Landmarks Constellation Based Position Estimation for Spacecraft Pinpoint Landing

Bach Van Pham, Simon Lacroix, Michel Devy

LAAS-CNRS
7 Avenue du Colonel Roche,
31077 Toulouse Cedex 4, France
bvpham@laas.fr, Simon.Lacroix@laas.fr, michel@laas.fr

Abstract: This paper presents a vision-based approach to estimate the absolute position of a planetary lander during the last stages of the descent. The approach relies on the matching of landmarks detected in the descent imagery with landmarks previously detected in orbiter images. The matching process must be robust with respect to scale and radiometry differences in the image data: it mainly relies on the geometric repartition of the landmarks, rather than on radiometric signatures computed from the image signal. First results using a simulator are presented and discussed.

I. INTRODUCTION

Planetary exploration missions may require a precise landing position, in order to avoid obstacles or to get close to scientifically interesting areas assessed on the basis of orbiter images. However, current Entry, Descent and Landing (EDL) techniques are still far from this capability. Hence, much of research has been conducted in the field of absolute localization (referred to as "pinpoint landing") [1][2][3][4][5], which allows the spacecraft to localize itself with respect to a known reference and navigate to a pre-defined landing site.

Related work: Various technologies for absolute localization with respect to initial data for aerial devices have been developed for military purposes (cruise missiles). Such technologies have been modified to safely land a spacecraft on a pre-defined spot on extraterrestrial planets. For example, A vision aided inertial navigation (VISINAV) system is proposed in [1]: surface landmarks in the descent images are extracted and matched to an ortho-rectified image of the scene that is registered with a digital elevation map. The mapped landmarks returned by this process are then used either to estimate or update the spacecraft position. In parallel with these mapped landmarks, the VISINAV system employs the persistent features to track landmarks through a sequence of descent images to estimate the global position of the spacecraft, and the opportunistic features to estimate the spacecraft five degree-of-freedom motion. These estimations along with the IMU measurements are then provided to an Extended Kalman Filter to update the spacecraft status. A main drawback of this system is its high memory requirement due to the usage of image correlation in finding mapped landmarks.

Another pinpoint landing system proposed in [2] uses craters for precise position estimation. Craters are excellent landmarks for the system thanks to their illumination, scale and rotation invariance properties. The crater shapes commonly follow a known geometric model (ellipse), invariant to scale changes and orientation between the spacecraft camera and the surface. Naturally, areas with either too few or too many craters challenge such an approach. Difficulties also arise in regions with blur rimmed, broken, overlapped or overshadowed (with low sun's elevation) craters.

In [3], a pinpoint landing system which uses SIFT features (Scale Invariant Feature Transform) [11] as mapped landmark to estimate the spacecraft's global position is proposed. SIFT detector converts interest points in both the descent image and the geo-referenced orbiter image into feature vectors, which are later matched to each other. Unlike craters, SIFT does not rely on a known model of terrain shape but on the image texture. It neither requires an IMU nor an altimeter like VISINAV to find mapped landmarks. SIFT features are also well-known for their scale invariance properties. However, they are only invariant to an affine transformation in illumination change. Thus, they are quite vulnerable in case of important illumination changes between the orbiter and descent images, such as caused by the difference of the sun’s direction.

Another method proposed in [4] relies on a Lidar sensor and the matching of landmark constellations (star tracker like method). Firstly, surface landmarks, which are either maximum/minimum of the surface's height and slope, are extracted from the surface elevation image provided by the Lidar sensor. The metric distances of one landmark with its neighbors are used as its signature. Surface and DEM (digital elevation map) landmarks matching is performed by comparing the landmark's signature with those of the map. The main drawback of this system is its hardware requirement. In fact, this system requires that the resolution of the surface image obtained with the Lidar sensor must be equivalent to that of the map image. That means the sensor's field of view must be adaptively changed with the spacecraft's speed and altitude to have equal surface sampling step. Another Lidar based approach cited in [5] which uses Range Image to DEM Correlation also suffers from this problem.

The absolute positioning system Landstel (Landmark Constellation) introduced in this paper uses camera as its primary sensor, along with an altimeter and an inertial sensor, with which the lander always equips. Similarly to [4], the Landstel system also uses the relationship of one landmark with its neighbors as its signature.

