MVA2000 IAPR Workshop on Machine Vision Applications, Nov. 28–30, 2000, The University of Tokyo, Japan

12—5

Shape and Motion Estimation from Geometric Primitives and Parametric Modelling

Pierre-Louis Bazin* Jean-Marc Vézien André Gagalowicz

I.N.R.I.A. Rocquencourt,
France

Abstract

This paper presents a new approach to shape and motion estimation based on geometric primitives and relations in a model-based framework. A description of a scene in terms of structured geometric elements sharing relationships allows to derive a parametric model with Euclidian constraints, and a camera model is also proposed to reduce the problem dimensionality. It leads to a sequential MAP estimation, that gives accurate and comprehensible results on real images.

1 Introduction

The problem of recovering geometric information about the 3D world from streams of images has been proved to be solvable by various means [12, 13, 14, 8, 4, 5], but few practical solutions have been carried out. Depending on the applicative field, some approaches may give unsatisfactory results or rely on unadapted hypotheses.

In video post-production (our primary application), fast and precise algorithms for motion tracking, camera motion recovery and 3D reconstruction are important tools for tasks such as special effects generation and augmented reality. Here, the visual quality of the result and the ease of use are the principal goals. Real-time computation is not needed, and an operator can feed the algorithm with *a priori* knowledge of the scene described in the images.

The method proposed here uses geometric primitives namely line segments, rectangles and trihedral corners as base features. Geometric relationships of orthogonality, parallelism, collinearity and coplanarity are also specified between the primitives. Most algorithms for 3D shape and motion estimation are based on feature points or lines. Complete 3D models have also been investigated [6, 10], but there has not been much work on using such intermediary features or geometric relationships between distinct objects.

The primitives of interest are interactively defined by the user on a first key frame, then tracked along the image sequence. Specified geometric relationships are integrated by a reduction algorithm in a parametric model of the 3D scene, and the camera path is also modelled by motion parameters. Finally, the maximum-a-posteriori estimator of the parameters is obtained, and a frame-by-frame non-linear optimisation process allows to compute it efficiently. We present an example of tracking, modelling and reconstruction from real images and finally discuss remaining error sources and possible improvements.

2 Primitive Tracking

Geometric primitives are our base features. Their use allows more precise and robust tracking than points, by feeding explicit geometric information into the process. Our tracking algorithm extends basic methods of point correlation trackers with a robust edge matching technique, and also makes explicit use of 3D-perspective models, whenever possible, to better fit the observed deformations of projected objects (cf [3] for further details).

The algorithm outputs stable primitive tracks upon long image sequences (cf fig.1), $\{Y_{it}\}_{i=0..N,t=0..T}$, where Y_{it} are the image coordinates of the primitive i on the frame t. The dimension of Y_{it} depends on the primitive type.

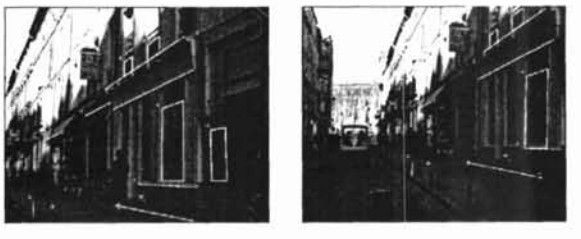

Figure 1: An example of tracking: the praxitele sequence, frames 1 and 30 with tracked primitives.

*Address: Domaine de Voluceau, Rocquencourt B.P.105, 78153 Le Chesnay Cedex, France. E-mail: pierre-louis.bazin@inria.fr

Along the sequence, there are possible undefined positions due to loss of track, occlusion or motion out of the camera field. Moreover, each kind of primitive model leads to different noise on positions, and the tracking quality score issued by the algorithm is also available. To use all this information, an inverse variance matrix $\mathbf{V}_{it}^{-1}$ is associated to the primitive tracks. The matrix coefficients are weighted by the quality score or set to zero when the primitive is lost, and variance ellispes are oriented along primitive edges.

3 Geometric Scene Reduction

Following the tracking process, geometric relations between primitives are introduced. The possible relations are parallelism, orthogonality, collinearity and coplanarity, which are general for any 0D, 1D and 2D primitives. These constraints, along with inner constraints of specific primitives like rectangles or corners, make the raw description of the scene in terms of points or separate shapes over-determined. To cope with it, the use of a constraint solver [11] along with linearised constrained minimisation or constraint space-projected minimisation [7] would be far from optimality and the knowledge from relations would remain under-used.

