Monocular Vision Based SLAM for Mobile Robots

E. Mouragnon[1,2], M. Lhuillier[1], M. Dhome[1], F. Dekeyser[2], P. Sayd[2]
[1]LASMEA UMR 6602, Université Blaise Pascal/CNRS, 63177 Aubière Cedex, France
[2]Image and embedded computer lab., CEA/LIST/DTSI/SARC, 91191 Gif s/Yvette Cedex, France

Abstract

This paper describes a new vision based method for the Simultaneous Localization and Mapping of mobile robots. The only data used is a video input from a moving calibrated monocular camera. From the detection and matching of interest points in images at video rate, robust estimates of the camera poses are computed in real-time and a 3D map of the environment is reconstructed. The computed 3D structure is constantly refined thanks to the introduction of a fast and local bundle adjustment method that makes this approach particularly accurate and reliable. Actually, this method can be seen as a new visual tool that may be used in conjunction with usual systems (GPS, inertia sensors, etc) in SLAM applications.

1. Introduction

Simultaneous Localization and Mapping (SLAM) is an essential capability for mobile robots exploring unknown environments, using very different sensors or sources of information (Figure 1). Recently, many works were carried out on SLAM, and this paper focus on a vision based method that only uses data from a moving monocular camera. The robust and automatic estimate of the movement of a perspective camera (calibrated or not) and observed points, from a sequence of images have been largely studied [13, 17, 8, 1, 4, 12, 9, 15]. In Vision community, the problem is called SFM, for Structure From Motion and is the subject of many works. Initially, interest points are detected and matched between successive frames. Then, robust estimates of relative movement are made with random samples, and a model of the environment is reconstructed in three dimensions.

One can note two main types of approaches for SFM algorithms. First, there are off-line methods [13, 17, 4, 12, 9, 15] carrying out a bundle adjustment optimization of the global geometry. Bundle adjustment [19] is a process which adjusts iteratively the pose of cameras as well as points position in order to obtain the minimal reprojection error (due to the difference between points detected in the images and the reprojections of 3D points through the cameras). Most articles refer to Levenberg-Marquardt (LM) to solve the non linear criterion involved in bundle adjustment, a method which combines the Gauss-Newton algorithm and the descent of gradient. In that case, a very accurate model is generated but it is very expensive in terms of computing time because of the resolution of linear systems (whose size is proportional to the number of estimated parameters) and can not be implemented in a real time application. On the other hand, there are methods without global optimization. They are really fast but their accuracy is questionable since errors accumulate in time. Among those works, Nistér [11] presents a method called "visual odometry". This method estimates the movement of a stereo head or a simple camera in real time from the only visual data: the aim is to guide robots. Davison [2] proposes a real time camera pose calculation but he assumes that number of landmarks is small (under about 100 landmarks). This approach best suits to indoor environments and is not appropriate for long displacements because of algorithmic complexity and growing uncertainty.

In this paper, we propose a complete method from the acquisition of images with the camera, to an estimate of the current position and uncertainty, and a 3D map of the environment. The method takes benefit from bundle adjustment methods accuracy against Kalman filters [2, 16], and from speed of incremental methods [11, 20, 18]. This has been possible with the introduction of a fast and local bundle adjustment process which is carried out each time a new camera is added to the system.The paper is organized as follows. First, we explain in details our complete method to compute camera motion and 3D structure from a video flow. We explain our incremental method with local bundle adjustment: we propose to only optimize the end of the 3D structure with a set of parameters restricted to the last cameras and 3D points observed by these cameras. In a second part, we present experiments and results on real data, and we compare to GPS ground truth.

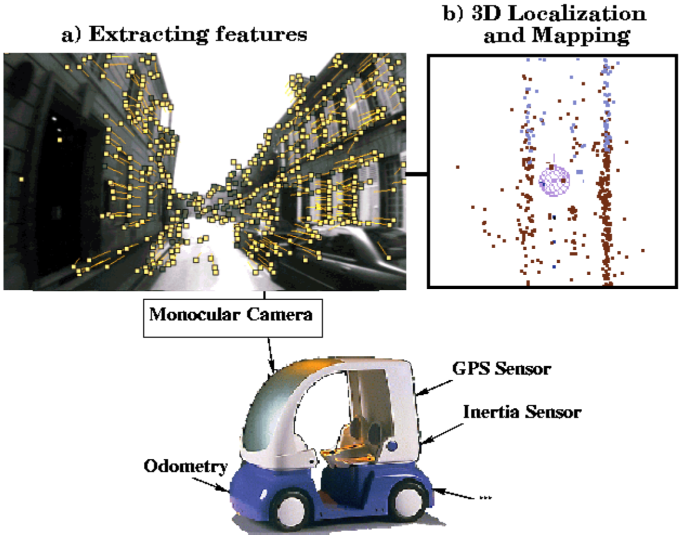

Figure 1. Monocular Vision among other sensors. a) An example of image and points tracks. b) Top view of the real time localization. We can see 3D reconstructed points and the ellipsoid of confidence for the current camera pose.

