

# Motion Curves for Parametric Shape and Motion Estimation

Pierre-Louis Bazin and Jean-Marc Vézien

INRIA Rocquencourt,  
Domaine de Voluceau, Rocquencourt B.P.105,  
78153 Le Chesnay Cedex, France

**Abstract.** This paper presents a novel approach to camera motion parametrization for the structure and motion problem. In a model-based framework, the hypothesis of (relatively) continuous and smooth sensor motion enables to reformulate the motion recovery problem as a global curve estimation problem on the camera path. Curves of incremental complexity are fitted using model selection to take into account incoming image data. No first estimate guess is needed. The use of modeling curves lead to a meaningful description of the camera trajectories, with a drastic reduction in the number of degrees of freedom. In order to characterize the behaviour and performances of the approach, experiments with various long video sequences, both synthetic and real, are undertaken. Several candidate curve models for motion estimation are presented and compared, and the results validate the work in terms of reconstruction accuracy, noise robustness and model compacity.

**Keywords:** Structure from motion, camera modeling, model selection, motion curves, model-based estimation.

## 1 Introduction

The structure and motion estimation problem is one of the central concerns of computer vision. It can be formulated in several ways, depending on the requirements of the underlying applications. Real-time constraint, stereo imaging, availability of a complete image sequence or of a human operator are important elements that can make some solutions better than others. In our primary application, i.e. video post-production, fast and precise algorithms dedicated to camera motion recovery and 3D reconstruction are key tools for tasks such as special effects generation and augmented reality. Here, the visual quality of the result and the ease of use are important constraints but an operator can feed the algorithm with high level a priori knowledge of the scene described in the images.

Camera motion is very constrained in professional video imaging: the motion is generally very fluid, even for highly composite motions. Some sequences can be very complex, including various compositions of translations and rotations, while others are pure translations along a line or panoramic-like rotations. Curiously, most of the current algorithms for shape and motion recovery do not take advantage of such constraints. Some approaches based on filtering have also suggested the use of motion models to predict the camera position at one frame from previous ones [6] [4] [7]. Model selection between motion models have been proposed in [9] [10] [11]. In these approaches, the

different models to choose from describe the motion between two images. Regular video sequences in post-production may span several seconds, with 25 or 30 frames per second, and the number of motion parameters to be estimated would remain very high, even with the simplest motion models. A better solution in such cases would be to model the complete motion of the camera, in the entire sequence, regardless of the number of images.

A natural way of modeling such motion is to represent the camera trajectory as a 3D curve. Translation and rotation can be separated and modeled independently. The remaining problem is to fix the curve complexity: a parametric curve with many parameters will properly fit very complex motions, but will overfit simpler motions. On the other side, simple curves will perform well only for simple motions, so we need to estimate the real complexity of the unknown camera motion and the corresponding motion parameters simultaneously. An alternative modeling strategy would be to classify the most usual types of motion, such as uniform camera traveling or panoramic motion, then model each of them separately and test every hypothetical motion model. This approach has two disadvantages: it requires the estimation of all different models, and remains highly irrelevant for unusual, erratic or noisy motion.

Both aspects of modeling the trajectories and estimating the motion complexity are addressed in the present work. With a parametric estimation viewpoint, we can describe the shape and motion problem using models, linear or not, that make a direct link between observed data in the images, the underlying scene structure and the camera motion. The structure and motion problem then becomes a parametric estimation problem, in which the free parameters of the models have to be estimated to optimally fit the observed data. In this framework, we have designed a sequential estimation procedure and coupled it with a model selection step to compute the optimal complexity for the estimated motion. Different kinds of motion curves can be integrated in the procedure, and we present several general models based on polynomials.

The motions we consider are supposed to be smooth camera motions, and we show that the approach allows in that case an impressive reduction of the number of free parameters in the system, as well as accurate motion recovery. Under the term *smooth* motion, we do not consider  $C^\infty$  functions but only continuous motions with reasonable acceleration. The technique can handle some trembling and fast changes in the camera motion.

In section 2, we briefly outline the estimation procedure allowing the computation of the trajectory models. A Bayesian selection procedure is proposed in section 3 to set the complexity of the curves. Section 4 reviews the motion curves we have tested. Finally, section 5 presents estimation results on both simulated and real sequences and section 6 discuss them.

## 2 Overview of the Method

The problem of shape and motion recovery we address can be summarized as follows: we observe  $K$  geometric primitives of dimension  $d$  in each of  $T$  images, through primitive tracking. Different parametric models are proposed to recover structure and motion, sharing the same shape model but with more or less complex motion models. With a

Bayesian estimator coupled with an adapted model selection criterion, we proceed to compute the optimal model and associated parameters that best interpret the data.

