A Trajectory-Based Computational Model for Optical Flow Estimation

Krishnendu Chaudhury and Rajiv Mehrotra

Abstract—**A new computational model for optical flow estimation is proposed. The proposed model utilizes trajectory information present in a multiframe spatio-temporal volume. Optical flow estimation is formulated as an optimization problem. The solution to this optimization problem yields a velocity field corresponding to smoothest and shortest trajectories of constant intensity points within the spatio-temporal volume. The approach is motivated by principles of *inertia of motion* and *least action* in physics and vision psychology. An analogy between a trajectory and a "thin wire" is discussed. A simple mechanism for handling trajectory discontinuities is also incorporated. The optimization problem is solved by stochastic relaxation techniques. Some experimental results and performance comparisons with two existing optical flow estimation techniques are presented to demonstrate the effectiveness of the proposed approach.**

I. Introduction

Constraining the flow field to satisfy various realistic assumptions about the physical world is a key factor in successful optical flow estimation. One basic assumption utilized by almost all existing optical flow techniques is the constancy of image intensity along the trajectory of a moving point. However, using this constraint alone does not yield enough information to estimate the velocity completely at every point. Additional constraints are required to overcome this problem. The most widely used additional constraint is that of spatial smoothness, which assumes that the flow field varies smoothly in space in most parts of the image [1], [2], [3], [16], [28]. An extension to the idea of constant intensity along a trajectory can be found in [26], [27] where the rigidity of local gray level structures is assumed to obtain optical flow at gray level corner locations. Recently, Konrad and Dubois utilized the concepts of data conservation and spatial smoothness in a Bayesian Maximum A Posteriori (MAP) estimation-based technique for optical flow computation [17], [16], [18]. In [18], they demonstrate that the stochastic formulation yields significantly higher accuracy than Horn and Schunk's method [1].

A common property of all these approaches is that the constraints can be derived from the information present in only two successive frames and therefore the motion information present in an extended frame sequence is not explicitly utilized. This conflicts with the findings in vision psychology, where it has been shown that human visual system requires an extended sequence of images to recover the structure of moving patterns [4], [5]. Sporadic efforts have been made to utilize an extended frame sequence for optical flow estimation. Geometry of the intensity hypesurface has been utilized to estimate the velocity field over the spatio-temporal volume [8], [6]. The former is based on the constraint that arc length of a contour does not change if that contour is moved in the direction of motion on the surface. Unfortunately, this observation is true only under the rather restrictive assumptions of rigid motion. On the other hand, the latter finds only the normal component of the velocities. Peng and Medioni [7] also proposed a method which computes the normal components of motion at every point, using slices of the spatio-temporal volume. The method performs a postprocessing to obtain the tangential components. Heeger[22] proposed a model that combines the outputs of a set of spatio-temporal motion-energy filters to extract optical flow. Yachida [11] proposed an approach which uses a three-dimensional spatio-temporal window to smooth the velocity field.

This paper introduces a new constraint, which is motivated by the principle of inertia of motion and principle of least action in physics. It is also indicated by various studies in vision psychology. The proposed constraint is based on the physical assumption that, given a set of restrictions on motion, the trajectory of each moving point is as "smooth" and as "short" as possible under the existing restrictions. This constraint, makes explicit utilization of the additional information present in an extended frame sequence, and as such, cannot be used when only two frames are provided.

Smoothness and length of trajectory of moving points has been used in feature point-based motion detection [9], [10]. Sethi and Jain [23] use smoothness and length of trajectory of moving feature points exclusively to detect feature correspondence in an extended sequence of monocular images. Rangarajan and Shah [24] proposed a proximal uniformity based algorithm to solve the feature point matching problem. Yuille and Grzywacz forwarded a theory of motion coherence. At first it included only spatial coherence [12] but later it was extended to include temporal coherence [13].

We believe that the notion of smoothness and shortness of trajectories holds equally well for intensity flow pattern over a sequence of raw intensity images, as for sets of feature points. In fact, the richer set of input information utilized improves its claim for accuracy. Also the fact that the current approach tries to minimize the data conservation error improves its claim for accuracy compared to methods like [23], [24] which utilize trajectory smoothness alone.

II. Constraints on the Trajectory

In this section, we explain the concepts of smoothness and length of the trajectory of a moving point and establish an analogy where the trajectory of an individual moving point corresponds to a thin wire. Physical motivations behind the constraints are also explained.

The direction of motion of a physical entity does not change instantaneously, due to inertia. Consequently, if our sampling rate in time is high enough (so as not to allow any dramatic change in motion between successive frames), the trajectory of every moving point (which is not occluded/disoccluded) is smooth. It should be noted that the constraint of smooth trajectories can be applied only when a sequence consisting of more than two image frames, called an extended sequence, is available. In practice, such an extended sequence is almost always available. The smoothness of trajectory concept is also supported by the principle of "visual inertia" in vision psychology [5] which suggests that when any object moves in one direction at uniform velocity we tend to perceive it as continuing its motion in that direction. The concept of short trajectories is also strongly supported by studies in vision psychology. The principle of "least action" [14] suggests that when we perceive an object moving, we tend to perceive it as moving along a path that in some sense is the shortest, simplest or most direct. A different computational model, utilizing the concepts of trajectory smoothness can also be found in [25].

Manuscript received September 13, 1993; revised June 10, 1994. This work was supported in part by the NASA-Langley Research Center under Grant NAG-1-1276. Some portions of this work appeared in the Proceedings of 1993 IEEE International Conference on Robotics and Automation, Atlanta, GA.

K. Chaudhury is with the Department of Computer Science, University of Kentucky, Lexington, KY 40506 USA.

R. Mehrotra is with the Department of Mathematics and Computer Science, University of Missouri, St. Louis, MO 63121 USA.

IEEE Log Number 9411929.

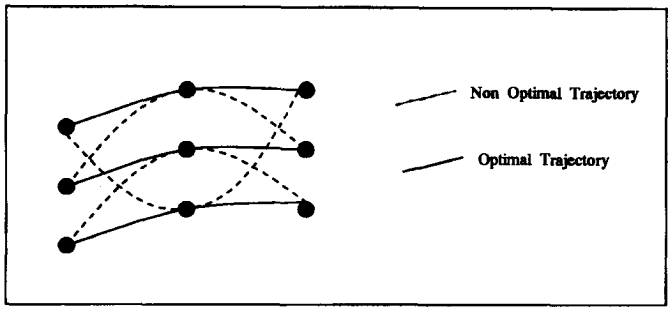

Fig. 1. Trajectory smoothness.

The projection of a smooth 3-D trajectory is a smooth 2-D trajectory in both orthographic and perspective projections [23]. Hence, the 2-D trajectory of moving intensity points should also be as smooth and as short as possible, under the constraint of data conservation.