2. Description of the incremental algorithm

Let us consider a video sequence acquired with a camera settled on a vehicle moving in an unknown environment. The goal of this work is to find the position and the orientation in a global reference frame of the camera at several times t as well as the 3D position of a set of points (viewed along the scene). We use a monocular camera whose intrinsic parameters (including radial distortion) are known and assumed to be unchanged throughout the sequence.
The algorithm begins with determining a first triplet of images that will be used to set up the global frame and the system geometry. After that, a robust pose calculation is carried out for each frame of the video flow using features detection and matching. Some of the frames are selected and become key-frames that are used for 3D points triangulation. The system operates in an incremental way, and when a new key-frame and 3D points are added, we proceed to a local bundle adjustment. The output is the current position of the camera and its uncertainty and the final result (see Figure 3) is a complete trajectory and the 3D coordinates of points seen in images.

Interest points detection and matching The whole method is based on the detection and matching of features points (Figure 1 a.). In each frame, Harris corners [7] are detected and matched with points detected in last key frame by computing a Zero Normalized Cross Correlation score in a region of interest. The pairs with the high-scores are selected to provide a list of corresponding point pairs between the two images. The step "*detection and matching*" of the method has been efficiently implemented using SIMD extensions of modern processors.

Real-time robust pose estimation The sequence initialization and global coordinate system have been set up using the 5-points algorithm [10] and a RANSAC [5] approach on a sub-sample of three frames (among other possibilities).

Now, let us suppose that pose of cameras C_1 to C_{i-1} corresponding to selected key-frames I_1 to I_{i-1} have previously been calculated in the reference reconstruction frame. We have also found a set of points whose projections are in the corresponding images. The goal is to calculate camera pose C corresponding to the last acquired frame I. For that, we match I (last acquired frame) and I_{i-1} (last selected key frame) to determine a set of points p whose projections on the cameras $(C_{i-2}\ C_{i-1}\ C)$ are known and whose 3D coordinates have been computed before. From 3D points reconstructed from C_{i-2} and C_{i-1}, we use Grunert's pose estimation algorithm as described in [6] to compute the location of camera C. A RANSAC process gives an initial estimate of camera C pose which is then refined using a fast LM optimization stage with only 6 parameters (3 for optical center position and 3 for orientation). At this stage, the covariance matrix of camera C pose is calculated and we are able to draw an ellipsoid of confidence at 90% (see Figure 1 b.). If Cov is the covariance matrix of camera C pose, the ellipsoide of confidence is given by $\Delta x^T Cov^{-1} \Delta x \leq 6.25$ since $\Delta x^T Cov^{-1} \Delta x$ obeys a $\mathcal{X}_2$ distribution with 3 dof.

Key frames selection and 3D points reconstruction The motion between two frames must be sufficiently large to accurately compute the 3D positions of matched points. So, not all the frames of the input are taken into account for the 3D reconstruction, but only a sub-sample of the video. We select frames relatively far from each other but that have enough common points. For each frame, the normal way is to compute the corresponding localization using the last two key frames. We set up a criterion that indicates if a new frame must be added as a key frame or not. First, if the number of matched points with the last key frame I_{i-1} is not sufficient (typically inferior to a fixed level M, $M = 400$ in experiments), we have to introduce a new key-frame. We have also to take a new key frame if the the uncertainty of the calculated position is too high (for example, superior to the mean inter-distance between two consecutive key positions). Obviously, it is not the frame for which criterion is refused that becomes a key frame but the one which immediately precedes. After that, new points (ie. those which are only observed in I_{i-2}, I_{i-1} and I_i) are reconstructed using a standard triangulation method.

