	GPR-ALL	20.60	5.83	11.39	28.06
	Kernel method [19]	6.07	53.67	68.33	82.56
HU	MCGPR	57.3	4.28	8.17	17.50
	MCGPR-ALL	75.21	1.33	2.33	6.56
	GPR	69.5	1.06	2.39	5.17
	GPR-ALL	83.94	0.67	1.33	2.89
	kernel method [19]	80.37	1.22	2.61	6.44
FD	MCGPR	23.2	<i>8.44</i>	16.94	35.78
	MCGPR-ALL	34.81	3.50	7.11	16.89
	GPR	38.9	3.17	6.78	14.94
	GPR-ALL	56.54	0.94	2.39	6.22
	Kernel method [19]	27.09	8.50	16.28	32.83
GI	MCGPR	0.29	93.06	98.00	99.72
	MCGPR-ALL	0.43	90.44	96.56	99.50
	GPR	3.02	52.33	72.44	90.67
	GPR-ALL	3.84	38.89	62.00	86.72
	Kernel method [19]	0.29	93.06	97.94	99.72
	ResNet [46]	4.5206	17.22	31.67	74.44
HOG	DenseNet [47]	22.41	2.22	6.11	15.56
	MCGPR	5.32	43.61	64.72	82.83
	MCGPR-ALL	6.04	27.44	48.61	77
	GPR	15.5	9.67	18.89	42.06
	GPR-ALL	16.51	6.61	13.67	35.11
	Kernel method [19]	5.36	43.50	64.72	82.94

the same indicators as in [19], i.e., using the absolute error (AE), which can be defined as the absolute value of the difference between the ground truth angle and the estimated pose angle. We report the mean absolute error (MAE) to evaluate the pose estimation performance on the entire testing set. At the same time, in order to analyze the distribution of angle errors in detail, we also report the percentage of testing images of which the pose angle is correctly estimated with an AE less than a threshold (1°, 2°, or 5°). Since most recent vision-based satellite pose estimation methods [27, 29] are not learning-based and not suitable for comparison, we compared our proposed method with the most comparative kernel regression method [19]. In addition, for comprehensive comparison with recently available learning-based methods, we also adjusted the network structures of ResNet

[46] and DenseNet [47] and trained them in a regression way to estimate 1D and 2D pose on BUAA-SID 1.5.

Experimental results are shown in Tables 5 and 6. It can be seen that the proposed method performs better than or similar as the state-of-the-art [19] in both 1D and 2D pose estimations. Deep learning networks ResNet [46] and DenseNet [47] perform relatively worse for pose estimation. This may be caused by the limited training data and the highly synthesized deep features, resulting terrible regression performance on the BUAA-SID 1.5 dataset. Comparing with the original GPR, MCGPR achieves significant improvement, surprisingly improving about an order of magnitude in MAE of 1D pose estimation using GI representation. This validates the role of the manifold constraint. We can see that GI performs the best among all the six kinds of image

TABLE 6: Pose estimation results on the 2D subset of BUAA-SID 1.5. Results are in bold if the proposed method performs the best and in italic if the second best.

Representation	Pose	Method	MAE (°)	AE < 1° (%)	AE < 2° (%)	AE < 5° (%)
BI	Pitch	MCGPR	27.8	8.23	14.01	28.99
		GPR	32.3	4.90	9.60	19.10
		Kernel method [19]	27.75	8.23	13.91	28.89
BI	Yaw	MCGPR	24.0	7.35	<i>14.79</i>	30.17
		GPR	30.6	3.92	7.25	16.85
		Kernel method [19]	24.04	7.35	14.89	29.97
DT	Pitch	MCGPR	15.4	8.03	15.67	33.69
		GPR	21.1	4.80	9.50	22.53
		Kernel method [19]	15.53	7.54	15.38	32.71
DT	Yaw	MCGPR	14.7	<i>10.19</i>	<i>19.00</i>	41.33
		GPR	23.6	4.31	9.01	22.92
		Kernel method [19]	14.85	10.97	19.10	40.84
HU	Pitch	MCGPR	58.7	<i>1.08</i>	2.45	6.17
		GPR	42.4	5.48	5.88	5.88
		Kernel method [19]	61.68	1.08	2.84	6.17
HU	Yaw	MCGPR	89.3	0.88	<i>1.47</i>	3.23
		GPR	90.0	0.29	0.39	0.98
		Kernel method [19]	89.34	0.69	1.57	3.33
FD	Pitch	MCGPR	40.8	1.86	4.11	10.48
		GPR	40.3	<i>1.67</i>	<i>3.04</i>	8.81
		Kernel method [19]	69.62	1.57	2.25	6.37
FD	Yaw	MCGPR	50.6	0.69	<i>1.37</i>	<i>4.70</i>
		GPR	52.5	0.69	0.98	3.82
		Kernel method [19]	75.95	0.59	1.47	5.39
GI	Pitch	MCGPR	1.04	69.05	87.07	98.04
		GPR	1.89	44.66	66.01	92.95
		Kernel method [19]	1.04	69.15	87.07	98.04
		ResNet [46]	3.35	19.5	37.81	74.93
		DenseNet [47]	3.84	12.73	35.65	72.87
GI	Yaw	MCGPR	<i>0.94</i>	72.87	88.05	97.65
		GPR	8.21	19.49	34.48	64.15
		Kernel method [19]	0.93	72.77	87.86	97.75
		ResNet [46]	47.62	2.25	4.21	8.42
		DenseNet [47]	47.45	1.18	2.64	5.29
HOG	Pitch	MCGPR	3.79	26.44	47.40	77.08
		GPR	7.99	13.52	26.93	55.53
		Kernel method [19]	3.79	26.44	47.31	77.08
HOG	Yaw	MCGPR	4.29	<i>27.13</i>	47.11	81.29
		GPR	13.7	7.74	14.79	36.14
		Kernel method [19]	4.29	27.23	47.01	81.29

representations, as it is more sensitive to pose variation. HU and FD seem to be unsuccessful in pose estimation, with quite high MAE on both the 1D and 2D subsets. This may be because they are view-invariant features and not sensitive to pose variation, and such invariance reduces pose estimation performance. Thus, to get better pose estimation results,

we need to use pose-sensitive representations. Comparing results in Tables 5 and 6, it can be obviously seen that 2D results are worse than 1D results, which can be easily understood by considering the increasing difficulty of the pose estimation problem when the dimensionality rises from 1D to 2D. Similarly, due to the growing variance within training

data, GPR-ALL/MCGPR-ALL performs worse than GPR/MCGPR. Therefore, pose estimation models should better be trained for individual targets.

3.2.3. Joint Satellite Recognition and Pose Estimation. According to the findings in Section 3.2.2, previous recognition approach can provide category prior to selecting proper models for better pose estimation. Thus, we use pose-after-recognition strategy to achieve joint satellite recognition and pose estimation based on individual approaches introduced in Sections 2.2 and 2.3 of this paper; i.e., we recognize the category of a testing data using GPR and then estimate its pose using MCGPR specially trained for the recognized category. Table 7 shows joint satellite recognition and pose estimation results of our proposed method and the comparison with the state-of-the-art networks [12, 19]. We perform the best pose estimation results for five out of all the six kinds of image representations, except the second best for BI representation. Noticing the good recognition accuracy shown in Table 3, our proposed method achieves the state-of-the-art results for joint satellite recognition and pose estimation.

3.3. Uncertainty Analysis. The most important advantage of the GPR method than previous works [12, 17, 19, 31] is that GPR can provide variance of each predicted output value, which can be used to calculate the uncertainty of the predicted value as introduced in Section 2.4. Such uncertainty can describe the credibility of the predicted value and give us a way to choose convincing results. This is very helpful in aerospace applications.

3.3.1. Uncertainty of Satellite Recognition. We chose the satellite recognition results using FD representation in Table 3 to analyze the uncertainties of predicted class labels, since the recognition accuracy of FD is less than 100%, meaning that some of testing images were incorrectly classified. We computed the Euclidean distance between the predicted label vector and the ground truth label vector to quantitatively evaluate the error of recognition. Figure 1 shows the relationship between uncertainty and the Euclidean distance. It can be seen that predicted labels with large Euclidean distances, i.e., large recognition errors, correspond to the large uncertainties. According to the data in Figure 1, if we select an uncertainty threshold $T_u = 0.014$ and regard the recognition results of the uncertainties more than T_u as untrustworthy results, we can get the recognition accuracy of the rest results raised to 100%. It means that we can choose more convincing recognition results according to uncertainties.

