Towards Direct Recovery of Shape and Motion Parameters from Image Sequences

Stephen Benoit and Frank P. Ferrie
McGill University,
Center for Intelligent Machines,
3480 University St., Montréal, Québec, CANADA H3A 2A7
Tel.: +1 514 398-2185 FAX: +1 514 398-7348
{*benoits, ferrie*}*@cim.mcgill.ca*

Abstract

A novel procedure is presented to construct image-domain filters (receptive fields) that directly recover local motion and shape parameters. These receptive fields are derived from training on image deformations that best discriminate between different shape and motion parameters.

Beginning with the construction of 1-D receptive fields that detect local surface shape and motion parameters within cross sections, we show how the recovered shape and motion model parameters are sufficient to produce local estimates of time to collision.

In general, filter pairs (receptive fields) can be synthesized to perform or detect specific image deformations. At the heart of the method is the use of a matrix to represent image deformation correspondence between individual pixels of two views of a surface. The image correspondence matrix can be decomposed using Singular Value Decomposition to yield a pair of corresponding receptive fields that detect image changes due to the deformation of interest.

1. Introduction

Recovery of structure from motion has been examined from a variety of approaches, mainly feature point extraction and correspondence[9, 13] or computing dense optical flow[14, 1]. Typically, the Fundamental Matrix framework or a global motion model is used to solve for global motion after which the relative 3-D positions of points of interest in the scene can be computed[16, 21].

Direct recovery of parameters by recognising characteristic motions in a scene has been examined using point correspondences[15] and local optical flow field deformations[4, 8, 18]. However, both methods require the extraction of motion information from the image sequence before parameter recovery. Many of these techniques require local regularization; insufficient information is available in each sample to reconstruct the surface shape[20].

Appearance-based methods have been mostly discarded for structure from motion because much of the shape and motion information are so confounded that they cannot be recovered separately or locally[3]. Soatto proved that perspective is non-linear, therefore no coordinate system will linearize perspective effects[17].

Part of the difficulty involves how regional image changes can be represented in an appearance-based framework. Most approaches involve solving a local flow field as a weighted sum of basis flow fields with some perceptual significance[5, 6] or tracking specific image features[12] with wavelets[7], typically training on image sequences of motions of interest[22]. Converting these representations into image domain operations[11, 19] could allow direct recovery of significant model parameters without solving a local optimization problem by gradient descent.

Two causes prevent the effective direct recovery of local structure from motion with appearance-based methods. First is representation: how to encode image deformation independently of image texture and without the need of feature points. Second is how to map the image deformations into some optimal image operators.

This paper addresses both issues by describing a representation and a methodology for designing and using these optimal image domain operators for appearance-based local recovery of some shape and motion parameters. This feed-forward system is used to compute a dense map of time to collision in image sequences. The optimal operator synthesis will discover what spatial scales most appropriately describe the different deformations for the camera model.

Instead of attempting to recover the shape and motion parameters of a scene over all the possible parameter space, only those shapes and motions that cause distinctly different

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set
0-7695-1950-4/03 $17.00 © 2003 IEEE

image deformations are considered. Even with aliasing of some parameters into similar appearances, enough information can be extracted from the image deformations to recognise specific shapes or motions which are useful for specific tasks such as computing time to collision.

By understanding the image formation process, the mapping from the shape and motion model parameters to image deformations (image correspondences) can be expressed in a matrix form that precisely encodes the image plane correspondences of a given model instance. In this paper, we will apply the theory to 1-D cases without loss of generality.

A point-correspondence mapping $\mathcal{M}$ from a point $\mathbf{x}$ in a first space to a second space at point $\mathbf{y}$ is expressed as:

$$\mathbf{x} \subset \mathbf{X} \rightarrow \quad \mathcal{M} \quad \rightarrow \mathbf{y} \subset \mathbf{Y} \tag{1}$$

$$\mathbf{y} \subset \mathbf{Y} \rightarrow \quad \mathcal{M}^{-1} \quad \rightarrow \mathbf{x} \subset \mathbf{X} \tag{2}$$

The function $\mathcal{M}$ may be continuous, discontinuous, piecewise linear or non-linear, multi-valued, and in short, arbitrarily complex. But if the spaces $\mathbf{X}$ and $\mathbf{Y}$ can be discretized, the function $\mathcal{M}$ can be expressed as a correspondence matrix mapping the index of a discrete $[\mathbf{x}]$ to the index of a discrete $[\mathbf{y}]$ and back.

The correspondence matrix $\mathbf{H}$ is a representation of such an image deformation or a coordinate remapping. Element $\mathbf{H}_{ij}$ is a non-negative real number indicating the amount (probability) of correspondence between coordinate i in the first space, $[\mathbf{X}]_i$ and coordinate j in the second space, $[\mathbf{Y}]_j$.

$$\mathbf{H}_{ij} \triangleq \mathrm{P}\left([\mathbf{X}]_i \overset{\mathcal{M}}{\Longleftrightarrow} [\mathbf{Y}]_j\right) \tag{3}$$

$$\mathbf{H}_{ij} = \frac{\begin{pmatrix} \mathcal{A}\left(\mathcal{M}([\mathbf{X}]_i) \cap [\mathbf{Y}]_j\right) \\ + \mathcal{A}\left([\mathbf{X}]_i \cap \mathcal{M}^{-1}([\mathbf{Y}]_j)\right) \end{pmatrix}}{\mathcal{A}\left([\mathbf{X}]_i\right) + \mathcal{A}\left([\mathbf{Y}]_j\right)} \subset [0,1] \tag{4}$$

where $\mathcal{A}(\mathcal{N})$ is a measure of area or volume in the neighborhood $\mathcal{N}$, a measure of the size of the Voronoi cell of $\mathcal{N}$.

The correspondence matrix thus described is not to be confused with the *fuzzy correspondence matrix* of Ben-Ezra *et al.*[10] which described the correspondences at each point $\mathbf{p}$ with a matrix $\mathbf{M}(\mathbf{p})$ where each cell (i,j) corresponds to a probability that point $\mathbf{p}$ has a displacement (i,j).

