April, 1987 LIDS-P-1663

AN ESTIMATION-BASED APPROACH TO THE RECONSTRUCTION OF OPTICAL FLOW

Anne Rougée*, Bernard C. Levy**, Alan S. Willsky**

Laboratory for Information and Decision Systems
M.I.T., Cambridge, MA. 02139

Abstract

The problem of reconstructing the apparent velocity field (optical flow) in a sequence of images is formulated as a linear estimation problem. Estimation-based interpretations are provided for well-known formulations and methods, allowing us to use the machinery of recursive estimation theory to construct both new and efficient algorithms for these problems and a flexible framework for the development of algorithms for modified or related problems. The first problem we address is the estimation of the velocity field along a moving contour given a stochastic model of this field and measurements of the component of velocity normal to the contour. The methods of 1-D linear smoothing theory provide recursive algorithms, in contrast to the iterative method of Hildreth for the same problem. We then consider the problem of estimating the optical flow inside a bounded domain, given an estimate on the boundary and observations inside the domain, which we formulate as an estimation problem for a 2-D boundary value stochastic process. The resulting estimator is then obtained as the solution of the same system of elliptic partial differential equations derived in a very different way by Horn and Schunck. We then develop an efficient implementation of this estimator using a recently developed local relaxation method.

* The work of this author was supported by I.N.R.I.A. (Institut National de la Recherche en Informatique et Automatique), Le Chesnay, France.

** The work of these authors was supported in part by the National Science Foundation under Grant ECS-83-12921, and in part by the Army Research Office under Grant DAAG-29-84-K-0005.

I. INTRODUCTION

The objective of this paper is to develop a stochastic estimation-theoretic framework for the reconstruction of the apparent velocity field, or optical flow, in a sequence of images. Specifically the approach that we follow is to construct a stochastic model for the optical flow, and then to use this, together with 1-D and 2-D estimation techniques to estimate the optical flow from available measurements. This method can be viewed as a model-based implementation of the regularization techniques which have been proposed recently [1] for ill-posed problems in computer vision. Indeed, as we will see, two well-known reconstruction methods can be exactly interpreted as stochastic estimation problems. Not only does this allow us to use the machinery of recursive estimation to develop efficient new algorithms, but it also provides us with a flexible framework in which a variety of other problems and new algorithms can be readily addressed and developed simply by modifying the model under consideration.

The problem of motion estimation has been a topic of interest in image processing since the early seventies, motivated by applications such as target tracking in the military domain, or motion compensation in television image coding. Also, applications such as robot navigation have clear needs for motion estimation, although often the interest is not directly in the velocity fiels itself but in some other information (possibly derived from the optical flow) such as depth, rigid body motion parameters, etc.

A main feature of motion estimation problems, in comparison with other image processing problems such as edge detection or object recognition, is that it requires explicitly the introduction of a physical model. Indeed, unlike other applications, in motion estimation the data is not produced by a single image, but from a sequence of images. It is therefore important to use models to describe the relation existing between these successive images or, in the case of rigid motion, to describe motion parameters such as rotation and translation vectors.

The fundamental equation for motion estimation is the brightness constraint, which relates the brightness function and the velocity at any point in the image through the constraint that the brightness of a particular moving point is constant in time. In an actual image sequence this imposes a model constraint on a first order approximation of the brightness difference in time. This single scalar equation does not allow the reconstruction of both components of the optical flow, and most of the methods found in the literature for solving this equation are regularization methods, which select a particular solution by minimizing an error criterion containing a regularity constraint for the reconstructed optical flow.

One of the first and most important reconstruction methods was introduced by Horn and Schunck [2], and several extensions were developed

thereafter by Cornelius and Kanade [3], Nagel [4] and others. According to this technique, the optical flow is estimated by minimizing a criterion which includes both the average error in satisfying the brightness constraint on the whole surface of the image and a regularity constraint for the gradient of the optical flow. Using the calculus of variations, Horn and Schunck [2] obtained a solution satisfying a system of elliptic partial differential equations (more precisely, a system of coupled Poisson equations). They used the Gauss-Seidel method to solve this system iteratively, but more recently Glazer [6] and Terzopoulos [7] have implemented multilevel relaxation methods, which are more efficient from a computational point of view.

Nevertheless, problems occur at discontinuities of the velocity field, due to occluding boundaries for example, which need to be treated beforehand. This has motivated researchers to develop estimation techniques where the objective is only to reconstruct the optical flow along a moving contour, instead of the whole domain, and where it is assumed that the normal component of the velocity field along the contour can be computed by local methods.

Hildreth [8] has implemented a regularization method based upon the same criterion as in Horn and Schunck [2], but restricted to a contour, and where the conjugate gradient method is used for minimizing this criterion. Another method was introduced by Bouthemy [9], using a totally different point of view: in a first step, moving edges and the corresponding perpendicular velocity components are simultaneously locally detected and estimated by hypothesis testing, avoiding the use of the brightness constraint which is an approximate equation and is not valid across discontinuities. Then, in a second step the complete velocity is recovered by using a stochastic gradient algorithm along the detected contour in order to minimize the error in the perpendicular velocity component.

Although the second of these methods is computationally superior, both methods are iterative. In this paper, by modeling the velocity field along the contour as a Brownian motion or Wiener stochastic process, it will be shown that precisely the same reconstruction problem considered by Hildreth can be formulated as a 1-D smoothing problem, for which a number of exact, recursive solutions are available (see Ljung and Kailath [10], [11]). The main advantage of these optimal estimation solutions is that they are non-iterative techniques, and require therefore considerably less computational effort than the two methods mentioned above.

Once the velocity field has been reconstructed along a closed contour, the next step is to estimate the field inside the bounded domain defined by this contour. In practice, this means that the detected edges have to be linked together, in order to segment the image into bounded domains which are homogeneous regions for the velocity. As was noted by Horn [5], the image segmentation and velocity estimation problems are not independent,

since in theory the image can be segmented only if the velocity field is already known. However some segmentation methods can be implemented without estimating explicitly the velocity beforehand (Bouthemy [12]).

Concerning this second problem, the approach considered here relies on the introduction of a 2-D internal stochastic model for the velocity field. The velocity field is modeled as a 2-D Brownian motion process, and the observations are given by the constraint relating the brightness function to the velocity at any point in the image. It will be shown that this approach generalizes the one considered for the contour problem.

This paper is organized as follows. In Section **II** we first investigate the case of a moving contour, and give new interpretations of the problem considered by Hildreth, first in terms of an optimal control problem and then, equivalently in terms of a 1-D fixed-interval smoothing problem. This allows us to use any of the variety of exact smoothing algorithms for this reconstruction problem. In Section **III**, we consider the case of a bounded domain defined by a closed contour on which the velocity field is given (or previously estimated), and formulate this problem as a linear estimation problem for a 2-D boundary value process. This leads us to a solution satisfying the same elliptic partial differential equations obtained by Horn and Schunck [2]. In Section **IV** we give some experimental results, where we vary the parameters of the 1-D and 2-D stochastic models. Finally, Section **V** contains some conclusions and some thoughts for further research.

II. ESTIMATION OF THE VELOCITY FIELD ON A CONTOUR

In this section, we consider the case of a moving contour, whose motion has been detected beforehand from a sequence of images, and on which it is only possible to estimate locally the normal component of the optical flow. The objective is to use this information to reconstruct the complete optical flow along the contour. As indicated in the preceding section, this problem has been addressed by Hildreth [8], and to begin we review the essential elements of her formulation.

Specifically we assume that at time t we are given a contour C of constant brightness in an image. Suppose that this contour is parametrized as

$$C = \{ (x(s),y(s)) , s \in [0,L] \}$$

where s denotes the arclength, and x(s), y(s) are the spatial coordinates in the image plane. The apparent local velocity vector at point s on C and at time t is defined by

$$\mathbf{V}(s) = (u , v)^T \tag{2.1}$$

with

$$u = dx/dt \quad , \quad v = dy/dt \tag{2.2}$$

Let the image brightness at point (x,y) in the image plane at time t be denoted by E(x,y,t). The brightness is assumed to be constant in time on the contour C, so that, taking a first order approximation of its global differential in time, we obtain the so called brightness constraint equation

$$E_x u + E_y v + E_t = 0$$

where the subscripts x, y and t denote the partial derivatives with respect to x, y and t respectively. This equality can be rewritten as

$$(\nabla E)^T \mathbf{V} = - E_t \tag{2.3}$$

where ∇E is the gradient of E(x,y,t) with respect to the spatial coordinates (x,y), i.e. $\nabla E = (E_x,E_y)^T$. Note that ∇E is perpendicular to the contour C, since the brightness function is constant on C. Consequently, only the component of **V** perpendicular to the contour C can be estimated by local methods, i.e. methods based on local computations of approximations to E_x, E_y and E_t and the subsequent use of (2.3) to estimate the normal component of **V** at a point on the contour. Therefore some type of global computation is needed to allow complete reconstruction.

Let us introduce some notation. Specifically, let z(s) denote the normal component of the velocity field on the contour. This can be expressed as

$$\mathbf{n}(s)^T \mathbf{V}(s) = z(s) \tag{2.4}$$

where $\mathbf{n}(s)$ denotes the unit vector perpendicular to the contour, and where from (2.3) we note that

$$z(s) = - E_t / \| \nabla E \| \tag{2.5a}$$

and

$$\mathbf{n}(s) = (E_x , E_y)^T / \| \nabla E \| \ . \tag{2.5b}$$

In order to perform the required reconstruction we need to impose an additional constraint or an optimization criterion. One method proposed by Hildreth [8] is to use a criterion which includes a tradeoff between the accuracy with which the reconstructed field $\mathbf{V}(s)$ matches the measurements (2.4) (i.e. how accurately we match the brightness constraint) and the smoothness of the velocity estimate. Specifically, Hildreth addresses the problem of determining $\mathbf{V}(s)$ to minimize

$$J = 1/2 \int_C (a \| e(s) \|^2 + \| d\mathbf{V}/ds \|^2) \, ds \tag{2.6}$$

where

$$e(s) = z(s) - \mathbf{n}(s)^T \mathbf{V}(s) \ , \tag{2.7}$$

and a is a weighting factor, controlling the relative importance of the two terms in the criterion[1].

As a first observation, let us note that the problem we have just posed is what is known in optimal control theory as an optimal tracking problem. Specifically, if we define the "control variable" $\mathbf{U}(s)$ by

$$d\mathbf{V}/ds = \mathbf{U}(s) \tag{2.8}$$

and the tracking error by (2.7), we see that (2.6) is a weighted sum of squared tracking error and control energy $\| \mathbf{U}(s) \|^2$. A variety of forms of the solution to this problem can be found in the control literature (see, for example, [13]). In particular, one algorithm yielding the optimal

[1]The subscript C here indicates integration along the contour, from $s = 0$ to $s = L$.

reconstruction is the following. We first solve the Riccati matrix differential equation

$$d\Theta/ds = \Theta(s)^2 - a\,\mathbf{n}(s)\mathbf{n}(s)^T \tag{2.9}$$

backward from s = L to s = 0, using the final condition $\Theta(L) = 0$. Here $\Theta(s)$ is a 2x2 matrix (which in fact is symmetric and positive semi-definite). Simultaneously we solve the linear differential equation

$$d\mathbf{q}/ds = \Theta(s)\mathbf{q}(s) - a\,z(s)\mathbf{n}(s) \tag{2.10}$$

backward from the final condition $\mathbf{q}(L) = 0$. Here $\mathbf{q}(s)$ is a 2-dimensional vector. The optimal reconstruction of the optical flow is then obtained by the <u>forward</u> integration of the linear differential equation

$$d\hat{\mathbf{V}}/ds = -\Theta(s)\hat{\mathbf{V}}(s) + \mathbf{q}(s) \tag{2.11a}$$

starting from the initial condition

$$\hat{\mathbf{V}}(0) = \Theta(0)^{-1}\mathbf{q}(0) \quad . \tag{2.11b}$$

It is also possible to give this same problem and the resulting algorithms an estimation-theoretic interpretation which we will find useful in the sequel. In particular, as shown by Mayne [15] and Bryson and Frazier [14], the problem of minimizing (2.6) is equivalent to a 1-D fixed interval smoothing problem. Specifically, consider the following stochastic model of $\mathbf{V}(s)$

$$d\mathbf{V}/ds = \mathbf{U}(s) \tag{2.12}$$

where $\mathbf{U}(s)$ is a white noise process of unit intensity, i.e.

$$E[\,\mathbf{U}(s)\,\mathbf{U}(s')^T\,] = I\,\delta(s-s') \tag{2.13}$$

where I is the 2x2 identity matrix. Also, suppose we model our measurements as

$$z(s) = \mathbf{n}(s)^T\,\mathbf{V}(s) + e(s) \quad , \quad 0 \leqslant s \leqslant L \tag{2.14}$$

where $e(s)$ is also a white noise process, uncorrelated with $\mathbf{V}(s)$, with

$$E[\,e(s)\,e(s')^T\,] = 1/a\,\delta(s-s') \quad . \tag{2.15}$$

The objective is to compute the linear least-squares estimate of $\mathbf{V}(s)$ given the data $z(s)$, $0 \leqslant s \leqslant L$, i.e. to compute

$$\hat{\mathbf{V}}_S(s) = E\{ \mathbf{V}(s) \mid z(\sigma), 0 \leqslant \sigma \leqslant L \} \quad . \tag{2.16}$$

As shown in [14], [15], this estimate is identical to the reconstructed velocity $\hat{\mathbf{V}}(s)$ obtained from the algorithm (2.9)-(2.11).

Having established this relationship between the formulation in [8] and optimal smoothing, we are in a position to make a number of additional observations. First of all, note that the model (2.12)-(2.15) provides us with a stochastic interpretation of the criterion (2.6). Specifically the observation equation (2.14) is nothing other than the brightness constraint with some allowance for error in our measurement of the normal component of $\mathbf{V}(s)$. More precisely, the first term in (2.6) corresponds to the measurement equation (2.14), (2.15) with the error $e(s)$ modeled as white noise with intensity equal to the reciprocal of the weighting factor a. Next, consider the model (2.12) for $\mathbf{V}(s)$. Here we see that the second term in (2.12) corresponds to modeling $\mathbf{V}(s)$ as an independent increment or random walk process: the change in velocity $\mathbf{V}(s_2)-\mathbf{V}(s_1)$ with $s_1 \leqslant s_2$ is independent of $\mathbf{V}(s_1)$.

An important feature of this estimation-based framework is that it is based on such explicit models. This allows one to critique model assumptions and make modifications which directly lead to changes in the estimation algorithm that affect the details of the equations to be implemented but not the form or the complexity of the algorithm. For example, suppose that one wished to use a spatially-varying weighting factor $a(s)$ in (2.6). This might arise if there were differing contrasts at different points along the contour, leading to varying accuracies in the measurement of the normal component of $\mathbf{V}(s)$. In our estimation framework this would correspond to replacing a with $a(s)$ in the model (2.15) and in the two differential equations (2.9), (2.10) that form the backward sweep of the algorithm. Similarly, suppose we wished to modify (2.12). For example, suppose that we wished to introduce an "oriented smoothness constraint" such as introduced by Nagel [4], [24], i.e. we replace the second term in (2.6) by

$$(d\mathbf{V}/ds)^T \, W(s) \, (d\mathbf{V}/ds) \tag{2.17}$$

where $W(s)$ is a (possibly spatially-varying) 2x2 weighting matrix. What this corresponds to in our model is a modification of (2.13) where

$$E[\mathbf{U}(s) \, \mathbf{U}(s')^T] = W(s)^{-1} \, \delta(s-s') \quad . \tag{2.18}$$

The algorithm (2.9), (2.10) in this case becomes

$$d\Theta/ds = \Theta(s)W(s)^{-1}\Theta(s) - a\,\mathbf{n}(s)\mathbf{n}(s)^T \tag{2.19}$$

$$d\mathbf{q}/ds = \Theta(s)W(s)^{-1}\mathbf{q}(s) - a\,z(s)\mathbf{n}(s) \tag{2.20}$$

both integrated backward from $\Theta(L) = 0$ and $\mathbf{q}(L) = 0$, respectively, and

$$d\hat{\mathbf{V}}/ds = -W(s)^{-1}\Theta(s)\hat{\mathbf{V}}(s) + W(s)^{-1}\mathbf{q}(s) \tag{2.21}$$

integrated forward from the initial condition (2.11b).

Another potentially useful modification to the dynamics involves building in a different correlation structure for $\mathbf{V}(s)$. Specifically, the independent increment property of (2.12) is an idealization that may or may not be accurate enough for its intended purpose. Note, for example, that $\mathbf{V}(s)$ as defined in (2.12) is non-stationary with a growing variance. As an alternative, one might prefer a model admitting finite-variance stationary statistics such as

$$d\mathbf{V}/ds = -b\,\mathbf{V}(s) + \mathbf{U}(s) \tag{2.22}$$

Here $1/b$ has an interpretation as a correlation length, i.e. one expects high correlation between values of $\mathbf{V}$ at points less than $1/b$ apart along the contour. The optimal reconstruction algorithm for this model (i.e. (2.22) and (2.13)-(2.15)) is

$$d\Theta/ds = -2b\,\Theta(s) + \Theta(s)^2 - a\,\mathbf{n}(s)\mathbf{n}(s)^T \tag{2.23}$$

$$d\mathbf{q}/ds = [\,\Theta(s) - b\,\mathbf{I}\,]\,\mathbf{q}(s) - a\,z(s)\mathbf{n}(s) \tag{2.24}$$

again integrated backward from final values of 0 at $s = L$, and

$$d\hat{\mathbf{V}}/ds = [\,b\,\mathbf{I} - \Theta(s)\,]\,\hat{\mathbf{V}}(s) + \mathbf{q}(s) \tag{2.25}$$

integrated forward from (2.11b) at $s = 0$.

