Optical Flow Using Overlapped Basis Functions for Solving Global Motion Problems

Sridhar Srinivasan and Rama Chellappa *

Department of Electrical Engineering and Center for Automation Research
University of Maryland, College Park, MD 20742, U.S.A.
{shridhar,rama}@cfar.umd.edu

Abstract. Motion problems in which the scene motion largely conforms to a low order global motion model are called global motion problems, examples of which are stabilization, mosaicking and motion super-resolution. In this paper, we propose a two-step solution for robustly estimating the global motion parameters that characterize global motion problems. Our primary contribution is an improved estimation algorithm for modeling the optical flow field of a sequence using overlapped basis functions. Moreover, we show that the parametrized flow estimates can be consolidated through an iterative process that estimates global deformation while ensuring robustness to systematic errors such as those caused by moving foreground objects or occlusion. We demonstrate the validity of our model and accuracy of the algorithm on synthetic and real data. Our technique is computationally efficient, and is ideally suited for the application areas discussed here, *viz.* stabilization, mosaicking and super-resolution.

1 Introduction

Motion problems in which the scene motion largely conforms to a global motion model are termed global motion problems. Electronic stabilization of video, creating mosaics from image sequences and performing motion super-resolution are examples of global motion problems. These are often encountered in surveillance, navigation (tele-operation), automatic target recognition (ATR) and forensic science. Reliable motion estimation is critical to these tasks, which is particularly challenging when the sequences display random as well as highly structured systematic errors. The former is usually a result of sensor noise, atmospheric turbulence, etc. while the latter is caused by occlusion, shadows and independently moving foreground objects. The goal in global motion problems is to maintain the integrity of the solution in the presence of both types of errors.

Prior work on global motion problems can be classified into *flow based, direct* and *feature based* approaches. A low-order model is fit to the dense flow

* The support of this research by the Defense Advanced Research Projects Agency (DARPA Order No. C635) and the Office of Naval Research under Contract N00014-95-1-0521 is gratefully acknowledged. We also thank the anonymous reviewers for their valuable comments and criticism.

field estimated from the sequence in the flow based approach [1], [4], [11]. Direct methods estimate the motion parameters from the original sequence in one step [2], [13], [18], while feature based methods match sparse feature sets extracted from each image for registering successive images [16], [25]. The shortcoming of flow based methods is that estimating a dense flow field is both ill-conditioned and an overkill for the global motion problem, besides complicating the process of consolidating the flow field into global motion parameters. On the other hand, estimating the motion parameters directly from the luminance data saves computations at the cost of failure in the presence of structured errors. Finally, feature based methods suffer due to their excessive dependence on reliable feature extraction and correspondence.

In this paper, we propose a two step algorithm for solving the global motion problem. The first and critical step of the algorithm is based on a new optical flow estimator which models the flow field as a linear combination of an overlapped set of basis functions. Consolidation of the flow field into global motion parameters, which is the second step, is performed in the parameter (feature) space, like in the feature based methods. While the flow field model is similar to spline based motion computation [23], the estimation process is very different. Indeed, we show that our estimate of the flow field parameters is relatively unaffected by errors in the estimates of spatial gradients. Moreover, we develop a computationally efficient strategy for single-scale motion estimation. Since the consolidation process operates on the model weights which are far fewer in number than the image size, the dimensionality of the problem is small. This permits robust and efficient computation of the global motion parameters. In practice, we have achieved rates of over 5 frame/s on a sequential workstation for image stabilization.

This paper is organized as follows: section 2 introduces the optical flow model, an estimation methodology together with a mathematical justification for its robustness and an efficient algorithm for its solution. Consolidation of the flow model is covered in section 3. Section 4 describes our experiments and tabulates the results obtained by our algorithm.

2 Optical Flow Modeling and Estimation

When the projected time-varying image field of a scene is given by $\psi = \psi(x, y, t)$, preservation of luminance patterns implies the *gradient constraint equation*

$$\frac{\partial \psi}{\partial t} + u\frac{\partial \psi}{\partial x} + v\frac{\partial \psi}{\partial y} = 0 \qquad \forall\, x, y, t \ . \tag{1}$$

u and v in (1) denote the horizontal and vertical velocities, as functions of space and time, respectively. Together, they constitute the optical flow field of the sequence. For every triplet (x, y, t) in (1), there are two unknowns, making the problem of computing the optical flow ill-conditioned.

The problem of estimating a pixelwise flow field has been approached by regularization of the optical flow constraint [3], [6], [8]-[10], [14], [15], [17]. In

these methods, the process of estimating optical flow is converted to a nonlinear optimization problem, although (1) is linear in the velocity field (u, v). Results obtained by these techniques are often significantly off the mark, highly sensitive to numerical precision, determined only at a sparse set of points with reasonable confidence and carry with them the pitfalls associated with nonlinear optimization [6].

Our proposed technique models the optical flow field as a linear combination of an overlapped set of basis functions that is solved by a linear system of equations. This system is sparse and its solution involves a sparse matrix inversion. A numerically stable solution is obtained by using the method of conjugate gradients, giving a dense and reliable flow field even when the matrix is ill conditioned.

An alternative to computing the optical flow on a pixelwise basis is to model the motion fields u and v in terms of a weighted sum of basis functions and to estimate the weights which constitute the flow model parameters [20], [22], [23]. In this approach, the motion field is force-fitted to the model and derives the smoothness properties from the model basis functions. Let $\{\phi = \phi(x, y, t)\}$ be a family of basis functions, and let the flow field be modeled as

$$u = \sum_{k=0}^{K} u_k \phi_k \text{ and } v = \sum_{k=0}^{K} v_k \phi_k \ . \tag{2}$$

The patchwise constant, affine or polynomial model is a special case of (2), for instance setting ϕ_k to a rectangular window function is equivalent to the constant-in-a-patch model. Likewise, choosing a wavelet basis for $\{\phi_k\}$ is tantamount to performing a multiresolution optical flow computation. Since the optical flow field of a sequence is largely smooth, it seems reasonable to model the field using (2) and an appropriate basis $\{\phi_k\}$.