Paper outline: The next section sketches the overall architecture in which the Landstel system is integrated, and presents the various step of the proposed landmark matching algorithm. Section III depicts these steps and provide some matching results obtained in simulation with PANGU [6]. Section IV briefly explains how the spacecraft's absolute position can be estimation with matched landmarks, and presents some localization results. Section V and VI then respectively discuss the approach and concludes the paper.

II. OVERALL SYSTEM DESCRIPTION

Fig.1 presents the overall navigation system architecture. The Landstel system is composed of one off-line and one on-line part. In the off-line part, the Digital Terrain Map and the associated 2D ortho-image of the foreseen landing area (30x30 km^2 for the current technology [7]) is obtained on the basis of orbiter imagery – this process being naturally out of the scope of our system. Visual landmarks ("initial landmarks") are then extracted in the ortho-image, using the Harris feature point detector [8]. A signature is defined for each of the extracted feature points, according to the process introduced in section III.3. The initial landmarks' 2D position, their signature and their 3D absolute co-ordinates on the surface constitute a database that is stored in the lander's memory.

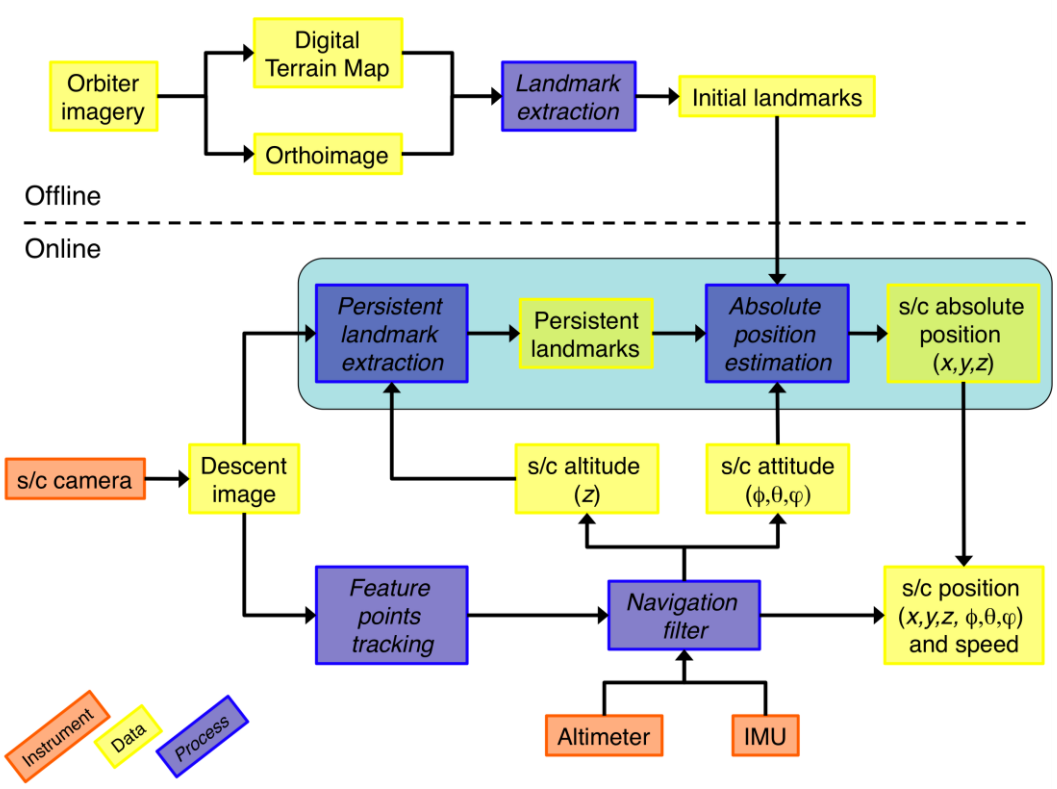


Fig. 1: Overall System Architecture. The "feature point tracking" and "navigation filter" processes are no further described in this paper. The navigation filter process is only sketched here to exhibit its interfaces with the Landstel system (highlighted in blue in the figure): current altitude and attitude estimates are used to assist the landmark matching process, whereas the output of the Landstel system (translation components of the spacecraft absolute position) is fused with the altimeter, IMU and feature tracking processes to produce a global s/c position estimate.