Local bundle adjustment When the last key frame I_i is selected and added to the system, a stage of optimization is carried out. It is a bundle adjustment or Levenberg-Marquardt minimization of the cost function $f^i(\mathcal{C}^i, \mathcal{P}^i)$

where $\mathcal{C}^i$ and $\mathcal{P}^i$ are respectively the cameras parameters (extrinsic parameters) and 3D points chosen for this stage i. The idea is to reduce the number of calculated parameters in optimizing only the extrinsic parameters of the n last cameras and taking account of the 2D reprojections in the N (with $N \geq n$) last frames (see Figure 2). Thus, $\mathcal{C}^i = \{C_{i-n+1}..C_i\}$ and $\mathcal{P}^i$ contains all the 3D points projected on cameras $\mathcal{C}^i$. Cost function f^i is the sum of points $\mathcal{P}^i$ reprojection errors in the last frames C_{i-N+1} to C_i:

$$f^i(\mathcal{C}^i, \mathcal{P}^i) = \sum_{C_i \in \{C_{i-N+1}\ ;\ C_i\}} \sum_{p_j \in \mathcal{P}^i} \left(\varepsilon_{ij}^2\right)$$

where $\varepsilon_{ij}^2 = d^2(p_{ij}, K_i p_j)$ is the square of Euclidean distance between $K_i p_j$, estimated projection of point p_j through the camera C_i and the measured corresponding observation. K_i is the projection matrix 3×4 of camera i composed of C_i extrinsic parameters and known intrinsic parameters.

Thus, n (number of optimized cameras at each stage) and N (number of images taken into account in the reprojection function) are the 2 main parameters involved in the optimization process. We must have $N \geq n+2$ to fix the reconstruction frame and the scale factor at the sequence end, and we have found that $n = 3$ or 4 and $6 \leq N \leq 11$ are sufficient values in practice.

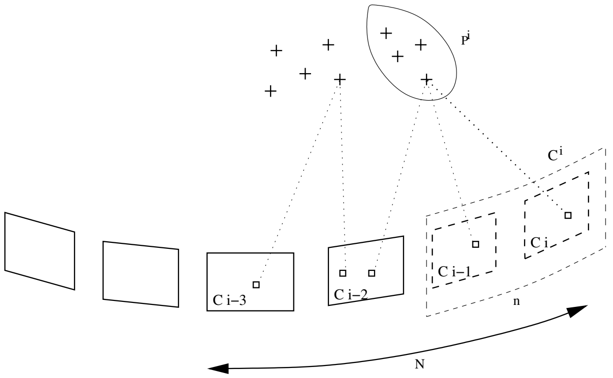

Figure 2. Local bundle adjustment when camera C_i is added. Only surrounded points and cameras are optimized. Nevertheless, we take account of 3D points reprojections in the N last images.

3. Experiments on real data

We applied our incremental localization and mapping algorithm to a semi-urban scene and to a down-town scene. Here, the goal is to evaluate the robustness to perturbations in a complex environment, and the accuracy compared to ground truth provided by a Real Time Kinematics Differential GPS (whose precision is about one inch in the horizontal plane).

Hardware settings In our experiments, the camera was settled on an experimental vehicle and Image size is 512×384 $pixels$. We used a standard Linux PC (Pentium 4 at 2.8 GHz, 1Go of RAM memory at 800 MHZ) for the reconstruction process.

Semi urban scene: comparison with GPS ground truth

- Speed and trajectory length: 70 $meters$ at 1.1 m/s
- Video: 1 min long at 7.5 fps (445 $frames$)
- Reconstruction: 94 key positions and more than 4.000 3D points.

This sequence is particularly interesting because of images contents (people walking in front of the camera, sunshine, etc...) that does not favor the reconstruction process. Moreover, the environment is more appropriate to a GPS localization because the satellites in the sky are not occulted by high buildings. It is also interesting because of the trajectory: a turn on the right, two turns on the left and a straight line.Time measured includes feature detection (#1500 Harris points per frame), matching, and pose calculation for all frames. For key frames, treatment time is longer because of points 3D reconstruction and local bundle adjustment. We can note that speed results are very interesting with an average of 0.09 s for normal frames and 0.28 s for key frames (let us notice that time between two frames is 0.133 s at 7.5 fps). Results are reported in table 1.

Frames	Max Time	Mean Time	Total
Non-key frames	0.14	0.09	30.69
Key frames	0.43	0.28	26.29

Table 1. Computation times in $seconds$.