Substituting (2) into (1), we get

$$\frac{\partial \psi}{\partial t} + \sum_k u_k \phi_k \frac{\partial \psi}{\partial x} + \sum_k v_k \phi_k \frac{\partial \psi}{\partial y} = 0 \quad \forall x, y, t \ . \tag{3}$$

This continuum of equations in 3-space (3) can be reduced to a scalar equation for each instant of time by integrating with a multiplicative kernel $\theta = \theta(x, y)$,

$$\int \frac{\partial \psi}{\partial t} \theta dx dy + \sum_k u_k \int \phi_k \frac{\partial \psi}{\partial x} \theta dx dy + \sum_k v_k \int \phi_k \frac{\partial \psi}{\partial y} \theta dx dy = 0 \ . \tag{4}$$

Equation (4) exists for every square integrable kernel θ, and every instant of time. In order to solve for $\{u_k, v_k\}$, it is necessary to choose appropriate kernels in (4).

2.1 Solutions

System (3) is linear in the unknowns $\{u_k, v_k\}$ and is analogous to the matrix-vector system

$$Ax \rightarrow b \quad A \in \mathbb{R}^{M \times N}, M > N \ , \tag{5}$$

where x corresponds to the vector $(u_0, v_0, \ldots)'$. The analogy implies the applicability of solutions and results of (5) to (3). In the discrete domain, the analogy is obvious since an equation of the type (3) exists for each pixel in the current frame, corresponding to one row of the composite matrix $[A|b]$. The least squares (LS) solution of (5) is given by $A'Ax = A'b$. Likewise, choosing θ from the family $\{\phi_k \frac{\partial \psi}{\partial x}, \phi_k \frac{\partial \psi}{\partial y}\}$ gives the LS solution of (3). In practice, only discretized data is available for the image luminance field ψ. The LS solution assumes knowledge of the spatial derivatives of ψ, which may not be known reliably. Any minor and random non-compliance of (1) is accounted by the observation error in b. A robust approach must try to minimize sensitive dependence of the solution on the spatial as well as temporal derivatives. In other words, in the analogue (5), the solution must be accurate and robust to errors in A as well as in b, which can be stated as:

Assume x_0 is the exact solution for the overconstrained linear system $Ax \rightarrow b$. Let Δ and δ be zero mean, independent additive observation noise in A and b respectively, i.e. the quantities $\hat{A} = A + \Delta$ and $\hat{b} = b + \delta$ are observed. Find an 'optimal' estimate x of x_0 given $\hat{A}$ and $\hat{b}$.

In the remainder of this section, we analyze three solutions *viz.* the LS, total least squares (TLS) and our proposed method we refer to as the extended least squares (ELS), in terms of their optimality quantified by the bias and variance of estimates. We assume Δ and δ to be uncorrelated with A and b, and the matrix $A'A$ to be invertible. In addition, we assume the availability of the observation $\hat{A}'A$ for the ELS estimate, the existence of which we prove later. The symbol $\approx$ denotes a first order approximation. R_Δ denotes the covariance matrix of rows of Δ and r_δ is the variance of δ_i, assuming that the rows are *iid.*

LS solution. The LS solution of (5) is $x_{\mathrm{LS}} = (\hat{A}\hat{A}')^{-1}\hat{A}'b$. The error associated with this solution can be shown to be

$$e_{\mathrm{LS}} \approx (A'A)^{-1}A'(\delta - \Delta x_0) - \underbrace{(A'A)^{-1}A'R_\Delta x_{\mathrm{LS}}}_{\text{bias}} . \tag{6}$$

x_{LS} is the minimizer of $\|Ax - b\|_2$ when the observation error is present only in $\hat{b}$. Bias sets in when $\hat{A} \neq A$. The two random components of e_{LS} are independent by the assumption of independence of Δ and δ. A modification of LS, the corrected least squares (CLS) method removes bias while preserving the error covariance. However, this assumes knowledge of the covariance matrix R_Δ.

TLS solution. The TLS principle has emerged as an alternative to LS since it is capable of handling errors in potentially all observations in a linear system, not merely on the 'right hand side' [12]. The TLS solution is obtained by minimally perturbing the composite observation matrix $[\hat{A}|\hat{b}]$ to reduce its rank to N. x_{TLS}, the TLS estimate, is given by $x_{\mathrm{TLS}} = -v_{n+1}^{(n)}/v_{n+1,n+1}$, where $v_{n+1}^{(n)}$ is the vector formed by the first n components of v_{n+1}, which is the eigenvector of $[\hat{A}|\hat{b}]'[\hat{A}|\hat{b}]$

corresponding to its smallest eigenvalue μ. $v_{n+1,n+1}$ is the $n+1$th component of v_{n+1}.

e_{TLS} is not easy to determine for the TLS case, even when the observation errors are small. Setting $E[e_{\mathrm{TLS}}] = 0$ gives a necessary condition for a zero bias estimate

$$(r_\delta I - R_\Delta)x_0 \to 0 \ . \tag{7}$$

Thus, the TLS estimate is unbiased *only if* the error in estimating the temporal gradient is equal in variance to the error in estimating the (windowed) spatial gradient. However, in typical image sequence problems, the temporal gradient is calculated over a smaller number of frames than the spread of the spatial gradient operator. In order to satisfy (7), it becomes necessary to discard useful information by narrowing down the support of the spatial gradient operator, which is not desirable. In addition, it has been argued in [12] that the covariance of an unbiased TLS estimate is larger than that of the LS estimate, in the first order approximation as well as in simulations. There is, therefore, no fundamental gain in choosing the TLS over the LS solution.

