# Cloud Tracking by Scale Space Classification

Dipti Prasad Mukherjee and Scott T. Acton, *Senior Member, IEEE*

***Abstract*—The problem of cloud tracking within a sequence of geo-stationary satellite images has direct relevance to the analysis of cloud life cycles and to the detection of cloud motion vectors (CMVs). The proposed approach first identifies a homogeneous consistent cloud mass for tracking and then establishes motion correspondence within an image sequence. In contrast to the cross-correlation based approach as adopted in automatic CMV detection analysis, a scale space classifier is designed to detect cloud mass in the source image taken at time t and the destination image at time $t + \delta t$. Boundaries of the extracted cloud segments are matched by computing a correspondence between high curvature points. This shape based method is capable of tracking in the cases of rotation, scaling, and shearing, while the correlation technique is limited to translational motion. The final tracking results provide motion magnitude and direction for each contour point, allowing reliable estimation of meteorological events and wind velocities aloft. With comparable computational expense, the scale space classification technique exceeds the performance of the traditional correlation-based approach in terms of reduced localization error and false matches.**

***Index Terms*—Cloud tracking, image classification, motion correspondence.**

## I. Introduction

SPATIO–TEMPORAL analysis of meteorological events is an important part of routine numerical weather analysis. In that context, a cloud tracking method is presented here for a sequence of geostationary satellite images. Given a pair of remotely sensed images, captured at a fixed time interval (typically, 30 min), the objective is to derive motion vectors associated with the cloud mass. This correspondence process is a useful precursor to cloud motion vector (CMV) studies and spatio–temporal analysis of cloud life cycles. The spatio–temporal life cycle includes the generation, dissipation and assimilation of clouds that can be observed in a sequence of geostationary satellite images.

Cloud tracking is an example of motion analysis of deformable shapes in a sequence of monocular images. In the absence of any meteorological disturbance, an assumption of path and shape coherence of the cloud mass between subsequent image frames is tenable given a relatively short (<1 h) duration of observation.

The tracking of clouds involves two major subproblems. First, a homogeneous cloud mass needs to be detected for tracking. The detection of the cloud mass boundary is essentially a problem of image segmentation, and a scale-sensitive image classification approach is introduced in this paper to achieve this objective. The classification is performed on a scale space representation of the satellite imagery generated using area-based morphological operators. This classifier detects homogeneous cloud segments with minimum intra-segment classification error. The scale space approach gives improved segment integrity over the fixed-scale approach.

The second subproblem is the evaluation of motion vectors. In this case, feature points are specified on the boundary of an extracted cloud segment. These feature points are points of inflection on the contour that remain *almost* stable (maintain curvature properties) over a brief period of observation. A cost function that enforces both path and shape coherence properties of these feature points is minimized in order to establish correspondences between feature points.

The organization of the paper is as follows. In the next section we review the related work in the literature. This is followed by a description of the image segmentation technique in Section III. Section IV presents the process of generating motion vectors. In Section V, results of cloud tracking from the geostationary satellite images are provided, followed by concluding remarks.

## II. Background

Cloud classification techniques have attracted considerable attention. A review of such schemes is provided in [18]. A multispectral classification of meteorological satellite images using a pixel-clustering algorithm is described in [19]. A number of cloud segmentation approaches have used pixel-wise neural-network-based classifiers that exploit textural and spectral features [13], [20], [25], [30]. In contrast, we have developed an area-based, shape-preserving morphological segmentation scheme for cloud extraction. Since our primary objective is cloud tracking, we are interested in detecting a homogeneous cloud mass which remains *almost* consistent during the period of observation and then extracting the cloud segment boundary. Using a pixelwise classification with a predetermined set of cloud classes results in a number of extraneous segments (minor regions) that lack significance for cloud tracking. Such details also increase the occurrence of intrasegment classification error. The proposed area morphological operators generate a *scale space*. The scale space contains a coarse-to-fine collection of image representations, for the input imagery, where the scale is defined as a function of the area of *connected components* within image *level sets*. The conventional classification schemes are adapted to classify the scale space of the remotely sensed imagery. In Section III, a detailed description of the segmentation process is given.

Cloud tracking is of interest to meteorologists particularly in the estimation of wind velocity. Furthermore, tracking could provide valuable information with respect to growth, dissipation and disintegration of cloud masses that transport pollu-

Manuscript received June 2, 1999; revised November 27, 2001.

D. P. Mukherjee is with the Electronics and Communication Sciences Unit, Indian Statistical Institute, Calcutta, India 700035 (e-mail: dipti@isical.ac.in).

S. T. Acton is with the Department of Electrical Engineering, University of Virginia, Charlottesville, VA 22904 USA (e-mail: acton@virginia.edu).

Publisher Item Identifier S 0196-2892(02)01875-2.

tants. The complexity of the cloud tracking problem stems from the fact that clouds are deformable in nature, and hence the principles of rigid motion cannot be utilized naively [27]. The most common technique for tracking meteorological structures is the cross-correlation-based approach. For example, in [8], cross-correlation values at the inflection points of cloud contour were used in cloud tracking.

The maximum cross-correlation measure and template matching using a Euclidean measure are, by and large, adopted by different operational agencies for CMV analysis [9], [10]. The major motivation for using correlation is computational simplicity. In the cases of bias and normalization of intensity profiles, the Euclidean distance based matching of source and target regions may yield better results. A detailed qualitative and quantitative comparison of these two methods is given in [7]. Fully automated cloud drift wind estimation is reported by NESDIS [17]. The NESDIS operational system includes tracer selection, height assignment, tracking, and a quality control step. Selection of the tracer involves estimation of local intensity gradient and a spatial coherence analysis in the target image. The coherent cloud mass is then assigned a height based on the numerical forecast model of the vertical atmospheric temperature profile. This height assignment in turn assigns a numerical forecast of the wind to predict the position of the search domain in the destination image for the particular tracer under consideration. A correspondence for tracking is then established by a cross-correlation technique. A broad summary of the status and development of operational GOES wind products is presented in [6].

The major limitation of the correlation approach is that the correlation surface between the image sequence is typically multimodal (due to nonuniqueness of internal intensity features within the cloud mass) and thus difficult to interpret. Also, the cloud mass under observation typically undergoes not only translation but also rotation and scaling (and possibly shearing), which cannot be accommodated by basic correlation techniques. In light of these drawbacks, [29] has proposed a combination of correlation and relaxation labeling to improve the result of correlation based CMV analysis. Our solution focuses on the tracking of cloud boundary features, instead of correlating unreliable internal intensity features.

The fully automated NESDIS CMV detection system requires a number of parameters such as a threshold for estimating the coherency of the cloud mass, parameters for assigning cloud top temperature, and a threshold for selecting maximum cross-correlation measure. Most of the parameters are empirically determined and tested for an exhaustive set of conditions. However, no attempt has been made to incorporate cloud shape information in establishing a correspondence between the source and destination images.