Thus, in this paper, computation of optical flow is formulated as an optimization problem. The cost function to be minimized in the overall computation model thus comprises of the following items:

- *Data Conservation:* In practice, due to noise and other inaccuracies, the intensity of a moving point is not expected to remain exactly the same. Hence it is a common practice to incorporate data conservation error as another term in the cost function of the minimization problem [1].
- *Smoothness of Trajectory:* Given a two-dimensional curve, its curvature at any point can be regarded as a direct measure of its local smoothness. The sum of the curvatures over all the points on the curve ("total curvature") provides a measure for its overall smoothness. Hence, the following line integral

$$\int_{t_0}^{t_1} \kappa^2(t)dt \tag{1}$$

provides a measure of deviation from smoothness over the portion of trajectory extending between $t = t_0$ and $t = t_1$. (t is parameter that varies along the trajectory, and $\kappa(t)$ is the curvature at t).
- *Length of Trajectory:* In order to derive a measure for the trajectory length constraint, let t denote any parameter that varies along the trajectory and $\vec{r}(t)$ denote the position vector of its points as a function of t. Then the following expression gives a measure of the total length of the trajectory between $t = t_0$ and $t = t_1$.

$$\int_{t_0}^{t_1} \left\| \frac{d\vec{r}}{dt} \right\| dt. \tag{2}$$

Fig. 1 shows a set of equi-intensity points (points lying on the same vertical straight line belong to the same frame). Since, all the points have same intensity, any trajectory drawn through them satisfies the data conservation constraint. Under this circumstance, the trajectories that minimize the total curvature and length are chosen. Optimal trajectories are shown with solid lines in the figure. A set of nonoptimal trajectories is also shown with broken lines.

In this context, it may seem that since a minimum length trajectory approaches a straight line, it also implies a minimum curvature trajectory and vice versa. We now illustrate the separate roles of the two constraints via a counter example where a longer trajectory is smoother than the shorter one. Fig. 2 shows another set of equi-intensity points. If the set of points are joined by the pair of trajectories shown by dotted lines in Fig. 2, the total length of the trajectories is lesser compared to the pair of trajectories shown by solid lines. But the overall curvature is larger in the former case.

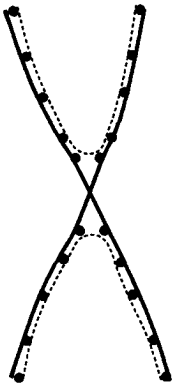

Fig. 2. Minimum length versus minimum curvature.

At this point, we would like to point out that temporal smoothness can be viewed in a subtly different sense than we have done in this work. It can also be viewed as the relative uniformity of motion of intensity points passing through a given location in the image. Such an approach has been adopted, for example, in [11]. In a sequence of images belonging to a dynamic scene, a given location is *not* occupied, in general, by the "same" intensity point. Our approach, on the other hand, tries to "track" an individual intensity point and ensure the smoothness of its trajectory. We believe that this latter approach has more physical justification.

A. Thin Wire Model for a Trajectory

In this subsection, we model each trajectory by a thin wire. A thin wire is defined as a one-dimensional continuum, whose potential energy has two components, potential energy of deformation, and, potential energy of stretching or elongation. The potential energy of deformation is proportional to the integral of the square of its curvature, extended over its length [21]. Thus, if t denotes any parameter that varies along the wire, $\kappa(t)$ denotes the curvatures of its points expressed as a function of t, the potential energy of deformation of the wire extending from $t = t_0$ to $t = t_1$ is

$$\mu \int_{t_0}^{t_1} \kappa^2(t)dt \tag{3}$$

where μ is a measure of the wire's (un)willingness to deform. Thus, if the thin wire models a trajectory, from the above discussions, it is clear that its potential energy of deformation (see (3)) is a measure of the deviation from smoothness over the entire trajectory (i.e., overall change in direction, as expressed in (1)).

The other component of the thin wire's potential energy corresponds to elongation. The thin wire "objects" to stretching or elongation. Thus, the the potential energy of the wire is proportional to the increase in length compared to the length at rest, the factor of proportionality being α, which measures the wire's (un)willingness to stretch [21]. Thus, if t denotes any parameter that varies along the wire, and $\vec{r}(t)$ denotes the position vector of its points as a function of t, the following is a measure of its potential energy

$$\alpha \int_{t_0}^{t_1} \left\| \frac{d\vec{r}}{dt} \right\| dt. \tag{4}$$

The principles of least action require that the path followed by any moving point is the shortest under the existing constraints. This constraint can be suitably expressed in the thin wire model for the trajectory via the potential energy of elongation in (4), which is analogous to (2).

In Section IV, the above model will be augmented by allowing the thin wire to be "weak," to account for step and crease discontinuities in the trajectory, caused by occlusion/disocclusion.

III. The Optical Flow Computation Model

A. Mathematical Expressions for Trajectory Constraints

In this subsection, we derive expressions for trajectory constraints, in terms of the 2-D velocity vector field (optical flow field) within the spatio-temporal volume.

Let $\vec{V}(x,y,t) = [u(x,y,t) \quad v(x,y,t)]^T$ denote the velocity field within the spatio-temporal volume. Accordingly, consider a specific intensity point at $(X(t),Y(t))$ at time t. Its trajectory is a 2-D curve which can be described by the parametric equation (t, which physically corresponds to time, being the parameter):

$$\vec{r}(t) = X(t)\vec{i} + Y(t)\vec{j} \tag{5}$$

where $\vec{i}$ and $\vec{j}$ denote the unit vectors in the directions of x and y, respectively. Its velocity is given by

$$\frac{d\vec{r}}{dt} = \frac{dX}{dt}\vec{i} + \frac{dY}{dt}\vec{j} = u(X,Y,t)\vec{i} + v(X,Y,t)\vec{j} \tag{6}$$

where the second equality follows from the definition of u and v. Thus, the cost function corresponding to the length constraint (potential energy of elongation) of a trajectory is given by

$$\int_{t_0}^{t^1} \left\| \frac{d\vec{r}}{dt} \right\| dt = \int_{t_0}^{t^1} \sqrt{(u^2+v^2)}\,dt \tag{7}$$

t_0 and t_1 being the times corresponding to the initial and final frames, respectively.

The derivation for the smoothness constraint (potential energy of deformation) is slightly more involved. The rate of change of the x-component of the point's velocity with respect to time, i.e., d^2X/dt^2, can be expressed as

$$\frac{d^2X}{dt^2} = \frac{du(X(t),Y(t),t)}{dt} = u_x u + u_y v + u_t. \tag{8}$$

Similarly, the time rate of change of y-component of the point's velocity

$$\frac{d^2Y}{dt^2} = \frac{dv(X(t),Y(t),t)}{dt} = v_x u + v_y v + v_t. \tag{9}$$

Thus, (8) and (9) give the x and y components of the acceleration of a given point. For the sake of convenience, the following symbols are used to denote the two components of acceleration:

$$P(x,y,t) = u_x u + u_y v + u_t \tag{10}$$

$$Q(x,y,t) = v_x u + v_y v + v_t. \tag{11}$$

Thus, the expression for the acceleration of a point is as follows:

$$\frac{d^2\vec{r}}{dt^2} = \frac{d\vec{V}(X,Y,t)}{dt} = P(x,y,t)\vec{i} + Q(x,y,t)\vec{j}. \tag{12}$$

It should be noted that the derivative of the velocity field used here is the *full derivative (and not partial)*. This is what allows us to "track" the motion of individual points. Using partial derivative would correspond to observing the points passing through a specific location in the image.

The *curvature of the trajectory of a intensity point* (which measures the rate of change of the unit tangent vector to the trajectory along the curve) is evaluated as follows: Letting s be the arc-length parameter corresponding to the trajectory curve, we have

$$\left\| \frac{d^2\vec{r}}{ds^2} \right\|^2 = \frac{(Pv-Qu)^2}{(u^2+v^2)^3}. \tag{13}$$

By definition, the curvature is the magnitude of the rate of change of the unit tangent vector along the curve. Hence, (13) gives the expression for the curvature at any point on the trajectory.