ELS solution. Neither the LS nor the TLS solutions of (5) are unbiased in the general case, and the CLS solution shows sensitive dependence on the errors in $\hat{A}$ as well as in $\hat{b}$. Interestingly, if an additional observation, *viz.* $\hat{G} = \hat{A}'A$ is available, the ELS solution proposed here shows no dependence on the error Δ in the observation of A. Moreover, the observation corresponding to $\hat{G}$ is available for (3). The ELS solution of (5) is the solution of $\hat{G}x = \hat{A}'\hat{b}$,

$$x_{\mathrm{ELS}} = \hat{G}^{-1}\hat{A}'\hat{b} \ . \tag{8}$$

A linearized analysis shows the corresponding estimation error to be

$$e_{\mathrm{ELS}} = \underbrace{\hat{G}^{-1}}_{\approx (A'A)^{-1}} [\Delta' \underbrace{(Ax_0 - b)}_{=0} + \underbrace{\Delta'\delta}_{O(2)} + A'\delta] \approx (A'A)^{-1}A'\delta \ , \tag{9}$$

proving the claim. The subtle difference between x_{LS} and x_{ELS} leads to the replacement of the error term $-(A'A)^{-1}A'\Delta x_0$ in (6) by $\hat{G}^{-1}\Delta'\delta$ in (9). Assuming small observation errors, the latter is insignificant. Therefore, x_{ELS} is unbiased, and has a smaller covariance than x_{LS} or x_{CLS}. In the original problem (3), the ELS solution is obtained when θ in (4) is chosen from the family $\{\phi_k \frac{\partial\hat{\psi}}{\partial x}, \phi_k \frac{\partial\hat{\psi}}{\partial y}\}$ where the quantity $\frac{\partial\hat{\psi}}{\partial\{x,y\}}$ is an *estimate* of the derivative, giving

$$\begin{aligned}
\int \frac{\partial\hat{\psi}}{\partial t}\phi_l\frac{\partial\hat{\psi}}{\partial x} + \sum_k u_k \int \phi_k\frac{\partial\psi}{\partial x}\phi_l\frac{\partial\hat{\psi}}{\partial x} + \sum_k v_k \int \phi_k\frac{\partial\psi}{\partial y}\phi_l\frac{\partial\hat{\psi}}{\partial x} = 0 \\
\int \frac{\partial\hat{\psi}}{\partial t}\phi_l\frac{\partial\hat{\psi}}{\partial y} + \sum_k u_k \int \phi_k\frac{\partial\psi}{\partial x}\phi_l\frac{\partial\hat{\psi}}{\partial y} + \sum_k v_k \int \phi_k\frac{\partial\psi}{\partial y}\phi_l\frac{\partial\hat{\psi}}{\partial y} = 0
\end{aligned} \tag{10}$$

where the integrals are over XY and the estimated temporal derivative is $\frac{\partial\hat{\psi}}{\partial t}$. We now prove the availability of the observations $\hat{G} = \hat{A}'A$ and $\hat{A}'\hat{b}$, which is

equivalent to proving the computability of the integrals in (10), under certain weak assumptions on the functional form of ϕ_k and the estimate $\frac{\partial\hat{\psi}}{\partial\{x,y\}}$.

2.2 Eliminating Derivatives

Consider the integral $I(y) = \int \phi_k \frac{\partial\psi}{\partial x}\phi_l \frac{\partial\hat{\psi}}{\partial x}dx$. Assume that the estimate $\frac{\partial\hat{\psi}}{\partial\{x,y\}}$ has a differentiable functional form, *i.e.* the derivative $\frac{\partial}{\partial\{x,y\}}\frac{\partial\hat{\psi}}{\partial\{x,y\}}$ is known exactly. This holds for even the simplest of discrete gradient masks like $(-1,1)$ since the masks assume a smooth underlying functional form. Also, assume that $\{\phi_k\}$ are differentiable and that $\phi_k(x,y) \to 0$ as $x \to \pm\infty$ or $y \to \pm\infty$. Integrating $I(y)$ by parts over $(-\infty,\infty)$, we get

$$I(y) = \underbrace{\left[\phi_k\phi_l\frac{\partial\hat{\psi}}{\partial x}\psi\right]_{-\infty}^{\infty}}_{=0} - \int_{-\infty}^{\infty}\psi\frac{\partial\phi_k\phi_l\frac{\partial\hat{\psi}}{\partial x}}{\partial x}dx \tag{11}$$

which is computable reliably without knowing the exact derivatives $\frac{\partial\psi}{\partial\{x,y\}}$. Applying this reasoning to (10) gives

$$\begin{aligned}
\sum_k u_k \int \frac{\partial\phi_k\phi_l\frac{\partial\hat{\psi}}{\partial x}}{\partial x}\psi + \sum_k v_k \int \frac{\partial\phi_k\phi_l\frac{\partial\hat{\psi}}{\partial x}}{\partial y}\psi &= \int \frac{\partial\hat{\psi}}{\partial t}\phi_l\frac{\partial\hat{\psi}}{\partial x} \\
\sum_k u_k \int \frac{\partial\phi_k\phi_l\frac{\partial\hat{\psi}}{\partial y}}{\partial x}\psi + \sum_k v_k \int \frac{\partial\phi_k\phi_l\frac{\partial\hat{\psi}}{\partial y}}{\partial y}\psi &= \int \frac{\partial\hat{\psi}}{\partial t}\phi_l\frac{\partial\hat{\psi}}{\partial y}
\end{aligned} \tag{12}$$

with the following desirable properties:

- The accuracy of spatio-temporal image derivatives is not critical to the accuracy of computation.
- The computed image flow is force-fitted on a model. The only conditions on the model are that it be space-limited and differentiable.
- With finite extent basis functions ϕ_k, the system of equations gives a sparse, banded matrix structure.

It is only by eliminating derivatives that the ELS method is achievable. In practice, the gains of using ELS are eroded because of the following reasons. We have implicitly assumed that noise is introduced only on differentiation while in reality even the image data ψ is noisy. Also, the integrals in (12) are evaluated from discretely sampled data and are not true integrals. Although the difference between $\hat{G}$ and $\hat{A}'\hat{A}$ is small, it is sufficient to prove the empirical superiority of the ELS.

2.3 Choice of Basis Functions

As in all estimation problems, there is an inevitable tradeoff between sensitivity and selectivity, or equivalently, between accuracy and localization. The specific basis function family $\{\phi_k\}$ determines the shape of the ROC curve, and its density (defined as the number of basis functions per unit area) fixes the operating point on the curve. Not all differentiable, compactly supported curves are meaningful to the problem at hand. Since the variations in optical flow are typically high-frequency and localized or low-frequency and global, periodic bases do not offer any advantage. Orthogonal bases do not lead to a simpler solution than non-orthogonal bases because there are no integrals of the form $\int \phi_k \phi_l$. However, for every non-overlapping pair $\{\phi_i, \phi_j\}$, the corresponding entries in the observation matrix $\hat{G}$, $\hat{G}_{ij}$ and $\hat{G}_{ji}$, are zero. Thus, from the computational aspect, it is desirable to minimize the number of overlapping pairs. Besides, since systematic errors show spatial concentration, a locally supported basis function set ensures that the damage caused by a systematic error is limited to one or a few model weights determining the flow field.