Reference [4] has modeled the atmospheric disturbances with geometric shapes such as logarithmic spirals. A shape based tracking method is then applied, which depends on the presence of specific geometric shapes in the satellite image. Newland *et al.* have used definition of a fuzzy object to extract homogeneous cloud masses and to relate closely spaced CMV for improved correlation based CMV results [16]. Reference [5] has defined a cost path for tracking using the minimal geodesic norm between the contours of cloud observed at 30–min intervals. However, they have indicated that geometric properties such as curvature information in a small neighborhood of the contour should be incorporated in the cost function. In the proposed approach, points of inflection on the contour are used for cloud tracking. It is assumed that the amount of deformation is marginal at or close to these high curvature points. The proposed cost function measures the path and shape coherence of these feature points in an image sequence. As opposed to conventional tracking approaches [22], the correspondence between feature points of image sequence is limited by an ordering constraint and is established by generating an interpolated contour. The correspondence algorithm is detailed in Section IV.

## III. Cloud Segmentation

In our point-based approach, cloud tracking is implemented on the segmented boundaries of the cloud masses. Cloud segmentation is achieved through *scale space classification*, where a set of scaled images (a scale space) is constructed using shape preserving area morphological operators. In the next section, some basic definitions and properties of this scale space are discussed. This discussion is followed by the introduction of scale space classification approach. To achieve a classification within the scale space, a clustering technique must be adapted specifically for the scale space. The fuzzy c-means (FCM) clustering approach is used to achieve this objective. The proposed classification technique is defined as scale space fuzzy c-means (SSFCM) clustering. Finally the computational complexity of the proposed segmentation approach is discussed.

### *A. Area Open–Close Scale Space*

Scale is an important feature in cloud tracking. At the finest scale, pixel-wise methods can be adversely affected by noise and insignificant internal cloud features. We choose area morphological operators to provide scaled versions of the remotely sensed imagery. The area morphological operators can provide scaled image representations (that constitute a scale space) without distorting the region boundaries. Since our cloud tracking method operates on the region boundaries, the area morphological operators are well matched to our tracking technique.

The area morphological operators are founded on *level set* theory. In a threshold decomposition of the image $\mathbf{I}$, an associated level set $L(\mathbf{I}, t)$ is a set of pixels that meet a given threshold $t$: $(x, y) \in L(\mathbf{I}, t)$ if $I(x, y) \geq t$. For a discrete range of K intensities, an image $\mathbf{I}$ can be decomposed into K level sets $L(\mathbf{I}, t)$ where

$$I(x, y) = \max[t: (x, y) \in L(\mathbf{I}, t)]. \tag{1}$$

Typically, for an eight-bit gray level image, K$= 256$. Therefore, such an image will have 256 different level sets. Each such level set consists of connected components of "on" pixels (pixel value $=$ 1) where $I(x, y) \geq t$ and "off" pixels (pixel value $=$ 0) where $I(x, y) < t$. This representation forms a three-dimensional (3-D) stack of 256 level set (binary) matrices. The intensity of the image at position $(x, y)$ may be defined by the highest (in terms of $t$) level for which $(x, y)$ is a member, as depicted in (1). With the area morphological filters, we essentially process

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 225 | 225 | 224 | 225 | 225 | 226 | 226 | 225 | 226 | 228 |
| 225 | 225 | 223 | 224 | 225 | 226 | 227 | 226 | 226 | 229 |
| 224 | 223 | 224 | 224 | 225 | 227 | 227 | 226 | 228 | 231 |
| 223 | 220 | 217 | 220 | 224 | 225 | 227 | 227 | 228 | 233 |
| 208 | 186 | 173 | 187 | 211 | 222 | 227 | 228 | 230 | 234 |
| 141 | 104 | 80 | 100 | 155 | 209 | 219 | 225 | 230 | 235 |
| 65 | 44 | 38 | 44 | 77 | 158 | 200 | 215 | 228 | 237 |
| 37 | 26 | 25 | 25 | 29 | 78 | 172 | 205 | 222 | 237 |
| 25 | 25 | 25 | 30 | 38 | 50 | 118 | 201 | 217 | 234 |
| 25 | 25 | 25 | 64 | 73 | 61 | 76 | 174 | 211 | 228 |

(a)

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

(b)

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 224 | 224 | 224 | 225 | 225 | 226 | 226 | 226 | 226 | 226 |
| 224 | 224 | 223 | 224 | 225 | 226 | 226 | 226 | 226 | 226 |
| 224 | 223 | 224 | 224 | 225 | 226 | 226 | 226 | 226 | 226 |
| 223 | 220 | 217 | 220 | 224 | 225 | 226 | 226 | 226 | 226 |
| 208 | 186 | 173 | 187 | 211 | 222 | 226 | 226 | 226 | 226 |
| 141 | 104 | 80 | 100 | 155 | 209 | 219 | 225 | 226 | 226 |
| 80 | 80 | 80 | 80 | 80 | 158 | 200 | 215 | 226 | 226 |
| 80 | 80 | 80 | 80 | 80 | 80 | 172 | 205 | 222 | 226 |
| 80 | 80 | 80 | 80 | 80 | 80 | 118 | 201 | 217 | 226 |
| 80 | 80 | 80 | 80 | 80 | 80 | 80 | 174 | 211 | 226 |

(c)

Fig. 1. (a) Pixel values within a 10 × 10 image. (b) Level set $L(\mathbf{I}, 225)$ for image in Fig. 1(a) (=1 for intensities ≥225, 0 otherwise). (c) After AOC of image of Fig. 1(a) at scale s = $25$.

each level set independently as a binary image. Then, using (1), we can define a grayscale output image.

With respect to a given level set $L(\mathbf{I}, t)$, an area open operator, denoted by $\overset{s}{\circ}(\mathbf{I})$, removes all connected components within the level set that do not have a minimum area of s. Here, removal signifies replacement of the connected component with pixel value zero, if one views a level set as a binary (0/1 valued) image. An area close operator, denoted by $\overset{s}{\bullet}(\mathbf{I})$, removes connected components in the complemented level set $L^c(\mathbf{I}, t)$ that do not possess the minimum area s. Applied to each level set in the image (for each intensity in the range of possible intensities), area open removes small bright objects, while area close removes small dark objects in an image [21]. The area open-close (AOC) operator, denoted by $\overset{s}{\bullet}(\overset{s}{\circ}(\mathbf{I}))$, is defined as the concatenation of the area open and close operators. Thus, the AOC operator is able to control the scale of both positive-going bright objects and negative-going dark objects.

The AOC operation is illustrated using a small 10 × 10 matrix, shown in Fig. 1(a). An example level set $L(\mathbf{I}, 225)$ of the image of Fig. 1(a), at intensity value greater or equal to 225, is shown in Fig. 1(b). To process this level set at scale s = $25$, all the connected components (of ones) having area less than 25 are eliminated (replaced by zeros) in the area open operation. For example, the connected component of ones at the top left corner of the matrix shown in Fig. 1(b), with an area of four pixels, is replaced by zeros at scale s = $25$ due to area opening operation. Note that this region is part of a larger connected component in $L(\mathbf{I}, 224)$, with an area of 44, so the resultant intensity [shown in Fig. 1(c)] becomes 224 in the upper left corner.

Fig. 2. (a) Noisy synthetic image (100 × 100 pixels). (b) After AOC at s= $100$. (c) After AOC at s= $500$.

