Digital Elevation Map Building from Low Altitude Stereo Imagery

Simon Lacroix, Il-Kyun Jung and Anthony Mallet

LAAS / CNRS
7, Ave du Colonel Roche
F-31077 TOULOUSE Cedex
FRANCE
Firstname.Name@laas.fr

Abstract

This paper presents a method to build fine resolution digital terrain maps, on the basis of a set of low altitude aerial stereovision images. Stereovision images are the only data available: to build a global map, we introduce an algorithm that uses interest point matches between successive images to estimate their relative position. Experimental results are presented and discussed, they show how one can build thousands square meters fine resolution maps, integrating hundreds of stereovision images acquired by a tethered blimp.

1 Introduction

In all the applications contexts where the development of exploration and intervention robotics is considered, air/ground robotic ensembles bring forth several opportunities from an operational point of view [1]. Be it for planetary exploration, environment monitoring and surveillance, demining or reconnaissance, a variety of scenarios can be envisaged. For instance, drones or airships can operate in a preliminary phase, in order to gather informations that will be used in both mission planning and execution for the ground vehicles. But one can also foresee cooperative scenarios, where aircrafts would support ground robots with communication links and global informations, or where both kinds of machines would cooperate to fulfill a given mission.

In this context, we recently initiated an internal project concerning the development of autonomous airships [2], in which two issues are currently considered. The first one is related to *airship flight control*: we aim at developing functionalities to endow an airship with the capacities to execute pre-defined trajectories on the basis of GPS and attitude informations, and we would also like to give the airship commands at a bit higher level of abstraction, such as `"follow the vehicle on the ground"` or `"follow this feature - road, river,electric line..."`. The development of such functionalities calls for the establishment of a dynamic model of airships and the study of their controllability [3]. The second issue is related to *environment modeling using images acquired on-board the airship*. Such a functionality is required to send the airship commands defined with respect to the environment, but is also critical to tackle the effective *cooperation* between the airship and ground robots, which calls for the building and share of common environments representations between the two kinds of robots.

This paper presents some preliminary results related to the generation of fine digital elevation maps (DEM), on the *sole basis* of stereovision images acquired at low altitude. DEMs appear to be an interesting representation to share between ground and aerial robots, as they are very convenient to manipulate, and are able to catch the gist of the perceived surface geometry (for instance, several feature detection algorithms can be applied on such maps [4, 5]). DEMs are of course very common in geographic information systems, for which low resolution maps are usually derived from aerial or satellite imagery (radar, lidar or vision - *e.g.* see [6]). Roboticists also paid a lot of attention to the building of such maps [7, 8, 9], but to our knowledge only a few contribution are related to the building of *very high resolution* DEMs using low altitude imagery [10].

The paper is organized as follows: the next section describes our experimental setup (a tethered blimp) to acquire low altitude stereovision images. Section 3 presents the problems related to DEM building whith such images, and

describes our algorithm. Section 4 is devoted to the precise estimation of the blimp displacements between the various stereovision acquisitions, using an interest point matching algorithm. Section 5 presents some map building results, and a discussion concludes the paper.

2 Experimental setup

To acquire low altitude data, we equipped a $25m^3$ tethered blimp with a stereo bench (figure 1). The stereo bench is composed of two black and white 752×582 pixels CCD cameras; the baseline is approximately 1.20 meters. Calibration of the bench has been made possible with a set of images pairs of a calibration frame laid on the ground. A few hundreds of stereo pairs have been acquired over an approximately $3000m^2$ surface, at altitudes varying from 10 to 40 meters, during three different acquisition runs. The surface over which the blimp flew contains elements of various natures: rocky areas, grass areas, fences, tall trees and bushes, a small building and a parking lot (figure 1).