Note that $\mathbf{H}$ is a discretized representation of a possibly continuous mapping $\mathcal{M}$. Both i and j can represent single-axis (i.e. 1-D) positions, but could also represent an arbitrary indexing into a multi-dimensional space. To construct a correspondence matrix $\mathbf{H}$,

1. Discretize the first image space $\mathbf{X}$ into M elements.
2. Discretize the second image space $\mathbf{Y}$ into N elements.
3. Initialize $\underset{M\times N}{\mathbf{H}} = \mathbf{H}|_{\mathcal{M}} = \mathbf{H}|_{\mathcal{M}^{-1}} = 0$.
4. For each $i \in \{1,\ldots,M\}$:
   - using a uniform distribution of points in $[\mathbf{X}]_i$,
   - compute the distribution of points $[\mathbf{Y}]_j$ at or near $\mathcal{M}([\mathbf{X}]_i)$,
   - assign $H_{ij}|_{\mathcal{M}} = \mathcal{A}\left(\mathcal{M}([\mathbf{X}]_i) \cap [\mathbf{Y}]_j\right)$.
5. For each $j \in \{1,\ldots,N\}$:
   - using a uniform distribution of points in $[\mathbf{Y}]_j$,
   - compute the distribution of points $[\mathbf{X}]_i$ at or near $\mathcal{M}^{-1}([\mathbf{Y}]_j)$,
   - assign $H_{ij}|_{\mathcal{M}^{-1}} = \mathcal{A}\left([\mathbf{X}]_i \cap \mathcal{M}^{-1}([\mathbf{Y}]_j)\right)$.
6. Assign $\mathbf{H}_{ij} = \frac{\mathcal{A}([\mathbf{Y}]_j)\mathbf{H}_{ij}|_{\mathcal{M}} + \mathcal{A}([\mathbf{X}]_i)\mathbf{H}_{ij}|_{\mathcal{M}^{-1}}}{\mathcal{A}([\mathbf{X}]_i) + \mathcal{A}([\mathbf{Y}]_j)}$.

By choosing a model that may alias the appearance changes due to some parameters together but generates distinct appearance changes for the most important parameters, the model parameter space can be discretized over the range of interesting scene events. Only a finite number of these correspondence matrices are required to encode the appearance changes for changes of those most important parameters over all variations for the aliased parameters.

The scenario presented in Section 2 is the recovery of shape and motion using a simple surface model that captures image translation and scale change. By sampling the model space and generating the correspondence matrices for key shapes and motions, the recoverable parameters are enough to deduce the time to collision from an image sequence. The operators are tested with synthetic and real data in Section 4 and demonstrate the successful recovery of time to collision for synthetic and real image sequences.

With image formation models in mind, Section 3 will derive the unified solution of converting the image formation models into image domain operators, and how they are used to detect image events of interest.

2. Image Formation Model for Local Shape and Motion

This section details a model that encodes planar image translation, like optical flow, as well as depth (or scale) change. By building detectors for this more evolved model, a local operator can detect both translation and scale change to recover depth cues. Using these detectors in the earliest vision layer should yield superior optical flow.

While recovering depth information from an image sequence can only be found up to a scale factor, the absolute time to collision can be recovered from the same cues. To this end, we define the following image formation model.

A single oriented slit aperture for a perspective image formation model can be parametrized with an aperture angle, α and the number of pixels N that are visible on the image plane within that aperture.

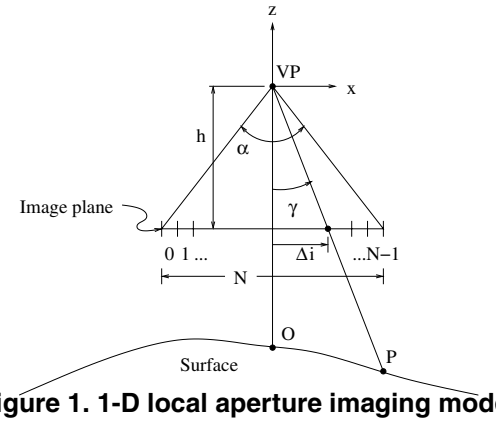


Figure 1. 1-D local aperture imaging model.

As shown in Figure 1, the *fixation point* O is the point on the surface in the center of the aperture's field of view, i.e. view angle $\gamma = 0$. γ ranges in value from $-\alpha/2$ to $\alpha/2$, with positive γ to the right (positive x direction).

For any view angle γ formed between the $VP - O$ and $VP - P$ lines within $-\alpha/2$ and $\alpha/2$, there is a corresponding value for the image plane coordinate x that can be projected into one of the pixel coordinates between 0 and $N-1$,

$$i \quad = \quad \left(\frac{N-1}{2}\right) + \frac{N}{2}\frac{\tan(\gamma)}{\tan\left(\frac{\alpha}{2}\right)} \; . \qquad (5)$$

And, conversely, the image coordinate i can be mapped back to the view angle γ,

$$\gamma \quad = \quad \tan^{-1}\left[\frac{2i-N+1}{N}\tan\left(\frac{\alpha}{2}\right)\right] \; . \qquad (6)$$

This camera model will be used to map surface points to image coordinates to build the correspondence matrix $\mathbf{H}$ with a scene structure model, described next.

Two views of a 1-D cross section of a surface can be locally modeled using 5 parameters. The surface shape along one direction can be characterized by a curvature K, a normal vector $\vec{N}$ and distance from the viewpoint, d, shown in Figure 2. Distance d scales all lengths of the diagram, so it is factored out to a canonical representation with unit distance between first viewpoint VP and the fixation point on the surface 0. The surface normal vector $\vec{N}$ at 0 is encoded by the angle η with respect to the first view axis $VP - 0$. The curvature of the canonical surface becomes $k = Kd$.