### 2.1 Scene Description

Our applicative context has led us to consider as primitive features points, segments and rectangles. Moreover, we allow the user to define additional relations of parallelism, orthogonality, collinearity and coplanarity (see Fig.1).



- primitives:
- 11 segments,
- 10 rectangles.
- relations:
- 6,7,11,20 : coplanar,
- 2,3,4,5 : coplanar,
- 17,18,19 : orthogonal,
- 9,13 : orthogonal,
- 2,20 : orthogonal,
- 13,15,19 : parallel,
- 11,12,17 : parallel.

**Fig. 1.** A scene, as described by the user. The primitives are drawn on a key frame and relations are specified beside.

**Geometric reduction.** 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, a specific geometric reduction algorithm is used to derive the minimal set of unconstrained parameters from the initial primitives and their relationships (see [1] for a detailed explanation). This reduction procedure is complementary to the motion modeling scheme we develop in the present work.

### 2.2 The Projection Model

In video imaging, a pinhole camera model gives a generally good approximation of the real camera. It performs the following perspective projection for a 3D point  $P$ :

$$\begin{aligned}
 u &= f(x + u_0) \\
 v &= f(y + v_0)/r
 \end{aligned}
 \quad \text{with} \quad
 \begin{aligned}
 x &= \frac{(\mathbf{R} \cdot (\mathbf{P} - \mathbf{T}))_x}{(\mathbf{R} \cdot (\mathbf{P} - \mathbf{T}))_z} \\
 y &= \frac{(\mathbf{R} \cdot (\mathbf{P} - \mathbf{T}))_y}{(\mathbf{R} \cdot (\mathbf{P} - \mathbf{T}))_z}
 \end{aligned}$$

We will assume that intrinsic parameters are known and constant, with the exception of the focal length. This assumption is reasonable, as the parameters can be either obtained

in camera specs, or computed through off-line calibration. The center of projection  $(u_0, v_0)$  and the aspect ratio  $r$  are fixed by the user and a rough approximation of the focal length value  $f$  is supposed known. The remaining parameters are the translation vector  $\mathbf{T}$  and the rotation matrix  $\mathbf{R}$ , that represent the change of coordinates from a reference coordinate system to the camera coordinate system.

Independently from the projection model, the camera moves at each frame  $t$ , so the translation vector  $\mathbf{T}_t$  and the three pose angles  $\theta_t$  that determine the rotation  $\mathbf{R}_t = \mathbf{R}(\theta_t)$  are changing through time.  $(\mathbf{T}_t, \theta_t)$  reflect the camera motion from one frame to the next: they are directly used as motion parameters in most structure and motion algorithms. In the present approach,  $\mathbf{T}_t$  and  $\theta_t$  are modeled with the 3D curves proposed in Sect.4.

### 2.3 Primitive Tracking

The primitives specified in the first image frame are tracked in the images of the sequence, with a correlation and matching procedure (see Fig.2). Some of them are lost due to



Fig. 2. Tracking for the House Sequence: frames 1, 8, 15, 21.

occlusion, loss of track or when they get out of the camera field of view. The relationships between primitives are not enforced in that step, but the inner shape of rectangles is preserved. The 2D tracks obtained are not complete, and are corrupted by noise. This step will be avoided (i.e. simulated) in the synthetic experiments. Details on the actual algorithm dedicated to the tracking step can be found in [2].

### 2.4 Bayesian Estimation

In order to compute the parameters of the scene and the camera motion from noisy 2D tracks, the problem is formulated in a Bayesian estimation framework [5].

**Maximum a posteriori.** We use the *Maximum a Posteriori* estimator (MAP):

$$\hat{\Theta}_{MAP} \triangleq \arg \max_{\Theta} p(\Theta | \{X_{k,t}\} MI) \\ = \arg \min_{\Theta} [-\log p(\{X_{k,t}\} | \Theta MI) - \log p(\Theta | MI)]$$

to obtain the set of shape and motion parameters  $\Theta$  that best explains the observed data  $\{X_{k,t}\}$ , where  $X_{k,t}$  are the coordinates of the primitive  $k$  in the image frame  $t$ . The prior

information  $I$  represents the context of the problem and the model  $M$  represents the final parametric model for both shape and motion, that depends on the set of parameters  $\Theta$  to be estimated.

