High resolution terrain mapping using low altitude aerial stereo imagery

Il-Kyun Jung and Simon Lacroix
LAAS/CNRS
7, Ave du Colonel Roche
31077 Toulouse Cedex 4 FRANCE
Il-Kyun.Jung, Simon.Lacroix@laas.fr

Abstract

This paper presents an approach to build high resolution digital elevation maps from a sequence of unregistered low altitude stereovision image pairs. The approach first uses a visual motion estimation algorithm that determines the 3D motions of the cameras between consecutive acquisitions, on the basis of visually detected and matched environment features. An extended Kalman filter then estimates both the 6 position parameters and the 3D positions of the memorized features as images are acquired. Details are given on the filter implementation and on the estimation of the uncertainties on the feature observations and motion estimations. Experimental results show that the precision of the method enables to build spatially consistent very large maps.

1. Introduction

The main difficulty to build digital terrain maps is to *precisely* determine the sensor position and orientation as it moves. Dead reckoning techniques that integrate over time the data provided by motion estimation sensors, such as inertial sensors, are not sufficient because they are intrinsically prone to generate position estimates with unbounded error growth. Precise visual motion estimation techniques that use stereovision and visual features tracking or matching have been proposed for ground rovers [1], but their errors also cumulate over time, since they do not memorize any environment feature. The only solution to diminish the errors on the position estimates is to rely on stable environment features, that are detected and *memorized* as the sensor moves. In the robotic community, it has early been understood that the problem of mapping such features and estimating the robot location are intimately tied together, and that they must therefore be solved in a unified manner [2]. This problem, known as the "SLAM problem[1]" has now been widely studied (an historical presentation of the main contributions can be found in the introduction of [3]).

Contributions on "structure from motion" addresses the same problem. Successful approaches have been reported, but they require batch processing (*i.e.* with the whole sequence of the acquired images) and a global optimization (*e.g.* bundle adjustment) to refine the camera positions and the 3D coordinates of matched features. In contrast, SLAM is an incremental approach: the 3D feature map and the sensor position are simultaneously built and updated.

Among the different approaches to tackle the SLAM problem, the Kalman filter based approach is the most popular. It is theoretically well grounded, and it has been proved that its application to the SLAM problem converges [3]. Some contributions cope with its main practical drawback, *i.e.* its complexity which is cubic in the dimension of the considered state [4]: such developments are necessary when mapping very large areas. Other approaches to the SLAM problem have been proposed, mainly to overcome the assumption that the various error probability distributions are Gaussian, which is required by the Kalman filter. Set membership approaches just need the knowledge of bounds on the errors [5], but they are practically difficult to implement when the number of position parameters exceeds 3, and are somehow sub-optimal. Expectation minimization algorithms (EM) have also been successfully adapted to the SLAM problem [6]. In terms of sensor modality, solutions to the SLAM problem has mainly been experimented with range sensors in indoor environments, in the case of robots moving on planes, *i.e.* in 2 dimensions [7, 6, 3]. To our knowledge, there are very few contributions to the SLAM problem based on vision (*e.g.* [8]).

This paper presents an approach to the SLAM problem in 3 dimensions, using *only* a set of non-registered low altitude stereovision image pairs. The approach is presented in the following section, and section 3 summarizes the basic algorithms on which it relies: stereovision, interest points detection and matching, and visual motion estimation. Section 4 details our implementation of the Extended Kalman Filter, with a focus on the identification of the various errors. Localization results and the building of digital elevation maps with a stereovision bench mounted on a blimp flying at a few tens of meter altitude are then presented.

2. Overview of the approach

Landmarks are *interest points*, *i.e.* visual features that can be matched when perceived from various positions, and whose 3D coordinates are provided by stereovision. We use

[1] SLAM stands for "Simultaneous Localization And Mapping"

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set
0-7695-1950-4/03 $17.00 © 2003 IEEE

IEEE COMPUTER SOCIETY

an extended Kalman filter (EKF) as the recursive filter: the state vector of the EKF is the concatenation of the stereo bench position (6 parameters) and the landmark's positions (3 parameters for each landmark). The key algorithm that allows both motion estimation between consecutive stereovision frames (prediction) and the observation and matching of landmarks (data association) is a robust interest point matching algorithm.