The reasoning thus far, including the final linear system (12), holds for any choice of basis functions which are differentiable and which decay to zero. In the remainder of this section, we place certain additional restrictions on the choice of $\{\phi_k\}$ that ensure computational ease and have intuitive appeal for modeling a motion field. We construct $\{\phi_k\}$ from translations of a prototype function ϕ_0 along a uniformly spaced square grid of spacing w, with the following additional requirements:

- *Separability:* $\phi_0(x, y) = \phi_0(x)\phi_0(y)$
- *Symmetry about the origin:* $\phi_0(x) = \phi_0(-x)$
- *Peak at the origin:* $|\phi_0(x)| \leq \phi_0(0) = 1$
- *Compact support:* $\phi_0(x) = 0 \quad \forall |x| > w$
- *Constancy:* $\phi_0(x) + \phi_0(x - w) \simeq 1, \forall x \in [0, w]$
- *Linearity:* $\phi_0(x) + \lambda\phi_0(x - w) \simeq 1 + (\lambda - 1)\frac{x}{w}\ \forall x \in [0, w]$

Compact support ensures that each basis function overlaps with exactly nine of its neighbors, in the cardinal and diagonal directions. *Constancy* is essential for modeling the simple case of uniform translation, and *linearity* is necessary for exactly modeling an affine flow field. *Constancy* is implied by *linearity.*

The spacing w of the grid at which the basis functions are centered determines the density of the basis functions, thereby determining the position of the algorithm on the ROC curve. A larger spacing implies a larger support for ϕ_k which reduces error in estimates of the integrals in (12). However, increasing w means that the parameter space spans fewer dimensions, giving fewer degrees of freedom for modeling the flow field. Trying to fit a stiff model leads to non-compliance at motion discontinuities. Therefore, w determines the tradeoff between robustness and accuracy of the algorithm.

2.4 Sparse Matrix Inversion

In our experiments, we used the cosine window

$$\phi_0(x) = \frac{1}{2}\left[1 + \cos(\frac{\pi x}{w})\right] \quad x \in [-w, w] \tag{13}$$

as the prototype basis function. The entire bases is generated from shifts of the prototype along a rectangular grid with spacing w. The *linearity* property is only approximately satisfied for this choice. The observation matrix $\hat{G}$ is block tridiagonal, given by

$$\hat{G} = \begin{bmatrix} D_1 & U_1 & 0 & 0 & \\ L_2 & D_2 & U_2 & 0 & \cdots \\ 0 & L_3 & D_3 & U_3 & \\ & \vdots & & & \ddots \end{bmatrix}, \qquad D_i, U_i, L_i \sim \begin{bmatrix} \times & \times & 0 & 0 & \\ \times & \times & \times & 0 & \cdots \\ 0 & \times & \times & \times & \\ & \vdots & & & \ddots \end{bmatrix} \tag{14}$$

where each of the submatrices D_i, U_i and L_i are in turn block tridiagonal, and $\times$ denotes a 2 by 2 submatrix with data dependent coefficients. In addition, $\hat{G}$ is block diagonal dominant, *almost* symmetric and *almost* positive semidefinite[1]. In order to solve (12), we employ the method of *Preconditioned Biconjugate Gradients (PBCG)* [5], [19]. The structure of $\hat{G}$ allows for a good choice of useful preconditioners, one of which is the matrix $\tilde{G}$ formed by the symmetric component of the diagonal 2 x 2 submatrices of $\hat{G}$. In effect, $\tilde{G}$ is the component of $\hat{G}$ comprised purely of within-grid interactions.

3 Consolidation

With a good choice of local basis functions, the proposed flow algorithm yields model parameters $\{u_k, v_k\}$ that largely conform to the global motion model. Parameters corresponding to areas showing systematic errors like moving foreground regions show large deviations from the global model. In order to compute the global motion model parameters, it is necessary to simultaneously segment out the "good" parameters. This is achieved by consolidation.

Since there may exist foreground areas moving quite differently from the background, the distribution of flow velocities, even when estimated without error, may be multimodal. Model parameter estimation involves locating the fundamental mode and its membership - a robust estimation problem with no mathematically concise solution. Also, although the optical flow is computed over the whole image, its reliability is local gradient dependent. Areas which have large gradients are typically, though not always, associated with more reliable flow estimates. In the first pass, grids showing significant gradient content are picked out as "reliable" and the flow estimates at these grid points are combined in a least squares framework to give a set of model parameters. These parameters are recomputed after rejecting a fraction of the data points with the worst fit.

[1] Since $\hat{G} = \hat{A}'A \approx A'A$.

This step is repeated a few times giving the final global motion model parameters. The above process is known as iterative weighted least squares.

While pruning the flow field model coefficients used for computing the global motion parameters, the angular error measure employed in [6] is used. Assume that the true and computed flows at a point (x, y) in a particular frame are $(u_0, v_0)'$ and $(u, v)'$ respectively. Define vectors $\mathbf{v}_0 = (u_0, v_0, 1)'$ and $\mathbf{v} = (u, v, 1)'$. The error angle ϵ at (x, y) is given by $\epsilon = \arccos\left(\frac{\mathbf{v}_0 . \mathbf{v}}{\|\mathbf{v}_0\|\|\mathbf{v}\|}\right)$. ϵ is insensitive to the magnitude of the motion vector and offers a normalized measure against which a range of velocities can be compared meaningfully.

4 Results

In this paper, we claim that the proposed technique is well suited to the class of global motion problems. The first step in proving our claim is to demonstrate that the optical flow algorithm proposed here does indeed model and estimate optical flow accurately. This is covered in the next section, followed by experimental results on the global motion problems of stabilization, mosaicking and super-resolution in the following sections.

4.1 Performance of Optical Flow Algorithm

For demonstrating the quantitative performance of the proposed optical flow method, we perform a series of experiments on synthetic data along the lines of [6]. Exact flows are known for this data, which consists of the *sinusoid, square, translating tree, diverging tree* and *Yosemite* sequences. *Sinusoid* is a uniformly translating modulated 2D pattern with a wavelength of 6 pixels. *Square* is the simple case of a white square moving against a dark background. The *tree* sequences are synthesized from realistic data simulating camera translation with respect to the 2D scene. *Yosemite* is a 2D rendering of a 3D model, with motion discontinuities introduced along the edges of the mountains. Moreover, the clouds in this sequence translate uniformly across the image while undergoing a steady luminance change. In addition, there is significant aliasing near the lower portions of the image making this sequence particularly challenging.