3.3.2. Uncertainty of Pose Estimation. To validate the capability of pose estimation uncertainty provided by our method, the performance of the representation GI on the pose estimation problem, we used 1D pose estimation result GI representation in Table 5 to analyze the absolute errors and uncertainties of the estimated 1D pose angles. Figure 2 shows the results of three satellites. We can see that both GPR and MCGPR have one thing in common that estimated poses with large uncertainties usually have high absolute errors, resulting in obvious peaks at the same positions on the

TABLE 7: Joint satellite recognition and pose estimation results on the 1D subset of BUAA-SID 1.5. Results are in bold if the proposed method performs the best and in italic if the second best.

Representation	Method	MAE (°)
BI	Homeomorphic manifold [12]	4.33
	SVM [12]	9.98
	Kernel method [19]	6.80
	Proposed method	4.83
DT	Homeomorphic manifold [12]	7.95
	SVM [12]	13.5
	Kernel method [19]	10.47
	Proposed method	6.78
HU	Homeomorphic manifold [12]	64.0
	SVM [12]	74.7
	Kernel method [19]	84.65
	Proposed method	57.46
FD	Homeomorphic manifold [12]	27.7
	SVM [12]	25.5
	Kernel method [19]	58.34
	Proposed method	21.85
GI	Homeomorphic manifold [12]	0.51
	SVM [12]	3.05
	Kernel method [19]	0.43
	Proposed method	0.29
HOG	Homeomorphic manifold [12]	8.73
	SVM [12]	10.2
	Kernel method [19]	6.47
	Proposed method	5.32

abscissa in the figure. This can verify the feasibility of the definition of pose uncertainty in Section 2.4. We can also see that the mean absolute error of MCGPR is 4.4055° , quite less than 10.3643° of GPR in Figure 2. However, the mean uncertainty of MCGPR is 30.8165° , larger than 7.0988° of GPR. This means that MCGPR improves the pose estimation performance of GPR at the expense of enlarging the uncertainties of some predicted values. Such enlarging can be explained by equation (2) where the uncertainty of MCGPR is an additive combination of uncertainties of individual GPRs. It should be noticed that the spikes in Figure 2 appear when the appearances of the satellite of symmetrical poses (the difference between pose angles is near 180°) are quite similar since the satellite is usually of geometrical symmetry. In this case, the errors and uncertainties will tend to peak.

Similar results can also be found in Figure 3 for the uncertainty of 2D pose. We can see that the positive correlations between the absolute error and the uncertainty are pronounced in the overall trend but locally various. This may be caused by the more challenging difficulty in 2D pose estimation than the 1D case.

3.4. Noise Robustness. Noise in the data is an important factor affecting the performance of machine learning models. In

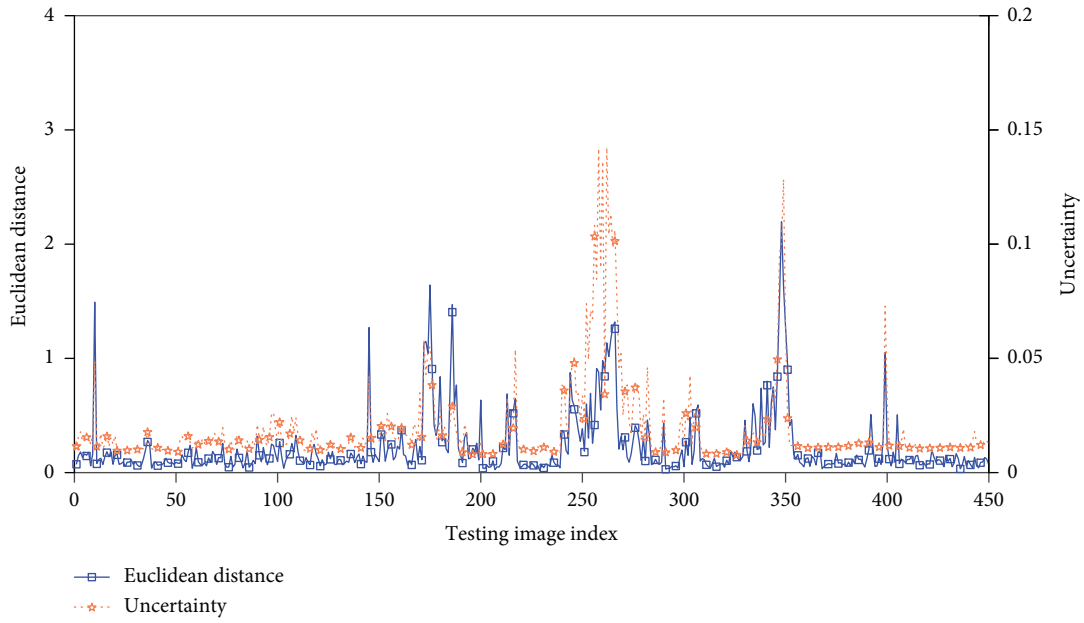


FIGURE 1: Uncertainty analysis results of satellite recognition.

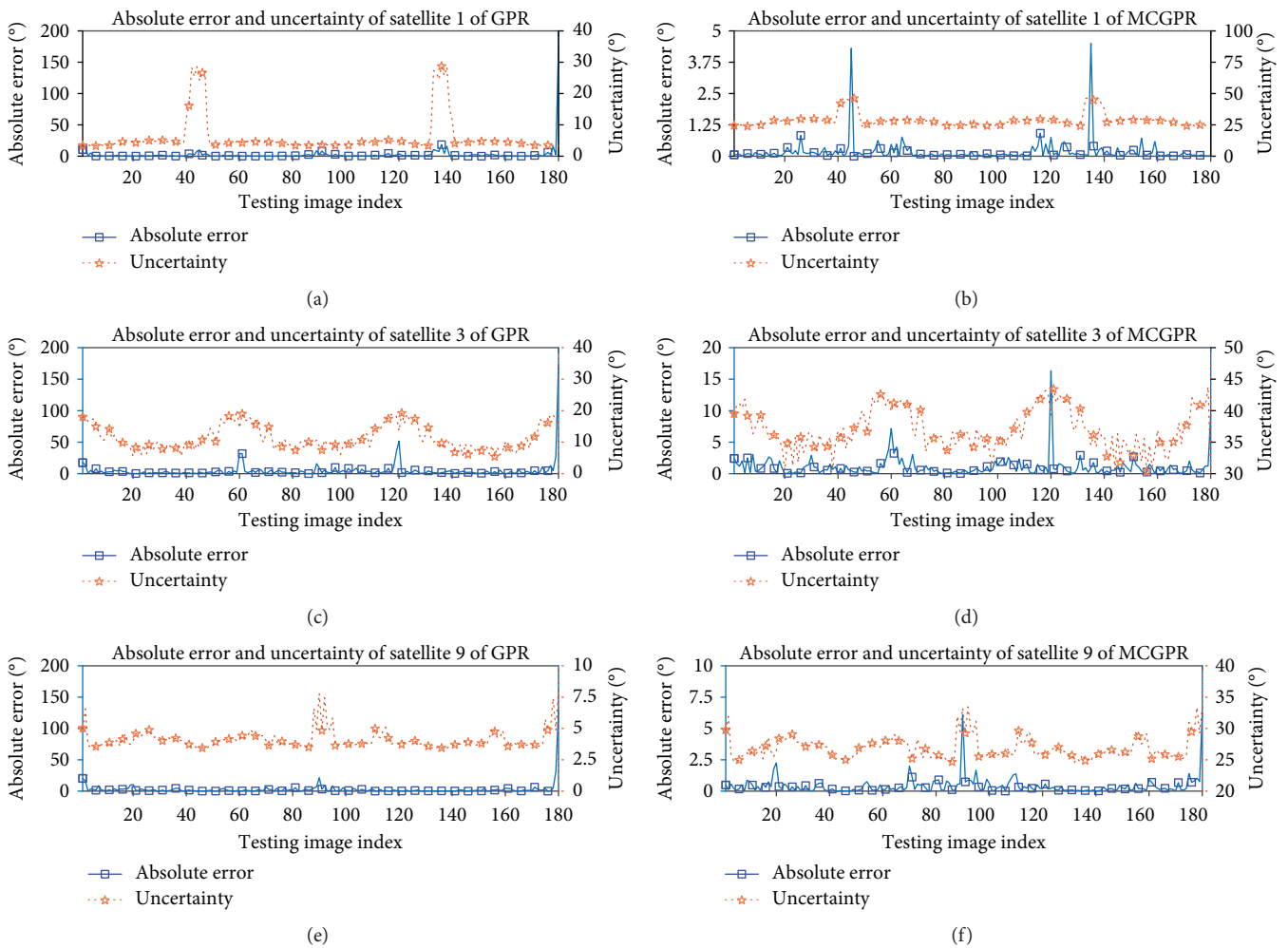


FIGURE 2: Uncertainty analysis results of three satellites on the 1D subset of BUAA-SID 1.5.

