Acta Astronautica 140 (2017) 235–246

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Database construction for vision aided navigation in planetary landing

Meng Yu [a,b], Hutao Cui [b], Shuang Li [a,*], Yang Tian [b]

[a] Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
[b] Harbin Institute of Technology, Harbin 150001, China

ARTICLE INFO

Keywords:
Vision-aided navigation
Planetary landing
Database construction
Feature exploitability

ABSTRACT

In this paper, a novel database construction method for passive-image based navigation system within the planetary precise pin-point landing (PPL) background is presented. The key concept is selecting qualified visual features to construct the visual database by examining their contribution to the navigation system. We first define a metric named feature exploitability to evaluate the visual feature's distinctiveness and its spatial imagery distribution. After that, a greedy selection method is employed to construct the database by selecting features with high feature-exploitability scores. Then, a hierarchical feature retrieval method is proposed to achieve the adaptation of image-scale variation during landing and improve the efficiency of feature retrieval. To evaluate our proposed approach, the Monte Carlo simulation and an experimental test are conducted, simulation results show the advantage of the feature exploitability driven database construction method over other database construction methods and the necessity of proper database construction in a vision-aided navigation system for PPL mission.

1. Introduction

Future planetary precise pin-point landing (PPL) mission requires the real-time update of lander's position and attitude (pose information) with respect to the pre-determined landing [1], the Passive Image based Navigation System (PINS) is considered as a promising solution for such mission, as it provides drift-free absolute lander pose information in the PPL scenario [2]. In the PINS, the planetary terrain is first sensed by local visual feature extraction using an on-board camera, and then matched to a globally-referenced visual database to gain the absolute pose of lander [3]. The development of PINS in PPL mission has drawn much attention in recent years [4–8]. The pioneer research dates back to the study carried out by Cuseo et al. [9], in which terrain relative navigation (TRN) is first introduced in the context of planetary landing to achieve a more precise navigation. The Japan Aerospace Exploration Agency (JAXA) designed a multi-sensor navigation system for MUSES-C (the lander) in Hayabusa exploration mission [10], in which the Harris features are regarded as visual landmarks, line-of-sight from lander to feature is employed to guide an on-board LiDAR for active ranging. NASA developed the Descent Image Motion Estimation System (DIMES) for Mars landing mission [11], it tracks Harris features in descent images and uses template matching to achieve data association, by such means, the estimation of horizontal velocity is evidently improved. Cheng et al. [12] proposed a crater recognition based visual navigation solution for asteroid landing, in which Canny edges are detected and paired to represent crater, which serves as the visual landmark for navigation. Shuang Li et al. [13] proposed a complete vision based navigation framework for planetary landing, A. I. Mourikis et al. [6] proposed a vision-aided inertial navigation solution for Mars landing, in which Harris corners are considered as the visual features and matched to the database using the normalized template matching method. A. E. Johnson and Montgomery et al. [4] presented a detailed review of TRN methods for PPL landing, in which a number of visual cues and their effectiveness on TRN are investigated. The most recent work has focused on developing a real-time GN&C prototype named Lander Vision System (LVS) for the Mars 2020 mission [14], from which an IMU and camera are interfaced using Field Programmable Gate Array (FPGA) to carry out precise pin-point landing, the TRN works in a coarse and then fine feature matching mode, these matches are then fused with IMU data under a standard EKF filter, which is similar to the vision-aided navigation structure proposed in Ref. [6], their helicopter field test shows promising results [15].

The visual database for VINS must be constructed prior to planetary landing and cover the entire landing ellipse. The landing ellipse is determined in the prior investigation, where factors like entry uncertainty, wind disturbance or actuator failure are taken into account by scientists to reach a consensus [16], for example, the landing ellipse in Mars landing must cover an area of $20 \times 10\ km^2$ near the equator (latest

* Corresponding author.
E-mail addresses: yuxy21@nuaa.edu.cn (M. Yu), lishuang@nuaa.edu.cn (S. Li).

http://dx.doi.org/10.1016/j.actaastro.2017.07.041
Received 26 April 2017; Received in revised form 18 June 2017; Accepted 27 July 2017
Available online 1 September 2017
0094-5765/

result from Mars Science Laboratory, MSL mission [16–18]). Among studies in the context of visual navigation system for PPL mission, one of the pioneer works that involved selecting visual features to construct the visual database is [19], in which the visual features with small mutual imagery distance are grouped as a "constellation" in a database, however, although this approach improves the efficiency of feature retrieval, the database is still too large to store on-board a planetary lander. Delaune et al. [20] proposed to delete the landmarks that are too close to one another when constructing the database, their approach results in the reduction of feature mis-matching. T. J. Steiner et al. [21], proposed an efficient database construction method, where two metrics are leveraged in selecting visual features to construct database, one is related to the probability of camera observing and recognizing the features during landing and the other is the mutual distance between visual features, based upon their approach, features that hold greater likelihood of being recognized and in the meanwhile, contributes more to position estimation are selected, their Monte Carlo experimental test confirmed the necessity of proper database construction.

In theory, the database would be constructed by collecting all the visual features within the entire landing ellipse, however, the assembly of all the features is too large to store on-board the computer or retrieve in real-time [22]. In addition, consider the fact that planetary terrain is quite barren and lacks structured landmarks, mismatch is more apt to occur in feature matching compared to the earth ground applications. To overcome the above mentioned issues, the visual database can instead consist of several representative visual features, which can be selected from a collection of orbit images or high-resolution images acquired by the planetary rover in prior investigations.

