# Velocity Vectors for Features of Sequential Oceanographic Images

E. C. Cho, S. S. Iyengar, Fellow, IEEE, Guna Seetharaman, Member, IEEE, Ronald J. Holyer, and Matthew Lybanon

Abstract—This paper investigates a fundamental problem of determining the position, orientation, and velocity field of the Gulf Stream in time-varying imagery. We propose an approximation method to characterize the deformation of these image motions for the purpose of estimating the velocity field of these images. The technique is focused on the interpretation of the change in the extracted features of the Gulf Stream. The underlying technique employs a triangulation of the region by a simplicial approximation of the velocity field on each triangle. A generalized computational framework, an outline of the mathematical foundation, and an implementation are presented in this paper.

Index Terms-Features, oceanographic images, simplicial approximations, triangulation, velocity vectors.

#### I. INTRODUCTION

NFRARED (IR) images of the ocean obtained from satellite sensors are widely used for the sensors are sensors are widely used for the sensors are senso sensors are widely used for the study of ocean dynamics (Garcia [4], Kelly [6], [7], and Vastano [13], [14]). One oceanographic application of satellite IR imagery that is especially fruitful is the study of mesoscale features. Streams, cold eddies, and warm eddies are examples of mesoscale ocean features with dimensions on the order of 50-300 km. An example of a north Atlantic image is shown in Fig. 1. The Gulf Stream and its associated eddies are examples of mesoscale features. The Gulf Stream is warmer than the Surgasso Sea to its south and much warmer than the waters to its north. The movement of these features compounds the problems associated with the detection of features. For a general problem on edge detection of these features, refer to Krishnamurthy et al. [8], Holyer [5], and Stommel [11].

#### A. Motivation

With the present increased interest in climatology and global change, many studies are under way at the Naval Research Laboratory, Stennis Space Center, MS, involving the analysis of large data sets of IR imagery. Oceanographers desire

Manuscript received June 10, 1996; revised March 10, 1997. This work was supported in part by a grant from the Office of Naval Research through the Naval Research Laboratory, Bay St. Louis, MS, by a grant from the Kentucky National Science Foundation EPSCoR, and by another grant from the National Science Foundation under Contract NSF-IRI-9210926.

E. C. Cho is with Kentucky State University, Frankfort, KY 40601 USA.

S. S. Iyengar is with Louisiana State University, Baton Rouge, LA 70803-4020 USA (e-mail: iyengar@bit.csc.lsu.edu).

G. Seetharaman is with the University of Southwestern Louisiana, Lafayette, LA 70501 USA.

R. J. Holyer and M. Lybanon are with the Naval Research Laboratory, Bay St. Louis, MS 39529 USA.

Publisher Item Identifier S 0196-2892(98)01133-4.

Fig. 1. Advanced Very High Resolution Radiometer (AVHRR) image of the Gulf Stream region of the north Atlantic acquired on April 17 and 18, 1989. This is a warmest-pixel composite of images acquired by an AVHRR aboard the NOAA-11 satellite. The rectangular area highlights the location of the dynamic feature that is being tracked (refer to Section III-F and Fig. 7).

accurate methods of tracking features in satellite images of the ocean to observe and quantify surface layer dynamics. IR images of the ocean showing sea surface temperatures are widely used for studies of this type. (Kelly [6] and Vastano [13]).

Estimating velocity vectors of features in oceanographic images remained an open problem for a long time. The high deformation of these features from image to image compounds the problem. Previous oceanographic work includes inferring the velocity field from image sequences of sea surface temperature, which follow features without regard to the actual temperatures (Vastano [13] and Emery et al. [3]) and which use the heat equation and the measured sea surface temperature (Kelly [6]). The maximum cross-correlation (MCC) method of Emery et al. [3] is a computational method for deriving sea surface advective velocities that consist of identifying the MCC in a lagged matrix between two subareas of a pair of sequential images. The MCC method, however, is insensitive to rotation motion. For a comparison of velocity estimates from different methods, refer to Kelly [7].

This paper describes a comprehensive methodology for estimating the velocity vectors of the features using simplicial approximation and discusses the implementation of the algo-



rithm. The simplicial approximation decomposes an arbitrarily shaped two-dimensional (2-D) region into a set of nonoverlapping triangles. It facilitates a method of interpolating functions over that region. The region must be a simply connected set of points, and the value of the smooth function must be known at many locations. We present some new results in the velocity of the image flow on oceanographic images, more specifically, we discuss a new approximation method to estimate the velocity field of the Gulf Stream from a sequence of satellite images. In this method, connected components of the region representing the Gulf Stream are identified and triangulated and the velocity field on the region is estimated by an affine approximation on each triangle.

The simplicial approximation of geometric objects and the mappings between them is a well-established theory in pure mathematics (for example, Rotman [10]). However, its implementation for practical applications is not straightforward and requires more work to be useful for our purpose. The idea of simplicial approximation is used in the finite element method, for example, in the computation of numerical solution to partial differential equations. In particular, when dealing with irregular shapes, this method is more efficient since the finite element method allows putting computational elements where they are needed. (Babuska [1] and Press [9]).

#### **II. MATHEMATICAL PRELIMINARIES**

An image f is a map defined on a rectangular array, where f(x, y) is the value at the coordinate (x, y) in the image. The array represents a grid of points located on a bounded region in  $R^2$ . We introduce some definitions and notation in *n*-dimensional Euclidean space as well as affine geometry, which is necessary for our application in oceanography. We will be concerned with the 2-D case. However, we give definitions and notation in a general-dimensional affine geometry, which is useful for other image analysis studies.

#### A. Euclidean Spaces and Affine Subsets

Images may be viewed as a map of a domain in the 2-D plane onto the set  $\{0,1\}$  (if the image is bilevel) or onto the closed unit interval [0,1] (if the image is grayscale). Bilevel imagery assigns the values one or zero to each point in the domain in the plane. The plane is the 2-D Euclidean space. We start with the more general Euclidean *n*-space. The set

$$R^n = \{(x_1, \ldots, x_n) : x_i \in R\}$$

with the usual Euclidean inner product

$$\langle x, y \rangle = \sum_{1}^{n} x_i y_i$$

is called the Euclidean n-space. The usual Euclidean norm

$$||x|| = \langle x, x \rangle^{\frac{1}{2}} = \left(\sum_{1}^{n} x_{i}^{2}\right)^{\frac{1}{2}}$$

induces the metric

$$d(x,y) = ||y - x||$$
  
=  $((y_1 - x_1)^2 + \dots + (y_n - x_n)^2)^{\frac{1}{2}}$  (1)

on  $\mathbb{R}^n$ . On  $\mathbb{R}^2$ , (1) is the familiar Euclidean distance derived from the Pythagorean theorem. In this paper, a vector v is a row vector and its transpose  $v^T$  is a column vector.

Vectors may be viewed as transformations of the whole space into itself moving all the points of the whole space in the same direction by the same magnitude. In other words, a vector may be viewed as a constant vector field on the whole space. Vectors, however, are also viewed as representations of position in the space. With a fixed point specified as the origin, every point in the image can be represented as a position vector in the image plane. Regions of interest in oceanographic images, for example, streams or eddies, are of irregular shape. However, the connected components of the regions can be closely approximated by more regular shapes, namely, triangles. To give a general and precise idea of such an approximation, we need the following definitions.

Definition 1: A subset A of  $\mathbb{R}^n$  is called an affine subset if, for every pair of distinct points of A, the line determined by the points is contained in A.

The only affine subsets of the Euclidean plane  $R^2$  are either the empty set, single point sets (trivially), lines, or the whole plane  $R^2$ . If we require that the line segment joining two points instead of the whole line be included, the set is called convex. Obviously, the condition of being affine is stronger than the condition of being convex. A disk or a closed region bounded by lines, for example, the triangle

$$\{(x,y) : x+y \le 1, x \ge 0, y \ge 0\}$$

is convex but not affine. Regions in oceanographic images are not convex in general, however, we can approximate an irregular region by convex subregions. The subdivision can be done by the simplicial approximation described in the following section.