The various algorithmic stages achieved every time a stereovision image pair is acquired are the following:

1. Stereovision: a dense 3D image is provided by stereovision (section 3.1).

2. Interest points detection and matching between consecutive frames, and with past frames in which old landmarks are visible (section 3.2).

3. Landmark selection: a set of selection criteria are applied to the matched interest points, in order to partition them in three sets: an a non-landmark set, a candidate-landmarks set and observed-landmark set (section 4.2).

4. Visual motion estimation (VME): the interest points retained as "non-landmarks" are used to estimate the 6 motion parameters between the previous and current frames (section 3.3).

5. Update of the Kalman filter state (section 4).

Finally, after every SLAM cycle defined by these steps, a digital elevation map is updated with the acquired images (section 5.2).

Step 3 is necessary for two reasons: first, only non-landmarks points should be used to to estimate the local motion, in order to de-correlate the prediction and update steps of the Kalman filter and second, new landmarks should be cautiously added to the filter state, in order to avoid a rapid growth of its dimension and to obtain a regular landmark coverage of the perceived scenes.

3. Basic algorithms

3.1. Stereovision

We use a classical pixel-based stereovision algorithm, that relies on a calibrated binocular stereovision bench (figure 1). A dense disparity image is produced thanks to a correlation-based pixel matching algorithm, false matches being filtered out thanks to a reverse correlation. The 3D coordinates of the matched pixels are determined, with an associated uncertainty whose computation is depicted in section 4.1.

3.2. Interest points detection and matching

Visual landmarks must be invariant to image translation, rotation, scaling, partial illumination changes and viewpoint changes. *Interest points*, such as detected by the popular Harris detector, has proven to have good stability properties [9]. When a there is prior knowledge on the scale change, even approximate, a scale adaptive version of Harris detector yields a repeatability high enough to allow robust matches [10].

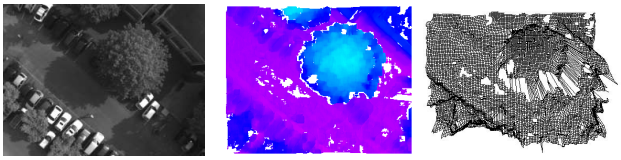

Figure 1: *A result of the stereovision algorithm, with an image pair taken at about* $30\ m$ *altitude. From left to right: one of the original image, disparity map (shown here in a blue/close red/far color scale), and 3D image, rendered as a mesh for readability purposes. Pixels are properly matched in all the perceived areas, even the low textured ones.*

To match interest points, we use a matching algorithm that relies on local interest point groups matching, imposing a combination of geometric and signal similarity constraints, thus being more robust than approaches solely based on local point signal characteristics (details can be found in [11]). Figure 2 shows that this algorithm generates a lot of good matches, even when the view point change between the considered images is quite high.

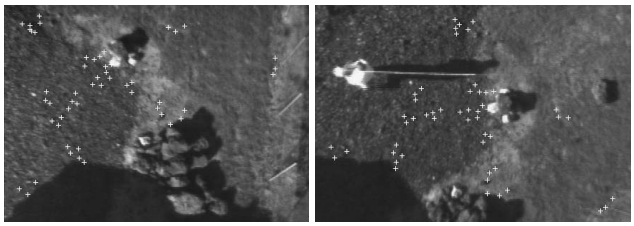

Figure 2: *A result of our interest point matching between two non registered aerial images*

3.3. VME (Visual Motion Estimation)

The interest points matched between consecutive images and the corresponding 3D coordinates provided by stereovision are used to estimate the 6 displacement parameters between the images, using the least square minimization technique presented in [12].

Conventional techniques to get rid of the outliers using fundamental matrix estimation, would not cope for stereovision errors, such as the ones that occur along depth discontinuities. Therefore, matches that imply a 3D point whose coordinates uncertainties are over a threshold are first discarded (the threshold is empirically determined by statistical analysis of stereovision errors). Then, a 3D transformation with the remaining matches is estimated and the 3D matches of which error is over k times the mean of the residual errors are eliminated: k should initially be at least greater than 3. k is set to $k - 1$ and the procedure is reiterated until $k = 3$.