The motion model is chosen to minimize image deformation due to translation, defining the second viewpoint VP' at a given distance δ at an angle β from the first view axis $VP - 0$. VP' is fixated on point Q on the surface, a view rotation Ω away from the first fixation point 0.

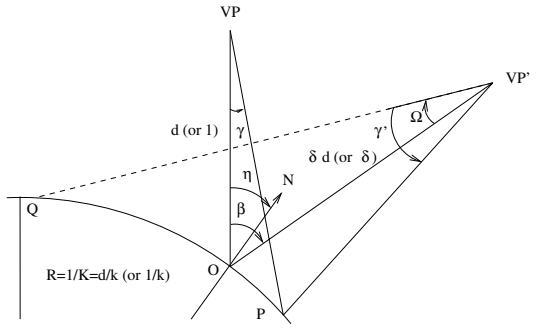


Figure 2. 1-D local aperture section model.

This 1-D cross section shape and motion model can be aligned with multiple orientations θ at each position in the image plane to reconstruct 3-D surface shape and motion.

Some experimental observations[2], beyond the scope of this paper, revealed that the motion parameters Ω and δ are not separable, but can be found jointly; together they produce distinct motion fields. These can be considered first order structure from motion parameters. In contrast, the view angle change β and the shape parameters k and η were observed to be confounded. Even in apertures of $N = 32$ pixels, there is very little difference in the motion field to distinguish different shapes from a single aperture. These 3 parameters require either neighborhoods of support of curvature consistency between neighbors or a global solution. This extra work would qualify the parameters β, k, η as second order structure from motion parameters, and will not be recovered for the purposes of this paper.

For image pairs of interest, the views will overlap in the camera aperture of angle α (shown in Figure 1), restricting the useful range of Ω to about $(-\alpha/2, +\alpha/2)$, and typically, $\alpha < 10°$. With these angles so small, Ω will be linearly proportional to the translation along the image slit axis. The parameter δ is the reciprocal of image scale change,

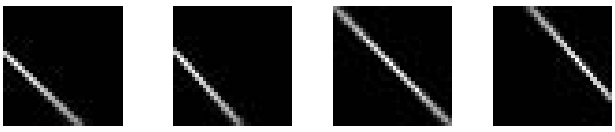
$$\delta \quad = \quad \frac{\|VP' - O\|}{\|VP - O\|} \; . \qquad (7)$$

Recovering Ω and δ locally in forward time can be augmented by recovering Ω' and δ' by reversing the sequence of the images. A direct method of computing the time to collision between two images separated by a delay of Δt is:

$$T \quad = \quad \frac{\Delta t}{2}\left(\frac{\delta}{1-\delta} + \frac{1}{\delta' - 1}\right) \; . \qquad (8)$$

Calculating the time to collision requires recovering the δ and Ω parameters. Figure 3 illustrates some of the unique $\mathbf{H}$ for specific instances of model parameters $(\Omega, \delta, k, \eta, \beta)$. The rows are the index into the first image $\mathcal{I}$, and the columns are the index into the second image $\mathcal{I}'$. Ω shifts the white curve horizontally, and δ affects its slope. This paper

aims to recover $\mathbf{H}$ given image pair $(\mathcal{I}, \mathcal{I}')$. Section 3 details the general theory for constructing image domain operators to detect their characteristic image deformations.



$\Omega = -1.8°$ $\delta = 0.9487$ | $\Omega = -1.8°$ $\delta = 1.1111$ | $\Omega = 0.0°$ $\delta = 1.0$ | $\Omega = 1.20°$ $\delta = 1.1111$

Figure 3. Correspondence matrices $\mathbf{H}_{\Omega,\delta}$.

3. Theory

The image structure from two images separated by time can be encoded by a correspondence matrix $\mathbf{H}$. This section will show that the Singular Value Decomposition of $\mathbf{H}$ leads to a unique set of image domain operators that can code for specific scene events as encoded by $\mathbf{H}$.

The correspondence matrix $\mathbf{H}$ relates the elements of a first image $\mathcal{I}$ and a second image $\mathcal{I}'$. The images are first normalized as $\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'$ for a zero mean intensity and a contrast of 1 by finding the image's brightness $\mu_{\mathcal{I}}$ and contrast $\Delta_{\mathcal{I}}$.

$$\begin{aligned} \mu_{\mathcal{I}} &\triangleq \frac{\sum_i \mathcal{I}_i + \sum_i \mathcal{I}'_i}{2N} , \\ \Delta_{\mathcal{I}} &\triangleq \frac{\max_i(|\mathcal{I}_i - \mu_{\mathcal{I}}|, |\mathcal{I}'_i - \mu_{\mathcal{I}}|)}{\mu_{\mathcal{I}}} \subset (0,1) , \end{aligned}$$

$$\tilde{\mathcal{I}} = \frac{\mathcal{I} - \mu_{\mathcal{I}}}{\mu_{\mathcal{I}}\Delta_{\mathcal{I}}} \quad , \quad \tilde{\mathcal{I}}' = \frac{\mathcal{I}' - \mu_{\mathcal{I}}}{\mu_{\mathcal{I}}\Delta_{\mathcal{I}}} . \tag{9}$$

3.1. Image Mapping H

Image correspondence between normalized images $(\tilde{\mathcal{I}}, \tilde{\mathcal{I}}')$ is determined by $\mathbf{H}$, but not necessarily all information contained in $\tilde{\mathcal{I}}$ is present in $\tilde{\mathcal{I}}'$ nor vice versa. This means, for example, that some elements or pixels of $\tilde{\mathcal{I}}$ do not come from $\tilde{\mathcal{I}}'$ but come instead from some other source.