Definition 2: A set of k + 1 points  $\{p_0, p_1, \ldots, p_k\}$  in  $\mathbb{R}^n$  are said to form an affine independent set if the k vectors  $p_1 - p_0, p_2 - p_0, \ldots, p_k - p_0$  are linearly independent.

For example, three vertex points of a proper triangle are affine independent, but any four points on a plane are affine dependent. The definition of affine independence does not depend upon specific ordering of these points.

*Remark:* The points  $p_0, p_1, \ldots, p_k$  in  $\mathbb{R}^n$  are affine independent if and only if the  $k \times n$  matrix

$$\begin{pmatrix} p_1 - p_0 \\ p_2 - p_0 \\ \vdots \\ p_k - p_0 \end{pmatrix}$$

has rank k. Note each  $p_i - p_0$  is a row vector of dimension n.

The rank of a matrix is the maximum number of rows (or columns) that are linearly independent, which is easily found by applying elementary row operations to the matrix. It is obvious that more than n + 1 points in  $\mathbb{R}^n$  cannot be affine independent. Points  $p_0, \ldots, p_n$  in  $\mathbb{R}^n$  are affine independent if

$$\begin{pmatrix} p_1 - p_0 \\ p_2 - p_0 \\ \vdots \\ p_n - p_0 \end{pmatrix}$$

is nonzero. If  $p_0, \ldots, p_n$  are affine independent, any point  $p \in \mathbb{R}^n$  can be uniquely represented as

$$p = a_0 p_0 + a_1 p_1 + \dots + a_n p_n$$

for some real numbers  $a_i$  satisfying  $a_0 + a_1 + \cdots + a_n = 1$ . For example, when n = 2, the  $a_i$ 's are found by solving the following system of linear equations:

$$a_0 + a_1 + a_2 = 1$$
  
$$x_0a_0 + x_1a_1 + x_2a_2 = x$$
  
$$y_0a_0 + y_1a_1 + y_2a_2 = y$$

where p = (x, y) and  $p_i = (x_i, y_i)$ .

*Example 1:* The points  $p_0 = (1,1), p_1 = (2,1)$  and  $p_2 = (1,2)$  in  $\mathbb{R}^2$  are affine independent since the corresponding matrix

$$\begin{pmatrix} p_1 - p_0 \\ p_2 - p_0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

has a nonzero determinant equal to one. Thus, an arbitrary point on the plane can be uniquely represented by a combination of  $p_0$ ,  $p_1$ , and  $p_2$ . In fact, three points in  $\mathbb{R}^2$  are affine independent if and only if they are not collinear, in which case, the three points are said to be in general position.

#### B. Simplex and Simplicial Approximation

A simply connected region in oceanographic images [simply connected means the region is like a disk (topologically), so it has no holes inside] can be identified by its boundary curve. A ring-shaped region is not simply connected. The boundary curve can be approximated by a continuous piecewise linear curve (a concatenation of connected line segments). Therefore, a simply connected region is approximated by a polygonal region, which is a union of triangles. This is an example of simplicial approximation in one and two dimensions. We state some of the general definitions necessary for our application of simplicial approximation.

Let  $\{p_0, \ldots, p_n\}$  be an affine independent subset of  $\mathbb{R}^n$ . The convex hull of the set is the smallest convex set containing all  $p_i$ 's. It is the intersection of all convex sets containing these points. More explicitly, the convex hull of  $\{p_0, \ldots, p_n\}$  is the set of all affine combination of  $p_i$ 's

$$\left\{\sum_{0}^{n} a_{j} p_{j} : a_{j} \in R, a_{j} \ge 0, \sum_{0}^{n} a_{j} = 1\right\}.$$

For example, the convex hull of two distinct points is the line segment joining them and the convex hull of three noncollinear points is the triangle (including its interior) with vertices at those points. The more general definition follows.

Definition 3: The *n*-simplex generated by the set of affine independent points  $p_0, \ldots, p_n$  is the convex hull of these points and is denoted by  $[p_0, \ldots, p_n]$ .

*Example 2:* [0,1] is the unit closed interval, [(0,0), (1,0), (0,1)] is a right-angled isosceles triangle, and [(0,0,0), (1,0,0), (0,1,0), (0,0,1)] is a rectangular tetrahedron. Note that onesimplex is a line segment, two-simplex is a triangle, threesimplex is a tetrahedron, etc.

We anticipate that the domains of many image segments are in motion, and the effective way to track them is to use the *barycentric coordinates* of the pixel points in the image. The definition of barycentric coordinates and an example are given below to illustrate this point. Let  $e_i = (0, ..., 1, ..., 0)$ , where one appears in the *i*th place.  $\{e_1, ..., e_n\}$  is called the standard basis of  $\mathbb{R}^n$ . Let  $e_0$  be the zero vector. The standard *n*-simplex  $[e_0, e_1, ..., e_n]$  as a set is

$$\left\{ (x_1, \dots, x_n) \in \mathbb{R}^n : x_j \ge 0, \sum_{1}^n x_j \le 1 \right\}.$$

Definition 4: Let  $T = [p_0, \ldots, p_n]$ . Then any point  $p \in T$  is uniquely represented as

$$p = \sum_{0}^{n} a_{j} p_{j} \tag{2}$$

where  $a_j \ge 0$  for all j and  $\sum_{0}^{n} a_j = 1$ . Equation (2) is an affine combination of  $p_0, \ldots, p_n$ . The row vector  $(a_0, \ldots, a_n)$  is called the barycentric coordinate of p, with respect to the set:  $p_0, \ldots, p_n$ .

*Example 3:* The barycentric coordinate of a point  $(x_1, \ldots, x_n)$ , with respect to  $e_0, e_1, \ldots, e_n$ , is  $(1 - \sum_{j=1}^{n} x_j, x_1, \ldots, x_n)$ .

The barycentric coordinates of a point, with respect to a given set of affine independent points, are found by solving a linear equation as follows. Let  $p_0 = (x_0, y_0), p_1 = (x_1, y_1)$  and  $p_2 = (x_2, y_2)$  be affine independent points in  $\mathbb{R}^2$  and T be the two-simplex  $[p_0, p_1, p_2]$ . Let

$$p = a_0 p_0 + a_1 p_1 + a_2 p_2$$

where  $a_0, a_1, a_2 \ge 0, a_0+a_1+a_2 = 1$  is the affine combination of  $p_0, p_1$ , and  $p_2$ , representing a point  $p \in T$ . Let

$$v_0 = p_0$$
  

$$v = p - p_0 = (x - x_0, y - y_0)$$
  

$$v_1 = p_1 - p_0 = (x_1 - x_0, y_1 - y_0)$$
  

$$v_2 = p_2 - p_0 = (x_2 - x_0, y_2 - y_0)$$
  

$$A = (v_1^T, v_2^T).$$

Then solve the following system of linear equations  $v^T = A(a_1, a_2)^T$  or by substitution

$$\begin{pmatrix} x - x_0 \\ y - y_0 \end{pmatrix} = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}.$$

The above system of equations always has a unique solution since the  $2 \times 2$  matrix A is nonsingular, which follows from the assumption that  $v_1$  and  $v_2$  are linearly independent. The system

$$v^T = A(a_1, a_2)^T$$

has the solution

$$(a_1, a_2)^T = A^{-1} v^T$$

Once  $a_1$  and  $a_2$  are found,  $a_0$  is computed from the relation  $a_0 + a_1 + a_2 = 1$ .