This outlier rejection algorithm considering both matching and stereovision errors yields a precise 3D motion estimation between consecutive stereovision frames (see results

IEEE COMPUTER SOCIETY

in sections 4.1 and 5.1), which is used for the prediction stage of the Kalman filter.

4. Kalman filter setup

The EKF is an extension of the standard linear Kalman filter, that linearizes the nonlinear prediction and observation models around the predicted state. The discrete nonlinear system and the observations are modeled as:

$$x(k+1) = \mathbf{f}(x(k), u(k+1)) + v(k+1)$$

$$z(k+1) = \mathbf{h}(x(k+1)) + w(k+1)$$

where $u(k)$ is a control input and v, w are vectors of temporally uncorrelated errors with zero mean and covariance $\mathbf{P}_v(k)$, $\mathbf{P}_w(k)$.

In our approach, the state of the filter $\mathbf{x}(k) = [\mathbf{x}_p, \mathbf{m}_1 \cdots \mathbf{m}_N]$ is composed of the 6 position parameters $\mathbf{x}_p = [\phi, \theta, \psi, t_x, t_y, t_z]$ of the stereovision bench and of a set of N landmarks 3D coordinates $\mathbf{m}_i = [x_i, y_i, z_i]$, $0 < i \leq N$. The associated state covariance $\mathbf{P}$ is composed of the the stereo bench pose covariance $\mathbf{P}_{pp}$, the landmarks covariance $\mathbf{P}_{mm}$ and the cross-covariance between the bench pose and landmarks $\mathbf{P}_{pm}$ [3]. In the Kalman filter framework, the state estimation encompasses three stages: prediction, observation and update of the state and covariance estimates.

Prediction. Under the assumption that landmarks are stationary, the state prediction is:

$$\hat{\mathbf{x}}(k+1 \mid k) = \mathbf{f}(k)(\hat{\mathbf{x}}(k), \mathbf{u}(k+1))$$

where $\mathbf{u}(k+1) = (\Delta\phi, \Delta\theta, \Delta\psi, \Delta t_x, \Delta t_y, \Delta t_z)$ is the visual motion estimation result between k and $k+1$ positions. The associated state covariance prediction is written as:

$$\begin{aligned}\mathbf{P}_{pp}(k+1 \mid k) = \nabla_p \mathbf{f}(k) \mathbf{P}_{pp}(k) \nabla_p \mathbf{f}^T(k) + \\ \nabla_u \mathbf{f}(k) \mathbf{R}_u(k) \nabla_u \mathbf{f}^T(k) + \mathbf{P}_v(k+1)\end{aligned} \quad (1)$$

$$\mathbf{P}_{pm}(k+1 \mid k) = \nabla_p \mathbf{f}(k) \mathbf{P}_{pm}(k)$$

where $\mathbf{R}_u$ represents *the error covariance of the visual motion estimation result*. Note that the covariance of landmarks is not changed in the prediction stage.

Observation. When observing the i^{th} landmark, the observation model and the Jacobian of the observation function are written as:

$$\hat{\mathbf{z}}_i(k+1 \mid k) = \mathbf{h}_i(k)(\hat{\mathbf{x}}(k+1 \mid k))$$

where $\mathbf{h}_i(k)(\hat{\mathbf{x}}(k+1 \mid k))$ is a function of the predicted robot state and the i^{th} landmark in the state vector of the filter, which maps the state space into the observation state. The innovation and the associated covariance is written as:

$$\boldsymbol{\nu}_i(k+1) = \mathbf{z}_i(k+1) - \hat{\mathbf{z}}_i(k+1 \mid k) \quad (2)$$

$$\mathbf{S}_i(k+1) = \nabla \mathbf{h}_i(k) \mathbf{P}(k+1 \mid k) \nabla \mathbf{h}_i{}^T(k) + \mathbf{R}_i(k+1) \quad (3)$$

where $\nabla \mathbf{h}_i(k) = [\nabla_p \mathbf{h}_i(k), 0 \cdots, 0 \nabla_{m_i} \mathbf{h}_i(k), 0 \cdots 0]$ and $\mathbf{R}_i$, *the error covariance of ith landmark observation*.