On-line, the current altitude estimate is exploited to extract landmarks (denoted as "persistent landmarks"), and the current s/c orientation estimate is used to warp the landmark coordinates, so as to enable matches with initial landmarks. The Landstel system outputs and estimate of the s/c absolute position, which is used as an input to the navigation filter, the latter exploiting its own visual features to estimate the global s/c state, *e.g.* according to [9]. In the remainder of the paper, the terms "geo-referenced image" and "orbiter image" are interchangeably used – similarly, the terms "landing image" and "spacecraft image" denote the same image.

III. MAPPED LANDMARK MATCHING

III.1 Landmark Detection

Unlike the landmark extraction method used for the geo-referenced image, which is purely a Harris operator, the landmarks of the landing image are extracted with a scale adjustment operator [10]. With the use of an altimeter, the system can detect the difference between scales of the geo-referenced image and the landing one. Similarly to the scale notion introduced in [10], which is the fraction between the resolutions of two images with the same image's sizes and taken with the same camera, the scale notion we used here is its expansion, applied for two images obtained with two different cameras.

Given *(M, M)* the size of a square 2D ortho geo-referenced image, *H* its altitude and *F* its camera's field of view, the geo-referenced image's resolution can be calculated with:

$$R = \frac{\left(H \tan\left(\frac{F}{2}\right)\right)}{2M} (m/pixel) \tag{1}$$

similarly, with *(m,m)* the size of the landing image, *h* its altitude obtained with the altimeter and *f* its camera's field of view, supposing that the camera points nadir, then the landing image's resolution is:

$$r = \frac{\left(h \tan\left(\frac{f}{2}\right)\right)}{2m} (m/pixel) \tag{2}$$

The scale difference between the two images is:

$$s = \frac{R}{r} \tag{3}$$

As stated in [10], the "scale-space associated with an image *I* can be obtained by convolving the initial image with a Gaussian kernel whose standard deviation is increasing monotonically, say $s\sigma$ with $s>1$". In Fig.2, the landmarks which are visible at high altitude are detected in Fig.2.b ($\sigma = 3$), and the landmarks which are invisible at high altitude (shown in Fig.2.a) are discarded in Fig.2.b. The Landstel system, in addition, uses the scale-space pyramid like for the SIFT detection method [11] to compensate the altimeter's error. However, the Landstel landmark detector is less costly than SIFT since it does not calculate the SIFT descriptor.

The usage of scale adjustment operator, on the one hand, helps to remove landmarks in the landing image that have not been extracted in the geo-referenced image. On the other hand, it filters the sensor noise thanks to the use of Gaussian kernel. Fig.2.c shows landmarks detected in an image altered by a white noise (0, 0.01) with $\sigma = 4$.

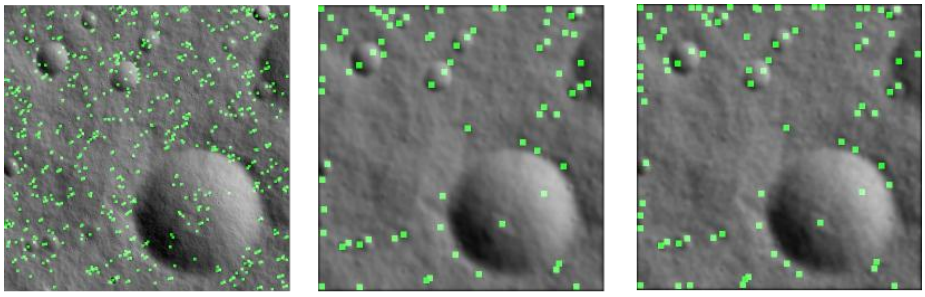


Fig. 2: Landmarks Detection with Scale Adjustment Operator

(a) Normal operator (b) Scale Adjustment Operator (c) Scale Adjustment Operator with Noise

III.2 Landmark Rectification

One they have been extracted from the spacecraft's image with the scale adjustment operator, the interest points are warped to match the orbiter's image's orientation with a homography transformation (Fig.3.a), estimated thanks to the s/c orientation provided by the navigation filter. This step will naturally ease the landmark matching process. Fig.3 shows that the detected landmarks are pretty well matching those of the geo-referenced image thanks to the usage of the scale adjustment operator and the image warping.

III.3 Landmark Signature Extraction

Thanks to the scale adjustment, the detected landmarks have a pixel metric resolution similar to the landmarks of the orbiter's image. The Landstel system uses this property to find matches between landmarks extracted from these two images: the signature the landmark is indeed defined using their relative geometric repartition, measured using a polar grid This is inspired from the PoleStar algorithm, used in a star tracker system [12].

