MIT Media Laboratory, Perceptual Computing Technical Report #243
To appear IEEE PAMI; manuscript submitted July 1994

Recursive Estimation of Motion, Structure, and Focal Length

A. Azarbayejani and A. Pentland

Abstract— **We present a formulation for recursive recovery of motion, pointwise structure, and focal length from feature correspondences tracked through an image sequence. In addition to adding focal length to the state vector, several representational improvements are made over earlier structure from motion formulations, yielding a stable and accurate estimation framework which applies uniformly to both true perspective and orthographic projection. Results on synthetic and real imagery illustrate the performance of the estimator.**

Keywords— **Structure from motion, camera model, camera calibration, recursive estimation, 3-D representation, 3-D modeling.**

I. Introduction

Recovering scene structure and camera motion from image sequences of rigid motion has been an important topic in computer vision and has been approached in many ways. Researchers have tended to treat either the orthographic or true perspective case and have used either image flow or feature correspondences as measurement input. Starting from traditional mathematical formulations of the problem, much existing work has focused on developing reliable computational frameworks. These have generally either required prior knowledge of the camera, or do not attempt to recover metric geometry. In this paper, we shall refer to this entire class of motion-based estimation research as the *structure from motion* (SfM) problem. It should be clarified, however, that SfM usually means recovery of motion *and* structure, and, in our case, focal length as well.

The work described here returns to the fundamentals of the SfM problem. We demonstrate how reconsideration of traditional data representations leads to significant improvements in stability and accuracy performance. Although we use the extended Kalman filter (EKF) as the computational framework and point feature correspondences as measurements, most of the issues we discuss here concern parameterization of the basic geometrical concepts and are therefore equally pertinent to solving the problem using other computational methods and other types of measurements.

In addition to these representational issues, which we have shown to yield considerably improved results [4], two significant new capabilities have arisen from this work.

First, we have found that focal length estimation, simultaneous with structure and motion estimation, is not only theoretically possible, but can be achieved reliably and to high accuracy. The important practical implication is the ability to process motion sequences without prior knowledge of the camera.

Secondly, as a part of our examination of data representation, we have discovered that an alternate mathematical representation of the perspective camera model has several functional and numerical advantages, including that it allows the orthographic projection case to be treated as simply a special case of perspective projection (with finite, estimable parameters).

The authors are with the Media Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139.

A. Previous Work

Our work is closely related to a large body of previous research in structure and motion estimation. We discuss here some of the most relevant recent research.

All feature-based motion estimation can be related to the classic *relative orientation* problem as analyzed by Horn in [14;15]. It is a basic two-frame structure and motion recovery based on perspective projection and is usually solved my minimizing a nonlinear objective function. Ultimately, our work can be viewed as a relative orientation extended to sequences, done recursively, and including focal length estimation.

The computational framework we use for recursive estimation is the EKF, which has been the subject of much recent work on image sequences [1;6;5;8;11;13;20;32], including the seminal work by Broida, Chandrashekhar, and Chellappa on structure and motion estimation [5]. As far as the basic computational procedure, our filter can be viewed as an extension of this work, including focal length estimation. However, the representational differences, described in the next section, especially the treatment of translation, structure, and scale, we believe contribute to improved stability and accuracy.

Partly in response to perceived unreliability of earlier EKF-based estimators, some authors identify the linearization at each time step in the EKF as a crucial part of performance problems and have proposed alternate optimal estimation techniques based on minimization of nonlinear objective functions on batches of images using the Levenberg-Marquardt procedure[31;18;26]. These are reported to have better convergence properties than the EKF. Spetsakis and Aloimonos [25] contributed yet another treatment, providing physically-based insight to solving the nonlinear constrained minimization. And Tomasi and Kanade [28] have demonstrated the effectiveness of a *factorization method* based on the singular value decomposition for orthographic sequences. More recently, an extension [22] to paraperspective projection has also been developed which is based on a better linear approximation to true perspective projection. Also, Szeliski and Kang [27] have demonstrated structure and motion recovery with a calibrated camera and focal length recovery from known structure in a nonlinear batch process that can handle partial point tracks well and also can accept line segment measurements.

At the expense of more complex computational procedures and requiring batches of frames as input, these have been reported to be more stable and to converge faster. This tradeoff will always exist to some extent between batch and recursive methods; the proper choice of computational framework is application dependent. Our results demonstrate that an improved representation makes the recursive approach much more competitive in stability with sophisticated batch computation. An interesting avenue of future research would be to ascertain the relative effects of parameterization versus computational method on the stability and accuracy of estimation.

Back in the domain of recursive estimation, there have also been many new results based on the EKF since [5]. Most of these utilize some external assumptions about motion or structure or are based on stereo measurements. Most closely related to the general rigid motion problem that we are concerned with are works by Oliensis and Thomas [21] and Soatto et al [24]. Both use a two-stage approach where each pair of frames is first analyzed using a classic two-frame estimate such as [15] or [19] and the result is filtered by an EKF. The argument is that since the measurement vector used in these EKFs are the same as the state vector, the EKF is then linear, completely observable, and should not suffer any effects of linearization that may have plagued original EKF formulations. The price that is paid is primarily complexity, including the requirement of having an external mechanism for maintaining scale. In this type of implementation, the EKF serves a different function than ours; it is essentially a smoothing filter for an external state estimator, whereas our filter, in the spirit of [5], actually performs the computation that inverts the (linearized) nonlinear measurement equation.

Above all, however, what separates our estimator from all these previous methods is that, since it is formulated to recover focal length, it does not require a calibrated camera to recover metric geometry. Furthermore, it is not specific to only perspective or orthographic projection. The implication is that camera output can be processed directly with no calibration procedure on the camera and that existing video, from an unknown camera, can also be processed directly.

Recent works on the recovery of non-metric structure invariants [16;10;23] share the same functional advantage of not requiring a calibrated camera and have aroused a great deal of interest as a result. These methods can do with even fewer assumptions about the camera, including not knowing principal point or image coordinate scaling, with the result that the representations computed are correspondingly weaker. The tradeoff, it can be argued [10;23], is that the weaker representations can presumably be computed with greater stability and are sufficient descriptions for many computer vision applications. When metric structure is desired, however, our estimator appears to be unique in the published literature as to its capabilities.

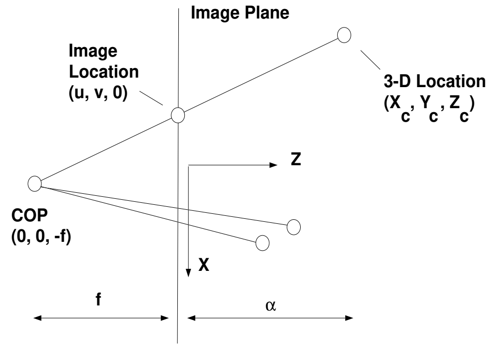


Fig. 1. Model of central projection; the coordinate center is placed at the image plane rather than at the center of projection.

We demonstrate our estimator on both real and synthetic sequences in Sections III and IV. The data for some of these experiments are available via anonymous ftp from `whitechapel.media.mit.edu`.

II. Representational Issues

In this section, we develop mathematical descriptions for the geometric data related to scene structure, camera motion, and camera geometry and discuss their importance to the estimation process. We also briefly present issues related to recursive estimation and our EKF implementation.

A. Camera Model

Equation 1 is a model for central projection, where (X_C, Y_C, Z_C) is the 3-D location of a point in the camera reference frame, (u, v) is the image location, and $\beta = 1/f$ is the inverse focal length:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} X_C \\ Y_C \end{pmatrix} \frac{1}{1 + Z_C\beta}. \tag{1}$$

Geometrically, this model is identical to the usual model,

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} X_C \\ Y_C \end{pmatrix} \frac{f}{Z_C} \tag{2}$$

with two representational changes. First, as illustrated in Figure 1, the coordinate system origin is fixed at the image plane rather than the center of projection (COP). Second, inverse focal length β, rather than focal length f, is the model parameter. Similar camera models have long been used for similar reasons in photogrammetry and have recently been used also by Szeliski and Kang [26;27] in a nonlinear least squares formulation.

One important property of this model is that it decouples the representation of the camera from the representation of depth in a desirable way. With the usual model (Equation 2), if the "depth" Z_C of an object is fixed and the "focal length" parameter f is altered, *the imaging geometry* *remains the same and only the scale of the image changes.* (The cone of perspective rays remains fixed while the focal plane translates along the optical axis.) Conversely, note that there is no way to alter the imaging geometry (angles of rays) without also altering the representation of depth (Z_C). Thus the parameter Z_C really encodes both the camera imaging geometry *and* depth, while f is really only a scale factor from world units to pixels.

On the other hand, altering the inverse focal length (β) in Equation 1 alters the imaging geometry independent of the representation of object depth. This decoupling becomes important when focal length is being estimated in addition to structure.

More importantly, this model also has the property that it does not become numerically "ill-conditioned" as focal length becomes large. In fact, orthographic projection, which is only a theoretical mathematical limit with Equation 2, is represented perfectly well with finite parameters ($\beta = 0$) using Equation 1. Thus, all theoretical results that hold for general values of β apply uniformly to both perspective and orthographic projection, two cases which are traditionally treated quite differently in computer vision.

This camera geometry and associated mathematical model give us a parameterization

$$(camera) = (\beta)$$

which handles both orthographic and perspective projections. We demonstrate both cases in our experiments of Sections III and IV.