A polygonal region P with n edges is decomposed as a union of n - 2 triangles, each of which is either disjoint from others or intersects at a common edge or vertex. More generally, any connected region R of a surface can be approximated by a simplicial complex, which is a collection of simplices (triangles, in dimension two) such that each simplex is either disjoint from others or intersects other simplices with one vertex or common whole edge. This is called a simplicial approximation of the region. Any smooth surface can be approximated by two-simplices (triangles).

Suppose a disk-shaped region in an oceanographic image sequence is moving and the corresponding regions in the following images in the sequence are identified. Instead of tracking every point on the boundary (a circle in this case) of the region, it is efficient to approximate the disk by an inscribed regular polygon and track the vertices in the image sequence. The unit disk can be approximated by an inscribed regular n-gon, which is the union of n isosceles triangles. The region can be approximated more closely by increasing the value of n.

#### C. Smooth Vector Fields

The notion of a vector field is useful to describe timevarying features in oceanographic images. A vector field associates a vector to each point in the domain of the image. The vector associated with a point corresponds to the movement of the point in the image sequence. A mapping that translates the whole space by a fixed vector v is represented by a constant vector field F(x,y) = v. For example, the vector (1,0) can be viewed as the constant vector field representing the mapping of the whole plane space onto itself and matching each point (x, y) to (x + 1, y). Most movements (including rigid motions), however, do not move points in this manner. For example, the counterclockwise rotation of  $R^2$ , by 90° about the origin moves the point (1,0) to a point on the unit circle, but the origin (0,0) into itself. The vector field representing this rotation will assign the vector (0,1) to the point (1,0) and the vector (0,0) to the point (0,0), as we will see in the following examples.

In oceanographic images, the regions of interests are those representing streams or eddies, hence, a domain in  $R^2$ . However, we give the definition of a vector field defined on a more general domain called a smooth manifold.

Definition 5: A vector field (more precisely, a tangent vector field) on a smooth manifold X is a smooth mapping of X, assigning each point p of X with a vector in the tangent space  $T_pX$  at the point p. The range of this mapping is the union of all tangent spaces  $T_pX$  called the tangent bundle of X and is denoted by TX.

*Remark* If X is a domain of  $R^2$ , the tangent bundle TX is just  $X \times R^2$ , a product bundle. We use the term "smooth" to mean continuously differentiable as many times as necessary. Vector fields in this paper are assumed to be smooth.

We may represent a smooth vector field on a domain  $D \subset R^2$  by

$$F(x,y) = (u(x,y), v(x,y))$$



Fig. 2. Velocity field assigns a vector  $\mathbf{v}(x, y)$  to each point (x, y) in the image domain. In this case,  $\mathbf{v}(x, y) = (-y, x)$ . The x component of the velocity, for all points located on the x axis, is zero since  $v_x = y = 0$ . Likewise, the y component of the velocity is zero for all points located on the y axis since  $v_y = x = 0$ .

where u(x, y) and v(x, y) are real-valued smooth functions defined on D.

Example 4: Let

$$F(x,y) = (-y,x) \tag{3}$$

for all  $(x, y) \in \mathbb{R}^2$ , that is, u(x, y) = -y and v(x, y) = x. It is a smooth vector field associated with the rotation of  $\mathbb{R}^2$  around the origin. (Fig. 2.)

Suppose each point p in a domain X is moving so that the trace of the point is a smooth curve in X. Then the tangent vector to the trace at each point is called the velocity vector at the point. This leads to the following.

Definition 6: The velocity field F(p) on X is a vector field on X that assigns each point p in X with the velocity vector at the point p.

The velocity field can be visualized by considering each particle at the position x moving with the velocity vector given by the velocity field.

*Definition 7:* The solutions of the system of differential equations

$$\frac{dx}{dt} = u(x,y), \quad \frac{dy}{dt} = v(x,y)$$

are called the integral curves of the velocity field F(x,y) = (u(x,y), v(x,y)). The integral curve describes the trajectory of a particle.

*Example 5:* Consider the previous example with field 3. It is easy to check that  $x = c \cos t$  and  $y = c \sin t$ , for any constant c, is a solution of the differential equation

$$\frac{dx}{dt} = -y, \quad \frac{dy}{dt} = x.$$

The parametrized curve

$$C(t) = (c\cos t, c\sin t)$$

represents the circle  $x^2 + y^2 = c^2$ . This means that the differential equation describes a dynamical system on the plane in which each particle rotates counterclockwise around the origin.

In the following sections, we establish a computational framework that adapts the affine mappings and approximation technique for vector fields.

## D. Affine Mappings and Affine Approximation Technique for Characterizing Motion

To estimate velocity fields from a sequence of oceanographic images, we will use affine mappings of a domain in  $R^2$  onto  $R^2$ . The estimation by affine mapping is efficient and simple to implement computationally since any affine mapping f from a two-simplex  $T = [p_0, p_1, p_2]$  onto  $R^2$  is determined by the values  $f(p_0), f(p_1)$  and  $f(p_2)$  only and regions in oceanographic images are approximated by a finite number of two-simplices.

Let  $\{p_0, \ldots, p_m\}$  be affine independent and A be the affine set these points span, that is,

$$A = \left\{ \sum_{0}^{n} a_j p_j : \sum_{0}^{n} a_j = 1 \right\}.$$

Here  $a_j$  may be negative. If  $a_j \ge 0$  for all j, then A is an n-simplex.

Definition 8: An affine mapping f of A onto  $\mathbb{R}^n$  is a mapping preserving the affine combination, that is, f satisfies

$$f\left(\sum_{0}^{n} a_j p_j\right) = \sum_{0}^{n} a_j f(p_j)$$

whenever  $\sum_{0}^{n} a_j = 1$ . The restriction of f to the *n*-simplex  $[p_0, \ldots, p_n]$  is also called an affine mapping.

For example, the mapping assigning  $v + v_0$  to  $v \in R^2$  for some fixed vector  $v_0$  in  $R^2$  is an affine mapping of  $R^2$  onto  $R^2$ . This mapping is called the translation by  $v_0$ .

*Remark:* Any linear mapping or a translation mapping of  $R^n$  onto itself is an affine mapping. Any affine mapping of  $R^1$  onto  $R^1$  is of the form y = ax + b, where a, b are real constants. Any rotation of  $R^2$  or a translation of  $R^2$  is an affine mapping, and the composition of affine mappings is affine. The image of an affine set (or a simplex) under an affine mapping is affine (or a simplex).

The following example represents the rotation of the plane by  $\pi/2$  radian counterclockwise about the origin as an affine mapping by specifying the movement of the three vertices of a specific two-simplex (triangle).

*Example 6:* Let  $p_0 = (0,0), p_1 = (1,0)$  and  $p_2 = (0,1)$ . If f is an affine mapping of  $T = [p_0, p_1, p_2]$  onto  $R^2$  with  $f(p_0) = (0,0), f(p_1) = p_2 = (0,1)$  and  $f(p_2) = -p_1 = (-1,0)$ , since  $(a_1, a_2) = a_0p_0 + a_1p_1 + a_2p_2$ 

$$f(a_1, a_2) = f\left(\sum_{i=0}^{2} a_i p_i\right)$$
  
=  $\sum_{i=0}^{2} a_i f(p_i)$   
=  $a_0(0, 0) + a_1(0, 1) + a_2(-1, 0)$   
=  $(-a_2, a_1)$ 

where  $a_0, a_1, a_2 \ge 0$  and  $\sum_{i=1}^{2} a_i = 1$ .

The value of an affine mapping f(x) at  $x \in T$  is the weighted average of  $f(p_i)$ 's, the values of f at the vertices of T. If  $x \in T$  is the midpoint of  $p_i$  and  $p_j$ , the value f(x) will be the midpoint of  $f(p_i)$  and  $f(p_j)$ . More generally, the barycenter of a simplex S is mapped to the barycenter of the image f(S). The next example describes an ideal (but highly unrealistic) situation in which the region of interest in the oceanographic image sequence varies only in position as a translation (a simple rigid motion). In this case, we need only identify one point of the region and the corresponding point in the subsequent images to estimate the velocity vector field.