It should be noted that the vector $\frac{v}{\sqrt{u^2+v^2}}\vec{i} + \frac{-u}{\sqrt{u^2+v^2}}\vec{j}$ is orthogonal to the velocity vector $u\vec{i}+v\vec{j}$. In fact, it is the unit normal vector to the trajectory. Also, from (12), $P\vec{i}+Q\vec{j}$ is the acceleration vector. The inner product of these two vectors is

$$A_n = \frac{Pv-Qu}{\sqrt{u^2+v^2}} \tag{14}$$

which corresponds to the *normal component of the acceleration*. Therefore,

$$\left\| \frac{d^2\vec{r}}{ds^2} \right\|^2 = \frac{A_n^2}{\|\vec{V}\|^4}. \tag{15}$$

The above result is not surprising because it is well known that it is the *normal component of the acceleration* that causes the *change in direction of the trajectory*. Thus, the total deviation from smoothness of the entire trajectory is given by

$$\int_{t_0}^{t_1} \frac{(Pv-Qu)^2}{(u^2+v^2)^3}\,dt. \tag{16}$$

B. The Optimization Problem

The overall cost for stretching and deformation of a trajectory is hence given by

$$\int_{t_0}^{t_1} \alpha\sqrt{(u^2+v^2)} + \mu\frac{(Pv-Qu)^2}{(u^2+v^2)^3}\,dt. \tag{17}$$

The total cost for all the trajectories in the spatio-temporal volume is

$$\int\int\int_R \alpha\sqrt{(u^2+v^2)} + \mu\frac{(Pv-Qu)^2}{(u^2+v^2)^3}\,dtdxdy \tag{18}$$

where R denotes the entire spatio-temporal volume. Our objective is to find a vector optical flow field that minimizes the above cost functional along with the data conservation error. Hence the goal is to find a vector optic flow field $\vec{V}(x,y,t)$ that minimizes the following functional:

$$H(u,v) = \int\int_R\int [\lambda(E_x u + E_y v + E_t)^2 + \alpha\sqrt{(u^2+v^2)} + \mu\frac{(Pv-Qu)^2}{(u^2+v^2)^3}]dtdxdy \tag{19}$$

where the region of integration R is the entire spatio-temporal volume and λ, α and μ are chosen so as to reflect the relative importance of the corresponding terms. In other words, λ, α and μ measure the unwillingness of the trajectory to change its intensity, stretch and deform, respectively. Without loss of generality, we can assign $\lambda = 1$.

In practice, the image intensity field is not given in the form of a closed form function $E(x,y,t)$. The spatio-temporal volume R is sampled on a three-dimensional lattice $\mathcal{L}$ in $\Re^3$. Thus, given a sequence of M_3 images, each with $M = M_1 \times M_2$ pixels (where M_1, M_2 denote the horizontal and vertical image size, respectively), we have a lattice $\mathcal{L}$, of size, $|\mathcal{L}| = M_1 \times M_2 \times M_3$ points. We define a grid of *sites*

$$S = \{s_1, s_2, \ldots, s_{|\mathcal{L}|}\} \mid \forall w \quad 0 \le x(s_w) \le M_1, \; 0 \le y(s_w) \le M_2, 0 \le t(s_w) \le M_3$$

where $(x(s_w), y(s_w), t(s_w))$ denote the pixel coordinates of site s_w in the spatio-temporal volume. The unknown velocity field will be estimated on the same lattice $\mathcal{L}$. Thus the detected velocity field will comprise a set of 2-D vectors $\vec{V}_s = [u_s v_s]^T$, $s \epsilon \mathcal{L}$, one vector for each

site s. The *state space* of the velocity vector at every site is discrete. We denote this state space for site s as $\mathcal{V}_s$. In our implementation, the state space is identical for all the sites. Thus, if u^m and v^m denote the maximum possible absolute values of u and v, respectively, $\mathcal{V}_s = \{(u_s, v_s) : -u^m \leq u_s \leq u^m, -v^m \leq v_s \leq v^m\}$. If δ_u and δ_v denote the resolution in u and v, the size of the state space at every pixel is $(2u^m/\delta_u + 1) \times (2v^m/\delta_v + 1)$. The resolution of the state space should be sufficiently high to yield the desired accuracy in the velocity estimates of the optical flow field. u^m and v^m should correspond to the upper bounds on the expected absolute values of u_s and v_s, respectively. The overall state space of all the pixels is the N-fold Cartesian product of the state space of each individual pixel (N being the number of sites).

Corresponding discrete approximation to the cost function in (19) corresponding to a velocity configuration $\vec{V} = [u \;\; v]^T = \left\{\vec{V}_s = [u_s \;\; v_s]^T : s\epsilon\mathcal{L}\right\}$ is

$$H(u,v) = \sum_{s\epsilon S} (\phi_1(E_s)u_s + \phi_2(E_s)v_s + \phi_3(E_s))^2 + \alpha\sqrt{(u_s^2 + v_s^2)} + \mu\kappa_s^2 \tag{20}$$

where P_s and Q_s are as follows

$$P_s = \phi_1(u_s)u_s + \phi_2(u_s)v_s + \phi_3(u_s) \tag{21}$$

$$Q_s = \phi_1(v_s)u_s + \phi_2(v_s)v_s + \phi_3(v_s) \tag{22}$$

and κ_s (curvature at site s) given by

$$\kappa_s^2 = \frac{(P_s v_s - Q_s u_s)^2}{(u_s^2 + v_s^2)^3} \tag{23}$$

ϕ_1, ϕ_2, and ϕ_3 are operators for estimating partial derivatives at a given site along the x, y, and t axes, respectively.[1] It should be noted that ϕ_1, ϕ_2, and ϕ_3 are *local* operators, i.e., they involve *only* the values at the sites belonging to a (usually small) neighborhood (in the 3-D lattice) around the given point. In fact, the cost function in (20) can be rewritten as

$$H(u,v) = \sum_{s\epsilon S} H_s(u,v)$$

where

$$H_s(u,v) = (\phi_1(E_s)u_s + \phi_2(E_s)v_s + \phi_3(E_s))^2 + \alpha\sqrt{(u_s^2 + v_s^2)} + \mu\kappa_s^2.$$

Given a particular configuration of the velocity field $\vec{V} = \left\{\vec{V}_s : s\epsilon\mathcal{L}\right\}$, $H_s(u,v)$ is the *local cost* associated with the site s, corresponding to the velocity field configuration $\vec{V}$. Thus, the overall cost $H(u,v)$ is the sum of the local costs over all the sites in the spatio-temporal volume. This property is important because it will enable us to apply the Stochastic Relaxation technique to solve the optimization problem.

