Effective Analysis of Image Sequences for 3D Camera Motion Estimation

Andrea Dell'Acqua, Augusto Sarti, Stefano Tubaro

Dipartimento di Elettronica e Informazione
Politecnico di Milano
Piazza L. Da Vinci 32, 20133 Milano
Italy

ABSTRACT

In this paper we propose an effective technique for the estimation of the 3D camera motion based on the analysis of the acquired image sequence. The method is based on the detection and the tracking of significant point features for a joint estimation of camera motion and 3D coordinates of the features points. The approach is based on Extended Kalman Filtering, and enables an on-the-fly feature replacement when the information provided by them becomes too degraded.

1. INTRODUCTION

Given a video sequence, the joint estimation of the 3D scene geometry, of the camera motion and of its *optical* and *geometric* parameters is a well-known problem known as *Structure from Motion (SfM)*. SfM techniques available in the literature [1, 5, 6, 7, 8, 9, 10, 11, 12, 13]) apply specific constraints on the unknowns, which can be roughly classified in *geometrical* and *physical-temporal* ones. Geometrical constraints defines relationships between the 3D structure and its projection onto the available views. Physical-temporal constraints are normally used when dealing with image sequences acquired by a moving camera. Indeed, the camera is not expected to jump from one position to another, therefore the temporal evolution of the motion parameters can be assumed as smooth.

In general, both classes of solutions are based on the detection and the tracking of significant point features [14], therefore we need a joint estimation of camera motion and 3D coordinates of the features points. Depending on the constraints it is possible to classify *SfM* algorithms in two main classes. In the first category are those that works on a small set of images (tipically pairs or triplets). These solutions usually work in just one pass, and rely on principles of multi-view geometry (epipolar and multilinear constraints). Furthermore, they normally need further processing to obtain motion trajectories that exhibit a certain temporal continuity. A second class of solutions are the *differential* algorithms. These methods which combine physical constraints with geometrical constraints in order to recover the incremental evolution of the parameters; and this is done frame by frame.

The approach that we propose is in this latter category. In fact, it assesses the *SfM* problem using an *(Extended) Kalman filtering* approach [1, 2, 3, 4].

2. STATE PARAMETERS

The parameterization that we adopted is the one proposed in [1]. The projection model is a central one, where the origin of the coordinate system lies on the image plane instead of the camera's optical center. The 3D location of each point is

$$\begin{pmatrix} X^0 \\ Y^0 \\ Z^0 \end{pmatrix} = \begin{pmatrix} u^0 \\ v^0 \\ 0 \end{pmatrix} + \alpha \begin{pmatrix} \frac{u^0}{f} \\ \frac{v^0}{f} \\ 1 \end{pmatrix} \quad (1)$$

where α is the depth of the point referred to the first frame (Z^0), f is the camera focal length and (u^0,v^0) are the image coordinates of the point on the first frame. The 3D structure is thus characterized by only one parameter per point (depth), instead of three (see Fig. 1).

As for camera 3D motion, the impact on the location of the features of a camera motion along the optical axis is much less noticeable than that of a translational motion that is parallel to the image plane. In order to improve the EKF performance, the three parameters that we estimate are t_x and t_y (translation parallel to the image plane), plus the ratio t_z/f [1, 2]. The 3D global rotation is parametrized by a unit quaternion, while the interframe rotation in the state vector is estimated for each frame using incremental Euler angles.

In conclusion, the state vector is made of two portions: the former includes 7 parameters (t_x, t_y, $t_z\beta$, ω_x, ω_y, ω_z, f), six for the camera motion and one for the camera's internal geometry; and the latter includes the n parameters that describe the 3D point locations, which the depths ($\alpha_{1 \ldots}\alpha_n$) of the n tracked features. The available measurements at each step are the 2D feature coordinates in the current frame. As the camera motion is completely unknown, there is no a-priori information on the system dynamics, therefore the EKF updating model suggested in [1, 2, 3, 4] is a simple *random walk*.

The state vector will have some parameters that change continuously and, at the same time, parameters that tend to settle to a constant value. Typically, motion and focal length are *dynamic* parameters (zooming often occurs in a video sequence), but some structure parameters may exhibit a rather stationary behavior (e.g. the focal length f is often fixed). The management of the variance set of the parameters is quite critical. In fact, choosing a high variance for a parameter makes it "reactive" but more sensitive to noise. On the other hand, a static parameter (close to convergence) tends to have a low variance, which limits its noise sensitivity. In order to guarantee a good stability, a balance between parameters of each type must be guaranteed in the state vector.