First, we verify the validity of the optical flow field model by fitting the known correct flows of the sequences to the model using a least squares estimator. Computed model parameters are used to generate flow fields and the mean error metric $\bar{\epsilon}$ is calculated between the original flow field and its model. This is indicated in Tables 1 and 2 as the *Perfect Model Estimate.*

Computed flows for *sinusoid* and *square* sequences, suitably downsampled and rescaled, are shown in Fig. 1. The error metric ϵ for these scenes is tabulated, together with the standard deviation $\bar{\sigma}_\epsilon$, which indicates the spread of errors over the image frame, in Tables 1 and 2. For a comparison, the error values of the conventional optical flow techniques obtained from [6] are shown in the tables as well. Since the subspace flow method gives a dense flow estimate, the comparison with alternative optical flow techniques is limited to those yielding a dense flow

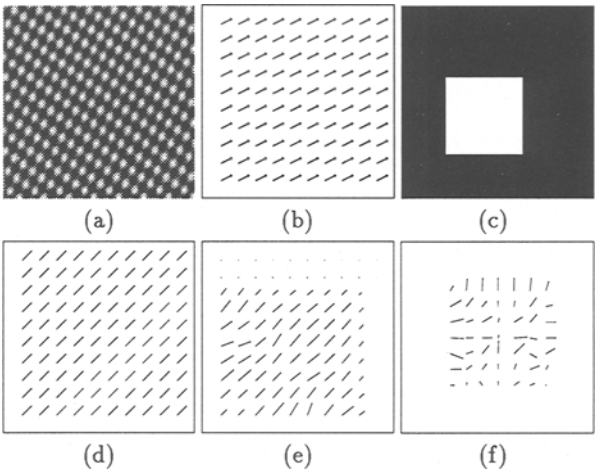

Fig. 1. *Sinusoid* sequence: (a) center frame, (b) computed flow (visually identical to true flow). *Square* sequence: (c) center frame (d) true flow, computed flow using (e) proposed algorithm and (f) Anandan's algorithm.

field. Even among these, many algorithms estimate optical flow only within an inscribed region whose margin width is determined by the spread of the gradient operator, while the proposed technique gives a corner to corner flow estimate.

For the *Sinusoid* data set (Fig. 1 (a)-(b), Table 1), the proposed technique ranks after Fleet and Jepson's. The latter benefits greatly from the sharp frequency spectrum of the input sinusoid, while Singh's method which is comparable in accuracy to ours. The proposed technique is the best for *Square* (Fig. 1 (c)-(f), table 1) by a wide margin. For this sequence, Fleet and Jepson's algorithm is unable to produce a 100% dense flow field, and $\bar{\epsilon}$ of Singh's algorithm is greater than twice $\bar{\epsilon}$ of the proposed. Interestingly, while the "true" flow field of *Square* has been deemed to be a uniform translation across the image (Fig. 1 (d)), it is a moot point whether the background is indeed moving along with the white square. The output of our algorithm (Fig. 1 (e)) shows mainly the square in motion, which is an interpretation as credible as the "true" flow. On the other hand, Anandan's algorithm which ranks second in Table 1, computes flow only for approximately half the image area due to border effects, and the computed flow pattern is visually meaningless (Fig. 1 (f)). Our algorithm performs very well for the synthetic data sets *diverging tree, translating tree* and *Yosemite* (Table 2). It ranks at the top for *translating tree* and *Yosemite* and in the second place for *diverging tree*. For the latter sequence, the flow can be made more accurate by changing w. Indeed, $\bar{\epsilon} \approx 2.1$ when w is decreased from 32 to 16 for this case, making the proposed algorithm the best in the list. However, we have chosen to compare the performance of our algorithm with no hand-adjustment of parameters whatsoever. Note that the error standard deviation $\bar{\sigma}$ is significantly

Table 1. Performance of optical flow algorithm on *sinusoid* and *square* data

Experiment	Sinusoid		Square	
	$\bar{\epsilon}$	$\bar{\sigma}$	$\bar{\epsilon}$	$\bar{\sigma}$
Horn & Schunck (original)	4.19	0.50	47.21	14.60
Horn & Schunck (mod.)	2.55	0.59	32.81	13.67
Lucas & Kanade	2.47	0.16	×	×
Uras *et. al.*	2.59	0.71	×	×
Nagel	2.55	0.93	34.57	14.38
Anandan	30.80	5.45	31.46	18.31
Singh ($n = w = 2, N = 2$)	2.24	0.02	49.03	21.38
Singh ($n = w = 2, N = 4$)	91.71	0.04	45.16	21.10
Fleet & Jepson ($\tau = 1.25$)	0.03	0.01	×	×
Perfect Model Estimate	0.10	0.08	0.12	0.09
Proposed	2.46	0.03	20.71	22.11

smaller than $\bar{\sigma}$ of the other techniques. The results indicate that the proposed technique is consistently accurate over the range of synthetic imagery.

The difference between using the ELS and LS solutions is less dramatic, although the ELS consistently shows a smaller $\bar{\epsilon}$. The difference is largest for *square* whose ELS and LS estimate errors are 20.714 and 20.765 respectively. For the other sequences, the differences are in the third decimal place.