TABLE I
MEAN BRIGHTNESS (IN SCALE OF 0–255) OF CLOUD CLASSES

| Class | Fig. 5(a) | Fig. 5(b) | Fig. 5(c) | Fig. 14(a) | Fig. 14(b) | Fig. 14(c) |
|---|---|---|---|---|---|---|
| 1 | 51.9 | 52.8 | 53.3 | 54.0 | 54.1 | 54.6 |
| 2 | 69.3 | 75.4 | 78.3 | 64.1 | 63.0 | 61.2 |
| 3 | 113.1 | 113.3 | 111.3 | 77.6 | 73.6 | 72.5 |

The area close portion of the AOC operation is complementary in function to the area open. So, area close removes small (with area beneath $s$) connected components of zeros, replacing them with ones in the processed level set. This level set operation is repeated for all 256 level sets. Take, for example, the pixel located at row one, column eight of Fig. 1(a) with pixel value 225. Within the level set $L(\mathbf{I}, 226)$, this location has a value of zero while all its neighbors have value one. During the area close operation, the complement of level set $L(\mathbf{I}, 226)$ is taken when the pixel value at row one, column eight becomes unity-valued, while the neighbors change to zero. The length of the connected component at row one, column eight in the complement of $L(\mathbf{I}, 226)$ is less than scale s = $25$ and is removed (replaced by zero in the complement set). Hence this location in $L(\mathbf{I}, 226)$ is now changed to one instead of its original value zero. This forces the stray pixel at row one column eight to take the value of 226 when all 256 level sets are added together. Note that both area open and close operation encourages homogeneous region to grow by removing stray pixels and connected components less than the prespecified scale. Reconstructing via (1), the final AOC processed image for the input image in Fig. 1(a) is shown in Fig. 1(c).

Fig. 2(a) shows a noisy synthetic image. Results of the AOC operation on this image at scales 100 and 500 are shown in Fig. 2(b) and (c), respectively. These examples show the scaling and noise reduction capabilities of area morphology in qualitative terms.

The scale space $\{\mathbf{I}\}$ contains images $\mathbf{I}_s$ corresponding to a scale $s$. To guarantee that new features are not created with increased scale, the scaled representations are defined recursively

$$\mathbf{I}_{s+} = \overset{s+}{\bullet}(\overset{s+}{\circ}(\mathbf{I}_s)) \tag{2}$$

where $\mathbf{I}_{s+}$ is the next coarser scale representation in the scale space after $\mathbf{I}_s$. The initial scale representation $\mathbf{I}_0$ is the original input image.

The use of the AOC scale space for classification is motivated by several attractive properties. Excluding error due to discretization, the scaling method is invariant to translation and rotation. Hence, an input image rotated or translated should result in an equivalently rotated, translated scale space representation. The AOC scale space is also causal [14] in the sense that new regions are not introduced at coarser scales. If the scaling method introduced new objects (new connected components within an image level set), the scale space classification could produce false regions in the classification. Because our cloud tracking technique utilizes the cloud contours, the preservation of edge locations is also important. In contrast to linear techniques such as Gaussian filtering and nonlinear techniques such as traditional morphological filtering, the AOC-generated scale space preserves the location of *level lines* (connected component boundaries) through scale.

An illustration of these properties is provided in Fig. 2. The AOC processing results in Fig. 2(b) and (c) at scales 100 and 500 have removed the noise and objects that do not meet the specified minimum scale. This scaling is achieved without distorting the edges of the significant objects and without generating any new objects.

Now, we discuss the use of the AOC scale space in classifying the cloud imagery for tracking.

### *B. Scale Space Classification*

The idea behind using the scale space for classification is that the set of scaled representations allows objects of similar intensity *and* similar scale characteristics to be clustered together. It is important to note that this is not simply a matter of treating scale as an additional classification parameter. In fact, a pixel in the image may be a member of multiple objects of different scales. For cloud imagery, this property holds. We often view several layers of clouds and several cloud components simultaneously in the same remotely sensed image.

For a scale space $\{\mathbf{I}\}$, we inspect the intensity $I_s(x, y)$ at position $(x, y)$ and scale s through scale space. We call $\mathbf{I}(x, y)$ the scale space vector at $(x, y)$; it is a one-dimensional signal parameterized by scale s. The scale space classifier clusters pixels in the input imagery according to the similarity between scale space vectors.

Clustering, discussed in Section III-C, groups pixels in the image according to a distance from a cluster center (or cluster mean). The distance between scale space vector $\mathbf{I}(x, y)$ and a cluster mean $\mu(l)$ for class $l$ is given by

$$d(\mathbf{I}(x, y), \mu(l)) = \left[\sum_{\Omega_s} |I_s(x, y) - \mu_s(l)|^2\right]^{1/2} \tag{3}$$

where $\Omega_s$ is the set of scales in the scale space.

The scale space classifier improves upon the traditional fixed scale classifier in two ways. First, if two pixels were members of cloud components of different scales, but had similar intensities, the two pixels would be clustered in the same class for the fixed scale case. But, for the scale space classifier, the distance between these two points would increase in scale space, since one point would be altered before the other (as the scales of the cloud components are different). So, the scale space classifier avoids clustering pixels that are similar in intensity, but are not physically similar in the sense of region scale similarity.

The second case in which the scale space classifier excels concerns intra-object classification error. Given an object in the image, such as a cloud, the entire object should be classified into one class. If noise or subtle variation of intensity in the input image exists, then two pixels in the same object may be clustered into different classes for the fixed-scale classification approach. However, with the scale space classifier, the difference in intensity between the two pixels in the same object will decrease as scale is increased. For example, a fixed-scale classification of Fig. 2(a) would result in classifying the interior objects (inside the large objects) as separate objects. Using the AOC-scaled image shown in Fig. 2(c), the two distinct objects could be effectively extracted.

### *C. Clustering Algorithm*

For both the fixed-scale case and the scale space classifier, the pixels are grouped based on a similarity measure. Traditionally, both parametric and nonparametric classification schemes are used for remote sensing applications [23]. Due to the possibility of a single pixel belonging to different objects at various scales, we have adopted an unsupervised fuzzy c-means classifier [3]. The fuzzy $c$-means technique allows fuzzy membership to multiple classes for a single pixel.

The goal of the fuzzy $c$-means clustering algorithm may be stated as minimizing the squared distance between the pixel intensities (or scale space vectors) and their respective cluster centers. For a set of cluster centers $\mu$ and partition of the image $U$ into $C$ classes, we seek to minimize

$$J_m(U, \mu) = \sum_{\Omega} \sum_{i=1}^{C} (u_i(x, y))^2 ||d_i(x, y)||^2. \tag{4}$$

For the scale space vector $\mathbf{I}(x, y)$ at location $(x, y)$, the measure $||d_i(x, y)|| = ||\mathbf{I}(x, y) - \mu_i||$ is the distance between the scale space vector and the $i$th cluster center $\mu_i$. In the fixed scale case, this distance is computed between the intensity $I(x, y)$ and the cluster center $\mu_i$. The squared distances are weighted by the fuzzy membership value of each scale space vector $u_i(x, y)$ for the ith class. For every scale space vector $\mathbf{I}(x, y)$, the error criterion $J_m(U, \mu)$ is minimized subject to the conditions $\sum_{i=1}^{l} u_i(x, y) = 1,\ 0 < \sum_{\Omega} u_i(x, y) < |\Omega|$ and $u_i(x, y) \geq 0$, where $\Omega$ is the domain of the image.

