Dynamic Reconstruction of 3D Structure and 3D Motion

Hiroshi Ando
Dept. of Brain and Cognitive Sciences
and Center for Biological Information Processing, MIT

Abstract
This paper presents a computational scheme for reconstructing 3D structure, 3D motion and complete surfaces from instantaneous images. The scheme is motivated by psychophysical observations and designed to achieve efficient, robust and flexible behavior. The scheme consists of multiple interacting stages. At the structure-from-motion stage, relative depths and 3D velocities are estimated by minimizing the error in fit to the instantaneous 2D motion measurements and the overall deviation from rigidity. This nonlinear optimization is achieved by an efficient two-stage iterative algorithm. The temporal integration stage is based on Kalman filtering effectively improves the 3D structure. The scheme is motivated by psychophysical observations and attempts to achieve efficient, robust and flexible behavior. The scheme proposed in this paper has the following properties: 1) The structure-from-motion algorithm minimizes the error in fit to the image motion measurements and the overall deviation from rigidity. This non-linear optimization is achieved by an efficient two-stage algorithm that iterates between computing depths and 3D velocities. 2) The algorithm recovers 3D structure and motion from instantaneous motion information in the image computed from two successive frames. But the recursive algorithm based on Kalman filtering effectively improves the quality of the recovered structure over an extended time. 3) The structure-from-motion algorithm directly computes the 3D velocity field rather than the motion parameters of the observer or rigid object, which can be estimated at a later stage. This may increase flexibility to cope with multiple rigid objects and locally rigid objects, since it allows variable weighting of the strength of rigidity between pairs of points on the object. 4) The scheme incorporates a surface reconstruction process that further constrains and stabilizes the recovery of 3D shape.

The scheme presented in this paper has many properties that are consistent with human perception. In particular, it can account for the perceptual demonstrations regarding interactions between two superimposed cylinders explored by Ramachandran et al. [46]. It also can account for experimental results using displays of moving points with short lifetimes [28, 55].

The paper is organized as follows: Section 2 describes some issues and results on human perception of 3D structure from motion. Section 3 sets basic requirements that computational theories should satisfy. Section 4 explores a computational scheme for reconstructing 3D structure and motion field. Section 5 describes the results of computer simulations illustrating the basic behavior of the algorithm and its comparison with psychophysical findings. Finally, Section 6 summarises the results of this paper.

2 Human Perception of 3D Structure from Motion

Human observers can derive a vivid impression of the 3D structure of a moving object from dynamic displays of its 2D projections. The human ability to perceive 3D structure from motion was first studied systematically by Wallach and O'Connell [60], who called this phenomenon the kinetic depth effect. This section summarizes some relevant psychophysical results.

2.1 Resolving ambiguity in various conditions
The fundamental problem in the recovery of 3D structure from motion is its inherent ambiguity, i.e., 3D structure cannot be determined uniquely from motion information in the 2D images unless we impose some constraints. From the infinite number of possible 3D structures and 3D motions, it has been suggested that the human visual system prefers a rigid interpretation of the scene [60, 14]. Rigidity is a property of an object where the distance between any pair of locations on the object in 3D space remains unchanged over time.

A number of kinetic depth demonstrations have supported the use of a rigidity constraint by the human visual system. Human observers can perceive rigid structures from an image sequence of various types of rigid 3D objects; such as wire-frame objects [60], random dots on the surface of a rotating cylinder [e.g. 57, 51, 28, 55], a rotating sphere [33, 10], or a rotating plane [34]. Moreover, when the observer translates relative to the rigid environment, the 3D structure of the scene and the observer's motion can also be inferred [13, 35]. The structure-from-motion process does not seem to require additional spatial constraints, such as the assumption of a smooth surface or plane, since human observers can perceive 3D structure from images of wire-frame objects or random dots in volume where no clear surface or surface boundaries exist [60, 53].
Human observers can also perceive 3D structure even in the presence of large amounts of noise in the images [10, 52, 24]. In the natural world, there exist not only solid objects but elastic objects whose 3D structure can change over time. Human observers can tolerate substantial deviations from rigidity, deriving some 3D structure from the images of nonrigid deformations and distortions, such as stretching, bending, elastic or biological motion [29, 30, 51]. Furthermore, humans can perceive 3D structure from dynamic displays generated by either orthographic or perspective projection.

The above observations suggest that 1) the human visual system may employ the rigidity assumption as a prime constraint for resolving the inherent ambiguity in the recovery of a wide range of 3D structure and 3D motion; and 2) the structure recovery process is robust enough to tolerate substantial deviations from rigidity, which may be caused by noise in the motion measurement process, nonrigidity of an object, or different projection geometries.