The image correspondence constraints are in the form of a weighted sum of elements in the complementary image:

$$\begin{aligned} \left(\sum_j \mathbf{H}_{ij}\right)\tilde{\mathcal{I}}_i &= \sum_j \mathbf{H}_{ij}\tilde{\mathcal{I}}'_j \quad , \forall i \in \{1...N\} , \\ \left(\sum_i \mathbf{H}_{ij}\right)\tilde{\mathcal{I}}'_j &= \sum_i \mathbf{H}_{ij}\tilde{\mathcal{I}}_i \quad , \forall j \in \{1...N\} . \end{aligned} \tag{10}$$

The image correspondence equations can be re-written:

$$\mathbf{S}\tilde{\mathcal{I}} = \mathbf{H}\tilde{\mathcal{I}}' \quad , \quad \mathbf{S}'\tilde{\mathcal{I}}' = \mathbf{H}^T\tilde{\mathcal{I}} . \tag{11}$$

$$\begin{aligned} \mathbf{S} &= \begin{bmatrix} \sum_j \mathbf{H}_{1j} & & & 0 \\ & \sum_j \mathbf{H}_{2j} & & \\ & & \ddots & \\ 0 & & & \sum_j \mathbf{H}_{Nj} \end{bmatrix} \\ \mathbf{S}' &= \begin{bmatrix} \sum_i \mathbf{H}_{i1} & & & 0 \\ & \sum_i \mathbf{H}_{i2} & & \\ & & \ddots & \\ 0 & & & \sum_i \mathbf{H}_{iN} \end{bmatrix} \end{aligned} \tag{12}$$

Equation Set 11 deals automatically with non-shared information: note that elements in one space that have no correspondent in the other space will have a zero row sum of $\mathbf{H}$ (element of $\mathbf{S}$) or zero column sum of $\mathbf{H}$ (element of $\mathbf{S}'$).

3.2. Operator Synthesis

Given an image pair $(\tilde{\mathcal{I}}, \tilde{\mathcal{I}}')$, the goal is to find the $\mathbf{H}$ that best describes their deformation and identify that $\mathbf{H}$'s model parameters. In order to recognise the image events for a given correspondence matrix, there needs to be a transport function to map between the two views. The image pair $(\tilde{\mathcal{I}}, \tilde{\mathcal{I}}')$ can be represented as two different transforms of a common texture vector $\vec{w}$. In order for the elements of texture vector $\vec{w}$ to be independent and linear, the image synthesis functions must be:

$$\underset{N\times 1}{\tilde{\mathcal{I}}} \approx \underset{N\times NN\times 1}{\mathbf{A}\,\vec{w}} \quad , \quad \underset{N\times 1}{\tilde{\mathcal{I}}'} \approx \underset{N\times NN\times 1}{\mathbf{B}\,\vec{w}} . \tag{13}$$

where $\mathbf{A}$ is orthonormal and $\mathbf{B}$ is also orthonormal. $\mathbf{A}$ and $\mathbf{B}$ are not expected to be exact solutions, because the image formation process does not lend itself to a direct linear mapping between the image pair. Image domain deformation caused by perspective projections of 3-D objects, for example, confounds the surface texture signal and the geometric deformation into one image signal, and the two components are generally not separable. $\mathbf{A}$ and $\mathbf{B}$ are, however, the best linear approximation to the geometric deformation image correspondence in the sense of minimizing some error.

The texture information shared by the two images via $\mathbf{H}$ must exist in the space spanned by $\mathbf{H}$, and thus can be expressed compactly in r independent coefficients, where $r = \mathrm{rank}(\mathbf{H})$. The first r coefficients of $\vec{w}$ encode the stationary signal shared between $\tilde{\mathcal{I}}$ and $\tilde{\mathcal{I}}'$ that are expressed through $\mathbf{H}$. The remaining $N - r$ coefficients encode the non-shared image signals mapped into the null space of $\mathbf{H}$.

Because $\mathbf{A}$ and $\mathbf{B}$ are chosen to be orthonormal and square, they are invertible (by transposition). This means that the texture vector $\vec{w}$ can be recovered from images $(\tilde{\mathcal{I}}, \tilde{\mathcal{I}}')$ knowing the correspondence matrix $\mathbf{H}$.

$$\mathbf{A}^T\mathbf{A}\hat{w} = \mathbf{A}^T\tilde{\mathcal{I}} \quad \Rightarrow \quad \hat{w} = \mathbf{A}^T\tilde{\mathcal{I}} \tag{14}$$

$$\mathbf{B}^T\mathbf{B}\hat{w}' = \mathbf{B}^T\tilde{\mathcal{I}}' \quad \Rightarrow \quad \hat{w}' = \mathbf{B}^T\tilde{\mathcal{I}}' \tag{15}$$

3.3. Solution via Singular Value Decomposition

Substituting Equation set 13 into Equation set 11,

$$\begin{aligned} \mathbf{S}\mathbf{A}\vec{w} &\approx \mathbf{H}\mathbf{B}\vec{w} \quad , \forall \vec{w} \\ \mathbf{S}'\mathbf{B}\vec{w} &\approx \mathbf{H}^T\mathbf{A}\vec{w} \quad , \forall \vec{w} \, . \end{aligned} \tag{16}$$

The optimization problem to solve for $\mathbf{A}$ and $\mathbf{B}$ is thus:

$$\underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{S}\mathbf{A}\vec{w} - \mathbf{H}\mathbf{B}\vec{w}\| \, , \; \underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{S}'\mathbf{B}\vec{w} - \mathbf{H}^T\mathbf{A}\vec{w}\| \, . \tag{17}$$

Because the matrices $\mathbf{A}$ and $\mathbf{B}$ are expected to work independently of the surface texture encoded in $\vec{w}$, the optimization problem can be expressed as

$$\underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{S}\mathbf{A} - \mathbf{H}\mathbf{B}\| \, , \; \underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{S}'\mathbf{B} - \mathbf{H}^T\mathbf{A}\| \, . \tag{18}$$

The minimization terms can be recombined as

$$\underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{S}\mathbf{A}\mathbf{B}^T - \mathbf{H}\| \, , \; \underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{S}'\mathbf{B}\mathbf{A}^T - \mathbf{H}^T\| \, . \tag{19}$$