We set  $p(\{X_{k,t}\}|\Theta MI)$  to be a Gaussian distribution centered on the true primitive coordinates with covariance matrices  $\{A_{k,t}\}$ , and the prior  $p(\Theta|MI)$  is supposed to be a uniform distribution  $p(\Theta|MI) = \prod_i \frac{1}{r_i}$  on a bounded domain. The final function to minimize is a non-linear weighted least-square function:

$$J(\Theta) = \sum_{k,t} (X_{k,t} - \hat{X}_{k,t}(\Theta))^T A_{k,t}^{-1} (X_{k,t} - \hat{X}_{k,t}(\Theta))$$

where  $\hat{X}_{k,t}(\Theta)$  represents the non-linear models for the primitive  $k$  projected in the image  $t$ . To find the parameters  $\hat{\Theta}$  that minimize  $J(\Theta)$ , we use the Levenberg-Marquardt minimization algorithm, which has fast convergence with such non-linear least-square functions. Like all local minimization algorithms, it needs a good first estimate of the parameters, which can be problematic to provide.

**Frame-by-frame minimization.** To overcome the initial estimate problem, we proceed on a frame-by-frame basis. The following relation holds:

$$p(\Theta|\{X_{k,t}\}_{t=0\dots T_0+1}MI) \sim \frac{p(\{X_{k,t}\}_{t=0\dots T_0+1}|\Theta MI)}{p(\{X_{k,t}\}_{t=0\dots T_0}|\Theta MI)} p(\Theta|\{X_{k,t}\}_{t=0\dots T_0}MI)$$

Thus, if we take the negative log of the formula, the function to minimize for data in the frames 0 to  $T_0 + 1$  is the sum of the function already minimized for frames 0 to  $T_0$  and the least square error on the new data at frame  $T_0 + 1$ , so the MAP estimate  $\hat{\Theta}_{T_0}$  is an efficient first guess for  $\hat{\Theta}_{T_0+1}$ .

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. The camera motion is then recovered gradually, and the procedure for optimally adjusting the motion complexity will be naturally integrated in this sequential framework.

### 3 Selection of the Complexity

The MAP estimator gives, at every frame, the optimal parameter set for a given model. To properly model camera motion, motion curves of different complexity must be tested and the best one should be selected. Thanks to the frame-by-frame estimation, we can start using the simplest curve model, then test more complex models while adding data from new frames.

#### 3.1 Model Selection Criteria

To select the best model, we need a selection criterion that ranks the concurrent models. In statistical modeling, different kinds of criteria are used, relying on different information

theories. The global idea is to choose the simplest model  $M$  that represents the relevant data but not the noise. The criterion gives a quantitative balanced value of both the model complexity and the error on the data  $D$ .

Ad hoc criteria can be employed, usually a simple function of the number  $p$  of parameters  $\Theta$  in the model  $M$ , but such methods must be tuned and tested extensively. From considerations on the likelihood, the *Akaike Information Criterion*  $AIC = L + 2p$  has been derived, where  $L = -2 \log p(D|\Theta MI)$  is the inverse log-likelihood. *Minimum Description Length* (MDL) is another well-known criterion, based on algorithmic theory of information. The MDL criterion computes the code length necessary to describe the model and the code length necessary to describe the remaining stochastic complexity of the data [12]. The *Bayesian Information Criterion*  $BIC = L + p \log N$ , with  $N$  the number of data measurements, is used for Bayesian inference and maximizes the evidence  $p(D|MI)$ .

All these criteria are very similar, and the distinctions between them on a practical point of view is often subtle [3]. In statistical modeling, these criteria are asymptotic and may differ for geometric problems [8]. A transposition in the geometric context makes different approximations related to the models in use, that can lead to slightly different criteria [9] [11]. For parameters estimation, we used a Bayesian formalism. To maintain coherence, we choose to use the evidence maximization for model selection as well.

### 3.2 Bayesian Evidence Criterion

The geometric BIC criterion is not directly adapted here and we must compute a new approximating criterion for evidence. The outline of the computation is similar to BIC derivation from evidence in [11], but asymptotic approximations are avoided here. The evidence is  $p(D|MI) = \int_{\Theta} p(D|\Theta MI)p(\Theta|MI)d\Theta$ . Stating  $\Phi(\Theta) = -\log(p(D|\Theta MI)p(\Theta|MI))$ ,  $p(D|MI) = \int_{\Theta} \exp -\Phi(\Theta)d\Theta$ . As  $\hat{\Theta}$  is the minimum of  $\Phi(\Theta)$  and  $\nabla\nabla\Phi(\hat{\Theta}) = \hat{\Lambda}_{\hat{\Theta}}^{-1}$ , a second order Taylor approximation of the function  $\Phi(\Theta)$  at  $\hat{\Theta}$  gives:

$$\begin{aligned} p(D|MI) &\approx \int_{\Theta} \exp -\Phi(\hat{\Theta}) - \frac{1}{2}(\Theta - \hat{\Theta})^T \hat{\Lambda}_{\hat{\Theta}}(\Theta - \hat{\Theta})d\Theta \\ &= (2\pi)^{p/2} |\hat{\Lambda}_{\hat{\Theta}}|^{1/2} \exp -\Phi(\hat{\Theta}) \end{aligned}$$