Iterative updating of the cluster centers and membership values is implemented according the FCM guidelines in [3]. The update of the membership values and cluster centers is terminated when insignificant (1–2% of the current value) changes in $\mu_i$ are observed between consecutive iterations. Then, the fuzzy class membership is defuzzified by selecting the highest class membership for each image location.

The scale space classification approach improves upon the fixed-scale approach in terms of classification error and intra-object classification error. On five tracking image examples tested with accompanying ground truth data, the scale space approach provided between 10% and 20% decrease in overall classification error, compared to fixed scale fuzzy c-means classification. In terms of the intra-object classification error (instances of classifying an object into two or more classes), the scale space approach again provided improvements in the range of 10 to 20%.

Various approaches [12], [24] have been attempted for determining the number of potential classes. However, the straightforward intensity histogram based technique works well for cloud imagery. Because of the reasonable contrast of the cloud image, the number of peaks in the intensity histogram corresponds to the number of classes present in the image. The number of significant maxima in the image histogram automatically determines the number of classes [26] and can be used for operational purposes.

In summary, the AOC scale space allows a meaningful clustering that can be exploited in the case of cloud tracking. The scale space preserves properties that are important for cloud tracking, including the preservation and localization of contours that are used for cloud tracking. Furthermore, the AOC scale space does not introduce new segments (artifacts) with increased scale. In the next section, the method for cloud tracking, using the extracted segments, is detailed.

## IV. Cloud Tracking

The cloud tracking process involves two specific steps: shape extraction and point correspondence. First, we determine the areas of homogeneous cloud mass in the geostationary satellite images at time t and $t + \delta t$. After scale space classification is applied, we extract the segments with average intensities above the average intensity of the (cloudless) background. The cloudless background represents the pixel intensities corresponding to the landmass and water bodies as seen in a subjectively labeled cloud-free satellite image. For operational use in the case of an IR image, the cloud class with maximum brightness (consequently with minimum temperature) is selected for tracking. Table I gives the mean brightness for each class used in the example sequence.

A set of feature points on the contour of this segment yields the candidate points that are used to generate motion vectors. These feature points are points of inflection along the contour. The next step in cloud tracking involves establishing correspondence between these feature points of the images at time $t$ and $t+\delta t$. Let contours of cloud segments detected at time $t$ and $t+\delta t$ be labeled as *source* and *destination contours* and denoted by $C^s$ and $C^d$, respectively. Points of inflection are determined on $C^s$ and $C^d$ by fitting polygon chains and computing points of maximal curvature change. The point coordinates are stored in sets $\mathbf{L^s}$ and $\mathbf{L^d}$ after counter-clockwise scanning of the contours $C^s$ and $C^d$, respectively. These points of inflections are preserved even after projective image transformation [15], [28].

The problem of tracking is posed as a correspondence between $\mathbf{L^s}$ and $\mathbf{L^d}$. This correspondence is established by minimizing a cost functional based on disparities between $\mathbf{L^s}$ and $\mathbf{L^d}$. The selection of cost function is guided by following principles

1) The function should be nonnegative.
2) The function should impose an ordering constraint in case of potential matches between $\mathbf{L^s}$ and $\mathbf{L^d}$. For example, for a feature point on the source contour $l_i^s$, there are three potential matches on the destination contour: $l_{i-1}^d$, $l_i^d$, and $l_{i+1}^d$ (in a specific counter-clockwise order). Given a match between $l_i^s$ and $l_i^d$, $l_{i-1}^s$ is not allowed to match with $l_{i+1}^d$. Thus, the spatial ordering of the matched points is not violated.
3) The function should enforce both path and shape coherence between the contours. In the absence of any meteorological disturbances, the deformation is assumed to be smooth. The path coherence encourages potential matches based on a minimum distance measure. The shape coherence is measured as the minimum deformation between the destination contour and an interpolated contour based on potential matches between $\mathbf{L^s}$ and $\mathbf{L^d}$. Next we detail the implementation of the interpolated contour.

Given the above principles, the correspondence $(l^s, l^d)$ is determined by minimizing the following cost function:

$$C = \sum_{j=1}^{n_s} \sum_{i=1}^{n_o} d\left(l_j^s, l_i^d\right) + \lambda \sum_{j=1}^{n_s} \sum_{i=1}^{n_o} \delta a\left(l_j^s, l_i^d\right) \tag{5}$$

where $d(l^s, l^d)$ is the distance measure for a possible match, given $n_o$ possible matches for each of the $n_s$ points on the source contour. The parameter $\delta a(l^s, l^d)$ is the sum of distance between the interpolated contour and $C^d$ for a possible match between $(l^s, l^d)$. The weight parameter $\lambda$ can be computed using the cross-validation technique [2].

To minimize the cost function (5), a computationally expensive, combinatorial search strategy is avoided using an ordering constraint. The potential matches of $l^s$ are limited to immediate neighbors in $l^d$ (both in the clockwise and counter clockwise directions). The absolute minimum of $C$ for all such cases provides the best possible match.

To analyze the movement of contour points that are not inflection points, an interpolated contour is computed. The interpolated contour is generated using, following steps.

a) For $n_o$, potential correspondences between $(l^s, l^d)$, consider a unit mass particle that is accelerating from $l^s$ to $l^d$ with a zero initial velocity. The tension $T^s$ at $l^s$ could be evaluated from the knowledge of time of travel and the distance between $l^s$ and $l^d$. Therefore, for any two consecutive points of inflection on the source contour, $l_i^s$ and $l_{i+1}^s$, tensions $T_i^s$ and $T_{i+1}^s$ could be estimated.
b) For contour pixels between $l_i^s$ and $l_{i+1}^s$, the tension at every point could be estimated from a combination of tensions $T_i^s$ and $T_{i+1}^s$. The tension at a contour point $l_j^s$ lies between $l_i^s$ and $l_{i+1}^s$, $T_j^s$, and is computed using $T_j^s = \alpha T_i^s + (1-\alpha) T_{i+1}^s$, $0 \leq \alpha \leq 1$, where, the weight $\alpha$ is the normalized distance between contour points $l_j^s$ and $l_{i+1}^s$.
c) The interpolated contour is computed along the normal at each source contour point from the tension at every point $T_j^s$ and the time interval between the source and the destination contour.

So, minimization of the cost functional allows a correspondence to be established between inflection points. The correspondence of noninflection points is interpolated using a tension model. In Section V, results are given that demonstrate the combination of scale space classification and contour matching.

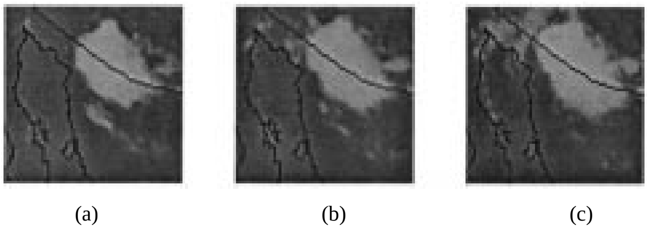


Fig. 3. (a)–(c) Sequence I: Cloud mass in GOES 8 IR image between 30–40 N latitude and 110–120 W longitude at 0215Z, 0245Z, and 0315Z on August 16th, 2000.