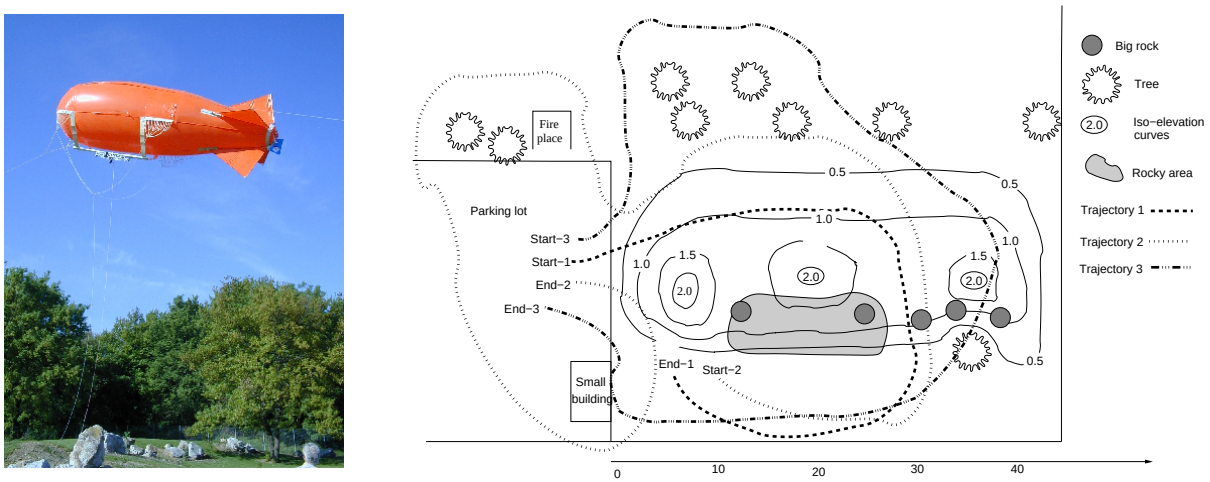

Fig. 1: The tethered balloon we used to get low altitude stereo images (left), and the three acquisition runs drawn on a sketch map of the area (right)

Images were transmitted to a frame grabber located on the ground via a $50m$ video cable, and are therefore of quite poor quality. Also, the video cable, the tether and the operator that moved the balloon can be seen on almost all images (figure 2): we will see in section 5 that it does not distrub too much the mapping algorithm.

Fig. 2: Three examples of images taken from the blimp. From left to right: view of the rocky area, view of the fire place (note the tree and its shadow on the top-left part of the image), and view of the roof of the small building (note the shadow of the building).

Stereovision. Our stereovision algorithm is a classical pixel correlation algorithm [11]. A dense disparity image is produced from a pair of images thanks to a correlation-based pixel matching algorithm (we use either the ZNCC

criteria or the census matching criteria), false matches are avoided thanks to a reverse correlation. With good quality images, the reverse correlation phase is enough to remove the few false matches that occur. But when the images are not of very good quality, three thresholds defined on the correlation score curve are used to discard false matches (on the value of the highest score, on the difference between this score and the second highest peak in the curve, and on the "sharpness" of the peak). Although the images acquired from the blimp are not of very good quality, setting these thresholds was easy. However, the introduction of these thresholds removes good matches, especially in low textured and shadow areas, in which noise is higher than luminance variations. Figure 3 presents some results of the algorithm on image pairs acquired with the blimp.

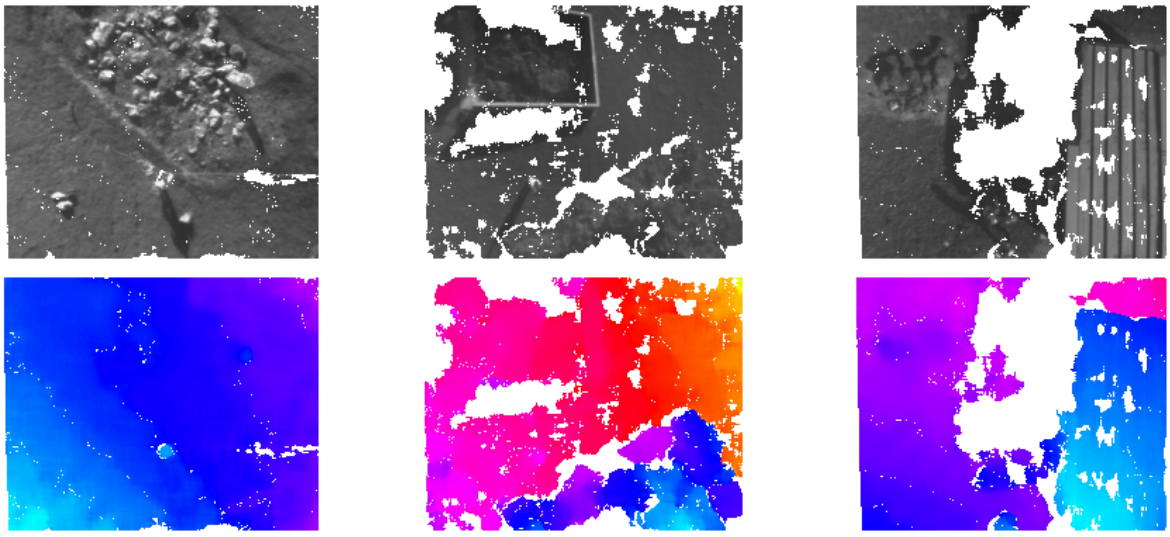

Fig. 3: Results of the stereovision algorithm on the three example images of figure 2. Original images where non-matched pixels are white are on the top line, disparity images are on the bottom line. The disparity is here coded with grey levels: the darker the pixels, the closer they are to the cameras. The left result is an ideal case, where most of the pixels are matched. The middle and right examples show that the stereovision algorithm hardly find matches in the shadowed areas.

3 Digital elevation maps

Although there has been several contributions to the problem of building digital elevation maps with data acquired from rovers [8, 9], we think that it has still not been addressed in sufficiently satisfactory way. The main difficulty comes from the uncertainties on the 3D input data, that can hardly be propagated throughout the computations and represented in the grid structure. Figure 4 presents the problem for various cases of sensor configurations: in the 3D data, the range (depth) coordinate is the most unprecise, especially in the case of stereovision, where this uncertainty grows quadratically with the distance. To take into account these data properties, the best representation would therefore be a 3D occupancy grid [12].