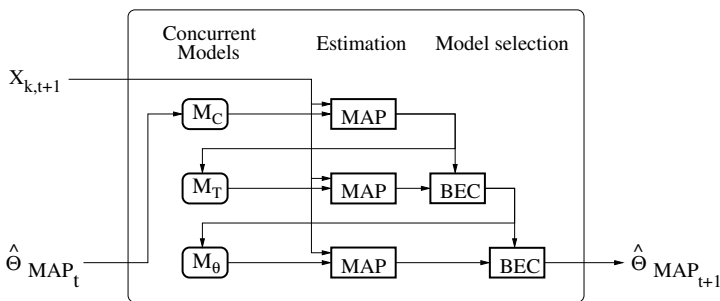
The expression of  $\Phi(\Theta)$  specific to our problem is given by  $\Phi(\Theta) = \frac{1}{2}J(\Theta) + \frac{N_d T}{2} \log(2\pi) + \frac{1}{2} \sum_{k,t} \log |A_{k,t}| + \sum_i \log r_i$ , where  $N_d T$  represents the total number of data coordinates over the  $T$  frames of the sequence, and finally:

$$\begin{aligned} BEC = -2 \log p(D|MI) &\approx J(\hat{\Theta}) + (N_d T - p) \log(2\pi) + \sum_{k,t} \log |A_{k,t}| \\ &\quad + \log |\hat{\Lambda}_{\hat{\Theta}}^{-1}| + 2 \sum_i \log r_i \end{aligned}$$

The quantities in the criterion are all set a priori or evaluated in the MAP estimation of parameters, thus no further approximation is needed.

### 3.3 Complete Procedure

To integrate the model selection step in the sequential estimation procedure, we have to estimate all possible models, then compare them with the criterion. If the number of possible models is important, the computational effort becomes rapidly untractable. To avoid such problem, the models must be *nested*, in the sense that models of lower complexity must be contained in models of higher complexity. Therefore, more complex models are initialized from simpler models. At each frame, only the currently selected model is memorized and more complex models can be computed from it. The motion complexity is supposed to stay unchanged or increase from one frame to the next, independently for translation and rotation parameters. Three different estimations are then necessary to select the optimal model (see Fig.3).



**Fig. 3.** The current model  $M_C$  and more complex models for translation  $M_T$  and rotation  $M_\theta$  are estimated and compared at each frame.

## 4 Motion Curves

The sequential estimation and selection algorithm proposed here impose constraints on the motion model we can consider: the motion curves must be nested, and more or less complex curves should share parameters close together. Three families of curves are proposed: a polynomial of high degree representing the complete motion, linear splines that model the motion more locally, and a hybrid model made of piecewise polynomials of various possible degrees.

### 4.1 Chebyshev Polynomials

In the first model, the parameters  $\mathbf{T}(t), \boldsymbol{\theta}(t)$  are functions of a single polynomial of degree  $K, L$ :

$$\mathbf{T}(t) = \sum_{n=0}^N \mathbf{a}_n P_n(t)$$

$$\boldsymbol{\theta}(t) = \sum_{m=0}^M \mathbf{b}_m P_m(t)$$

where  $P_n(t)$  is the Chebyshev polynomial of degree  $n$ . Therefore, the motion parameters become  $\{\mathbf{a}_n, \mathbf{b}_m\}$ , reducing the degrees of freedom from  $6T$  to  $3N+3M+6$  parameters.

Chebyshev polynomials are preferred for several reasons. First, they are known to be close to the minimax approximation polynomial for general functions. Next, Chebyshev coefficients  $a_n$  smoothly decrease to zero when the degree  $n$  increases. The polynomials values range from -1 to +1, so the coefficients are homogeneous to the approximated values. Finally, the polynomial curve changes gradually when high order coefficients are set to zero.

The complexity of the motion is represented by the degree of polynomials, and the concurrent motion models to choose from in the model selection phase are simply polynomials of higher or lower degree. The models are nested and the polynomial coefficients common to simple and complex models are close together, the coefficients of simple models directly giving a good first estimate for models of higher complexity.