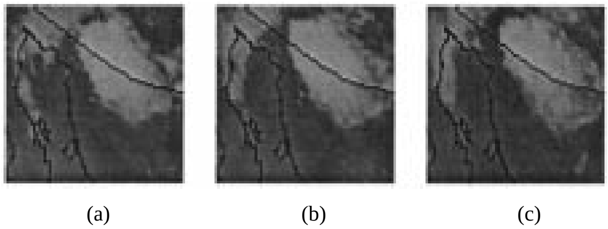


Fig. 4. (a)–(c) Subsequent frames from Sequence I acquired at 0345Z, 0415Z, and 0445Z.

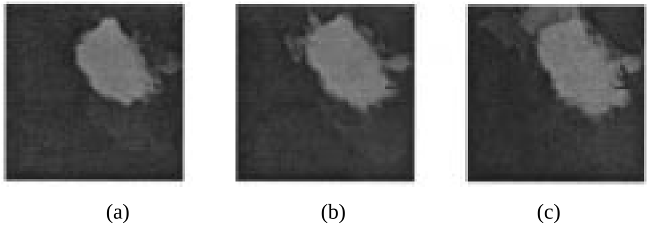


Fig. 5. (a)–(c) AOC scale space images of Fig. 3(a)–(c), respectively, at scale $s = 200$.

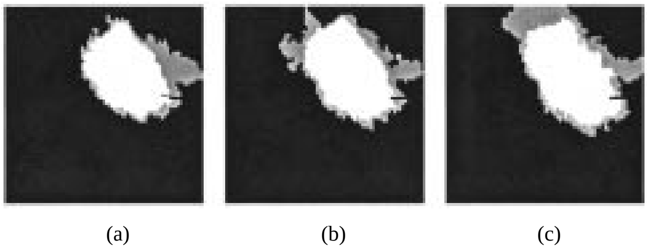


Fig. 6. (a)–(c) Three-class SSFCM results from Fig. 5.

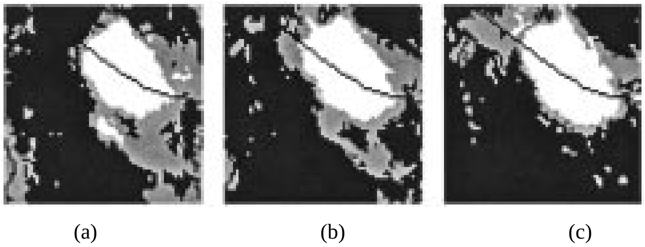


Fig. 7. (a)–(c) Three-class FCM results using Fig. 3 as input.

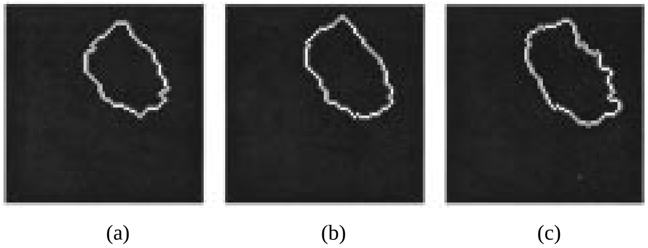


Fig. 8. (a)–(c) Cloud boundary extracted from Fig. 6.

## V. Results

Two GOES image triplet sequences are used to demonstrate and compare our cloud tracking results with the CMVs generated by the NOAA/NESDIS operational procedure. The data collected depict two distinct cloud movements. From each image triplet, we have generated two cloud tracking results for each 30 min interval. The efficacy of the proposed technique is demonstrated by checking the consistency between these two results produced by our algorithm. Also, results of our method are compared with the correlation based method results. Finally, the results are compared with NOAA/NESDIS high-density CMV results for the cloud patch under observation.

Sequence I, shown in Fig. 3(a)–(c), consists of subimages from three 30 min GOES 8 East Conus sector IR images acquired on August 16th, 2000 taken at 0215Z, 0245Z, and 0315Z, respectively. The resolution of the GOES 8 imagery is 8 km. For Sequence I, the clouds shown are of the cumulonimbus variety, under observation over a region subjected to tropical continental air circulation. This region is located from 40 °N latitude to just above the tropic of Cancer [11]. The vast dry areas of Texas, Mexico, New Mexico, and Arizona generate hot, dry, tropical continental air. This air mixes with tropical maritime airflow from the Pacific and generates a swirling wind flow pattern. The cloud tracking result developed using the point based method corroborates such development. This is further observed in the images of the same cloud mass taken beyond 0315Z h. Fig. 4(a)–(c) are the identical IR images of GOES 8 East Conus sector taken at 0345Z, 0415Z, and 0445Z, respectively. The main cloud mass shifts toward the north (up in the image) against polar continental cold currents due to the swirling action of tropical continental air during late summer.

For Sequence I, the corresponding AOC scale space images at scale $s = 200$ are shown in Fig. 5(a)–(c) with the corresponding SSFCM results in Fig. 6(a)–(c). As defined in Section III, the unit of scale $s$ defines the minimum number of connected pixels in an image level set.[1] In this case, $s = 500$ corresponds to 8000 km$^2$. Note that these images are classified into three classes. As stated earlier, the cluster of pixels with maximum intensity corresponding to minimum temperature is considered as the cloud mass to be tracked in this sequence of images. Notice the improvement of image segmentation result compared to the output obtained from straightforward implementation of FCM image classification technique as shown in Fig. 7.

The boundaries used for tracking in the point based method are shown in Fig. 8, while the vectors representing cloud motion are depicted in Fig. 9. For visualization, direction vectors of uniform length (five-pixels in this case) are drawn starting at every point of inflection on the source contour and pointing toward corresponding point of inflection on the destination contour. Naturally, in some cases, direction vectors end in desti-

[1] This scale could be interpreted in terms of the spatial resolution of satellite image. In the case of the GOES imager, spatial resolution at nadir is 4 km except for the bands at (0.55–0.75) and (6.5–7.0) $\mu$m where the resolution is 1 and 8 km, respectively. So, for 4 km resolution, one pixel $=$ 16 km$^2$ in actual area. So, the units of scale s can be multiplied by 16 km$^2$ to assess physical meaning.