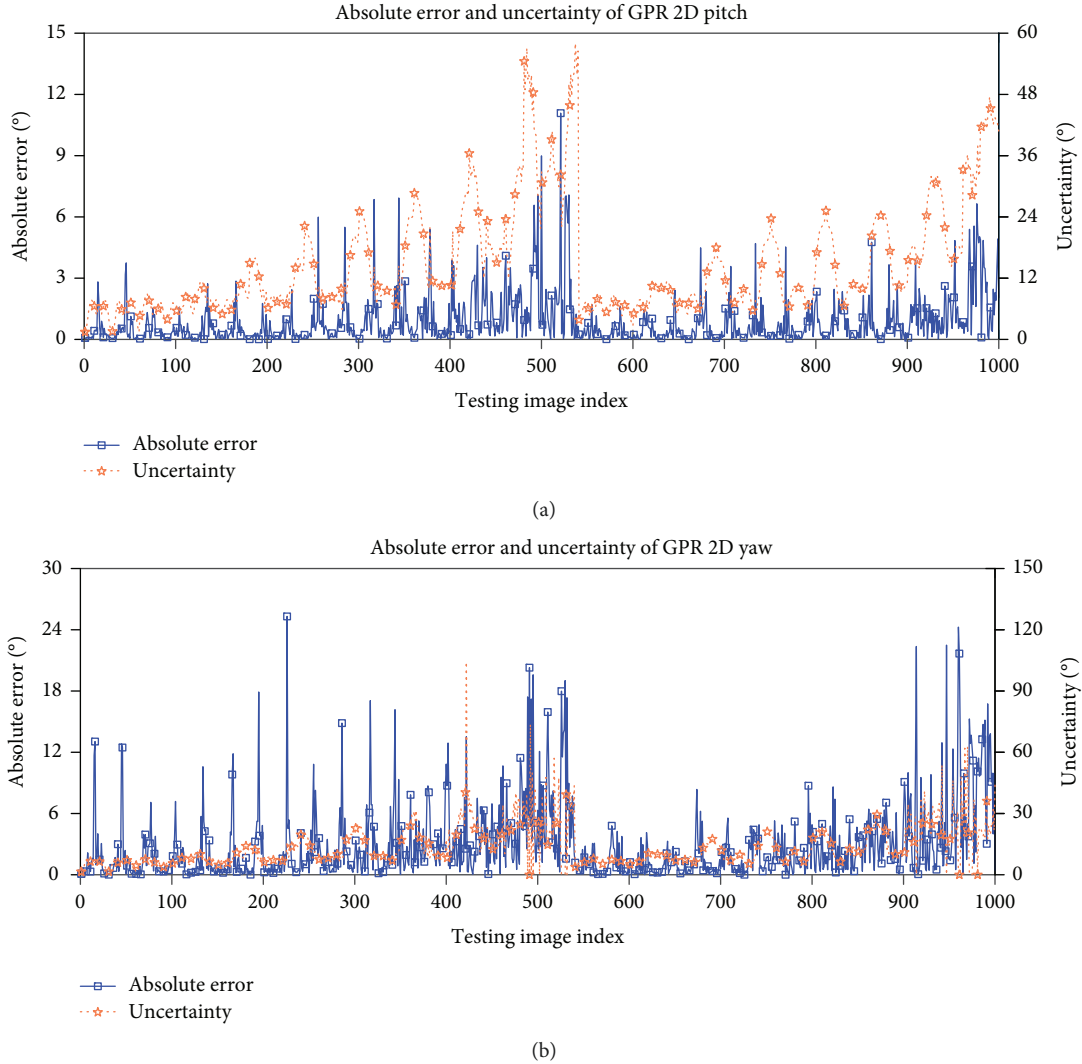


FIGURE 3: Uncertainty analysis results on the 2D subset of BUAA-SID 1.5.

this section, we analyzed the satellite recognition and pose estimation performance of the proposed models trained by noise and noise-free data.

3.4.1. Noise Robustness of Satellite Recognition. To evaluate the robustness against noise for satellite recognition, we used the models trained on the training set of the 1D subset of BUAA-SID 1.5 in Section 3.2.1 and tested on the noisy images in the noise subset of BUAA-SID 1.5 corresponding to the testing set of the 1D subset. In other words, we used noisy data to test the model learned by noise-free data. Figure 4(a) shows the experimental results. It can be seen that noise significantly affects recognition performance except GI. BI and DT perform better than HU, FD, and HOG. The noise robustness of GI is particularly prominent because its recognition accuracy remains 100% when the variance of Gaussian noise increases. Figure 4(b) shows experimental results of GPR models trained on noisy training data, i.e., the noisy images in the noise subset of BUAA-SID 1.5 corresponding

to the training set of the 1D subset. The performance of noise GPR models is improved obviously than the noise-free models in Figure 4(a). This means that GPR can model the noise in the training data and get promising results on realistic noisy data. Therefore, it is necessary to model the noise in the learning procedure for real applications.

3.4.2. Noise Robustness of Pose Estimation. We also experimented on the noise subset of BUAA-SID 1.5 to evaluate the noise robustness of pose estimation. We compared the performance of models learned in Section 3.2.2 using noise-free training data (indicated as “GPR/MCGPR+image representation”) and the corresponding models retrained by noisy training data (indicated as “NOISE+image representation”) in Figures 5, 6, and 7. It is shown that noise significantly affects pose estimation performance as well. Results get worse rapidly when the variance of noise increases, and GI has better noise robustness than other image representations. By modeling the noise in the learning procedure, most

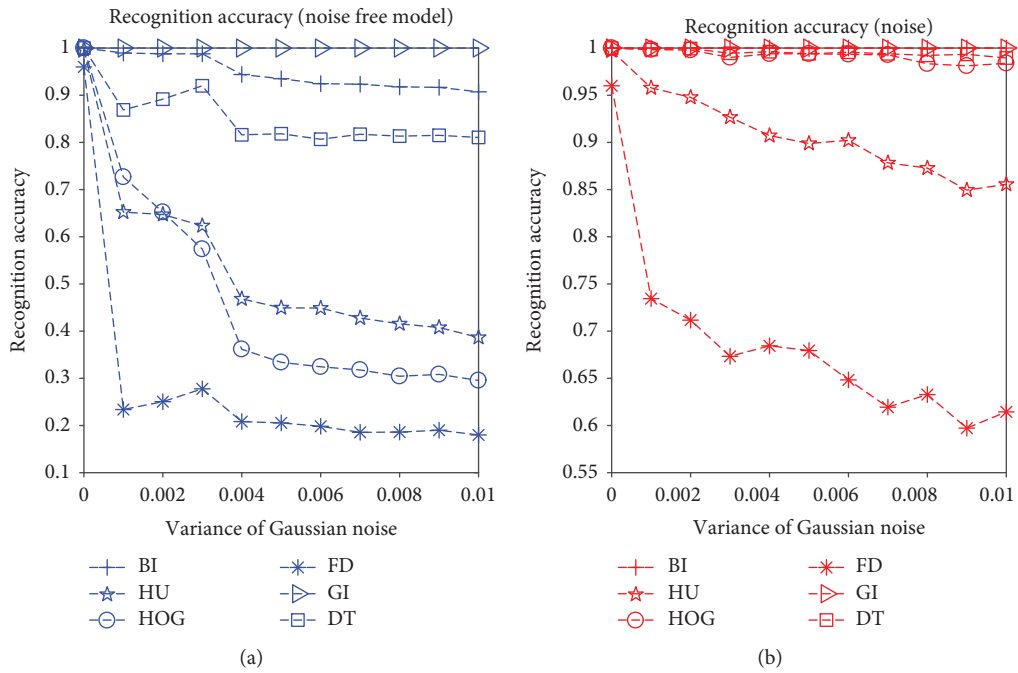


FIGURE 4: Noise robustness results of GPR for satellite recognition.