### 4.2 Linear Splines

Camera motions can be viewed as a sequence of simple moves, and it is reasonable to hope that changes on the motion estimation on one of the last frames should not induce changes on the motion corresponding to the first frames. For such motion, piecewise polynomials can be preferred to a global high degree polynomial. As the simplest piecewise models, we use the linear spline:

$$\begin{aligned} \mathbf{T}(t) &= \mathbf{a}_n(1 - \tau_n) + \mathbf{a}_{n+1}\tau_n & \text{if } t \in [t_n, t_{n+1}] \\ \boldsymbol{\theta}(t) &= \mathbf{b}_m(1 - \tau_m) + \mathbf{b}_{m+1}\tau_m & \text{if } t \in [t_m, t_{m+1}] \end{aligned}$$

where  $\tau_n = \frac{t-t_n}{t_{n+1}-t_n}$ . The definition is identical for  $\tau_m$ . The motion parameters are  $\{\mathbf{a}_n, \mathbf{b}_m\}$  along with the two series of time intervals  $\{[t_n, t_{n+1}]\}, \{[t_m, t_{m+1}]\}$ . The selection problem is not to find the proper degrees, that are fixed, but to determine how many pieces to make and where to place them on the time interval of the sequence.

In the model selection procedure, concurrent models will be models with more or less pieces. More complex curves are obtained by adding a new interval when adding a new frame, whereas the intervals for simpler models remain unchanged. This model family is nested, like Chebyshev polynomials: models of higher complexity just have additional polynomial pieces. Intervals with the same pieces have the same coefficients, regardless of the rest of the curve and polynomials of lower complexity are a good first estimate for polynomials of higher complexity.

### 4.3 Piecewise Chebyshev Polynomials

Finally, a hybrid model between Chebyshev and piecewise curves has been tested. This model is a disjoint succession of Chebyshev polynomials:

$$\begin{aligned} \mathbf{T}(t) &= \sum_{n=0}^{N_i} \mathbf{a}_{n,i} P_n(\tau_i) & \text{if } t \in [t_i, t_{i+1}[ , \\ \boldsymbol{\theta}(t) &= \sum_{m=0}^{M_j} \mathbf{b}_{m,j} P_m(\tau_j) & \text{if } t \in [t_j, t_{j+1}[ , \end{aligned}$$



This model has the smoothness of Chebyshev polynomials on regular parts, and can accommodate for composite motions. It is nested, and simple curves are good initializations for more complex ones. For this family of models, there are two possibilities to increase the complexity: we can add a degree to the last polynomial which describes the current motion, or we can create a new polynomial to fit the motion independently of the past frames. Therefore, this model necessitates to perform two additional estimation and selection steps.

#### 4.4 Other Curve Models

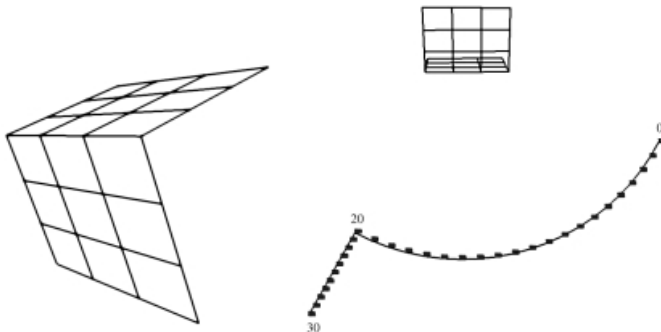
Other curve models could come to mind to describe the motion, like Bernstein polynomials or cubic splines. However, these are generally suited for interpolation purposes, and the modeling procedure proposed here is more related to approximation problems. Some experiments with such interpolation curves led to poor results, mainly due to instabilities of the curve coefficients when the complexity is modified.

## 5 Experiments

The technique proposed in this paper has been tested on both synthetic and real video sequences, in order to characterize its performances. Cramer-Rao lower bounds for accuracy have been computed on the simulated experiments. Finally, classical estimation without motion model have also been performed.

### 5.1 Simulated Data

To perform the synthetic experiments, a 3D scene was created. The scene consists in two sides of a Rubik's cube, and the camera orbits the cube for 20 frames, then goes backward along a straight line for 10 frames (see Fig.4). The scene description is composed of 2



**Fig. 4.** Simulation settings: left: the scene observed at  $t = 0$ ; right: the camera path.

rectangles, 8 segments as well as geometric constraints. The tracking part is completely simulated by exact projection of the true 3D points and addition of Gaussian noise of increasing magnitude.

Experiments have been performed with a Gaussian noise of zero mean and standard deviation of 0, 1, 2, 3, 5, 7 and 10 pixels. For each configuration and each noise level, three instances of synthetic data were generated for estimation and the results were averaged. Errors in 3D shape, translation and rotation are computed with respect to the true values, after an appropriate transformation that makes the scenes suitable for comparison. This transformation places the origin of space at the center of the 3D scene, rotates the coordinate system to match the scene eigenvectors and normalizes the standard deviation of scene points from the origin. The final error curves have been obtained by averaging the results with the three noisy sets. The complexity of the optimal curves are defined as the total number of 3D parameter vectors needed to describe them. Theoretical Cramer-Rao lower bounds (CRLB) have also been computed from the original scene.