B. Structure Model

A "feature point" is defined in terms of its image location in the first frame in which it appears. Feature tracking (of some kind) results in measurements of image location of the feature in subsequent frames. It is assumed that feature tracking is noisy with zero-mean errors.

The 3-D location of the point can be represented using Equation 3 below. The first term represents the image location and the second term represents the perspective ray scaled by the unknown depth α:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} u \\ v \\ 0 \end{pmatrix} + \alpha \begin{pmatrix} u\beta \\ v\beta \\ 1 \end{pmatrix} \qquad (3)$$

Pointwise structure, therefore, can be represented with one parameter per point.

This representation is consistent with early analyses such as [14], but inconsistent with representations used in much of the image sequence work, including [5;21;30], which use three parameters per point. It is critical for estimation stability, however, either to use this basic parameterization or to understand and properly handle the additional parameters. Here we describe the computational implications of our parameterization and in Section II.G we show how it relates to alternate parameterizations.

Before stability, first consider the basic requirements for a batch solution of F frames and N points. There are $6(F-1)$ motion parameters and $3N$ structure parameters. Point measurements contribute $2NF$ constraints and one arbitrary scale constraint must be applied. Hence, the problem can be solved uniquely when $2NF + 1 > 6(F-1) + 3N$. Thus all motion and structure parameters can in principle be recovered from any batch of images for which $F \geq 2$ *and* $N \geq 6$.

However, in a recursive solution, not all constraints are applied concurrently. At each step of computation, one frame of measurements constrains all of the structure parameters and one set of motion parameters, i.e. $2N$ measurements constrain $6 + 3N$ degrees of freedom at each frame. This is always an underdetermined computation having the undesirable property that the more features that are added, the more underdetermined it is. Unless one already has low prior uncertainty on the structure (effectively reducing the dimensionality of unknowns), one should expect unstable and unpredictable estimation behavior from such a formulation. Indeed, in [5], it was proposed that such filters only be used for "tracking" after a batch procedure is applied for initialization.

On the other hand, in our formulation, constraints $(1 + 2N)$ outnumber degrees of freedom $(6+1+N)$ for motion, camera, and structure at every frame when $N > 7$. The more measurements available the larger the gap. Our experiments verify that the overdeterminancy results in better stability, allowing for good convergence and tracking in most cases without the requirement of good prior information. In both types of formulation, once structure (and camera, in our case) has converged, each step is effectively overconstrained; the only issue is stability when structure (and camera) is grossly uncertain.

It is clear, then, that excess parameters are undesirable for stability, but how can both $3N$- and N-parameter representations describe the same structure? Section II.G relates the two and demonstrates that when measurement biases are exactly zero (or known) the $3N$ space really only has N degrees of freedom. Even in the presence of bias, most uncertainty remains along these N DOFs, justifying the structure parameterization

$$(\text{structure}) = (\alpha_1, \cdots, \alpha_N).$$

We show experimentally in Section III that even relatively large biases do not have a strong adverse effect on accuracy using this more concise model.

C. Translation Model

The translational motion is represented as the 3-D location of the object reference frame relative to the current camera reference frame using the vector

$$\mathbf{t} = (t_X, t_Y, t_Z).$$

The t_X and t_Y components correspond to directions parallel to the image plane, while the t_Z component corresponds to the depth of the object along the optical axis. As such, the sensitivity of image plane motion to t_X and t_Y motion will be similar to each other, while the sensitivity to t_Z motion will differ, to a level dependent upon the focal length of the imaging geometry.

For typical video camera focal lengths, even with "wide angle" lenses, there is already much less sensitivity to t_Z motion than there is to (t_X, t_Y) motion. For longer focal lengths the sensitivity decreases until in the limiting orthographic case there is zero image plane sensitivity to t_Z motion.

For this reason, t_Z cannot be represented explicitly in our estimation process. Instead, the product $t_Z\beta$ is estimated. The coordinate frame transformation equation

$$\begin{pmatrix} X_C \\ Y_C \\ Z_C\beta \end{pmatrix} = \begin{pmatrix} t_X \\ t_Y \\ t_Z\beta \end{pmatrix} + \begin{pmatrix} 1 & & \\ & 1 & \\ & & \beta \end{pmatrix} \mathbf{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (4)$$

combined with Equation 1 demonstrates that only $t_Z\beta$ is actually required to generate an equation for the image plane measurements (u, v) as a function of the motion, structure, and camera parameters (rotation $\mathbf{R}$ is discussed below).

Furthermore, the sensitivity of $t_Z\beta$ does not degenerate at long focal lengths as does t_Z. For example, the sensitivities of the u image coordinate to both t_Z and $t_Z\beta$ are

$$\frac{\partial u}{\partial t_Z} = \frac{-X_C\beta}{(1+Z_C\beta)^2} \quad \text{and} \quad \frac{\partial u}{\partial(t_Z\beta)} = \frac{-X_C}{(1+Z_C\beta)^2}$$

demonstrating that $t_Z\beta$ remains observable from the measurements and is therefore estimable for long focal lengths, while t_Z is not (β approaches zero for long focal lengths).

Thus we parameterize translational motion with the vector

$$(translation) = (t_X, t_Y, t_Z\beta)\,.$$

True translation $\mathbf{t}$ can be recovered post-estimation simply by dividing out the focal parameter from $t_Z\beta$. This, of course, can only be done if β is non-zero (non-orthographic), which is desirable, because t_Z is not geometrically recoverable in the orthographic case. To see this mathematically, the error variance on t_Z will be the error variance on $t_Z\beta$ scaled by $1/\beta^2$, which gets large for narrow fields of view.

D. Rotation Model

The 3-D rotation is defined as the relative rotation between the object reference frame and the current camera reference frame. This is represented using a unit quaternion, from which the rotation matrix can be generated:

$$\mathbf{R}(\mathbf{q}) = \begin{pmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{pmatrix} \quad (5)$$

The four elements of the unit quaternion only have three degrees of freedom due to the normality constraint. Thus, all four cannot be estimated independently; only a nonlinear constrained minimization will work to recover the quaternion directly. Since the EKF utilizes a linearization at each step, the nonlinear normality constraint cannot be rigidly enforced within the EKF computational structure.

However, a 3-parameter incremental rotation representation, similar to that used in [6], can be used in the EKF to estimate interframe rotation at each frame. Incremental Euler angles centered about zero (or discrete-time "rotational velocity") do not overparameterize rotation and are approximately independent and therefore can be used reliably in a system linearization.

The incremental rotation quaternion is a function of these three parameters:

$$\delta\mathbf{q} = \left(\sqrt{1-\epsilon}, \omega_X/2, \omega_Y/2, \omega_Z/2\right) \quad (6)$$

$$\epsilon = (\omega_X^2 + \omega_Y^2 + \omega_Z^2)/4 \quad (7)$$

This incremental rotation can be computed at each frame and then composed with an external rotation quaternion to maintain an estimate of global rotation. The global quaternion is then used in the linearization process at the next frame.

Thus, we have,

$$(\text{interframe rotation}) = (\omega_X, \omega_Y, \omega_Z)$$

$$(\text{global rotation}) = (q_0, q_1, q_2, q_3)$$

where interframe rotation is part of the EKF state vector and global rotation is maintained and used in the linearization at each step.

E. The Issue of Scale

It is well known that the shape and motion geometry in SfM problems such as this are subject to arbitrary scaling and that this scale factor cannot be recovered. (The imaging geometry β and the rotation *are* recoverable and not subject to this scaling.) In two-frame problems with no information about true lengths in the scene, scale factor is usually set by fixing the length of the "baseline" between the two cameras. This corresponds to the magnitude of the translational motion.

It is equally acceptable to fix any other single length associated with the motion or the structure. In many previous formulations, including [5;21] some component of the translational motion is fixed at a finite value. This is not a good practice for two reasons. First, if the fixed component, e.g. the magnitude of translation is actually zero (or small), the estimation becomes numerically ill-conditioned. Second, every component of motion is generally dynamic, which means the scale changes at every frame! This is disasterous for stability and also requires some post-process to rectify the scale.

A better approach to setting the scale is to fix a static parameter. Since we are dealing with rigid objects, all of the shape parameters $\{\alpha_i\}$ are static. Thus, fixing any one of these establishes a uniform scale for all motion and structure parameters over the entire sequence. The result is a well-conditioned, stable representation. Setting scale is simple and elegant in the EKF; the initial variance on, say, α_0 is set to zero, which will fix that parameter at its initial value. All other parameters then automatically scale themselves to accommodate this constraint. This behavior can be observed in the experimental results.

F. The EKF Implementation

Using the representations discussed thus far, our composite state vector consists of $7+N$ parameters—6 for motion, 1 for camera geometry, and N for structure—where N is the number of features tracked:

$$\mathbf{x} = (t_X, t_Y, t_Z\beta, \omega_X, \omega_Y, \omega_Z, \beta, \alpha_1, \cdots, \alpha_N) \tag{8}$$

The vector $\mathbf{x}$ is the state vector used in a standard EKF implementation, where the measurement vector contains the image locations of all the tracked features in a new frame. As described in Section II.D, an additional quaternion is required for maintaining a description of the global rotation external to the EKF.

The dynamics model in the EKF can be chosen trivially as an identity transform plus noise, unless additional prior information on dynamics is available. The measurement equation is simply obtained by combining Equations 1, 4, and 3. The RHS (u, v) in Equation 3 is the defining image location of the feature in its initial frame, and the LHS (u, v) in Equation 1 is the measurement.