2.2 Spatio-temporal characteristics

Recently, the human recovery of structure from motion has been examined more quantitatively by varying spatio-temporal viewing conditions. The following properties are suggested from the spatio-temporal characteristics of the human perception of 3D structure. What information does the human visual system use as inputs to the structure-from-motion process? Experiments using dynamic displays where points persist for a limited time period suggest that the visual system may use instantaneous image velocity information [28, 55]. The experiments explored by Dosher et al. [11] and Landy et al. [32] also suggest that the visual system may use instantaneous motion information based on spatio-temporal energy. The use of instantaneous motion is further supported by other experimental results suggesting that two frame sequences of dynamic displays in oscillation may be sufficient to derive some 3D structure [33, 10, 53, 54]. In the case of two frame oscillation, however, the perceived structure appears to be highly sensitive to small perturbations [33]. Accurate perception of 3D structure may require an extended time. If more than two frames are used, the human system can tolerate larger amounts of noise [42, 10] and the accuracy of the recovery of 3D structure improves over an extended time up to about 1 sec of viewing [24, 55].

Humans also perceive smooth, complete surfaces from displays of sparse points in motion, suggesting that surface values are filled in between depth information derived at the points. More direct perceptual evidence comes from experiments using displays of moving points with short lifetimes [28, 55]. The results suggest that the brain may incorporate an additional interpolation mechanism that maintains and stabilizes 3D surface structure [23].

The above experimental results suggest that 1) the human visual system may use instantaneous motion information computed from two successive frames; 2) it may have a mechanism that improves the accuracy of the recovered 3D structure over an extended time; and 3) it may incorporate the surface interpolation process that further stabilizes the recovery of 3D structure.

2.3 Interaction with various visual processes

Human observers sometimes perceive nonrigid structure and motion from the images of a rigid object in motion. Such illusory perception may be caused by other conflicting static 3D information in the image, such as perspective or static shape effects [2, 60, 48, see also 57, chap.5, 59], or object boundaries [46, 23]. On the other hand, structure from motion can sometimes override other 3D cues, such as stereo. For instance, human observers can perceive the 3D structure of an object from its projected motion generated on a flat display even if it is viewed binocularly. The above observations suggest that the structure-from-motion process is not strictly isolated but has interactions with various other 3D recovery processes.

3 Computational Requirements

From functional properties of the human recovery of structure from motion, we set basic requirements that the computational scheme for the reconstruction of 3D structure should satisfy.

1) Disambiguation: The scheme should resolve the ambiguity inherent in the recovery of 3D structure from motion. It should recover the correct 3D structure and motion from an object with arbitrary (or a wide range of) 3D structure and motion, unless other conflicting static 3D information is present in the images.

2) Efficiency: The scheme should recover at least rough 3D structure from instantaneous motion information computed from two successive frames. The scheme should involve relatively simple computations. If iteration is involved in the computation, the number of iterations should be reasonably small. The recovered structure may be initially crude, but its quality should improve over an extended time.

3) Robustness: The scheme should be robust and stable with random noise. Noise can occur in various processes, such as blur or distortion in the imaging process or error in the motion measurement process. The recovered 3D structure and motion should not be too sensitive to noise but degrade gracefully with increasing noise. The reliability of the estimated 3D structure should improve over time as more motion measurements are integrated.

4) Flexibility: The scheme should be sufficiently flexible to cope with complex natural scenes. More specifically, the following conditions should be considered: (1) Multiple rigid objects: The scheme should cope with multiple rigid objects moving independently in the scene and have some way to segment the scene into multiple rigid objects. (2) Nonrigid objects: The scheme should be able to assume strict rigidity, rather than it should tolerate significant deviation from rigidity. (3) Surface reconstruction: The scheme should interpolate the surface if depths are derived only at sparse locations on the surface. It should also cope with multiple transparent surfaces. (4) Object boundaries: The scheme should incorporate constraints at the object boundaries, since the boundary conditions can affect the reconstruction of 3D shape of both occluded and occluding surfaces. (5) Integration of 3D information: The scheme should be able to integrate various sources of 3D information, such as shape from stereo, texture, perspective, shading and so on.

As described in the previous section, the human visual system may satisfy all these properties. However, it is intuitively very difficult to meet these requirements perfectly. For instance, an algorithm that is very robust and flexible may be computationally expensive and slow. A computational framework presented in this paper attempts to satisfy all of the above requirements, but not all the issues described in 4) are explored in this paper.

4 A Computational Framework

4.1 Related work

This section briefly reviews some existing structure-from-motion algorithms that are based on the rigidity constraint. Earlier computational studies focused on the mathematical analysis of uniqueness of the solution based on the rigidity constraint [e.g. 58 for a review]. It has been shown that under various conditions, motion information in the image over a small extent in space and in time is
sufficient to derive a unique 3D structure [e.g., 57, 44, 37, 56, 25, 49, 63]. The algorithms that integrate image data over a small extent in space and in time, however, are often sensitive to noise in the image. Recent computational studies have explored more practical and robust structure-from-motion algorithms that use motion data in extended space and time and estimate the optimal solutions for 3D structure and 3D motion. The following algorithms estimate the depths and the motion parameters using all available motion information in the image. Bra\-\-\-\-n and Horn [9] developed the least-squares scheme that estimates the motion parameters from the optical flow of the entire image. Heeger and Jepson [19] use the same least-squares cost function but propose an alternative algorithm that first finds the translational components of motion by evaluating a residual function over all candidate translation directions, and then estimates the rotation and the depths. Horn and Weldon [27] extended the least-squares scheme to estimate the motion parameters directly from the derivatives of image brightness for special cases of pure rotation, pure translation, or when the rotation is known.