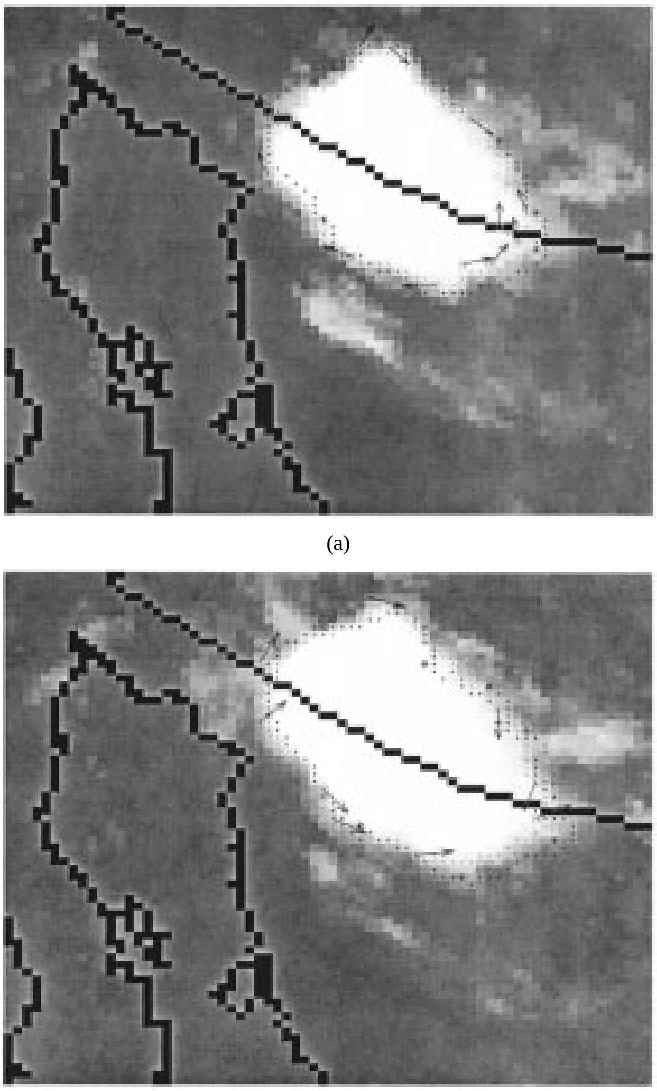


Fig. 9. (a) Direction vectors are shown on Fig. 3(a) using the proposed point based tracking method. Here, the cloud motion is considered between Fig. 3(a) and (b). The squares are inflection points on the source contour, and the asterisks are points of inflection on the destination contour. (b) Direction vectors shown superimposed on Fig. 3(b) using the proposed point based tracking method. The cloud motion is tracked between Fig. 3(b) and (c).

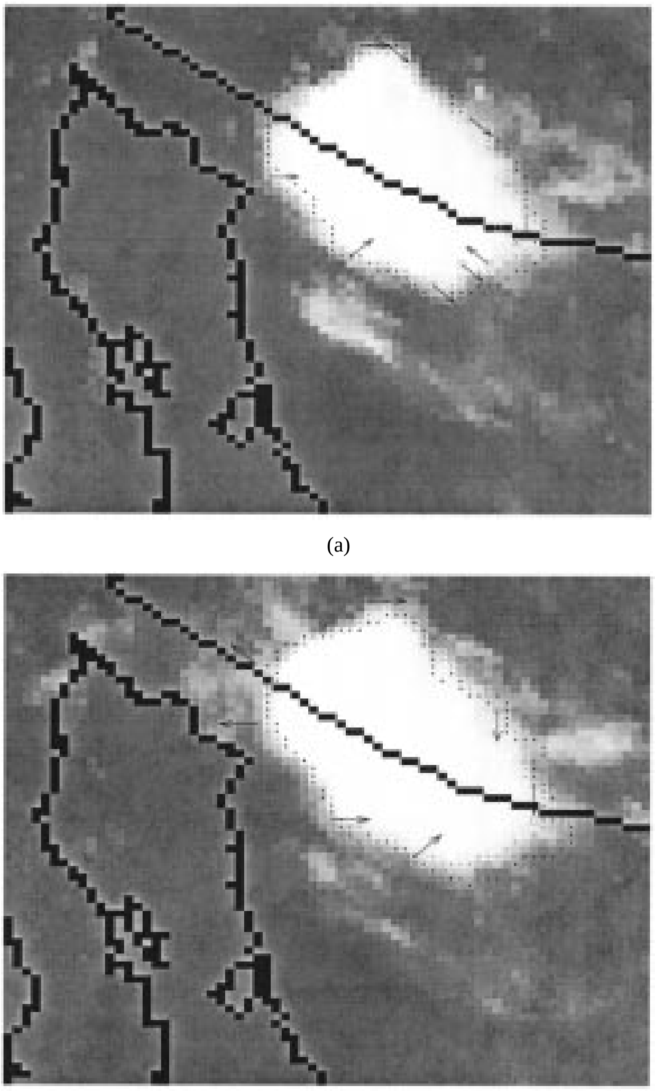


Fig. 10. (a) Direction vectors obtained from correlation method for motion incurred between Fig. 3(a) and (b). (b) Direction vectors obtained from correlation method for motion incurred between Fig. 3(a) and (b).

nation points that are within five-pixel distance from the corresponding source points. Also, for marginal deformation of cloud shape, some source points *almost* coincide with the destination points. Note that while each direction vector is pointing to a destination point, in some cases there are destination points that do not have direction vectors pointing to them. This is due to the mismatch of number of point of inflections in source and destination contours. Also note that during the period of observation, new points of inflection may be generated in the destination contour with no correspondence with any of the points of inflection on the source contour.

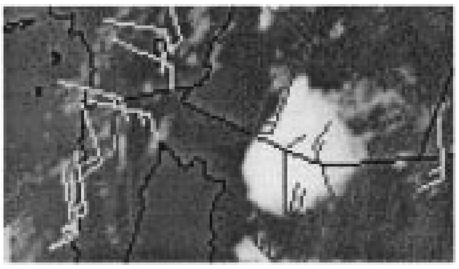


Fig. 11. CMVs shown in high density cloud drift wind result for Sequence I. This is a GOES 8 product from the NOAA/NESDIS forecast product development team at 0300Z on August 16, 2000.

In Fig. 10, the results for the traditional correlation based approach are given. Here, the standard operational procedure is applied to detect the maximal cross correlation. Correlation is evaluated for a 15 × 15 pixel mask around the (source) control point and the mask of identical size within the search window of 61 × 61 pixels (assuming presence of medium to high clouds) in the destination image [17]. The search window centers in the destination image are the same as those of the control points of source contour. Fig. 11 gives the GOES East hemispheric (75 W Long) cloud drift wind result at 0300Z on August 16th, 2000 for the cloud mass under observation. This image provides the experimental high density cloud drift motion vectors generated in

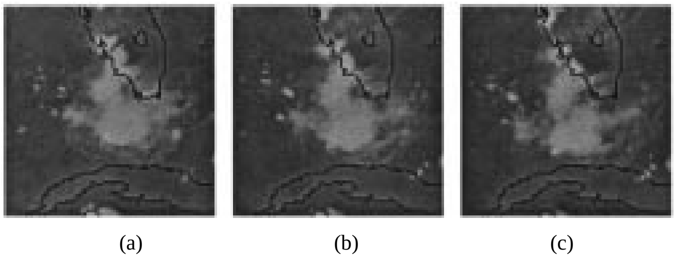


Fig. 12. (a)–(c) Sequence II: Cloud mass in GOES 8 IR image between 25–30 N latitude and 80–85 W longitude off the Florida coast at 0215Z, 0245Z, and 0315Z on August 16, 2000.

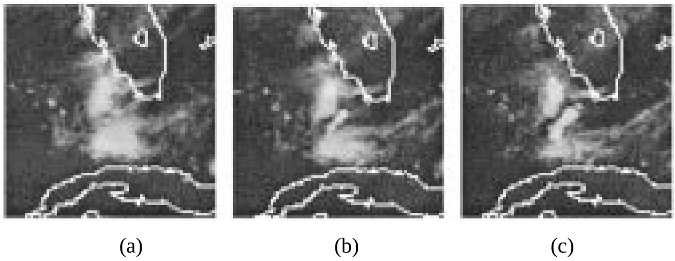


Fig. 13. (a)–(c) Subsequent frames from Sequence II acquired at 0345Z, 0415Z, and 0445Z.