Instead of using the spherical coordinates (right ascension and declination), the Landstel system uses the landmarks pixelic distance. For a landmark L_i, its signature is extracted by using the modified polar grid algorithm, which is composed of the following steps:

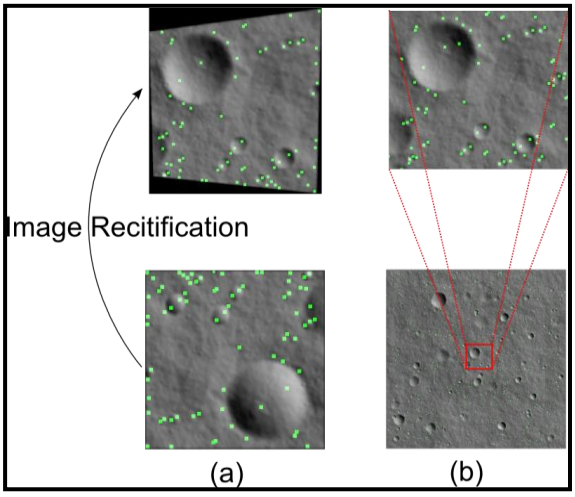


Fig. 3: (a) Rectification of spacecraft image

(b) Corresponding zoomed region in geo-referenced image

1. *Determination of Neighbors set*: a generic landmark L_j can be added to the Neighbors set of L_i if and only if its pixelic distance to L_i, D_{ij}, satisfies the following condition

$$br < D_{ij} < pr \tag{4}$$

where *br* is the minimum distance which is used to prevent noise and *pr* is the pattern radius (Fig.4.a and 4.b, extracted from [12]).

2. *Angular Distances Discretization and Bar-Code Generation*: the landmarks pixelic distances in the Neighbors set of L_i are then discretized into only *g* values, corresponding to *g* rings centered on L_i , the g-size array signature p_i of L_i is given by:

$$p_i\left[ceil\left(\frac{D_{ij}-br}{pr-br}*g\right)\right] = 1 \tag{5}$$

in which *ceil* is a round operator which rounds the element towards the nearest integer.

Therefore, for a landmark, its signature is a vector of *g* bits where a bit is assigned to 1 if there is at least one landmark in the corresponding ring (Fig.4.c and 4.d, extracted from [12]).

Notes:

1. In this algorithm, the use of the inner radius *br* is to avoid noise generated in the landing image's landmarks. Let δ the minimum distance between two landmarks, which is defined for both geo-referenced lander images. In the Landstel system, the minimum distance *δ'* of two rectified landmarks will be smaller than *δ*. Therefore, *br* will be used to reject those rectified landmarks which are too near to the current landmark L_i.
2. The value of *pr* is chosen considering the rectified image's size of the last image in the test sequence. For example, the last image of the first experimental series (section IV) is acquired at 3200m altitude. Thus, its rectified image will have approximately 286x286 pixels. In this case, *pr* is assigned to 100, which is used for the whole series.
3. The value of *g* (grid size) depends on the uncertainty of the landmarks' position, which is influenced by the noise on the image and the attitude.
4. Similarly to the strategy used in [12], the landing image's landmarks which are too near to image's border are not used to test/match in the system.

III.4 Landmarks Matching

In order to calculate the similarity between two landmarks, the Hamming distance between the two landmarks signatures is calculated. The couple whose distance is smaller than a threshold is considered as a potential match. Therefore, each landmark L_i in the landing image may have multiple possible matches. During this step, the spacecraft's landmarks and their corresponding possible matches are memorized in a data structure, called *matchIndex*.

III.5 Multiple hypothesis matches removal

The multiple possible matches found in the previous steps are filtered out with the