In this paper, we propose a novel visual database construction method based upon a new metric called "feature exploitability". This metric measures a visual feature's distinctiveness as well as its imagery space distribution, we further propose a hierarchical database retrieval structure to improve the efficiency of feature retrieving. To evaluate the performance of our proposed approach, we used a planetary landscape generator to simulate the planetary surface and employed a vision-aided inertial navigation system (VAINS) to compare the proposed approach with other database construction methods. In addition, an indoor experimental test is built to test the robustness of the proposed approach against the variation of solar elevation angle and lighting intensity, which are two main factors in visual feature characterization [25] and have an evident influence on the performance of the feature association. The remainder of this paper is organized as follows: Section 2 gives the definition of feature exploitability. Section 3 presents the feature exploitability based database construction algorithm. Section 4 presents the hierarchical feature retrieving approach. Section 5 presents the simulation results and discussion. Finally, conclusions are drawn in Section 6.

2. Feature exploitability

This section presents novel metric called feature exploitability to measure a feature's validity to the visual navigation system. We define feature exploitability as a metric composed of the two followed independent factors:

1. Similarity metric-*S*: It measures the distinctiveness of a visual feature, a visual feature F in database $\mathbf{C}$ should be easily distinguished from the others, and otherwise it will bring much ambiguity to the feature matching.
2. Distance metric-*D*: It measures the distance between two visual features, which is investigated to be approximately proportional to the pose covariance reduction.

Since these two factors are independent, we define their weighted sum as the "feature exploitability":

$$E(F,\mathbf{C}) = \alpha S(F,\mathbf{C}) + (1-\alpha)D(F,\mathbf{C}) \tag{2.1}$$

where E denotes the feature exploitability, F denotes a visual feature, $\mathbf{C}$denotes the visual database, α is a parameter that leverages the similarity metric-*S* and the distance metric-*D*.The exploitability of a visual feature relates to its association performance to a database as well as its contribution to the pose estimation. To be specific, selecting a sparse feature is more apt to cause mismatch between features, that further results in the performance degradation of the pose estimation [23]; secondly, the image distance between visual features in the database also has impact on the pose estimation, for example, the author in Ref. [21] claimed that the accuracy of pose estimation decreases as the recognized features grow more distant, the same result can also be found in Ref. [24] by a sensitivity analysis. In what follows, we will explain these two metrics in detail.

2.1. Similarity metric-S

In this work, we choose the SURF feature [25]as the visual feature. An example of SURF features detected in planetary image is shown below:

As shown in Fig. 1, the image on the right shows the SURF descriptor, which is a 64 dimensional vector expressed by its characteristic scale and orientation. SURF feature is a speed-up alternative to SIFT [27], the reason we chose SURF over SIFT is its robustness against image noise [25], especially when coping with images with sparse textures (for example, planetary [28]and underwater images [29]).

The dot product of SURF descriptors is used to match features, which is

$$s(F_1,F_2) = \frac{F_1^T \times F_2}{\|F_1\|\cdot\|F_2\|} \tag{2.2}$$

where F_1 and F_2 are two SURF features. This metric can be interpreted as the angle between two descriptors, thus the value of $s(\cdot)$ is approximately proportional to the resemblance of these two features. In planetary image, similar features usually occur around regions with plain texture. The left image in Fig. 1 presents two similar SURF features and their ambient regions (the squared sub-regions), obviously, these two features are less distinctive compared to other features, thus both of them should be excluded from database. Given Eq. (2.2), the similarity metric $S$is defined by

$$S(F,\mathbf{C}) = 1 - \max[s(F,\mathbf{C})] \tag{2.3}$$

where $\max[s(F,\mathbf{C})]$ is the resemblance score of the feature F to its most similar feature in a database $\mathbf{C}$, if feature $F$is easily distinguished from all the other features in $\mathbf{C}$, its similarity metric Swould yield a score near 1.

2.2. Distance metric-D

The author in Ref. [26] claimed that the vehicle pose estimation uncertainty decreases as the features grow increasingly distant. In this section we choose a simple scenario to demonstrate such effect, assuming the camera remains stationary, the Discrete-Time (DT) evolution model can be expressed as

$$\boldsymbol{p}(k) = \boldsymbol{p}(k-1) + \omega(k) \tag{2.4}$$

where $\boldsymbol{p}(k) = [p_x(k), p_y(k), p_z(k)]^T$ is the position of camera at time step k, $\omega(k) \sim \mathcal{N}(0, Q_k)$ is the state propagation noise modeled by zero-mean Gaussian process with covariance Q_k. Assume a landmark (or feature) with global position $\boldsymbol{L} = [l_x, l_y, l_z]^T$ is recognized at time step $k$with the observation model:

$$\mathbf{z}(k) = \begin{bmatrix} \boldsymbol{\mu}(k) \\ \mathbf{v}(k) \end{bmatrix} = f\begin{bmatrix} (p_x(k)-l_x)/(p_z(k)-l_z) \\ (p_y(k)-l_y)/(p_z(k)-l_z) \end{bmatrix} + \nu(k) \tag{2.5}$$

where $\mathbf{z}(k) = [\boldsymbol{\mu}(k), \mathbf{v}(k)]^T$ is the 2D position of a SURF feature, f is the