The EKF implementation is straightforward and standard, with the only additional computation being the quaternion maintenance. Since the EKF is considered a common tool now in computer vision and is described in detail in [5;7;12] and others, further implementation details will not be repeated here.

Computationally, the filter requires inverting a $2N$x$2N$ matrix (i.e. the size of the measurement vector) [7;12], which is not a large task for the typical number of features on a single object. Since all parameters are overdetermined with 7 or more points, N rarely needs to be more than 15 or 20 for good results, yielding filter steps which can be computed in real-time on modern workstations.

G. Biased Measurements

We turn attention here to the issue of biased measurement noise in the EKF and how it relates to representation of object structure.

We have assumed that features are identified in the first frame and that measurements are obtained by comparing new images to the previous images and that our measurements are zero-mean or very close to zero-mean. This thinking leads to the description of structure given in Section II.B, in which the single unknown depth for each feature fully describes structure. These parameters can be computed very effectively using the EKF, which assumes zero-mean measurements.

It is common to use Kalman filters even when measurements are not truly zero-mean. Good results can be obtained if the biases are small. However, if the measurements are biased a great deal, results may be inaccurate. In the case of large biases, the biases are observable in the measurements and can therefore be estimated by augmenting the state vector with additional parameters representing the biases of the measurements. In this way, the Kalman filter can in principle be used to estimate biases in all the measurements.

However, there is a tradeoff between the accuracy that might be gained by estimating bias and the stability of the filter, which is reduced when the state vector is enlarged. When the biases are large, i.e. compared to the standard deviation of the noise, they can be estimated and can contribute to increased accuracy. But if the biases are small, they cannot be accurately estimated and they do not affect accuracy much. Thus, it is only worth augmenting the state vector to account for biases when the biases are known to be significant relative to the noise variance.

In the SfM problem, augmenting the state vector to account for bias adds two additional parameters per feature. This results in a geometry representation having a total of $7+3N$ parameters. Although we do not recommend this level of state augmentation, it is interesting because it can be related to the large state vector used in [5;21] and others, where each structure point is represented using three free parameters (X,Y,Z).

If we add noise bias parameters (b_u,b_v), Equation 3 can be written

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} (1+\alpha\beta)(u+b_u) \\ (1+\alpha\beta)(v+b_v) \\ \alpha \end{pmatrix}. \tag{9}$$

This relation is invertible, i.e.

$$\begin{pmatrix} \alpha \\ b_u \\ b_v \end{pmatrix} = \begin{pmatrix} Z \\ \frac{X}{1+Z\beta} - u \\ \frac{Y}{1+Z\beta} - v \end{pmatrix} \tag{10}$$

so the two representations are analytically equivalent.

However, geometrically the (α, b_u, b_v) parameterization is more elucidating than the (X, Y, Z) parameterization because it parameterizes structure along axes physically relevant to the measurement process. Thus, it allows us to more effectively tune the filter, ultimately reducing the dimensionality of the state space quite significantly.

It is clear that, in general, uncertainty in α trivializes uncertainty in the direction of the biases. By using initial error variance on α that is high in comparison to the error variances on (b_u, b_v), the state space is essentially reduced because the system responds more stiffly in the direction of the biases, favoring instead to correct the depths. In the limit (zero-mean-error tracking) the biases can be removed completely, resulting in the strictly lower dimensional formulation that we use in this paper.

Our experimental results (see Experiment 3) demonstrate that bias is indeed a second-order effect and is justifiably ignored in most cases. By describing the structure using the more geometrically relevant (α, b_u, b_v) parameters, we have posed the problem in a way that allows us to effectively reduce the dimensionality of the unknowns under conditions of low bias by simply removing unimportant parameters from the state vector.

III. Synthetic Experiments

In this section we present four experiments showing typical performance of the filter on synthetic data under various circumstances and we present Monte Carlo results of several different motions under various fields of view and noise levels.

- Experiment 1: Increasing Noise Level
- Experiment 2: Orthographic Case
- Experiment 3: Increasing Noise Bias
- Experiment 4: Degenerate Rotational Motion
- Experiment 5: Monte Carlo Results

A. Experiment 1: Noise Level

The first experiment illustrates typical performance of the filter under "normal" circumstances. The camera is a perspective projection with focal length 1.0 (53° field of view), there is no measurement bias, and neither the structure of the object nor the camera are known at the outset. The true structure consists of 26 points on a spherical surface. Figure 2 illustrates motion tracking and recovery of structure and focal length over 100 frames for two noise levels. The accompanying table gives error statistics and convergence rates for these and three other noise levels. In the translation plot, the solid line is t_X, the dashed line is $t_Z\beta$. In the rotation plot, the solid line is q_0, the dashed line is q_2.

The initial condition on the structure is essentially that all points are in a plane parallel to the image. The initial condition on the camera is $\beta = .5$ (28° field of view), representing a typical video focal length. The initial motion is, of course, known to be zero. Similar initial conditions are used throughout the synthetic and real experiments.

(We occasionally experience depth reversal of the structure and motion when focal lengths are long and all points are initialized to a plane. To avoid depth reversals, a mild convexity bias can be introduced by starting one central point slightly closer to the camera and one peripheral point slightly farther. For consistency, our synthetic experiments share this additional initial bias, although most of them do not require it. For our real imagery experiments, this bias was unnecessary.)

Zero-mean uniformly-distributed noise over the interval $[-n, n]$ is added to each coordinate of each measurement, where $n = 0, 2, 4, 6,$ and 12 pixels, based on an image size of (512,512). Since the object is less than one third the size of the image, these noise intervals correspond to roughly 2.5%, 5%, 7.5%, and 15% of the size of the object.

The graphs and statistics illustrate that the estimator performs stably and accurately, even at extreme levels of noise. Each graph plots state variables as the ordinate versus frame number as the abscissa. The table demonstrates quantitatively the gradual degradation in performance with increased noise. For the motion estimates, (m_t,σ_t) are the mean and RMS errors for translation and, likewise, (m_q,σ_q) are mean and RMS error-cone angular errors for the rotation (in radians).

For structure and camera estimates, a rate of convergence is computed by fitting a decaying exponential function to the absolute error. Thus, for each trial the mean structure error is closely described by $A_s \exp(-r_s t)$ and the camera parameter error by $A_c \exp(-r_c t)$, where t is the frame number. Thus, for example, at the 40th frame in Experiment 1, the mean structure error is approximately 0.68 (.34% of mean depth) for the zero-noise case and 1.07 (.54% of mean depth) for the 6-pixel noise case.

B. Experiment 2: Orthographic

The same experiment is run here, except under orthography, to demonstrate that the estimator behaves effectively in this case as well. Results are shown in Figure 3. Again, the dashed line on the translation plot is $t_Z\beta$; it is zero because β is zero — t_Z is the same as in Experiment 1, but cannot be recovered here because of orthography, as discussed earlier.

C. Experiment 3: Bias

In II.G it was pointed out that the difference between our structure model and earlier structure models can be equated to whether or not biased measurements are allowed. Our formulation proposes that the assumption of zero-bias is as good or better for feature-tacking systems as any other. But we would still like to see how biases would affect the estimation process when they do exist.

A noise level of 4 pixels (5% of object size) is used, with random biases added to each set of measurements. The biases are chosen from a unform distribution over the interval $[-n, n]$, where we choose n to be 0, 2, 4, 8, and 12 pixels. Results are shown in Figure 4.

It is clear that the estimation remains effective even in the presence of large biases and noise, justifying our theoretical claims that the excess parameters are much less important and can usually be ignored.

D. Experiment 4: Rotation about the COP ("Pure Rotation")

In a general perspective projection, all optical rays pass through the center of projection (COP). Since traditional camera models have their origin at the COP, a rotation about the COP is often referred to as a "pure rotation". (Our camera model does *not* have its origin at the COP, so the term is somewhat of a misnomer here. Rotation about the COP will have a translation component, but geometrically the same degeneracy exists.)

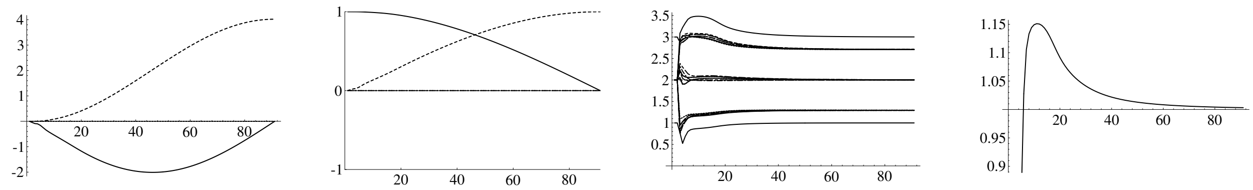
Theoretically, in a rotational motion around the COP, the structure cannot be recovered at all, yet the focal length and the motion parameters corresponding to the rotation about the COP *can* be recovered. Figure 5 confirms that our estimator in fact recovers what is estimable and fails to recover the structure and the translational component that depends on structure.

