DETERMINING DISPLACEMENT FIELDS ALONG CONTOURS FROM IMAGE SEQUENCES

Patrick BOUTHEMY

IRISA / INRIA
Campus de Beaulieu
35042 RENNES, France

ABSTRACT

We propose a framework for image flow analysis consisting of three major stages; i.e.:
- determine moving edges by some local process;
- integrate motion information along linked contours;
- propagate motion estimation through homogeneous regions obtained inside these contours.

This paper is concerned with the two first stages. A procedure, based on some local modeling and maximum likelihood scheme, has been designed to perform the first step. After some linking process, constraints provided by the measurements gained from the first stage can be combined to compute the velocity field along contours, by minimizing some simple functional. To this end, a gradient algorithm is used with a recursive estimation from one point to its successor in the chain.

RESUME

Nous proposons un schéma d'obtention du champ des vitesses dans une séquence d'images s'articulant en trois étapes, à savoir:
- déterminer localement les éléments de contour en mouvement;
- intégrer l'information de mouvement le long des lignes contours chaînées;
- propager l'estimation du mouvement à l'intérieur des zones homogènes délimitées par ces lignes contours.

Le papier traite des deux premières étapes. Une procédure, basée sur une modélisation locale et un critère de maximum de vraisemblance, a été conçue afin de réaliser le premier point. Après chaînage, les mesures issues du premier niveau peuvent être combinées afin de calculer le champ des vitesses complet le long des lignes contours via la minimisation d'une fonctionnelle simple. A cette fin, un algorithme de gradient est mis en oeuvre avec une récurrence de point en point le long de la chaîne contour.

KEYWORDS: image sequence, moving edge determination, motion estimation, local modeling, maximum likelihood test, stochastic gradient.

I INTRODUCTION

Image sequence analysis has received more and more attention since 70's. In particular substantial studies have been concerned with motion estimation across changing two-dimensional images. Two main motivations have subtened there research efforts. First, motion computation represents an attractive challenge in order to design some robust, tractable and general-purpose method. On the other hand, application areas never stop broadening, [1].

Meteorological applications (determining wind fields owing to cloud motion estimation, [2]), military domain (target tracking) were among pioneer ones. Then came interframe image coding for broadcast television or videoconferencing purpose [3,4]. For a few years, other potential applications have appeared: biomedical (e.g., angiocardiography [5]), robotics (mobile robot, [6]), traffic monitoring, graphics . . . These new domains are not only interested in two-dimensional motion as it is, but as intrinsic features conveying information about the depicted 3D-scene. Indeed motion in the imaging plane provides primary cues to relative depth, structure and 3D-movements of objects in space. (As one's uses a discrete representation of an image sequence, displacement vector fields and velocity vector fields are usually confused, although mathematically of different nature).

Discriminating discontinuities in the velocity field is a key problem in motion estimation schemes whatever they are. Indeed, feature-based methods require cooperative matching procedures [9], and gradient-based methods involve some smoothing constraint [10,11]. Thus, we have designed a method whose first task is to cope with these discontinuities, which are tied to contours in the image, such as occluding contours, joint ones . . .

We propose a framework for image flow analysis consisting of three major stages; i.e.:
- determine moving edges by some local process;
- link these edges and integrate motion information along contours;
- propagate motion estimation through homogeneous re-
Two hypotheses (or local configurations) can be obtained inside these contours.

The two first issues are addressed in this paper.

The first step can be considered as an early processing whose output contains location and spatial direction of an edge element and component of its displacement in the direction perpendicular to the local orientation of the edge. It is well-known that only such partial motion information can be reached by local operations (this point is often referred to as the aperture problem). A maximum likelihood method, based on some local modeling has been designed for this purpose.

Then constraints provided by these measurements gained from the first stage can be combined to compute the velocity field along contours, if however variations in spatial orientations occur along such contours. This computation results from the minimization of some simple functional by a stochastic gradient algorithm.