The following algorithms use motion data both in extended space and time. To avoid storing a number of past images, it is desirable to keep only the current estimates and update the estimates sequentially as more images are obtained. Ullman [59] proposed an algorithm called the incremental rigidity scheme that recovers relative depths incrementally over time from image sequences. The scheme finds depths at feature points by maximizing overall rigidity relative to the current depth estimates. The accuracy of estimated depths usually improves incrementally over time. The scheme can also tolerate deviations from rigidity. Grzywacz and Hildreth [16] developed a continuous formulation of the scheme that uses velocity information instead of positions of features in discrete time. The algorithm, however, requires large spatial displacements of features between frames for stable convergence. The convergence becomes slower for smaller spatial displacements of features and the scheme is weakly stable for instantaneous velocity inputs.

An alternative approach for using multiple frames is to use a recursive estimation method based on Kalman filtering. Mathies et al. [40] developed Kalman filter-based algorithms that estimate dense depths from image sequences using known motion parameters of the observer. Heel [20, 21] developed Kalman filter-based algorithms that estimate a dense depth map as well as motion parameters from optical flow or directly from the gradient of image brightness.

The scheme presented in this paper employs the idea of maximizing rigidity to relax strict rigidity [59, 16], but it is designed to cope with instantaneous motion inputs. It also explores the recursive estimation scheme to improve the quality of the recovered structure over an extended time without using global motion parameters.

### 4.2 The overall scheme

The scheme presented in this paper consists of multiple interacting stages as shown in Fig. 1. The inputs and the goal of the computation at each stage and the interactions among these stages are summarized as follows: 1) The motion measurement stage computes instantaneous 2D motion measurements in the images from changing image brightness. 2) The structure-from-motion stage recovers relative depths and 3D velocities from 2D motion measurements. The algorithm is based on a nonlinear constrained optimization. This optimization is achieved by an efficient two-stage algorithm that iterates between computing depths and 3D velocities. 3) The temporal integration stage improves the recovered 3D structure over an extended time period by effectively averaging the newly estimated depths with the depths estimated in the past. 4) The surface reconstruction stage reconstructs complete 3D surfaces by fitting smooth surfaces to the recovered sparse depth data. This stage also detects and interprets object boundaries based on discontinuities in the recovered depths or the 2D image motion. The reconstructed surfaces can then be used to further improve the 3D structure recovery. 5) The observer/object motion computation stage estimates the global motion of the observer or each constituent object using the recovered 3D structure and 3D velocities. 6) The overall scheme structures the entire image motion with instantaneous motion inputs. It also explores the recursive estimation scheme to improve the quality of the recovered structure over an extended time without using global motion parameters.

### 4.3 Representations and algorithms

This section describes the input and output representations and algorithms in detail for each of the stages shown in Fig. 1.

#### 4.3.1 2D motion measurement stage

The 2D motion measurement stage computes instantaneous motion measurements in the images from image brightness. This section briefly reviews and categorizes some of the existing algorithms. Output from any of these algorithms could be used as the input to the later stages. 1) The early motion measurements: The early motion measurements describe some form of spatio-temporal gradient of brightness and direction of motion, including the spatial and temporal derivatives of brightness [26, 41], the velocity component in the direction of the intensity gradient [22] and the spatio-temporal energy [1, 62, 47]. 2) The 2D velocity field (the optical flow field): To compute a full 2D velocity field, the following

![Figure 1. The overall scheme for reconstructing 3D structure, 3D motion and surfaces.](image-url)
schemes combine the early motion measurements to solve the so-called aperture problem, using additional constraints, such as the smoothness of the velocity field or the pure translation within a local region: the gradient-based schemes [e.g., 26, 22], the correlation-based schemes [36, 3] and the spatio-temporal energy schemes [18, 17, 3]. The 2D velocities of features: The locations of isolated features, such as brightness edges, are first extracted in the image, and their image velocities are then computed by tracking them or establishing their correspondence between successive frames [57].

4.3.2 Structure-from-motion stage

The structure-from-motion stage estimates the 3D position and velocities of each point of an object from its projected position and motion in the image. Since the 3D structure and motion cannot be determined uniquely from image motion alone, we regularize this ill-posed problem by imposing the rigidity constraint on the 3D structure and motion. The algorithm is based on a constrained optimization method that minimizes both the error in the fit to the data and the deviation from the constraint on the solutions.