3. MOTION ESTIMATION STRATEGY

The aim of our work is to devise and implement an effective technique for the estimation of the 3D camera motion. The use we have for this type of information is for a complete reconstruction of a 3D scene or an object through a variety of methods. Indeed, with this goal in mind, the availability of 3D information on the feature points can be exploited to our advantage. With respect the original approach proposed by Pentland [1, 2], we introduced a number of improvements that will be illustrated in the following.

3.1. Feature replacement

Expecting that the features selected on the first frame will be trackable throughout the sequence, is a rather optimistic assumption. Unfortunately, removing a feature point and introducing a new one, has the effect of introducing a larger depth error in one of the parameters of the state vector, which causes a parameter transient that affects all the the other parameters. A good solution for this problem can be obtained by estimating the depth of the new feature and the corresponding camera motion information in advance (before the insertion point). Using back-projection of the 2D location it is possible to estimate the 3D position of the new feature. Usually it is sufficient to consider 10-20 frames (depending on the camera motion) to obtain a good estimation of the correct depth value.

3.2. Selecting features

Although the pre-estimation of the feature's depth greatly reduces the impact of its insertion in the model, it is usually preferable to avoid too many feature replacements. In order to minimize the replacements, we establish a strong link between the *tracking* and the *3D-modeling* modules. The idea is to perform only those replacements that will maximize the feature's lifespan, in order for them to provide us with the maximum information without perturbing the estimation process. This can be achieved by tracking a number of features that is much larger than what the Kalman filter needs for its computations (from three to five time as many), and then cleverly *selecting* those that are, in fact, used.

3.3. Variations of the focal length

The 3D structure estimated by the Kalman filter strictly depends on the focal length. Since the 3D structure can only be estimated up to a scale factor, this can cause some problems. In order to determine the global scale factor one depth needs to be locked to a fixed value (α_{scal}^{fix}): this way the scaled depth (α_{scal}) of a 3D point changes depending on focal length (eq.2)

$$\alpha_{scal} = (\alpha_{\mathrm{real}} + f)\left[\frac{\alpha_{scal}^{fix} + f}{\alpha_{\mathrm{real}}^{fix} + f} - f\right] \quad , \qquad (2)$$

therefore a change of f requires a global resettling of all depths $\alpha_{j,j=1\ldots n}$ in the state vector.

Allowing all parameters to have a dynamic behavior is usually quite dangerous for filter stability, therefore we need to lock the 3D structure. The introduction of scale factors in the equations that govern the filter, depending on the actual feature depths, enables the system to robustly track a time-varying focal length.

4. CONCLUSIONS

Compared to other motion estimation techniques, the proposed algorithm exhibits a remarkable ability to recover focal length changes and an impressive robustness against feature replacements. We tested our algorithm both on synthetic videos and on real sequences acquired with a hand-held camera. Where the parallax covered by the camera is wide enough (15-20 degree are sufficient depending on the amount of noise), the proposed approach performed particularly well. Some results relative to the performance of the proposed algorithm is presented in the following figures (2,3).

5. REFERENCES

[1] A. Azarbayejani, A.P. Pentland, "Recursive Estimation of Motion, Structure, and Focal Length", *IEEE Tr. PAMI*, Vol.17, No. 6, pp. 562-575, June 1995.

[2] T. Jebara, A. Azarbayejani, A. Pentland, "3D Structure from 2D Motion", *IEEE Signal Processing Magazine*, Vol.16, No. 3, pp. 66-84, Jan. 1998.

[3] G. Calvagno, R. Rinaldo, L. Sbaiz, "Three-Dimensional Motion Estimation of Objects for Video Coding", *IEEE J. Sel. Areas In Communications*, Vol. 16, No. 1, pp. 86-97, Jan. 1998.

[4] A. Chiuso, S. Soatto, "3-D Motion and Structure Causally Integrated Over Time: Theory (Stability) and Practice (Occlusions)", *ESSRL* Technical Report 99-003, October 1999.