II LOCAL DETERMINATION OF A MOVING EDGE

II.1 - Modeling of a moving edge

An image sequence is considered as a 3D-space \((x,y,t)\). A spatial 2D-edge in an image is modeled as a small linear segment. Hence a moving 2D-edge is locally modeled as a small planar patch in the spatio-temporal 3D-space \((x,y,t)\). The direction \(\theta\) (w.r.t. the x-axis) of the 2D-edge centered in \((x_0,y_0)\) in the xy-plane at time \(t_0\) and its velocity \(\dot{y} = (\frac{dx}{dt}, \frac{dy}{dt}, 1)\) determine the orientation of this planar patch (see figure 1). This planar modeling is equivalent to the first order approximation that most gradient-based methods take into account.

![Figure 1: Local modeling of a moving edge as a planar patch.](image)

Let us consider an elementary volume \(\Pi\), in the 3D-space \((x,y,t)\), located around point \(E=(x_0,y_0,t_0)\). Two hypotheses (or local configurations) can be actuating:

\(H_0\): there is no spatio-temporal edge inside \(\Pi\); then the intensity distribution within \(\Pi\) is modeled as \(c_0+b\), where \(c_0\) is a constant and \(b\) denotes a zero-mean Gaussian noise with variance \(\sigma^2\).

\(H_1\): there is a spatio-temporal edge inside \(\Pi\), i.e. a small planar patch \(P\) splitting \(\Pi\) between two sub-regions, \(\Pi_1\) and \(\Pi_2\). Then the intensity distribution is modeled as: \(c_1+b\) within \(\Pi_1\); \(c_2+b\) within \(\Pi_2\), where \(c_1\) and \(c_2\) are two different constants.

The orientation of the planar patch can be defined by the two following angles: \(\theta\) (w.r.t. to the x-axis) and \(\psi\) (w.r.t. to the t-axis) as illustrated in figure 1. The component \(V\) of \(\dot{y}\) perpendicular to the spatial 2D-edge and projected in plane \(t-t_0\), is given by: \(V = \tan \psi\). It is obvious that only this component \(V\) can be inferred from the local determination of this planar patch. Note that the case of a static edge belongs to hypothesis \(H_2\); \(V=(c_1,1)\) and \(\psi=0\). Indeed such an edge will be considered as a "moving" edge, whose displacement is zero.

The problem now is how to select one hypothesis versus the other one. The test in order to decide between these two hypotheses will be designed using some maximum likelihood scheme.

II.2 - Maximum likelihood test

Details in mathematical developments can be found in [12], concerning the maximum likelihood test designed for detecting moving edges along with estimating their parameters. It is expressed by:

\[
\max_{\ell, \theta, \psi} \min_{c_1, c_2} \text{LRV} \geq \lambda
\]

where LRV is the log-ratio of likelihood functions \(L_1\) and \(L_0\), respectively associated with hypotheses \(H_1\) and \(H_0\). The likelihood function is merely the joint probability density function of the intensities within elementary volume \(\Pi\). It is easily derived as Gaussian distributions are involved and independent intensity random variables are assumed. \(\lambda\) is a predetermined threshold.

Clearly, hypothesis \(H_1\) is selected if the obtained maximum value of LRV is greater than \(\lambda\). Then one can conclude that a moving edge is located at \(E\) with spatial direction \(\theta\) and "perpendicular" velocity \(V = \tan \psi\), where \(\ell, \theta, \psi\) are precisely values of \((\ell, \theta, \psi)\) which have satisfied the mentioned criterium (1).

Yet one problem arises. No analytical closed-forms can be derived to express the optimal estimators \(\hat{\ell}, \hat{\theta}, \hat{\psi}\) corresponding to the geometrical characteristics of the model. Thus a predefined set of given configurations \(\Phi_j\) \(j=1,\ldots, G\) will be considered.

For a given geometric configuration \((\theta_j, \psi_j)\), the optimal estimators \(\hat{\ell}_j\) \(j=0,1,2\) satisfying intensity aspects satisfy:

\[
\frac{\partial \text{LRV}(\ell, \theta, \psi)}{3c_1^2} = 0
\]

which leads to

\[
\hat{c}_0 = \frac{1}{n} \sum_{p \in \Pi} f(p) ; \quad \hat{c}_1 = \frac{1}{n_1} \sum_{p \in \Pi_1} f(p) ; \quad \hat{c}_2 = \frac{1}{n_2} \sum_{p \in \Pi_2} f(p)
\]

where \(f(p)\) are observed intensity values within \(\Pi\) (resp. \(n_1, n_2\)) is the number of points within \(\Pi\) (resp. \(\Pi_1, \Pi_2\)).