We can formulate the algorithm based either on features or on image pixels. The feature-based-scheme estimates the depths and 3D velocities for features such as points or intensity edges from their velocities in the image, whereas the image-based scheme estimates the depths and the 3D velocity for each image pixel. In this paper, we formulate the algorithm based on features.

Let \( r_i = (x_i, y_i) \) and \( r_t = (x_t, y_t) \) denote the 2D position and velocity of the \( i \)-th feature in the image and let \( R_i = (X_i, Y_i, Z_i) \) and \( R_t = (X_t, Y_t, Z_t) \) denote its 3D position and velocity in space. If we assume perspective projection with a focal length of one, then

\[
\begin{align*}
\hat{r}_i &= \left( \frac{X_i}{Z_i}, \frac{Y_i}{Z_i} \right) \\
\hat{r}_t &= \left( \frac{X_t}{Z_t}, \frac{Y_t}{Z_t} \right)
\end{align*}
\]

At each moment, the algorithm estimates the depths \( Z_i \) and 3D velocities \( R_i = (X_i, Y_i, Z_i) \) that minimize a cost function consisting of two terms:

\[
E = E_0 + \lambda E_x
\]

where the data term \( E_0 \) describes the total error in the fit of the estimates to the 2D velocity measurements, the regularization term \( E_x \) describes the total deviation from rigidity and \( \lambda \) is a constant that captures the trade-off between the two terms.

From the equation (2), the data term \( E_0 \) is written as follows:

\[
E_0 = \sum_i \left[ a_{i1} (\hat{X}_i Z_i + x_i - X_i)^2 + a_{i2} (\hat{Y}_i Z_i + y_i - Y_i)^2 \right]
\]

where \( a_{i1} \) and \( a_{i2} \) are the weights that describe the confidence or reliability associated with individual velocity measurements.

The regularization term \( E_x \) describes the total deviation from rigidity, which is defined as the overall instantaneous change in 3D distances between features [16]. This term also incorporates weighting factors \( w_i \) that capture the strength of the rigidity of the connection between the feature \( i \) and feature \( j \). The variable weights could be used to segment multiple rigid objects and cope with locally rigid objects. Therefore the term \( E_x \) is written as

\[
E_x = \sum_i \sum_j w_{ij} \left( \frac{d_i}{dt} l_{ij} \right)^2
\]

where \( l_{ij} \) is the 3D distance between feature \( i \) and \( j \):

\[
l_{ij} = |R_i - R_j|
\]

The term \( E_x \) can be rewritten in terms of the 3D velocities, the depths and the positions of features in the image as follows:

\[
E_x = \sum_i \left[ \frac{(x_i Z_i - x_t Z_t)^2 + (y_i Z_i - y_t Z_t)^2 + (Z_i - Z_t Z_i Z_t)^2}{(x_i Z_i x_t Z_t)^2 + (y_i Z_i y_t Z_t)^2 + (Z_i Z_t)^2} \right]
\]

The above cost functional \( E \) is non-quadratic in the depths \( Z_i \) and the 3D velocities \( R_i = (X_i, Y_i, Z_i) \). Minimization of a quadratic functional leads to linear optimization problems whose solutions can generally be found uniquely, as in the case of many early vision problems [43, 5]. But minimization of a non-quadratic functional requires to solve a set of nonlinear equations. Standard nonlinear optimization algorithms, such as the steepest descent method, the conjugate gradient method or the quasi-Newton method [38, 45], are usually slow and can become trapped in a local minimum of the solution space. To avoid the use of these standard optimization methods, a more efficient algorithm is proposed. The algorithm uses a two-stage strategy to perform the minimization that alternates between computing depths \( Z_i \) and 3D velocities \( R_i = (X_i, Y_i, Z_i) \).

Specifically, the algorithm sequentially estimates the 3D velocities and the depths by repeating the following two stages until the algorithm converges to an equilibrium state:

1) The 3D velocity-estimation stage: This stage estimates the 3D velocities \( R_i = (X_i, Y_i, Z_i) \) by minimizing the cost function (3), assuming that the depths \( Z_i \) are fixed. In this case, the cost function becomes quadratic and a unique set of 3D velocities is normally found by solving a system of linear equations. Standard relaxation methods can be used to find the solution, such as the Gauss-Seidel method for serial implementation and the Jacobi or Chebyshev method for parallel implementation. For initial values of the depths, some constant value can be used unless prior depth information is available from other sources of 3D information, such as stereo.

2) The depth-estimation stage: This stage computes the depths \( Z_i \) using the 3D velocities \( R_i = (X_i, Y_i, Z_i) \) estimated at the 3D velocity-estimation stage. The depths can be computed in one step using the equation (2) as follows:

\[
Z_i = \left( \frac{X_i - x_i Z_i}{\dot{x}_i} \right), \quad Z_t = \left( \frac{Y_t - y_t Z_t}{\dot{y}_t} \right)
\]

These two estimates of depths can be combined to obtain a single estimate using a weighted average of the two. The weight can depend on the absolute value of each image velocity component, since equation (8) shows that the computed depth is more sensitive to noise for smaller image velocity components.