This can be visualized more clearly by substituting $\mathbf{H}$ with its Singular Value Decomposition,

$$\underset{N\times N}{\mathbf{H}} = \underset{N\times N}{\mathbf{U}} \underset{N\times N}{\mathbf{\Sigma}} \underset{N\times N}{\mathbf{V}^T} \, . \tag{20}$$

where $\mathbf{U}$ is the matrix whose columns are the left-hand eigenvectors of $\mathbf{H}$ and the columns of $\mathbf{V}$ are the right-hand eigenvectors. That is, the columns of $\mathbf{U}$ are the eigenvectors of $\mathbf{H}\mathbf{H}^T$, and the columns of $\mathbf{V}$ are the eigenvectors of $\mathbf{H}^T\mathbf{H}$, both sorted in decreasing order of eigenvalues. The singular values of $\mathbf{H}$ are the elements of the diagonal matrix $\mathbf{\Sigma}$, which are the square root of the eigenvalues from $\mathbf{H}\mathbf{H}^T$ or $\mathbf{H}^T\mathbf{H}$. Substituting the SVD, one must solve for:

$$\underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{S}\mathbf{A}\mathbf{B}^T - \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\| \, , \; \underset{\mathbf{A},\mathbf{B}}{\operatorname{argmin}} \|\mathbf{A}\mathbf{B}^T\mathbf{S}' - \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\| \, . \tag{21}$$

The remainder of this proof builds $\mathbf{A}$ and $\mathbf{B}$ one column at a time, using successively better approximations of $\mathbf{H}$. The k^{th} order approximation, $\mathbf{H}_k$ uses the first k columns of $\mathbf{U}$ and $\mathbf{V}$ and the first k diagonal elements of Σ.

To solve for unknown orthogonal matrices $\mathbf{A}$ and $\mathbf{B}$, consider the 1^{st} order approximation of $\mathbf{H}$, $\mathbf{H}_1 = \sigma_1 \vec{u}_1 \vec{v}_1^T$.

$$\mathbf{S}\mathbf{A}_1\mathbf{B}_1^T \approx \sigma_1 \vec{u}_1 \vec{v}_1^T \quad , \quad \mathbf{A}_1\mathbf{B}_1^T\mathbf{S}' \approx \sigma_1 \vec{u}_1 \vec{v}_1^T \tag{22}$$

The only choice for $\mathbf{A}_1$ and $\mathbf{B}_1$ that spans the same space as $\vec{u}_1$ and $\vec{v}_1$, satisfying both constraining equations is

$$\mathbf{A}_1 = \left[\vec{u}_1, \vec{0}, \ldots, \vec{0}\right] \quad , \quad \mathbf{B}_1 = \left[\vec{v}_1, \vec{0}, \ldots, \vec{0}\right] \, . \tag{23}$$

An inductive proof introduces higher k^{th} order approximations $\mathbf{H}_k$ to solve for the corresponding $\mathbf{A}_k$ and $\mathbf{B}_k$.

$$\mathbf{S}\mathbf{A}_k\mathbf{B}_k^T \approx \sum_{i=1}^{k} \sigma_i \vec{u}_i \vec{v}_i^T \tag{24}$$

$$\mathbf{S}\mathbf{A}_{k-1}\mathbf{B}_{k-1}^T \approx \sum_{i=1}^{k-1} \sigma_i \vec{u}_i \vec{v}_i^T \tag{25}$$

$$\mathbf{S}(\mathbf{A}_k\mathbf{B}_k^T - \mathbf{A}_{k-1}\mathbf{B}_{k-1}^T) \approx \sigma_k \vec{u}_k \vec{v}_k^T \, . \tag{26}$$

Similarly,

$$(\mathbf{A}_k\mathbf{B}_k^T - \mathbf{A}_{k-1}\mathbf{B}_{k-1}^T)\mathbf{S}' \approx \sigma_k \vec{u}_k \vec{v}_k^T \, . \tag{27}$$

This implies that $\mathbf{A}_k$ is $\mathbf{A}_{k-1}$ plus an orthogonal component in the direction of $\vec{u}_k$ and that $\mathbf{B}_k$ is $\mathbf{B}_{k-1}$ plus an orthogonal component in the direction of $\vec{v}_k$. Therefore, the k^{th} column of $\mathbf{A}$ is the k^{th} column of $\mathbf{U}$ and the k^{th} column of $\mathbf{B}$ is the k^{th} column of $\mathbf{V}$. Therefore, the least squares error solution for Equation Set 16, satisfying full rank and orthonormality of $\mathbf{A}$ and $\mathbf{B}$ is to substitute

$$\mathbf{A} = \mathbf{U} \quad , \quad \mathbf{B} = \mathbf{V}. \tag{28}$$

Informally, the SVD expresses the correspondence matrix $\mathbf{H}$ as a linear remapping $\mathbf{U}^T$ from $\tilde{\mathcal{I}}$ to the same space as a linear remapping $\mathbf{V}^T$ from $\tilde{\mathcal{I}}'$. The SVD of $\mathbf{H}$ optimizes the spectral representation of the transform between a pair of data sets (images), maximizing the independence between channels (elements of $\vec{w}$), ranked according to significance in the sense of minimizing least squares error.

3.4. Distance to Data Metric

The maximum likelihood hypothesis $\mathbf{H}$ for an image pair $(\tilde{\mathcal{I}}, \tilde{\mathcal{I}}')$ minimizes the residual error $\vec{r}$ between the input images and their reconstructions predicted by $\mathbf{H}$. Determine the feature vector $\hat{w}$ that uses k^{th} order approximations $\mathbf{U_k}$ and $\mathbf{V_k}$ to represent the image vectors $\tilde{\mathcal{I}}$ and $\tilde{\mathcal{I}}'$.

$$\hat{w} = \left[\; \mathbf{U_k}^T/\sqrt{2} \; \vdots \; \mathbf{V_k}^T/\sqrt{2} \; \right] \begin{bmatrix} \tilde{\mathcal{I}} \\ \cdots \\ \tilde{\mathcal{I}}' \end{bmatrix} \tag{29}$$

The feature vector $\hat{w}$ is now the best parameterization for the image pair assuming deformation $\mathbf{H}$.