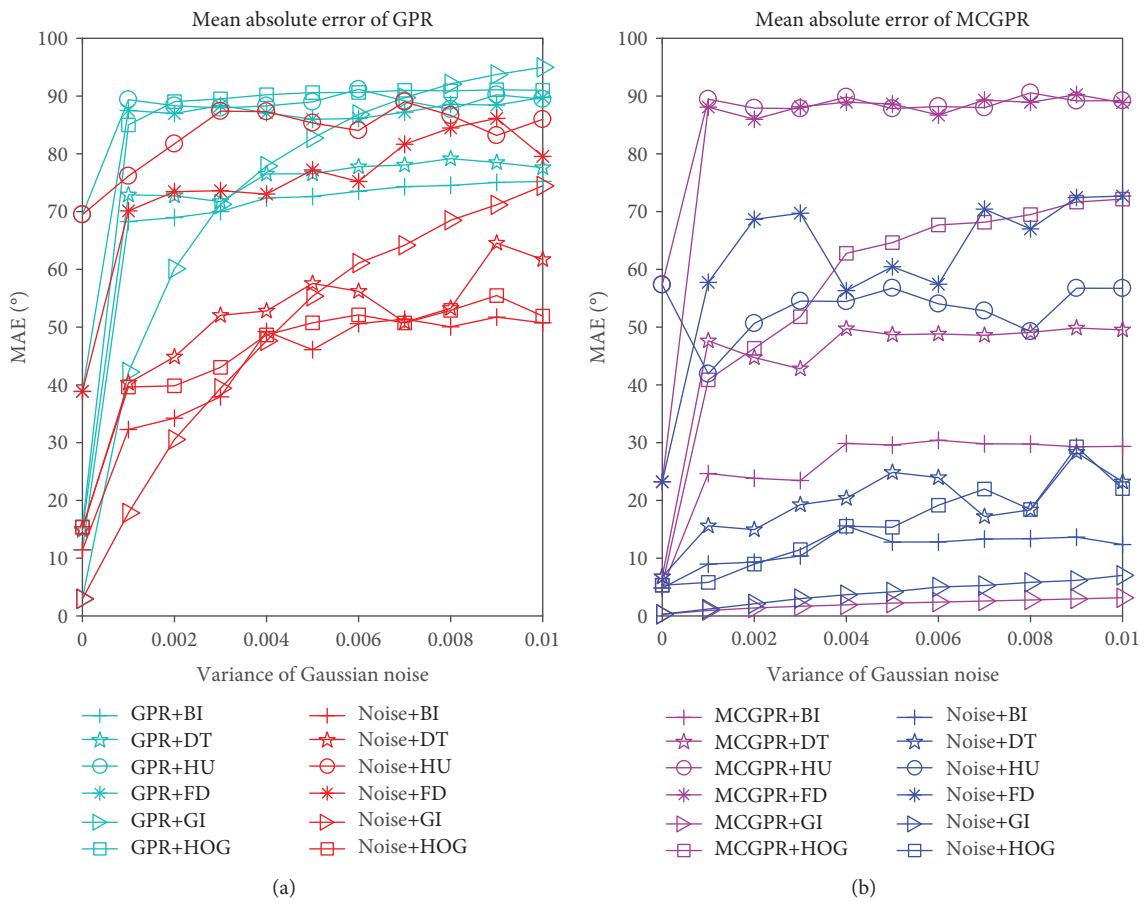


FIGURE 5: Noise robustness results of GPR and MCGPR for pose estimation (MAE).

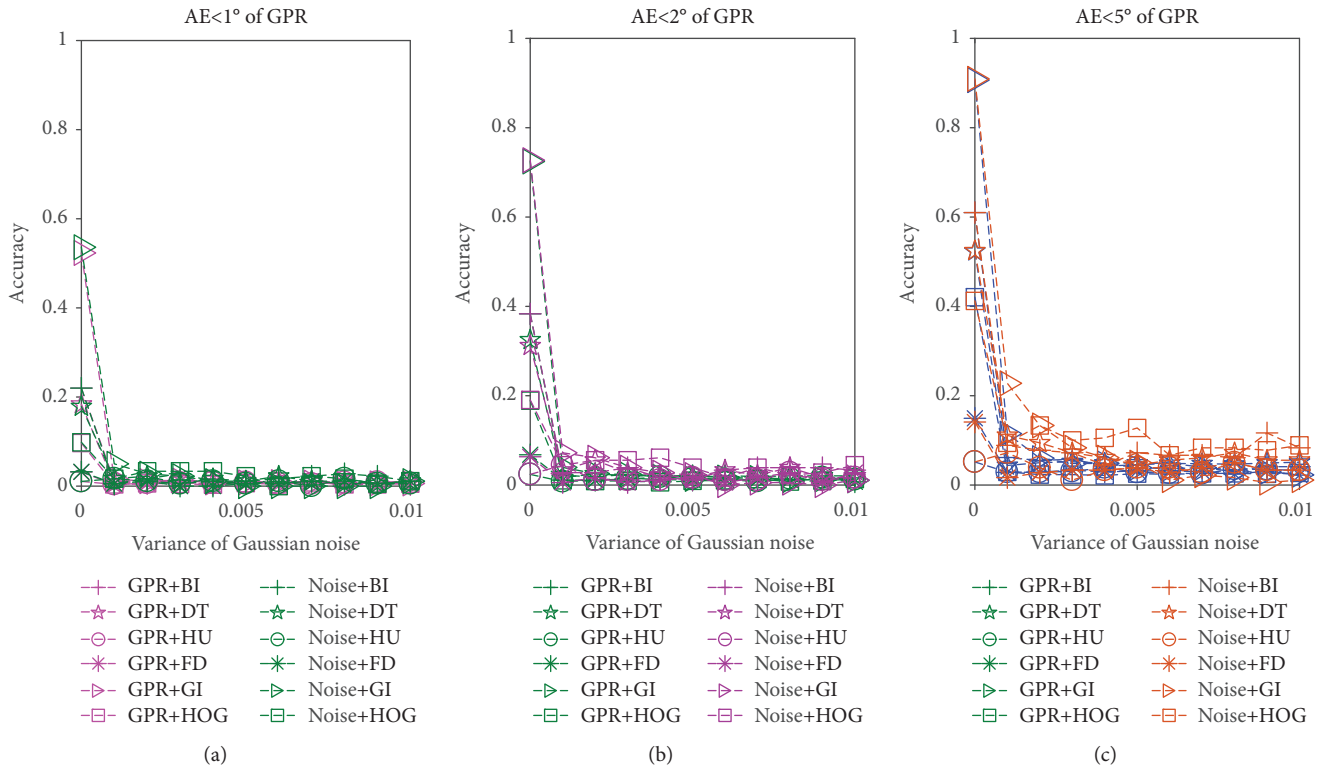


FIGURE 6: Noise robustness results of GPR for pose estimation (% AE less than a threshold).