Table 2. Performance of proposed algorithm on *translating tree*, *diverging tree* and *Yosemite* sequences

Experiment	Trans. Tree		Diverg. Tree		Yosemite	
	$\bar{\epsilon}$	$\bar{\sigma}$	$\bar{\epsilon}$	$\bar{\sigma}$	$\bar{\epsilon}$	$\bar{\sigma}$
Horn and Schunck (original)	38.72	27.67	12.02	11.72	31.69	31.18
Horn and Schunck (modified)	2.02	2.27	2.55	3.67	9.78	16.19
Uras *et. al.* (unthresholded)	0.62	0.52	4.64	3.48	8.94	15.61
Nagel	2.44	3.06	2.94	3.23	10.22	16.51
Anandan	4.54	3.10	7.64	4.96	13.36	15.64
Singh (step 1, $n = 2$, $w = 2$)	1.64	2.44	17.66	14.25	15.28	19.61
Singh (step 2, $n = 2$, $w = 2$)	1.25	3.29	8.60	5.60	10.44	13.94
Perfect Model Estimate	0.09	0.09	1.40	1.04	3.03	6.48
Proposed	0.61	0.26	2.94	1.64	8.94	10.63

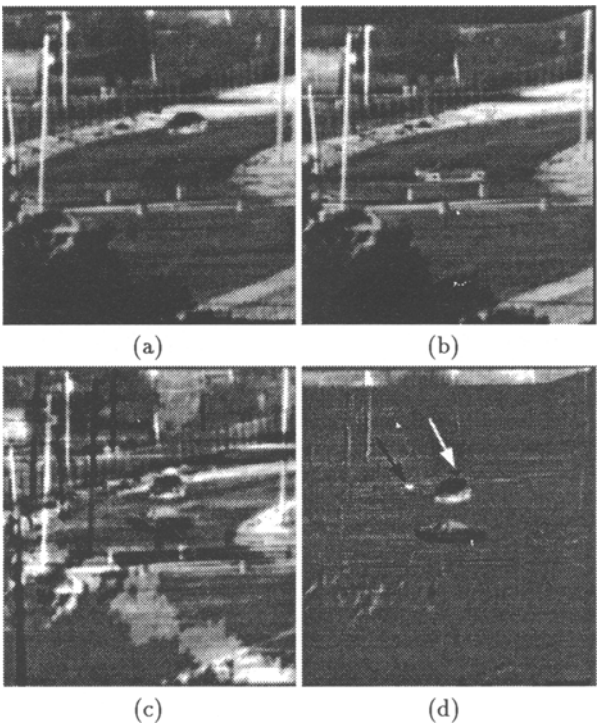

Fig. 2. *TI car* sequence: (a) frame 1, (b) stabilized frame 42, (c) difference between unstabilized frames, and (d) difference between stabilized frames

4.2 Stabilization

We present two sequences, reflecting disparate operating conditions, for demonstrating the performance of the stabilization algorithm. The first sequence is a dynamic scene comprising a car, a cyclist, two pedestrians and ruffling foliage. This scene, shown in Fig. 2, is imaged by an infra-red camera fixated on the cyclist. It is impossible to locate the cyclist without stabilizing for camera motion. The camera undergoes panning with no rotation about the optical axis, and no translation. The first and forty-second frames are shown in Figs. 2 (a) and (b), and the difference between these frames without and with stabilization is shown in Figs. 2 (c) and (d) respectively. The effect of stabilization in locating moving objects, especially the cyclist and pedestrians is obvious in the figure. Notice that the cyclist shows up in the stabilized frame difference as a dark and a bright spot near the car (corresponding to his/her position in the first and last frames respectively), indicated by arrows.

The second image sequence reflects a navigation scenario where a forward looking camera is mounted on a moving platform. The platform motion involves translation and rotation about all three axes. The camera has a wide field of

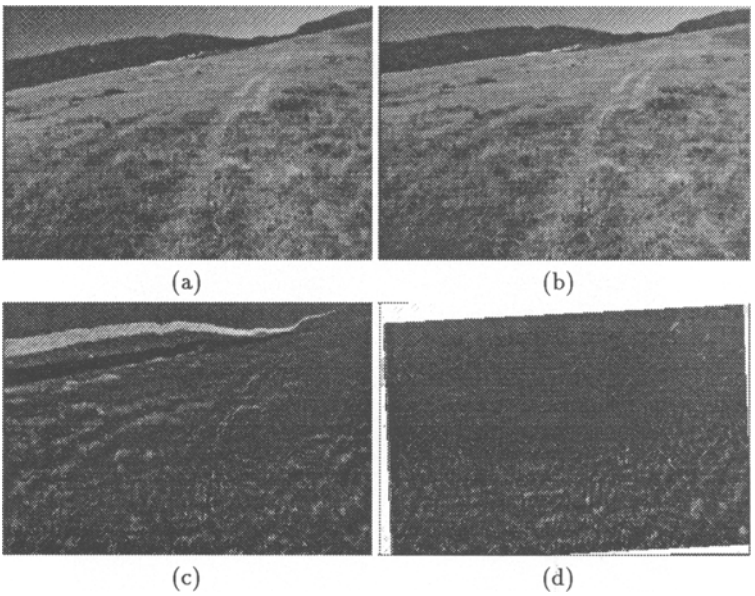

Fig. 3. *Radius* sequence: (a) frame 3, (b) frame 20, (c) unstabilized difference, and (d) stabilized difference between frame 20 and frame 3

view, and this implies a large coverage of the foreground as well. The third and twentieth frames are shown in Figs. 3 (a) and (b). Unstabilized and stabilized frame differences shown in Figs. 3 (c) and (d) demonstrate the ability of the algorithm to compensate for unwanted motion. The six parameter affine motion model is used for both these examples.

4.3 Mosaicking

We consider two cases of generating a wide angle view by forming a mosaic from (*i*) a panning camera and (*ii*) a camera translating along the optical axis. While the former leads to a topologically correct reconstruction, the latter can be, at best, a qualitative approximation of a true wide angle view.

The first case, presented in Fig. 4, is an interesting application of mosaicking. The input sequence is a zoomed and panned shot of a moving bus which nearly fills a third to a half of the image area. While constructing the mosaic, only newly appearing areas are included, and the center one-third band is ignored in the consolidation process. The clearance between the front of the bus and the left boundary of the image is sufficient to ensure that data is available to reconstruct the background despite severe occlusion by the moving bus. Although apriori knowledge is incorporated by deleting the central horizontal band, portions of the moving bus often spill out into the "useful" area. Fig. 4 (d) shows the reconstructed mosaic from the first 128 frames, registered to the coordinate