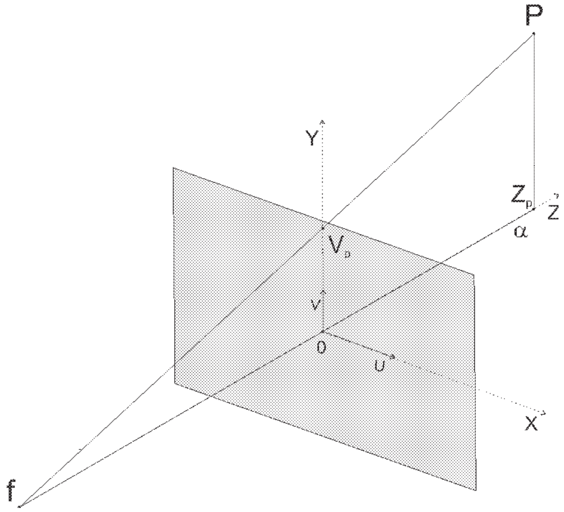

Figure 1: Adopted projective model. α is the depth of the point referred to the first frame (Z^p), f is the camera focal length and (u^p, v^p) is the location of the point on the first mage of the sequence.

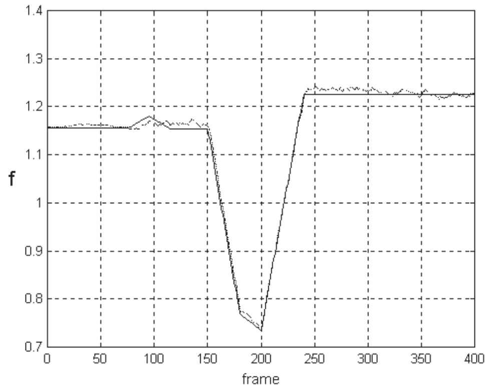

Figure 2: Application of the proposed algorithm to a synthetic case. The 3D scene is made of 16 points uniformely distributed in a volume of 1 m. The camera moves around the scene covering an angle of 120 degrees. Moreover, the camera changes its focal length between frames 75 and 115 and between 150 and 238. An uniformely distributed noise (between -0.8 and 0.8 pixels) is added to the 2D coordinates of the considered points. In the figure, the evolution of the estimated focal length (in gray) is compared to the real one (in black). The focal length is a very critical parameter and the quality of its estimation leads to a good estimation of the other parameters.

[5] L. Van Gool, A.Zissermann, “Automatic 3D Model Building from Video Sequences”, *European Transactions on Telecommunications*, Vol. 8, No. 4, pp. 369-378, Jul.-Aug. 1997.

[6] R. Koch, M. Pollefeys, B. Heigl, L. Van Gool, H. Niemann, “Calibration of Hand-held Camera Sequences for Plenoptic Modeling”, *ICCV’99*, pp. 585-591, Corfù, Greece, 1999.

[7] R. Koch, M. Pollefeys, L. Van Gool, “Realistic 3D Scene Modeling from Uncalibrated Image Sequences”, *ICIP-99*, Oct. 25-28, 1999, Kobe, Japan.

[8] C.J. Tsai, A.K. Katsaggelos, “Sequential Construction of 3D-Based Scene Description”, submitted to, *IEEE Tr. Circuits and Systems for Video Technology.*

[9] J. Oliensis, J.I. Thomas, “Incorporating Motion Error in Multi-Frame Structure from Motion”, *IEEE Workshop on Visual Motion*, Los Alamitos, CA, pp. 8-13, Oct. 1991.

[10] M. Pollefeys, “Self-Calibration and Metric 3D Reconstruction From Uncalibrated Image Sequences”, Ph.D. Thesis, ESAT-PSI, K.U.Leuven, May 1999.

[11] H. Longuet-Higgins, “A Computer Algorithm for Reconstructing a Scene from Two Projections”, *Nature*, 293:133-135, 1981.

[12] R. Tsai, T. Huang, “Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces”, *IEEE Tr. on PAMI*, Vol. 6, pp. 13-27, Jan. 1984.

[13] W.Fitzgibbon, A.Zissermann, “Automatic Camera Recovery for Closed or Open Image Sequences”, *ECCV’98*, Vol.1, Lecture Notes in Computer Science, Vol. 1406, Springer-Verlag, 1998, pp. 311-326.

[14] J. Shi, C. Tomasi, “Good Features to Track”, CVPR, pp.593-600, 1994.

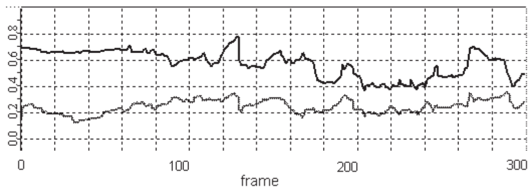

Figure 3: Typical back-projection error in a real sequence of 300 frames, where all the 16 features are replaced between frame 65 and 190. Mean value (in black) and standard deviation (in gray).