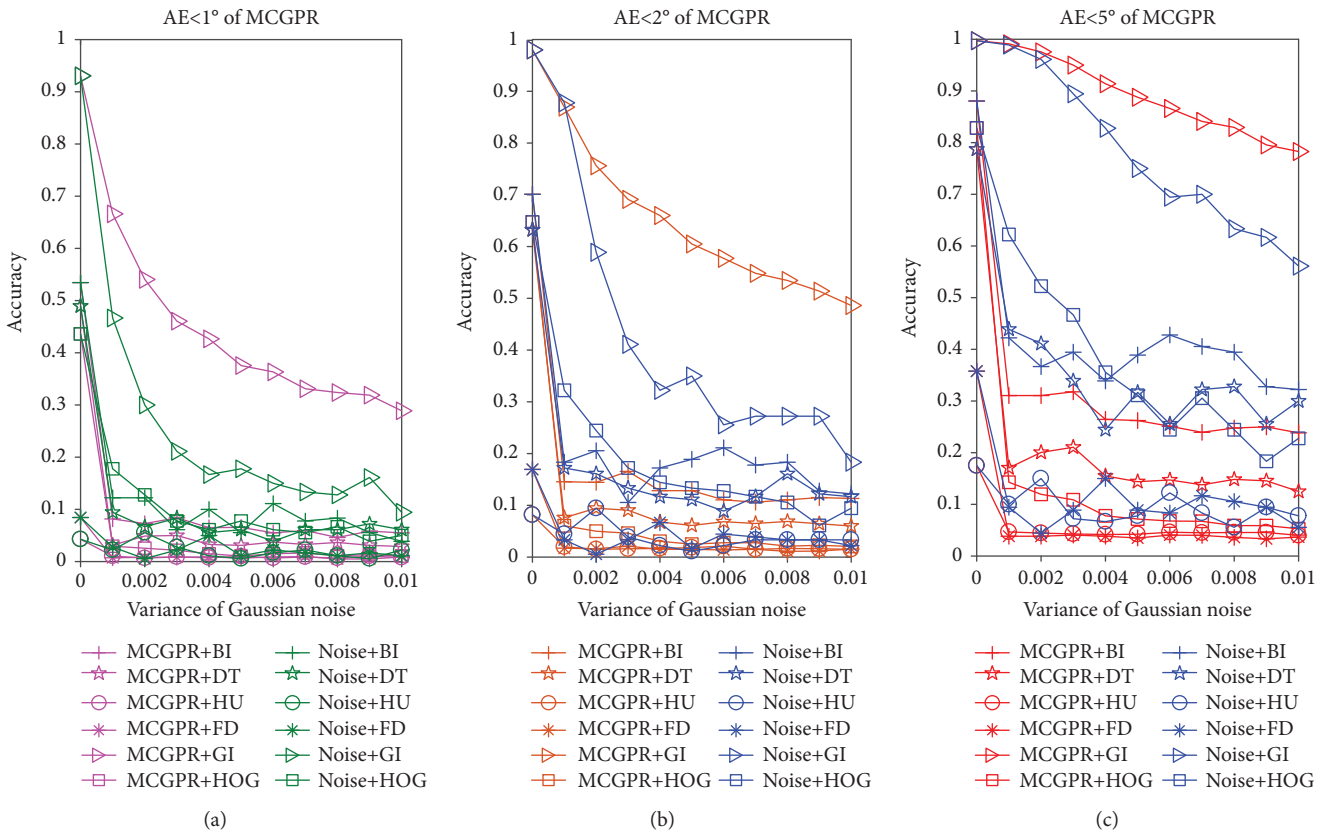


FIGURE 7: Noise robustness results of MCGPR for pose estimation (% AE less than a threshold).

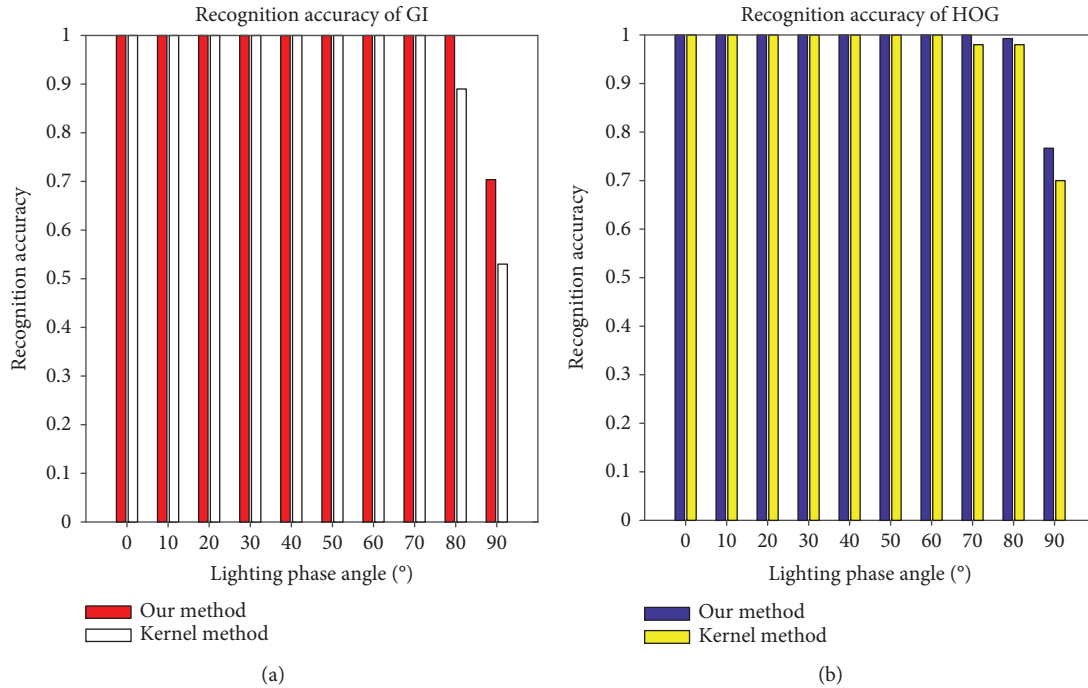


FIGURE 8: Lighting robustness results of satellite recognition.

GPR/MCGPR models get improved pose estimation performance, including the lower mean absolute error and higher percentage accuracy with AE less than 1° , 2° , or 5° . MCGPR with GI representation is an exception. We explain that, since the manifold constraint in MCGPR has enhanced the pose estimation ability of GPR and GI has been proved as the image representation with best noise robustness, the noise in the training data may disturb the modeling of MCGPR for accurate pose estimation and thus reduce the pose estimation performance.

3.5. Lighting Robustness. Lighting is another important factor affecting satellite recognition and pose estimation. Lighting condition will change along with the lighting phase angle (also called the sun phase angle), which is determined by the relative position between the sun, the imaging sensor, and the target satellite. Such changes may affect appearance representations, e.g., GI and HOG. Thus, we analyzed lighting robustness on the lighting subset of BUAA-SID 1.5 using the model learned on training data of the 1D subset BUAA-SID 1.5 in Sections 3.2.1 and 3.2.2, respectively.

3.5.1. Lighting Robustness of Satellite Recognition. Figure 8 shows the lighting robustness results of GPR with GI and HOG. The comparison with the kernel method [19] shows that our method has a better lighting robustness of satellite recognition than the kernel method [19] when the lighting phase angle is larger than 60° . It can also be seen that HOG is more robust than GI.

3.5.2. Lighting Robustness of Pose Estimation. Experimental results for pose estimation are shown in Figure 9. We can also see that HOG is more robust than GI for pose estimation, and

MCGPR with HOG representation can still achieve MAE less than 10° even in bad lighting condition; i.e., the lighting phase angle is larger than 60° .

3.6. Performance on Sparse Training Data. Actually, it is difficult to get a large quantity of images for training due to the limitation of real imaging systems in space. If we can achieve a good performance by using fewer training images, it will be more helpful and applicative in practice. So it is important and necessary to clarify how many training images our proposed method actually need. Thus, in this section, we kept the testing set and the parameter of the model learned on the training data of the 1D subset of BUAA-SID 1.5 in Sections 3.2.1 and 3.2.2 and just reduced the number of training images for each satellite from 180 to 10 in order to analyze the satellite recognition and pose estimation performance on sparse training data.

3.6.1. Satellite Recognition Performance on Sparse Training Data. Figure 10 shows the results for satellite recognition. It can be seen that most representations can still perform 100% recognition accuracy given only 10 training images. The worse performance of HU and FD is around 90% in this case, which is also acceptable. This shows the strong ability of GPR for satellite recognition in terms of sparse training data.