A better approach consists in searching for a parametric representation that directly merges the relations within a reduced set of parameters. To perform this geometric reduction, algebraic reduction methods and regular rewriting systems are not well adapted [1].

We created a specific geometric reduction algorithm to derive the minimal set of unconstrained parameters from the initial points and their relationships (cf [2] for further details). This reduction is possible with any set of primitives and relations, and transforms the geometric scene into a model with as many dimensions as there are degrees of freedom in the system. This reduced model enforces all relations, reduces the search space by several dimensions, and separates the highly correlated scene points into low-correlation parameters.

Moreover, the transform procedure can be interpreted as a regular function, and its derivatives can be computed analytically. If we set $\{X_i\}$ as the 3D point coordinates of the projected primitives $\{Y_{it}\}$, we perform a reduction process that constructs the functions $X_i = \hat{X}_i(P_S)$ for all i, where P_S are the reduced scene parameters.

4 Camera Modelling

Once we have a geometric model of the scene, we must apply a projection and motion model for the camera. We use a pinhole camera model, that performs the following projection for a 3D point P:

$$\begin{array}{l} u = f(x + u_0) \\ v = f(y + v_0)/r \end{array} \quad \text{with} \quad \begin{array}{l} x = \frac{(R \cdot P + T)_X}{(R \cdot P + T)_Z} \\ y = \frac{(R \cdot P + T)_Y}{(R \cdot P + T)_Z} \end{array}$$

The center of projection (u_0, v_0) and the aspect ratio r are supposed known by the user and fixed. The focal length f can be fixed or moving, and a relevant approximation of its value is supposed known. The remaining parameters are the translation vector T and the rotation matrix R, that change with the camera motion.

For each frame t, the translation vector T_t, the three pose angles θ_t that determine the rotation $R_t = R(\theta_t)$ and possibly the focal length f_t are the motion primary parameters. In a model-based framework, such parameters provide too many degrees of freedom on the motion. As the camera motion is continuous and rather smooth, a parametric model must be used to reduce the number of free motion parameters. To do this, we use Chebyshev polynomials to model the motion and pose curves:

$$\begin{array}{l} T_t = \sum_{k=0}^{K} \mathbf{a}_k P_k(t) \\ \theta_t = \sum_{k=0}^{K} \mathbf{b}_k P_k(t) \\ f_t = \sum_{k=0}^{K} c_k P_k(t) \end{array}$$

where $P_k(t)$ is the Chebyshev polynomial of degree k. Therefore, the motion parameters become $P_M = \{\mathbf{a}_k, \mathbf{b}_k, c_k\}$, reducing the degrees of freedom from $7T$ to $7(K+1)$ parameters. The degree K of the polynomials is a hyper-parameter of the problem, that must be set arbitrarily by the user.

5 Parametric Estimation

The modelling of the scene and the camera leads to a non-linear parametric model, which embodies our prior geometric knowledge of the scene. The only statistical knowledge available *a priori* is the model of the tracking errors, that are assumed to be Gaussian perturbations. Following the statistical theory of Kanatani [9], we derive a maximum-a-posteriori (MAP) estimation in a Bayesian viewpoint:

$$\begin{array}{rl} \hat{P} = & \arg\max p(P|\{Y_{it}\}, M) \\ = & \arg\max p(\{Y_{it}\}|P, M) p(P|M) \end{array}$$

where $P = \{P_S, P_M\}$ is the shape/motion parameter vector and M is the prior modelling knowledge.

If we suppose the measurement noise to be Gaussian with variance $\mathbf{V}_{it}$ from the tracking, we have:

$$p(\{Y_{it}\}|P, M) = \prod_{i,t} \frac{1}{\sqrt{(2\pi)^n |\mathbf{V}_{it}^{-1}|}} \times \exp -\tfrac{1}{2}(Y_{it} - \hat{Y}_{it}(P)) \mathbf{V}_{it}^{-1} (Y_{it} - \hat{Y}_{it}(P))$$

where $\hat{Y}_{it}(P)$ is the projection of the primitive $\hat{X}_i(P_S)$ on the frame t. With no *a priori* assumptions on the parameters P, we can set $p(P|M)$ as a fixed constant, to finally derive a weighted least-square estimator:

$$\hat{P} = \arg\min \sum_{i=0}^{N} \sum_{t=0}^{T} (Y_{it} - \hat{Y}_{it}(P)) \mathbf{V}_{it}^{-1} (Y_{it} - \hat{Y}_{it}(P))$$