C. Stochastic Relaxation

The stochastic relaxation scheme is briefly discussed here. For a detailed discussion, including proof of convergence, the reader is referred to [20]. The objective is to obtain a velocity configuration $\vec{V} = [u,v]^T$ that will minimize the cost function in (20). Let $k = 0, 1, 2 \cdots$ index the successive stages of iteration. Let $\vec{V}(k) \doteq \left(\vec{V}_{s_1}(k), \vec{V}_{s_2}(k), \cdots \vec{V}_{s_{|\mathcal{L}|}}(k)\right)$ denote the configuration at iteration k. In other words, $\vec{V}_{s_i}(k)$ denote the velocity estimate of the pixel at site s_i at iteration k. The process starts with an arbitrary configuration $\vec{V}(0)$. All the sites in the spatio-temporal volume are visited in some specific order (e.g., raster scan order). At each discrete iteration step, a specific site is visited and a change is made in the velocity estimate at *that* site only. Thus $\vec{V}(k-1)$ and $\vec{V}(k)$ can have different velocity estimates at no more than one site. When a particular site is visited, the velocity estimate at that site is updated, based on the current velocity estimates of its neighbors. The process of updation is as follows: Let k denote the current iteration, $\vec{V}(k)$ being the corresponding velocity configuration. Let s denote the site being visited currently. $\vec{V}(k)$ and $\vec{V}(k+1)$ will be same at every site except s. The new vector $\vec{V}_s(k+1)$ has to be chosen from the state space of site s, viz., $\mathcal{V}_s$. If we replace the current velocity vector at site s with a new vector $\tilde{\vec{V}}_s$ we get a new configuration $\tilde{\vec{V}}_{k+1}$, with an associated local cost $H_s(\tilde{u}_{k+1}, \tilde{v}_{k+1})$. We can create a probability distribution, where each vector in the state space $\mathcal{V}_s$ has an associated probability

$$\frac{1}{Z} e^{-\frac{H_s((\tilde{u}_{k+1}, \tilde{v}_{k+1})}{\mathcal{T}}}$$

(Z being the normalization constant and $\mathcal{T}$ being the so called "temperature parameter").[2] A random sample is drawn from the above probability distribution, and this sampled vector is the updated value $\vec{V}_s(k+1)$. It should be noted that in general, velocity vectors that *reduce* the local cost have a higher probability of being chosen. However, there exists a small, but nonzero probability of choosing a vector which actually *increases* the local cost function. Such "uphill" moves will be made from time to time. They enable the relaxation process to escape from a local minima.

The temperature parameter $\mathcal{T}$ controls the degree of "peaking" in the probability distribution. At higher temperatures, the probability of making an "uphill" move is larger. As the temperature is reduced, the process starts degenerating into a strict descent process. The extreme cases $\mathcal{T} = 0$ and $\mathcal{T} = \infty$ correspond to "greedy descent" and "random search," respectively. In [20], it has been shown that if we start with a sufficiently high temperature, and reduce the temperature with every round of visiting all the sites, according to a logarithmic "cooling schedule"

$$\mathcal{T}(k) = \frac{C}{\log(1+k)} \qquad 1 \leq k \leq K$$

($\mathcal{T}(k)$ denotes the temperature at kth full sweep over all the sites), the relaxation process is guaranteed to converge to a global minima, irrespective of the starting configuration $\vec{V}(0)$.

IV. Handling Discontinuities

A primitive mechanism to handle discontinuities in the velocity field was implemented, based on the idea of "line process" [20], [19]. This mechanism simply identifies the pixel sites at which some kind of discontinuity occurs (due to occlusion/disocclusion, noise spike etc.) and prevents the dubious velocity estimation at these points from influencing the estimate at other points. Accordingly, the energy (cost) function in (20) is augmented as follows

$$H(u,v) = \sum_{s\epsilon S} H_s(u,v)(1 - l_s) + \lambda_d l_s \tag{24}$$

where l_s is a binary variable ("line process"). Initially, $l_s = 0, \quad \forall s$. The optimization process assigns the value $l_s = 1$ at sites of discontinuity. The constant λ_d is the penalty for introducing a discontinuity and must be sufficiently high to prevent the labeling of regular sites as discontinuous. In terms of the thin wire model for trajectory, we now make the thin wire "weak," i.e., it is allowed to

[1] In our implementation, partial derivatives are estimated by fitting a surface to the neighborhood see Section V-A.

[2] It should be noted that above distribution can be constructed from only a small neighborhood around the site s. The exact nature of the neighborhood depends on the nature of the partial derivative operators.

have C^0 and C^1 discontinuities. The state space at every site now has an additional state. While updating the velocity estimate at a particular site, the relaxation process does not consider the neighbors that are currently labeled as discontinuous (i.e., $l_s = 1$). Also, while calculating partial derivatives by fitting a surface to the neighborhood, such discontinuous sites are excluded.

V. Implementation and Results

The proposed technique has been implemented on a Sun sparcstation 2 in C language. Numerous experiments have been conducted on sequences of real and synthetic images to test the efficacy of the proposed approach. Some of these results are presented in this section. The performance of the proposed technique has been compared with that of Horn and Schunk [1] and [15]. In order to conduct a more meaningful comparison with our approach, which tracks individual intensity points in space, Horn and Schunk's approach has been modified to include a temporal smoothness term that enforces smoothness of velocity at a given spatial location over time. Thus the modified cost functional to be minimized takes the form

$$\int\int\int (E_x u + E_y v + E_t)^2 + \lambda\left(u_x^2 + u_y^2 + v_x^2 + v_y^2 + u_t^2 + v_t^2\right) dx dy dt.$$

Horn and Schunk approach is thus augmented to a multiframe approach. Black and Anandan's [15] approach has been chosen for comparison for two reasons. Firstly, it is a multiframe approach and includes a temporal smoothness constraint that is slightly more sophisticated than the above modified Horn and Schunk's approach. It tries to enforce relative uniformity of not only the velocity, but also its time rate of change, at a given spatial location over time. Secondly, it includes a novel technique for handling the problem of discontinuities occurring at motion boundaries. Instead of the traditional squared error function, they have used an error function which is robust in presence of outliers. This error function behaves like the traditional squared error function when the error is small, but approaches zero as the error becomes large. Thus the influence of outliers (corresponding to uncorrelated motion at motion boundaries) is reduced.

For quantitative evaluation of the accuracy of the detected optical flow, an angular error measure similar to the one used in [30] is employed. A 2-D velocity vector $\vec{V} = [uv]^T$ is written as a 3-D unit vector

$$\hat{V} = \frac{1}{\sqrt{(u^2 + v^2 + 1)}}[uv1]^T.$$

Given the ideal velocity vector $\vec{V}_i$ and the computed velocity vector $\vec{V}_c$ the corresponding error is given by

$$\arccos\left(\hat{V}_i^T \hat{V}_c\right).$$

The advantage of this error measure lies in the fact that it is normalized with respect to large and small speeds. To compute the overall error associated with the entire velocity field average of the normalized errors at each point of the spatio-temporal volume is computed.

A. Partial Derivatives

The partial derivatives of the intensity field, needed in order to compute the data conservation error, at a site s are computed by fitting a bicubic surface to the 5×5 spatial neighborhood of the site. The partial derivatives of u and v with respect to x and y at site s are also computed by bicubic surface fitting as above. The partial derivatives with respect to time, at site s corresponding to position x, y, t, are computed by fitting a least squares cubic curve to the set of 5 points, corresponding to the frames $t-2, t-1, t, t+1, t+2$ and location x, y in spatio-temporal volume. The points marked with trajectory discontinuity are excluded from the above fittings.