But one can see on figure 4 that the problem is better conditioned when using aerial images looking downwards. Not only the data resolution on the ground is more regular, but also the uncertainties on the 3D data "fits" better a representation of the uncertainties in the digital map: the uncertainties in the data can be fairly well estimated by a standard deviation on the cell elevation.

Considering these properties of the data, our algorithm to build a DEM therefore comes down to computing the elevation of the cell by averaging the elevations of the 3D points that are projected in the cells. The standard deviation on the elevation is also straightforwardly computed, and since to each 3D point is associated a luminance value, it is also possible to compute a mean luminance value for each map cell (figure 5).

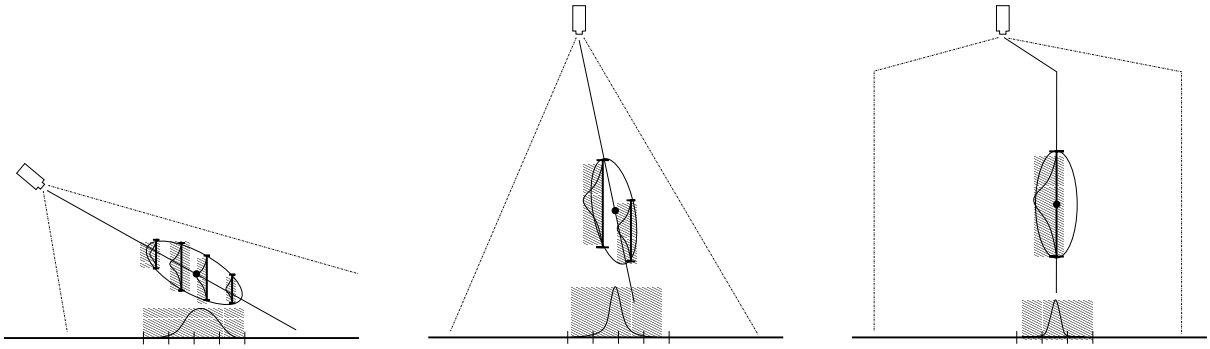

Fig. 4: Sensor errors with respect to an horizontal grid. In the leftmost case, which is the one encountered with range images acquired from a rover, the precision of the data, which is essentially in the range estimate, should be transformed into an occupancy probability in the DEM cells. In the rightmost case, which correspond to data acquired from a flying device, the problem is much better conditioned: the precision on the data can be directly represented by a precision on the elevation computed for the cells. The middle case corresponds to a very low altitude acquisition, which is the case of our images: in our algorithm, we however consider that we are close to the rightmost case. One can also guess on these figures that the variable resolution of the data on the ground play an important role for ground rovers, and is much more regular with aerial images.

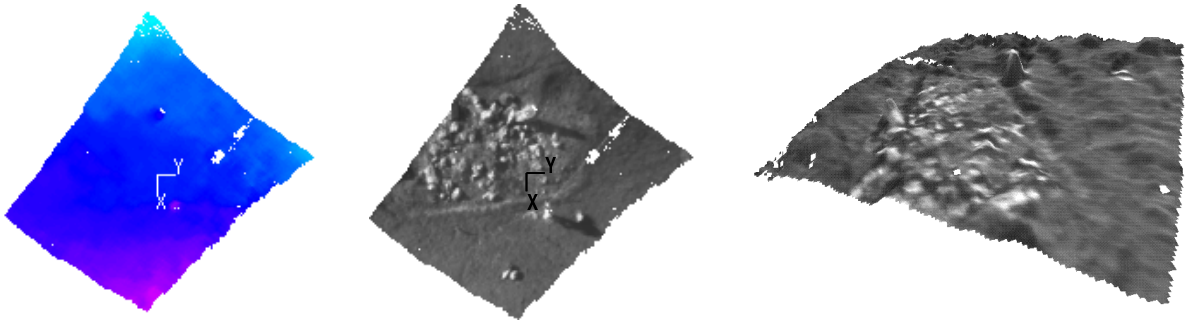

Fig. 5: A digital elevation map computed with a single stereovision image (the leftmost one of figure 3). From left to right: top view of the elevations coded in grey levels, top view of the luminance values stored in the DEM, and 3D view. The horizontal plane of the DEM is defined orthogonally to the view axis of the stereovision image considered.

4 Motion estimation

Since the blimp is not equipped with any positioning device, one must determine the relative positions of the system between successive stereo frames in order to build a global digital elevation map. For that purpose, we adapted a motion estimation algorithm initially developed for ground rovers [13, 14] (a similar algorithm can be found in [15]). The technique is able to estimate the 6 parameters of the robot displacements in any kind of environments, provided it is textured enough so that pixel-based stereovision works well: the presence of no particular landmark is required.