Since the modelling functions $\hat{Y}_{it}(P)$ are non-linear, an iterative optimisation algorithm is needed to perform the minimisation. We use the Levenberg-Marquardt algorithm [15], which is particularly adapted to least-square forms and has fast convergence even with non-linear functions. It needs the computation of gradients, that are analytically obtained in the reduction step, and a good first estimate of the parameters, that can be hard to provide.

To overcome the first estimate problem, we proceed in a frame-by-frame basis. A Bayesian formula states that:

$$p(P|\{Y_{it}\}_{t=0..T}, M) \sim \frac{p(\{Y_{it}\}_{t=0..T}|P,M)}{p(\{Y_{it}\}_{t=0..T-1}|P,M)} p(P|\{Y_{it}\}_{t=0..T-1}, M)$$

Thus, if we take the negative log of the formula, with our Gaussian assumption, the function to minimise at the frame T is the sum of the function already minimised at frame $T-1$ and the least square error on the new data at frame T, so $\hat{P}_{T-1}$ is an efficient first guess for $\hat{P}_T$. An initial guess for the parameters must be computed for $T = 0$, when there is no camera motion: the scene can be derived from an arbitrary flat reconstruction of the projected points (cf fig.2). The initial parameters are

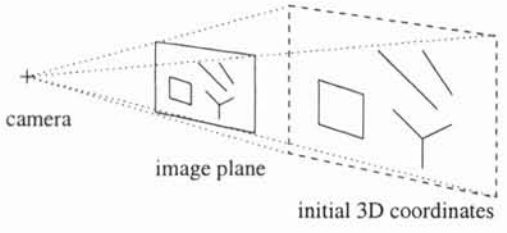

Figure 2: The flat reconstruction from first frame.

computed in the reduction step from these points, then the Levenberg-Marquardt minimisation is performed. This method avoids the computation of a complex initial solution and increases the minimisation complexity gradually as the number of frames increases.

6 Results and Discussion

The complete algorithm has been tested on various video sequences[1] with different kinds of camera motion, giving low residual errors as well as an acceptable reconstructed shape (cf fig.3). The camera

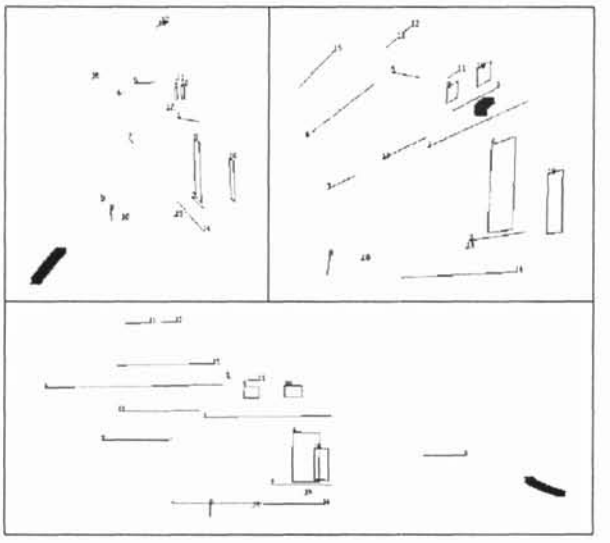

Figure 3: Reconstruction for the praxitele sequence: top-left: front view, top-right: near camera view, bottom: side view. The black thick line is the camera path in the view.

motion is realistic and free of the jittering effects often observed in unconstrained motion recovery (cf fig.4). The estimated shape and motion are both

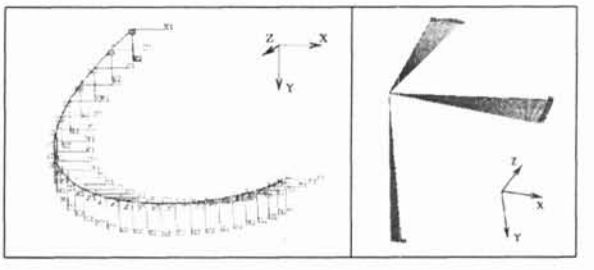

Figure 4: Camera estimate for the praxitele sequence: left: motion, right: pose. The camera real motion was a panoramic-like rotation around a vertical axis, and the direction of view is along the Z axis.