The residual error can be computed by projecting the feature vector back into the image space. If the assumed deformation $\mathbf{H}$ is sufficiently close to the scene geometry, then residual signal error $\vec{r}$, the difference between the original image signal and the reconstructed image signal will be low.

$$\vec{r} = \left(\begin{bmatrix} \tilde{\mathcal{I}} \\ \cdots \\ \tilde{\mathcal{I}}' \end{bmatrix} - \begin{bmatrix} \mathbf{U_k} \\ \cdots \\ \mathbf{V_k} \end{bmatrix} \hat{w} \right) \Big/ \left\| \begin{bmatrix} \tilde{\mathcal{I}} \\ \cdots \\ \tilde{\mathcal{I}}' \end{bmatrix} \right\| \tag{30}$$

The likelihood of correspondence $\mathbf{H}$ given evidence $(\tilde{\mathcal{I}}, \tilde{\mathcal{I}}')$ can be expressed as a function $\mathcal{L}\left(\mathbf{H}|\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'\right)$.

$$\mathcal{L}\left(\mathbf{H}|\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'\right) \quad \triangleq \quad e^{-\|\vec{r}\|} \subset (0, 1] \tag{31}$$

The uncertainty of the maximum likelihood choice can be expressed as the entropy h of the likelihoods for all the different hypotheses.

$$h = \frac{-\sum_{i=1}^{n} \mathcal{L}\left(\mathbf{H}_i|\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'\right) \log\left(\mathcal{L}\left(\mathbf{H}_i|\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'\right)\right)}{\log(n)} \subset (0, 1] \tag{32}$$

To summarize the procedure for parametric recovery,

Preparation (off-line)

1. Discretize parameter space $\mathbf{M}$ into n representative parameter vectors $\vec{\mathbf{m}}_i$, $i \in \{1, \ldots, n\}$.
2. Synthesize a correspondence matrix $\mathbf{H}_i$ to represent the coordinate mapping $\mathcal{M}(\vec{\mathbf{m}}_i)$.
3. Synthesize detectors for $\mathbf{H}_i$, using SVD.

Usage (on-line)

1. Using the detectors for $\mathbf{H}_i$, given the observations $\tilde{\mathcal{I}}$ and $\tilde{\mathcal{I}}'$, compute $\mathcal{L}\left(\mathbf{H}_i|\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'\right)$.
2. Find the maximum likelihood $\mathbf{H}_i|\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'$, and set the maximum likelihood $\vec{\mathbf{m}}^* = \vec{\mathbf{m}}_i$.
3. Use the distribution of $\mathcal{L}\left(\mathbf{H}_i|\tilde{\mathcal{I}}, \tilde{\mathcal{I}}'\right)$ to find the entropy h of the maximum likelihood choice.

4. Experiments on Time to Collision

4.1. Synthesis of Operators

The model space of $(\Omega, \delta, k, \eta, \beta)$ was discretized into 19 levels for each of Ω and δ, and 5 levels each for k, η, β and α, binning the correspondence matrices into mean correspondence matrices indexed by (Ω, δ) for a total of 361 distinct correspondence matrices. Applying the procedures of Section 2, 361 new receptive field banks are synthesized by retaining the first 16 filter pairs of each, illustrated in Figure 4. Each row i of the upper grayscale images at time t_0 is a deformed sinusoid which corresponds to another deformed sinusoid at time $t_1 = t_0 + \Delta t$, in the corresponding row i in the lower grayscale images. The first 8 detectors are shown.

Note that the most significant optimal detectors for this motion model are windowed sinusoids concentrated in the lower spatial frequencies. The SVD

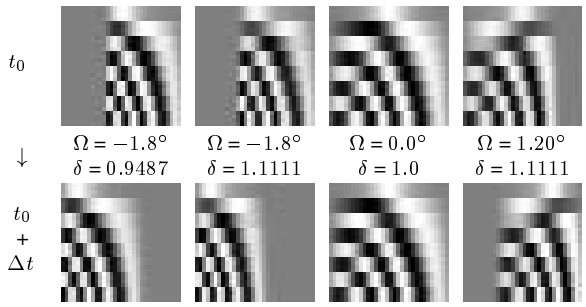


Figure 4. U, V of correspondence matrices.

automatically discovered the appropriate spatial scales for detecting each deformation. The parameters used are:

Number of pixels:	N	64
View translation $\angle$:	$\Omega[i]$	$2.4° \left(\frac{i-9}{9}\right), \quad i \in \{0, .., 18\}$
Distance to surface:	$\delta[i]$	$1.23456^{\left(\frac{i-9}{9}\right)}, \quad i \in \{0, .., 18\}$
Surface normal $\angle$:	$\eta[i]$	$\{-45°, -22.5°, \ldots, 45°\}$
View change $\angle$:	$\beta[i]$	$\{-10°, -5°, 0°, 5°, 10°\}$
Surface curvature:	$k[i]$	$\{-4, -2, 0, 2, 4\}$
View aperture $\angle$:	$\alpha[i]$	$\{8°, 8.5°, 9°, 9.5°, 10°\}$

These experiments were performed using slits of length $N = 64$ with widths of 32 pixels. The filter banks are applied to a 32×32 gridding of the image pair at 6 different orientations θ $(0°, 30°, 60°, 90°, 120°, 150°)$. The same procedure is applied by swapping the two images to recover the maximum likelihood for the time reversal, $(\Omega', \delta', \theta')$.

Ω and Ω' are useful for optical flow, but the local depth (or time to collision) cues are only available through the δ and δ' measures. Ω and δ are found jointly, and the usefulness of δ is sensitive to the correct compensation of Ω.