II.3 - Computational scheme

It turns out, [12], that maximizing LRV comes to maximizing the following expression:
The same does not hold for differential inclusion boundaries and concerning measurable motion where circle arc or rotation component, could also be other sets of masks to be considered in expression of inherent restrictions concerning kinds of edges likely to be successfully handled (in particular, occlusion boundaries) and concerning measurable motion magnitude. The same does not hold for differential methods. The extent of measured motion is directly constrained by the smoothing extent used to compute the spatial gradient of the image intensity. Therefore, the differential approach is more appropriate for small displacements. Moreover, flow fields are often incorrect near occlusion, since assumptions required for differentiation do not hold any longer in such areas. On the other hand, this maximum-likelihood technique can be CPU-time consuming with a general-purpose computer, but CPU-time can be drastically reduced if an array processor is used.

III COMPLETE ESTIMATION OF THE VELOCITY FIELD ALONG CONTOURS

In the previous section, a procedure has been described which detects moving edges and, at the same time, estimates their local spatial direction and component of their velocity perpendicular to the contour. The goal of the second stage is to compute component of velocity vectors tangent to the contour.

In order to achieve the second stage of the image flow analysis, edge linking is prerequisite. To this end, only local spatial directions of detected moving edges are taken into account. One-pixel gaps can be filled up. The linking technique is similar to the one presented in [13]. Then, we get a set of contours, i.e., a set of chains of linked spatiotemporal edges.

To compute the entire velocity field along the contours, the second stage of analysis must combine the local measurements yielded by the first stage. This combination stage is efficient if enough variations in spatial orientation occur along obtained contours. For instance, a straight line contour remains a singular case.

Let \( \mathbf{v} = (u^x, u^y) = \left( \frac{dx}{dt}, \frac{dy}{dt} \right) \) be the restriction of the velocity field to the plane \((x, y)\). Let us consider \( \mathbf{e}_L(\omega) = \omega \cdot \mathbf{v}_L^T - \mathbf{V}_L^T \) (6)

where \( \mathbf{y} \) is the velocity field to be estimated, \( \mathbf{e}_L \) is the unitary vector normal to the local edge element at point \( \mathbf{e}_L \), \( \mathbf{n}_L = (-\sin \theta_L, \cos \theta_L) \), \( \mathbf{V}_L \) is the measured perpendicular component of velocity at point \( \mathbf{e}_L \), \( \mathbf{e}_L(\omega) \) is supposed to be a stationary random variable.

Then, the measurement of velocity field \( \mathbf{v} \) along a given contour \( \mathbf{C} \) is formulated as the minimization of the following function:

\[
J(\omega) = \frac{1}{2} \mathbb{E}_C (\mathbf{e}_L(\omega))^2
\]

(7)

where \( \mathbb{E} \) denotes expectation. Motivations for such a criterion can be found in [14]. A stochastic gradient algorithm is used to minimize \( J(\omega) \). The recursive estimation is pursued from one point to its successor in the chain. More precisely, it is expressed as follows:

\[
\omega_{t+1} = T \omega_t - \gamma \cdot \mathbf{V}_t \mathbf{e}_L(\omega_t) \cdot \mathbf{e}_L(\omega_t)
\]

where \( \gamma \) is a gain matrix, and \( \mathbf{V} \) denotes the gradient.
IV RESULTS