Another aspect of our estimation formulation is that it immediately makes available to us a number of methods of implementation. In particular (2.9)-(2.11) can be identified as (a spatially reversed version of) the Rauch-Tung-Streibel [17] form of the optimal smoother. This form has a "two-sweep" structure, i.e. we process the data in one direction and then sweep back in the other. There are other implementations such as two filter formulas and the innovations form that have different algorithmic structures (see [10], [11], [15], [16]). For example, the two filter form allows a parallel implementation: as before, we solve (2.9), (2.10) backward from final values of 0 at $s = L$; however <u>simultaneously</u> we solve

$$d\Psi/ds = -\Psi(s)^2 + a\,\mathbf{n}(s)\mathbf{n}(s)^T \tag{2.26}$$

and

$$d\mathbf{r}/ds = -\Psi(s)\,\mathbf{r}(s) + a\,z(s)\mathbf{n}(s) \tag{2.27}$$

forward from initial values $\Psi(0) = 0$, $\mathbf{r}(0) = 0$ at $s = 0$. We can then compute the reconstructed estimate as

$$\hat{\mathbf{V}}(s) = P(s)\,[\,\mathbf{q}(s) + \mathbf{r}(s)\,] \tag{2.28}$$

where

$$P(s) = [\,\Theta(s) + \Psi(s)\,]^{-1} \quad . \tag{2.29}$$

Note that the initial values on $\Psi(0)$ and $\mathbf{r}(0)$ correspond to an assumption that we have no a priori information on the velocity field. Typically we may have some information, such as knowledge of the maximum possible velocity. This can be translated into an assumption that $\mathbf{V}(0)$ has some mean m_0 and covariance Π_0. In this case, the initial conditions for (2.26), (2.27) would be $\Psi(0) = \Pi_0^{-1}$, $\mathbf{r}(0) = \Pi_0^{-1} m_0$.

The various quantities in this form of the smoother have important interpretations. Specifically $\Psi(s)^{-1}\mathbf{r}(s)$ is the optimal estimate of $\mathbf{V}(s)$ given the data $z(\sigma)$ for $0 \leqslant \sigma < s$, while $\Theta(s)^{-1}\mathbf{q}(s)$ is the corresponding estimate based on $z(\sigma)$, for $s < \sigma \leqslant L$. Also $\Psi(s)^{-1}$ and $\Theta(s)^{-1}$ are the corresponding error covariances for these estimates. Thus (2.28) can be viewed as a weighted average of these two estimates, and $P(s)$ is the estimation error covariance in estimating $\mathbf{V}(s)$ based on all of the available data. Note that this quantity provides us with a precise measure of how well we can estimate $\mathbf{V}(s)$. In fact, this is closely connected with Brockett's recent paper [25] relating estimation of optical flow to the concept of observability that has proven useful in the analysis of models such as those we have discussed. For example, consider the special case in which $\mathbf{V}(s)$ is constant, corresponding to rigid translational motion. This would correspond to our model (2.12), (2.18) with $W(s)^{-1} = 0$ (so that $d\mathbf{V}/ds = 0$), and a simple computation (assuming $\Psi(0) = 0$) shows that in this case

$$P(s)^{-1} = P^{-1} = a \int_C \mathbf{n}(s)\mathbf{n}(s)^T\,ds \tag{2.30}$$

The integral on the right-hand side of (2.30) is precisely Brockett's "observability grammian", which provides a quantitative way to measure the effect the shape of an object has on one's ability to estimate optical flow. For

example, if the contour is a straight line, P^{-1} is obviously singular indicative of the fact that it is impossible in this case to estimate the tangential velocity component. While such a simple grammian interpretation isn't available in the general case (with $d\mathbf{V}/ds \neq 0$), P(s) still provides a precise measure of how well we can estimate $\mathbf{V}(s)$. Also, one can draw some intuition. For example, if $\mathbf{V}(s)$ is given by the model (2.22), accurate estimation of $\mathbf{V}$ would require a normal $\mathbf{n}(s)$ that changes direction significantly over distances along the curve on the order of $1/b$.

Finally, it is worth noting that none of the formulations we have described, from (2.6) on, make use of the fact that the contour under consideration is closed or not. Indeed our stochastic interpretation clearly shows that the reconstructed estimate will <u>not</u> have $\hat{\mathbf{V}}(0) = \hat{\mathbf{V}}(L)$ even if the curve is closed. In order to address this problem we can use the recently-developed results of Adams, et al. [18] on estimation for boundary value models. In particular, consider the model (2.12)-(2.15) with one additional constraint, namely

$$\hat{\mathbf{V}}(0) = \hat{\mathbf{V}}(L) \quad . \tag{2.31}$$

The results in [18] provide one form of the optimal reconstruction algorithm for this problem. This solution, which is similar to but somewhat more complex than the two filter algorithm (2.9), (2.10), (2.26)-(2.28) for reconstruction without the constraint (2.31), is presented in Appendix A.

III. ESTIMATION OF THE VELOCITY FIELD IN A BOUNDED DOMAIN

We now investigate the problem of reconstructing the velocity field in a bounded region in an image given the value or an estimate of the value of the velocity on the boundary. As in the preceding section we will adopt an estimation-theoretic formulation of this problem, in this case building on the work of Adams et al. [18] on estimation for boundary-value models in one and several dimensions. The result of this development is a generalization of the formulation and methods of Horn and Schunck [2].

We assume that we are given a bounded, simply connected region D of the image plane at some time t, with a smooth boundary C. The objective is to reconstruct the velocity field **V** inside D based on the observed brightness changes within the region and possibly some knowledge of **V** on C. This last point deserves some comment. The region D may correspond to an identified object in the field of view, obtained for example by some edge detection and image segmentation scheme, or it may be a more-or-less arbitrarily specified patch in the image. In the former case, C is the object boundary and the assumption of constant brightness along C is reasonable. Consequently, in this case one can apply the method of Section **II** to obtain estimates of **V** along C before applying the method described in this section. If D is an arbitrary region, such prior estimates along C may not be available (indeed this is the case considered by Horn and Schunck). Our formulation will capture both of these cases.

The stochastic model we use in this section is the direct 2-D counterpart of the model (2.12)-(2.15) introduced in the preceding section. Specifically, let ∇ denote the gradient operator, i.e.

$$\nabla f = (f_x , f_y)^T \quad . \tag{3.1}$$

The gradient of the vector **V** in (2.1), (2.2) is defined to be the composite vector consisting of the gradients of the components of **V**, i.e. we define the operator L as

$$L\mathbf{V} = \begin{bmatrix} \nabla u \\ \\ \nabla v \end{bmatrix} = (u_x , u_y , v_x , v_y)^T \quad . \tag{3.2}$$

The differential operator L can be formally expressed as

$$L = (I \otimes \nabla) \tag{3.3}$$

where $\otimes$ denotes the Kronecker product, which indicates that the gradient operates on each component of **V**. The model for **V** is then given by

$$L\mathbf{V}(x,y) = \mathbf{U}(x,y) \tag{3.4}$$

where $\mathbf{U}(x,y)$ is a 2-D stochastic process. The specific model we use here for $\mathbf{U}(x,y)$ is

$$\mathbf{U}(x,y) = L\mathbf{B}(x,y) \tag{3.5a}$$

where $\mathbf{B}(x,y)$ is a 2-vector whose components $B_1(x,y)$ and $B_2(x,y)$ are independent 2-D Brownian motions [26]. The covariance structure of $B_i(x,y)$ is given by

$$E[\,B_i(x_1,y_1)B_i(x_2,y_2)\,] = (x_1^2+y_1^2)^{1/2} + (x_2^2+y_2^2)^{1/2} - [\,(x_1-x_2)^2+(y_1-y_2)^2\,]^{1/2} \quad . \tag{3.5b}$$

It is straightforward to check that $\nabla B_i(x,y)$ is a 1-D white noise process of unit intensity along straight lines.

The boundary condition associated with (3.4) corresponds to our prior knowledge of $\mathbf{V}$ on C, i.e.

$$\mathbf{V}(x,y) = \mathbf{V}_C(x,y) \quad , \quad (x,y) \in C \tag{3.6}$$

where $\mathbf{V}_C$ is a Gaussian white noise process on C with mean $\hat{\mathbf{V}}_C$ and intensity matrix P_C. The form of this condition allows us to accomodate a variety of situations. In particular, at one extreme $P_C = 0$ corresponds to perfect knowledge of $\mathbf{V}_C$, while $P_C = \infty$ implies no prior information about $\mathbf{V}_C$. In some cases, such as when some boundaries are occluded or a set of disconnected edges are linked to form a closed contour, there may be portions of C along which $\mathbf{V}_C$ can be estimated and others where it can't. This can be captured by using a spatially-varying P_C.

As in the previous section, the brightness constraint (2.3) provides us with our measurements. Specifically, we have available observations of $E_t(x,y)$, the time rate of change of intensity at points throughout D. If we let $z(x,y) = -E_t(x,y)$, the brightness constraint yields the measurement equation

$$z(x,y) = H(x,y)\mathbf{V}(x,y) + e(x,y) \tag{3.7}$$

where

$$H = (E_x, E_y) \tag{3.8}$$

and e(x,y) is a 2-D white noise process of intensity 1/a included as before to model the presence of brightness variations as well as errors in calculating E_x, E_y, and E_t.

Our problem, then is to estimate the process **V** specified by (3.4)-(3.6) given the measurements (3.7). This problem is precisely of the type investigated by Adams, et al. [18]. In Appendix B we briefly review the elements of their method for constructing the equations satisfied by the optimal estimator for boundary-value processes and apply it to the present problem. The result is that the optimal estimates $\hat{u}$, $\hat{v}$ satisfy the following system of partial differential equations

$$\Delta\hat{u} = a E_x (E_t + E_x\hat{u} + E_y\hat{v}) \tag{3.9}$$

$$\Delta\hat{v} = a E_y (E_t + E_x\hat{u} + E_y\hat{v}) \quad . \tag{3.10}$$

with the boundary condition

$$\begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} + P_C \begin{bmatrix} \partial\hat{u}/\partial n \\ \partial\hat{v}/\partial n \end{bmatrix} = \hat{\mathbf{V}}_C \quad \text{on } C \tag{3.11}$$

where Δ is the laplacian, i.e.

$$\Delta f = \partial^2 f/\partial x^2 + \partial^2 f/\partial y^2 \tag{3.12}$$

and where $\partial\hat{u}/\partial n$ and $\partial\hat{v}/\partial n$ denote the normal derivatives of $\hat{u}$ and $\hat{v}$, respectively.

These equations deserve some comment. First note that (3.9), (3.10) are precisely the same PDE's as those obtained by Horn and Schunck [2] who derived them using the calculus of variations applied to the minimization of the quadratic criterion

$$J = 1/2 \int_D (a [z(x,y) - H(x,y)\mathbf{V}(x,y)]^2 + \| L\mathbf{V}(x,y) \|^2) \, dx \, dy \quad . \tag{3.13}$$

Note also that the boundary condition (3.11) is in general of a mixed type. At one extreme, is the case $P_C = 0$, i.e. when $\mathbf{V}_C$ is assumed to be known perfectly. In this case (3.11) is simply a Dirichlet type condition

$$\hat{\mathbf{V}} = \hat{\mathbf{V}}_C \quad . \tag{3.14}$$

At the other extreme is the case $P_C = \infty$, corresponding to no information about **V** on the boundary. In this case (3.11) is a Neumann type condition

$$\partial \hat{\mathbf{V}}/\partial n = 0 \quad \text{on } C \ . \tag{3.15}$$

This is the condition used by Horn and Schunck in their study. The condition (3.11) generalizes these cases to accomodate prior but imperfect information on the boundary.

Equations (3.9), (3.10) are coupled Poisson equations. In Horn and Schunck [2], the Gauss-Seidel method was used to solve this system of equations. However, one can expect that faster methods such as overrelaxation or multigrid methods might provide greater efficiency. Although the use of such sophisticated PDE solvers may appear inappropriate in image processing, where fast solutions are usually desired, it is useful to keep in mind the fact that the solutions that we seek do not have severe accuracy requirements, i.e. the number of significant digits required is much smaller than in typical PDE applications, and if we use very efficient overrelaxation or multigrid PDE solvers, only a few iterations will be needed to obtain a good estimate of the optical flow. In addition, both overrelaxation and multigrid methods may be implemented in parallel on special purpose computers, so that speed is unlikely to limit the applicability of the optical flow estimation procedure described above. A method of this type is described and used in the next section.

Note that, as in the previous section, the introduction of a stochastic model-based approach to optical flow estimation provides a flexible framework for considering modifications to the problem formulation by modifying the model and for constructing the resulting estimation algorithms. For example, one can readily derive the modifications to (3.9)-(3.11) that would result from the introduction of an oriented smoothness constraint (which would enter as a spatially-varying intensity for the process $\mathbf{U}$ in (3.4)) or from the use of a different model for $\mathbf{V}$ that yields stationary statistics and an identifiable correlation length for the optical flow (much as (2.22) did in the previous section).

Finally, it is interesting to note that the 2-D model (3.4), (3.5) used in this section is consistent with the 1-D model (2.12), used in the previous section, with the exception that the intensity of the white noise process in (2.12) that is obtained by restricting (3.4), (3.5) to C is spatially-varying in intensity, with an "oriented intensity" determined by the curve C. Specifically, along the contour we have the following relation

$$\begin{aligned} d\mathbf{V}/ds &= (u_x\, dx/ds + u_y\, dy/ds \,,\, v_x\, dx/ds + v_y\, dy/ds)^T \\ &= (\mathbf{t}(s)^T \nabla u \,,\, \mathbf{t}(s)^T \nabla v)^T \\ &= (I \otimes \mathbf{t}(s)^T) \nabla \mathbf{V} \end{aligned} \tag{3.16}$$

where $\mathbf{t}(s)$ denotes the unit tangent to the contour C, i.e.

$$\mathbf{t}(s) = (dx/ds , dy/ds)^T \quad . \tag{3.17}$$

Using (2.12) and (3.4), (3.5) we have

$$\mathbf{U}(s) = (I \otimes \mathbf{t}(s)^T) \, \mathbf{LB} \quad . \tag{3.18}$$

IV. EXPERIMENTAL RESULTS

IV.1. Contour case

In this section we present several examples illustrating the reconstruction of the optical flow along a contour. In these examples we used two different polygonal contours and consider the reconstruction of optical flow for both translational and rotational rigid body motions. In Figures 1 and 2 we illustrate the case of pure translation. Part (a) of each figure depicts the true velocity fields along the contours, while the exact normal components are illustrated in part (b). In these figures we have depicted all of the points along the curves at which velocity values were sampled and reconstructed, so that the optical flow is more densely sampled in Figure 1 than in Figure 2. Part (c) of these figures illustrates the reconstructed velocity estimates when the normal components are measured perfectly. The algorithm used in this case is that given in (2.9)-(2.11). In this case the reconstructed estimates are perfect for any nonzero value of a. In Figure 3 we illustrate the effect that random errors in measuring the normal component have on the reconstruction of the velocity field in Figure 2a. In this case the value of a (which we interpret as the reciprocal of the noise intensity), controls the amount of smoothing and noise rejection that is done (larger values of a correspond to less smoothing). As one would expect, increased noise intensity leads to degraded performance but the smoothing performed by the estimator minimizes this degradation.

Figures 4 and 5 illustrate the case of rotational motion. In all of the experiments represented in this figure noise-free measurements of the normal component were used. Note that in practice this does not mean that the measurement z(s) is perfect, since there is the error introduced by the (first-order) approximation to the brightness constraint used to specify the measurements at the discrete sampling points in space and time. In each of these figures we illustrate the effect of using different values of a. In particular, Figures 4c and 4d depict the estimated velocity and the error when a = 0.95, while Figures 4e and 4f correspond to the value a = 0.1. Typically in examples such as this we have found a value of a between 0.5 and 1.0 yields to the best results. Note, however, that the estimates in Figure 4, while quite good, are not perfect even with perfect measurements. This is even more pronounced in Figure 5. The reasons for this behavior are several. First there is the brightness constraint approximation mentioned previously. A second, more fundamental reason relates to the model (2.12). Note that if we take (unconditional) expected values in this model we obtain

$$d\bar{\mathbf{V}}/ds = 0 \qquad (4.1)$$

(where $\bar{\mathbf{V}} = E[\mathbf{V}]$), i.e. an interpretation of (2.12) is as a random walk about a constant mean velocity. The perfect performance of the smoother in the case of translational motion can be traced to the fact that the actual velocity field satisfies (4.1). On the other hand, this is not true for rotation, so the smoothing and interpolation performed by optimal estimator do well but not perfectly.