The error curves in Fig.5 show that the results are similar with the different models and without model. The optimization procedure has always properly converged, as the 2D residual errors are very close to their theoretical bounds. Reconstruction errors are also close to the Cramer-Rao bounds, especially for the motion.

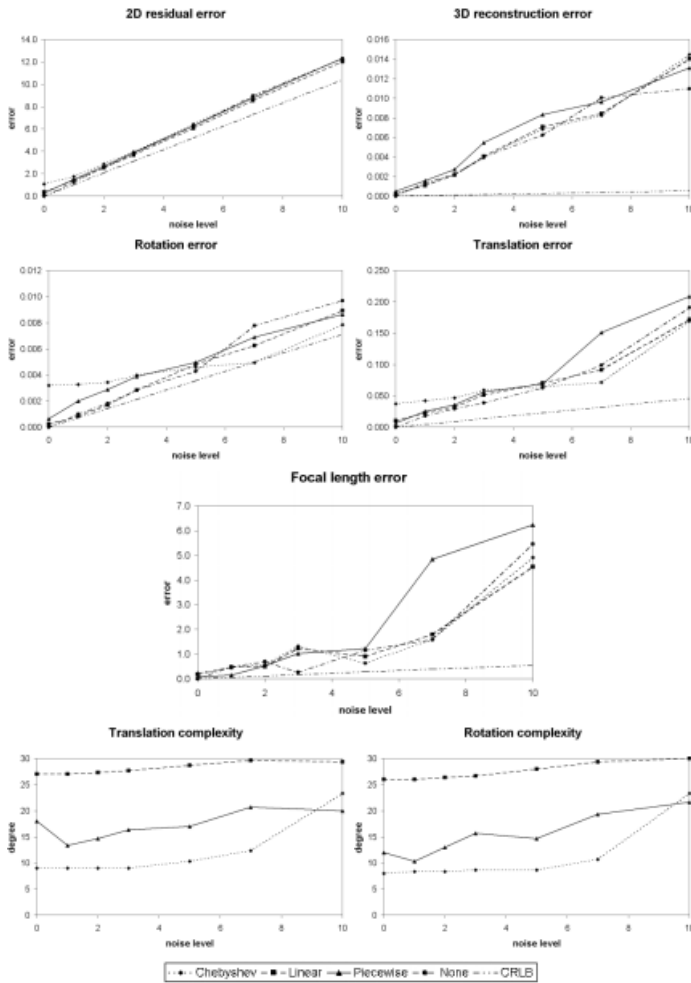
The big difference here lie in the complexity of the motion models needed to achieve such accuracy. The complexity of Chebyshev polynomials is the lowest, and remains extremely stable as noise increase. The linear spline model is not very efficient, as its complexity remains very high, but is also stable. Piecewise polynomials are in-between, with larger complexity fluctuations: in practice, they often hesitate between a long Chebyshev curve of high degree and numerous curves of degree 1 or 2. This stability is remarkable, as it shows that the estimated complexity is really related to the complexity of the motion itself and not to the noise. The corresponding motion curves are not an approximation, as they manage to reach the same accuracy levels as model-free estimation.

## 5.2 Real Sequences

The capabilities on real video sequences of the algorithm has also been tested. The following videos, borrowed from various laboratories, have been used:

- the "Sport" sequence (from Inria Syntim): the camera motion is a slow translation in straight line,
- the "House" sequence (from Inria Movi): the camera undergoes a fast circular motion around a model house,
- the "StainX" sequence (from Alias|Wavefront): the camera is moving towards the scene in a translation/rotation motion,
- the "Begijnhof" sequence (from K. U. Leuven Visics): the camera is hand-held and moving backward almost linearly, but not smoothly.

For every sequence, 10 to 20 features (points, lines and rectangles) have been tracked, some of them only on a limited part of the sequence (see the features on the first images on Tab.1). The structure of the scene and the camera motion have been computed with the three motion models, and with the model-free algorithm. Reconstruction results are


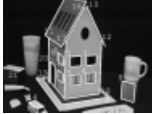




**Fig. 5.** Error and complexity results in the synthetic experiments, for increasing noise levels.

more or less subjective for real sequences, and error curves cannot be drawn. We can still compare the visual 3D reconstruction obtained with the different motion curves (see Fig.7), but the only reliable metric is the 2D residual error. Its mean and standard deviation values are given on Tab.1, along with the complexities for the three models. The visual correctness of the reconstruction can be assessed by mean of augmented reality (addition of synthetic elements), as presented on Fig.6.