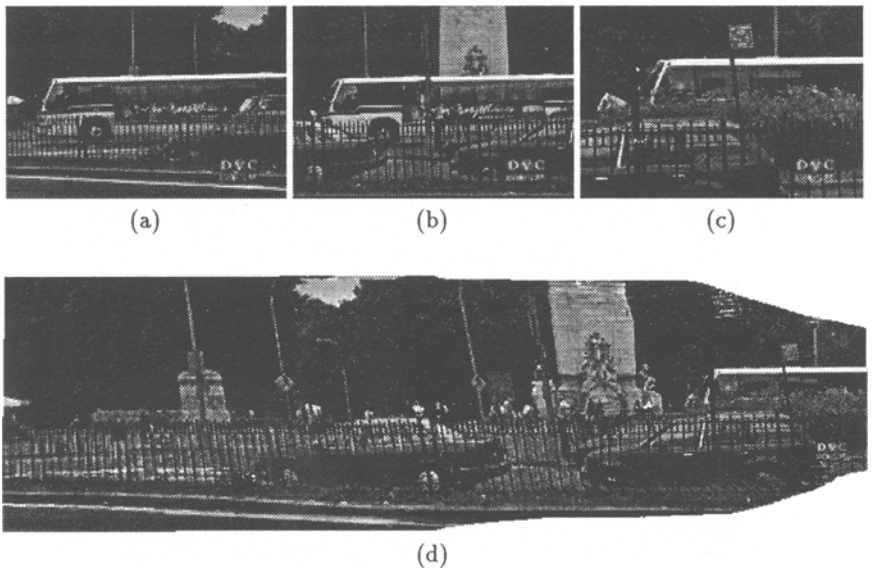

Fig. 4. *Bus* sequence: (a) frame 125, (b) frame 63, (c) frame 0, (d) mosaic using 120 frames

system of the first frame. Zooming is evident from the shape of the envelope of the mosaic. Since our method is based on frame-to-frame registration, errors tend to propogate. This can be seen from the slight slant in the central portion of the mosaic. It must be borne in mind that since the slant is built up over around 60 frames, the per-frame erroneous warping is miniscule, and mosaicking is accurate.

The second example shown in Fig. 5 is a scene richly structured in 3D. The dominant motion of the camera is translation along the optical axis. It is impossible to reconstruct the true wide angle view since there exist regions in the scene that would be imaged by wide angled optics but do not appear in the sequence. Nevertheless, this situation arises commonly in teleoperation of robotic vehicles. It is desirable to present contextual information to the teleoperator in the form of snippets from previous scenes that are not visible in the current scene. This effect can be realized by building a mosaic using a low degree of freedom global motion model on the image sequence. In our experiment, we used 36 frames of a sequence whose first and last frames are shown in Fig. 5 (a) and (b) respectively. Since the current frame is used as the reference, the mosaic construction proceeds in the opposite sense to the previous example, with the latest view overwriting the previous views. The qualitative mosaic reconstructed by our algorithm is presented in Fig. 5 (c). We have used the six parameter affine model for the two examples of mosaicking.

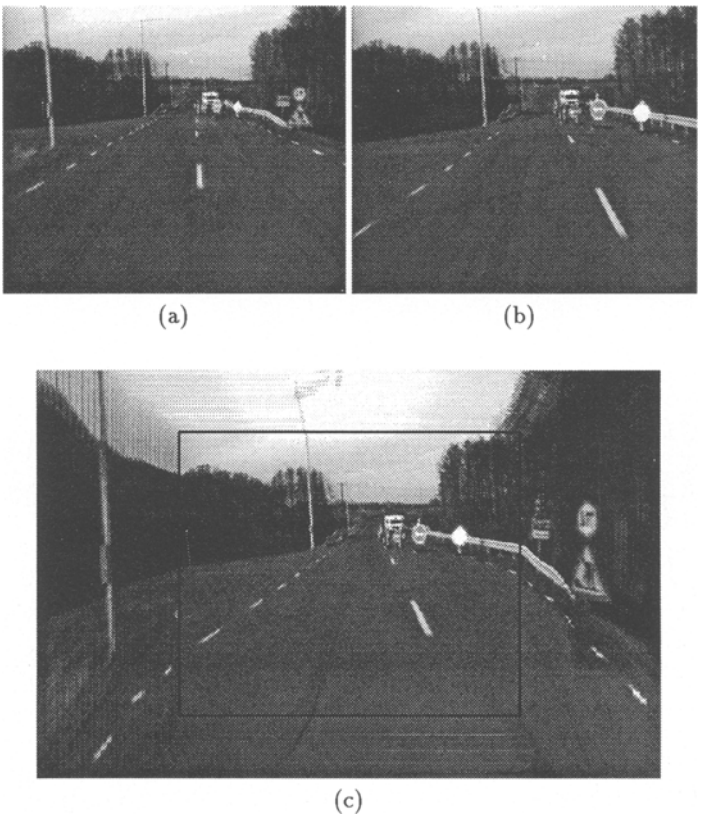

Fig. 5. *Saab* sequence: (a) frame 1, (b) frame 36, (c) qualitative mosaic using 36 frames

4.4 Motion Super-resolution

The last experiment we present here demonstrates the ability of our algorithm to perform super-resolution. An image sequence gathered by an aerial platform was regenerated from an MPEG compressed bitstream. The first frame is shown in Fig. 6 (a). Vehicles on the highway at the bottom of the image are at most few pixels in size and move with velocities < 1 pixel/frame. In order to identify these moving vehicles, we build a $2\times$ super-resolved mosaic of 20 frames (shown in Fig. 6 (c)). Temporal filtering eliminates the foreground objects, which are in turn recovered by deleting the computed background from the images. Fig. 6 (b) shows the moving vehicles as dark spots. Without the ability to resolve at the sub-pixel level, the targets are liable to be lost in the quantization noise of lossy MPEG encoding. The algorithm is resistant to compression artifacts while producing accurate results.