Finally, we recall that all of the estimates up to this point have been computed using the algorithm (2.9)-(2.11). This algorithm, which minimizes the criterion used by Hildreth, does not account for the periodicity in the velocity field. In Figure 6 we illustrate the application of the estimator described in Appendix A that constrains $\hat{\mathbf{V}}(0)$ to be equal to $\hat{\mathbf{V}}(L)$. By comparing Figures 5 and 6, we see that the estimator with the periodicity constraint distributes more evenly the estimation errors along the contour C.

IV.2 Velocity estimation in a bounded region

In this section we present several examples of velocity reconstruction inside a bounded set which is taken to be either a rectangle or a convex polygon (see, for example, Figure 7). In each example two successive images are generated and noise is added to the intensity gradient and its time derivative which are computed numerically from the two images. To solve the Poisson equations (3.9), (3.10), a local relaxation procedure developed by Kuo, Levy and Musicus [19] was implemented. The main feature of this method is that it allows the optimal relaxation parameter to be space-dependent, and it is therefore very convenient for space-variant PDEs such as (3.9), (3.10). To implement this method, the domain and its boundary were discretized on a uniform grid with a grid spacing equal to h (for clarity in the figures we depict only one point in 10 (rectangle) or one point in 5 (polygon) in each direction). Then, the local relaxation procedure, which is based on a red/black or checkerboard partition of the domain, can be expressed as follows.

For each iteration n:

- Red points (i+j is even):

$$u_{i,j}^{(n+1)} = (1 - w_{i,j})\, u_{i,j}^{(n)} + w_{i,j}\, d_{i,j}^{-1} \left(u_{i-1,j}^{(n)} + u_{i+1,j}^{(n)} + u_{i,j-1}^{(n)} + u_{i,j+1}^{(n)} - a h^2 E_x (E_y v_{i,j}^{(n)} + E_t) \right) \tag{4.2}$$

$$v_{i,j}^{(n+1)} = (1 - w'_{i,j})\, v_{i,j}^{(n)} \tag{4.3}$$

$$+ w'_{i,j}\, d'^{-1}_{i,j} \left(v_{i-1,j}^{(n)} + v_{i+1,j}^{(n)} + v_{i,j-1}^{(n)} + v_{i,j+1}^{(n)} \right.$$

$$\left. - a\, h^2 E_y \left(E_x u_{i,j}^{(n+1)} + E_t \right) \right) \quad .$$

- Black points (i+j is odd):

$$u_{i,j}^{(n+1)} = \left(1 - w_{i,j} \right) u_{i,j}^{(n)} \tag{4.4}$$

$$+ w_{i,j}\, d_{i,j}^{-1} \left(u_{i-1,j}^{(n+1)} + u_{i+1,j}^{(n+1)} + u_{i,j-1}^{(n+1)} + u_{i,j+1}^{(n+1)} \right.$$

$$\left. - a\, h^2 E_x \left(E_y v_{i,j}^{(n)} + E_t \right) \right)$$

$$v_{i,j}^{(n+1)} = \left(1 - w'_{i,j} \right) v_{i,j}^{(n)} \tag{4.5}$$

$$+ w'_{i,j}\, d'^{-1}_{i,j} \left(v_{i-1,j}^{(n+1)} + v_{i+1,j}^{(n+1)} + v_{i,j-1}^{(n+1)} + v_{i,j+1}^{(n+1)} \right.$$

$$\left. - a\, h^2 E_y \left(E_x u_{i,j}^{(n+1)} + E_t \right) \right) \quad .$$

In the above equations the coefficients $d_{i,j}$ and $d'_{i,j}$ are given by

$$d_{i,j} = 4 + a\, h^2 E_x^2 \tag{4.6}$$

$$d'_{i,j} = 4 + a\, h^2 E_y^2 \quad . \tag{4.7}$$

Here $w_{i,j}$ and $w'_{i,j}$ are the local relaxation parameters, the optimal value of which depend on the region and on the precise nature of the boundary conditions. In particular, for the case of Dirichlet conditions these parameters are given by

$$w_{i,j} = 2 \left(1 + \sqrt{1 - r_{i,j}^2} \right)^{-1} \tag{4.8}$$

$$w'_{i,j} = 2 \left(1 + \sqrt{1 - r'^2_{i,j}} \right)^{-1} \tag{4.9}$$

where in the rectangle case, corresponding to N=KxL points, the optimal choice for the parameters $r_{i,j}$ and $r'_{i,j}$ is the following

$$r_{i,j} = 2\, d_{i,j}^{-1} \left(\cos(\pi/(K+1)) + \cos(\pi/(L+1)) \right) \tag{4.10}$$

$$r'_{i,j} = 2\, d'^{-1}_{i,j} \left(\cos(\pi/(K+1)) + \cos(\pi/(L+1)) \right) \tag{4.11}$$

and in the polygon case we have chosen

$$r_{i,j} = 4\, d_{i,j}^{-1} \cos(\pi/\delta) \tag{4.12}$$

$$r'_{i,j} = 4\, d'^{-1}_{i,j} \cos(\pi/\delta) \tag{4.13}$$

where δ is the diameter of the domain considered, measured as the number of sample points. The theoretical study made in Kuo et al. [19] shows that this iterative method converges for a number of iterations proportional to the square root of the number of points N in the domain, instead of N for the Gauss-Seidel method.

In Figures 7-10 we depict a number of examples of reconstruction of rigid body translational (Figs. 7-8) and rotational motion (Figs. 9-10). In Figures 7 and 9 the intensity variation within the regions was taken to be linear in both directions, i.e.

$$E(x,y,t) = E_0 + E_1 x + E_2 y \tag{4.14}$$

while in Figures 8 and 10, the variation is one period of a sinusoid, i.e.

$$E(x,y,t) = E_0 + E_1 \cos(2\pi x/K) \cos(2\pi y/L) \tag{4.15}$$

where K, L are the dimensions of the moving object (K = 50 and L = 80 for the rectangle, K = 35 and L = 45 for the polygon), E_0 = 100 and E_1, E_2 are constants such as E(x,y,t) varies from 80 to 120.

In all of these examples we used the Dirichlet conditions, i.e. the estimator assumes it has perfect knowledge of the velocity on the boundary. For the rectangular region we provided the estimator with these exact values; however for the polygonal example we used estimated values along the boundary obtained by an application of method developed in Section II. The results obtained in all cases are quite satisfactory. For the case of the rectangle a good estimate is obtained after only 25 iterations for almost 4000 points. For the polygon we obtained a good estimate after only 15 or 20 iterations for 1121 points.

In Table 1 we present the relative errors corresponding to the results shown in Figs. 7-10 for the case of linear intensity variation, and, for comparison, we also present those obtained with the same number of iterations but with a weighting factor a equal to zero. Note that a= 0 corresponds to completely ignoring the measurements. That is, with a = 0, (3.9), (3.10) reduce to

$$\Delta\hat{u} = 0 \tag{4.16a}$$

$$\Delta\hat{v} = 0 \tag{4.16b}$$

so that the estimator is simply extrapolating inward from the specified boundary condition using Laplace's equation. If the values on the boundary

are known perfectly and if the spatial variation of the velocity is linear or nearly so, the reconstruction using a = 0 should be quite good. Table 1 verifies this. Note, however, that in the case of translational motion of the rectangle the estimation error should be zero since the values on the boundary are available without error and the velocity field is constant. The nonzero value in the table results from using only a finite number of iterations. Intuitively, if all of the information to be used comes from the boundary, it will take a significant number of iterations before this information propagates to the center of the region. By using a nonzero value of a, we allow the estimator to use local information which is available immediately. Thus we see that an additional benefit of setting $a \neq 0$ (beyond its use in controlling smoothing and noise rejection) is faster convergence.

As the preceding discussion indicates, linear or nearly linear variations in the velocity field are well-approximated by (4.16) and therefore can be accurately reconstructed from knowledge of the velocity field on the boundary. In Figure 11 we present an example in which the velocity field varies in a non linear fashion corresponding to a distortion of the interior of the rectangle. In particular, the velocity field in Figure 11a is given by

$$\begin{bmatrix} u \\ v \end{bmatrix} = \left[\left(\max \left(2|x|/K , 2|y|/L \right) \right)^{-\alpha} - 1 \right] \begin{bmatrix} x \\ y \end{bmatrix} \tag{4.17}$$

where K = 50, L = 80 are the dimensions of the rectangle and $\alpha = 0.2$. Note that in this case the velocity field is exactly zero on the boundary, so that the a = 0 reconstruction is useless. On the other hand, the reconstruction using a = 0.3 is quite good.

Finally in Figure 12 we reconsider the estimation of the velocity field of Figure 9. However in this case we take as boundary values the estimates produced using the method of Section II. Also, in this case, we use the mixed boundary conditions in (3.11), reflecting the fact that these estimates are imperfect. In this case optimization of the local relaxation parameters is somewhat more complicated. In a recent paper [19] Kuo et al. have investigated this case and we have applied their method. If we assume that P_C is a constant diagonal matrix, the optimal local relaxation parameters are given by (4,8), (4.9) with the following choice for the parameters $r_{i,j}$ and $r'_{i,j}$

$$r_{i,j} = 2\, d_{i,j}^{-1} \left(\cos(f_1/(K+1)) + \cos(f_2/(L+1)) \right) \tag{4.18}$$

$$r'_{i,j} = 2\, d'^{-1}_{i,j} \left(\cos(f_1/(K+1)) + \cos(f_2/(L+1)) \right) \tag{4.19}$$

where, if p_1 and p_2 denote respectively the constant diagonal values of P_C along the vertical and horizontal edges of the rectangle, the frequencies f_1 and f_2 are determined by

$$tg(f_j) = -2\, p_j f_j \left(1 - p_j^2 f_j^2 \right)^{-1} \qquad (4.20)$$

for $j = 1,2$. In the figure we present two results corresponding to two different values of P_C in (3.11) and therefore reflecting different levels of reliance on the boundary estimates. The specific values of P_C used in these tests were $P_C = I$ and $P_C = 0.01\ I$. These represent simplified approximations to the covariance resulting from the algorithm in Section **II** in that, while the actual P_C is diagonal for the rectangle case, it is not necessarily a constant multiple of the identity. The results in Figure 12 display the expected behavior: for large values of P_C one relies less on the boundary values (by using a larger value of a, in this case 0.006 vs 0.003). Associated with this is a predictable degradation in performance with increasing P_C due to the reduced quality and use of the information on the boundary.

V. CONCLUSION

In this paper we have developed an estimation-based approach to the reconstruction of optical flow in image sequences. We have considered the reconstruction of the velocity field both on a contour and inside a bounded region. Our results have several purposes. The first is that they provide stochastic model-based interpretation of the formulations studied by Hildreth [8] and Horn and Schunck [2], and a second is that we have been able to use our estimation-based perspective to develop new algorithms. In particular, the interpretation of Hildreth's formulation as an estimation problem immediately provides us with a number of recursive, non-iterative algorithms for reconstructing the optical flow along a contour involving in general two "sweeps" over the contour in opposite directions. This should be contrasted with the iterative conjugate gradient or adaptive stochastic gradient techniques of Hildreth [8] and Bouthemy [9] in which the contour needs to be scanned a number of times. Also, in the case of reconstruction within a region we have been able to use optimized iterative local relaxation methods to obtain efficient solution to the PDE's that must be solved.

While the development of these specific algorithms is not without interest, a more important point of this paper is the flexibility of the stochastic, model-based approach we have described. As we illustrated in Section II, it is quite simple in this framework to modify the models to incorporate additional information about the optical flow or about object motion and to modify the reconstruction algorithm in a similar manner. In addition, we also have at our disposal the full set of estimation-theoretic results and concepts that can allow us to develop procedures for related problems. For example, in Section II and Appendix A we have shown how the constraint of periodicity in the optical flow around a closed contour can be accommodated by applying recently-developed results on estimation of noncausal processes. It is also possible to use methods of adaptive estimation to identify parameters of our model (such as a spatially varying measurement noise parameter a(s) or an oriented-smoothness-like intensity $W^{-1}(s)$ for the white noise driving our model for the velocity field) directly from the data and then to use these optimized parameters in the reconstruction algorithm. In addition, it should be possible to use recent results [27] on the updating of smoothed estimates to develop efficient algorithms for tracking the velocity field over time as well as space (indeed the algorithm in [27] would suggest that tracking over time could be accomplished with a relatively modest increase in complexity as compared to independent velocity reconstruction at each point in time).

Finally, since the stochastic estimation approach that we have developed here can be viewed as a model-based regularization technique, it should be applicable to a number of other problems in computer vision to

which regularization methods have been applied in the past, such as robot navigation [20], shape from shading [21], [22], stereo vision [23], etc... Again a potential benefit of this approach is the extreme flexibility which exists in generating both the required multidimensional stochastic models and the corresponding reconstruction algorithms.

APPENDIX A

In this appendix we present an algorithm for the optimal estimation of the optical flow $\mathbf{V}(s)$ along a closed contour given the model (2.12)-(2.16) together with the constraint (2.31). The method used is the one developed in [18].

As in the two filter algorithm (2.19), (2.10), (2.26)-(2.28), the algorithm in this case consists of separate forward and backward sweeps followed by computations to combine the results of these sweeps. Specifically in the backward sweep we again solve (2.9), (2.10) backward from final values $\Theta(L) = 0$, $\mathbf{q}(L) = 0$ at $s = L$, and at the same time we also solve the backward 2x2 matrix equation

$$d\Phi_b/ds = \Theta(s)\Phi_b(s) \tag{A.1a}$$

$$\Phi_b(L) = I \quad . \tag{A.1b}$$

In the forward direction we solve (2.26), (2.27) from initial values $\Psi(0) = 0$, $\mathbf{r}(0) = 0$ at $s = 0$, and at the same time we solve the forward 2x2 matrix equation

$$d\Phi_f/ds = -\Psi(s)\Phi_f(s) \tag{A.2a}$$

$$\Phi_f(0) = I \tag{A.2b}$$

Following these computations, $\hat{\mathbf{V}}(s)$ is constructed as

$$\hat{\mathbf{V}}(s) = P(s)\,[\,\mathbf{q}(s) + \mathbf{r}(s) + \boldsymbol{\nu}(s)\,] \tag{A.3}$$

(compare to (2.28)), where $P(s)$ is given in (2.29) and the correction term $\boldsymbol{\nu}(s)$, which adjusts the estimates to account for the constraint that $\hat{\mathbf{V}}(0) = \hat{\mathbf{V}}(L)$, is given by

$$\boldsymbol{\nu}(s) = [\,\Phi_f(s) - \Phi_b(s)\,]\,R\,[\,\Psi(L)^{-1}\mathbf{r}(L) - \Theta(0)^{-1}\mathbf{q}(0)\,] \tag{A.4}$$

where

$$R = \{\,\Theta(0)^{-1}\,[\,I - \Phi_b(0)\,] + \Psi(L)^{-1}\,[\,I - \Phi_f(L)\,]\,\}^{-1} \quad . \tag{A.5}$$

APPENDIX B

In this appendix we briefly review and then apply the methodology developed in [18]. Specifically consider a stochastic process $\mathbf{x}$ on a region D and implicitly defined by the equations

$$L\mathbf{x} = B\mathbf{u} \tag{B.1}$$

$$V\mathbf{x}_b = \mathbf{v} \ . \tag{B.2}$$

Here L is a differential or difference operator in one or more dimensions, and $\mathbf{u}$ is a white noise process with intensity matrix Q. Also (B.2) represents the boundary conditions associated with (B.1). Here $\mathbf{x}_b$ is defined on the boundary C of D and consists of quantities such as the value of $\mathbf{x}$ and its normal derivative on C. The variable $\mathbf{v}$ is a white noise process on C with intensity matrix Π_v. Suppose that we observe

$$\mathbf{y} = C\mathbf{x} + \mathbf{r} \tag{B.3}$$

inside D, where $\mathbf{r}$ is white noise with intensity matrix R, and

$$\mathbf{y}_b = W\mathbf{x}_b + \mathbf{r}_b \tag{B.4}$$

on C, where $\mathbf{r}_b$ is white noise with intensity Π_b. The objective is to estimate $\mathbf{x}$ based on knowledge of $\mathbf{y}$ and $\mathbf{y}_b$.

The approach in [18] is based on the concept of complementary models. Specifically, as an intermediate step we construct a second process λ and associated observations $\mathbf{z}$ and $\mathbf{z}_b$ so that $\mathbf{z}$ and $\mathbf{z}_b$ are independent and complete the observations $\mathbf{y}$ and $\mathbf{y}_b$ in the sense that knowledge of $\{\mathbf{z}, \mathbf{z}_b, \mathbf{y}, \mathbf{y}_b\}$ is equivalent to the complete specification of all the processes $\{\mathbf{u}, \mathbf{v}, \mathbf{r}, \mathbf{r}_b\}$ that enter in determining $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{y}_b$. With this complementary model in hand, one can eliminate $\{\mathbf{u}, \mathbf{v}, \mathbf{r}, \mathbf{r}_b\}$ from the model equations, expressing $\mathbf{x}$ and λ completely in terms of $\{\mathbf{z}, \mathbf{z}_b, \mathbf{y}, \mathbf{y}_b\}$. The equations specifying the optimal estimator are then obtained simply by setting $\mathbf{z}$ and $\mathbf{z}_b$ equal to zero[1].