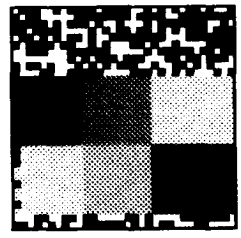

Fig. 3. Frame 0 of belt pair image sequence.

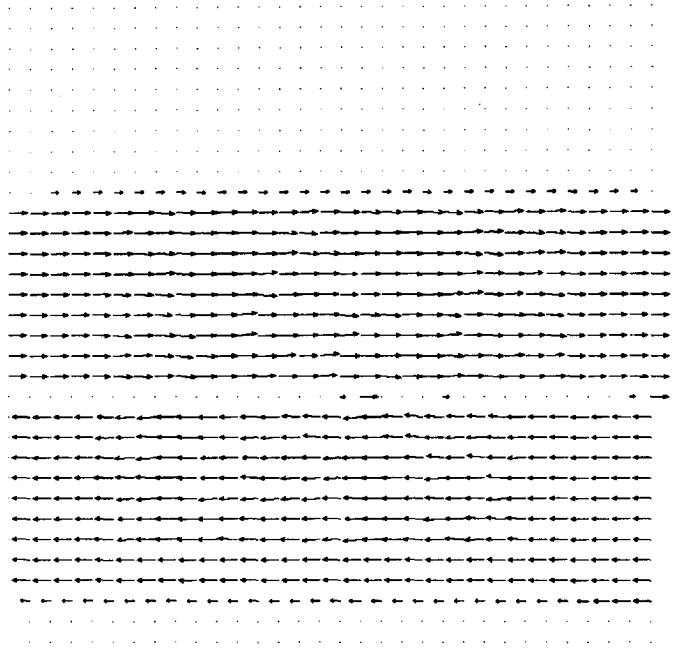

Fig. 4. Optical flow estimated by the proposed technique at frame 9 of belt pair sequence (normal error = 1.59°).

B. Cooling Scheme

The logarithmic cooling schedule described in [20] is impractically slow. In practice, a faster cooling schedule is generally chosen. Accordingly, in this paper, the schedule $\mathcal{T}(k) = \mathcal{T}_0 a^{(k-1)}$ is used, with $a = 0.99$ for all the experiments. $\mathcal{T}_0 = 1.0$ for the first two experiments and $\mathcal{T}_0 = 1.5$ for the last one.

C. Moving Belt Pair Experiment

In order to test the algorithm's performance in a situation where two dynamically different motions are present very close to each other, a synthetic image of size 64×64 was created with two striped belts lying side by side and translating in opposite directions. The amount of displacement between two successive frames varied sinusoidally, i.e., if d_t denotes the displacement of a certain intensity point at (discrete) time frame t

$$d_t = A \sin\left(\frac{\pi}{2} t\right) + B.$$

The entire spatio-temporal volume contained thirty frames. The value of the parameters were as follows $u^m = v^m = 7.0$ and $\delta_u = \delta_v = 0.25$, $\alpha = 0.5$, $\mu = 5$ and $\lambda_d = 30$. The results shown were obtained after 400 iterations.

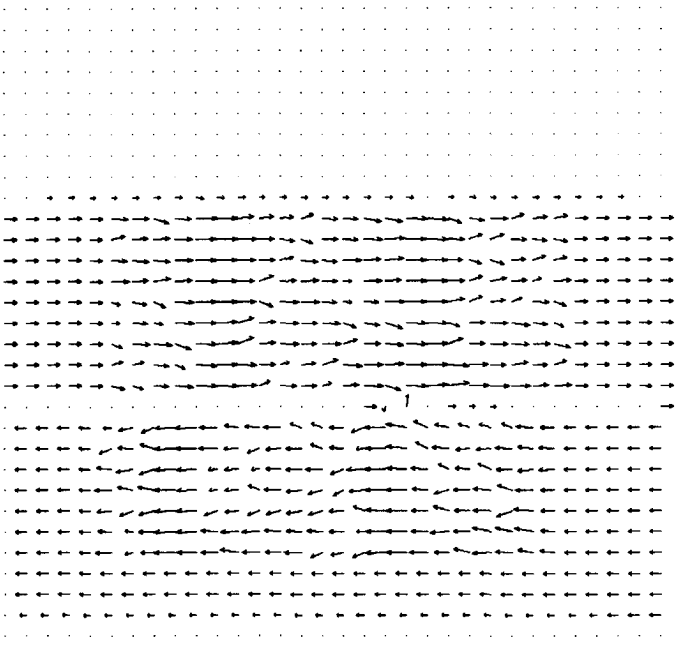

Fig. 5. Optical flow estimated by Black Anandan's technique at frame 9 of the belt pair sequence (normal error = 4.23°).

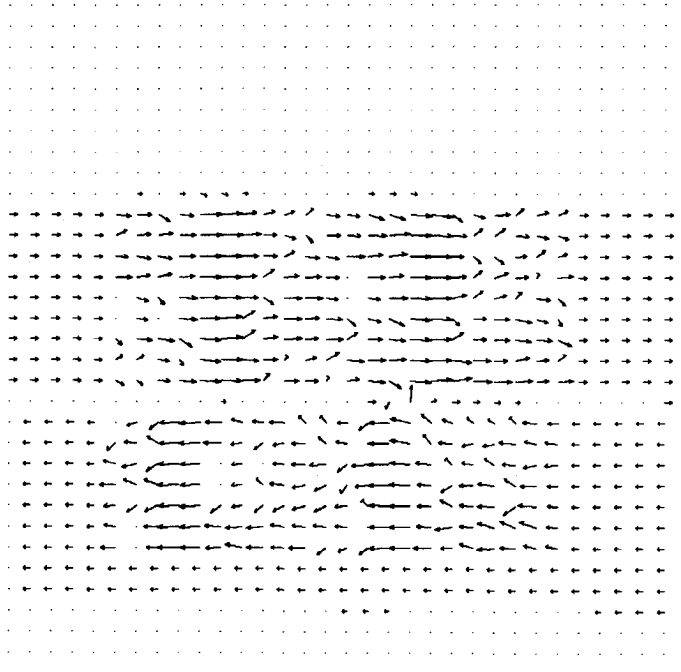

Fig. 6. Optical flow estimated by the modified Horn and Schunk technique at frame 9 of the belt pair sequence (normal error = 7.76°).

Fig. 7. Frame 0 of diverging tree sequence.

Each frame was corrupted separately with additive zero-mean Gaussian noise. Frame 0 of the sequence is shown in Fig. 3 and

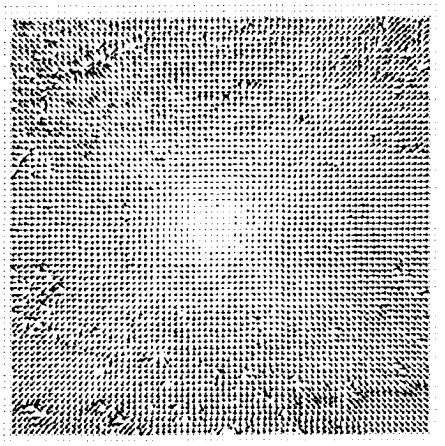

Fig. 8. Optical flow detected for diverging tree sequence using proposed technique (normal error = 6.58°).