The calculated trajectory obtained with our algorithm has been compared to data given by a GPS sensor. For the comparison, we applied a rigid transformation (rotation, translation and scale factor) to the trajectory as described in [3] to fit with GPS reference data. Figure 4 shows trajectory registration with GPS reference. As GPS positions are given in a metric frame we can compare camera locations and measure 3D positioning error in meters. The maximum measured error is 2.0 $meters$ with a 3D mean error of 41 $centimeters$ and a 2D mean error of less than 35 $centimeters$ in the horizontal plane.

Very long urban scene

- Speed and trajectory length: $400m$ at 8 km/h
- Video: 3 min 02 long at 15 fps (2731 $frames$)
- Reconstruction: 298 key positions and 18.403 3D points.

In Figure 5, we can see some frames from the video, a classical map of the down-town, and the 3D map resulting from

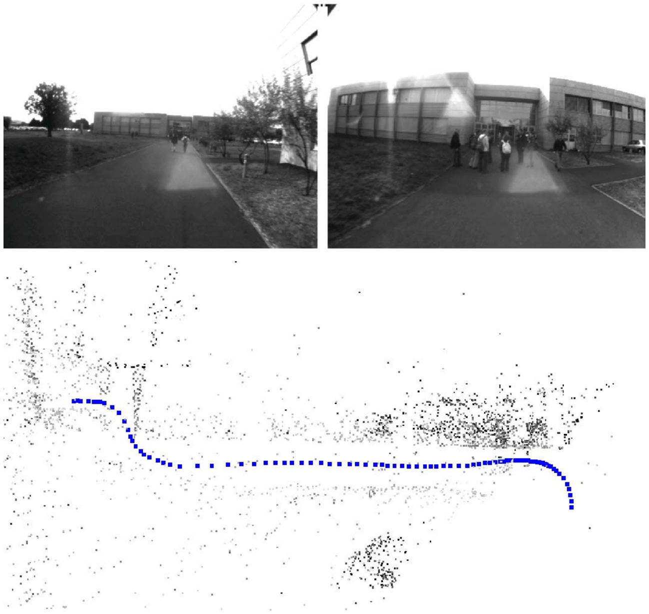

Figure 3. 2 frames from real data experiments and a top view of the reconstructed scene and trajectory (# 4.000 points and 94 key positions).

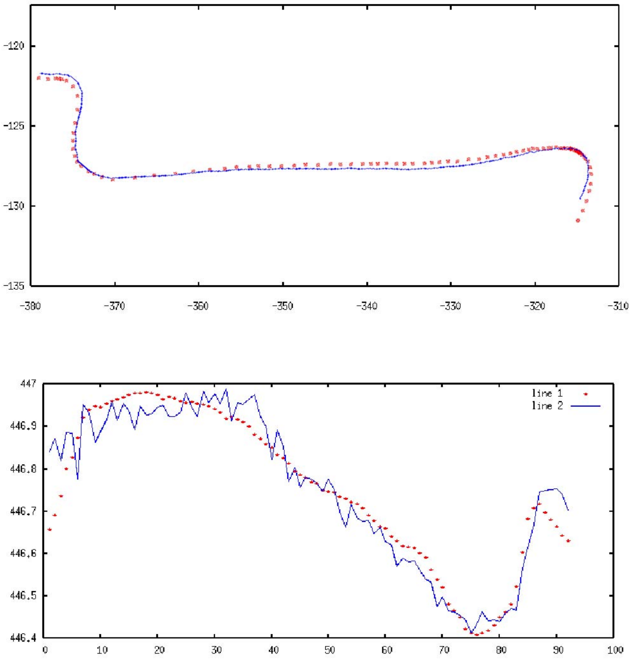

Figure 4. Registration with GPS reference, top: in horizontal plane, bottom: on altitude axis. Continuous line represents GPS trajectory and points represent estimated key positions. Coordinates are expressed in $meters$.

our 3D reconstruction and localization method. The video has been acquired in real urban conditions and the trajectory is nearly a loop; it is not a complete loop because of technical reasons. One can visually ensure that reconstruction is not much deformed and drift is very low compared to the covered distance. That shows that our algorithm, very appropriate to long scene reconstruction in term of computing time is also quite precise and robust. The estimated mean 3D position error compared to results obtained with a classical global bundle adjustment method is less than $3.5cm$.