The key to the construction of the complementary model and thus to the construction of the estimator is a generalized Green's identity

[1]The validity of this step can be seen as follows. The optimal estimate is obtained by conditioning on $\mathbf{y}$ and $\mathbf{y}_b$. However, $\mathbf{z}$ and $\mathbf{z}_b$ are independent of the $\mathbf{y}$'s.

$$\langle L\mathbf{x}, \boldsymbol{\lambda} \rangle_D = \langle \mathbf{x}, L^t\boldsymbol{\lambda} \rangle_D + \langle \mathbf{x}_b, E\boldsymbol{\lambda}_b \rangle_C \tag{B.5}$$

where $\langle .,. \rangle$ denotes inner products of vector functions on D or C. The two key quantities in (B.5) are the operators L^t and E. Having these, we can write down the so-called Hamiltonian form of the optimal estimator

$$\begin{bmatrix} L & -BQB^T \\ C^TR^{-1}C & L^t \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} 0 \\ C^TR^{-1}\mathbf{y} \end{bmatrix} \tag{B.6}$$

with boundary conditions

$$\left(W^T\Pi_b^{-1}W + V^T\Pi_v^{-1}V \mid E \right) \begin{bmatrix} \hat{\mathbf{x}}_b \\ \hat{\boldsymbol{\lambda}}_b \end{bmatrix} = W^T\Pi_b^{-1}\mathbf{y}_b \ . \tag{B.7}$$

Let us apply these results to the model (3.3)-(3.7). In this case we can make use of the classical Green's identity. Specifically, if f is a scalar function on D and if $\mathbf{g}$ is any 2-vector of such functions, then

$$\int_D [\nabla f(x,y)]^T \mathbf{g}(x,y)\, dx\, dy$$

$$= \int_D f(x,y)\, [-\nabla\cdot \mathbf{g}(x,y)]\, dx\, dy + \int_C f(x(s),y(s))\, [\mathbf{n}(s)^T\mathbf{g}(x(s),y(s))]\, ds \ . \tag{B.8}$$

Here $\nabla\cdot$ is the divergence operator, i.e. if $\mathbf{g}(x,y)^T = [g_1(x,y), g_2(x,y)]$, then

$$\nabla\cdot \mathbf{g}(x,y) = \partial/\partial x\ g_1(x,y) + \partial/\partial y\ g_2(x,y) \ . \tag{B.9}$$

Also in the last integral on the right-hand side of (B.8), s parametrizes distance along the contour C, (x(s),y(s)) is the corresponding point on the curve, and $\mathbf{n}(s)$ is the normal to the curve.

Consider then the operator L in (3.3). It is a simple exercise to check that the Green's identity in this case is

$$\int_D [L\mathbf{f}(x,y)]^T \mathbf{g}(x,y)\, dx\, dy$$

$$= \int_D \mathbf{f}(x,y)^T\, [L^t\mathbf{g}(x,y)]\, dx\, dy + \int_C \mathbf{f}(x(s),y(s))^T\, N(s)\mathbf{g}(x(s),y(s))\, ds \tag{B.10}$$

where $\mathbf{f}$ is a 2-vector, $\mathbf{g}$ is a 4-vector, and

$$L^{\dagger} = -(I \otimes \nabla\cdot) \tag{B.11}$$

$$N(s) = I \otimes \mathbf{n}(s)^T \tag{B.12}$$

i.e., if $\mathbf{g}(x,y) = [\mathbf{g_1}^T(x,y), \mathbf{g_2}^T(x,y)]^T$ with $\mathbf{g_1}$, $\mathbf{g_2}$ each 2-vectors, then

$$L^{\dagger}\mathbf{g} = [\, \nabla\cdot \mathbf{g_1} \,,\, \nabla\cdot \mathbf{g_2} \,]^T \tag{B.13}$$

and

$$N(s)\mathbf{g} = [\, \mathbf{n}(s)^T\mathbf{g_1} \,,\, \mathbf{n}(s)^T\mathbf{g_2} \,] \quad . \tag{B.14}$$

Applying (B.6) to this problem (with appropriate changes in notation) we obtain the following system of equations

$$L\hat{\mathbf{V}} = \hat{\mathbf{W}} \tag{B.15}$$

$$a\, H^T H\, \hat{\mathbf{V}} + L^{\dagger}\hat{\mathbf{W}} = a\, H^T z \tag{B.16}$$

(where $\mathbf{W}$ corresponds to λ in (B.6)). Using (B.15) to eliminate $\hat{\mathbf{W}}$ we obtain the equation

$$L^{\dagger}L\hat{\mathbf{V}} = a\, H^T (z - H\hat{\mathbf{V}}) \quad . \tag{B.17}$$

Note that

$$\begin{aligned} L^{\dagger}L &= -(I \otimes \nabla\cdot)(I \otimes \nabla) \\ &= -(I \otimes \nabla\cdot\nabla) = -(I \otimes \Delta) \end{aligned} \tag{B.18}$$

where Δ is the Laplacian. Equations (3.9), (3.10) then follow from (B.17), (B.18), and (3.8).

Similarly, the boundary condition (B.7) in this case becomes

$$P_C^{-1}[\, \hat{\mathbf{V}} - \hat{\mathbf{V}}_C \,] + N(s)\hat{\mathbf{W}} = 0 \quad \text{on } C \quad . \tag{B.19}$$

Rearranging and again using (B.15) to eliminate $\hat{\mathbf{W}}$, this condition becomes

$$\hat{\mathbf{V}} = \hat{\mathbf{V}}_C - P_C\, N(s)\, L\hat{\mathbf{V}} \quad \text{on } C \quad . \tag{B.20}$$

Equation (3.11) then follows from the definition (B.12) and (B.20).

REFERENCES

[1] **Poggio**, T. and **Torre**, V., "Ill-Posed Problems and Regularization Analysis in Early Vision", A.I. Lab. Memo No. 773, April 1984, M.I.T., Cambridge MA.

[2] **Horn**, B. K. P. and **Schunck**, B. G., "Determining Optical Flow", Artificial Intelligence 17, 1981, pp. 185-203.

[3] **Cornelius**, N. and **Kanade**, T., "Adapting Optical-Flow to Measure Object Motion in Reflectance and X-Ray Image Sequences", ACM SIGGRAPH/SIGART Interdisciplinary Workshop on Motion: Representation and Perception, April 1983, Toronto, Canada, pp. 50-58.

[4] **Nagel**, H.-H., "Displacement Vectors Derived from Second-Order Intensity Variations in Image Sequences", Computer Vision, Graphics, and Image Processing 21, 1983, pp. 85-117.

[5] **Horn**, B. K. P., Robot Vision, M.I.T. Press, McGraw-Hill, 1986.

[6] **Glazer**, F., "Multilevel Relaxation in Low Level Computer Vision", Multiresolution Image Processing and Analysis, A. Rosenfeld Ed., Springer-Verlag, 1984, pp. 312-330.

[7] **Terzopoulos**, D., "Image Analysis Using Multigrid Relaxation Methods", IEEE Trans. on Pattern Anal. and Mach. Intell., Vol. PAMI-8 No.2, 1986, pp. 129-139.

[8] **Hildreth**, E. C., "Computations Underlying the Measurement of Visual Motion", Artificial Intelligence 23, 1984, pp. 309-354.

[9] **Bouthemy**, P., "Determining Displacements Fields Along Contours From Image Sequences", Proc. Vision Interface '86, Vancouver, pp. 350-355.

[10] **Ljung**, L. and **Kailath**, T., "A Unified Approach to Smoothing Formulas", Automatica, Vol. 12 No.2, 1976, pp. 147-157.

[11] **Kailath**, T. and **Ljung**, L., "Two-filter Smoothing Formulas by Diagonalization of the Hamiltonian Equations", Int. Journal of Control, Vol. 36 No.4, 1982, pp. 663-673.

[12] **Bouthemy**, P., "A Method of Integrating Motion Information Along Contours Including Segmentation", Proc. of the 8th ICPR, October 1986, Paris.

[13] **Athans**, M. and **Falb**, P. L., Optimal Control. An Introduction to the Theory and Its Applications, McGraw-Hill, New York, 1966.

[14] **Bryson**, A.E. and **Frazier**, M., "Smoothing for Linear and Nonlinear Dynamic Systems", Aero. Sys. Div., Wright-Patterson Air Force Base, OH, TDR 63-119, 1964, pp. 353-364.

[15] **Mayne**, D. Q., "A Solution of the Smoothing Problem for Linear Dynamic Systems", Automatica, Vol. 4 No.2, 1966, pp.73-92.

[16] **Fraser**, D. C., "A New Technique for the Optimal Smoothing of Data", Sc.D. Dissertation, 1967, M.I.T., Cambridge MA.

[17] **Rauch**, H. E., **Tung**, F. and **Striebel**, C. T., "Maximum Likelihood Estimates of Linear Dynamic Systems", AIAA Journal, Vol. 3 No.8, 1965, pp. 1445-1450.

[18] **Adams**, M. B., **Willsky**, A. S. and **Levy**, B. C., "Linear Estimation of Boundary Value Stochastic Processes - Part I: The Role and Construction of Complementary Models", and "Part II: 1-D Smoothing Problems", IEEE Trans. on Auto. Control, Vol. AC-29 , Sept. 1984, pp. 803-811.

[19] **Kuo**, J. C.-C., **Levy**, B. C., **Musicus**, B. R., "A Local Relaxation Method For Solving Elliptic PDEs on Mesh-Connected Arrays", Report LIDS-P-1508, October 1985, M.I.T., Cambridge MA, to appear in SIAM J. Scientific. Statistical Computing, July 1987.

[20] **Negahdaripour**, S. and **Horn**, B. K. P., "Direct Passive Navigation", A.I. Lab. Memo No. 821, February 1985, M.I.T., Cambridge MA.

[21] **Bruckstein**, A. M., "On Shape From Shading Methods: Some Theoretical Considerations", Faculty of Electrical Engineering Pub. No. 524, Technion Israel Institute of Technology, Haifa, Israel, June 1985.

[22] **Ikeuchi**, K. and **Horn**, B. K. P., "Numerical Shape From Shading and Occluding Boundaries", Artificial Intelligence17, 1981, pp.141-184.

[23] **Marr**, D. and **Poggio**, T., "A Theory of Human Stereo Vision", Proc. of the Royal Soc. London, B 204, 1979, pp. 301-328.

[24] **Nagel**, H.-H. and **Enkelmann**, W., "An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Image Sequences", IEEE Trans. on Pattern Anal. and Mach. Intell., Vol. PAMI-8, Sept. 1986, pp. 565-593.

[25] **Brockett**, R. W., "Gramiann, Generalized Inverses, and Affine Approximation of Optical Flow", preprint.

[26] **Wong**, E. and **Hajek**, B., Stochastic Processes in Engineering Systems, Springer-Verlag, New York, 1985.

[27] **Willsky**, A. S., **Bello**, M. G., **Castanon**, D. A., **Levy**, B. C., and **Verghese**, G. C., "Combining and Updating of Local Estimates and Regional Maps Along Sets of One-Dimensional Tracks", IEEE Trans. on Auto. Control, Vol. AC-27, Aug. 1982, pp. 799-813.

CAPTION FOR TABLE 1

Table 1: Relative error in function of the parameter a for the case of linear intensity variation considered in Figs. 7 and 9, and when a translational or rotational motion is applied to a rectangular or polygonal domain.

	Rectangle				Polygon			
	Translation		Rotation		Translation		Rotation	
a	0	0.008	0	0.003	0	0.0015	0	0.0007
relative error	18 %	**9 %**	12 %	**10 %**	8.9 %	**6.9 %**	8.2 %	**8 %**

Table 1

FIGURE CAPTIONS

Figure 1: Estimated velocities obtained from noiseless observations of the perpendicular component of the optical flow, when a translational motion is applied to a piecewise horizontal and vertical polygon.

Figure 2: Estimated velocities obtained from noiseless observations, when a translational motion is applied to a general polygon.

Figure 3: Estimated velocities and errors obtained from noisy observations, with the noise intensity equal to 0.01 or 0.1, for a translational motion applied to the polygon of Fig. 2.

Figure 4: Estimated velocities and errors obtained from noiseless measurements, with the parameter a set equal to 0.95 or 0.1, for a rotational motion applied to the polygon of Fig. 1.

Figure 5: Estimated velocities and errors obtained from noiseless measurements, with the parameter a set equal to 0.7 or 0.1, for a rotational motion applied to the polygon of Fig. 2.

Figure 6: Estimated velocities and errors obtained by the algorithm with periodicity constraint of Appendix A for the measurements of Figure 5, and with a equal to 0.7 or 0.1 .

Figure 7: Estimated velocities and errors inside rectangular and polygonal domains, for linear intensity variations and a translational motion.

Figure 8: Estimated velocities and errors inside rectangular and polygonal domains, for sinusoidal intensity variations and a translational motion.

Figure 9: Estimated velocities and errors inside rectangular and polygonal domains, for linear intensity variations and a rotational motion.

Figure 10: Estimated velocities and errors inside rectangular and polygonal domains, for sinusoidal intensity variations and a rotational motion.

Figure 11: Estimated velocities and errors inside a rectangular domain, when the exact velocity field given by (4.17) corresponds to a non-rigid motion.

Figure 12: Estimated velocities and errors inside a rectangular domain for linear intensity variations and a rotational motion, given some estimates of the velocity field on the boundary with error variance $P_C = I$ or $P_C = 0.01$.