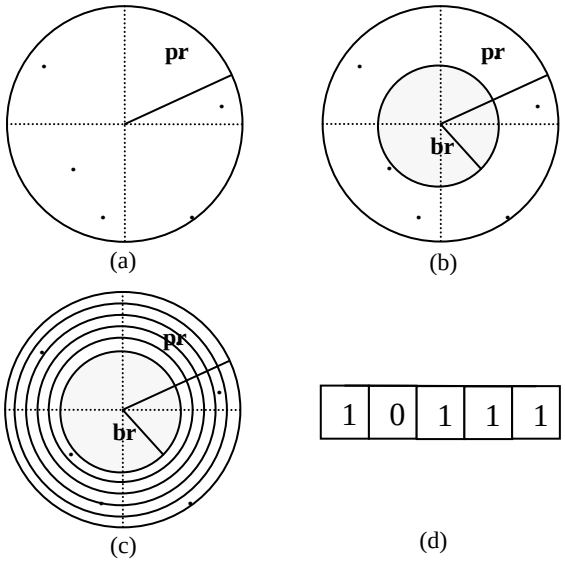


Fig. 4: Polar Grid Algorithm Principles

Multiple hypothesis matches removal procedure. The main idea of this procedure is that two corresponding sets of matched points must represent the same constellation (similarly to the case of star tracker). Therefore, the system will firstly find a set of 5 consistent matches. The search of these consistent matches is based on the vector distance. Given two couples (L_i,K_i) and (L_j,K_j) where L and K respectively represent the spacecraft's and orbiter's landmarks, these two matches are considered as consistent if and only if their vector distance $distVector([L_i,L_j],[K_i,K_j])$, defined as the difference between the two vectors lengths, is smaller than a predefined threshold $\mathcal{E}$. This vector distance is meaningful because the two landmark sets share the same coordinate system. Therefore, the only unknown factor is the translation value of the two sets.

The *Multiple hypothesis matches removal* procedure encompasses the following steps:

1. Find at least 5 consistent matches between the spacecraft's landmarks and the orbiter's landmarks using the vector distance. If the system cannot find at least 5 consistent matches, it will report a failure.
2. After having found 5 consistent matches, the affine transformation *AF* between the spacecraft's rectified landmarks and the orbiter's landmarks is calculated.
3. Apply the AF transformation calculated to every spacecraft's rectified landmarks. The result of each transformation is compared with the orbiter's landmarks. The couple which is consistent in orbiter's pixelic position is retained as a match, which means that their difference in position is smaller than $\mathcal{E}$.

The *Multiple hypothesis matches removal* can be described with the following pseudo code:

Let *mMatchIndexSize* the size of *matchIndex*.

1. Set mResult = 0 ; mMatch = NULL ;
2. For i = 1 : mMatchIndexSize - 4
- *a. Get landmark L_i in matchIndex*
- *b. Set nCouple = 0 ;*
- *c. For each possible match of Li (let K_i the current possible match)*
  - *i. For j = i + 1 : mMatchIndexSize*
    1. *Get landmarks L_j in matchIndex*
    2. *For each possible match of L_j (let K_j the current match)*
       - *a. If $distVector([L_i,L_j],[K_i,K_j]) < threshold\ \mathcal{E}$*
         - *i. Store couple (L_i, K_i) and (L_j, K_j) in mMatch, increase nCouple by 1*
         - *ii. If (nCouple > 4), set mResult = 1, go to step 4*
         - *iii. Else go to step 2.c.i*
       - *b. End*
    3. *End*
  - *ii. End*
- *d. End*

3. End
4. If (mResult == 1)
- *a. Compute the affine transformation AF from stored couple in mMatch*

b. *Using the transformation AF, compute the transformed position P_i for each spacecraft's landmarks. A landmark in the geo-referenced image is a match of P_i if and only if their difference in positions is smaller than $\mathcal{E}$.*

5. *Else report Failure*

Fig.5 shows the found landmarks matches with different illumination condition. The geo-referenced image (left side) is acquired with 55-25 (azimuth-elevation) Sun position while the spacecraft images (right side) are acquired at 5710m altitude with 145-10 Sun angle (a) and at 3052m altitude with 235-10 Sun angle (b). The center images show the corresponding match region of the right-side images in the geo-referenced image. The spacecraft images are rotated for comparison purpose.