Suppose that the 3D structure is strictly rigid and no noise is present in the image velocity measurements. If the correct depths are given, the 3D velocity-estimation stage computes the correct 3D velocities since a unique set of solutions can be found. If the correct 3D velocities are given, the depth-estimation stage computes the correct depths. Therefore, the correct solution must be a stable or unstable equilibrium state of this two-stage dynamical system. If the correct solution is a stable equilibrium state, the algorithm is expected to converge to the correct solution. But we further need to understand its stability, efficiency, existence of local minima, and sensitivity to noise. We analyze some of these properties through computer simulations in Section 5.
4.3.3 Temporal integration stage

The depths computed from only two frames may not be accurate, as errors can occur for various reasons: for example, the images can be blurred or distorted during the imaging process, the 2D motion measurements may contain random noise, or the structure and motion of objects may violate the underlying assumptions of the motion measurement or structure-from-motion algorithms, such as the rigidity assumption. The goal of temporal integration is to estimate more reliable depths by combining information from multiple frames. The idea behind this process is that random errors may be smoothed out by effectively averaging the 3D structures computed from multiple frames.

The integration algorithm presented here is based on an optimal estimation theory called the Kalman filter [31]. The Kalman filter embodies a general framework for estimating dynamically changing random variables from measurements containing noise [e.g., 12, 4]. The algorithm incorporates the representations of uncertainty in the estimates and sequentially reduces this uncertainty over time. Recently, Kalman filtering has been applied to the 3D structure estimation problem and has been shown to improve significantly the quality of the estimated structure over time. For example, Mathies et al. [40] developed Kalman filter-based algorithms that estimate dense depths from image sequences using known motion parameters of the camera or observer. Hee [20, 21] developed Kalman filter-based algorithms that estimate a dense depth map as well as motion parameters of a rigid object or the observer from optical flow or directly from the gradients of image brightness. This section presents a simple algorithm based on Kalman filtering that incrementally improves the accuracy of depth estimates. Unlike previous algorithms that use known or estimated motion parameters, it directly uses the 3D velocities of features or pixels computed at the structure-from-motion stage.

The Kalman filter estimates dynamically changing state variables from noisy measurements based on prior knowledge of the dynamics of state variables. We first specify the models of temporal dynamics that describe how the depth is changing over time (the system model) and how the depth measurement is generated (the measurement model). We regard the depth of each feature as a scalar state that we estimate and the velocity \( \frac{dz}{dt} \) as a known parameter to the system. This assumption significantly simplifies the algorithm. The algorithm therefore estimates the depth \( Z \) based upon a set of depth measurements up to time \( t \) that are corrupted by uncorrelated noise. The system model that describes the temporal change of the depth over time is written as

\[
Z_{t+1} = Z_t + V_t \Delta t + \alpha_t \quad \text{(9)}
\]

where \( \Delta t \) denotes the interframe time interval and \( \alpha_t \) is zero mean white noise of variance \( Q_t \). Here, we assume that the temporal derivatives higher than the first are negligible for the short time interval \( \Delta t \). The measurement model describes how the depth measurement is generated from the true depths and contaminated by noise. We assume that the depth measurement \( \hat{z} \) is the depth computed at the structure-from-motion stage at each time \( t \) which has additional noise. Thus we have

\[
\hat{z} = z + \eta_t \quad \text{(10)}
\]

where \( \eta_t \) is zero mean white noise of variance \( P_t \).

Assuming the above dynamical models, the algorithm estimates the depth that minimizes the uncertainty in the estimates of depth. This uncertainty can be expressed by the conditional error variance of the depth estimate. This estimate is called the conditional minimum variance (CMV) estimate. We could alternatively maximize the conditional density function for the depth conditioned on given past measurements. This estimate is called the maximum a posteriori (MAP) estimate. We could use either estimate, since the CMV is identical with the MAP if the conditional density function is Gaussian. (Refer to [12] for more details.)

The Kalman filtering algorithm finds the CMV estimate of depth at each time instant \( t \) from the depth measurements up to time \( t \). It is a recursive algorithm that sequentially updates the current estimate of depth using only the available depth measurements, so there is no need to store past measurements for the computation of the current estimate. The algorithm also maintains and updates the reliability (or the inverse of the uncertainty) of the depth estimate. The recursive algorithm consists of two sets of equations. The prediction equations describe how the depth estimate and its reliability are predicted from the current depth estimate. The update equations describe how the current estimate of depth and its reliability are improved by integrating the predicted depth with the new depth data. In this paper, we describe only the results of these equations, but their derivation is straightforward from the Kalman filter algorithm [12].