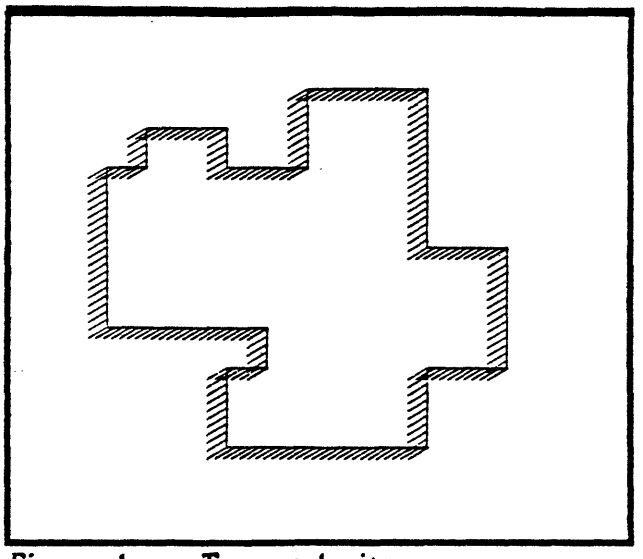

Figure 1.a: True velocity

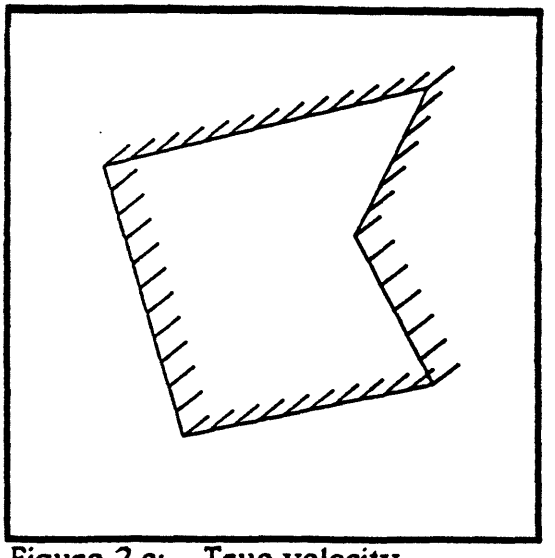

Figure 2.a: True velocity

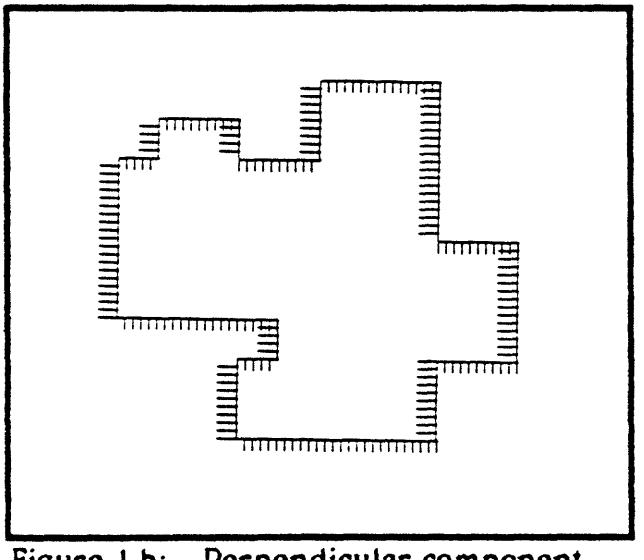

Figure 1.b: Perpendicular component

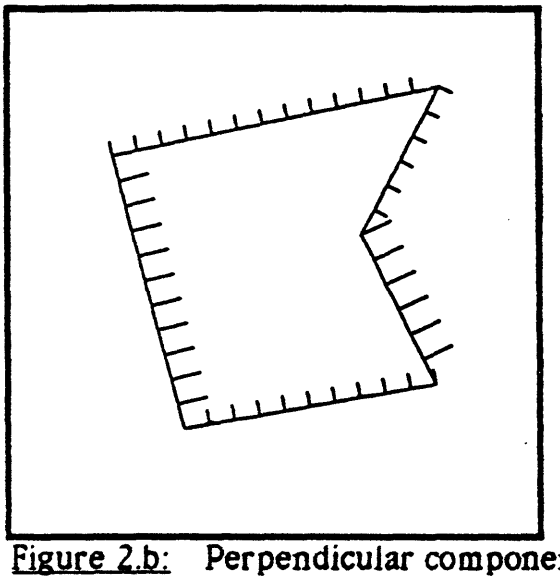

Figure 2.b: Perpendicular component

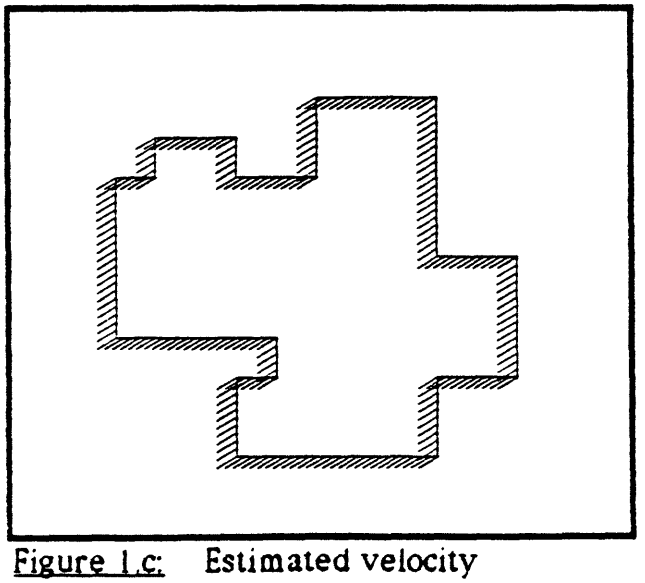

Figure 1.c: Estimated velocity

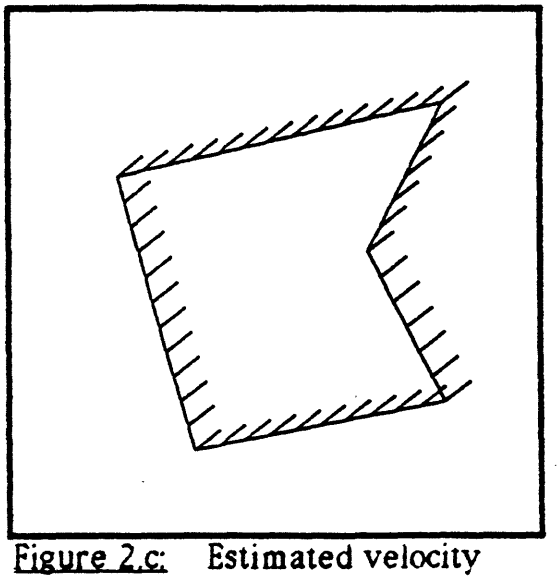

Figure 2.c: Estimated velocity

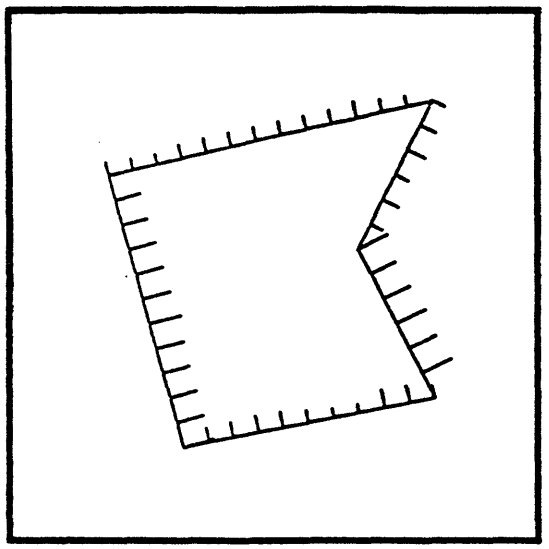

Figure 3.a: Observations, var= 0.01

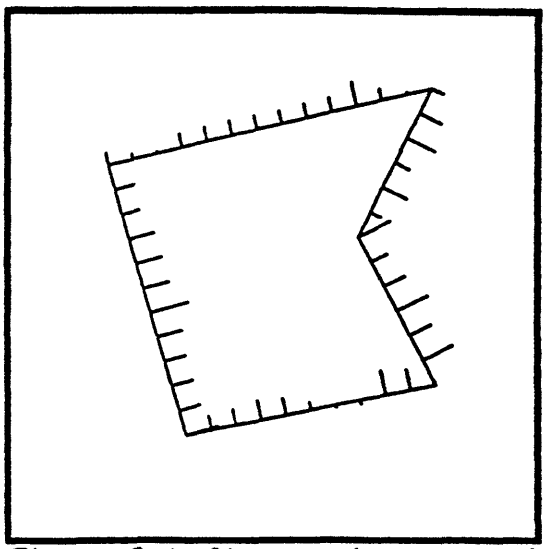

Figure 3.d: Observations, var= 0.1

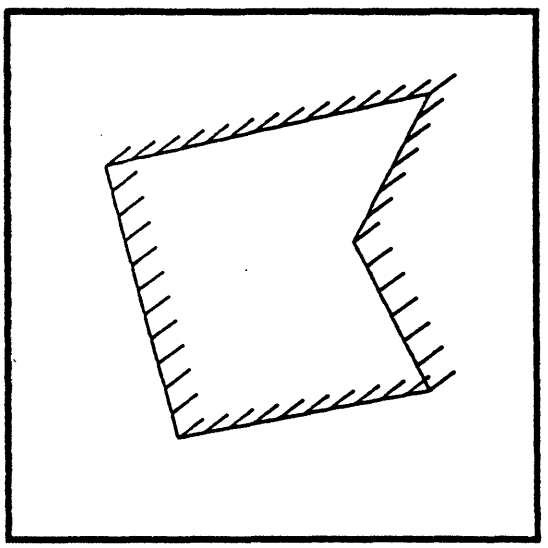

Figure 3.b: Estimated velocity

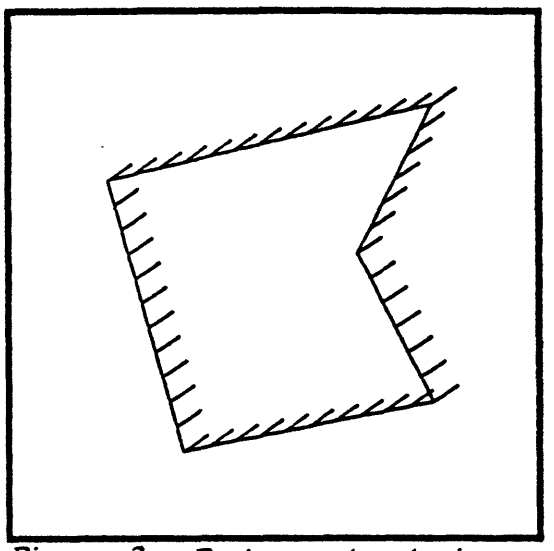

Figure 3.e: Estimated velocity

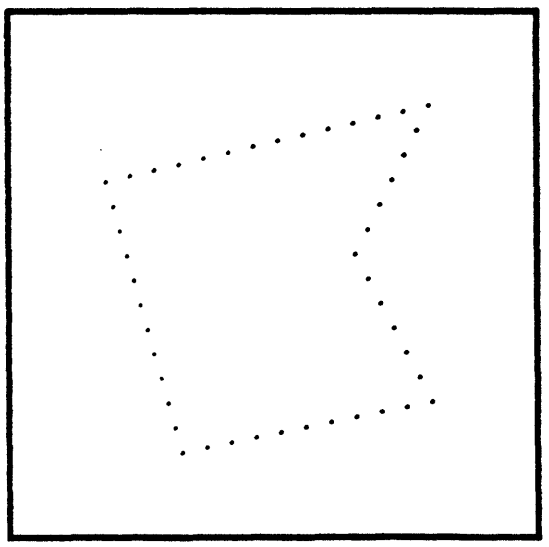

Figure 3.c: Estimation error

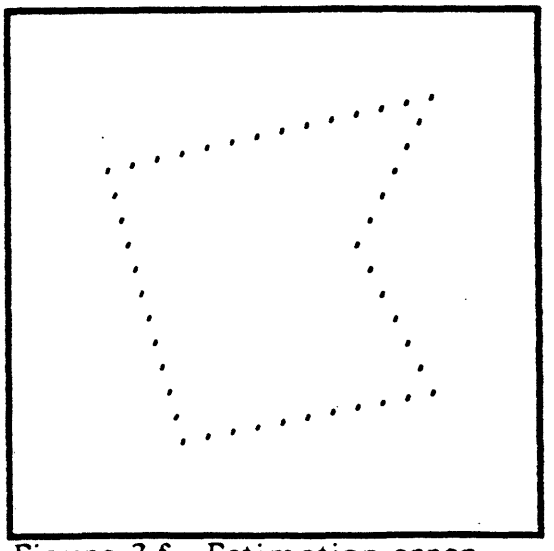

Figure 3.f: Estimation error

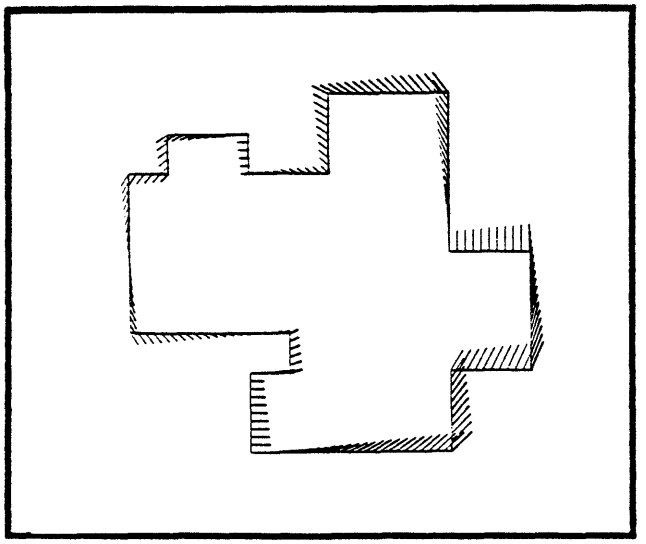

Figure 4.a: True velocity

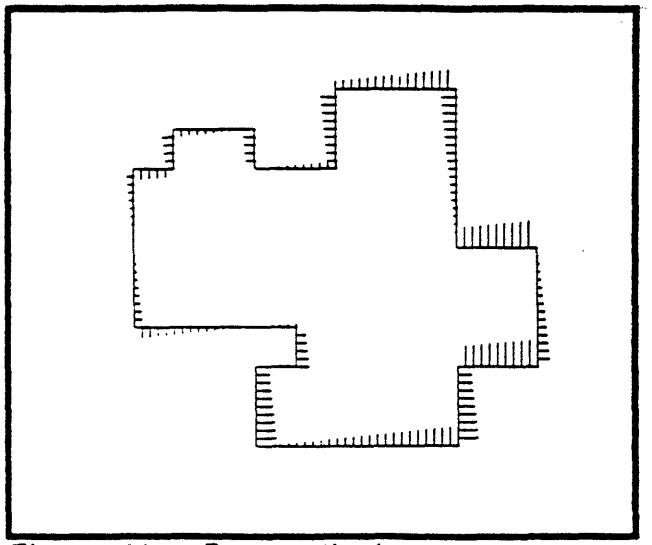

Figure 4.b: Perpendicular component

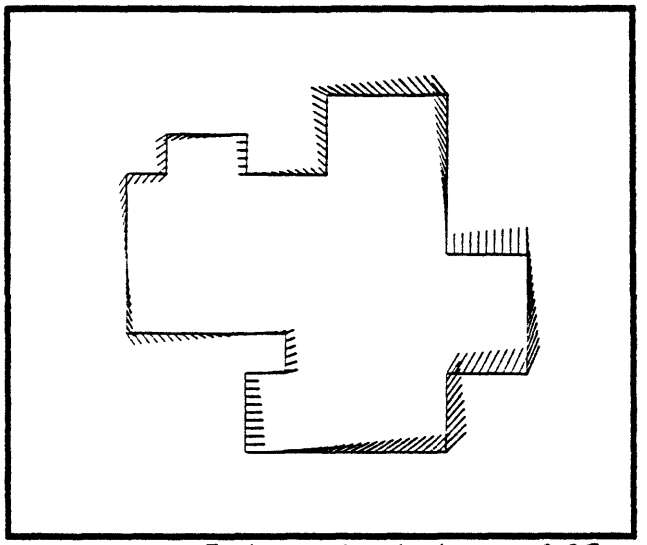

Figure 4.c: Estimated velocity, a= 0.95

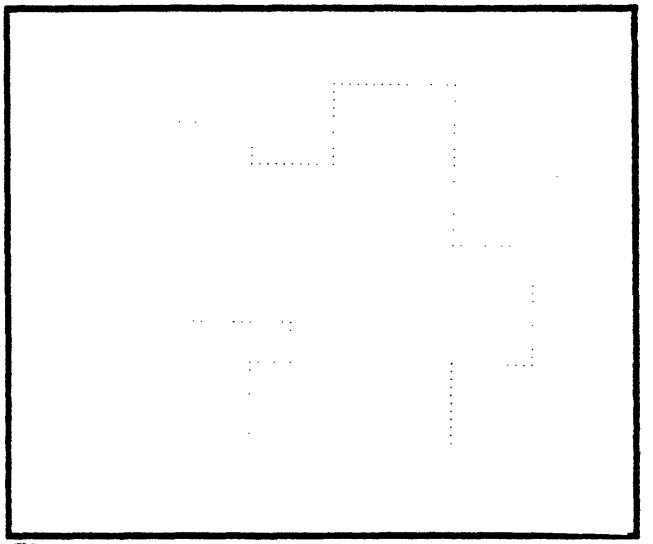

Figure 4.d: Estimation error, a= 0.95