*Example 7:* The constant mapping f of T onto  $R^2$  given by f(x) = (a, b) for all  $x \in T$  is an affine mapping. f is the velocity field on T associated with the translation of T by the vector (a, b), which maps  $x \in T$  onto x + f(x) = x + (a, b).

Now we consider a more general situation in which the regions vary both their shapes and locations with time. Obviously, this is not a rigid motion. We do not need to assume that the movement of the region is smooth to apply simplicial approximation. We need only piecewise continuity, which means the movement as a mapping is continuous on each component covering the domain. This allows some discontinuities, for example, at certain points or on certain portions of arcs in the domain.

It is reasonable to assume smoothness or at least piecewise smoothness in the study of oceanographic images at mesoscales. Suppose, for simplicity, a triangular region  $T = [p_0, p_1, p_2]$  in an oceanographic image has smooth motion. Then there corresponds a smooth velocity field F defined on T. If the (vector) values of the velocity field F at the vertices of T are known, say,  $F(p_0) = u_0, F(p_1) = u_1$  and  $F(p_2) = u_2$ , we can interpolate the value F(p) at any point p in T by the affine mapping of T, which coincides with F at the vertices of T, that is

$$F(p) = a_0 F(p_0) + a_1 F(p_1) + a_2 F(p_2)$$
  
=  $a_0 u_0 + a_1 u_1 + a_2 u_2$  (4)

where  $p = a_0p_0 + a_1p_1 + a_2p_2$ . Equation (4) is a natural generalization of the linear interpolation of a function with one variable. The error bound of a linear interpolation to a smooth function is easily found from the second-order term of the Taylor series of the function. Similarly, we can estimate an error bound of the affine approximation of the vector field F if F is smooth. To estimate the error bounds for affine approximations, we need the following definitions.

Definition 9: The diameter of an *n*-simplex  $T = [p_0, ..., p_n]$ , denoted by diam *T*, is the  $\max\{||p_i - p_j|| : i, j = 0, ..., n\}$ . This coincides with the usual definition of the diameter of any subset *S* of Euclidean space, the  $\sup\{||p-q|| : p, q \in S\}$ .

Definition 10: The norm of an  $n \times n$  matrix A, denoted by ||A||, is given by

$$\sup\{||Ax||/||x|| : x \in \mathbb{R}^n, x \neq 0\}.$$

Now we state, without proof, a proposition giving an error bound of the affine approximation of the smooth vector field F on a triangle (Dieudonné [2]).

Y 2 Χ 2 -2

Fig. 3. Unit disk is an elementary area centered at (0,0) for infinity norm distance.

*Proposition:* Let  $F = (F_1, F_2)$  be a smooth vector field defined on  $T = [p_0, p_1, p_2]$  and  $p = a_0p_0 + a_1p_1 + a_2p_2$  be a point in T. Then

$$\left\| F\left(\sum_{0}^{2} a_{i} p_{i}\right) - \sum_{0}^{2} a_{i} F(p_{i}) \right\| \le M ||T||$$

where  $M = \max\{||DF(p)|| : p \in T\}$ , the maximum of the norms of the Jacobian DF of F (the  $2 \times 2$  matrix with the *ij*th entry  $(DF)_{ij} = \partial F_i / \partial x_j$  on T and ||T|| is the diameter of T.

Remark: This proposition holds for higher dimension n, and the vector field f needs only be  $C^1$  (continuously differentiable, that is, all the first-order partial derivatives of f exist and are continuous).

*Remark:* It is simpler to use an alternate definition of norm on  $\mathbb{R}^n$  given by  $||x|| = \max\{|x_1|, \dots, |x_n|\}$ , which induces the distance on  $\mathbb{R}^n$ 

$$d(x,y) = ||x - y||$$
  
= max{|y<sub>1</sub> - x<sub>1</sub>|,..., |y<sub>n</sub> - x<sub>n</sub>|}

where  $x = (x_1, \ldots, x_n)$  and  $y = (y_1, \ldots, y_n)$ , and the alternate definition of matrix norm

$$||A|| = \max\left\{\sum_{j=1}^{n} |a_{ij}| : i = 1, 2, \dots, n\right\}.$$

This is in fact the infinity-norm of the vector A. It is equivalent to the usual norms defined above, but simpler to compute. It may not be suitable for certain applications since the geometry under this distance is different from the usual Euclidean geometry, as the following example shows. The set  $\{x \in \mathbb{R}^2 :$  $d(x, (0, 0)) \le 1$ , the "unit disk" centered at (0,0), with respect to this distance on  $\mathbb{R}^2$ , is the square  $[-1,1] \times [-1,1]$  in  $\mathbb{R}^2$ . (Fig. 3.)

Let F be a velocity vector field representing the movement of a region D in the image sequence. We may assume Fis smooth on the region D and D is connected, otherwise, we will apply our procedure separately to each connected component D on which F is smooth. Suppose D is given a triangulation, that is, D is approximated by a 2-D simplicial complex. Suppose the (vector) values of F at each vertex of the

simplicial complex triangulating D are known (or measured). Then we can approximate the value of F at any point  $p \in D$ by applying the above affine interpolation on each simplex. We can also estimate an error bound of the interpolation by the proposition, assuming we have an estimation of the derivatives of F, which is not available in practice. We may, however, use the derivatives of the functions approximating F.

### E. Algorithm for the Affine Approximation of a Smooth Vector Field

The algorithm of approximating a smooth vector field f on a triangulated region D, whose (vector) values are known at the vertices of simplices in the triangulation of D is given as follows.

- 1) Generate as many points as necessary to approximate the values of f. These points are easily generated by repeatedly applying barycentric subdivision (defined below) of D.
- 2) For any point p in D, we first identify the simplex T to which p belongs.
- 3) Approximate the value of F(p), based on the values of F at the vertices of the simplex T.

If p happens to be a vertex, there is no need to interpolate, and if p belongs to an edge, the interpolation becomes simpler by using only the values at the two vertices at the end of the edges. These cases are covered by the general interpolation formula and there is no ambiguity even if the point p belongs to more than one simplex. Any continuous mapping between simplicial complexes can be approximated by simplicial mappings, and the approximation can be made arbitrarily close by subdividing the simplex into smaller simplices.

To define barycentric subdivision, we need the following definitions.

Definition 11: The barycenter, also called the centroid, of an *n*-simplex is the point with barycentric coordinate  $(1/n+1,\ldots,1/n+1).$ 

The barycentric subdivision is defined inductively as follows.

Definition 12: The barycentric subdivision of a zerosimplex (a point) is the simplex itself. Let S be an n-simplex  $n \geq 1$ . If  $f_0, \ldots, f_n$  are (n-1)-dimensional faces of S and b is the barycenter of S, the barycentric subdivision of Sconsists of all *n*-simplexes spanned by *b* and (n-1)-simplexes in the barycentric subdivisions of  $f_i$ 's.

Barycentric subdivision divides a one-simplex (a line segment) into two line segments of half size (by adding the midpoint), a two-simplex (a triangle) into six two-simplexes by subdividing each edge (one-simplex) into two half line segments and joining the original vertices and the barycenters (midpoints) of the edges with the barycenter (centroid) of the triangle.

*Example 8:* The barycentric subdivision of a one-simplex (an interval) [0,1] consists of the vertices  $\{0\}, \{\frac{1}{2}\},$ and  $\{1\}$ and two open intervals  $(0, \frac{1}{2})$  and  $(\frac{1}{2}, 1)$ .