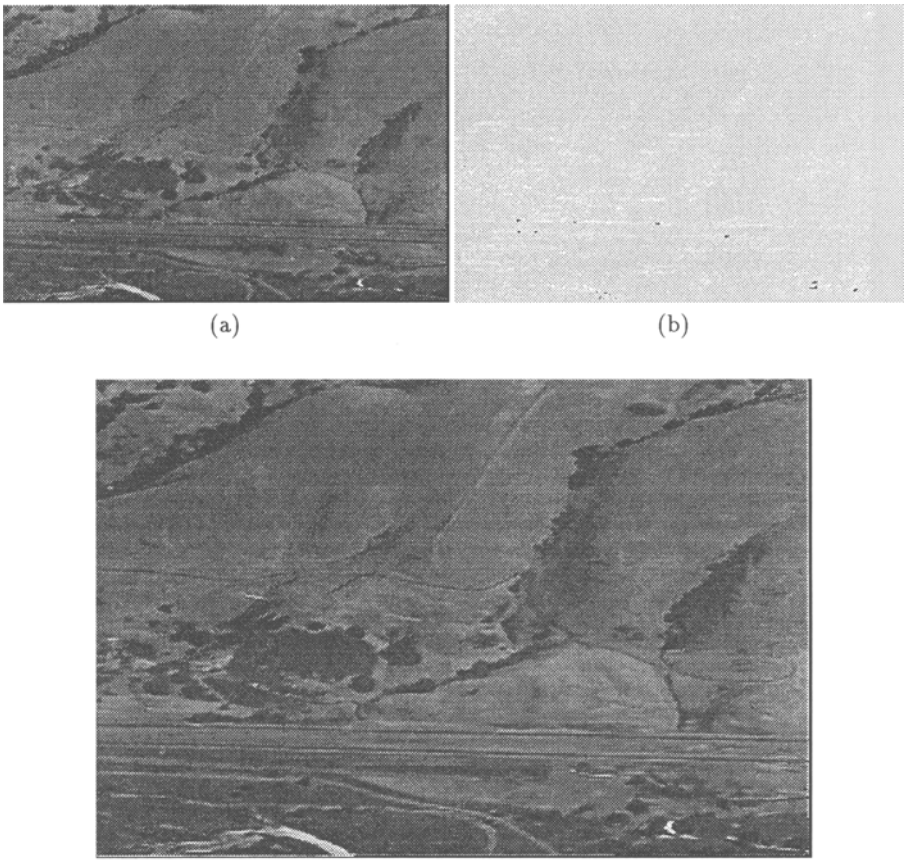

Fig. 6. *S69* sequence: (a) frame 1, (b) moving objects marked as dark spots, (c) 2× super-resolved image

4.5 Concluding Remarks

In this paper, we have proposed an alternative approach for solving global motion problems. We claim that the proposed method is superior in its robustness to random and systematic errors in the image sequence. We have shown the theoretical performance of the subspace optical flow technique and demonstrated, through numerous experiments, the ability of the algorithm to perform accurate optical flow estimation, stabilization, mosaicking and motion super-resolution, in support of our claim.

References

1. G. Adiv, "Determining 3-D Motion and Structure from Optical Flow Generated by Several Moving Objects", *IEEE PAMI*, vol. 7, no. 4, 1985.
2. M. S. Alam *et. al.*, "High-resolution Infrared Image Reconstruction using Multiple Randomly Shifted Low-resolution Aliased Frames", *Proc. SPIE 3063*, 1997.
3. P Anandan, "Measuring Visual Motion from Image Sequences", Ph. D. dissertation, University of Massachusetts, Amherst, 1987.
4. P. Anandan *et. al.*, "Real-time Scene Stabilization and Mosaic Construction", *ARPA Image Understanding Workshop*, 1994.
5. O. Axelsson, *Iterated Solution Methods*, Cambridge University Press, 1994.
6. J. L. Barron, D. J. Fleet and S. S. Beauchemin, "Performance of Optical Flow Techniques", *Int. Jour. of Comp. Vision*, vol. 12:1, pp. 43-77.
7. P. J. Burt and P. Anandan, "Image Stabilization by Registration to a Reference Mosaic", *ARPA Image Understanding Workshop*, 1994.
8. D. J. Heeger, "Model for the Extraction of Image Flow", *Jour. Opt. Soc. Amer.*, vol. 4, pp. 1455-1471.
9. B. K. P. Horn and B. G. Schunck, "Determining Optical Flow", *Artificial Intelligence*, vol. 17, pp. 185-204, 1981.
10. D. J. Fleet and A. D. Jepson, "Computation of Component Image Velocity from Local Phase Information", *Int. Jour. Comp. Vision*, vol. 5, pp. 77-104.
11. N. Gupta and L. Kanal, "Recovering 3-D motion from a Motion Field", *Special Issue on Computer Vision*, 1995.
12. S. V. Huffel and J. Vandewalle, *The Total Least Squares Problem - Computational Aspects and Analysis*, SIAM, 1991.
13. M. Irani and S. Peleg, "Improving Resolution by Image Registration", *Comp. Vision Graphics Image Proc.*, vol. 53, pp. 231-239.
14. H. Liu, "A General Motion Model and Spatio-Temporal Filters for 3-D Motion Interpretations", Ph. D. dissertation, Univ. of Maryland, 1995.
15. B. D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", *Proc. DARPA IUW*, 1991.
16. C. H. Morimoto and R. Chellappa, "Fast Electronic Digital Image Stabilization", *Proc. of IEEE ICPR*, 1996.
17. H. H. Nagel, "On the Estimation of Optical Flow", *Artificial Intelligence*, vol. 33, pp. 299-324, 1987.
18. S. Negahdaripour and B. K. P. Horn, "Direct Passive Navigation", *IEEE PAMI*, vol. 9, no. 1, pp. 168-176, 1987.
19. W. H. Press *et. al.*, *Numerical Recipes in C (2 ed.)*, Cambridge University Press, 1992.
20. S. Rakshit and C. H. Anderson, "Computation of Optical Flow Using Basis Functions", *IEEE IP*, vol. 6, no. 9, pp. 1246-1254, 1997.
21. A. Singh, "An Estimation-Theoretic Framework for Image-Flow Computation", *Proc. IEEE ICCV*, 1990.
22. S. Srinivasan and R. Chellappa, "Robust Modeling and Estimation of Optical Flow with Overlapped Basis Functions", CAR-TR-845, Univ. of Maryland, 1996.
23. R. Szeliski and J. Coughlan, "Spline-Based Image Registration", DEC-TR-CRL-94/1, Cambridge Research Laboratory, 1994.
24. S. Uras *et. al.*, "A Computational Approach to Motion Perception", *Biological Cybernetics*, vol. 60, pp. 79-97, 1988.
25. Y. S. Yao, "Electronic Stabilization and Feature Tracking in Long Image Sequences", Ph. D. dissertation CAR-TR-790, Univ. of Maryland, 1996.