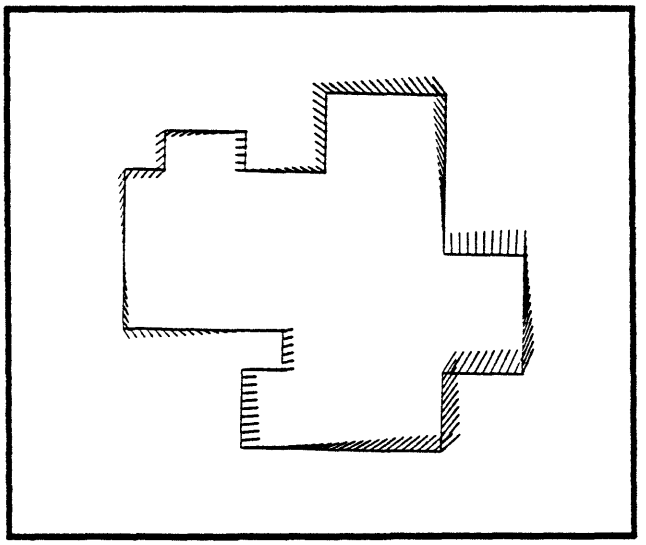

Figure 4.e: Estimated velocity, a= 0.1

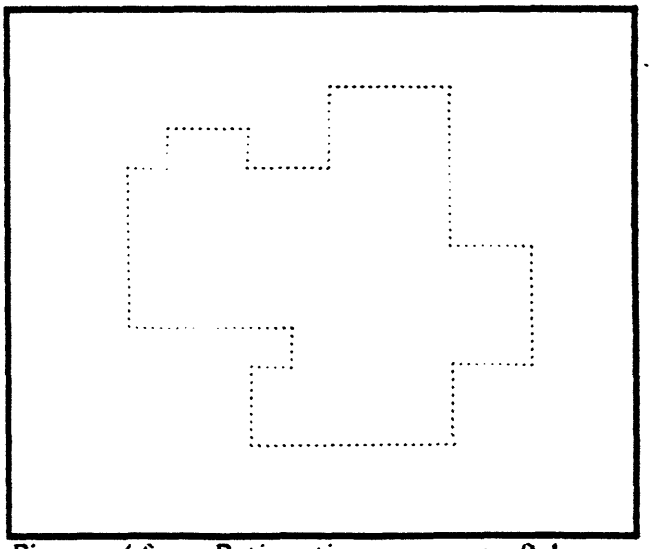

Figure 4.f: Estimation error, a= 0.1

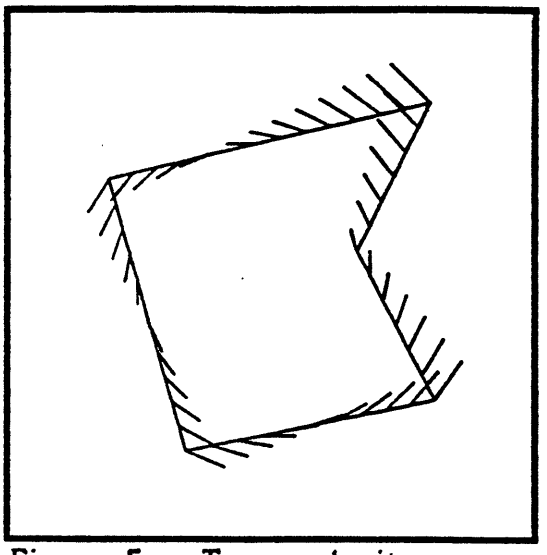

Figure 5.a: True velocity

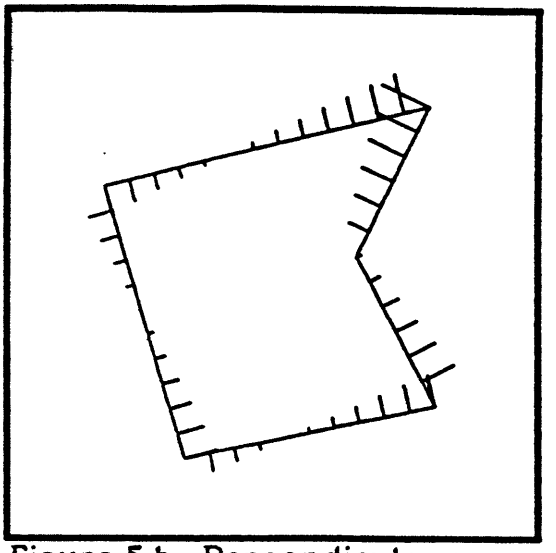

Figure 5.b: Perpendicular component

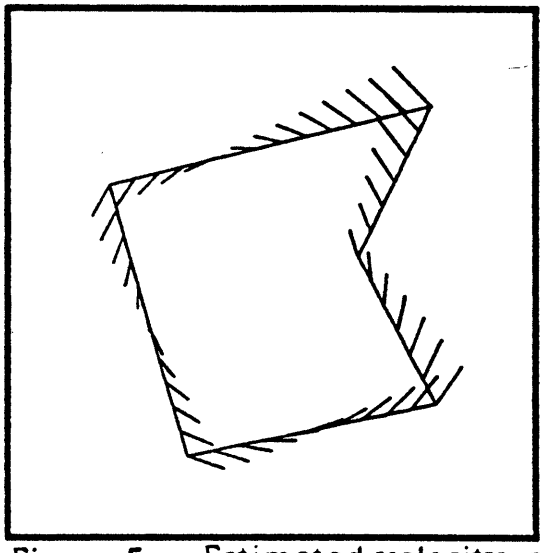

Figure 5.c: Estimated velocity, a=0.7

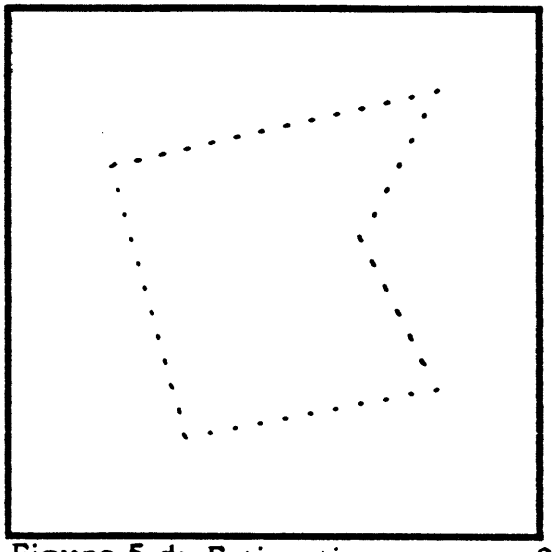

Figure 5.d: Estimation error, a=0.7

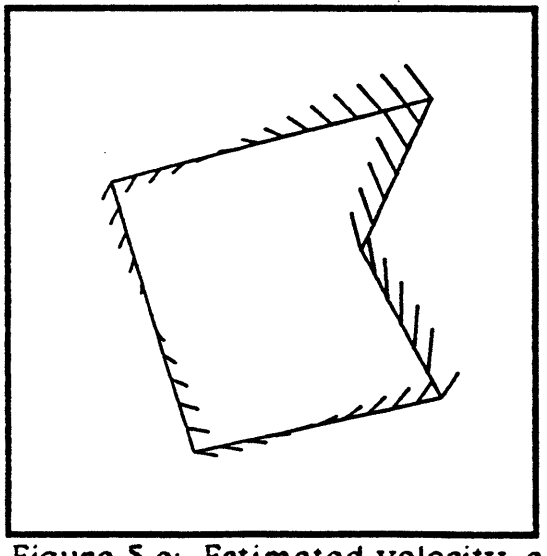

Figure 5.e: Estimated velocity, a=0.1

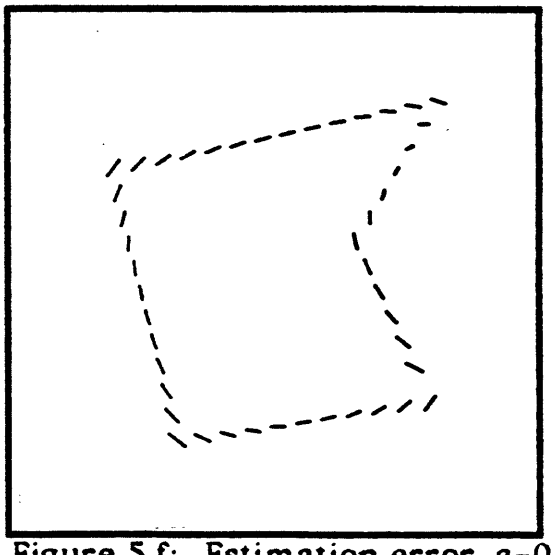

Figure 5.f: Estimation error, a=0.1

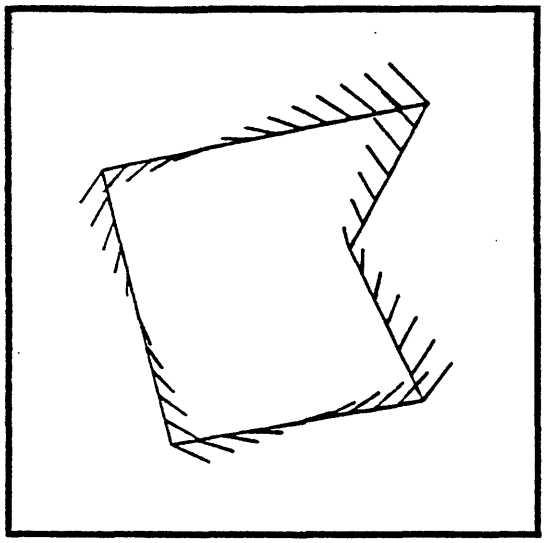

Figure 6.a: True velocity

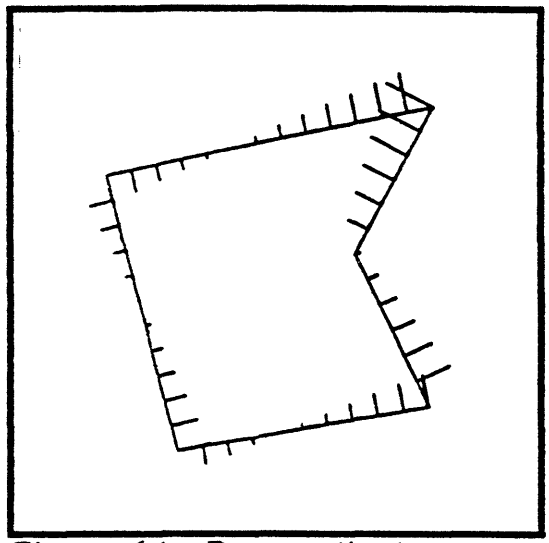

Figure 6.b: Perpendicular component

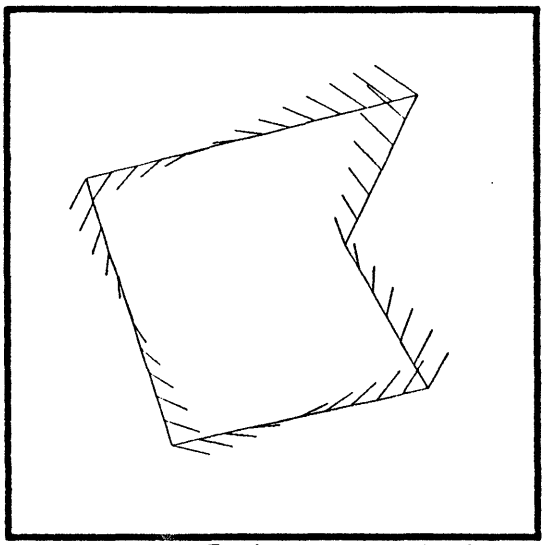

Figure 6.c: Estimated velocity, a=0.7

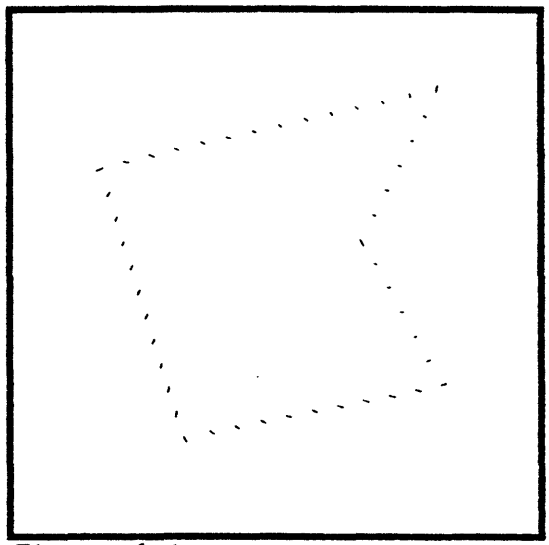

Figure 6.d: Estimation error, a=0.7

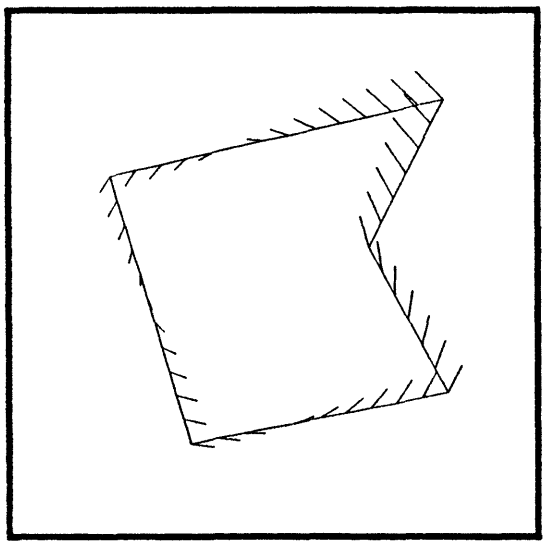

Figure 6.e: Estimated velocity, a=0.1

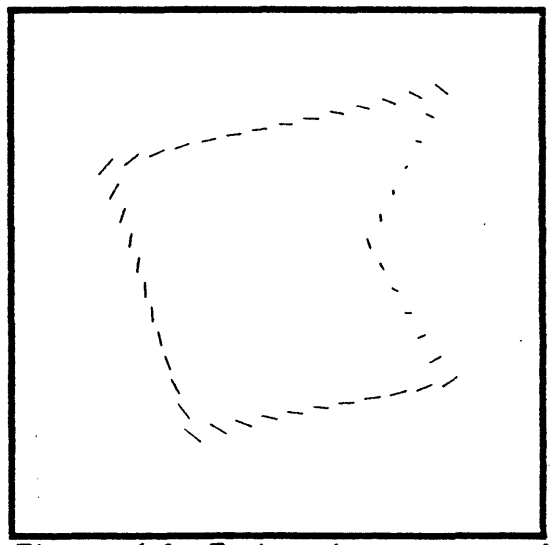

Figure 6.f: Estimation error, a=0.1

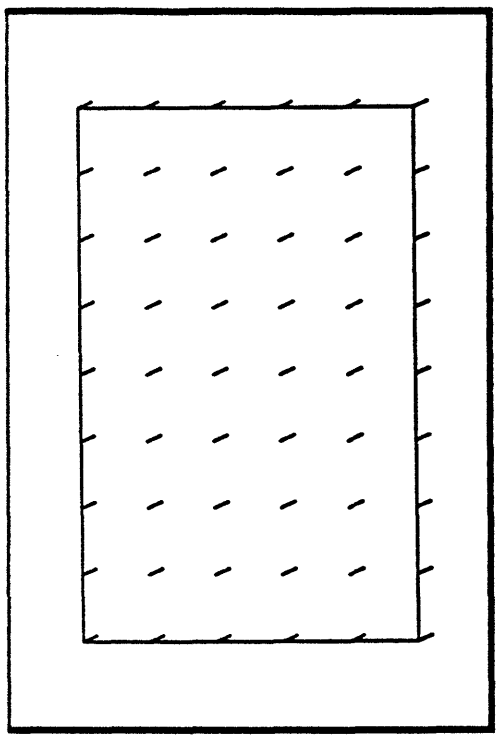

Fig. 7.a: True velocity

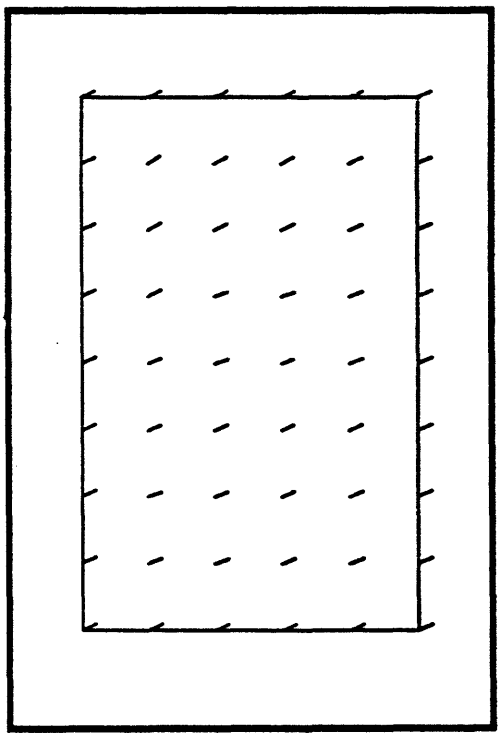