3.6.2. Pose Estimation Performance on Sparse Training Data. For pose estimation, we selected one satellite in the 1D subset of BUAA-SID 1.5 and chose GI as the image representation. Results are shown in Figure 11. We can see that performance changes slowly when the number of training images is more than 100, while the average uncertainty keeps falling although the speed becomes slower and slower. Thus, more

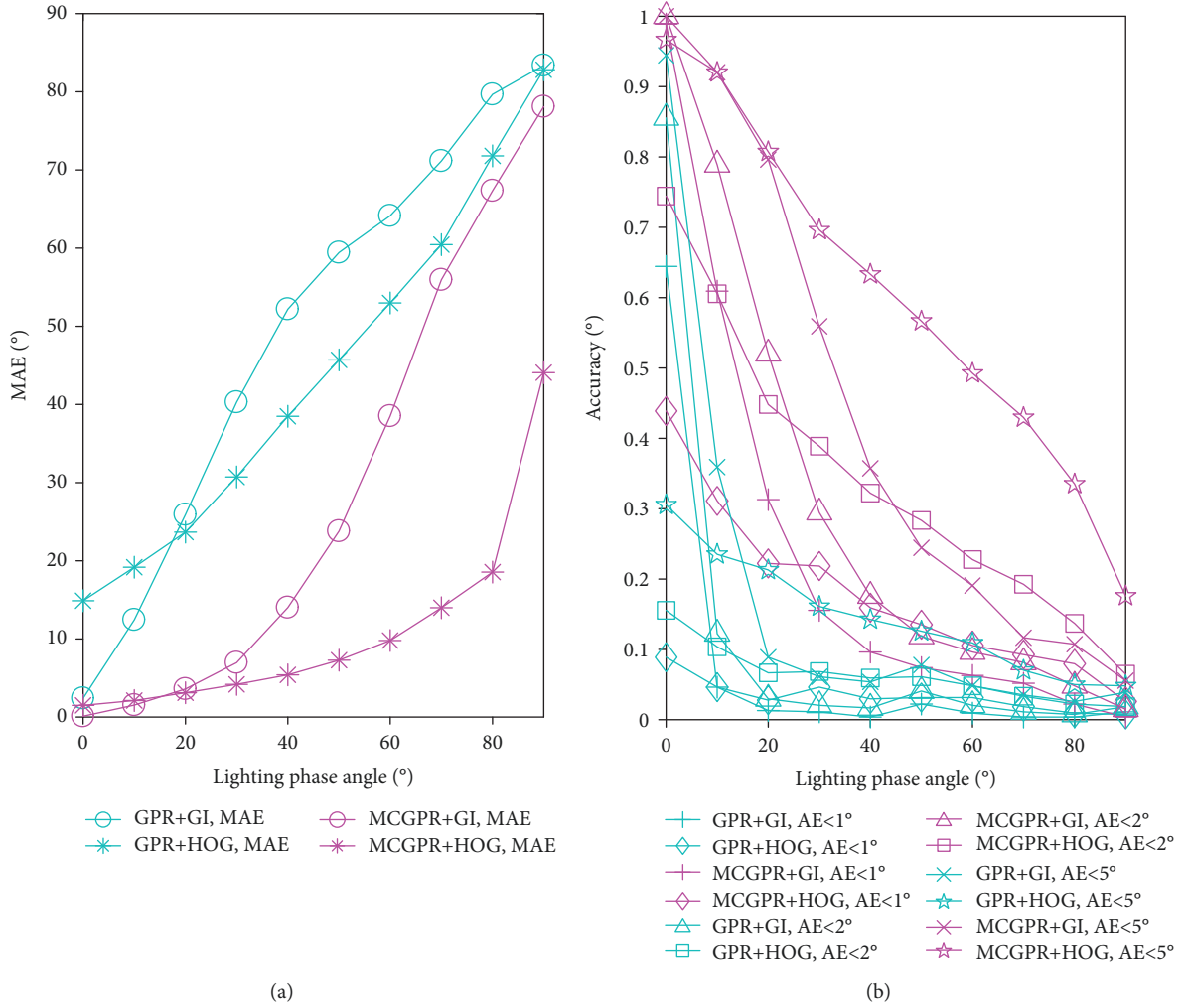


FIGURE 9: Lighting robustness results of pose estimation.

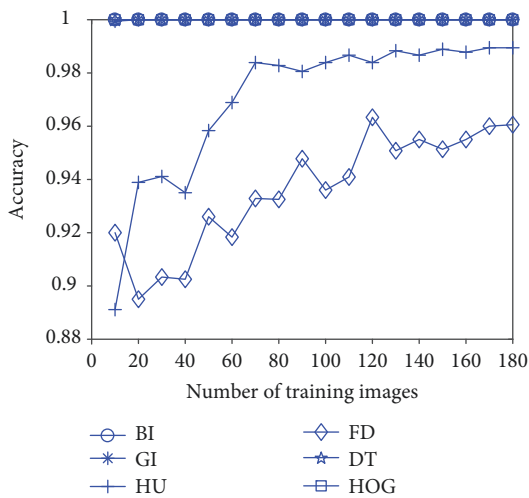


FIGURE 10: Satellite recognition performance on sparse training data.

training data may help to minimize the uncertainty, but pose estimation results may not be improved too much. Comparing with satellite recognition, pose estimation is more sensitive about the number of training images. Taking all factors into consideration, we suggest at least 50 training images to get acceptable pose estimation results and more than 100 necessary images for better results.

3.7. Cross-Validation for Parameter Determination. There are two parameters affecting the performance of our proposed method, i.e., the characteristic length-scale ℓ^2 of the covariance function in equation (2) and the variance σ_n^2 of the additive noise ε in the regression model (also in the likelihood function $\mathbf{y}|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma_n^2 I)$). Since we use the squared exponential function as the covariance function in our model, we empirically set the length scale by experience as

$$\ell^2 = 0.1 \times \max_{i,j \in \{1, \dots, N\}} \|\mathbf{x}_i - \mathbf{x}_j\|^2, \quad (15)$$

where $\|\cdot\|$ is the second norm in \mathcal{R}^d .

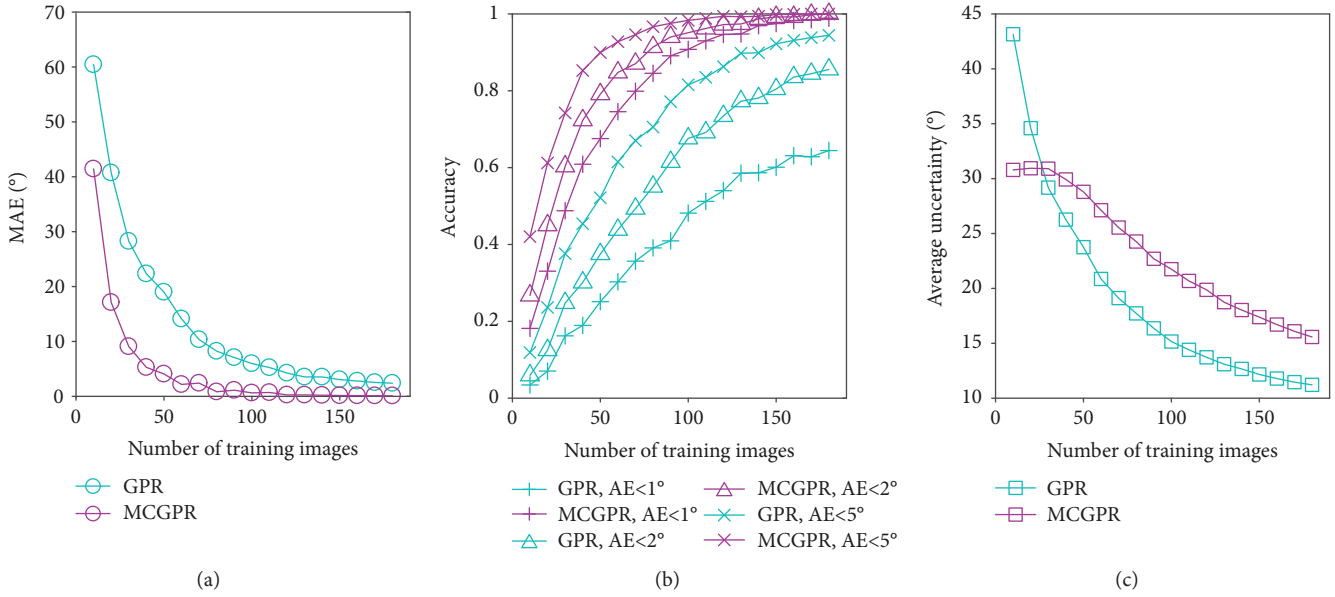


FIGURE 11: Pose estimation performance on sparse training data.