4.1 Principle of the algorithm

The motion parameters between two stereo frames are computed on the basis of a set of 3D point to 3D point matches, established by tracking the corresponding pixels in the image sequence acquired while the robot moves (figure 6 - as in stereovision, pixels are tracked using either the ZNCC or the census matching criteria). A lot of attention is paid to the selection of the pixel to track: on one hand, in order to avoid wrong correspondences, one must make sure that they can be faithfully tracked. This is done thanks to the application of an autocorrelation function on the image, that gives for every pixel an indicator related to the expected *precision* of the tracking algorithm. On the other hand, in order to have a precise estimation of the motion, one must choose pixels whose corresponding 3D point is known with a good accuracy. For that purpose, we use an error model of the stereovision algorithm, that uses the sharpness of the correlation curve as an indicator of the precision of the computed 3D coordinates [16].

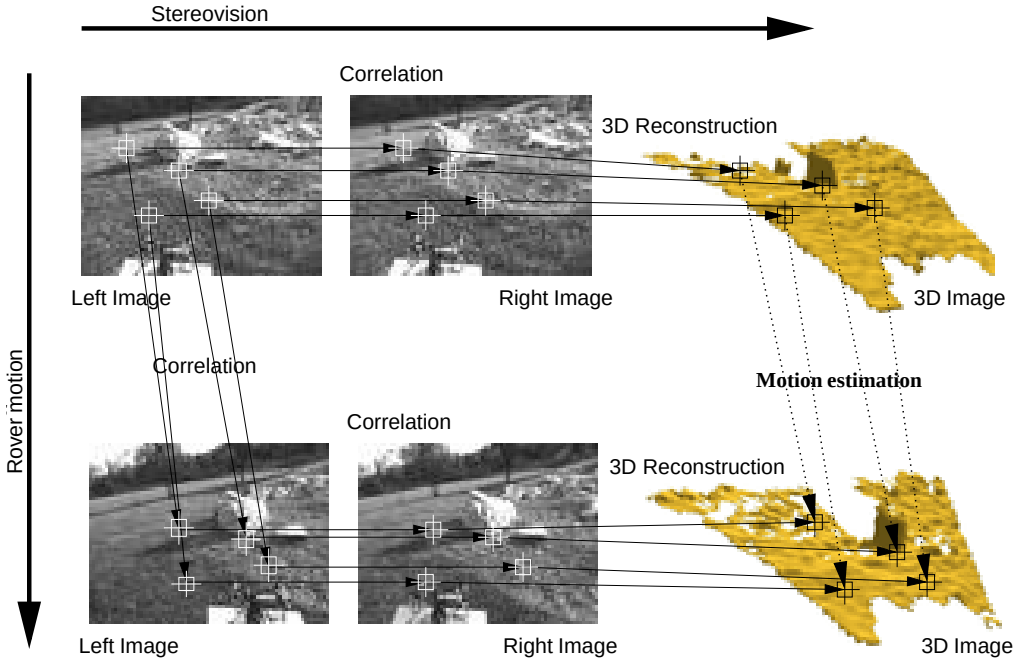

Fig. 6: Principle of the visual motion estimation technique. The steps of stereovision go from left to right, time goes from top to bottom. Given a stereovision image pair, a set of pixels are selected on the left image. They are tracked and matched in the new stereovision frame: this produces 3D points associations (right - the 3D images are represented as DEMs for readability purposes), from which the motion is estimated.

4.2 Establishing pixel correspondences between images

Tracking the selected pixels in an image sequence requires an estimate of the 3D motion of the camera to focus the search zone, to reduce both the computation time and the probability to establish wrong correspondences. With our ground rover, 3D odometry can provide this estimate, but there is no way to have such an estimate with the images taken from the blimp. To establish pixels matches between the stereovision frames, we therefore use an algorithm that matches interest points detected on a pair of grey level images taken from arbitrary points of view [17]. The algorithm provides dense matches and is very robust to outliers, *i.e.* interest points generated by noise or present in only one image because of occlusions or non overlap. This section summarizes this algorithm.

General principle: Once interest points have been computed for two successive images, first matching hypotheses are generated using a similarity measure of the interest points. Hypotheses are confirmed using local groups of interest points: group matches are based on a measure defined on an affine transformation estimate and on a correlation coefficient computed on the intensity of the interest points that are consistent with the estimated affine transformation. Once a reliable match has been determined for a given interest point and the corresponding local group, new group matches are found by propagating the estimated affine transformation (figure 7).

Interest points: The Harris detector [18] uses Gaussian functions to compute the derivatives of intensity, and the two eigenvalues of the auto-correlation matrix as the principal curvatures of the auto-correlation function. It is one of the most used: in [19], the authors compared its stability with respect to other detectors using a quantitative evaluation criteria, *repeatability*, which is the percentage of repeated interest points between two images, and assessed that the precise Harris detector is the one that provides the best repeatability.

We use *single* scalar value, the "cornerness" c_P, as a characteristic of an interest point P. It is defined using the two eigenvalues (λ_1, λ_2) of the auto-correlation matrix: $c_P = \| \lambda_1^2 + \lambda_2^2 \|$.