E. Experiment 5: Other motions, Monte Carlo results

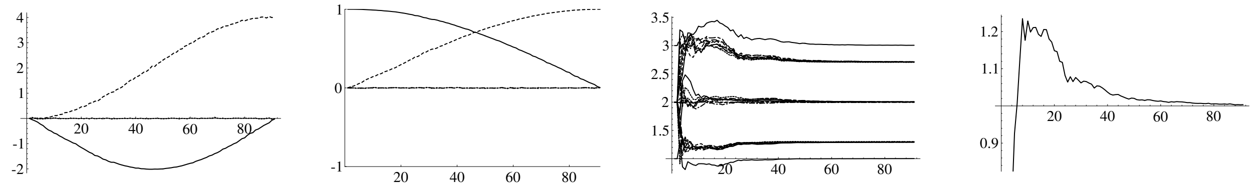
The rotational motion used in the first three experiments provides a large amount of triangulation on each feature so we should expect good estimability when the solution is formulated well. For comparison, we present here the results of Monte Carlo experiments run on over one thousand trials of several different motions using various fields of view and noise levels. For each trial, an arbitrary structure consisting of 20 points randomly chosen in a cubical area is

Experiment 1: Increasing Noise Level

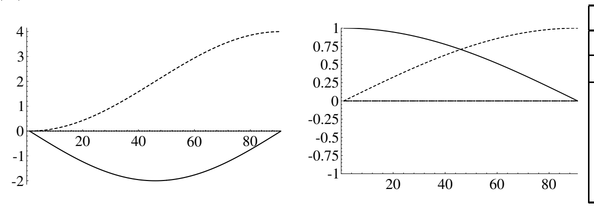
Translation (t_X, t_Y, $t_Z\beta$) Rotation (q_0, q_1, q_2, q_3) Structure (α_1, ..., α_{26}) Camera (β)



(a) Estimated parameters, no added noise.



(b) Estimated parameters, added noise uniform over ±6 pixels (7.5% of object size)



(c) Actual motion parameters.

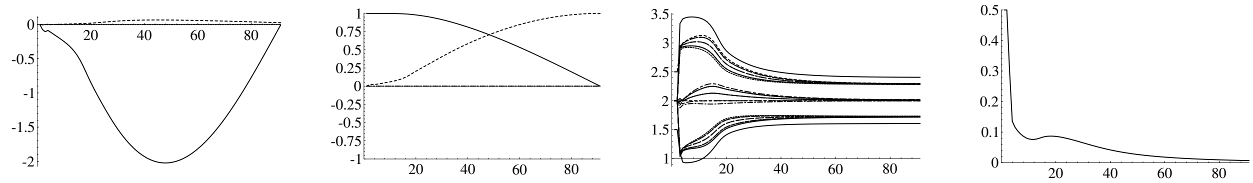
	Estimation Error				Convergence Rate	
Noise	Motion				Structure	Camera
pixels	m_t	σ_t	m_q	σ_q	$r_s(\mathrm{fr}^{-1})$	$r_c(\mathrm{fr}^{-1})$
0.	-0.0084	0.0006	0.0199	0.0007	0.1052	0.1262
2.	-0.0092	0.0006	0.0206	0.0007	0.0999	0.1042
4.	-0.0098	0.0007	0.0208	0.0008	0.1002	0.0946
6.	-0.0108	0.0008	0.0219	0.0009	0.0937	0.0850
12.	-0.0177	0.0021	0.0310	0.0017	0.0634	0.0571

(d) Statistics for several noise levels.

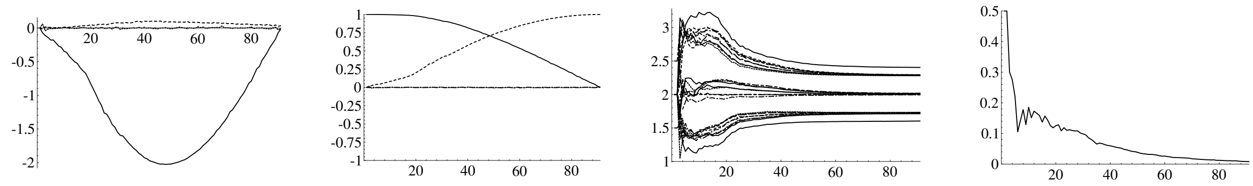
Fig. 2. Experiment 1, synthetic data with random noise added. Increasing measurement noise results in increasing estimation error on dynamic variables (motion) and slower convergence rate of static variables (structure, camera). Focal length is 1.0 (53° field of view), pixels are based on the image being (512,512). Initial mean structure error, $A_s = .4577$, initial camera error $A_c = .5$.

Experiment 2: Orthographic

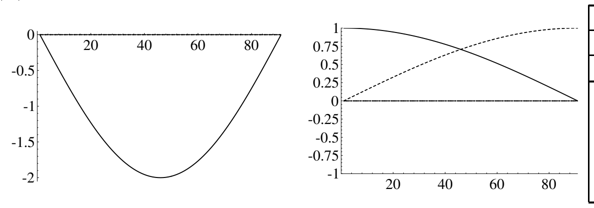
Translation (t_X, t_Y, $t_Z\beta$) Rotation (q_0, q_1, q_2, q_3) Structure (α_1, ..., α_{26}) Camera (β)



(a) Estimated parameters, no added noise.



(b) Estimated parameters, added noise uniform over ±6 pixels (7.5% of object size)



(c) Actual motion parameters.

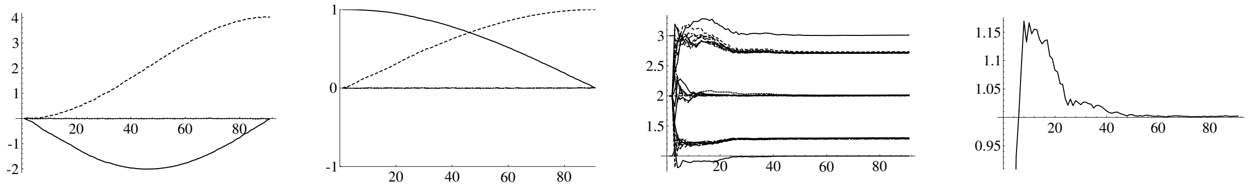
	Estimation Error				Convergence Rate	
Noise	Motion				Structure	Camera
pixels	m_t	σ_t	m_q	σ_q	$r_s(\mathrm{fr}^{-1})$	$r_c(\mathrm{fr}^{-1})$
0.	-0.0441	0.0130	0.0987	0.0174	0.0213	0.2293
2.	-0.0481	0.0136	0.1032	0.0184	0.0197	0.1399
4.	-0.0470	0.0124	0.1003	0.0172	0.0204	0.1004
6.	-0.0456	0.0113	0.0971	0.0161	0.0214	0.0902
12.	-0.0536	0.0136	0.1093	0.0197	0.0186	0.0621

(d) Statistics for several noise levels.

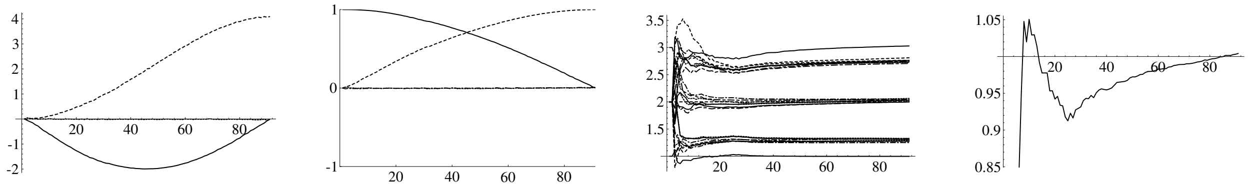
Fig. 3. Experiment 2, synthetic data with random noise added. As in the perspective projection case, increasing measurement noise results in increasing estimation error on dynamic variables (motion) and slower convergence rate of static variables (structure, camera). Focal length is ∞ (0° field of view), pixels are based on the image being (512,512). Initial mean structure error, $A_s = .1908$, initial camera error $A_c = .5$.

Experiment 3: Biased Measurements

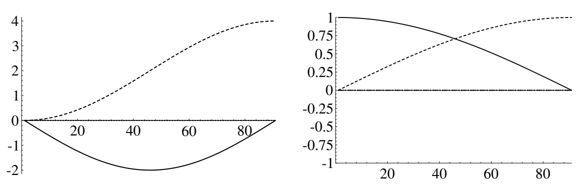
Translation (t_X, t_Y, $t_Z\beta$) Rotation (q_0, q_1, q_2, q_3) Structure (α_1, ..., α_{26}) Camera (β)



(a) Estimated parameters, biases uniform over ±2 pixels (2.5% of object size), noise level ±4 pixels (5% of object size).



(b) Estimated parameters, biases uniform over ±8 pixels (10.0% of object size), noise level ±4 pixels (5% of object size).



(c) Actual motion parameters.

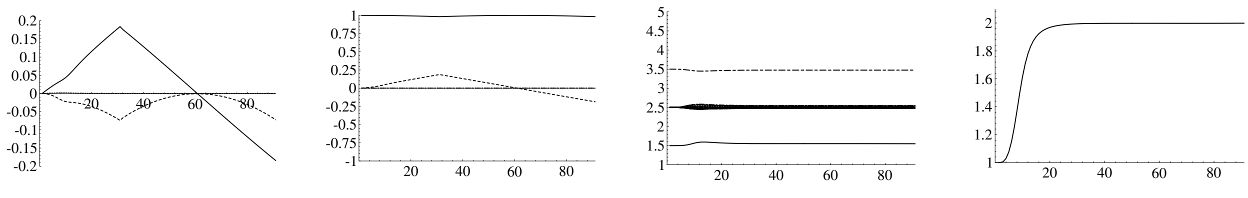
	Estimation Error				Convergence Rate	
Bias	Motion				Structure	Camera
pixels	m_t	σ_t	m_q	σ_q	$r_s(\mathrm{fr}^{-1})$	$r_c(\mathrm{fr}^{-1})$
0.	-0.0098	0.0007	0.0208	0.0008	0.1002	0.0946
2.	-0.0071	0.0004	0.0117	0.0003	0.1541	0.1433
4.	-0.0045	0.0003	0.0024	0.0001	0.1964	0.1977
8.	-0.0002	0.0006	-0.0134	0.0006	0.1505	0.2296
12.	0.0044	0.0014	-0.0262	0.0018	0.0842	0.1450

(d) Statistics for several bias levels.