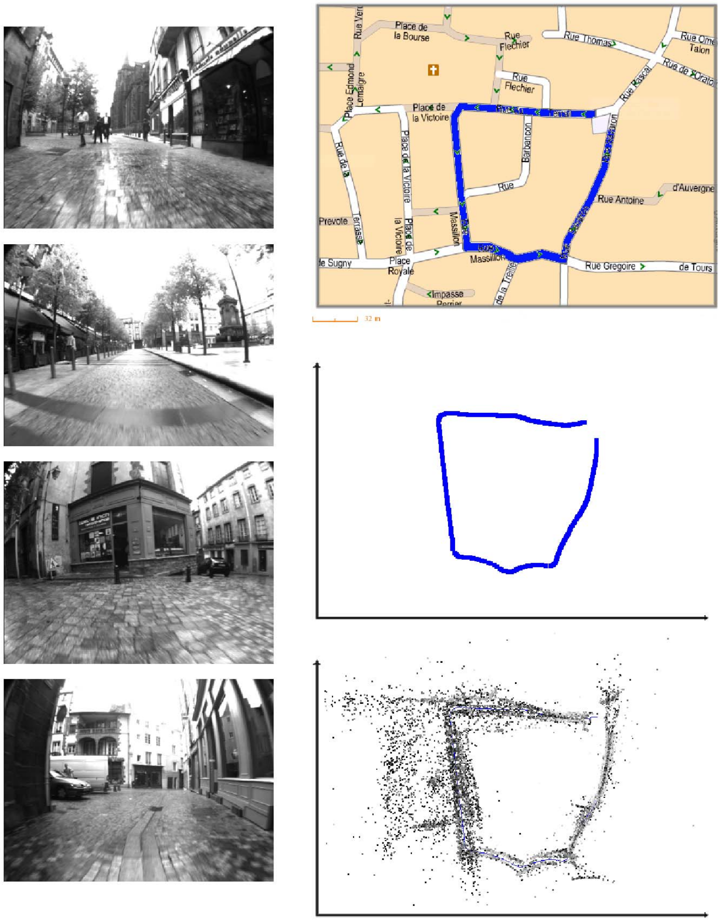

Figure 5. The long urban sequence a) left: some frames from the sequence, b) right: a map of the city with the trajectory in blue, the reconstruction result (trajectory and 18.403 3D points).

Indoor sequences Here, we present 3D reconstruction and trajectory results obtained for 2 indoor sequences. The first one has been acquired with a camera settled on a traveling tripod. The trajectory is a complete loop and the start point is the same as the final point. For the second sequence, the camera was freely handled.

4. Conclusion

We presented a fast and accurate method to estimate a vehicle motion using a calibrated monocular camera. The