Some neighborhood filtering is required, since δ is noisy. The maximum likelihood values for $\Omega, \delta, \Omega', \delta'$ in the grid of 32×32 elements are median filtered in a 3×3 neighborhood for Ω, Ω', and a 7×7 neighborhood for δ, δ'.

4.2. Synthetic Images, 4 Boxes

To test the Time to Collision detectors, a random dot texture was combined with synthetic range data shown in Figure 5 to generate the synthetic image pair shown in Figure 6. The surface consisted of 4 quadrants of varying distance (450, 500, 550 and 600mm from the camera focal point), and the camera moves by 50mm towards the center of the scene. The ground truth time to collision for the four quadrants are thus 8, 9, 10 and 11 units of time in the future.

The camera geometry is modeled as a pinhole perspective camera with 320×240 square pixels where 320 pixels are mapped onto a viewing angle of $46.31°$. This means that the image samples of 64 pixels will cover aperture angles α varying from $10°$ at the center to $8°$ at the periphery.

The recovered Ω, Ω' data is almost perfectly planar for each orientation, as shown in Figure 7. This is expected, as the scene motion generates a radial flow field. Ω is proportional to the image plane velocity, with linearly increasing magnitude as it moves further from the focus of expansion. Note that the Ω parameter is signed and directional.

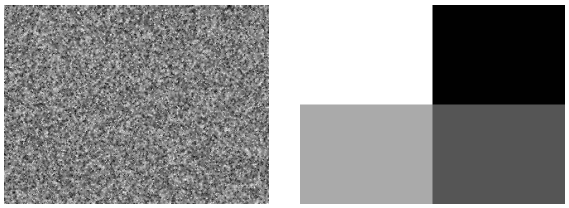


Figure 5. Synthetic texture and range. Black is 450mm from focal point, white is 600 mm.

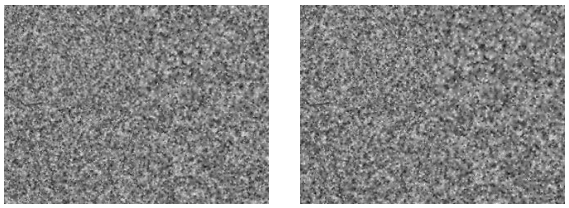


Figure 6. Synthetic image pair. The camera moves 50mm toward the center of the scene.

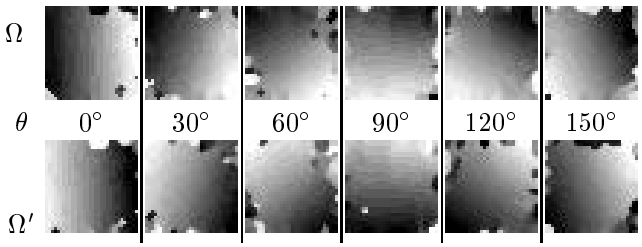


Figure 7. Recovered Ω and Ω' for 6 orientations at 32×32 image locations (black: $-2.4°$, gray: $0°$, white: $+2.4°$).

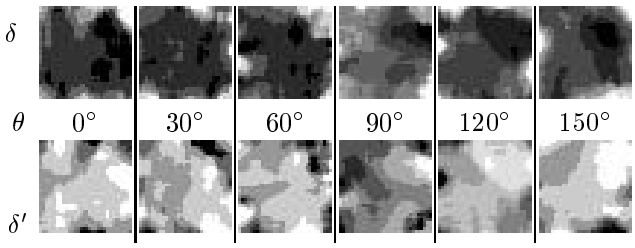


Figure 8. Recovered δ and δ' for 6 orientations at 32×32 image locations (black: 0.8100, gray: 1.0, white: 1.2346.

Since Ω, Ω' have been cleanly recovered, it is not surprising that the more sensitive δ, δ' have formed blobs roughly in the four quadrants of the scene, shown in Figure 8.

The local structure from motion problem is at its worst in image sequences with a forward-moving camera. Optical flow methods have no information at or near the focus of expansion, and small errors in the flow field even toward the periphery render depth recovery impractical without fitting a global model to the optical flow field. With this experiment, the time to collision detectors are designed specifically to work best at or near the focus of expansion.

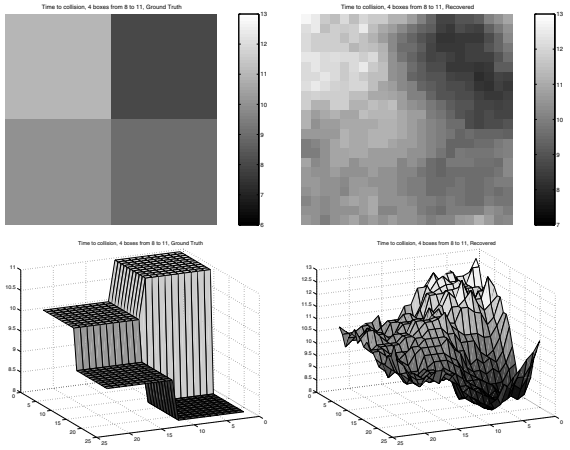


Figure 9. Time to Collision, ground truth versus recovered.

After observing the images of Figure 6 separated by 50mm and one unit of time, the receptive fields are able to detect the separate, yet tightly clustered, collision events that are 8, 9, 10 and 11 units of time in the future. With this in mind, the resulting time to collision results are reasonably close to the ground truth, shown side by side in Figure 9.

4.3. Natural images, Calibration Grid

The experiment was repeated using grayscale images captured by a camera mounted on a gantry robot looking at a flat planar calibration grid. The camera field of view is the same as the earlier synthetic example, but it is not an idealized pinhole camera. The camera lens starts at 500mm from the plane, and moves 50mm closer, for a time to collision of 9 units of time, show in Figure 10. The camera optics slightly bend the grid lines; the small squares toward the periphery are a bit smaller than the squares near the center. This spherical aberration makes the grid appear as a textured sphere shown close up, directly in front of the camera.

The camera's focal length is 7mm, placing the focal point about 15mm behind the lens from which camera distance was measured.