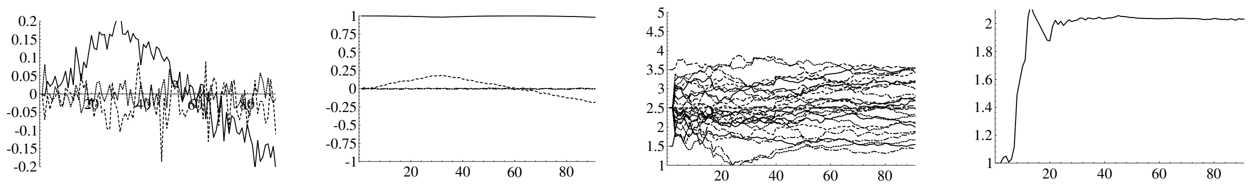
Fig. 4. Experiment 3, synthetic data with random biases and noise added. Even large biases have only a moderate effect on accuracy. Focal length is 1.0 (53° field of view), noise level ±4 pixels (5% of object size), pixels are based on the image being (512,512). Initial mean structure error, A_s = .4577, initial camera error A_c = .5.

Experiment 4: Degenerate Case – Rotation about COP

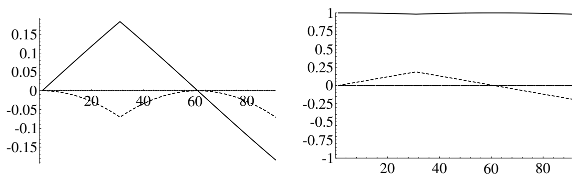
Translation (t_X, t_Y, $t_Z\beta$) Rotation (q_0, q_1, q_2, q_3) Structure (α_1, ..., α_{26}) Camera (β)



(a) Estimated Trajectory, no added noise.



(b) Estimated Trajectory, added noise uniform over ±6 pixels (7.5% of object size)



(c) Actual motion parameters.

	Estimation Error				Convergence Rate	
Noise	Motion				Structure	Camera
pixels	m_t	σ_t	m_q	σ_q	$r_s(\mathrm{fr}^{-1})$	$r_c(\mathrm{fr}^{-1})$
0.	0.0019	0.0000	0.0026	0.0000	(-0.0004)	0.1098
2.	0.0020	0.0003	0.0020	0.0001	(-0.0028)	0.1136
4.	0.0020	0.0008	0.0005	0.0001	(-0.0048)	0.1155
6.	0.0018	0.0013	-0.0006	0.0002	(-0.0072)	0.1119
12.	0.0011	0.0042	-0.0038	0.0004	(-0.0104)	0.0863

(d) Statistics for several noise levels.

Fig. 5. Experiment 4, synthetic data with random noise added. The degenerate case of rotation around the COP yields no information about structure. These data confirm that the "pure rotational" motion and the focal length *can* be recovered, while the structure and absolute translation *cannot* be recovered. The estimator remains well-conditioned due to the formulation. Focal length is .5 (90° field of view), pixels are based on the image being (512,512). Initial mean structure error, A_s = .4577, initial camera error A_c = 1.0.

used. Each trial is run over 100 frames.

Three types of motion were analyzed: rotational motion, as in the previous synthetic experiments, a translation parallel to the image plane, and a translation along the optical axis. We refer to these motions as the *Rotational*, *Parallel*, and *Axial* motions.

For each motion, field of view was varied from 0° (orthographic) to 90° holding the noise level constant at 1 pixel (in a 512 image). Next, the noise level was increased from zero to 10 pixels holding the field of view at 53°.

Figures 6–8 show the experimental Monte Carlo error results of motion, structure, and focal length estimation for each of the three types of motion. For each type of motion, error statistics are presented for various fields of view and various levels of added noise. Each bar on a graph represents mean error over 15 trials at that focal length and noise level, with a different, randomly-chosen structure used in each trial.

Structure errors are reported as percent of mean depth, where the errors are averaged over all points over all trials at each focal length and noise level. Camera errors are plotted as errors in the inverse focal length, β, but are quantified in terms of field of view error; the isolines on the graph represent constant field of view, which goes as $2\arctan\beta/2$. Translation errors are reported in terms of percent of the mean depth of the points. Rotation errors are reported as radians error from the true rotation.

Figure 6(a) shows results for fields of view ranging from 10° to 60° and Figure 6(b) shows results for noise levels up to 5 pixels, for the *Rotational* motion. Structure error is less than 1% of mean depth, translational motion errors are less than 1%, rotation errors within a .5° error cone, and field of view errors also within a .5° error cone.

For the *Parallel* motion, Figures 7(a) and 7(b) show that structure errors are somewhat larger (as expected), in the 2% – 7% range, translation errors are in in the 2% – 6% range, rotations remain in a .5° error cone, and the camera error bound increases to about 2.5°.

Finally, for the *Axial* motion, Figures 8(a) and 8(b) show that structure errors are 1% – 4%, translation is around 2%, rotation errors are around .2°, and camera error is in the 2° – 5° range.

Qualitatively we can observe from the controlled experiments that rotation estimation is relatively unaffected by the type of motion whereas structure, camera, and translation are much poorer for the parallel and axial motions. This is expected, since these motions do not provide as much difference in viewing angle to each of the points.

In response to increasing field of view, we find that rotation estimation is only slightly worse for wide angles as it is for orthographic projection. This is because in orthographic projection, rotation cannot be confused with translations; only rotations alter the relative positions of the points in the image. With wider fields of view, the potential for confusion under conditions of noise increases. Structure estimation improves with wider angles because the total change in viewpoint is greater for all motions. The camera estimate errors become numerically larger in beta, but as seen by comparing to the field of view isolines, the field of view error does not change dramatically. Since translation errors are tied to the beta parameter, they tend to go as the numerical beta error. The translational errors reflect primarily t_Z-error (along the optical axis).

In response to increasing noise, rotational estimation degrades smoothly in an expected way. Camera, structure, and translation appear to degrade more slightly with noise. Structure and translation are much more affected by the camera error, which in turn is much more affected by type of motion than by noise.

IV. Experiments on Real Imagery

In this section, we present results from applying our estimation formulation to three sequences of real imagery.

- Experiment 6: Egomotion – Models from Video
- Experiment 7: Object Tracking – Head
- Experiment 8: Object Tracking – BOX

A. Experiment 6: Egomotion, Models from Video

In this example, a texture-mapped model of a building is extracted from a 20-second video clip of a walk-around outside a building (the Media Laboratory, MIT). Figure 9(a) shows two frames of the original digitized video with feature points overlaid.

Twenty-one features on the building were tracked and used as measurement input to the EKF described in Section II. The resulting estimates of camera geometry, camera motion, and pointwise structure are shown in Figure 11. The EKF is iterated once to remove the initial transient.

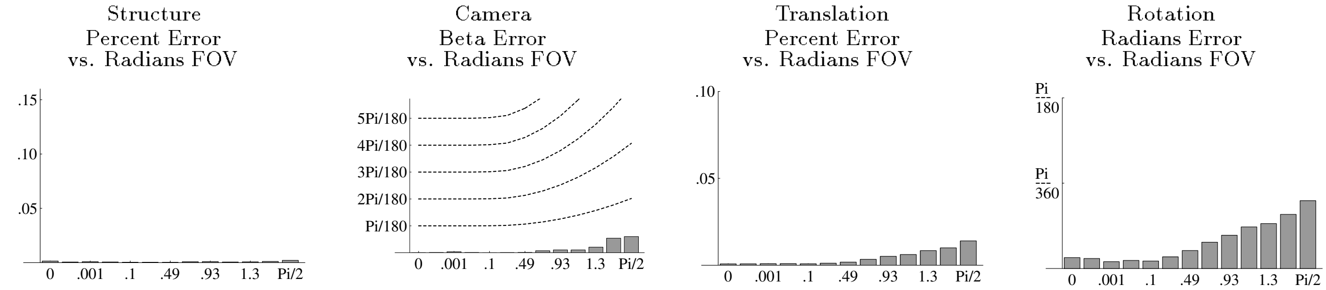
Recovered 3-D points were used to estimate the planar surfaces of the walls. The vertices were selected in an image by hand and back projected onto the planes to form 3-D polygons, depicted in wireframe in Figure 9(b). These polygons, along with the recovered motion and focal length were used to warp and combine video from 25 separate frames to synthesize texture maps for each wall using a procedure developed by Galyean [3]. In Figure 9(c,d), the texture-mapped model is rendered along the original trajectory and at some novel viewing positions.