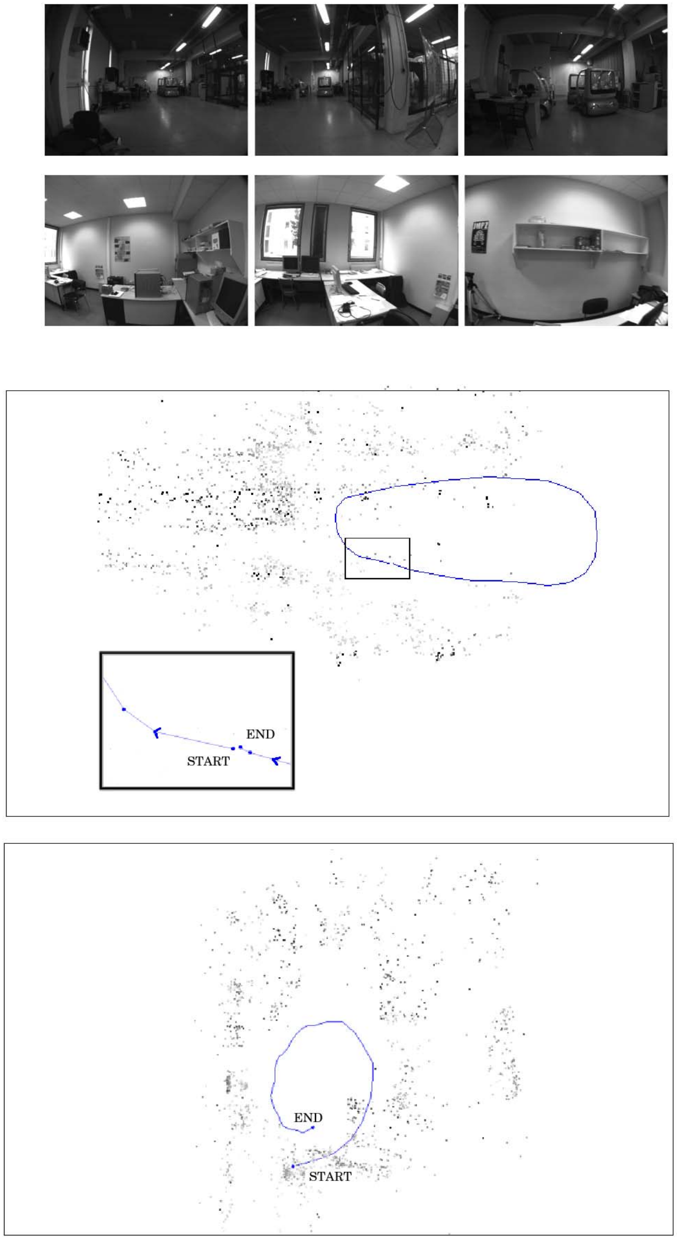

Figure 6. The 2 indoor reconstructions. a)top: some frames from the camera, b)middle: the complete loop, c)the freely handled camera sequence. The line represents the trajectory and points are 3D reconstructed points.

method also gives a 3D reconstruction of the environment, and the model is built with 3D points reconstructed from interest points extracted in images and matched at video rate. Results are very encouraging in term of accuracy.We plan to apply the method to an "automatic convoy of vehicles". The first vehicle is guided manually and makes a 3D map of the scene. It sends data to others vehicles that are able to navigate autonomously, and take the same path. More generally, the method can be adapted to many applications in robotics and many other fields where a 3D localization is needed.

References

[1] "Boujou," *2d3 Ltd*, http://www.2d3.com, 2000.

[2] A.J. Davison. Real-Time Simultaneous Localization and Mapping with a Single Camera. *ICCV'03*.

[3] O.D. Faugeras and M. Hebert. The representation, recognition, and locating of 3-D objects. *IJRR*, 5(3):27-52, 1986.

[4] O.D. Faugeras and Q.T. Luong. *The Geometry of Multiple Images,* The MIT Press, 2001.

[5] M. Fischler and R. Bolles. Random Sample Consensus: a Paradigm for Model Fitting with Application to Image Analysis ans Automated Cartography. *GIP*, 24(6):381-395, 1981

[6] R.M. Haralick, C.N. Lee, K. Ottenberg and M. Nolle. Review and analysis of solutions of the three point perspective pose estimation problem. *IJCV*, 13(3):331-356, 1994.

[7] C. Harris, M. Stephens. A Combined Corner and Edge Detector. *Alvey Vision Conference*, pp. 147-151, 1988.

[8] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision,* Cambridge University Press, 2000.

[9] M. Lhuillier and Long Quan. A Quasi-Dense Approach to Surface Reconstruction from Uncalibrated Images. *IEEE TPAMI*, 27(3):418-433, 2005.

[10] D. Nister. An efficient solution to the five-point relative pose problem. *CVPR'03*.

[11] D. Nister, O. Naroditsky and J. Bergen. Visual Odometry. *CVPR'04*.

[12] D. Nister. *Automatic Dense Reconstruction from Uncalibrated Video Sequences*, PhD Thesis, Ericsson and University of Stockholms, 2001.

[13] M. Pollefeys, R. Koch and L. Van Gool. Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters. *ICCV'98*.

[14] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery. *"Numerical Recipes in C: The Art of Scientific Computing",* Cambridge University Press, 1992.

[15] E. Royer, M. Lhuillier, M. Dhome and T. Chateau. Localization in urban environments: monocular vision compared to a differential GPS sensor. *CVPR'05*.

[16] S. Se, D. Lowe and J. Little. Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks. *IJRR*, volume 21, no.8, pp. 735-758, 2002.

[17] H. Shum, Q. KE, and Z. Zhang. Efficient bundle adjustment with virtual key frames: A hierarchical approach to multiframe structure from motion. *CVPR'99*.

[18] D. Steedly and I.A. Essa. Propagation of Innovative Information in Non-Linear Least-Squares Structure from Motion. *ICCV'01*.

[19] B. Triggs, P.F. McLauchlan, R.I. Hartley and A.W. Fitzibbon. Bundle adjustment - A modern synthesis. *LNCS* volume 1883 , pp. 298-375, Springer Verlag, 2000.

[20] Z. Zhang and Y. Shan. Incremental Motion Estimation through Modified Bundle Adjustment. *ICIP'03*.