The detector response in Figure 11 is so sensitive that the recovered time to collision captures the spherical aberration effect. The time to collision varies from 11 units of time at the center to 8 units of time at the periphery. The aberration

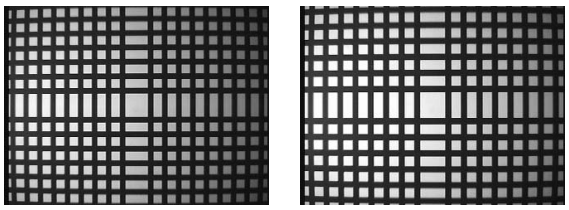


Figure 10. Planar calibration grid scene.

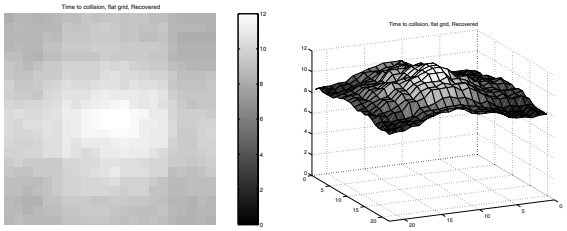


Figure 11. Time to collision, planar grid.

makes the periphery appear further than it actually is, moving faster than the center, hence a lower time to collision.

In this case, optical flow in the periphery is uninformative; there appears to be no translation, but there is texture expansion. This experimental result confirms that the local feed-forward operators can extract precise estimates of the real time to collision without a global scene motion model.

5. Conclusion

The direct recovery of some shape and motion parameters from image sequences is now possible with image-domain appearance, without a need for a global motion solution. By populating a synthetic retina with specialized receptive fields, the vision system can estimate time to collision with the obstacles in the scene.

A new framework for operator synthesis was introduced that constructed scene shape and motion detectors from scene motion models in a principled way, automatically discovering appropriate spatial scales. This technique can be applied to many domains, automatically finding optimal image-domain transfer functions between two images or two spaces.

Image events characterized by $\mathbf{H}$ lead directly to synthetic local feed-forward operators, and naturally produce uncertainty measures or confidence intervals on the recovered model parameters.

Preliminary results indicate that these purely local feed-forward operators perform reasonably accurate recovery of scene structure without the need to regularize.

References

[1] J. Barron and R. Eagleson. Motion and structure from time-varying optical flow. In *Vision Interface*, pages 104–111, May 1995.

[2] S. Benoit. *Towards Direct Motion and Shape Parameter Recovery from Image Sequences*. PhD thesis, McGill University, 2003.

[3] D. DiFranco and S. Kang. Is appearance-based structure from motion viable? In *2nd Int. Conference on 3-D Digital Imaging and Modeling*, Ottawa, Canada, Oct. 1999.

[4] S. Fejes and L. Davis. Direction-selective filters for ego-motion estimation. Technical Report CS-TR-3814,CAR-TR-865, University of Maryland at College Park, 1997.

[5] S. Fejes and L. Davis. Exploring visual motion using projections of motion fields. In *Proceedings of the ARPA Image Understanding Workshop*, pages 113–122, May 1997.

[6] S. Fejes and L. Davis. What can projections of flow fields tell us about the visual motion. In *ICCV*, 1998.

[7] G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE PAMI*, 20(10):1025–1039, 1998.

[8] J. Heel. Direct estimation of structure and motion from multiple frames. Technical Report AI Memo 1190, MIT AI Lab, March 1990.

[9] B. Krse, N. Vlassis, R. Bunschoten, and Y. Motomura. Feature selection for appearance-based robot localization. In *Proceedings 2000 RWC Symposium*, 2000.

[10] M. W. M. Ben-Ezra, S. Peleg. Real-time motion analysis with linear-programming. In *Proc. ICCV*, volume 2, page 703, Corfu, Greece, September 1999.

[11] R. Manmatha. Measuring the affine transform using gaussian filters. In *Proc. of the ECCV (2)*, pages 159–164, 1994.

[12] C. Nastar, B. Moghaddam, and A. Pentland. Generalized image matching: Statistical learning of physically-based deformations. In *Proc. of the Fourth European Conference on Computer Vision (ECCV'96*, Cambridge, UK, April 1996.

[13] J. Oliensis. Direct multi–frame structure from motion for hand–held cameras. In *ICPR Vol. I*, pages 889–895, 2000.

[14] S. Roy and I. J. Cox. Motion without structure. In *13th Int. Conference on Pattern Recognition, Vol. I*, pages 728–734, Vienna, Austria, August 1996. IEEE.

[15] S. Soatto. Observability/identifiability of rigid motion under perspective projection. Technical Report CIT-CDS 94-001, California Institute of Technology, Jan 1994.

[16] S. Soatto and P. Perona. Dynamic visual motion estimation from subspace constraints. Technical Report CIT-CDS 94-006, California Inst. of Tech., Pasadena, CA, Jan. 1994.

[17] S. Soatto and P. Perona. On the exact linearization of structure from motion. Technical Report CIT-CDS 94-011, California Institute of Technology, Pasadena, CA, May 1994.

[18] G. P. Stein and A. Shashua. Direct methods for estimation of structure and motion from three views. Technical Report AIM-1594, MIT AI Lab, Nov. 1996.

[19] C. Stiller and J. Konrad. Eigentransforms for region-based image processing. In *Proc. Int. Conf. on Consumer Electronics*, pages 286–287, Chicago, IL, USA, June 1995.

[20] C.-K. Tang and G. G. Medioni. Robust estimation of curvature information from noisy 3d data for shape description. In *Proceedings of the ICCV (1)*, pages 426–433, 1999.

[21] C. Tomasi. Input redundancy and output observability in the analysis of visual motion. In *Proc. Sixth Symposium on Robotics Research*, pages 213–222. MIT Press, 1993.

[22] Y. Yacoob and L. S. Davis. Learned models for estimation of rigid and articulated human motion from stationary or moving camera. *IJCV*, 36(1):5–30, 2000.