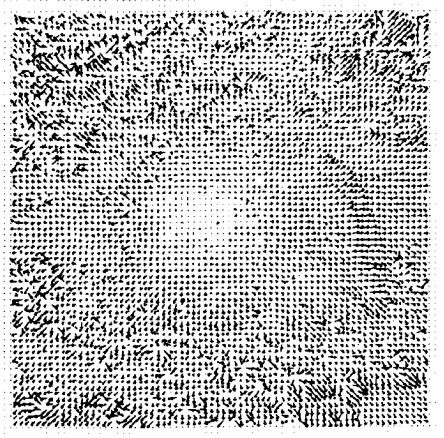

Fig. 9. Optical flow detected for diverging tree sequence using Black Anandan's technique (normal error = 11.9°).

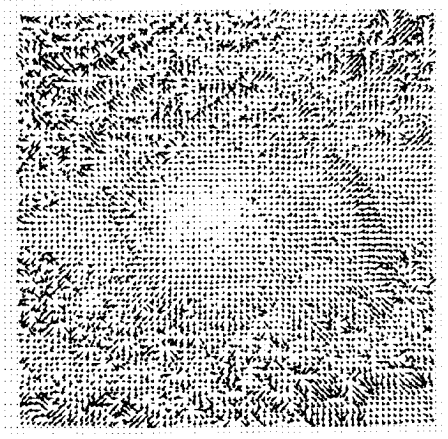

Fig. 10. Optical flow detected for diverging tree sequence using modified Horn and Schunck's technique (normal error = 18.37°).

the corresponding optical flow detected by the proposed, Black and Anandan's, and the modified Horn Schunk's techniques at frame 3 are shown as needle diagrams in Figs. 4, 5, and 6, respectively. Associated error measures are given in the figure itself. It is evident from the results of this experiment that the proposed technique is highly robust with respect to noise and motion discontinuities (appearing at the boundaries of the two moving belts).

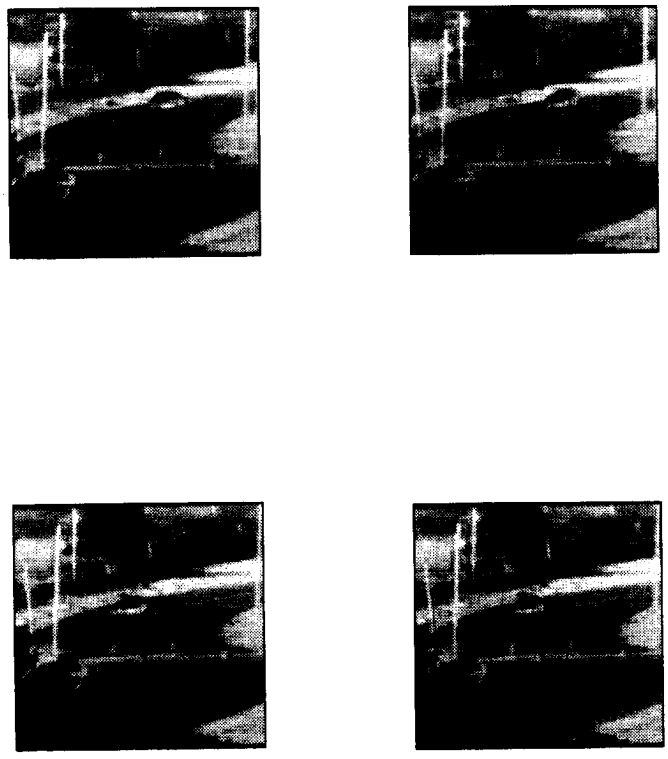

Fig. 11. Frames 1, 2, 9, and 10 of moving car image sequence.

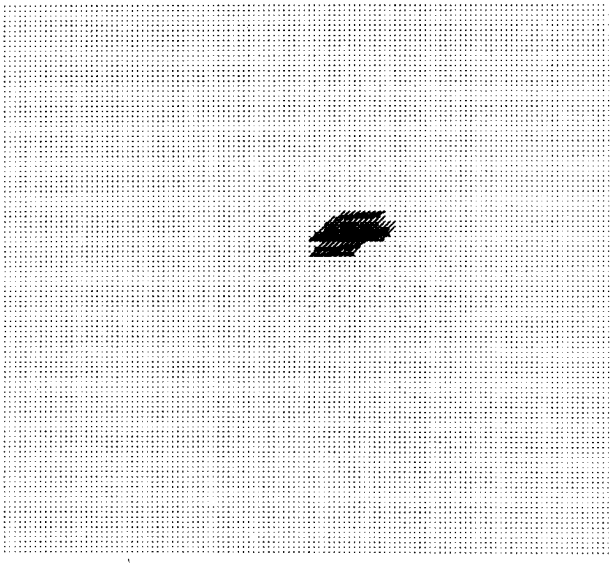

Fig. 12. Optical flow estimated by proposed technique for moving car sequence (frame 2).

D. Diverging Tree Sequence Experiment

This experiment involves a sequence of 10 frames intensity images of a scene containing trees. Fig. 7 shows a frame of a diverging tree sequence. The motion is synthetic where the camera moves along its line of sight sinusoidally. If Z_t denotes the distance between the mobile camera and a static scene point at (discrete) time t,

$$Z_{t+1} = Z_t - A \sin\left(\frac{\pi}{2}t\right) - B.$$

The focus of expansion is at the center of the image. $u^m = v^m = 12.0$ and $\delta_u = \delta_v = 0.25$. $\alpha = 0.5$, $\mu = 10$ and $\lambda_d = 20$. The results shown were obtained after 400 iterations. The detected optical flow

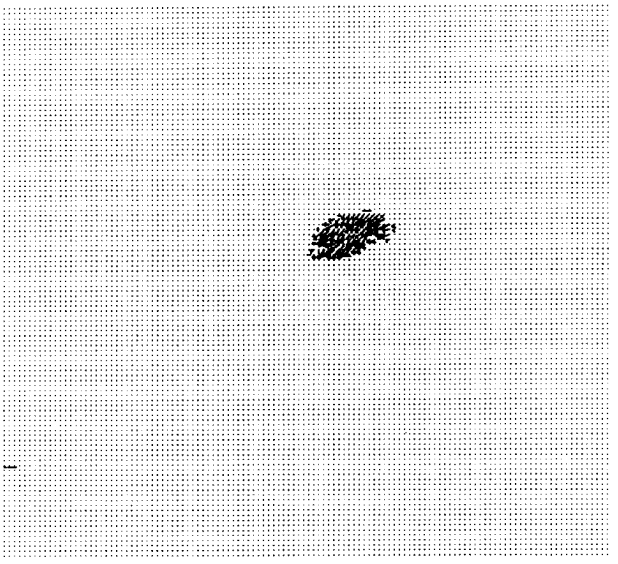

Fig. 13. Optical flow estimated by Black Anandan's technique for moving car sequence (frame 2).