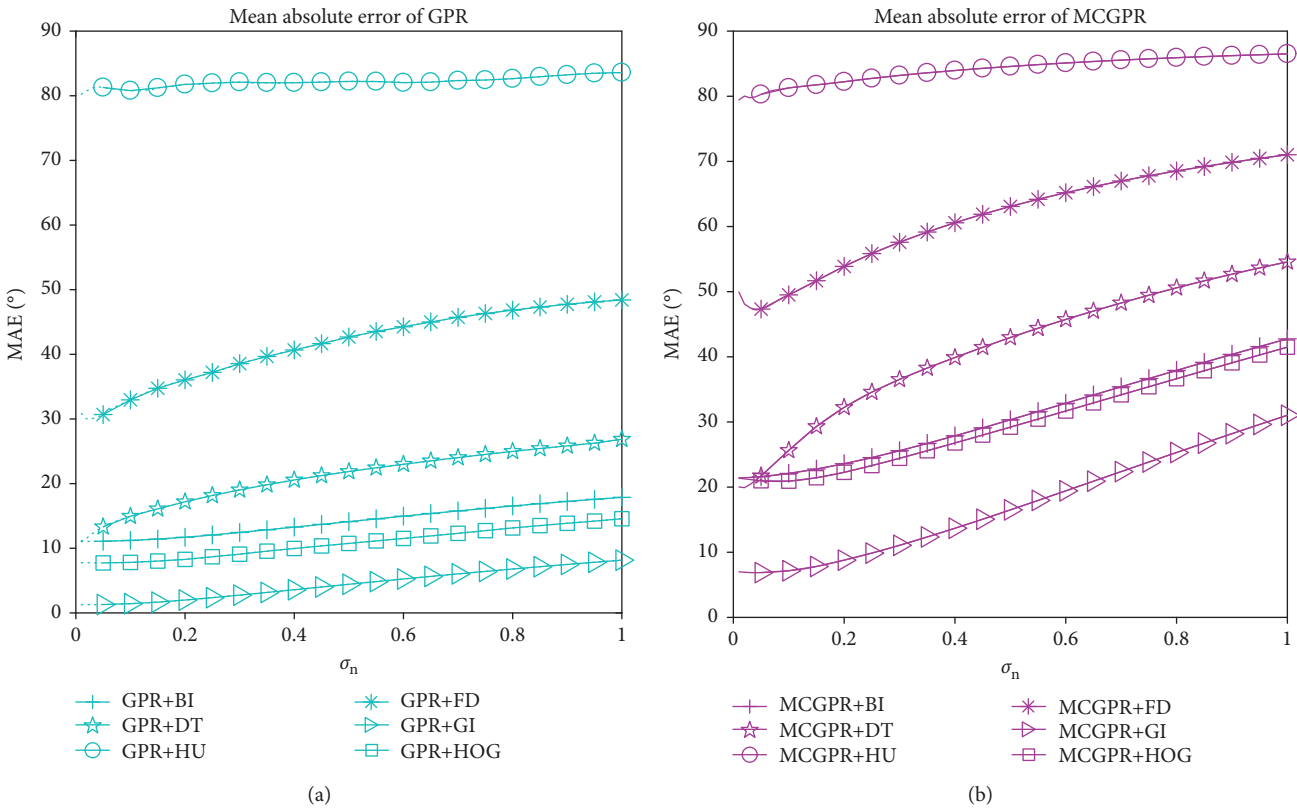


FIGURE 12: Cross-validation results on the 1D subset of BUAA-SID 1.5 for pose estimation.

We did 4-fold cross-validation on the training set to select the most suitable σ_n . We divided the training set to 4-fold and used onefold as the validation set in each round of the 4-round validation experiments. We use the average performance of recognition accuracy or MAE to select proper σ_n for satellite

recognition or pose estimation. Some cross-validation results for pose estimation on the 1D subset of BUAA-SID 1.5 are shown in Figure 12. Similar validation experiments were done on other subsets as well. The parameters achieving reported experimental results in this paper are listed in Tables 8 and 9.

TABLE 8: Parameters used for satellite recognition. $\sigma_n(i)$ indicates the variance in the i -th independent GPR model to predict the label vector.

Representation	$\sigma_n(1)$	$\sigma_n(2)$	$\sigma_n(3)$	$\sigma_n(4)$	$\sigma_n(5)$	$\sigma_n(6)$	$\sigma_n(7)$	$\sigma_n(8)$	$\sigma_n(9)$	$\sigma_n(10)$
BI	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
DT	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
HU	0.006	0.001	0.002	0.002	0.003	0.001	0.001	0.002	0.003	0.002
FD	0.01	0.01	0.01	0.008	0.007	0.01	0.01	0.01	0.01	0.01
GI	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
HOG	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

TABLE 9: Parameters used for pose estimation.

Method+representation	Subset	σ_n
MCGPR+BI	2D	0.01
MCGPR+DT	2D	0.01
MCGPR+HU	2D	1.00
MCGPR+FD	2D	0.45
MCGPR+GI	2D	0.01
MCGPR+HOG	2D	0.01
GPR+BI	2D	0.01
GPR+DT	2D	0.01
GPR+HU	2D	1.00
GPR+FD	2D	0.24
GPR+GI	2D	0.03
GPR+HOG	2D	0.01
MCGPR+BI	1D & noise	0.01
MCGPR+DT	1D & noise	0.01
MCGPR+HU	1D & noise	0.01
MCGPR+FD	1D & noise	0.02
MCGPR+GI	1D & noise	0.01
MCGPR+HOG	1D & noise	0.05
GPR+BI	1D & noise	0.01
GPR+DT	1D & noise	0.02
GPR+HU	1D & noise	0.01
GPR+FD	1D & noise	0.04
GPR+GI	1D & noise	0.04
GPR+HOG	1D & noise	0.08
MCGPR+GI	Lighting	0.01
MCGPR+HOG	Lighting	0.01
GPR+GI	Lighting	0.13
GPR+HOG	Lighting	0.01

4. Conclusion

In this paper, we have proposed a novel monocular vision-based method by employing Gaussian process regression to solve satellite recognition and pose estimation. Our approach can effectively recognize the categories of satellites and estimate their relative poses. We have achieved the state-of-the-art satellite recognition and pose estimation results on BUAA-SID datasets. For satellite recognition, only 10

training images of each satellite are needed to get near 100% recognition accuracy. For pose estimation, our method can obtain a pose error less than 5 degrees for most cases. In addition, no other requirements but training data are needed for our pose estimation model; thus, there are no aforementioned limitations, such as camera calibration and optical markers. Our GPR-based method can provide the uncertainty of the predicted values (pose angles or categories), which may be used to choose convincing results in applications. Because of the supervised learning procedure, our method may be more suitable for cooperative space objects of which enough images can be obtained for training.

Data Availability

The data, including BUAA-SID 1.0 and 1.5 datasets, used to support the findings of this study are available from the corresponding author upon request.

Disclosure

The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; and in the decision to publish the results.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

H.Z., C.Z., G.M., and Z.J. conceived and designed the experiments. H.Z. and Y.Y. performed the experiments for pose estimation. C.Z. performed the experiments for satellite recognition. H.Z., C.Z., G.M., and Y.Y. analyzed the data. H.Z. and C.Z. wrote the paper.

Acknowledgments

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 (Grant Nos. 2016YFB0501300 and 2016YFB0501302), and the Fundamental Research Funds for the Central Universities.