To measure the resemblance between two repeated points P and Q in two images, the *similarity* $S(P,Q)$ is used: $S(P,Q) = \frac{min(c_P, c_Q)}{max(c_P, c_Q)}$. Figure 8 shows the evolution of the mean of the similarity of repeated points for an example image under various known rotation and scale changes. Repeatability is over 80% for any rotation, and decreases down to 30% percent with a scale factor change of 1.5. For these transformations, the mean of similarity is always over 70%, and the standard deviation is not greater than 12%.

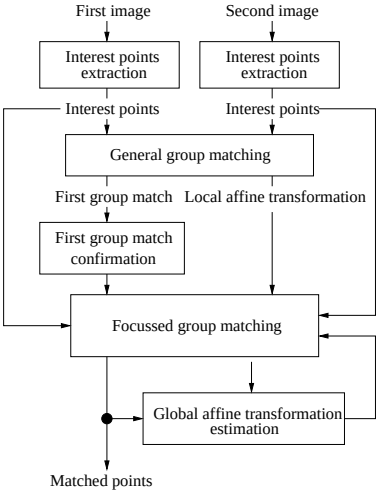

Fig. 7: The various steps of the interest points matching algorithm.

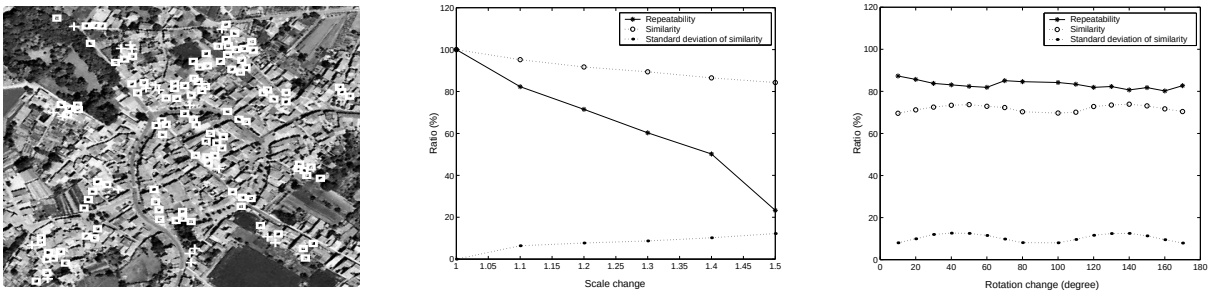

Fig. 8: Evolution of the interest points repeatability, mean similarity and similarity standard deviation with known rotations and scale factor changes of the image shown on the left.

Group matching procedure: On the basis of the similarity measure between two interest points, first matching hypotheses are generated. They are confirmed by a group matching procedure, which relies on the assumption that if two interest points match, their cornerness are similar and their close neighbors are also very likely to be matched together: the repeatability between matched groups is higher than for any other group pair.

The procedure goes as follows: given an interest point in the first image, all the interest points in the second image whose cornerness is similar are matching candidates. To determine which candidate is the good match, local groups around the considered point and all the candidates are built. For each candidate group, the local affine transformation that yields the highest repeatability is determined. The repeatability criteria is not sufficient to establish a valid group match: group matches are confirmed by the estimation of correlation coefficient computed on the intensity of the repeated points in matched groups (details of this procedure, which is the heart of the matching point algorithm, are presented in [17]).

Matches propagation: Once a reliable group match is found, a focussed group matching procedure is activated. Although the local affine transformation found for a group match is not globally stable on most scenes, it is locally stable. Therefore, it is propagated around the current group match found, in order to focus the search of match candidates, thus reducing both the number of candidates to check and the possibility of false matches occurrences.

False matches elimination: To discard the false matches (outliers) that might have occurred in the whole process, the essential matrix is computed on the basis of the matches. For that purpose, a median least square algorithm is used, so that the essential matrix is not affected by the presence of outliers. The application of a simple threshold on the distance between the matched pixel and their corresponding epipolar line removes the majority of the outliers.

Figure 9 shows some results of the interest points matching algorithm on some images acquired with the blimp: not surprisingly, no interest points are matched in the shadow and low texture areas. This is essentially due to the poor image quality, that generate unstable corners in such areas. Fortunately, the algorithm is extremely robust[1], and is not

[1] Ir has been tested with hundreds of images of various 3D scenes, taken in various conditions and environments.

affected by this.