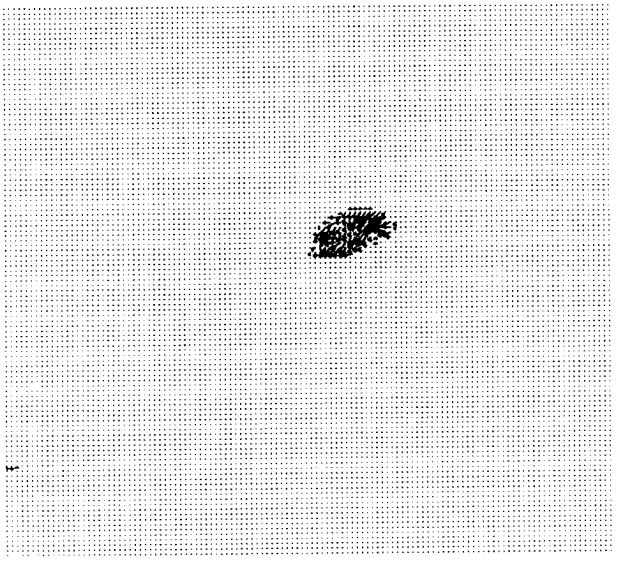

Fig. 14. Optical flow estimated by modified Horn Schunk's technique for moving car sequence (frame 2).

fields at frame 5, by using the proposed, Black and Anandan's, and the modified Horn and Schunck's techniques are shown in Figs. 8, 9, and 10, respectively. Associated error measures appear in the figure itself. Thus, in this experiment too, the proposed technique yields significantly higher accuracy.

E. The Moving Car Experiment

This experiment involves a real sequence of infrared images of a nighttime scene containing a moving car. Each image in the sequence is of size 220 × 240 and the entire sequence contains 30 images (frames 0 through 29). The state space is identical at every pixel with $u^m = v^m = 3.0$, and $\delta_u = \delta_v = 0.1$. $\alpha = 0.5$, $\mu = 10$ and $\lambda_d = 20$. The results shown were obtained after 400 iterations. This experiment poses several challenges. Firstly the images being infrared images taken at night, contain a high degree of noise. Secondly, the motion involved is a mixture of translation and rotation.

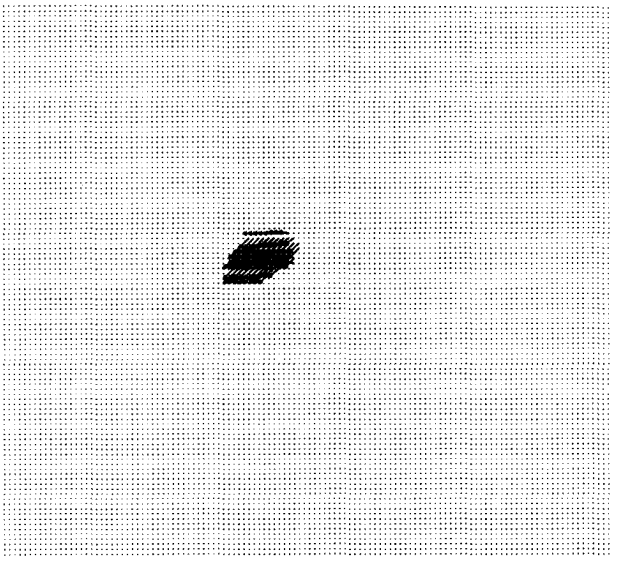

Fig. 15. Optical flow estimated by proposed technique for moving car sequence (frame 9).

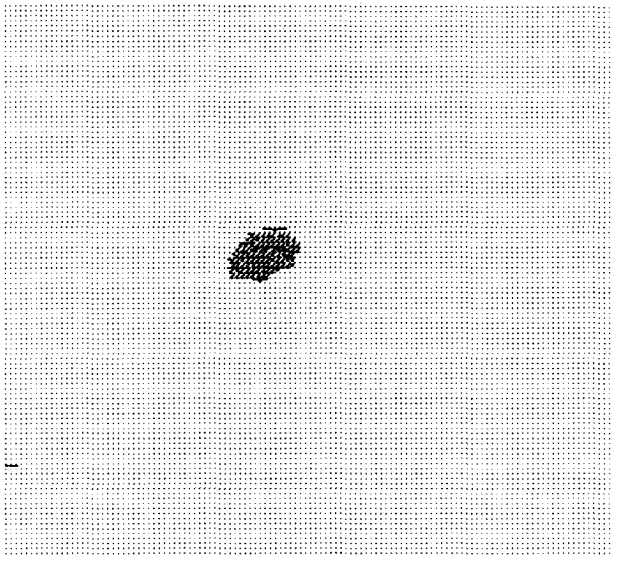

Fig. 16. Optical flow estimated by Black Anandan's technique for moving car sequence (frame 9).

Frames 1, 2, 9, and 10 are shown in Fig. 11 and the estimated optical flows at frames 2 and 9 for the proposed, Black and Anandan's, and the modified Horn and Schunk's techniques are shown in Figs. 12, 13, 14, 15, 16, and 17, respectively.

These and other experimental results clearly demonstrate the effectiveness and importance of the smoothness of intensity trajectory constraint and its modeling in terms of thin wire potential energies in optical flow computation from an extended sequence of images.

VI. Conclusion

Vision psychology studies have demonstrated the effectiveness and importance of smoothness and length of trajectories in motion detection. In this paper, a cost function is proposed which measures the total curvature and length of all the trajectories in the spatio-temporal volume, corresponding to a given optical flow field. An analogy between trajectories and the "thin wire" in physics has been shown, where the above cost function corresponds to the potential energy of stretching and deformation of the wire. Minimizing this cost associated with trajectories yields optical flow vectors corresponding to maximally smooth and short trajectories under the data conservation constraint. Experimental results demonstrating the effectiveness of the proposed model are provided, along with comparisons with the modified Horn Schunk technique, and Black and Anandan's technique. Normalized error measures are provided in cases where the ground truths were known. In all our experiments, where ground truths were known, the normalized error was minimum for the proposed technique. optical flow estimation in comparison to the other two multiframe methods We believe that this model can also be applied to a range image sequence analysis.

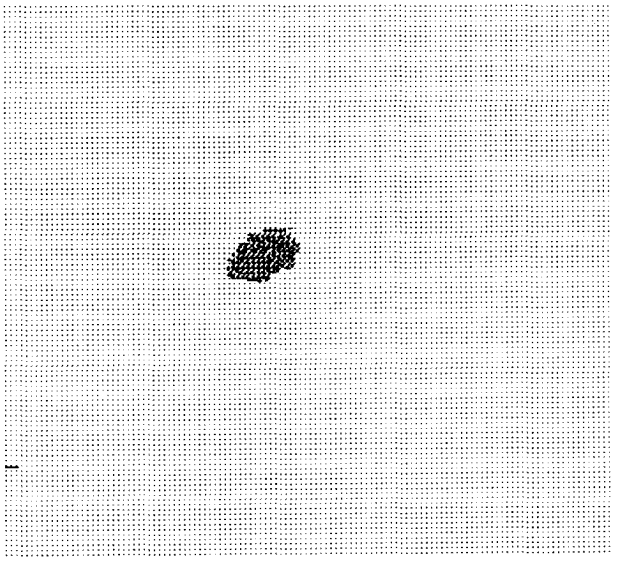

Fig. 17. Optical flow estimated by modified Horn Schunk's technique for moving car sequence (frame 9).

References

[1] B. K. P. Horn and B. G. Schunck, "Determining optical flow,"*Artificial Intell.*, vol. 17, pp. 185–203, 1981.

[2] B. G. Schunck, "Image flow: Fundamentals and future research," in *Proc. IEEE Conf. Pattern Recognition and Image Processing*, 1985, pp. 560–571.

[3] B. K. P. Horn, *Robot Vision*. Cambridge, MA: MIT Press, 1986.