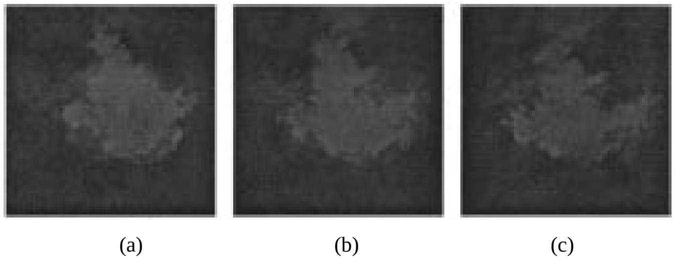


Fig. 14. (a)–(c) AOC scale space images from Fig. 12 at scale s $= 1200$.

every three hourly interval by NOAA/NESDIS forecast product development team. The result is available for the full disk GOES image and is divided in 12 sectors.

To show the consistency of the method, a second sequence (Sequence II) is used off the Florida coast between 80–85 longitude and 25–30 latitude for the same day at 0215Z, 0245Z, and 0315Z, shown in Fig. 12. In Sequence II, the clouds are developing into a cirro-cumulus formation. This region is active with various atmospheric phenomena such as the local warm currents due to Caribbean and Gulf streams from SE to NW and from S to NE off the Florida coast. The current mixes with tropical maritime air from Atlantic generating swirling movements toward the E or NE. Such swirling cloud movement is evident in the cloud tracking analysis results from the point based method. The results are supported by subsequent frames in the image sequence shown in Fig. 13(a)–(c) respectively, (acquired at 0345Z, 0415Z, and 0445Z, respectively).

Using the AOC, the images in Sequence II are scaled at scale s $= 1200$ (Fig. 14). The three-class SSFCM results are given in Fig. 15. Again, the efficiency of cloud mass extraction using SSFCM is clear compared to the corresponding results obtained using FCM classification process as shown in Fig. 16. In Fig. 17, the boundaries used for cloud tracking by the point based method are provided. The motion vectors developed using the point based algorithm are shown in Fig. 18(a) and (b).

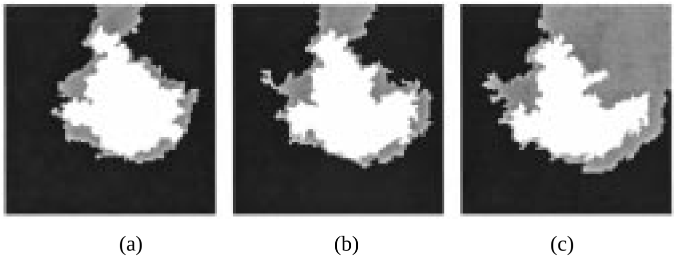


Fig. 15. (a)–(c) Three-class SSFCM results from Fig. 14.

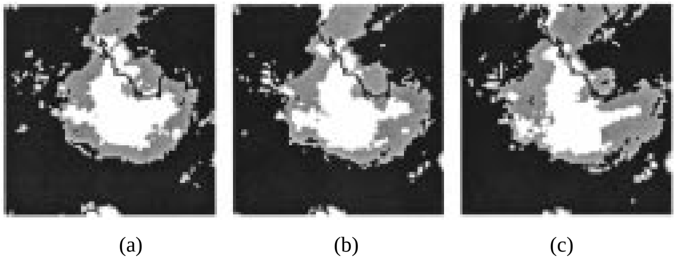


Fig. 16. (a)–(c) Three-class FCM results from Fig. 12.

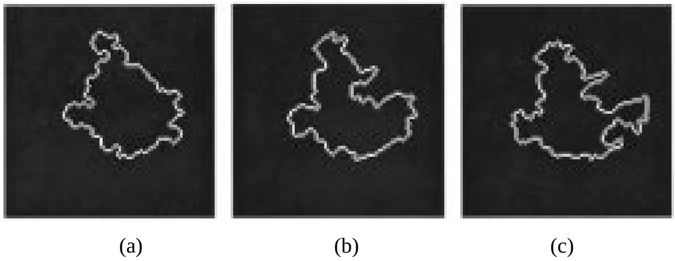
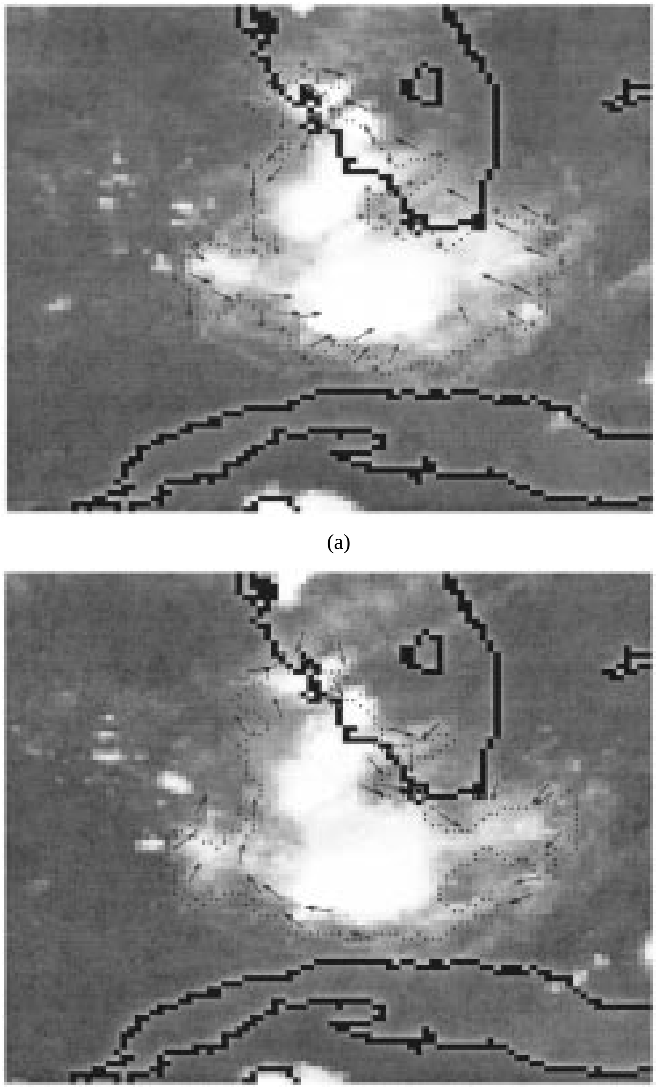


Fig. 17. (a)–(c) Cloud boundaries extracted from Fig. 15.

For comparison, the motion vectors based on the correlation based method are shown in Fig. 19(a) and (b). Again, the corresponding NOAA/NESDIS CMV result for the cloud mass under observation is given in Fig. 20.

We compare the tracking results from the point based and correlation based methods with the NESDIS high density cloud drift wind results. The mean square error (MSE) between motion vectors of the proposed method and the CMVs derived by NESDIS approach is calculated and shown in Table II. To calculate the MSE, the motion vectors within a neighborhood (20 × 20 pixels) of a given derived CMV (from the NESDIS result) are considered for both methods. As shown in Table II, the corresponding MSE measures are lower for the point based approach in all cases.