B. Experiment 7: Object Tracking – Head

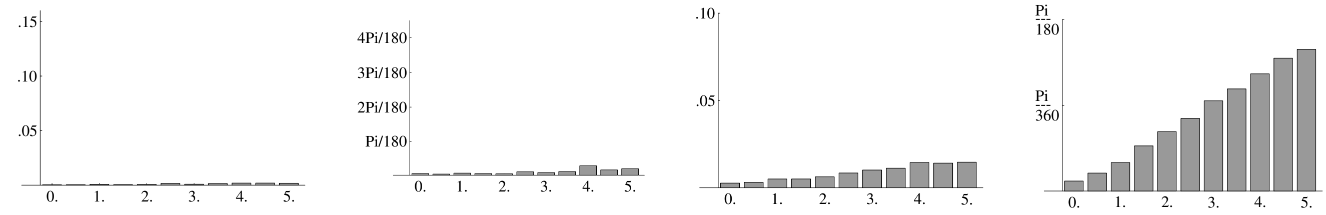
In this experiment, a person's head was tracked using both the vision algorithm and the Polhemus magnetic sensor simultaneously. Figure 10 shows the vision estimate and Polhemus measurements (after an *absolute orientation* [14] was performed to align the estimates properly). The RMS difference in translation is 0.11 units and the RMS difference in rotation is 2.35°. (The scale of translation is, of course, unknown, but is approximately 10–12cm per unit, yielding a RMS tracking error of approximately 1 cm.) This yields accuracy on the order of the observed accuracy of the Polhemus sensor, indicating that the vision estimate is at least as accurate as the Polhemus sensor.

This example is identical to the example presented in our earlier work on vision-based head tracking [2], except here we recover focal length and structure simultaneously with motion. The previous work relied on a rough, *a priori*

Experiment 5: Monte Carlo Results – Rotational Motion



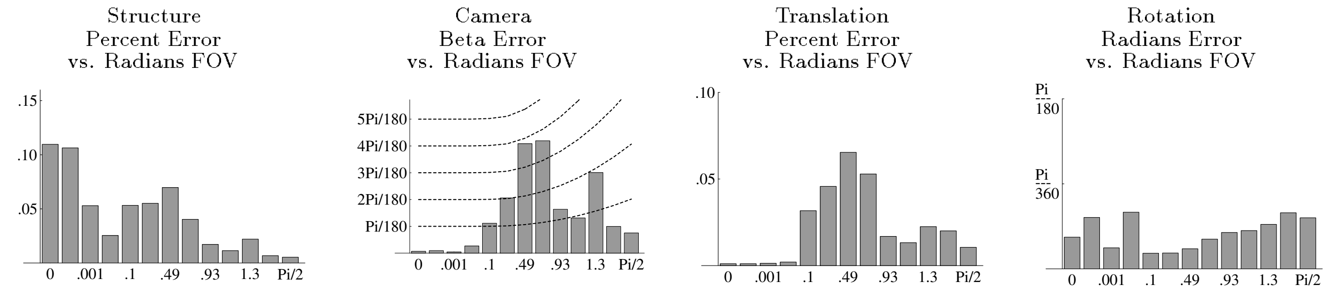
(a) Rotation, accuracy vs. field of view. The abscissa is field of view of the actual camera, ranging from 0° (orthographic) to 90°.



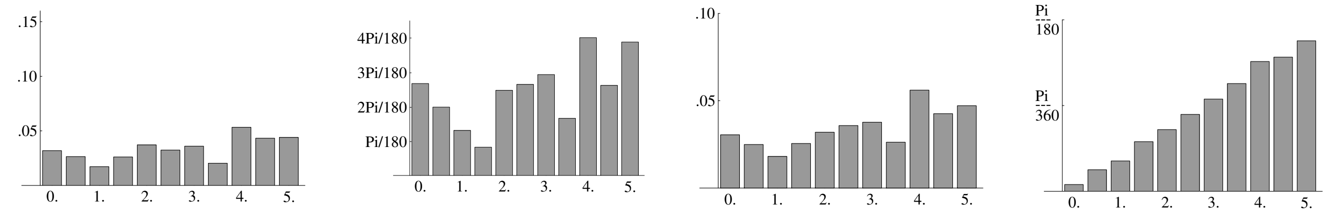
(b) Rotation, accuracy vs. noise level. The abscissa is the range in pixels of the added uniform noise, ranging from zero to 5 pixels.

Fig. 6. Experiment 5, Rotation motion. Accuracy versus field of view and noise level.

Experiment 5: Monte Carlo Results – Parallel Motion



(a) Parallel, accuracy vs. field of view. The abscissa is field of view of the actual camera, ranging from 0° (orthographic) to 90°.



(b) Parallel, accuracy vs. noise level. The abscissa is the range in pixels of the added uniform noise, ranging from zero to 5 pixels.

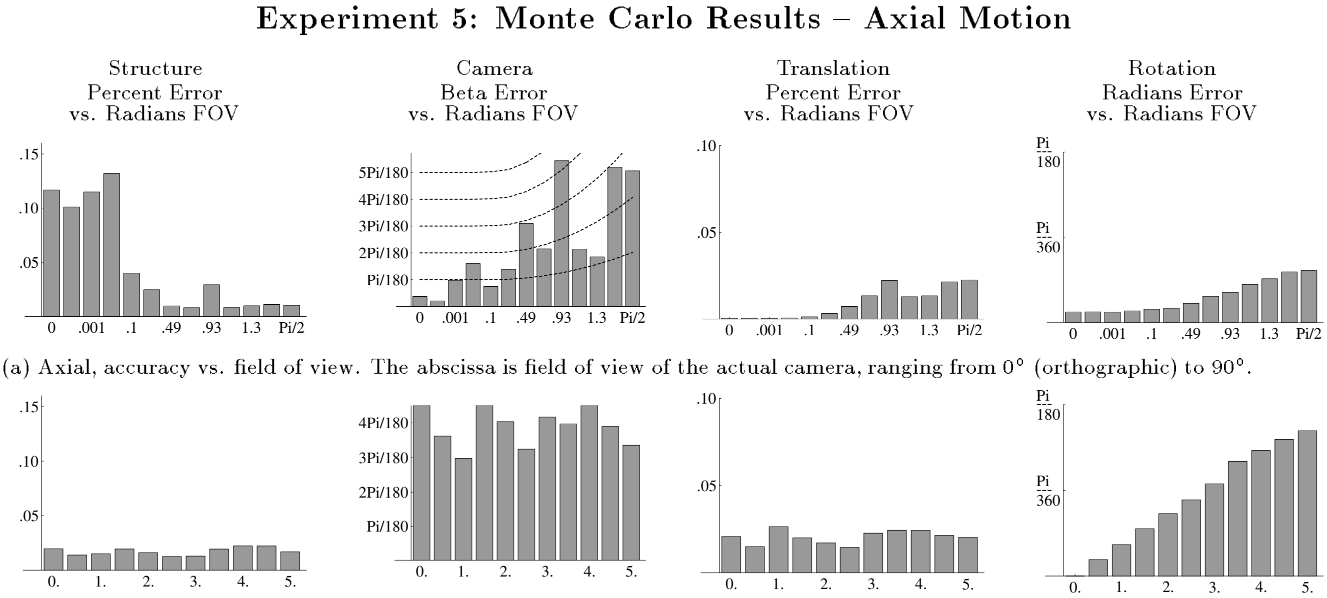
Fig. 7. Experiment 5, Parallel motion. Accuracy versus field of view and noise level.

structural model and calibration of focal length. The RMS errors between vision and Polhemus estimates for this example were slightly better than those in the previous study, (~1cm versus 1.67cm and 2.35 degrees versus 2.4 degrees).

C. Experiment 8: Object Tracking – BOX Sequence

To provide a means for comparing our estimator performance to others, we have performed a final experiment on a publicly available sequence. This is the BOX sequence from the UMass database [9], compiled by Inigo Thomas. Several authors have used this sequence as a test for geometry estimation [21;29;17], and reported rigid motion RMS errors of 0.1% and structure RMS errors of 0.2 to 0.3%. They did not estimate focal length.

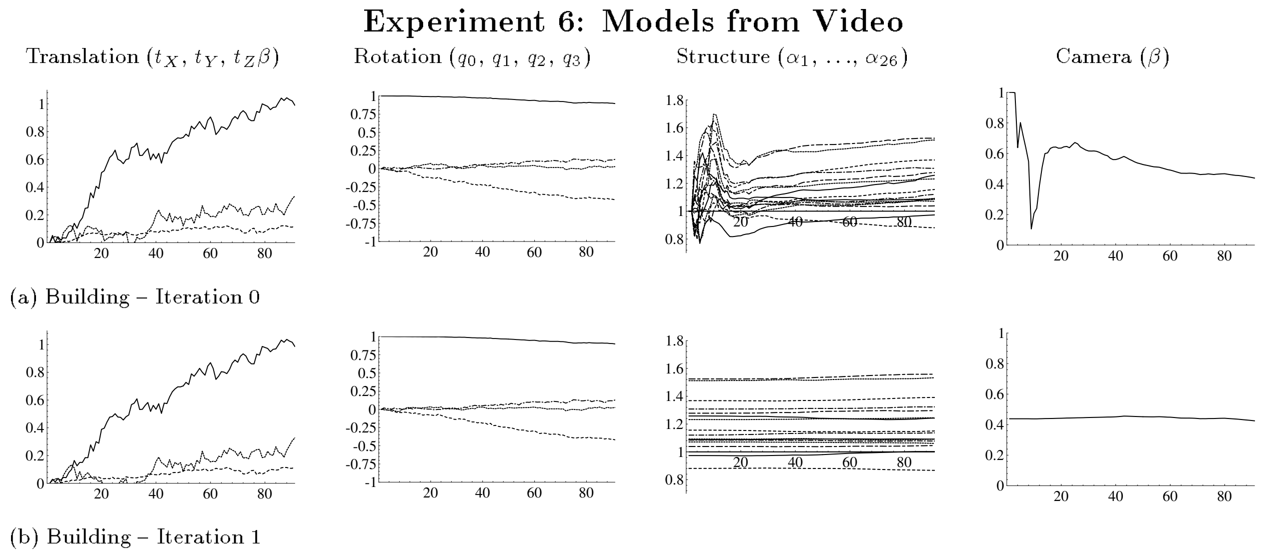
Direct comparison with these statistics is difficult because authors have either performed post-estimation transforms (rigid and scale, or affine) to fit the actual data before computing error statistics [21;29], or used radically different *a priori* information [17]. In our experiment we estimated focal length along with motion and structure, did *not* use special initial conditions, and did *not* use a post-estimation fitting transform before computing error. We



(a) Axial, accuracy vs. field of view. The abscissa is field of view of the actual camera, ranging from 0° (orthographic) to 90°.