Let the estimates of depth at time \( t \) be denoted by \( Z_t \) and \( \hat{Z}_t \), where the symbol "*" denotes an estimate, and the superscripts "+" and "-" denote the estimates before and after the updating stage, respectively. We also denote the reliability of depth before and after the update as \( q_t \) and \( a_t \), respectively. The algorithm first predicts the estimate of depth at time \( t \), \( \hat{Z}_t \), from the previous estimate \( \hat{Z}_{t-1} \) using the computed velocity in depth, \( \hat{z}_t \), whereas the reliability of depth decreases by some amount if the system noise of variance \( Q_t \) is not zero:

**Prediction**

\[
\hat{Z}_t = Z_{t-1} + \hat{z}_t \Delta t , \quad \text{(11)}
\]

\[
a_t = \frac{\alpha_t}{\alpha_t + Q_t} , \quad \text{(12)}
\]

where \( \alpha_t \) is a constant called the age-weighting factor [12]. If \( \alpha \) is greater than one, the reliability further decreases during the prediction so that the temporal integration relies on more recent depth measurements.

Then, the algorithm updates the current estimate of depth by taking an average of the predicted depth \( \hat{Z}_t \) and the newly computed depth \( Z'_t \), weighted by their reliability. The reliability of the depth estimate is updated by adding the previous reliability with the reliability of the newly computed depth, so that the reliability of the estimate increases as more reliable depth measurements are integrated:

**Update**

\[
\hat{Z}'_t = \frac{1}{a_t + \alpha_t} (\alpha_t \hat{Z}_t + \alpha_t Z'_t) , \quad \text{(13)}
\]

\[
a_t' = a_t + a_t \quad \text{(14)}
\]

where \( a_t \) is the inverse of the measurement variance, \( a_t = P_t^{-1} \), that describes the reliability of the newly computed depth.

The reliability of the depth measurement \( a_t \) depends on how the error in the depth is generated and conveyed in the earlier processes. Although it is difficult to model the sources of error precisely, some heuristics can be used; for example, when the velocity in the image...
is small, the computed depth is more sensitive to noise as shown in equation (8), so less reliability can be given to this feature.

4.3.4 Surface reconstruction stage

The surface reconstruction stage recovers complete smooth surfaces from sparse depth data estimated at the structure-from-motion stage. We use an algorithm that approximates the smooth surface using sparse depth data by minimizing the error in the fit to the depth data and the overall second spatial derivatives of depth [15, 50]. Let \( s(x, y) \) denote the depth of the surface and \( c_i \) denote the known depth points at \( (x_i, y_i) \). Then the cost function is written as follows:

\[
\sum_i [s(x_i, y_i) - c_i]^2 + \beta \left( \frac{\partial^2 s}{\partial x^2} + 2 \frac{\partial^2 s}{\partial x \partial y} + \frac{\partial^2 s}{\partial y^2} \right) \, dy, \tag{15}
\]

where \( \beta \) is a constant that captures the trade-off between the data term and the smoothness term. Standard optimization algorithms, such as Gauss-Seidal relaxation, can be used to minimize this cost function. We could use the algorithms that reconstruct the surface and detect depth discontinuities simultaneously by optimizing piece-wise smoothness of the surface [e.g., 39, 46], but for the computer simulations, we use the algorithm derived from the functional (15) for its simplicity.

There may exist multiple surfaces in the same visual direction, such as the case of the rotating transparent cylinder widely used in psychophysical experiments. In such cases, we segregate the depth data points into different groups based on their speed and direction of motion in the image. The surface approximation algorithm then operates on each group of depth data independently to reconstruct separate representations of surfaces.

4.3.5 Object / observer's motion computation stage

Once the 3D structure and the 3D velocities are computed at the structure-from-motion stage, we can compute the motion parameters of the observer or the object. The motion parameters consist of three translational components \( t = (U, V, W)^T \) and three rotational components \( \omega = (A, B, C)^T \). The 3D positions \( \textbf{R} = (X, Y, Z)^T \) and the 3D velocities \( \dot{\textbf{R}} = (\dot{X}, \dot{Y}, \dot{Z})^T \) have the following relation to the motion parameters of the observer:

\[
\dot{\textbf{R}} = t - \omega \times \textbf{R}. \tag{16}
\]

We can therefore find the motion parameters using the least squares method that minimizes the following function:

\[
\sum_i [\dot{R}_i - t + \omega \times R_i]^2. \tag{17}
\]

5 Computer Simulations

In this section, we examine the basic properties of the scheme through computer simulations and show that its behavior is consistent with some psychophysical observations.

5.1 Convergence and efficiency

The issues that we explore in this section are whether the structure-from-motion algorithm converges to the correct 3D structure and how fast the 3D structure and 3D velocities are recovered.

To examine the convergence of the structure-from-motion algorithm, 50 feature points are generated randomly in a cube and translated at a constant velocity in 3D space. The length of each side of the cube is 30 and the depth of the center of the cube is 100 from the observer. The points are translated by \( (\Delta x, \Delta y, \Delta z) = (1, 1, 1) \) between two successive frames. Fig. 2 (a) shows the projection of the true 3D structure and 3D velocity field onto the X-Z plane. Each dot represents the location of a point and the associated line represents its velocity. The velocities are shown ten times larger than their actual displacements between frames.