Update. The update stage fuses the prediction and the observation to produce the state estimate and its associated covariance:

$$\hat{\mathbf{x}}(k+1 \mid k+1) = \hat{\mathbf{x}}(k+1 \mid k) + \mathbf{K}_i(k+1) \boldsymbol{\nu}_i(k+1)$$

$$\mathbf{P}(k+1 \mid k+1) = \mathbf{P}(k+1 \mid k) - \mathbf{K}_i(k+1) \mathbf{S}_i(k+1) \mathbf{K}_i^T(k+1)$$

in which $\mathbf{K}_i(k+1) = \mathbf{P}(k+1 \mid k) \nabla \mathbf{h}_i^T(k) \mathbf{S}_i^{-1}(k+1)$ is the kalman filter gain matrix.

When detecting a new landmark, it is added to the state vector of the filter, that becomes $\hat{\mathbf{x}}(k) = [\hat{\mathbf{x}}_p(k), \hat{\mathbf{m}}_1(k) \cdots \hat{\mathbf{m}}_N(k), \hat{\mathbf{m}}_{N+1}(k)]$. The landmark initialization model is:

$$\hat{\mathbf{m}}_{N+1}(k) = \mathbf{g}(k)(\hat{\mathbf{x}}_p(k), \mathbf{z_{N+1}}(k)) \quad (4)$$

$$\mathbf{P}(k) = \begin{bmatrix} \mathbf{P}_{pp}(k) & \mathbf{P}_{pm}(k) & \mathbf{P}_{pz}(k)^T \\ \mathbf{P}_{pm}{}^T(k) & \mathbf{P}_{mm}(k) & \mathbf{P}_{mz}(k)^T \\ \mathbf{P}_{pz}(k) & \mathbf{P}_{mz}(k) & \mathbf{P}_{zz}(k) \end{bmatrix} \quad (5)$$

$$\mathbf{P}_{pz}(k) = \nabla_p \mathbf{g}(k) \mathbf{P}_{pp}(k), \ \mathbf{P}_{mz}(k) = \nabla_p \mathbf{g}(k) \mathbf{P}_{pm}(k)$$

$$\mathbf{P}_{zz}(k) = \nabla_p \mathbf{g}(k) \mathbf{P}_{pp}(k) \nabla_p \mathbf{g}^T(k) + \nabla_z \mathbf{g}(k) \mathbf{R}_m(k) \nabla_z \mathbf{g}^T(k)$$

where $\mathbf{z_{N+1}}(k)$ denotes the new landmark, $\mathbf{g}(k)$ the initialization function using the current robot pose estimate and $\mathbf{R}_m$ *the error covariance of the new landmark.*

4.1. Errors identification

Error identification is crucial to set up a Kalman filter, as a precise determination of these errors will avoid the empirical "filter tuning" step. In our context, the following errors must be estimated:

- the landmark initialization error ($\mathbf{R}_m$),
- the landmark observation error ($\mathbf{R}_i$ for the observed landmark i),
- and the error of the input control u, which is the visual motion estimation result ($\mathbf{R}_u$).

Note that in our approach, the lumped process noise v is set to 0, landmarks being stationary and the robot pose prediction being directly computed with the current pose and the result of the visual motion estimation.

Landmark initialization errors. When new landmarks are detected, their 3D coordinates being computed by stereovision, the covariance matrix $\mathbf{R}_m$ on new landmarks is totally defined by the stereovision error.

Statistics on image pairs acquired from the same position show that the distribution of the disparity computed on any given pixel can be well approximated by a Gaussian [13], and that there is a *correlation* between the shape of the similarity score curve around its peak and the standard deviation on the disparity: the sharper the peak, the more precise the disparity (figure 3). A standard deviation σ_d associated to

IEEE COMPUTER SOCIETY

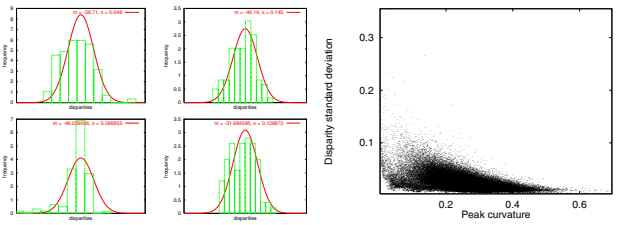

Figure 3: *Left: examples of some probability density functions of disparities computed on a set of 100 image pairs, with the corresponding Gaussian fit. Right: Standard deviation of the disparities as a function of the curvature of the similarity score curve at its peak.*

each computed disparity d is estimated using the curvature of the curve at its peak, approximated by fitting a parabola.