The barycentric subdivision of a two-simplex (triangle) consists of six smaller triangles, twelve one-faces (edges), and seven zero-faces (vertices). For a triangle whose vertices





Fig. 4. *Barycentric* subdivision of a 2-D simplex. The given triangle is made of three 2-D points, and the triangle is decomposed into six smaller triangles all sharing a common vertex *G* at its centroid.

are located at, a, b, and c on a 2-D plane, the seven vertices are  $\{a, b, c, a + b/2, b + c/2, c + a/2, a + b + c/3\}$ . This is illustrated in Fig. 4.

Barycentric subdivision can be applied repeatedly to make the mesh (the supremum of the diameters of faces of the simplex) size arbitrarily small.

*Example 9:* Let f be a mapping from the unit closed interval [0, 8] to itself given by  $f(x) = \frac{1}{8}x^2$ . The barycentric subdivision of [0, 8] consists of the vertices {0}, {4}, and {8} and two open intervals (0, 4) and (4, 8). The second barycentric subdivision of [0, 8] consists of the vertices {0}, {2}, {4}, {6}, and {8} and four open intervals (0, 2), (2, 4), (4, 6), and (6, 8). The estimate of f(1) using the second barycentric subdivision is

$$\frac{1}{2}(f(0) + f(2)) = \frac{1}{4}$$

while the true value is 1/8. The estimate of f(7) is

$$\frac{1}{2}(f(6) + f(8)) = \frac{1}{2}\left(\frac{36}{8} + 8\right) = \frac{50}{8}$$

while the true value is 49/8.

Example 10: Let

$$D = \{(x, y) \mid x \ge 0, y \ge 0, x + y \le 2\}$$

be the region on which a smooth vector field f is defined by

$$f(x,y) = (x^2 - y^2, xy).$$

For example, f(2,0) = (4,0), f(0,2) = (-4,0), and f(0,0) = (0,0). Suppose D is triangulated by the first barycentric subdivision. The triangulation has vertices (0, 0), (1, 0), (2, 0), (1, 1), (0, 2), (0, 1) and (2/3, 2/3). The estimate value of f at (1/2, 1/3) is found as follows. First we note the point (1/2, 1/3) is in the subsimplex [(0, 0), (2/3, 2/3), (1,0)] and the point (1/2, 1/3) has barycentric coordinates (1/2, 1/6, 1/3), that is

$$\left(\frac{1}{2},\frac{1}{3}\right) = \frac{1}{2}\left(\frac{2}{3},\frac{2}{3}\right) + \frac{1}{6}(1,0) + \frac{1}{3}(0,0).$$

Hence the affine estimate of f(1/2, 1/3) is

$$\frac{1}{2}f\left(\frac{2}{3},\frac{2}{3}\right) + \frac{1}{6}f(1,0) + \frac{1}{3}f(0,0)$$
$$= \frac{1}{2}\left(0,\frac{4}{9}\right) + \frac{1}{6}(1,0)$$
$$= \left(\frac{1}{6},\frac{2}{9}\right)$$



Fig. 5. Piecewise linear approximation of a parabola. The parabola  $y = x^2$  for  $-3 \le x \le 2$  is approximated by three line segments.

while the true value of f(1/2, 1/3) is (5/36, 1/6). The error is (-1/36, -1/18).

#### III. APPLICATION

#### A. Piecewise Linear Approximation of a Curve

The boundaries of regions in mesoscale oceanographic images can be approximated by piecewise smooth curves (cubic splines, for example) or piecewise linear curves. In the following, we give detailed computational procedures for constructing a piecewise linear approximation to the boundary curve of a region. The simplicial approximation of vector fields describing the changes of the region will be based on the simplicial approximation of the region, which is based on the piecewise linear approximation of the boundary curve of the region.

Let  $(p_1, \ldots, p_n)$  be a sequence of data points representing a smooth curve in an image. The idea of piecewise linear approximation is to approximate the portion of the curve between  $p_i$  and  $p_j$  by the line segment  $[p_i, p_j]$  if the points in between, namely,  $p_i, p_{i+1}, \ldots, p_j$  are approximately collinear. Higher order polynomials may be used, a parabola, for example, if the physics of underlying motion requires such higher order terms. Fig. 5 illustrates an instance in which a parabolic curve is more useful than piecewise line segments.

We consider points  $p_i, p_j$  and  $p_k$  for i < j < k to be approximately collinear if the distance between  $p_j$  and the line passing through  $p_i$  and  $p_k$  is less than a given value  $\epsilon > 0$ .

Let d(p, [q, r]) denote the distance between the point p and the line segment [q, r]. It is computed by

$$d(p,[q,r]) = \left\| (p-q) - \frac{\langle p-q,r-q \rangle}{\langle r-q,r-q \rangle} (r-q) \right\|$$

where  $\langle a, b \rangle$  is the Euclidean inner product of the vectors  $a = (a_1, a_2)$  and  $b = (b_1, b_2)$ , which is given by  $a_1b_1 + a_2b_2$ . ||a|| is the Euclidean norm of the vector a given by  $||a|| = \sqrt{\langle a, a \rangle}$ . This formula follows from the fact that the distance between p and the line passing through q, r is the length of the vector p - q projected onto the direction normal to the line [q, r].

*Example 11:* The distance between the point (1, 2) and the line passing through (0, 0) and (3, 4) is

$$\begin{aligned} \left\| (1,2) - \frac{\langle (1,2), (3,4) \rangle}{\langle (3,4), (3,4) \rangle} (3,4) \right\| \\ &= \left\| (1,2) - \frac{11}{25} (3,4) \right\| \\ &= \left\| \left( \frac{-8}{25}, \frac{6}{25} \right) \right\| \\ &= \frac{2}{5}. \end{aligned}$$

Let  $(p_{i_1}, \ldots, p_{i_k})$  be the subsequence of the data points  $(p_1, \ldots, p_n)$  such that the points  $p_{i_j}, p_{i_j+1}, \ldots, p_{i_{j+1}}$  form a maximal chain of points that are almost collinear, with respect to a given  $\epsilon$  for each j. The points in the subsequence will be called break points.

For example, if

$$d(p_2, [p_1, p_3]) \ge \epsilon$$

then  $p_2$  is included in the subsequence to form a line segment  $[p_1, p_2]$  and  $p_2$  is a break point. If

$$d(p_2, [p_1, p_3]) < \epsilon$$

but

$$d(p_2, [p_1, p_4]) \ge \epsilon$$

or

$$d(p_3, [p_1, p_4]) \ge \epsilon$$

then  $p_1, p_2, p_3$  is a maximal chain of almost collinear points and  $p_3$  is a break point.

Let  $\{q_0, q_1, \ldots, q_k\}$  be the set of all break points on the curve C. A parametrized form of the piecewise linear approximation f(t) of C is given by

$$f(t) = q_i + \frac{t - t_i}{t_{i+1} - t_i} (q_{i+1} - q_i)$$

for  $t \in (t_i, t_{i+1})$ ,  $i = 0, 1, \dots, k-1$ , where  $a = t_0 < t_1 < \dots < t_k = b$  is a partition of the domain [a, b] with  $f(t_i) = q_i$  for  $i = 0, 1, \dots, k$ . We will denote the piecewise linear approximation of C with break points  $q_0, q_1, \dots, q_k$  by  $[q_0, q_1, \dots, q_k]$ .

#### B. Finding Break Points

Region boundaries in oceanographic images may be viewed as smooth or very irregular, depending on the scale. In approximating such boundaries by a piecewise linear curve, we need to decide in advance how fine the approximating line segments will be. The following algorithm finds the points in the data where the approximating lines will be broken (but connected) to fit the data points within the preset error bound  $\epsilon$ .