The feature points are then projected onto the image plane by perspective projection with a focal length of one. From the locations of the points in the two successive images, 2D image velocities are computed analytically and used as inputs to the structure-from-motion algorithm. No noise is added to the image velocities in this case. Fig. 2 (b) shows the location of points and their 2D velocities in the image. Note that the algorithm has three levels of iterations (see Fig. 1). Let \( M \) denote the number of motion iterations for estimating 3D velocities, \( \sigma \) denote the number of structure iterations for estimating depth, and \( t \) denote the actual time step when the new motion data is obtained.

We assume that the initial structure is flat, that is, all feature points are located at the same distance from the image plane as shown in Fig. 2 (c). The algorithm first estimates the 3D velocities with the depths fixed at the given locations. The Gauss-Seidal relaxation method is used to find these 3D velocities by minimizing the cost function (3). 10 motion iterations were performed to find the solution. Then the depths of the points are computed from the perspective equations (8) using the estimated 3D velocities. These two stages are repeated for fifteen times. Fig. 2 (c) shows the projection of the 3D structure and 3D velocities recovered after 1, 2 and 15 structure iterations. In this simulation, \( \lambda \) was set to 0.15. The correct structure and 3D velocities are roughly recovered after

![Figure 2. Simulation results: random dots in a cube with pure translation. (a) The true 3D structure and 3D velocities. (b) The 2D velocity field in the image. (c) The initial and reconstructed 3D structures and 3D velocities after 1, 2, and 15 structure iterations.](image-url)
2 or 3 structure iterations and the recovered structure and 3D velocities are almost identical to the true structure and 3D velocities after 15 structure iterations. We also found that the algorithm can recover the correct 3D structure and motion even if the initial structure is random rather than flat.

To quantify the improvement of the estimated structure, the total cost defined by (3) and the error in the estimated structure are plotted over the number of structure iterations in Fig. 3 (a) and (b). Since 3D structure can only be recovered up to a scale factor from motion information, the size of the recovered structure depends on the initial depths of the feature points. For instance, if the initial structure is closer to the observer than the true structure, the structure is recovered around this initial structure, which is smaller than the true structure in size. The structure error is thus defined as the variance of the ratio between the estimated depth and the true depth. If the true structure is recovered up to a scale factor, the structure error must become zero. In these figures, the total cost and the structure error are normalized by the initial cost and the initial structure.

The graphs show that the total cost and the structure error become almost zero after 4 structure iterations when \( \lambda = 0.15 \), that is 40 motion iterations altogether. This convergence rate is much faster than usually required in standard nonlinear optimization methods. If we use smaller \( \lambda \), the convergence becomes slower as shown in Fig. 3 (a) and (b) and the recovered 3D structure is flatter in the earlier stages of iteration.

![Figure 3. The total cost and the structure error in the case of pure translation.](image)

The simulation results can be summarized as follows. The results are discussed in terms of computational and psychophysical considerations. 1) The structure-from-motion algorithm recovers the correct 3D structure from instantaneous velocities of feature points computed from two successive frames if there is no noise in the image. This result is compatible with psychophysical observations showing that two frames in oscillation are sufficient to perceive 3D structure[33, 10, 53, 54]. Treue et al. [55] also suggest that humans use instantaneous velocities as inputs to the structure-from-motion process, showing that the threshold of point lifetime for detecting 3D structure of a rotating cylinder is fairly short (50-85 ms) and constant over a wide range of number of points and velocities. 2) The algorithm recovers at least rough 3D structure with a relatively small number of iterations. The number of motion and structure

![Figure 4. Simulation results: random dots on the surface of a rotating cylinder.](image)
5.2 Robustness against noise

This section examines the robustness of the scheme when noise is present in the image velocities and shows that temporal integration based on Kalman filtering can incrementally improve the quality of the recovered 3D structure over an extended time.

In this simulation, 50 points are randomly positioned on the surface of a cylinder and rotated at 1 degree per frame. Gaussian noise is added to the image velocities of the points. The amount of noise is relative to the size of the image velocity with a standard deviation of 0.5, yielding an average error of 20% in the velocity components. Fig. 6 shows the fields of image velocities with and without noise at one instant. In the following simulations, we assume that the variance of the system noise is zero, the age-weighting factor is one and the reliability of depth measurement for each point is proportional to the size of its image velocities.

The results of simulations are shown in Fig. 7. The 3D structure of the cylinder is not clear at t=1 when a single velocity field is used because of the noise in the image. However, as more frames are integrated, the 3D structure of the cylinder becomes apparent and improves over time. Fig. 8 shows the structure error plotted over the number of frames. The solid and dotted lines indicate the results with and without the temporal integration, respectively. As shown in the graph, the temporal integration algorithm significantly reduces the structure error as more frames are used.