Once matches are established, the coordinates of the 3D points are computed with the usual triangulation formula: $z = \frac{b\alpha}{d}$, $x = \beta_u z$ and $y = \gamma_v z$, where z is the depth, b is the stereo baseline, and α, β_u and γ_v are calibration parameters (that depends on (u, v), the considered pixel image coordinates). Using a first order approximation, it comes:

$$\sigma_z = \frac{\sigma_d}{b\alpha} z^2$$

The covariance matrix of the point coordinates is then derived from the triangulation equations. When a new landmark is observed, its coordinates are added to the filter state, and the state covariance is updated according to equations (4) and (5).

Observation error. In our case, landmark observation is based on interest point matching. Outliers being rejected (section 3.3), only interest point location errors are considered to determine the matching error on image plane. With the precise Harris detector, the location estimate of interest point in two consecutive images taken from a very close point of view is precise enough to use 1.5 pixel as the maximum error tolerated to asses good matches [9] due to negligible projective distortion and occlusion. The expected matching error is therefore of the order of 0.5 pixel. In contrast, under different view point(i.e. when re-perceiving a landmark after a long loop), a more flexible tolerance limit is applied(i.e. around 2.5 pixel) [11]. In such case, the expected matching error value is then set to 1 pixel.

The observation error is defined by the reprojection of the matching error on the 3D plane estimated by stereovision (with an associated error - figure 4).

When the 2D matching error is set to 1 pixel, the errors provided by stereovision for the 3D matching point and its 8 closest neighbors are used to compute the expected 3D coordinate and associated variance of the matching point as follows:

$$\bar{\mathbf{X}} = \frac{1}{9}\sum_{i=0}^{8} \mathbf{X}_i, \qquad \boldsymbol{\sigma}^2_{\bar{\mathbf{X}}} = \frac{1}{9}\sum_{i=0}^{8} (\bar{\mathbf{X}} - \mathbf{X}_i)^2 + \boldsymbol{\sigma}_i^2$$

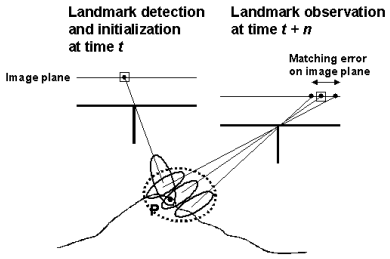

Figure 4: *Principle of the combination of the matching and stereovision errors. The points located in the square box are the projection of P on the image plane. Small ellipses indicate stereovision errors, the large dotted ellipsoid is the resulting observation error.*

where $\mathbf{X}_0$ and $\mathbf{X}_k$ are the 3D point coordinates of $\mathbf{p}_0$ and its neighbors, and $\boldsymbol{\sigma}_0$ and $\boldsymbol{\sigma}_k$ are the corresponding variances.

When the matching error is 0.5 pixel, the 3D coordinates being only computed on integer pixels by stereovision, we assume the 3D surface variation is locally linear: $\mathbf{X}_{i/2} = (\mathbf{X}_0 + \mathbf{X}_i)/2$ and $\boldsymbol{\sigma}_{i/2} = (\boldsymbol{\sigma}_0 + \boldsymbol{\sigma}_i)/2$. These coordinates and the associated variances are used in the equations (2) and (3).

Motion estimation errors. Given a set of 3D matched points $\hat{\boldsymbol{\mathcal{Q}}} = [\mathbf{X}_1, , , \mathbf{X}_N, \mathbf{X}_1, , , \mathbf{X'}_N]$, the function which is minimized to determine the corresponding motion is [12]:

$$J(\hat{\mathbf{u}}, \hat{\boldsymbol{\mathcal{Q}}}) = \sum_{n=1}^{N} (\mathbf{X'}_n - R(\hat{\phi}, \hat{\theta}, \hat{\psi})\mathbf{X}_n - [\hat{t}_x, \hat{t}_y, \hat{t}_z]^T)^2$$