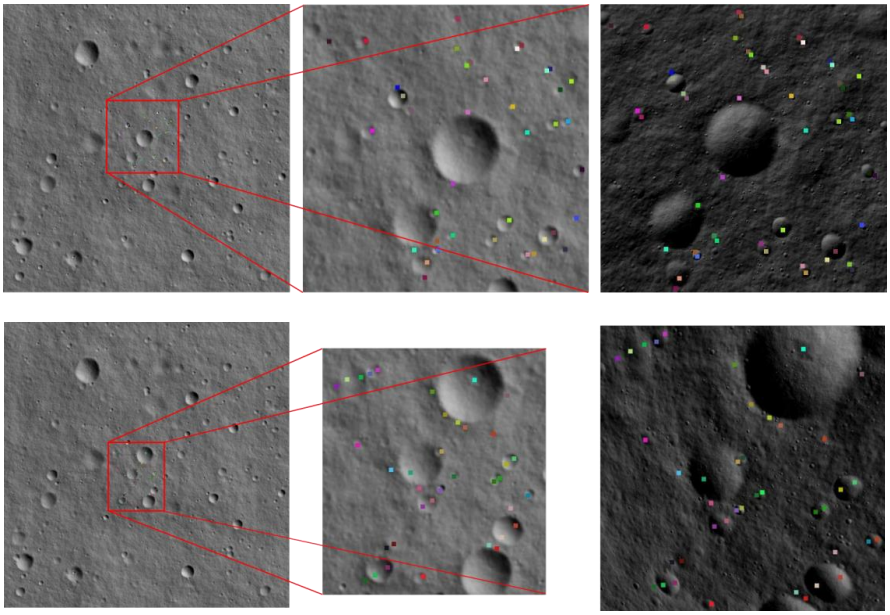


Fig. 5: Landmark Matches before the Multiple Hypothesis Removal process (top), and after (bottom)

IV. RESULTS

IV.1 Spacecraft Position Estimation

Given a set of matches between the landing image and the geo-referenced image, the spacecraft position can be estimated by using:

1. The landmarks 2D positions (*U*) in the landing image
2. The landmarks 3D positions (*M*) in the landing zone (deduced through their matches with the geo-referenced image) expressed in a known coordinates
3. The image projection function:

$$U = K[R, -RT]M$$

where K is the 3x3 intrinsic matrix of the camera, R the image rotation (provided by the navigation filter) and T the s/c position (Fig.6): for simplification the spacecraft reference frame is assimilated to the camera one here.

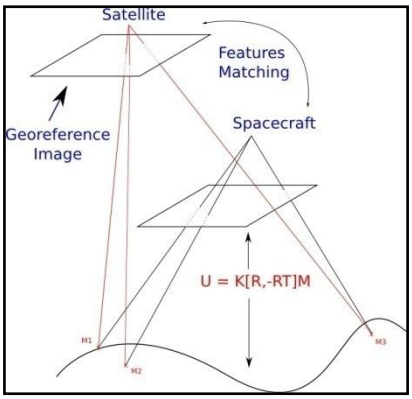


Fig. 6: S/C Position Estimation

Knowing U, K, R and M, the spacecraft position T can be calculated using a linear or non-linear optimization algorithm (e.g. Levenberg-Marquardt).

IV.2 Absolute position estimation results

The Landstel landmarks matching module has been tested with the PANGU simulator v2.7. In this experiment, instead of using a simulated trajectory of a spacecraft during its entry phase (from the early parachute phase to the end of powered guidance phase), images are independently taken from 5000m altitude down to 1000m altitude. Each image is independently matched with the geo-referenced image, using the altitude and attitude information provided by the navigation filter as an input – the resulting absolute position estimates are the core output of Landstel, and are not further fused by the navigation filter.

The purpose of this experiment, on one hand, is to verify the robustness of the Landstel system with different

altitudes. On the other hand, the experiment shows how to choose the corresponding geo-referenced image with respect to the spacecraft's altitude considering the scale robustness. In reality, one very high resolution image can hardly match with a coarse image. Therefore depending on the landing image's resolution, a corresponding geo-referenced image will be chosen for the matching step.

More precisely, there are two geo-referenced images used in this experiment. The first one is taken at 182280m altitude with a camera of 10 degrees FOV and 2048x2048 pixels. The second one is also taken at 182880m altitude with a 5 degrees FOV camera and 2048x2048 pixels, which gives a two times finer resolution. Two corresponding landing image sets (68 images for each set) have been generated with PANGU to be matched with these geo-referenced images. The first series is taken from 5000m to 3000m and is matched with the first geo-referenced image; the second is taken from 2500m to 1050m and matched with the second geo-referenced image.