A number of psychophysical studies have shown that accurate perception of 3D structure from motion may require an extended time [e.g. 42, 28, 24, 55]. In particular, the human recovery of structure from motion may be highly sensitive to noise when two frame sequences of random-dot displays are presented, but it can tolerate large amounts of noise if more than two frames are presented [33, 10]. This result suggests that the temporal improvement may not be due to the delay in the computation, but rather to an actual improvement in the quality of the recovered 3D structure. Thus, the human visual system may incorporate a temporal integration mechanism similar to the algorithm presented here that effectively averages the 3D information of multiple frames.
5.3 Recovering 3D structure with surface reconstruction

When depths are recovered only at sparse feature points, a complete 3D surface can be reconstructed by fitting a smooth surface to the estimated depths at the features. If this representation is maintained over an extended time, it is expected that the reconstructed data further stabilizes the 3D structure recovery even when the temporal integration for individual feature points is limited due to the noise in the image, for instance.

Such a role for surface reconstruction in the human recovery of structure from motion is suggested by the psychophysical experiments using moving points with short lifetimes [28, 55]. In these experiments, an individual point on a cylindrical surface persists for only a limited time period and reappears at a different location on the surface. The results show that although the lifetime of individual points can be very short, about 100 msec, accuracy of the recovered 3D structure improves over an extended time up to about a second. To improve the quality of recovered structure over time, the human visual system may incorporate an interpolation mechanism that maintains and stabilizes the 3D shape representation.

To simulate this experimental condition, the points are generated in such a way that at every 2 degrees of rotation, half of the points disappear and reappear at different locations on the surface of the cylinder, so that each point persists for only 4 degrees. Gaussian noise is also added to the image velocities. As shown in Fig. 9 (a), if the surface is not interpolated, the clear 3D structure of the cylinder cannot be recovered, because the depth of each point is obtained by integrating its motion for only 4 degrees. The simulation then incorporates the surface interpolation at every 2 degrees of rotation and uses the depths of the interpolated surface to assign depths for the reappeared points. During the interpolation process, the back and front surfaces of the cylinder are reconstructed separately as shown in Fig. 10, by segregating the points into two groups based on direction of motion. Fig. 9 (b) shows that the quality of cylindrical structure improves over time if surface interpolation is incorporated. The results suggest that surface reconstruction may play an important role in the human recovery of structure from motion. (For further implications on human perception, see [33].)

5.4 Coping with multiple rigid objects

The simulations in this section explore the flexibility of the scheme at coping with complex scenes. Human observers perceive nonrigid motion when two rotating cylinders of dots are superimposed in such a way that there is no single rigid object consistent with these displays (Ramachandran et al. [46]). In particular, 1) when the two superimposed cylinders have the same size but one cylinder is rotating faster than the other, we perceive two surfaces separated in depth, although the true surfaces coincide in depth, and 2) when the two superimposed cylinders have a different size but the 2D image velocities are the same in the center of the image of the cylinders, we perceive a single surface in the center of the cylinder, although the true surfaces are separated in depth.

Fig. 11 (a) shows the true 3D structures used in the simulations. The structure-from-motion algorithm maximizes the overall rigidity and does not require strict rigidity. Thus, it computes the most rigid interpretation of all the points taken together. As shown in Fig. 11 (b), the computed 3D structures agree with our perceptions. Algorithms that assume strict rigidity or use global motion parameters may not yield such structures. To reconstruct the underlying surfaces, we assume that the points are segregated into two groups based on their difference in image speed. Thus, the two transparent surfaces can be reconstructed separately, as shown in Fig. 11 (c).

Figure 9. (a) The recovered 3D structure without surface interpolation. (b) The recovered 3D structure with surface interpolation.

Figure 10. The reconstructed back and front surfaces of the cylinder.

Figure 11. Simulations on the two superimposed cylinders. 1) The two cylinders have the same size but one cylinder is rotating faster than the other. 2) The two cylinders have a different size but the 2D image velocities are the same in the center of the cylinders. (a) The true 3D structures. (b) The reconstructed 3D structures. (c) The reconstructed multiple surfaces.

6. Conclusion

This paper presents a computational scheme for reconstructing 3D structure and 3D motion from motion information in the image. The scheme is designed to achieve efficient, robust and flexible behavior simultaneously as suggested from human perception. The simulation results show that the scheme reasonably satisfies this requirement: 1) The scheme can recover the 3D rigid structure and 3D velocity field from instantaneous image motion with a relatively small number of iterations. 2) The temporal integration algorithm improves its quality over an extended time. 3) Its robust and flexible behavior is consistent with many psychophysical observations,
including the interaction between multiple rigid objects. Future research will further analyze its performance with various types of structure and motion and explore its potential ability to deal with nonrigid motion and segmentation of multiple moving objects.

Acknowledgements

I would like to thank Ellen Hildreth and Shimon Ullman for valuable discussions and their comments. Support for this research is provided by the Office of Naval Research, Cognitive and Neural Systems Division, the National Science Foundation (IRI-8557824), and the McDonnell Foundation.

References