Fig. 7.b: Estimated velocity
25 iterations, a= 0.008

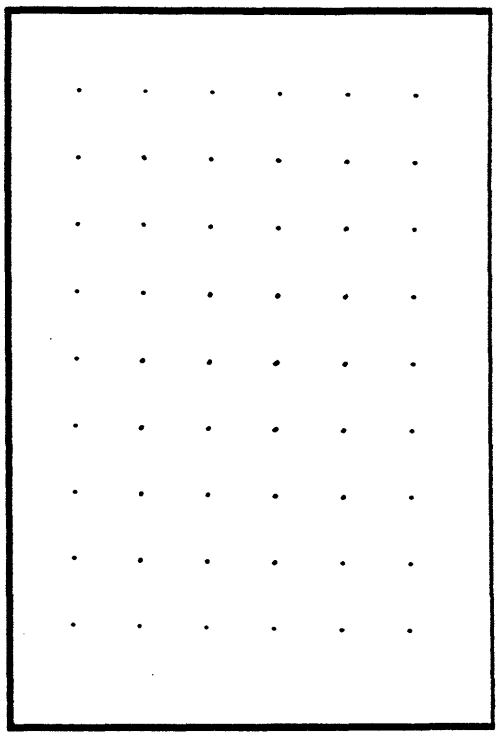

Fig. 7.c: Estimation error

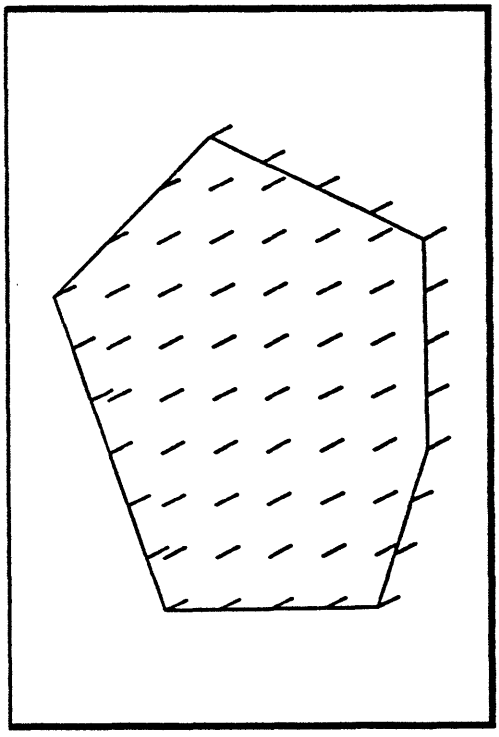

Fig. 7.d: True velocity

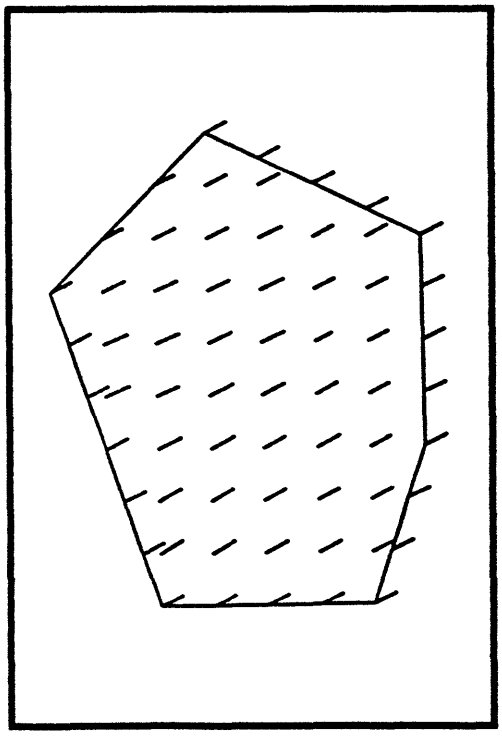

Fig. 7.e: Estimated velocity
15 iterations, a= 0.0015

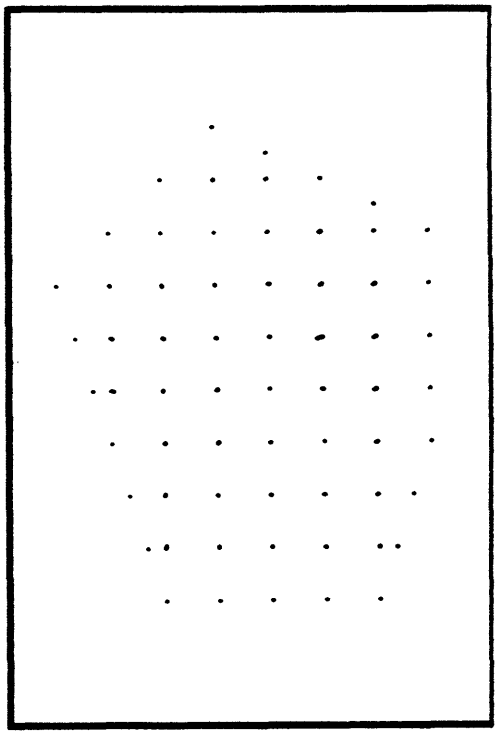

Fig. 7.f: Estimation error

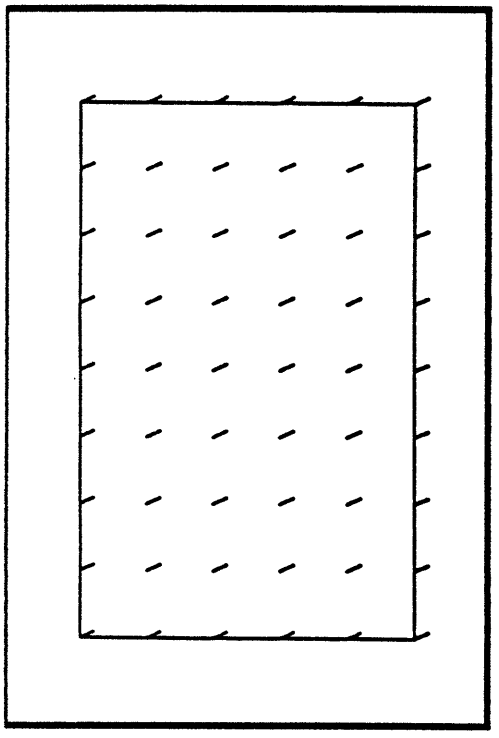

Fig. 8.a: True velocity

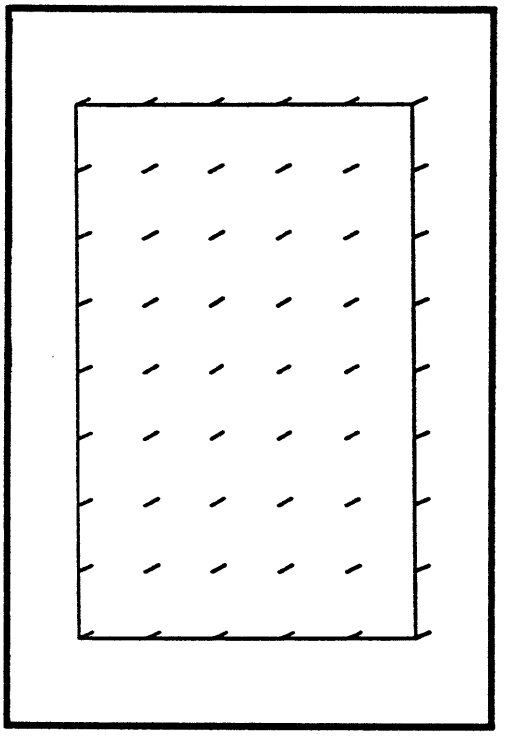

Fig. 8.b: Estimated velocity
25 iterations, a= 0.025

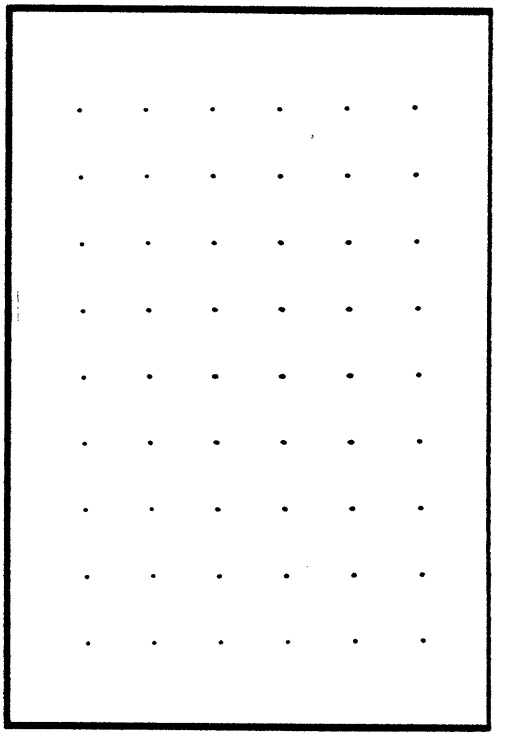

Fig. 8.c: Estimation error

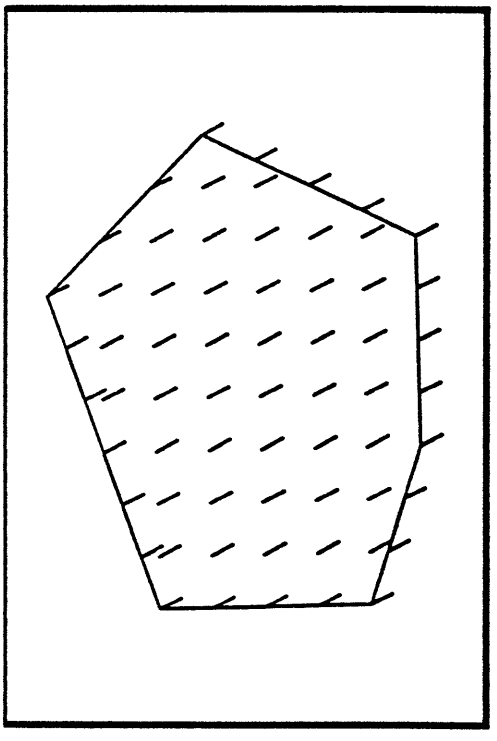

Fig. 8.d: True velocity

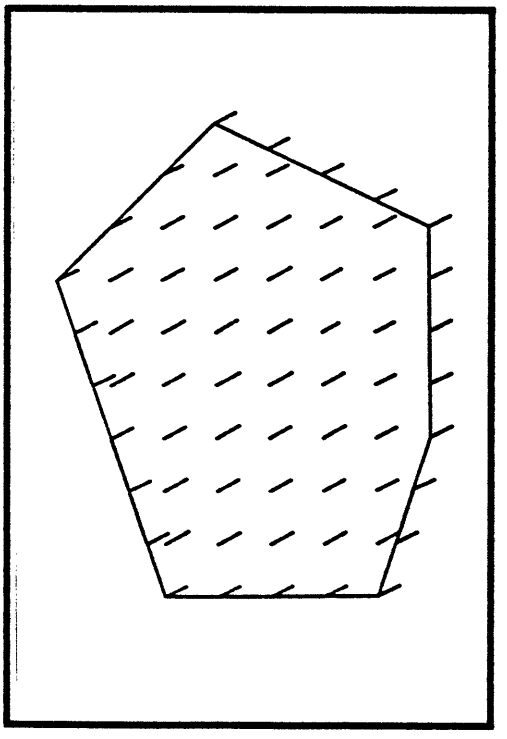

Fig. 8.e: Estimated velocity
15 iterations, a= 0.004

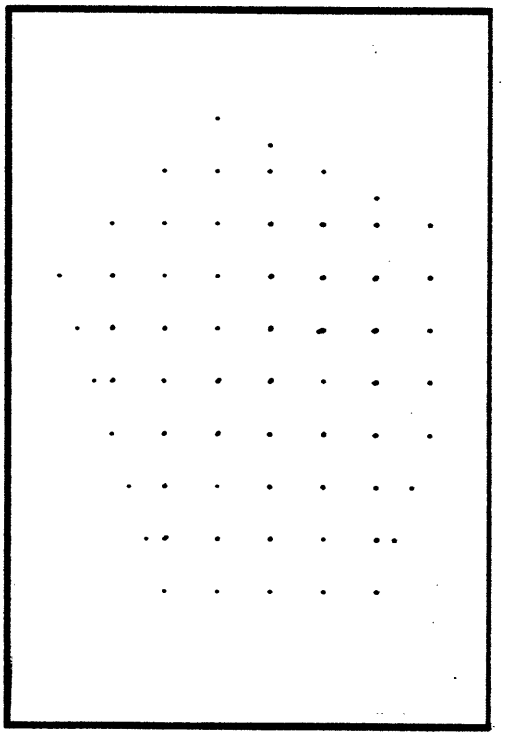

Fig. 8.f: Estimation error

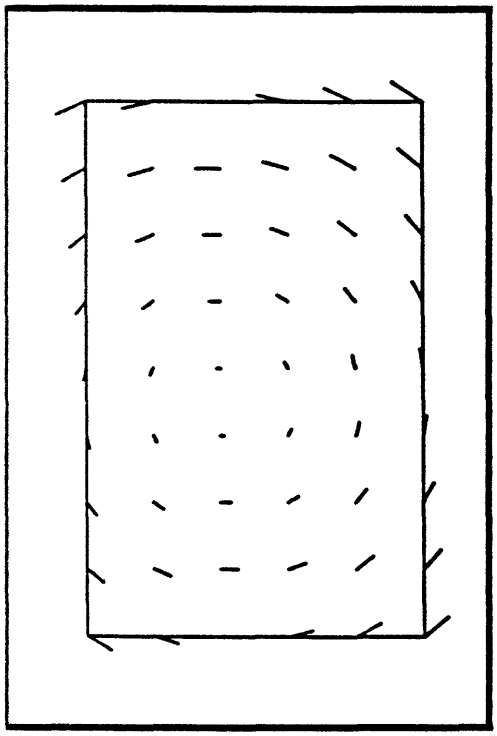

Fig. 9.a: True velocity

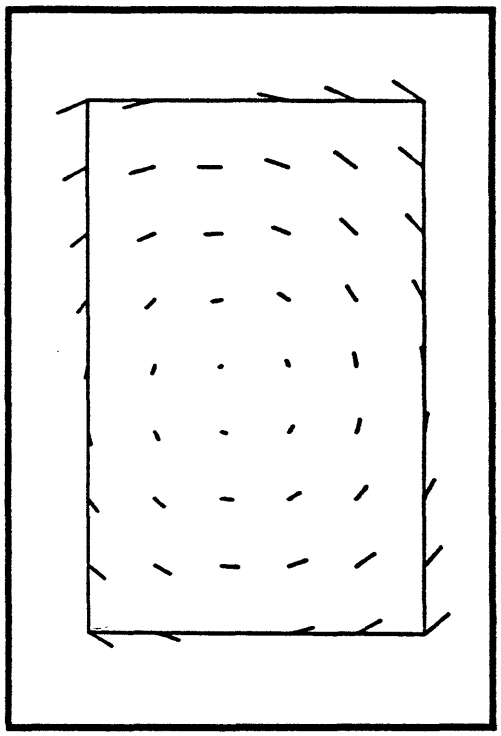

Fig. 9.b: Estimated velocity
25 iterations, a= 0.003

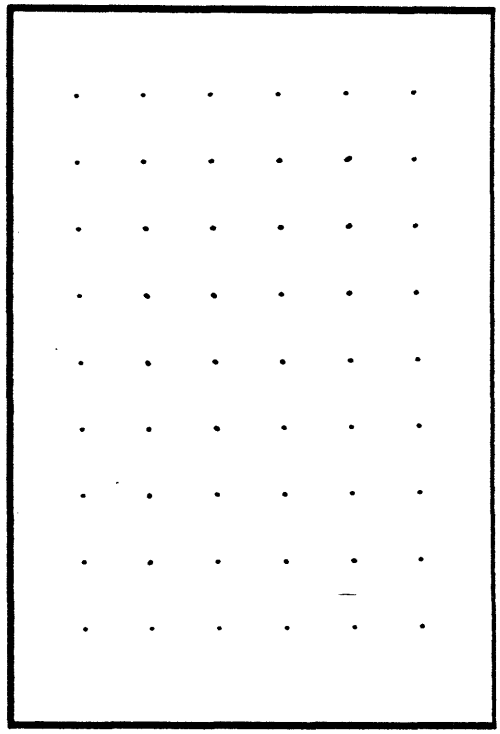

Fig. 9.c: Estimation error

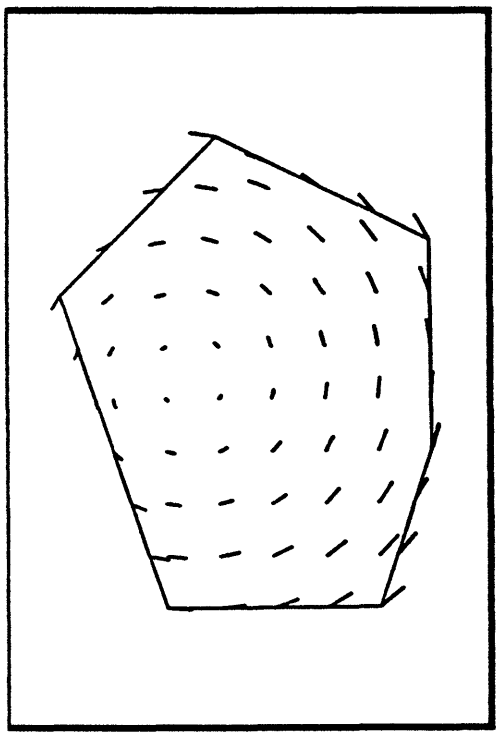

Fig. 9.d: True velocity

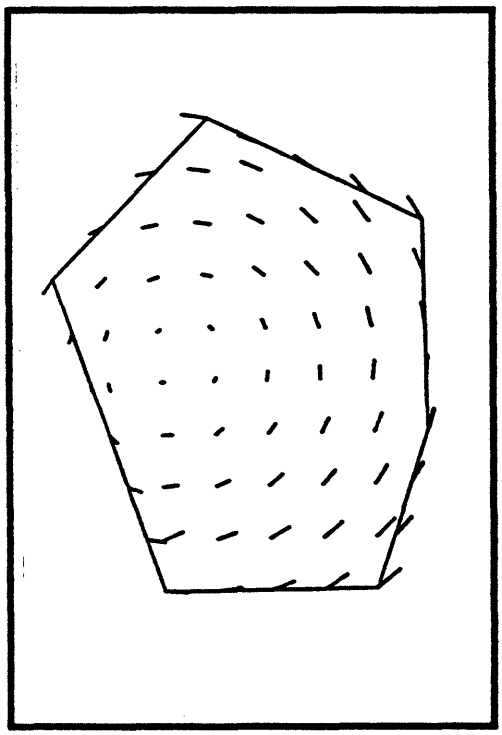

Fig. 9.e: Estimated velocity