Let  $(p_1, \ldots, p_n)$  be a sequence of data points representing a curve C. Let  $\epsilon > 0$  be given. We are interested in finding the subsequence  $(p_{i_1}, \ldots, p_{i_k})$  such that  $p_{i_1} = p_1$ ,  $p_{i_k} = p_n$  (the end points), and for every r, the points between  $p_{i_r}$  and  $p_{i_{r+1}}$  constitute the maximal chain of data points that are almost

collinear, with respect to  $\epsilon > 0$ . In other words, the points  $p_{i_r}, p_{i_r+1}, \ldots, p_{i_{r+1}}$  are almost collinear but including the point  $p_{i_{r+1}+1}$  (the point after  $p_{i_{r+1}}$ ) destroys the approximation of collinearity. The algorithm is implemented as follows:

Set 
$$q_1 = p_1$$
.  
if  $d(p_2, [q_1, p_3]) > \epsilon$ , then  
set  $q_2 = p_2$ ,  
elseif  $d(p_2, [q_1, p_4]) > \epsilon$  or  $d(p_3, [q_1, p_4]) > \epsilon$ , then  
set  $q_2 = p_3$ ,  
elseif  $d(p_2, [q_1, p_5]) > \epsilon$  or  $d(p_3, [q_1, p_5]) > \epsilon$   
or  $d(p_4, [q_1, p_5]) > \epsilon$  then  
set  $q_2 = p_4$ ,  
elseif ..., etc.

endif.

end of algorithm.

*Example 12:* Let  $p_i = (\cos t_i, \sin t_i), t_i = 2\pi i/n, i = 0, 1, ..., n$  be data points from the standard unit circle. If n = 30 and  $\epsilon = \frac{2-\sqrt{3}}{2}$ , the piecewise linear approximation is the hexagon with vertices at  $x_0, x_5, x_{10}, \ldots, x_{25}$ .

#### C. Finding Extreme Points

For a region in motion of an oceanographic image sequence, we assume that the points on the boundary correspond to the points on the boundaries of the corresponding regions in the image sequence. Sethi and Jain [12] showed the importance of detecting the action in images by using the idea of a correspondence problem. To estimate the velocity vector field of the motion, we assume that certain points on the boundaries can be located through the image sequence. We will locate the prominent points, that is, the points where the boundary curve changes direction rapidly. In differential geometric terms, these are the points where the curvature has a maximum absolute value. Then, the natural choice for feature points required to track a polygonal-shaped region would be its vertices, as shown in Fig. 6.

For simplicity, we write  $q_r = p_{i_r}$ , so that  $\{q_1, \ldots, q_k\}$  are nodes of the piecewise linear approximation of the curve C. From the construction, it is obvious that the slopes of the line segments change at the  $q_i$ 's. We are interested in locating the points where the slope of the curve (or the approximating polygonal curve) changes rapidly (equivalently, the points where the angle between the intersecting line segments is large). The angle between line segments is unambiguously defined since the line segments are directed. For example, if the curve follows a regular hexagon counterclockwise, the angle at each break point (i.e., vertex) is  $\pi/3$  radians.

We will call a break point p an extreme point if the angle at p exceeds a given value A. Note our use of the term "extreme point" is not the same as the usual standard use of the term, namely, the point where the curve has local maximum or minimum.

The angle  $\theta$  at q between the line segments [p,q] and [q,r] is computed by

$$\theta = \arccos \frac{\langle q-p,r-q\rangle}{||q-p||||r-q||}, \quad 0 < \theta < \frac{\pi}{2}$$



Fig. 6. Extreme points on a polygonal curve. At three vertices (0,2), (0,0), and (2,1), the direction of the curve changes more than at (1,2) or (1,0). The curvature at the three points is of larger value.

*Example 13:* Let C be a closed polygonal curve connecting (0, 0), (1, 0), (2, 1), (1, 2), and (0, 2). The angle measures at (0, 0), (2, 1), and (0, 2) are  $\pi/2$  and at (1, 0) and (1, 2) are  $\pi/4$ . For example, at (2, 1), the angle is computed by

$$\arccos\frac{\langle (1,1), (-1,1)\rangle}{\sqrt{2}\sqrt{2}} = \arccos 0 = \pi/2$$

If we take  $A = \pi/4$ , then (0, 0), (2, 1), and (0, 2) are extreme points.

*Example 14:* Let C be a polygonal curve connecting (0, 0), (3, 0), (3, 2), (1, 1), and (0, 0). If the preset value A is  $\pi/2$ , then (1, 1) is not an extreme point and all the other vertices are. The angle at (1, 1) measures

$$\arccos\frac{\langle (-2,-1),(-1,-1)\rangle}{\sqrt{10}} = \arccos\frac{3}{\sqrt{10}} < \frac{\pi}{2}$$

and the angles at (0, 0) measure  $3\pi/4$ .

*Example 15:* Let C be the standard regular hexagon with vertices at  $e^{\frac{1}{3}\pi i}$ , i = 0, 1, ..., 5. If we set the value  $A < \pi/3$ , then each vertex is an extreme point because the angle at each vertex is  $\pi/3$ .

# D. Estimating Velocity Vectors from the Approximation of Region Boundaries

Suppose a region in the image sequence is identified, its boundary curves are approximated by piecewise linear curves, and extreme points on each piecewise linear approximation of the boundary curves are found. We are assuming that the features in the oceanographic images are such that the extreme points of one image move to the extreme points of the next image in the sequence. Under this assumption, we can track extreme points and the boundary curve and eventually every point inside the region by interpolation based on the simplicial approximation of the region.

Let C be a boundary curve in an image and C' be the corresponding boundary curve in the next image of the sequence. Let  $[q_0, q_1, \ldots, q_k]$  be a piecewise linear approximation of C and  $[q'_0, q'_1, \ldots, q'_k]$  be a piecewise linear approximation of C'. Suppose  $\{e_1, \ldots, e_m\}$  is the set of all of the extreme points of the curve C and  $\{e'_1, \ldots, e'_m\}$  is the set of all extreme points of the corresponding curve C'. Note that the sequence  $\{e_1, \ldots, e_m\}$  of extreme points (respectively,  $\{e'_1, \ldots, e'_m\}$ ) is a subsequence of  $\{q_0, q_1, \ldots, q_k\}$ , the sequence of all break points of C (respectively,  $\{q'_0, q'_1, \ldots, q'_l\}$ , the sequence of all break points of C').

We are assuming the number of extreme points on two curves are the same. This can be done by adjusting the preset value A (respectively, A'), which determines the extreme points on C (respectively, C'). Since we assume the flow of the image is such that the extreme point  $e_i$  on C corresponds to  $e'_i$ on C', we approximate the velocity vector at  $e_i$ , representing the movement of the curve at  $e_i$  by taking the differences  $e'_i - e_i$ . We can extend this approximation of the velocity vector to generate a velocity vector field on the polygonal approximation  $[q_0, q_1, \ldots, q_k]$  of C by affine extension, as follows.

Let p = f(t) be a point on the polygonal curve  $[q_0, q_1, \ldots, q_k]$  such that p is between the extreme points  $e_i = f(s_i)$  and  $e_{i+1} = f(s_{i+1})$ . Let the corresponding extreme points be  $e'_i = f'(s'_i)$  and  $e'_{i+1} = f'(s'_{i+1})$ . Then we assign the vector f'(t') - f(t) to the point p, where

$$t' = s'_i + \frac{s'_{i+1} - s'_i}{s_{i+1} - s_i}(t - s_i).$$

This is an obvious extension to the velocity vector field on the polygonal approximation of the curve C. We will denote the vector field on C by V. If the curve C is closed so that it bounds a region, we can extend V to a velocity field, also denoted by V, on the simplicial approximation of the region by extending V on each simplex  $[p_i, p_{i+1}, p_{i+2}]$  in the approximation as

$$V(f(p)) = tV(f(p_i)) + sV(f(p_{i+1}) + rV(f(p_{i+2})))$$

where t,s,r are real numbers  $0 \leq s,t,r \leq 1$  such that s+t+r=1 and

$$p = tp_i + sp_{i+1} + rp_{i+2}$$
.