IV.1 - Results concerning the first analysis stage

Experiments on computer-generated images have been performed in order to warrant the estimation method of moving edges. Different kinds of motion have been considered (translation of the camera along its view axis, object rotation in the image plane). Results are presented in [17]. The algorithm has also been applied to actual images. Only two successive images are considered for each example reported here. Hence each mask corresponds to coefficients $a_j$'s for each predefined configuration $\theta_j,j=1,\ldots,G$, divided into two sub-masks. These $G$ masks are computed once the geometric configurations are chosen. Then they are available when images are processed. Choosing angle $\psi_4$ is equivalent to choosing displacement magnitude $v_4$ in the direction perpendicular to the local linear edge element whose spatial direction is $\theta_4$. Therefore, the location of the second submask in the second image with respect to location of the first one centered at current point $\ell$ in the first image is given by $V_{\ell}h_{\psi_4}$. Then, the $G$ configurations can also be denoted as $\{(\theta_1,\psi_4), q=1,Q\}$ with $RQ=W$. Thus the function $\text{CRV}$ can be written as follows:

$$\text{CRV}(\ell,\psi_{rd}) = \sum_{i \in E_i} \text{CRV}(\ell,\phi_{rd})$$

$$= |\text{CRV}(\ell,\phi_{rd}) + \sum_{i \in E_{i}} \text{CRV}(\ell,\phi_{rd})| \tag{9}$$

In order to save CPU-time, convolution operations with the whole set of masks, as previously explained, are not actually computed for each point $\ell$. If $\text{CRV}(\ell,\phi_{rd})<a$, with $a=0.25$, computations corresponding to the evaluation of $\text{CRV}(\ell,\phi_{rd})$, with $q=1,Q$, stop and $\text{CRV}(\ell,\phi_{rd})$, with $q=1,Q$ are set to 0.

The first example is extracted from a natural sequence acquired by a camera and depicting an urban scene. Figure 2 shows the first image. The set of masks consists of 66 masks including six possible spatial directions: $\theta = 0^\circ, 30^\circ, 60^\circ, 90^\circ, 120^\circ$, $150^\circ$ and eleven possible perpendicular displacement magnitudes, $v = -5, \ldots, -1, 0, 1, \ldots, 5$ ($G=66, R=6, Q=11$). Submask size is 5x5 pixels. The estimated perpendicular displacement field is presented in figure 3 along with spatial edges, for an image part.

An evaluation of the correctness of the resulting estimation is available. By means of some other tool, motion is known to be three pixels to the left in the whole image except for subparts corresponding to hung linen, some bushes. Quite satisfactory results are obtained.

The second example includes two images of a printer acquired by a CCD camera (Figure 4). Only the printer has been moved from one image to the next, camera and background remain fixed. 84 masks have been considered, that is to say four possible spatial directions $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$, and 21 possible perpendicular displacements $v = -10, \ldots, -1, 0, 1, \ldots, 10$. (Of course, metric is adapted when directions other than horizontal and vertical are considered). Determined moving edges are shown in Figure 5 with their perpendicular displacement, which can eventually be none.

IV.2 - Results concerning the second analysis stage

Two sets of experiments are presented involving computer-generated images including a single polygonal object and two kinds of motion: uniform translation and in-plane rotation. Superimposed silhouettes of the object in two successive positions are shown for both cases, respectively in Fig. 6a and 7a. Of course, the method is not restricted to such cases. In Fig. 6b and 7b is drawn the perpendicular displacement field. It has been estimated using the algorithm presented in this paper. Fig. 6c and 7c show the resulting complete displacement field after two recursive estimation cycles around the boundary. The last example points out that varying displacement field can be successfully handled.

V FUTURE WORK

Future research directions mainly include:
- corner displacement estimation (as complementary processing after linking)
- detection of possible motion boundaries along contours in parallel with the recursive estimation (e.g., this may happen if a boundary portion is an occlusion one) in order to reinitialize the recursion.

REFERENCES


ACKNOWLEDGEMENT

The author would like to thank Albert Benveniste for useful comments.

Graphics Interface '86 Vision Interface '86
Fig. 6a: Superimposed silhouettes of polygonal object 1

Fig. 6b: Perpendicular displacement field

Fig. 6c: Resulting complete displacement field

\[ \gamma = \begin{pmatrix} 0.03 & 0.01 \\ 0.01 & 0.03 \end{pmatrix} \]

Fig. 7a: Superimposed silhouettes of polygonal object 2

Fig. 7b: Perpendicular displacement field

Fig. 7c: Resulting complete displacement field

\[ \gamma = \begin{pmatrix} 0.04 & 0.01 \\ 0.01 & 0.04 \end{pmatrix} \]