(b) Axial, accuracy vs. noise level. The abscissa is the range in pixels of the added uniform noise, ranging from zero to 5 pixels.

Fig. 8. Experiment 5, Axial motion. Accuracy versus field of view and noise level.



(a) Building – Iteration 0

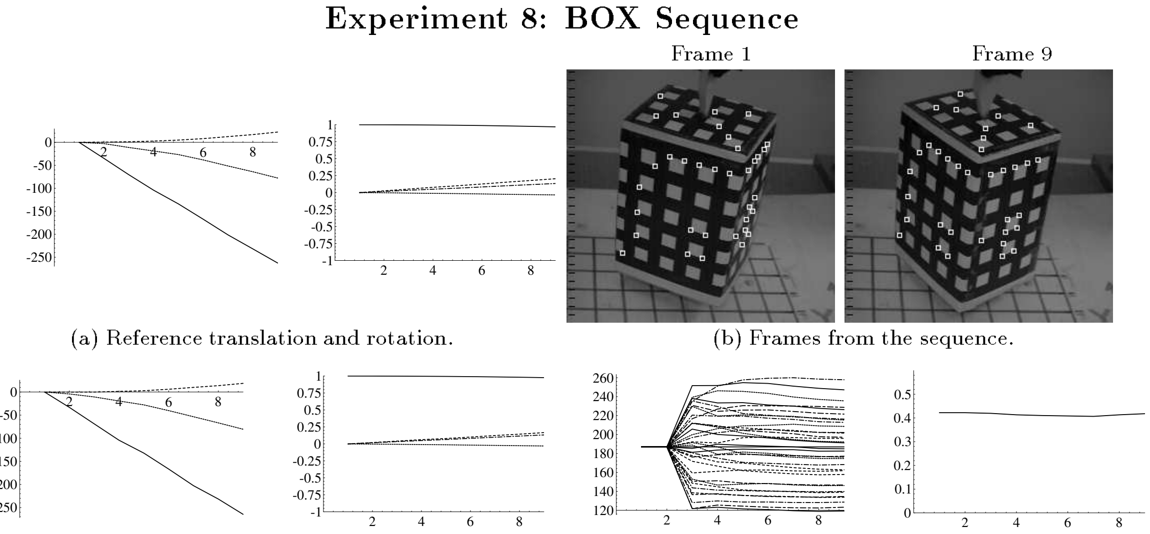
(b) Building – Iteration 1

Fig. 11. Experiment 6: Models from Video. Structure, motion, and focal length recovered from features shown in Figure 9(b)

obtained a rigid motion RMS error of 2.5 % and a structure RMS error of 0.7%. The estimated focal length was 4% different from the manufacturer's specification reported in the data set.

We can show that most of these residual errors are due to errors in the tracking data and in the "ground truth" parameters rather than in our estimate. By using the ground truth motion and the known 3D points and camera, we can synthesize the set of reference feature tracks and compare these to our input feature tracks. The result is that there is a mean RMS image-plane error, over all frames and all features, of 1.45 pixels. Using our recovered trajectory, structure, and focal length, we find only a 0.50 pixel mean RMS image-plane error between input data and resynthesized features. Thus, our recovered parameters comprise a better 3D rigid-motion explanation of the data than do the ground truth parameters. The lower error statistics reported by other authors seem to be in large part due to removing both estimation errors and reference errors through a process of post-estimation 3D alignment.

Figure 12(a) shows the "ground truth" motion parameters and Figure 12(c) shows the parameters recovered by our estimator. Figure 12(b) shows the first and last frames of the 9-frame sequence with features overlaid as white boxes.



(a) Reference translation and rotation.

(b) Frames from the sequence.

(c) Estimated translation, rotation, structure and inverse focal length.

Fig. 12. Experiment 8: Actual motion parameters, the original images, and estimated parameters. The estimated translation and structure have been scaled to match the numerical values of the "actual" parameters. For comparison to other papers using this sequence, the "alpha" depths are from the image plane; distance to COP is an additional 447.3 units.

V. Conclusion

We have presented a feature-based recursive estimator that uses an EKF for recovering the motion, pointwise structure, and focal length for arbitrarily long image sequences. Reformulation of the basic SfM problem using representations that are geometrically equivalent to but better suited for estimation than previous formulations has allowed us develop a filter that is stable and accurate for structure and motion recovery. We have demonstrated the new capability of being able to estimate inverse focal length in addition to structure and motion parameters, which allows for processing uncalibrated imagery. The formulation applies to general perspective projection, including the special case of orthographic projection.

The estimator typically recovers qualitative structure in several frames and converges from there to a more accurate solution. When the camera is unknown, convergence is limited by focal length estimation, which in our experiments takes 20 to 40 frames to converge within 5%. The estimator remains well behaved in degenerate trajectories where previous formulations are reported to have singularities.

The important practical advance, of course, is that, since our estimator recovers camera focal length and the filter does not break down numerically at long focal lengths, most image sequences can be processed directly, without requiring camera calibration or accurate knowledge of focal length. This is most critical when the imagery comes from an unknown camera, in which case the camera is unavailable for calibration and the only source of information regarding the camera is the image sequence itself.

In all of our examples, however, we assumed that the focal length is constant throughout the sequence. Tracking a changing focal length may be considerably more difficult because the measurements appear experimentally to be less sensitive to focal length changes as they are to motion and structure parameters. Also, focal length changes can be confused with translation. Exploring this issue is a topic for future research.

Also, as mentioned in the text, there are several unresolved issues that have not been addressed here and may be candidates for future research topics. One of these is ascertaining the performance tradeoff between our current formulation and a batch process based on the same mathematical models. We expect to see an increase in the convergence and stability properties at the expense of more computation and less flexibility. Another issue pertains to extending the formulation to include biased measurement noise. We expect to see an increase in accuracy at the expense of stability.

Acknowledgements

Many thanks to Bradley Horowitz, with whom the original ideas behind this work were formulated and the first experiments developed and performed. In this paper, Experiment 6 was performed in conjunction with Bradley Horowitz and Tinsley Galyean. Data for Experiment 8 was provided by John Oliensis, was compiled by Inigo Thomas, and originally taken by Harpreet Sawhney and Rakesh Kumar at University of Massachusetts. Some of the other experiments were inspired by discussions with Pietro Perona, Stefano Soatto, Richard Szeliski, Amnon Shashua, and Nassir Navab. This research is partially supported by ONR contract N00014-93-J-0172.

References

[1] Nicholas Ayache and Olivier Faugeras. Maintaining representations of the environment of a mobile robot. *IEEE Trans. Robotics Automation*, 5(6):804–819, 1989.

Experiment 6: Models from Video

Frame 10 Frame 60

(a)
(b)
(c)
(d)

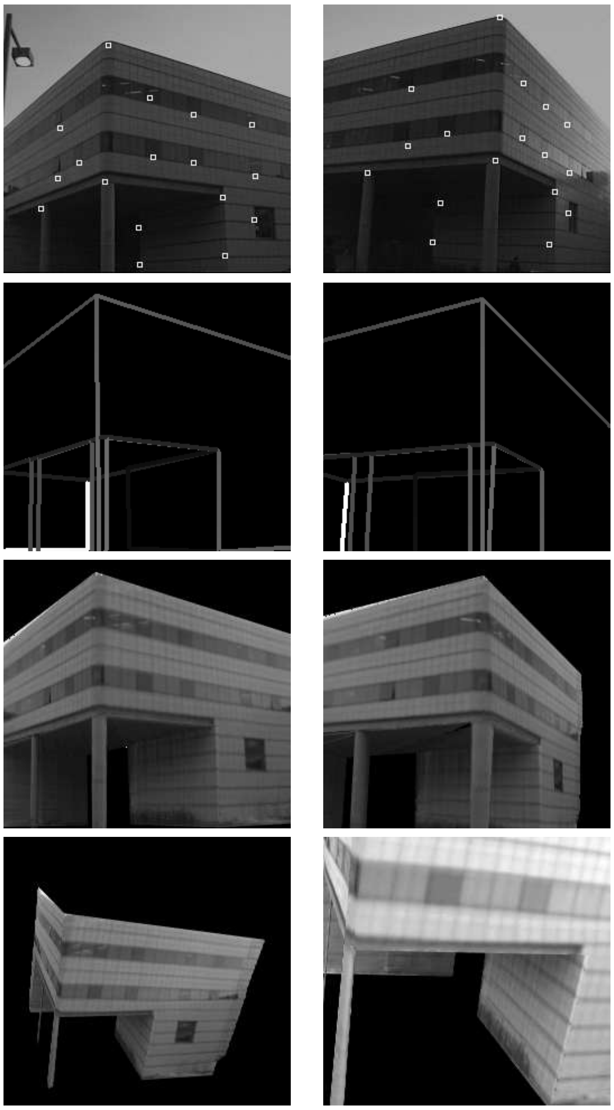
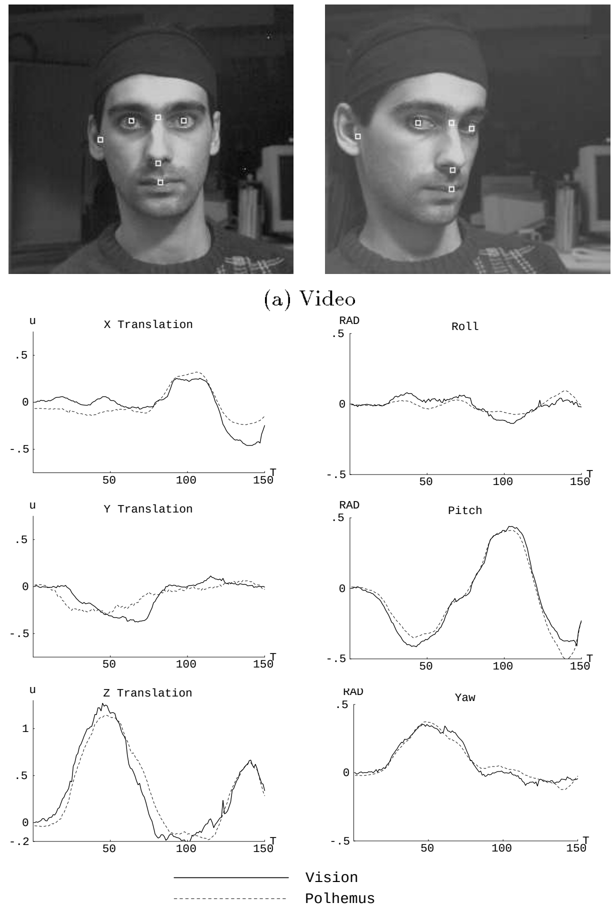