References

- [1] J. Kelsey, J. Byrne, M. Cosgrove, S. Seereeram, and R. Mehra, "Vision-based relative pose estimation for autonomous rendezvous and docking," in *2006 IEEE Aerospace Conference*, pp. 1–20, Big Sky, MT, USA, March 2006.
- [2] F. Aghili, M. Kuryllo, G. Okouneva, and C. English, "Robust vision-based pose estimation of moving objects for automated rendezvous & docking," in *2010 IEEE International Conference on Mechatronics and Automation*, pp. 305–311, Xi'an, China, August 2010.
- [3] A. Petit, E. Marchand, and K. Kanani, "Vision-based space autonomous rendezvous: a case study," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 619–624, San Francisco, CA, USA, September 2011.
- [4] C. Liu and W. Hu, "Relative pose estimation for cylinder-shaped spacecrafts using single image," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 3036–3056, 2014.
- [5] P. Rogata, E. di Sotto, F. Câmara et al., "Design and performance assessment of hazard avoidance techniques for vision-based landing," *Acta Astronautica*, vol. 61, no. 1-6, pp. 63–77, 2007.
- [6] A. Cropp, P. Palmer, P. McLauchlan, and C. Underwood, "Estimating pose of known target satellite," *Electronics Letters*, vol. 36, no. 15, pp. 1331–1332, 2000.
- [7] M. Abderrahim, J. Diaz, C. Rossi, and M. Salichs, "Experimental simulation of satellite relative navigation using computer vision," in *Proceedings of 2nd International Conference on Recent Advances in Space Technologies, 2005. RAST 2005*, pp. 379–384, Istanbul, Turkey, Turkey, June 2005.
- [8] X. Du, B. Liang, and Y. Tao, "Pose determination of large non-cooperative satellite in close range using coordinated cameras," in *2009 International Conference on Mechatronics and Automation*, pp. 3910–3915, Changchun, China, August 2009.
- [9] S. Segal, A. Carmi, and P. Gurfil, "Vision-based relative state estimation of non-cooperative spacecraft under modeling uncertainty," in *2011 Aerospace Conference*, pp. 1–8, Big Sky, MT, USA, March 2011.
- [10] H. Zhang, Z. Jiang, and A. Elgammal, "Vision-based pose estimation for cooperative space objects," *Acta Astronautica*, vol. 91, pp. 115–122, 2013.
- [11] H. Zhang and Z. Jiang, "Spacecraft attitude estimation based on attitude-weighted kernel regression," *Journal of Beijing University of Aeronautics and Astronautics*, vol. 40, pp. 494–499, 2014.
- [12] H. Zhang, Z. Jiang, and A. Elgammal, "Satellite recognition and pose estimation using homeomorphic manifold analysis," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 1, pp. 785–792, 2015.
- [13] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi, "Pose estimation for spacecraft relative navigation using model-based algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 1, pp. 431–447, 2017.
- [14] G. Zhang, H. Liu, J. Wang, and L. Jiang, "Vision-based system for satellite on-orbit self-servicing," in *2008 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 296–301, Xian, China, July 2008.
- [15] H. Benninghoff, T. Boge, and F. Rems, "Autonomous navigation for on-orbit servicing," *KI-Künstliche Intelligenz*, vol. 28, no. 2, pp. 77–83, 2014.
- [16] K. van Hecke, G. C. H. E. de Croon, D. Hennes, T. P. Setterfield, A. Saenz-Otero, and D. Izzo, "Self-supervised learning as an enabling technology for future space exploration robots: ISS experiments on monocular distance learning," *Acta Astronautica*, vol. 140, pp. 1–9, 2017.
- [17] M. Gang, J. Zhiguo, L. Zhengyi, Z. Haopeng, and Z. Danpei, "Full-viewpoint 3D space object recognition based on kernel locality preserving projections," *Chinese Journal of Aeronautics*, vol. 23, no. 5, pp. 563–572, 2010.
- [18] H. Ding, X. Li, and H. Zhao, "An approach for autonomous space object identification based on normalized AMI and illumination invariant MSA," *Acta Astronautica*, vol. 84, pp. 173–181, 2013.
- [19] H. Zhang and Z. Jiang, "Multi-view space object recognition and pose estimation based on kernel regression," *Chinese Journal of Aeronautics*, vol. 27, no. 5, pp. 1233–1241, 2014.
- [20] H. Zhang, Q. Wei, and Z. Jiang, "3D reconstruction of space objects from multi-views by a visible sensor," *Sensors*, vol. 17, no. 7, article 1689, 2017.
- [21] Q. Wei, Z. Jiang, and H. Zhang, "Robust spacecraft component detection in point clouds," *Sensors*, vol. 18, no. 4, p. 933, 2018.
- [22] J. Sharma, "Space-based visible space surveillance performance," *Journal of Guidance, Control, and Dynamics*, vol. 23, no. 1, pp. 153–158, 2000.
- [23] W. Glass, M. J. Duggin, R. A. Motes, K. A. Bush, and M. Klein, "Multi-spectral image analysis for improved space object characterization," in *Proceedings Volume 7467, Nanophotonics and Macrophotonics for Space Environments III; 74670K*, San Diego, CA, USA, August 2009.
- [24] P. Zimmer, J. McGraw, and M. Ackermann, "Affordable wide-field optical space surveillance using sCMOS and GPUs," in *Advanced Maui Optical and Space Surveillance Technologies Conference*, Wailea, Maui, Hawaii, September 2016.
- [25] A. Duncan, R. Kendrick, C. Ogden et al., "SPIDER: next generation chip scale imaging sensor update," in *Advanced Maui Optical and Space Surveillance Technologies Conference*, Wailea, Maui, Hawaii, September 2016.
- [26] X. Zhang, H. Zhang, Q. Wei, and Z. Jiang, "Pose estimation of space objects based on hybrid feature matching of contour points," in *Advances in Image and Graphics Technologies. IGTA 2016. Communications in Computer and Information Science, vol 634*, T. Tan, G. Wang, S. Wang, Y. Liu, X. Yuan, R. He, and S. Li, Eds., pp. 184–191, Springer, Singapore, 2016.
- [27] X. Zhang, Z. Jiang, H. Zhang, and Q. Wei, "Vision-based pose estimation for textureless space objects by contour points matching," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 5, pp. 2342–2355, 2018.
- [28] F. Terui, H. Kamimura, and S. Nishida, "Motion estimation to a failed satellite on orbit using stereo vision and 3D model matching," in *2006 9th International Conference on Control, Automation, Robotics and Vision*, pp. 1–8, Singapore, Singapore, December 2006.
- [29] C. Meng, Z. Li, H. Sun, D. Yuan, X. Bai, and F. Zhou, "Satellite pose estimation via single perspective circle and line," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 6, pp. 3084–3095, 2018.
- [30] R. Pan, G. Xiao, and Z. Jing, "Feature-based image fusion scheme for satellite recognition," in *2010 13th International Conference on Information Fusion*, Edinburgh, UK, July 2010.

- [31] J. Shi, Z. Jiang, H. Feng, H. Zhang, and G. Meng, "Elastic net sparse coding-based space object recognition," *Acta Aeronautica et Astronautica Sinica*, vol. 34, pp. 1129–1139, 2013.
- [32] D. Zhao, M. Lu, X. Zhang, J. Shi, and Z. Jiang, "Satellite recognition via sparse coding based probabilistic latent semantic analysis," *International Journal of Humanoid Robotics*, vol. 11, no. 2, article 1450008, 2014.
- [33] H. Zeng, Y. Xia, H. Zeng, and Y. Xia, "Space target recognition based on deep learning," in *2017 20th International Conference on Information Fusion (Fusion)*, pp. 1–5, Xi'an, China, July 2017.
- [34] H. Zhang, Z. Jiang, Y. Yao, and G. Meng, "Vision-based pose estimation for space objects by Gaussian process regression," in *2015 IEEE Aerospace Conference*, pp. 1–9, Big Sky, MT, USA, March 2015.
- [35] C. Williams and C. Rasmussen, "Gaussian processes for regression," in *Advances in Neural Information Processing Systems 8*, D. Touretzky, M. Mozer, and M. Hasselmo, Eds., pp. 514–520, MIT Press, Cambridge, MA, USA, 1996.
- [36] P. Boyle and M. Frean, "Dependent Gaussian processes," in *Advances in Neural Information Processing Systems 17*, L. Saul, Y. Weiss, and L. Bottou, Eds., MIT Press, Cambridge, MA, USA, 2005.
- [37] M. Kuss and C. Rasmussen, "Assessing approximations for Gaussian process classification," in *Advances in Neural Information Processing Systems*, Y. Weiss, B. Schölkopf, and J. Platt, Eds., vol. 18, MIT Press, Cambridge, MA, USA, 2006.
- [38] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, USA, 2006.
- [39] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Gaussian processes for object categorization," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 169–188, 2010.
- [40] S. Ambikasaran, D. Foreman-Mackey, L. Greengard, D. W. Hogg, and M. O'Neil, "Fast direct methods for Gaussian processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 252–265, 2016.
- [41] H. Zhang, Z. Liu, Z. Jiang, M. An, and D. Zhao, "BUAA-SID1.0 space object image dataset," *Spacecraft Recovery & Remote Sensing*, vol. 31, pp. 65–71, 2010.
- [42] H. Zhang, W. Zhang, and Z. Jiang, "Space object, high-resolution, optical imaging simulation of space-based systems," in *Proceedings Volume 8385, Sensors and Systems for Space Applications V; 838511*, Baltimore, MD, USA, May 2012.
- [43] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.
- [44] C. T. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curves," *IEEE Transactions on Computers*, vol. C-21, no. 3, pp. 269–281, 1972.
- [45] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pp. 886–893, San Diego, CA, USA, June 2005.
- [46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, Las Vegas, NV, USA, June 2016.
- [47] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, Honolulu, HI, USA, July 2017.