where $\hat{\mathbf{u}} = (\hat{\phi}, \hat{\theta}, \hat{\psi}, \hat{t}_x, \hat{t}_y, \hat{t}_z)$. $\hat{\mathbf{u}}$. $\hat{\boldsymbol{\mathcal{Q}}}$ can be written with random perturbations ($\hat{\mathbf{u}} = \mathbf{u} + \Delta\mathbf{u}$, $\hat{\boldsymbol{\mathcal{Q}}} = \boldsymbol{\mathcal{Q}} + \Delta\boldsymbol{\mathcal{Q}}$) and the true $\mathbf{u}$ and $\boldsymbol{\mathcal{Q}}$ are not observed. In order to measure the uncertainty of $\hat{\mathbf{u}}$, the uncertainties of landmarks $\mathbf{X}_n$ and their observation $\mathbf{X}'_n$ are propagated as shown in [14]: considered that $\mathbf{X}_n$ and $\mathbf{X'}_n$ are not correlated, the covariance estimate $\mathbf{P}_{\hat{\mathbf{u}}}$ can be also written as:

$$\mathbf{P}_{\hat{\mathbf{u}}} = (\frac{\partial g}{\partial \mathbf{u}}(\hat{\mathbf{u}}, \hat{\boldsymbol{\mathcal{Q}}}))^{-1} (\Lambda_{\mathbf{X}} + \Lambda_{\mathbf{X'}}) (\frac{\partial g}{\partial \mathbf{u}}(\hat{\mathbf{u}}, \hat{\boldsymbol{\mathcal{Q}}}))^{-1}$$

where $g = \frac{\partial J}{\partial \hat{\mathbf{u}}}$ is the Jacobian of the cost function and

$$\Lambda_{\mathbf{X}} = \sum_{n=1}^{N} \frac{\partial g}{\partial \mathbf{X}_n}(\hat{\mathbf{u}}, \mathbf{X}_n) \mathbf{P}_{\mathbf{X}_n} (\frac{\partial g}{\partial \mathbf{X}_n}(\hat{\mathbf{u}}, \mathbf{X}_n))^T$$

$$\Lambda_{\mathbf{X'}} = \sum_{n=1}^{N} \frac{\partial g}{\partial \mathbf{X'}_n}(\hat{\mathbf{u}}, \mathbf{X'}_n) \mathbf{P}_{\mathbf{X'}_n} (\frac{\partial g}{\partial \mathbf{X'}_n}(\hat{\mathbf{u}}, \mathbf{X'}_n))^T$$

$\mathbf{P}_{\hat{\mathbf{u}}} = \mathbf{R}_u$ is the input covariance matrix which is used in equation (1) to estimate the state variances during the filter prediction stage.

IEEE COMPUTER SOCIETY

4.2. Landmark selection

As explained in the section 2, the 3D matches established after the interest point matching step are split into three sets. The observed-landmarks set is simply the points that corresponds to landmarks already in the state vector of the EKF. The rest of the matches are then studied, to select the set of candidate-landmarks according to the following three criteria:

• **Observability.** Good landmarks should be observable in several consecutive frames.

• **Stability.** The 3D coordinates of good landmarks must be precisely estimated by stereovision.

• **Representability.** Good landmarks must efficiently represent a 3D scene. The stereovision bench state estimation will be more stable if landmarks are regularly dispatched in the perceived scene, and this regularity will avoid a rapid growth of the EKF state vector size.

The number of candidate landmarks that are checked is determined on the basis of the number of new interest point matches (*i.e.* the ones that do not match with an already mapped landmark). We use $10\ \%$ of the new interest points, as the visual motion estimation technique requires a lot of matches to yield a precise result. The landmark selection is made according a heuristic procedure so that they satisfy the above three criteria.

5. Results

Our developments have been tested with hundreds of images taken on-board a blimp, at altitudes ranging from 20 to $35\ m$. The cameras of the $2.2\ m$ wide stereo bench are $1/2''\ 1024\times 768$ pixels CCD sensors, with a a $4.8\,mm$ focal length lens.

5.1. Positioning errors

We do not have any localization mean that could be used as reference on-board the blimp (such as a centimeter accuracy GPS). However, when the blimp flies over an already perceived area, the VME can provide an precise estimate of the relative pose between the first and last frame of the sequence that overlaps.