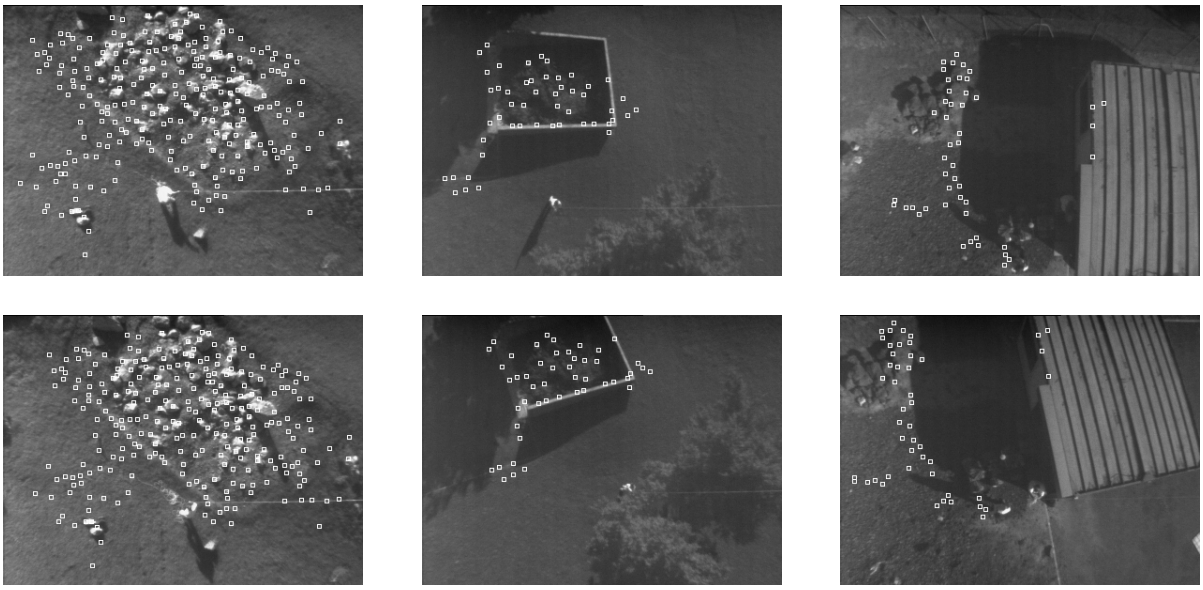

Fig. 9: Three results of the interest point matching algorithm, applied between the example images of figure 2 and the following ones in the acquisition sequence. The white squares indicate the interest points that were matched. Note that the viewpoint changes in the middle and right images are quite important: nevertheless, no erroneous matches were established.

4.3 Computation of the relative motion

Thanks to the interest point matching algorithm, a set of 3D points associations between two consecutive stereovision frames is available. Estimating the displacement between the two frames then simply comes down to computing the 6 motion parameters (translation and rotation) that minimizes a distance between matched points. Since there remain only a very small minority of outliers in the matches, a least-square minimization technique is very effective [20].

5 First results

Thanks to these three algorithms (stereovision, motion estimation and map building), we could build various digital elevations maps integrating several tens of stereo images. In the absence of any attitude estimation device on-board the blimp, the plane of the DEM is defined orthogonally to the view axis of the first stereovision images considered.

Figure 10 presents a digital elevation map built with 120 stereovision pairs, covering about $1500m^2$, with a cell resolution of $5 \times 5cm^2$. The trajectory executed by the blimp, which could be recovered using the localization algorithm, is an about $100m$ long loop. The last images overlaps the first images, and no discrepancies can be observed in the final model on this area: the localization algorithm gave here an *extremely precise position estimate*, with a final error of the order a map cell size, *i.e.* about 0.1%. Figure 11 show two maps built with images corresponfing to the second trajectory, and figure 12 show the map built with all the images of the third trajectory.: the position estimation in this latter case drifted of about 1%.

Note that one can distinguish in all this figures the various positions of the operator, that produces peaks in the elevation map and whitish areas in the luminance values (the operator wore a white shirt). This is because the algorithm that fills the map simply updates the cells without any consistency check. Such artifacts can be easily removed by checking if the elevation of the points of the 3D image to fuse is consistent with the current elevation and corresponding standard deviation of the map cells. Unconsistent 3D points correspond to moving parts in the environment, and can just be discarded.

All the images were processed off-line, since no CPU was on-board the blimp. However, the computations times are compatible with an on-line implementation, most of the time being consumed by stereovision and interest points matching, which both take about one second on an Ultra-10 Sparc CPU or a G3 PowerPc.