Fig. 9. Experiment 6: Recovering models from video. (a) The features are tracked in the video sequence using normalized correlation. (b) 3-D polygons are obtained by segmenting a 2-D image and back-projecting the vertices onto a 3-D plane. The plane for each polygon is computed from the recovered 3-D points corresponding to image features in the 2-D polygon. (c) Texture maps are obtained by projecting the video onto the 3-D polygons. The estimated motion and camera parameters are used to warp and combine the video from 25 separate frames to create the texture map for each polygon. (d) Alternate views of the recovered model.

Experiment 7: Head Tracking

Frame 1 Frame 109



(b) Vision versus Polhemus Estimates

Fig. 10. Experiment 7: head tracking. (a) Head being tracked, (b) Vision and Polhemus estimates of head position. Much of the observed error is known to be due to Polhemus error. RMS differences are 0.11 units and 2.35 degrees. Scale is unknown, of course, but one unit ≈ 10–12 cm. This experiment uses the same data set as in [2] except that here structure and focal length are estimated in addition to motion.

[2] Ali Azarbayejani, Thad Starner, Bradley Horowitz, and Alex Pentland. Visually controlled graphics. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(6):602–605, June 1993. (Special Section on 3-D Modeling in Image Analysis and Synthesis).

[3] Ali J. Azarbayejani, Tinsley Galyean, Bradley Horowitz, and Alex Pentland. Recursive estimation for CAD model recovery. In *2nd CAD-based Vision Workshop*, Los Alamitos, CA, February 1994. IEEE Computer Society, IEEE Computer Society Press. (Champion, PA).

[4] Ali J. Azarbayejani, Bradley Horowitz, and Alex Pentland. Recursive estimation of structure and motion using relative orientation constraints. In *1993 IEEE Conference on Computer Vision and Pattern Recognition*, pages 294–299, Los Alamitos, CA, June 1993. IEEE Computer Society, IEEE Computer Society Press. (New York).

[5] Ted J. Broida, S. Chandrashekhar, and Rama Chellappa. Recursive estimation of 3-d motion from a monocular image sequence. *IEEE Trans. Aerosp. Electron. Syst.*, 26(4):639–656, July 1990.

[6] Ted J. Broida and Rama Chellappa. Estimation of object motion parameters from noisy images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8(1):90–99, January 1986.

[7] Robert Grover Brown. *Introduction to Random Signal Analysis and Kalman Filtering*. John Wiley & Sons, New York, 1983.

[8] Ernst Dieter Dickmanns and Volker Graefe. Dynamic

monocular machine vision. *Machine Vision and Applications*, 1:223–240, 1988.

[9] R. Dutta, R. Manmatha, L. R. Williams, and E. M. Riseman. A data set for quantitative motion analysis. In *1989 IEEE Conference on Computer Vision and Pattern Recognition*, Los Alamitos, CA, June 1989. IEEE Computer Society, IEEE Computer Society Press. (San Diego, CA).

[10] Olivier Faugeras. What can be seen from an uncalibrated stereo rig? In *Proceedings of the European Conference on Computer Vision*, pages 563–578, June 1992. (Santa Margherita Ligure, Italy).

[11] Olivier D. Faugeras, Nicholas Ayache, and B. Faverjon. Building visual maps by combining noisy stereo measurements. In *Proc. IEEE Conf. on Robotics and Automation*, April 1986. (San Francisco, CA.).

[12] Arthur Gelb, editor. *Applied Optimal Estimation.* MIT Press, Cambridge, MA, 1974.

[13] Joachim Heel. Temporally integrated surface reconstruction. In *ICCV '90*. IEEE, 1990.

[14] Berthold Klaus Paul Horn. *Robot Vision.* MIT Press, 1986.

[15] Berthold Klaus Paul Horn. Relative orientation. *International Journal of Computer Vision*, 4(1):59–78, January 1990.

[16] J. J. Koenderink and A. J. Van Doorn. Affine structure from motion. *Journal of the Optical Society of America*, 8:377–385, 1991.

[17] Rakesh Kumar, Harpreet S. Sawhney, and Allen R. Hanson. 3d model acquisition from monocular image sequences. In *1992 IEEE Conference on Computer Vision and Pattern Recognition*, pages 209–215, Los Alamitos, CA, June 1992. IEEE Computer Society, IEEE Computer Society Press. (Champaign-Urbana, IL).

[18] Ratnam V. Raja Kumar, Arun Tirumalai, and Ramesh C. Jain. A non-linear optimization algorithm for the estimation of structure and motion parameters. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 136–143, June 1989. (San Diego, CA.).

[19] H. C. Longuet-Higgens. A computer algorithm for reconstructinga scene from two projections. *Nature*, 293:133–135, 1981.

[20] Larry Matthies, Takeo Kanade, and Richard Szeliski. Kalman filter based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3(3):209–236, 1989.

[21] J. Oliensis and J. Inigo Thomas. Incorporating motion error in multi-frame structure from motion. In *IEEE Workshop on Visual Motion*, pages 8–13, Los Alamitos, CA, October 1991. IEEE Computer Society, IEEE Computer Society Press. (Nassau Inn, Princeton, NJ.).

[22] Conrad J. Poelman and Takeo Kanade. A paraperspective factorization method for shape and motion recovery. CMU-CS 92-208, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213-3890, October 1992.

[23] Amnon Shashua. Projective depth: A geometric invariant for 3-d reconstruction from two perspective/orthographic views and for visual recognition. In *Proceedings: Fourth International Conference on Computer Vision*, pages 583–590, Los Alamitos, CA, May 1993. IEEE Computer Society and Gesellschaft für Informatik, IEEE Computer Society Press. (Berlin, Germany).

[24] S. Soatto, P. Perona, R. Fraezza, and G. Picci. Recursive motion and structure estimation with complete error characterization. In *1993 IEEE Conference on Computer Vision and Pattern Recognition*, pages 428–433, Los Alamitos, CA, June 1993. IEEE Computer Society, IEEE Computer Society Press. (New York).

[25] Minas Spetsakis and Yiannis Aloimonos. A multi-frame approach to visual motion perception. *International Journal of Computer Vision*, 6(3):245–255, August 1991.

[26] Richard Szeliski and Sing Bing Kang. Recovering 3d shape and motion from image streams using non-linear least squares. In *1993 IEEE Conference on Computer Vision and Pattern Recognition*, pages 752–753, Los Alamitos, CA, June 1993. IEEE Computer Society, IEEE Computer Society Press. (New York).

[27] Richard Szeliski and Sing Bing Kang. Recovering 3d shape and motion from image streams using non-linear least squares. CRL 93/3, Digital Equipment Corporation, Cambridge Research Labs, Cambridge, MA, March 1993.

[28] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, November 1992.

[29] Daphna Weinshall and Carlo Tomasi. Linear and incremental acquisition of invariant shape models from image sequences. In *Proceedings: Fourth International Conference on Computer Vision*, pages 675–682, Los Alamitos, CA, May 1993. IEEE Computer Society and Gesellschaft für Informatik, IEEE Computer Society Press. (Berlin, Germany).

[30] Juyang Weng, Narendra Ahuja, and Thomas S. Huang. Optimal motion and structure estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 144–152, June 1989. (San Diego, CA.).

[31] Juyang Weng, Narendra Ahuja, and Thomas S. Huang. Optimal motion and structure estimation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(9):864–884, September 1993.

[32] G-S. Young and Rama Chellappa. 3-d motion estimation using a sequence of noisy stereo images: Models, estimation and uniqueness. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(8):735–759, January 1990.