Figure 5 presents a comparison of the reconstructed loop trajectory, while figure 6 shows the evolution of the standard deviation of the 6 position parameters of the stereo bench when applying the EKF. Until image 25, the standard deviation grows, however much more slowly than when propagating only the errors of the VME. A few *old* landmarks are re-perceived in the following images: the standard deviations decreases, and stabilizes for the subsequent images where some old landmarks are still observed.

The quantitative figures summarized in table 1 compare the results of the final position estimate with respect to the reference: the precision enhancement brought by the EKF is noticeable, and the absolute estimated errors are all bounded by twice the estimated standard deviations. The

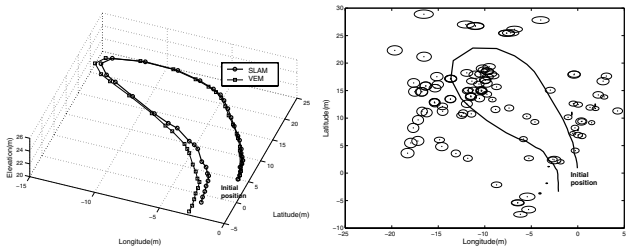

Figure 5: *A result of the SLAM implementation with a sequence of 40 stereovision pair. The left image show the reconstructed trajectory in 3D, the right one shows the 120 landmarks mapped, with 1σ uncertainty ellipses magnified by a factor of 40.*

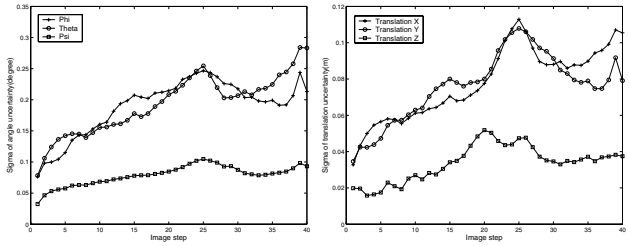

Figure 6: *Evolution of the standard deviations of the camera position parameters during the flight shown in figure 5.*

translation errors are below $0.1\ m$ in the three axes after an about $60\ m$ long trajectory, and angular errors are all below half a degree.

Figure 7 shows and other trajectory reconstructed with a set of 100 images.

5.2. Digital elevation maps

Thanks to the precise positioning estimation, the processed stereovision images can be fused after every update of the EKF into a *digital elevation map* (DEM), that describes the environment as a function $z = f(x, y)$, determined on every cell (x_i, y_i) of a regular Cartesian grid.

Our algorithm to build a DEM simply computes the elevation of each cell by averaging the elevations of the 3D points that are vertically projected on the cell surface. Since

	Reference std. dev.	VME abs. error	SLAM std. dev.	SLAM abs. error
Φ	$0.10°$	$3.30°$	$0.21°$	$0.20°$
Θ	$0.09°$	$2.40°$	$0.28°$	$0.61°$
Ψ	$0.04°$	$0.56°$	$0.09°$	$0.11°$
t_x	$0.04\ m$	$0.91\ m$	$0.11\ m$	$0.24\ m$
t_y	$0.04\ m$	$1.47\ m$	$0.08\ m$	$0.07\ m$
t_z	$0.01\ m$	$0.91\ m$	$0.04\ m$	$0.10\ m$

Table 1: *Comparison of the errors made by the propagation of the visual motion estimation alone and with the SLAM EKF approach, using as a reference the VME applied between images 1 and 40*

IEEE COMPUTER SOCIETY

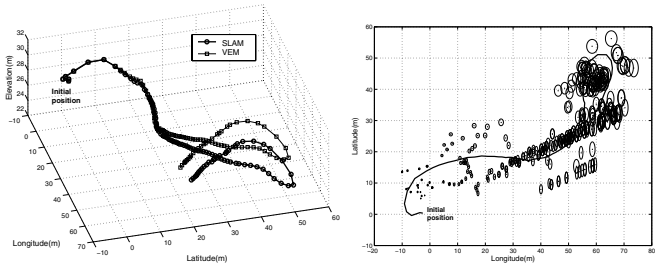

Figure 7: *A longer trajectory reconstructed with a sequence of 100 images. 320 landmarks have been mapped (the magnification factor of the uncertainty ellipses in the right image is here 20)*

a luminance value is associated to each 3D point produced by stereovision, it is also possible to compute a mean luminance value for each map cell. Figure 8 shows a digital elevation built from the 100 images during the trajectory of figure 7: the resolution of the grid is here $0.1\ m$, and no map discrepancies can be detected in the corresponding orthoimage.