[4] J. T. Todd, "Visual information about rigid and nonrigid motions," *J. Experimental Psychology, Human Perception Performance*, vol. 8, pp. 238–252, 1982.

[5] V. S. Ramachandran and S. M. Antis, "Extrapolation of motion path in human visual perception," *Vision Res.*, vol. 23, pp. 83–85, 1983.

[6] S. P. Liou and R. C. Jain, "Motion detection in spatiotemporal space," *Computer Vision Graphics and Image Processing*, vol. 45, pp. 227–250, 1989.

[7] S. L. Peng and G. Medioni, "Interpretation of image sequences by spatiotemporal analysis," in *Proc. Workshop on Visual Motion*, 1989, pp. 344–351.

[8] M. Allmen and C. Dyer, "Computing spatio-temporal surface flow," in *Proc. Third Int. Conf. Computer Vision*, Osaka, Japan, 1990, pp. 47–50.

[9] M. Jenkin, "Tracking three dimensional moving light displays," in *Proc. Workshop Motion: Representation and Control*, Toronto, Canada, 1983, pp. 66–70.

[10] M. Jenkins and M. Tsotsos, "Applying temporal constraints to the dynamic stereo problem," *Computer Vision, Graphics, and Image Processing*, vol. 33, pp. 16–33, 1986.

[11] M. Yachida, "Determining velocity map by 3-D iterative estimation," in *Proc. Seventh Int. Joint Conf. Artificial Intell.*, Vancouver, Canada, 1981, pp. 24–28.

[12] A. Yuille and N. Grzywacz, "The motion coherence theory," in *Proc. Second Int. Conf. Computer Vision*, Tampa, FL, 1988, pp. 344–353.

[13] N. Grcywacz, J. Smith, and A. Yuille, "A common theoretical framework for visual motion's spatial and temporal coherence," CH2716, pp. 148–155, 1989.
[14] R. Shepard and L. Cooper, *Mental Images and their Transformations*. Cambridge, MA: MIT Press, 1982.
[15] M. J. Black and P. Anandan, "Robust dynamic motion estimation over time," in *Proc. Computer Vision and Pattern Recognition*, Maui, Hawaii, 1991, pp. 296–302.
[16] J. Konrad and E. Dubois, "Bayesian estimation of motion vector fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-14, no. 9, pp. 910–927, Sept. 1992.
[17] , "Multigrid Bayesian estimation of image motion fields using stochastic relaxation," in *Proc. Second Int. Conf. Computer Vision*, Tampa, FL, 1988, pp. 354–362.
[18] ——, "Comparison of stochastic and deterministic solution methods in Bayesian estimation of 2D motion fields" *Image and Vision Computing*, vol. 9, no. 4, pp. 215–228, Aug. 1991.
[19] J. Hutchinson, C. Koch, J. Luo, and C. Mead, "Computing motion using analog and binary resistive network," *IEEE Computer*, vol. 21, pp. 52–63, Mar. 1988.
[20] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 6, pp. 721–741, Nov. 1987.
[21] R. Courant and D. Hilbert, *Methods of Mathematical Physics*, vol. 1. New York: Interscience, 1970.
[22] D. J. Heeger, "Optical flow from spatiotemporal filters," in *Proc. First Int. Conf. Computer Vision*, London, England, 1987.
[23] I. K. Sethi and R. C. Jain, "Finding trajectories of feature points in a monocular image sequence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 1, Jan. 1987.
[24] K. Rangarajan and M. Shah. "Establishing motion correspondence," in *Proc. Computer Vision and Pattern Recognition*, Lahaina, Hawaii, 1991, pp. 103–108.
[25] K. Chaudhury and R. Mehrotra, "Optical flow estimation using smoothness of intensity trajectories," CVGIP: *Image Understanding*, vol. 60, no. 2, pp. 230–244, Sept. 1994.
[26] H. H. Nagel, "Displacement vectors derived from second-order intensity variation in image sequences," *Computer Vision Graphics and Image Processing*, vol. 21, pp. 85–117, 1983.
[27] ——, "On the estimation of optical flow: Relations between different approaches and some new results," *Artificial Intell.*, vol. 33, pp. 299–324, 1987.
[28] E. C. Hildreth, "Computations underlying the measurement of visual motion," *Artificial Intell.*, vol. 23, pp. 309–354, 1984.
[29] D. Marr, *Vision*. New York: Freeman, 1982.
[30] J. L. Barron, D. J. Fleet, S. S. Beauchemin, and T. A. Burkitt, "Performance of optical flow techniques," in *Proc. Computer Vision and Pattern Recognition*, Champaign, IL, 1992, pp. 236–242.

Uncertainty in Object Pose Determination with Three Light-Stripe Range Measurements

Keiichi Kemmotsu and Takeo Kanade

Abstract—**The pose (position and orientation) of a polyhedral object can be determined by using a set of simple light-stripe range sensors. Data from these sensors, however, inherently contains errors which result in pose uncertainty. Therefore, in addition to calculating the pose itself, it is often desirable to estimate how certain the pose determination is. This paper presents a method for estimating the uncertainty in determining the pose of an arbitrarily positioned object with three light-stripe range finders. We analyze the perturbation of a least squares fit of sensed three-dimensional (3-D) segments to object faces, and obtain a relationship between the sensing error and the object pose error. Experiments demonstrate that the method provides the estimate of accuracy in pose determination.**

I. Introduction

Recognizing the pose of a three-dimensional (3-D) object in a workspace is a fundamental task in many computer vision applications, including automated assembly, inspection, and bin picking. Many object recognition algorithms have been developed. However, there has been little attention given to estimating the uncertainty of object pose determinations. In this paper, we study a problem of estimating uncertainty in determining the pose of a polyhedral object when using multiple light-stripe range finders.

Simple light-stripe range finders are among the fastest and least expensive ways to acquire accurate range data. Multiple range finders viewing an object from different perspectives can usually provide enough constraints to determine the pose of the object. Imagine that a polyhedral object is placed at an arbitrary pose in the workspace and that we place three simple light-stripe range finders above the workspace. Based on an interpretation tree search technique, 3-D line segments obtained by the range finders can be assigned to model faces consistent with geometric constraints. Once a feasible interpretation is found that satisfies the geometric constraints for all line segments, the transformation from the model coordinate frame to the world coordinate frame is obtained by a least squares method.

As a result of sensing error, the transformation contains inaccuracies. Therefore, we need to estimate the pose uncertainty. Using an error analysis based on the convergence properties of the least squares fit, we obtain a relationship between the covariance matrix of the endpoint positions of line segments and the covariance matrix of the positions of object vertices. The pose uncertainty can then be estimated from this relationship.

Manuscript received August 20, 1993; revised June 22, 1994. This work was supported by the Avionics Laboratory, Wright Research and Development Center, Aeronautical Systems Division (AFSC), U.S. Air Force, Wright-Patterson AFB, under Contract F33615-90-C-1465, ARPA Order 7597-597. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. government. Portions of this paper were presented at the IEEE International Conference on Robotics and Automation, Atlanta, GA, 1993.

K. Kemmotsu was with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA. He is now with the Advanced Technology Research Center, Mitsubishi Heavy Industries, Ltd., Kanazawa-ku, Yokohama 236 Japan.

T. Kanade is with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213 USA.

IEEE Log Number 9411924.