Euclidian, and allow direct integration of virtual elements in the scene (cf fig.5). Computation times are acceptable: in the praxitele example, the tracking step needed 12 minutes 30, the reduction step 30 milliseconds and the estimation step 7 minutes, with 19 primitives in 30 frames.

[1] additional results are available at: http://www.multimania.com/piloubaz/research.htm.

Figure 5: The praxitele sequence with virtual elements added: frames 1 and 30.

The remaining errors and shape distortion problems depend on three main factors. First, there exist many possibilities for the underlying shape and motion when the visual motion and the primitives are simple. These possibilities are augmented with the noise on tracked primitives. Next, the only metric informations present in the model are the coplanarity and collinearity relations, so that distortions on distances and dimensions can still accommodate for the noise. Finally, the camera model depends on a unique hyper-parameter K arbitrarily set, regardless of the effective motion.

The first two factors can be handled by improvements in the tracking algorithm to reduce noise or characterise it more accurately. Adding primitive length information should also input more dimensional constraints and reduce the number of equivalent solutions, as would do more complex primitives than segments.

The camera model has to be improved to better fit the real motion. Model-selection methods can be used in frame-by-frame basis, and remove the hyper-parameter problem. The set of models to choose from should also be more flexible.

7 Concluding Remarks

An algorithm for shape and motion estimation from geometric primitives has been presented. We have shown that the introduction of geometric relations and motion smoothness in a model-based strategy offers realistic results and lower the motion errors in an optimal way. Reconstructed shape and motion are both structured, and almost free of noise in various real video streams experiments.

References

[1] P. Balbiani, V. Dugat, L. F. nas del Cerro, and A. Lopez. *Eléments de géométrie mécanique.* Hermes, 1994.

[2] P. L. Bazin. A parametric scene reduction algorithm from geometric relations. In *Proc. Vision Geometry IX*, SPIE's annual meeting, San Diego, 2000.

[3] P. L. Bazin and J. M. Vézien. Tracking geometric primitives in video streams. In *Proc. Irish Machine Vision and Image Processing*, pp 43–50, Belfast, 2000.

[4] S. Christy and R. Horaud. Fast and reliable object pose estimation from line correspondances. In *Proc. Computer Analysis of Images and Patterns*, Kiel, 1997.

[5] D. B. Gennery. Visual tracking of known three-dimensional objects. *Int. J. Computer Vision*, 7(3):243–270, 1992.

[6] P. Gérard, J.-M. Vézien, and A. Gagalowicz. Three dimensional model-based tracking using texture learning and matching. In *Proc. Scandinavian Conf. on Image Analysis*, Kangerlussuaq, 1999.

[7] J. C. Gilbert. Optimisation: théorie et algorithmes, octobre 1998. course notes, from www-rocq.inria.fr/ gilbert/ensta/optim.html.

[8] R. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proc. Computer Vision and Pattern Recognition*, pp 761–764, 1992.

[9] K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice.* Elsevier, Amsterdam, 1996.

[10] D. Koller, K. Daniilidis, and H. H. Nagel. Model-based object tracking in monocular image sequences of road traffic scenes. *Int. J. Computer Vision*, 10(3):257–281, 1993.

[11] G. Kwaiter, V. Gaildrat, and R. Caubet. Modelling with constraints: a bibliographical survey. In *Proc. Int. Conf. on Information Visualization*, pp 211–220, London, 1998.

[12] S. J. Maybank and O. Faugeras. A theory of self-calibration of a moving camera. *Int. J. Computer Vision*, 8(2):123–151, 1992.

[13] C. J. Poelman and T. Kanade. A paraperspective factorization method for shape and motion recovery. *Trans. Pattern Analysis and Machine Intelligence*, 19(3):206–218, 1997.

[14] M. Pollefeys, R. Koch, and L. V. Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. Int. Conf. Computer Vision*, pp 90–95, 1998.

[15] W. Press, W. Vetterling, S. Teukolsky, and B. Flannery. *Numerical recipes in C.* Cambridge University Press, 1993.