*E.* Algorithm for the Construction of the Boundary Curve and the Velocity Vector Field

- 1) Input: A sequence of digital binary images  $I_k(x,y)$  for  $0 \le x, y \le n$ . Each image is an  $(n+1) \times (n+1)$  matrix with entries zero or one.
- 2) Apply the edge detection algorithm to each image  $I_k(i, j)$  to identify the boundary of a region we are interested in. The boundary does not have to be closed, that is, the region need not be bounded.
  - a) We assume the boundary is a Jordan curve. That is, as a discrete image B(i, j), it satisfies the following condition. (This condition is that the boundary is a Jordan curve, with respect to eight-neighbors topology.) We will use eight-neighbors topology, though this can also be done with four-neighbors topology.
  - b) If B(i,j) = 1, then B(x,y) = 1 at exactly two eight-neighbors (x,y) of (i,j).
  - c) The eight-neighbors of (i, j) are elements of the eight-neighborhood

$$\{(i \pm 1, j), (i, j \pm 1), (i \pm 1, j - 1), (i \pm 1, j + 1)\}$$

which equals

$$\{(a,b) \mid 0 < d((a,b), (i,j)) \le 1\}$$

where d is the metric given by

$$d((a,b), (i,j)) = \max(|a-i|, |b-j|)$$

d) The four-neighbors of (i, j) are elements of the fourneighborhood,  $\{(i \pm 1, j), (i, j \pm 1)\}$ , which equals

$$\{(a,b) \mid 0 < d((a,b), (i,j)) \le 1\}$$

where d is the metric given by

$$d((a,b), (i,j)) = |a-i| + |b-j|.$$

 On each image in the sequence, construct a sequence of points

$$A = \{ (i(c), j(c)) \mid 0 \le c \le C \}$$

where B(i(c), j(c)) = 1 and B(i, j) = 0 elsewhere. A represents a sequence of the positions (i, j) at which B has value one, that is, where the boundary curve traces.

- a) We find the sequence A inductively. Suppose the points  $(i(0), j(0)), \ldots, (i(k), j(k))$  have been found. Then (i(k+1), j(k+1)) is the point in the eight-nbd of (i(k), j(k)) such that B(i(k+1), j(k+1)) = 1 and  $(i(k+1), j(k+1)) \neq (i(c), j(c))$  for any  $0 < c \leq k$ . This is well defined because of the conditions in step 2). We need to specify first two points (i(0), j(0)) and (i(1), j(1)) to start the inductive steps. The sequence A is a Jordan curve, with respect to the eight-neighbors topology.
- Let A<sub>k</sub> = {(i(c), j(c)) | 0 ≤ c ≤ C}, the set of points describing the boundary of the region in the kth image in the sequence.
- 5) To simplify the notation, let the piecewise linear approximation of each  $A_k$  also be denoted by  $A_k$ . The velocity vector field is defined on each  $A_k$ .
- 6) To find the velocity vector at the points not on the boundaries, we can apply the affine interpolation. In practice, however, it is not necessary if the distance between the boundaries in the image sequence are small. Then we will have a fine grid of points on the boundaries over the whole region.

The construction step 3) requires that the boundary we found must be a Jordan curve [that is, satisfying all the conditions in step 2)]. If there is any point where the curve is slightly thick or has jump-disconnection [violating the conditions in step 2)], the step cannot proceed. We can still generate a sequence of data points representing the boundary when it is not a Jordan curve, however, the process will require more complex algorithms. We may relax the condition of Jordan curve by mixing four-neighborhood and eight-neighborhood, but the algorithm will be a little more complicated.

For example, we then allow

while the image will have to be

or

or

with respect to eight-neighbors topology. Similarly, we allow

$$\begin{pmatrix} * & & \\ * & * & * \\ & & * \end{pmatrix}$$

$$\begin{pmatrix} * & & \\ * & * & \\ & * & * \end{pmatrix}$$

and so on, if using four-neighbors topology. Here \* represents a one and the blank spaces are zeroes in the matrix.

*Example 16:* Let the boundary of the region in the first image be



and the next one in the image sequence would be



Since we index the points as entries in a matrix, (i, j) corresponding to the point (x, y) in the usual xy-plane coordinate system (with its origin at the top left) by x = j and y = -i, we have the velocity vector at every point on the boundary in the first image (namely, the diagonal). For example, the velocity vector at (2, 2) (the third point from the left top corner) is (1,2)-(2,2) = (-1,0), which should be interpreted as a unit vector in the direction of north and at (7, 7) (the last point at the right bottom corner) is (5,7)-(7,7) = (-2,0), which should be interpreted as a vector with length two pointing north.



Fig. 7. (a) The input images were selected over the  $81 \times 108$  rectangular window shown on the original image of Fig. 1. (b) The images were segmented using an interactive threshold operator followed by a median filter to smooth out the noise. (c) The edge and boundary map of the detected segments obtained through a morphological operator. The box on the left image indicates a curve segment used later in Fig. 8.

#### F. Simulation Results

The method developed in this paper was applied to a sequence of six AVHRR images of the Gulf Stream of the north Atlantic. These images were acquired by an AVHRR IR sensor onboard the NOAA-11 satellite. The image shown in Fig. 1 highlights a region of  $81 \times 108$  pixels that has been considered for our experimental verification procedure. The area is large enough to cover the dynamic features over the six image frames. In order to reduce the computation time, the images were cropped to a minimal size. Three such cropped images are shown in Fig. 7(a). Henceforth, we refer to these small images as the image data.

The images were first segmented using a simple thresholding operation, and the results are shown in Fig. 7(b). A morphological filter was applied to each segmented image to extract the boundaries and streaks. The extracted boundary images are shown in 7(c). Each set of points constituting an area of interest is considered for simplicial decomposition. These simplexes are then analyzed to compute the velocity field on the connected region as a whole.

The above segmentation procedure is somewhat simple. However, the performance is sufficient to illustrate how our algorithm will work on extracting and tracking the extreme points drawn from segmented oceanographic images. We used a subjective approach to try various values of  $\epsilon$  and to arrive at a number of good break points. For fully robust and automated identification of break points, advanced segmentation techniques, such as Canny [15], are available.

The following break points were obtained after applying the algorithm to a cropped image of the Gulf Stream. The value for  $\epsilon$  was taken to be 0.86. This was chosen after experimenting with various values of  $\epsilon$ . The break points obtained from the first image and the second image are plotted in Fig. 8, which are also highlighted in Fig. 7(c).

Let A(q) stand for the angle measure at the break point q between the line segments [p,q] and [q,r] Choosing the break points with angle  $A(q) \ge 20$  for image 1 and those with  $A(q) \ge 11$  for image 2, we get the extreme points for images 1 and 2. The extreme points are plotted in Fig. 8.

The velocity vector at  $q_i$  in image 1 is evaluated by the difference  $q'_i - q_i$ , where  $q'_i$  is the corresponding extreme point on image 2. The velocity vectors at the extreme points in image 1 are plotted in Fig. 9.



Fig. 8. Two piecewise linear curves extracted from the first two images [Fig. 7(a)]. These curves are drawn identifying the break points in the segmented boundary maps and joining the consecutive break points by a straight line. A small portion of the actual boundary is shown on an enlarged scale.



Fig. 9. Initial velocity field computed from the two piecewise linear curves. The vector values illustrated here emanate from the break points of the first image and end at the corresponding break point of the second image. This field must then be interpolated using the affine function model of smooth velocity fields, as developed in our paper.