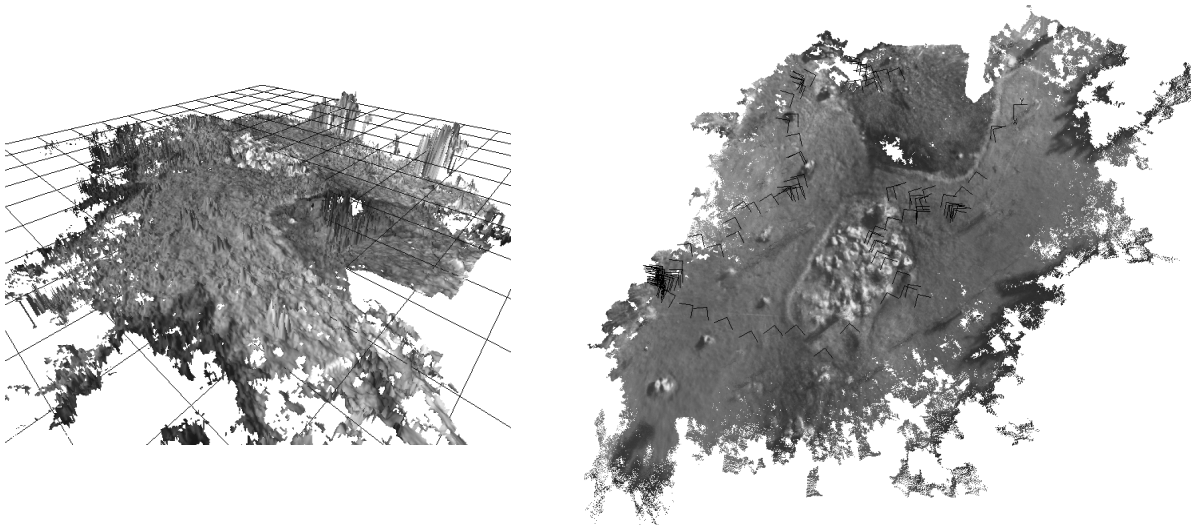

Fig. 10: Final digital elevation map produced with the 120 stereovision images corresponding to the first trajectory in the sketch map of figure 1. Left: 3D model, in which some peaks are artifacts due to the presence of the moving operator. Right: a top view of this model, which is actually an ortho-image of the terrain. The "vertical" projection of the blimp positions are shown as small black frames in this image.

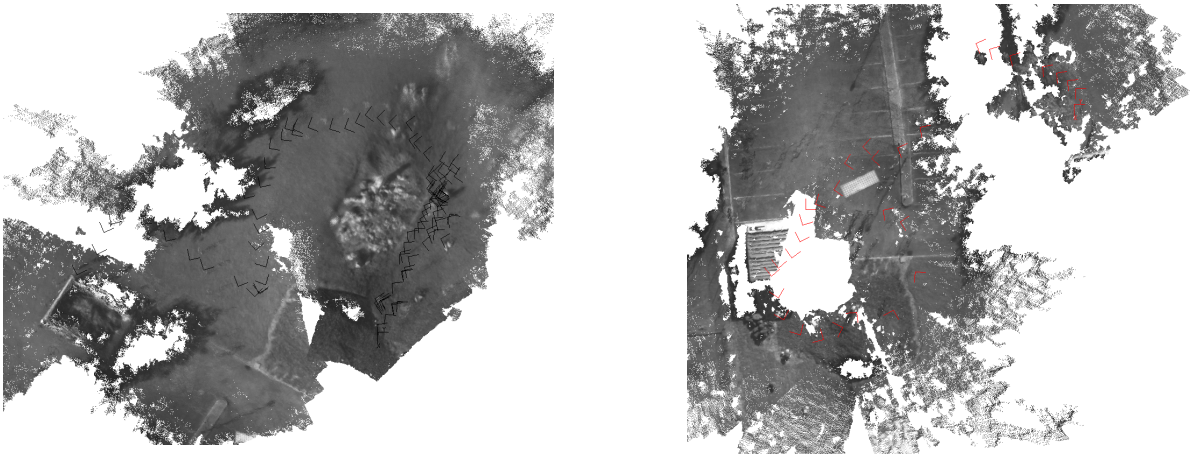

Fig. 11: The two ortho-images corresponding to DEMs built from images of the second trajectory in figure 1.

6 Discussion

Summary: We presented some preliminary results related to the building of fine resolution digital elevation maps with low altitude aerial images, using only a set of stereovision images taken from a tethered blimp as input data. Thanks to the development of an interest point matching algorithm that allows to precisely estimate the blimp motions, consistent $5 \times 5cm^2$ resolution maps covering areas of several thousands of square meters could be built. This appears to be a very cost-effective way of mapping terrains, that can be useful for any low altitude flying robot.
Open issues: These results are encouraging, but several important problems needs to be tackled in order to have a robust and reliable mapping module. They concern especially localization, and map building.

Our localization algorithm is a *motion estimation* algorithm, in the sense that it cumulates errors over time [21]: it processes successive images, without memorizing any particular feature. It could however be run in a different way when flying over an already modeled area: the interest points of the latest image could be matched with interest points memorized during the flight, or with interest points detected on the luminance values of the DEM. The algorithm would then turn into a *pose refinement* algorithm, that can reduce the error on the position. In such a case, one should re-estimate the aircraft last position, and thus *all the former positions*. To run such a back-propagation algorithm, one needs an error covariance matrix for all the position estimates: for that purpose, we are currently trying to determine an error model of the motion estimation algorithm.

Fig. 12: Final digital elevation map produced with the 80 stereovision images corresponding to the third trajectory of figure 1. The overall position estimation error can be seen on top of the right image, where the first and the last images overlap (see the calibration frame). The absolute translation error is about $1m$ at the end of the trajectory.

Several improvements are required for the DEM building algorithm. Indeed, one of the drawback of DEMs is that they consider that the perceived surface is convex: there is no way to represent vertical features and overheads with a single elevation value for each cell (see how bad the trees look in the figures of section 5). We are currently considering a much more sophisticated algorithm, akin to 3D occupancy grids [12], that maintains a discrete probability density function for the cell elevations.