In all cases, the 3D structure reconstructions are visually correct and seem identical. The camera motion also correspond to the qualitative descriptions given above. The residual errors are low, and both mean and standard deviation values are very close

**Table 1.** Residuals and complexity on the real sequences.

				
	Sport	House	StainX	Begijnhof
Number of frames:	90	20	70	42
2D residual error				
no model:	3.438	3.394	4.357	2.783
Chebyshev:	3.470	3.412	4.579	2.778
linear:	3.490	3.547	4.392	2.732
piecewise:	3.464	3.546	4.371	2.643
2D standard deviation				
no model:	2.972	3.367	2.461	2.159
Chebyshev:	3.095	3.429	2.570	2.202
linear:	3.376	3.469	2.527	2.198
piecewise:	3.056	3.478	2.498	2.219
translation complexity				
Chebyshev:	6	5	9	6
linear:	4	13	4	11
piecewise:	7	16	10	27
rotation complexity				
Chebyshev:	5	4	9	24
linear:	3	12	9	21
piecewise:	7	17	12	27

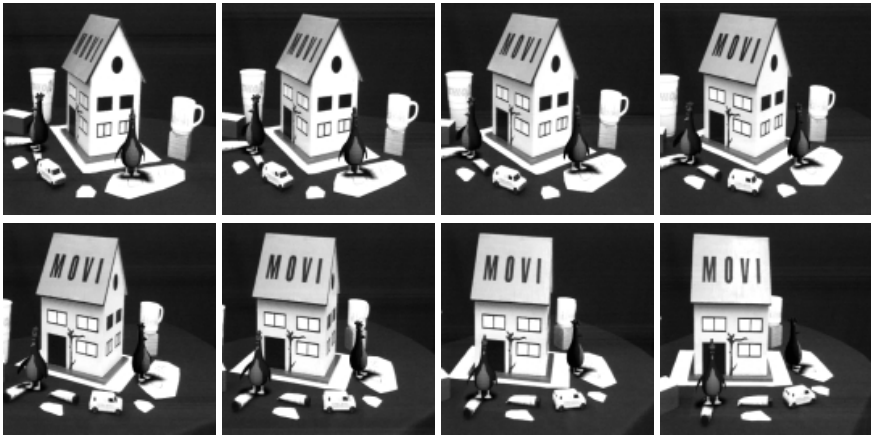
for any given sequence: in terms of the reprojection error, the results are all identically optimal.

The first important difference is the reduction of the motion complexity when we use motion curves. Even with the longest sequences, the complexity remains very low, and the accuracy is not jeopardized by the reduction. Moreover, the motion curves provide a better robustness: in Fig.7, the reconstructed path for the camera without model is strongly perturbed by noise, whereas the visual motion in the video sequence is perfectly smooth. These perturbations are removed with any of the motion models.

As in the synthetic experiments, Chebyshev polynomials have proven to give a very compact motion model without loss of accuracy. Their approximation properties allow very stable computations, and changes in the trajectory are correctly handled.

Linear splines perform better here. On almost linear motions, they go below Chebyshev polynomials (cf. the Sport sequence), but become limited for curved motions like the orbit motion of the simulations or the House sequence.

Piecewise Chebyshev polynomials still suffer from their flexibility: they perform well, but always with an increased complexity. It is hard to discriminate between the two possible ways to grow in complexity here, and the final models will involve long pieces of Chebyshev curves and small line segments or even points on the less regular parts.



**Fig. 6.** Augmented reality with virtual penguins on the House sequence: images 1, 5, 9, 13 (top), 17, 21, 25, 30 (bottom).