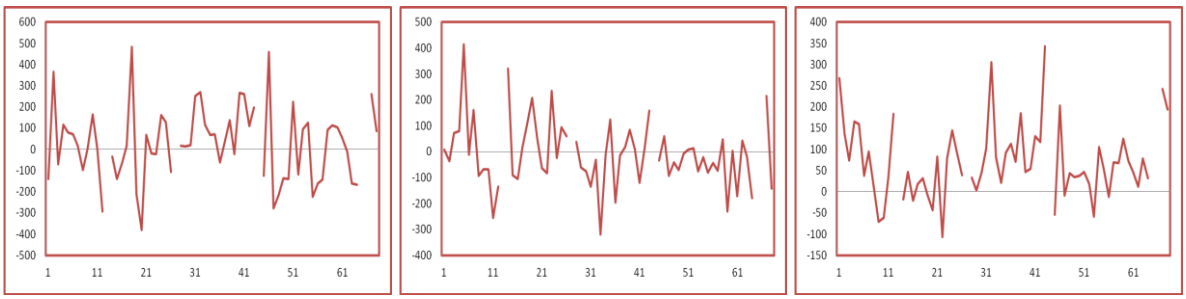
In order to verify the robustness of the system with different noise sources like illumination difference, camera noise, and sensor noise, the experiment is set up with the following configuration:

1. *Image illumination:*
    a. *Geo-referenced image: 55 deg in azimuth, 25 deg in elevation (sun)*
    b. *Landing image: 40 deg in azimuth, 20 deg in elevation (sun)*
2. *Image noise: white noise (0,0.007)*
3. *Sensor noise:*
    a. *Radar altimeter: 2.5% of measured distance (e.g. 5000m altitude with ±125m error)*
    b. *Gyroscope: error in margin ±5 degrees*

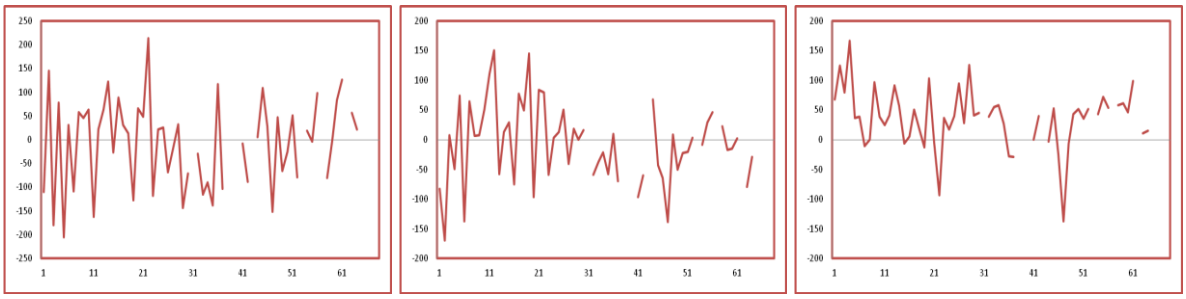
The *Landstel* system's parameters are set as:

1. *pr = 100 (outer radius)*
2. *br = 10 (inner radius)*
3. *g = 24 (grid size)*
4. *$\mathcal{E}$ = 5 (distVector threshold)*

Results of absolute position estimates are shown in Fig.7. The difference between the estimated value (X, Y, Z) and the true spacecraft position (ground truth) is calculated for each image. Errors in X, Y and Z are respectively displayed in the left, center and right columns. Fig.7.a shows the result of the first image series (5000-3200m). Fig.7.b shows the result of the second image series (2400-1050m). The gaps presented in the charts mean that (1) there are no correct matches found or (2) the estimated position is not consistent with the previous estimation.



(a)



(b)

Figure 7 Landstel Result (a) 1st series 5000-3200m (b) 2nd series 2500 – 1050m

V. DISCUSSION

The proposed landmark matching approach shows to be robust to different sources of error like illumination, camera noise or sensor noise – results shown Fig.7 are quite promising. From a huge uncertainty in the spacecraft

position (32000m), the system can localize the spacecraft within a much smaller uncertainty ±400m for the 1st series and ±200m for the 2nd series.

As shown in Fig.7.a or in Fig.7.b, the estimation error doesn't reduce significantly with the altitude. The reason for this phenomenon is that all images in the first (or in the second) series, are matched with the same geo-referenced image. More precisely, the image taken at 5000m and that taken at 3200m are all matched with the first geo-referenced image, whose resolution is approximate 15.6m/pixel, which is equal to the resolution of a spacecraft image taken at 5700m. In fact, the latter images in the series are harder to match with the geo-referenced image than the early images. The reason is that the further the spacecraft descends, the bigger the difference between the spacecraft's image and the orbiter's image is. The estimation error is, as a consequence, reduced in the second series when all images are matched with a more precise geo-referenced image, 7.8m/pixel in resolution. The system precision would therefore improve with a finer geo-referenced image.