#### IV. DISCUSSION

We have presented a comprehensive method of approximating the boundaries and connected components of regions in oceanographic images using the simplicial approximation method. The simplicial approximation of domains (boundary curves and the connected components of the regions) are then used to find simplicial approximations of mappings between the corresponding objects (curves and connected components) in the image sequence. The mappings between the corresponding objects are then interpreted to estimate the velocity field of the features in the image sequence. We assume the velocity field of features in the oceanographic images can be approximated by tracking the boundary curves of connected components and then interpolating the velocity inside the region by the simplicial approximation technique. When tracking the boundary curves, we identify local extreme points from each boundary curve and assume the local extreme points in one image match to the local extreme points in the next image in the sequence. The weakness of this assumption is that the extreme points sometimes merge or disappear and new extreme points emerge during the whole period of the image sequence. In these cases, we need to determine the correspondence between the local extreme points by other criteria (varying the threshold, preserving the lengths of the boundary curves).

#### V. CONCLUSION

A well-established mathematical tool of simplicial approximation of regions on a plane (or on a surface) and simplicial approximation of continuous mappings between regions were applied to locate and approximate connected regions representing the Gulf Stream and its associated eddies. Once the regions of interest were approximated by simplices (called triangulation), vertices of the simplices in the approximation were identified and used to evaluate the velocity vector field of the features in the image sequence. We need to identify the corresponding points in each image in the image sequence to be able to estimate the velocity vector field. This is done by locating extreme points on the boundary curves of the region in the image sequence by assuming that extreme points on one image move to extreme points on the following image in the sequence. From these estimates of the velocity vectors, we interpolate the velocity field on the whole region by an affine approximation.

The future scope of the present work is as follows. The method described here assumes the interpolatable nature of the velocity field. This is true for densely sampled ocean image sequences and always true for images sequences of rigid objects. A composite three-dimensional (3-D) motion of rigid planar patch produces a velocity field that is fully described by a single affine transform [16]. Then, it is implicit that the interpolated 2-D velocity of each pixel in a two-simplex is consistent with the actual and perceived velocity. The same does not hold true when the object being tracked is a composite of two or more planar patches, even if it undergoes a rigid motion, or a simple patch undergoing a nonrigid motion.

Errors in the triangulation process could produce a twosimplex whose three vertices come from different connected components of velocity fields. Any additional information, such as coarse segmentation of the images, must be used to avoid such instances. One approach to overcoming this difficulty is to incorporate intensity-based optical flow computation of the local velocity for the pixels covered by such triangles. Better yet, is a combination of the two-simplex and the classic optical flow computation. Such an approach is being

#### ACKNOWLEDGMENT

The authors would like to thank the reviewers of this paper, whose feedback has helped improve the readability. Implementation of the algorithms in this paper and testing of the implemented program on sample satellite infrared images were done by K. Simhadri and V. Veeranna at the Robotics Research Laboratory, Louisiana State University.

#### REFERENCES

- I. Babuska and H. Oh, "The *p*-version of the finite element method for domains with corners and for infinite domains," *Num. Meth. PDES*, vol. 6, pp. 371–392, June 1990.
- [2] J. Dieudonné, Foundations of Modern Analysis. New York: Academic, 1969, p. 160.
- [3] W. Emery, A. Thomas, M. Collins, W. Crawford, and D. Mackas, "An objective method for computing advective surface velocities from sequential infrared satellite images," *J. Geophys. Res.*, vol. 91, no. 12, pp. 12865–12878, 1986.
- [4] C. Garcia and I. Robinson, "Sea surface velocities in shallow seas extracted from sequential coastal zone scanner satellite data," J. Geophys. Res., vol. 94, no. C9, p. 12 681, Sept. 1989.
- [5] R. J. Holyer and S. H. Peckinpaugh, "Edge detection applied to satellite imagery of the oceans," *IEEE Trans. Geosci. Remote Sensing*, vol. 27, pp. 45–56, Jan. 1989.
- [6] K. Kelly, "An inverse model for near-surface velocity from infrared images," *J. Phys. Ocean.*, vol. 19, pp. 1845–1864, Dec. 1989.
   [7] , "Comparison of velocity estimates from advanced very high
- [7] \_\_\_\_\_, "Comparison of velocity estimates from advanced very high resolution radiometer in the coastal transition zone," J. Geophys. Res., vol. 97, no. c6, pp. 9653–9668, 1992.
- [8] S. Krishnamurthy, S. S. Iyengar, R. Holyer, and M. Lybanon, "Histogram-based morphological edge detection," *IEEE Trans. Geosci. Remote Sensing*, vol. 32, pp. 759–767, July 1994.
- [9] W. Press et al., Numerical Recipes in C. Cambridge, U.K.: Cambridge Univ. Press, 1990, p. 643.
- [10] J. J. Rotman, An Introduction to Algebraic Topology. New York: Springer-Verlag, 1988, pp. 31–38 and 136–138.
- [11] H. Stommel, *The Gulf Stream: A Physical and Dynamical Description*, 2nd ed. Berkeley, CA: Univ. of California Press, 1965.
- [12] E. Sethi and R. Jain, "Finding trajectories of feature points in a monocular image sequence," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 535–552, Jan. 1987.
- [13] A. Vastano and S. Borders, "Sea Surface motion over an anticyclonic eddy on the Oyashio front," *Remote Sens. Environ.*, vol. 16. no. 1, pp. 87–90, 1984.
- [14] A. Vastano and R. Reid, "Sea surface topography estimation with infrared satellite imagery," J. Atmos. Ocean. Technol., vol. 2, pp. 393–400, Sept. 1985.
- [15] J. F. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, June 1986.
- [16] G. Seetharaman, "Image sequence analysis for three dimensional perception of dynamic scenes," in *Handbook of Pattern Recognition and Image Processing: Computer Vision*, vol. 2, T. Y. Young, Ed. New York: Academic, 1994, ch. 10.
- [17] J. Zachary and S. S. Iyengar, "An affine approach to motion field estimation in oceanographic image sequences," submitted for publication.

**E. C. Cho** received the undergraduate degree from Seoul National University, Seoul, Korea, and the Ph.D. degree in mathematics from Rutgers University, New Brunswick, NJ.

He is currently an Associate Professor of Mathematics, Kentucky State University, Frankfort. His present research interest centers around algorithms for vision and image processing applications. **S. S. Iyengar** (M'88–SM'89–F'95), for a photograph and biography, see this issue, p. 778.

Ronald J. Holyer, for a photograph and biography, see this issue, p. 778.

Matthew Lybanon, for a photograph and biography, see this issue, p. 778.



**Guna Seetharaman** (S'82–M'88) received the B.E. degree in electronics and communication engineering from University of Madras, Madras, India, the M. Tech. degree in electrical engineering from Indian Institute of Technology, Madras, and the Ph.D. degree in electrical and computer engineering from the University of Miami, Coral Gables, FL, in 1980, 1982, and 1988, respectively. He has been with The Center for Advanced

Computer Studies, University of Southwestern Louisiana, Lafayette, since 1988, where he is

currently an Associate Professor of Computer Engineering. He has been the Director of the Computer Vision and Pattern Recognition Laboratory, University of Southwestern Louisiana, which he established in 1988. His research focus has been in the areas of three-dimensional displays and three-dimensional motion analysis. He has published several articles and book chapters in these areas. His research and the Computer Vision and Pattern Recognition Laboratory have been supported by Louisiana Educational Quality Support Funds and The National Science Foundation.

Dr. Seetharaman is a recipient of a Eliahu & Joyce Jury Award for outstanding graduate scholarship in 1986 and The National Science Research Initiation Award from 1992 to 1996. He is a member of Tau Beta Pi, Eta Kappa Nu, and Pi Epsilon. He coorganized The International Workshop on Data Fusion, Washington, DC, held in August 1996.