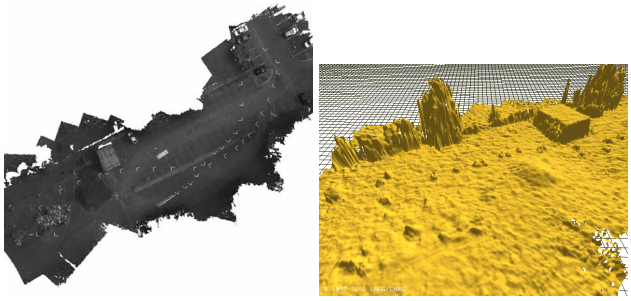

Figure 8: *The DEM computed with 100 images, positioned according to the trajectory of figure 7: orthoimage and 3D view of the bottom-left area. The map covers an area of about* $3500\ m^2$.

6. Summary

We presented a vision-based SLAM approach that allows the building of large high resolution terrain maps. To our knowledge, it is the first attempt to tackle a SLAM problem in 3D space, using exclusively informations provided by vision. The use of interest point as landmarks allows an active selection of the landmarks to properly map the environment without any prior knowledge. Our interest point matching algorithm provides robust data associations which makes possible the matching of already mapped landmark and the precise visual motion estimation between consecutive frames. A rigorous study and identification of the various errors estimates involved in the filter allows to set it up properly, without any empirical tuning stage.

References

[1] C. Olson, L. Matthies, M. Schoppers, and M. Maimone. Stereo ego-motion improvements for robust rover navigation. In *IEEE International Conference on Robotics and Automation*, pages 1099–1104, May 2001.

[2] R. Smith, M. Self, and P. Cheeseman. A stochastic map for uncertain spatial relationships. In *Robotics Research: The Fourth International Symposium, Santa Cruz (USA)*, pages 468–474, 1987.

[3] G. Dissanayake, P. M. Newman, H-F. Durrant-Whyte, S. Clark, and M. Csorba. A solution to the simultaneous localization and map building (slam) problem. *IEEE Transaction on Robotic and Automation*, 17(3):229–241, May 2001.

[4] J. Guivant and E. Nebot. Optimization of the simultaneous localization and map building algorithm for real time implementation. *IEEE Transactions on Robotics and Automation*, 17(3):242–257, June 2001.

[5] M. Kieffer, L. Jaulin, E. Walter, and D. Meizel. Robust autonomous robot localization using interval analysis. *Reliable Computing*, 6(3):337–362, Aug. 2000.

[6] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot with applications to multi-robot and 3d mapping. In *IEEE International Conference on Robotics and Automation, San Francisco, CA (USA)*, 2000.

[7] O. Wijk and H.I. Christensen. Triangulation based fusion of sonar data for robust robot pose tracking. *IEEE Transactions on Robotics and Automation*, 16(6):740–752, 2000.

[8] S. Se, D. Lowe, and J. Little. Local and global localization for mobile robots using visual landmarks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, Hawaii (USA)*, pages 414–420, October 2001.

[9] C. Schmid, R. Mohr, and C. Bauckhage. Comparing and evaluating interest points. In *Proceeding of the 6th International Conference on Computer Vision, Bombay (India)*, pages 230–235, January 1998.

[10] Y. Dufournaud, C. Schmid, and R. Horaud. Matching images with different resolutions. In *International Conference on Computer Vision and Pattern Recognition, Hilton Head Island, SC (USA)*, pages 612–618, Juin 2000.

[11] I-K. Jung and S. Lacroix. A robust interest point matching algorithm. In *8th International Conference on Computer Vision, Vancouver (Canada)*, July 2001.

[12] 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.

[13] L. Matthies. Toward stochastic modeling of obstacle detectability in passive stereo range imagery. In *IEEE International Conference on Computer Vision and Pattern Recognition, Champaign, Illinois (USA)*, pages 765–768, 1992.

[14] R.M. Haralick. Propagating covariances in computer vision. In *International Conference on Pattern Recognition*, pages 493–498, 1994.

IEEE COMPUTER SOCIETY