20 iterations, a= 0.007

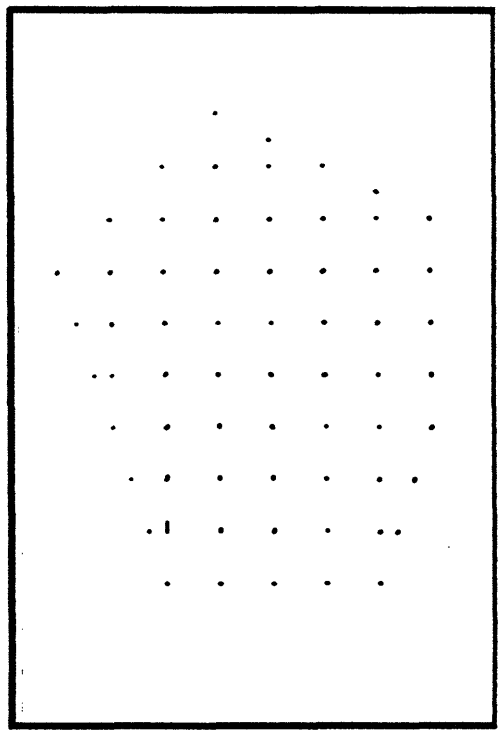

Fig. 9.f: Estimation error

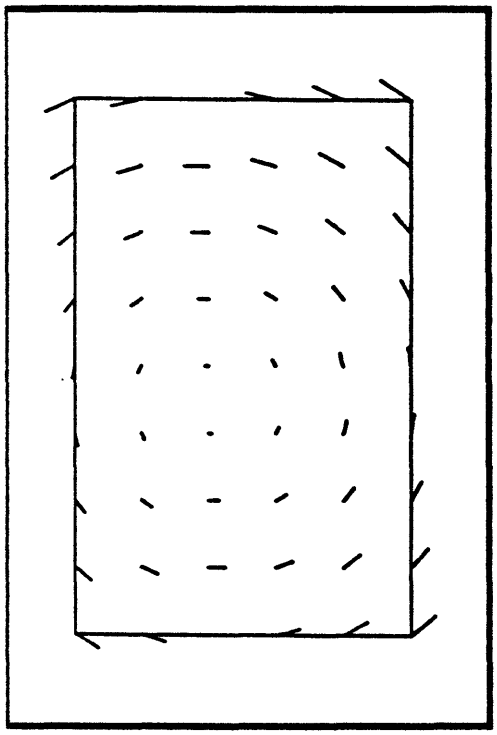

Fig. 10.a: True velocity

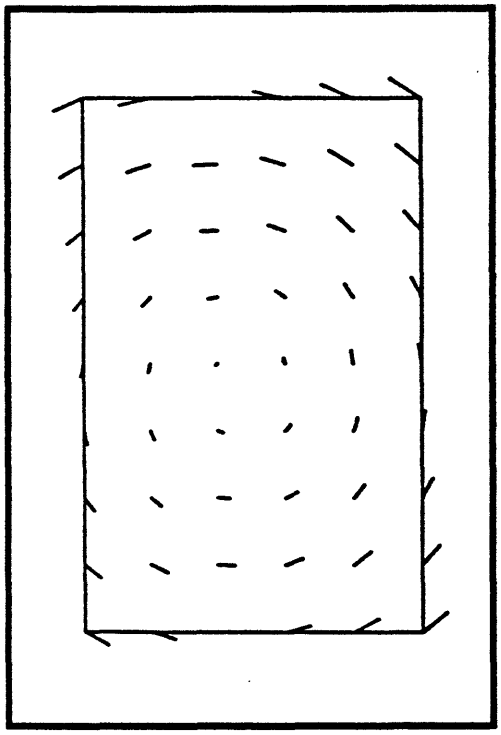

Fig. 10.b: Estimated velocity
25 iterations, a= 0.001

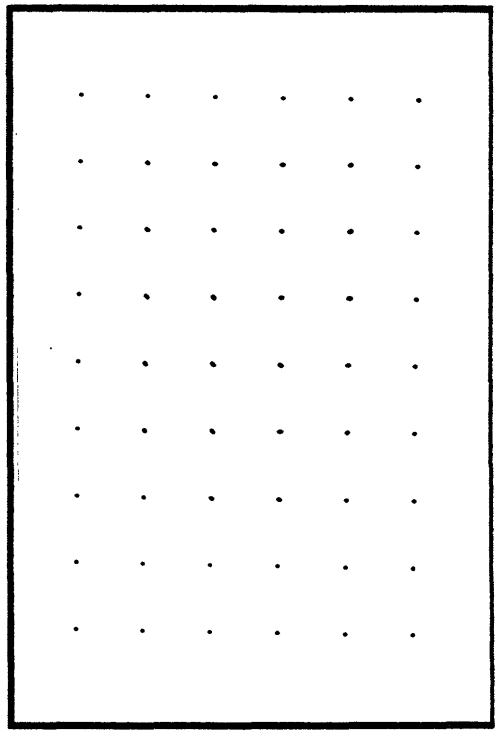

Fig. 10.c: Estimation error

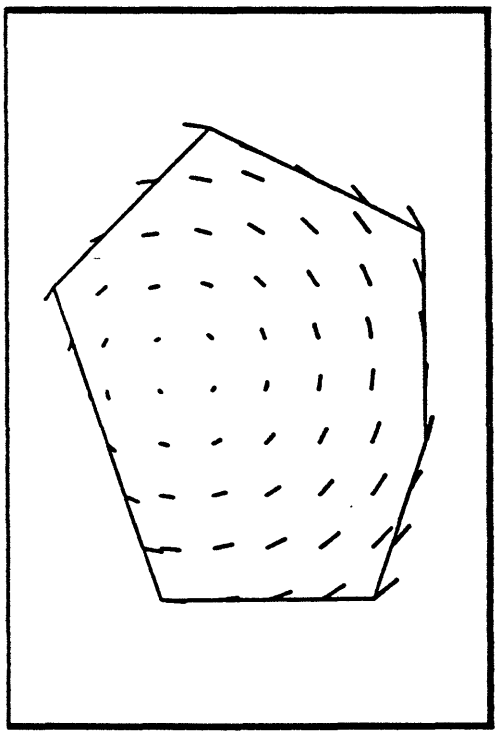

Fig. 10.d: True velocity

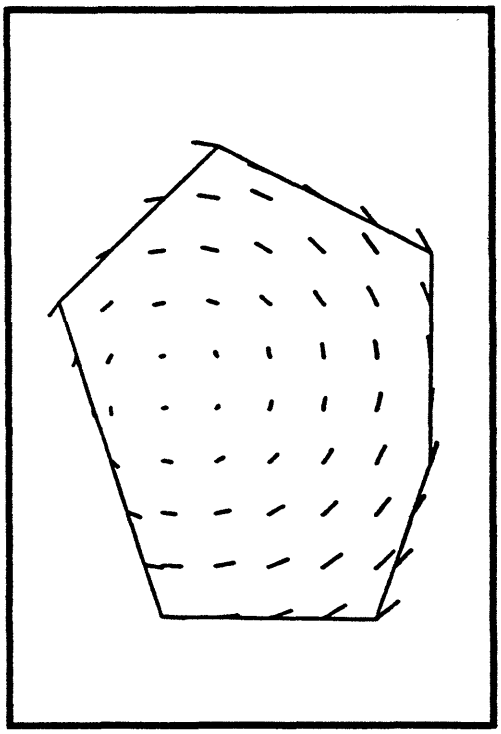

Fig. 10.e: Estimated velocity
20 iterations, a= 0.001

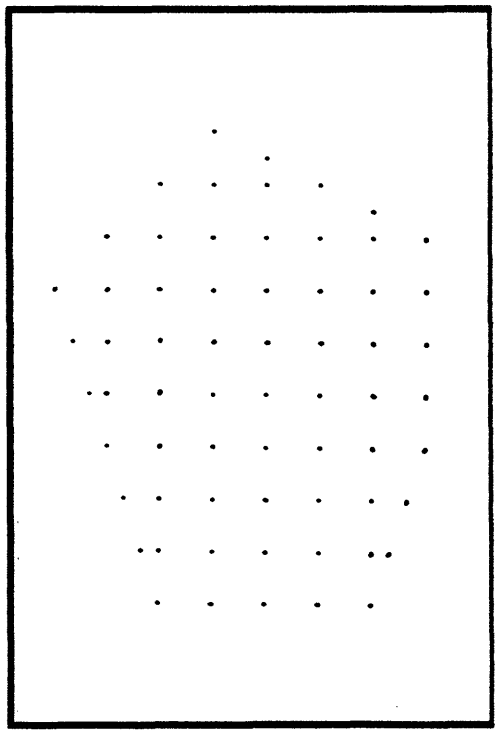

Fig. 10.f: Estimation error

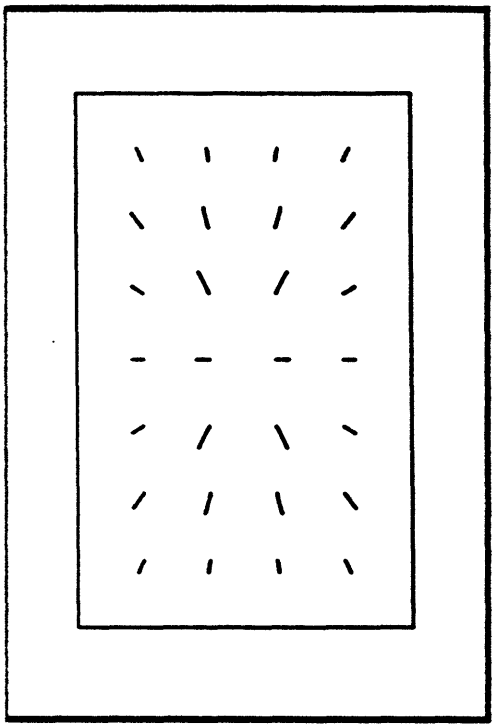

Fig. 11.a: True velocity

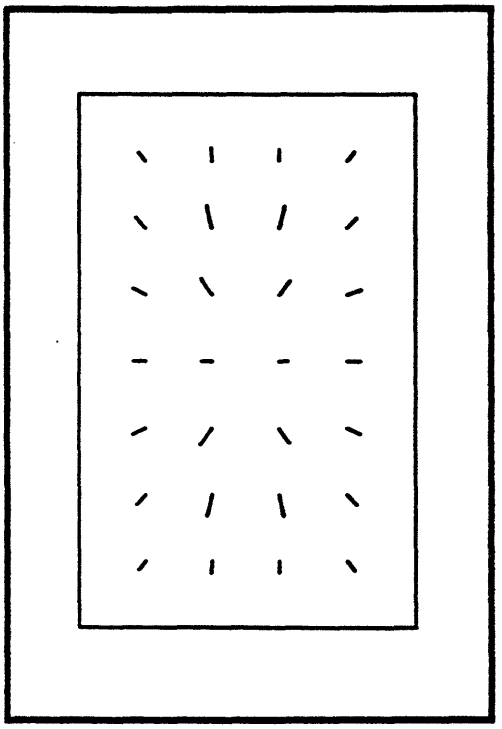

Fig. 11.b: Estimated velocity
25 iterations, a= 0.3

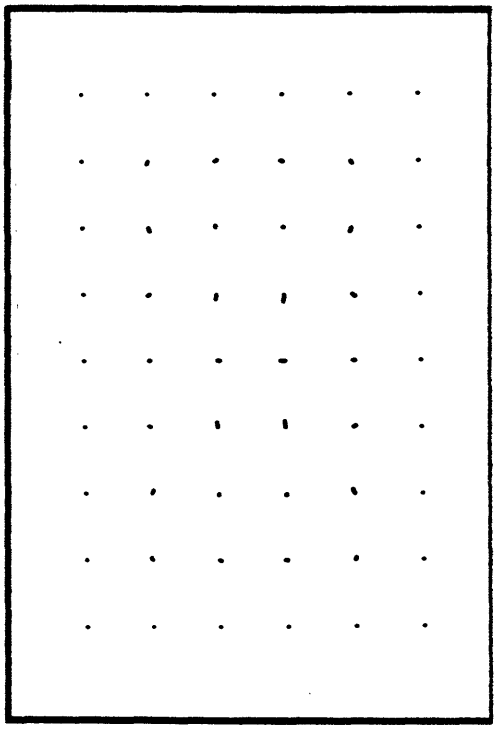

Fig. 11.c: Estimation error

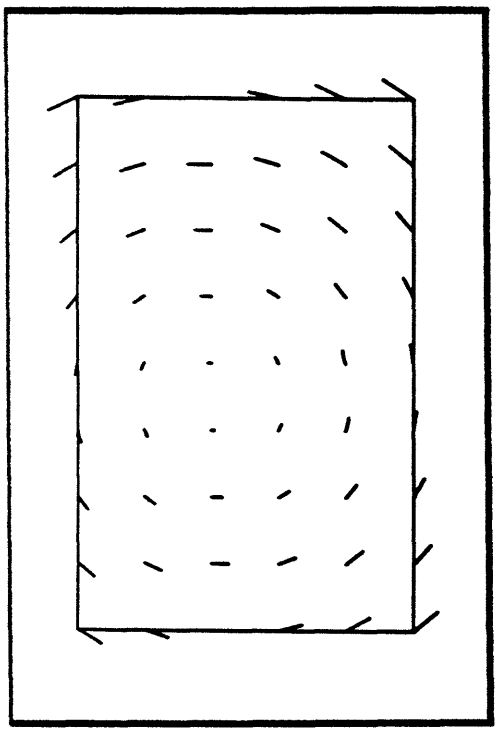

Fig. 12.a: True velocity

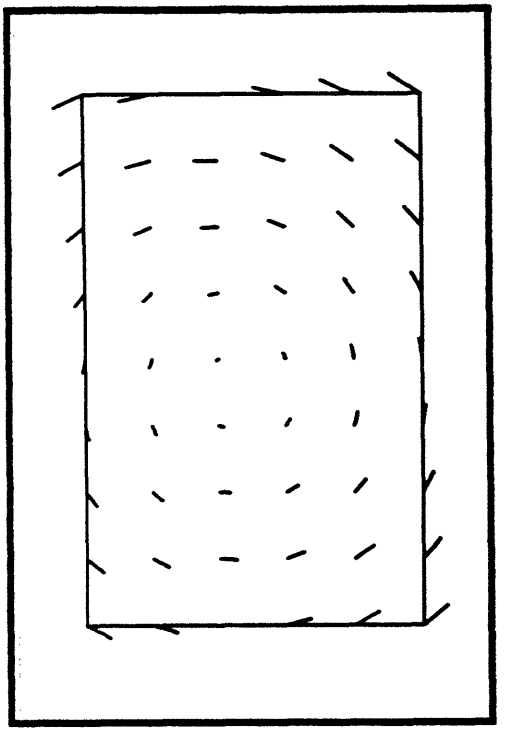

Fig. 12.b: Estimated velocity
P = 0.01 I, a= 0.003

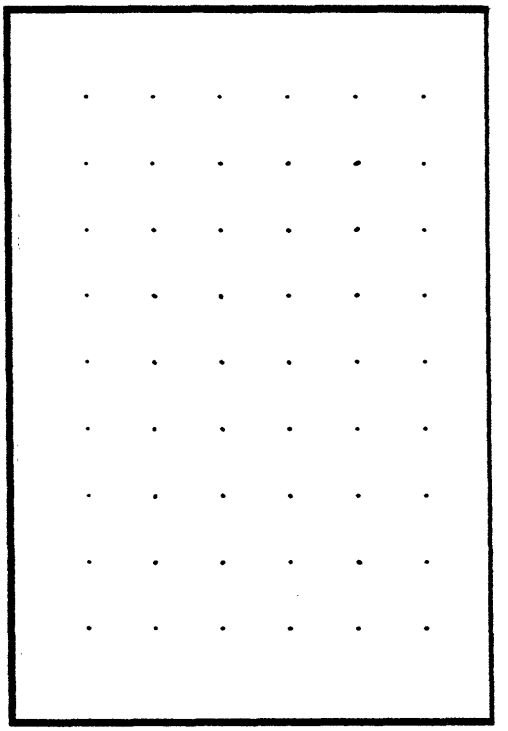

Fig. 12.c: Estimation error

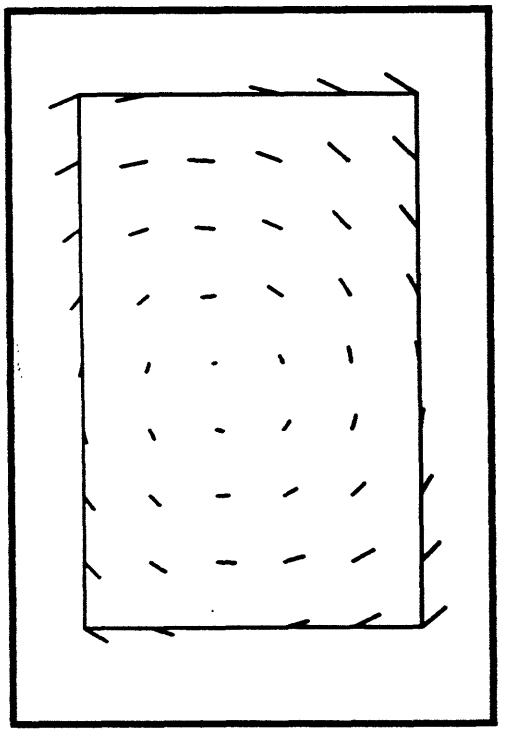

Fig. 12.d: Estimated velocity
P = I, a= 0.006

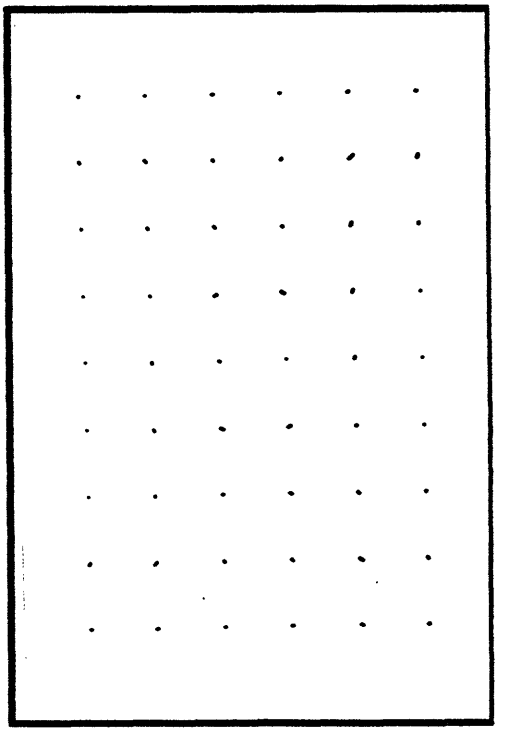

Fig. 12.e: Estimation error