Toward cooperative air/ground robotics: Whatever the cooperation scenario between aerial and ground robots is, we believe that to foster the development of such ensembles, one of the most important issue to address is building and management of environment representations using data provided by all possible sources. Indeed, not only each kind of machine (terrestrial, aerial) can benefit from the information gathered by the other, but also in order to plan and execute cooperative or coordinated actions, both kinds of machines must be able to build, manipulate and to reason on common consistent environment representations. Among the issues raised by these applications, the localization of a terrestrial rover with respect to a model built from aerial data is an important one, and digital elevation maps appear to be a well suited structure for that purpose. Such an ability is required to consider cooperative environment modeling, but it makes also a lot of sense when long range navigation missions are specified to a ground rover: it would guaranty a bounded error on the rover position estimations.

References

1. A. Elfes, N. Bergerman, J.R.H. Carvalho, E. Carneiro de Paiva, J.J.G. Ramaos, and S.S. Bueno. Air-ground robotic ensembles for cooperative applications : Concepts and preliminary results. In *2nd International Conference on Field and service Robotics, Pittsburgh, Pa (USA)*, pages 75–80. Auromation Institute, Center for Technology Information, Aug, 1999.
2. S. Lacroix. Toward autonomous airships: research and developments at laas/cnrs. In *3rd International Airship Convention and Exhibition, Friedrichshafen (Germany)*, July 2000.
3. E. Hygounenc, P. Soueres, and S. Lacroix. Developments on autonomous airship control at LAAS/CNRS. In *14th AIAA Lighter-Than-Air Systems Convention and Exhibition, Akron, Ohio (USA)*, July 2001.
4. I.S. Kweon and T. Kanade. Extracting topographic features for outdoor mobile robots. In *IEEE International Conference on Robotics and Automation, Sacramento, Ca (USA)*, pages 1992–1997, April 1991.
5. P. Fillatreau and M. Devy. Localization of an autonomous mobile robot from 3d depth images using heterogeneous features. In *International Workshop on Intelligent Robots and Systems, Yokohama (Japan)*, 1993.
6. H. Schultz, A. Hanson, E. Riseman, F. Stolle, , and Z. Zhu. A system for real-time generation of geo-referenced terrain models. In *SPIE Enabling Technologies for Law Enforcement, Boston, MA (USA)*. Umass, 2000.
7. M. Hebert, C.Caillas, E. Krotkov, I.S. Kweon, and T.Kanade. Terrain Mapping for a Roving Planetary Explorer. In *IEEE International Conference on Robotics and Automation, Scottsdale, Az. (USA)*, pages 997–1002, 1989.
8. I.S. Kweon and T. Kanade. High-resolution terrain map from multiple sensor data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):278–292, Feb. 1992.

9. C. Olson. Mobile robot self-localization by iconic matching of range maps. In *8th International Conference on Advanced Robotics, Monterey, Ca (USA)*, pages 447–452, July 1997.
10. R. Miller and O. Amidi. 3d site mapping with the cmu autonomous helicopter. In *5th International Conference on Intelligent Autonomous Systems*, 1998.
11. O. Faugeras, T. Vieville, E. Theron, J. Vuillemin, B. Hotz, Z. Zhang, L. Moll, P. Bertin, H. Mathieu, P. Fua, G. Berry, and C. Proy. Real-time correlation-based stereo : algorithm, implementations and application. Technical Report RR 2013, INRIA, August 1993.
12. A.P. Tirumalai, B.G. Schunck, and R.C. Jain. Evidential reasoning for building environment maps. *IEEE Transactions on Systems, Man and Cybernetics*, 25(1):10–20, Jan. 1995.
13. A. Mallet, S. Lacroix, and L. Gallo. Position estimation in outdoor environments using pixel tracking and stereovision. In *IEEE International Conference on Robotics and Automation, San Francisco, Ca (USA)*, pages 3519–3524, April 2000.
14. S. Lacroix, A. Mallet, and R. Chatila. Rover self localization in planetary-like environments. In *5th International Symposium on Artificial Intelligence, Robotics and Automation in Space, Noordwijk (The Netherlands)*, June 1999.
15. C. Olson, L. Matthies, M. Schoppers, and M. Maimone. Robust stereo ego-motion for long distance navigation. In *IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head Island, SC (USA)*. JPL, June 2000.
16. S. Gautama, S. Lacroix, and M. Devy. On the performance of stereo matching algorithms. In *Workshop on Vision, Modelling and Visualization, Erlangen (Germany)*, Nov. 1999.
17. I-K. Jung and S. Lacroix. A robust interest point matching algorithm. In *8th International Conference on Computer Vision, Vancouver (Canada)*, July 2001.
18. C. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, 1988.
19. C. Schmid, R. Mohr, and C. Bauckhage. Comparing and evaluating interest points. In *International Conference on Computer Vision*, Jan 1998.
20. R. Haralick, H. Joo, C.-N. Lee, X. Zhuang, V.G. Vaidya, and M.B. Kim. Pose estimation from corresponding point data. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1426–1446, Nov/Dec 1989.
21. S. Lacroix and A. Mallet. Integration of concurrent localization algorithms for a planetary rover. In *6th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, June 2001.

This article was processed using
the TEX macro package with SIRS2001 style