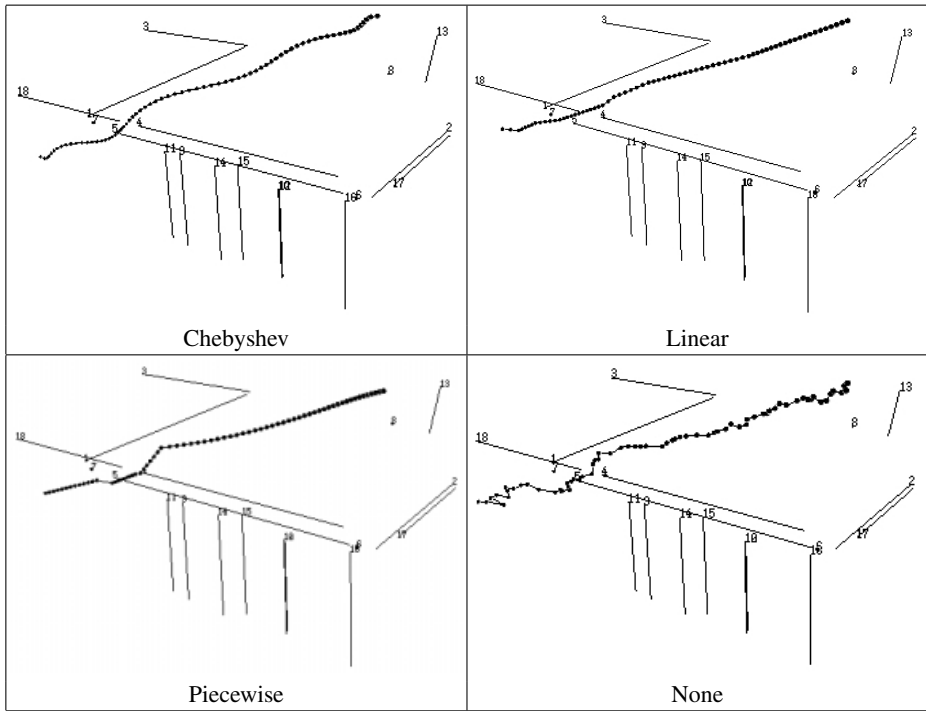
Finally, the smoothness assumption shared by the three models is not a strong constraint. In the Begijnhof sequence, the camera motion is not smooth, as the camera oscillates when the cameraman moves. The related motion models capture this phenomenon with an increased complexity of the orientation curve, and the results are comparable in 2D and in 3D with an unconstrained structure and motion recovery.

## 6 Conclusions

We presented here an estimation framework for shape and motion recovery from video sequences. The technique takes into account the continuity and smoothness of the camera motion, through motion curves. The curves are optimally fitted to more or less complex motions thanks to a sequential estimation and selection procedure.

As the motion curves drastically reduce the number of parameters to estimate, the computations are greatly simplified and the estimation is more robust to tracking noise. The accuracy of reconstruction is not decreased by the approximation introduced by the curves. The two major roles of the introduced curves are to reduce the complexity of the problem through a limited and relevant set of parameters and to factor out the tracking noise from the recovery by imposing some regularity.

The polynomial functions proposed as curve models can describe any kind of motion. Chebyshev polynomials are near optimal in the sense of function approximation, and always perform well in practice for this problem. Their only weakness is for describing sharp changes in the motion, generally related to a change in the nature of the motion itself. We plan to investigate more subtle models to get over this limitation. Studying descriptions of different kinds of camera motions could also help to refine even more the models, particularly for erratic and trembling motions like hand-held camera motions.



**Fig. 7.** Reconstructed scene from the StainX sequence, with the proposed motion models. The spheres indicate the camera position at each frame.

## References

1. P. L. Bazin. A parametric scene reduction algorithm from geometric relations. In *Proc. Vision Geometry IX*, SPIE's 45th annual meeting, San Diego, 2000.
2. P. L. Bazin and J. M. Vézien. Tracking geometric primitives in video streams. In *Proc. Irish Machine Vision and Image Processing Conference*, Belfast, 2000.
3. Kishore Bubna and Charles V. Stewart. Model selection and surface merging in reconstruction algorithms. In *Proc. International Conference on Computer Vision*, Bombay, 1998.
4. Alessandro Chiuso, Paolo Favaro, Hailin Jin, and Stefano Soatto. 3-d motion and structure from 2-d motion causally integrated over time: Implementation. In *Proc. European Conference on Computer Vision*, Dublin, 2000.
5. A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian data analysis*. Chapman & Hall, 1995.
6. Xiaoping Hu and Narendra Ahuja. Motion and structure estimation using long sequence motion models. *Image and Vision Computing*, 11(9):549–569, november 1993.
7. Frederik Kahl and Anders Heyden. Auto-calibration and euclidean reconstruction from continuous motion. In *Proc. International Conference on Computer Vision*, Vancouver, 2001.
8. K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier Science, Amsterdam, 1996.

9. Kenichi Kanatani and Chikara Matsunaga. Geometric mdl and its media applications. In *Proc. Workshop on Information-Based Induction Sciences*, Izu, 2000.
10. S. J. Maybank and P. Sturm. Minimum description length and the inference of scene structure from images. In *Proc. IEE Colloquium on Applications of Statistics to Pattern Recognition*, 1999.
11. P. H. S. Torr. *Data segmentation and model selection for computer vision*, chapter Model selection for structure and motion recovery from multiple images. Springer, 2000.
12. P. Vitányi and M. Li. Ideal mdl and its relation to bayesianism. In *Proc. Information, Statistics and Induction in Science Conference*, Melbourne, 1996.