As shown in Fig.5, the *Landstel* system is robust with respect to very different illumination conditions between the spacecraft's image and the orbiter's one. This illumination robustness is obtained thanks to the use of a landmark signature solely computed on the basis of geometric relations instead of radiometric information.

The Landstel system is also efficient in memory usage. In fact, for each landmark in the geo-referenced image, the system only needs 3 bytes (24 bits) for the signature and 3 float numbers (4 bytes for each) for the point's 3D position. Therefore, with 1500 points for the first geo-referenced image, the system only requires 22.500 bytes (22Kb). The second orbiter's image, which is 4 times larger than the first image, only requires 88Kb. The third orbiter's image, if it existed, will require approximately 352 Kb for its data. Therefore, the whole system will only need 462Kb to store the whole landing region's information (32x32km^2).

VI. CONCLUSION

In this paper, we have introduced a vision-based algorithm for spacecraft localization. In the first step, the algorithm uses a scale adjustment operator to detect the correct scaled features which are visible when viewing from high altitude. Then, the geometric relationship between one landmark with its neighbors is used to establish its signature. Hamming distance between the spacecraft's landmark's signature with that of geo-referenced image is used to filter possible matches, which results in multiple hypothesis matches. The algorithm, by using vector distance of landmark couples, can then discard incorrect matches. Finally, an affine transformation between these correct matches is calculated. Further matches are found thanks to the computed affine transformation. Through the experiment setup, the algorithm has shown its robustness to different sources of noise thanks to the use of geometric features. The proposed algorithm is also efficient in term of memory usage.

V. ACKNOWLEDGMENTS

This work is financially and technically supported by ESA and by EADS-ASTRIUM.

VI. REFERENCES

[1] Nikolas Trawny, Anastasios I. Mourikis, and Stergios I. Roumeliotis, "Coupled Vision and Inertial Navigation for Pin-point Landing", *NASA Science Technology Conference, 2007.*

[2] Yang Cheng and Adnan Ansar, "Landmark Based Position Estimation for Pinpoint Landing on Mars", *International Conference on Robotics and Automation*, 2005.

[3] Nikolas Trawny, Anastasios I. Mourikis, Stergios I. Roumeliotis, Andrew E. Johnson and James Montgomery, "*Vision-Aided Inertial Navigation for Pin-Point Landing using Observations for Mapped Landmarks*", Journal of Fields Robotics, 2006.

[4] Jean-Francois Hamel, David Neveu and Jean de Lafontaine, "Feature Matching Navigation Techniques for Lidar-Based Planetary Exploration", *AIAA Guidance, Navigation and Control Conference and Exhibit, 2006.*

[5] Andrew E. Johnson and James F. Montgomery, "Overview of Terrain Relative Navigation Approaches for Precise Lunar Landing", IEEE *Aerospace Conference, 2008.*

[6] S.M Parkes, I. Martin, M. Dunstan and D. Matthews, "Planet Surface Simulation with PANGU", *SpaceOps 2004.*

[7] Aron A. Wolf, Claude Graves, Richard Powell and Wyatt Johnson, "Systems for Pinpoint Landing at Mars", *14th AIAA/AAS Space Flight Mechanics Meeting, Maui, Hawaii, February 8-12, 2004.*

[8] *C. Harris and M. Stephens, "A combined corner and edge detector". Proceedings of the 4th Alvey Vision Conference: pp 147—151, 1988.*

[9] EADS-ASTRIUM, Galileo Avionica, University of Dundee, Ineti and Scisys, "Navigation for Planetary Approach & Landing", Final Report, 2006.

[10] Yves Dufournaud, Cordelia Schmid and Radu Horaud, "*Image matching with scale adjustment*", INRIA Report, RR-4858, 2002.

[11] David Lowe, "Object recognition from local scale-invariant features", *ICCV, 1999.*

[12] E. Silani and M. Lovera, "Star Identification Algorithms: Novel Approach & Comparison Study", *IEEE Trans on Aerospace and Electronic Systems, Vol. 42, No. 4, Oct 2006.*