The point-based results are clearly superior in internal consistency, as compared to the correlation based tracking, in all instances. The directional consistencies of derived motion vectors are compared between Figs. 9(a) and 10(a), 9(b) and 10(b), and 18(a) and 19(a), and, finally, between Figs. 18(b) and 19(b). It is observed that nearly 80% of the direction vectors derived for the first image are consistent with that of the next image in the sequence for the point-based method. We have earlier noted that for sequence I taken over the Southwestern U.S., the mixing of different airflows causes swirling wind flow pattern. This fact is clear from the result derived by the point-based tracking method. Because of this dynamic nature of the atmosphere and the swirling flow, the directional consistency of wind flow over



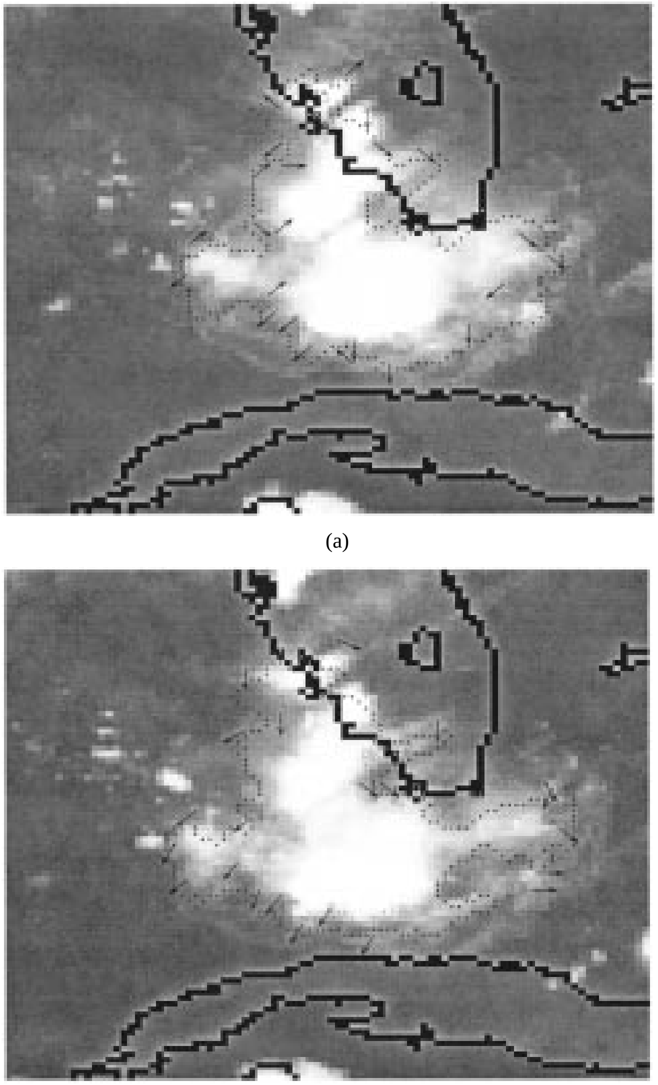
(a)

(b)

Fig. 18. (a) Direction vectors shown superimposed on Fig. 12(a) from the proposed point-based tracking method. The cloud motion represented occurs between Fig. 12(a) and (b). (b) Direction vectors superimposed on Fig. 12(b) using the proposed point based tracking method. The cloud motion considered occurs between Fig. 12(b) and (c).

a longer period is not sustained. This has resulted in marginal inconsistencies of wind flow directions observed in Fig. 9(a) and (b). This is particularly concentrated near the bottom right corner of the cloud mass. In the subsequent pictures of this sequence, it is observed that the shear in this cloud shape has originated from that location.

We note that the NESDIS result and the correlation based result derived in this paper exhibit differences in certain points of inflection on the contour. We have followed the NESDIS suggested template sizes for the correlation based tracking result [17]. However, the NESDIS result is subjected to stringent automatic quality control operations inclusive of the objective editing system for refining the CMVs calculated from the preliminary correlation result. In contrast, our result reflects only the maximum cross-correlation measure between source and the destination mask. The operational NESDIS CMV detection procedure is also verified with respect to collocated rawinsondes data and speed and directional biases are adjusted based on monthly collocation statistics [17]. The other important point in this context is that the correlation based results developed in this paper are derived from images at half hourly sequence. The corresponding NOAA/NESDIS results shown are developed at three 60–min intervals.



(a)

(b)

Fig. 19. (a) Direction vectors from correlation method for motion between Fig. 12(a) and (b). (b) Direction vectors from correlation method for motion between Fig. 12(b) and (c).

To further compare the new point based cloud tracking approach with the conventional correlation based approach, we conducted additional experiments with full disk GOES images. An example GOES 10 IR image sequence is used from May 31st, 2001. Ten representative cloud masses are observed in the sequence, and a total of 365 cloud contour points describing the shape of the clouds are tracked. To compare the internal consistency, the coefficient of variation (standard deviation divided by mean) is computed for magnitude and direction for both methods within each cloud structure (see Table III). In seven of ten cases, the point based method was more consistent in magnitude, and in six of ten cases, the point based method was more consistent in direction. The correlation

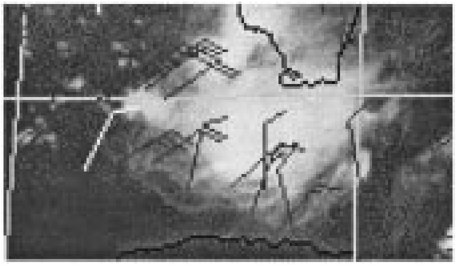


Fig. 20. CMVs shown in high density cloud drift wind result for Sequence II (0300Z on August 16, 2000).

TABLE II
COMPARISON OF TRACKING ACCURACY WITH RESPECT TO PERCEIVED CLOUD MOTION

| Example | Displacement/ Direction | MSE Point based vs. NESDIS CMV | MSE Correlation based vs. NESDIS CMV |
|---|---|---|---|
| Sequence I | Displacement | 23.3 | 37.4 |
| | Direction | 8.5 | 13.8 |
| Sequence II | Displacement | 18.7 | 28.9 |
| | Direction | 7.3 | 21.1 |

TABLE III
INTERNAL CONSISTENCY MEASURED VIA COEFFICIENT OF VARIATION (STANDARD DEVIATION/MEAN). KEY: CoMV = COEFFICIENT OF MAGNITUDE VARIATION, CoDV = COEFFICIENT OF DIRECTIONAL VARIATION, CB = CORRELATION BASED, AND PB = POINT BASED

| Cloud No. | CoMV (CB) | CoMV (PB) | CoDV (CB) | CoDV (PB) |
|---|---|---|---|---|
| Cloud 1 | 0.240 | 0.374 | 0.691 | 0.771 |
| Cloud 2 | 0.309 | 0.239 | 0.871 | 0.514 |
| Cloud 3 | 0.357 | 0.233 | 0.599 | 0.488 |
| Cloud 4 | 0.360 | 0.318 | 0.627 | 0.736 |
| Cloud 5 | 0.339 | 0.329 | 0.943 | 0.626 |
| Cloud 6 | 0.427 | 0.282 | 0.302 | 0.309 |
| Cloud 7 | 0.423 | 0.278 | 0.396 | 0.719 |
| Cloud 8 | 0.273 | 0.269 | 0.611 | 0.422 |
| Cloud 9 | 0.335 | 0.420 | 0.785 | 0.492 |
| Cloud 10 | 0.362 | 0.401 | 0.562 | 0.509 |

TABLE IV
CORRELATION BETWEEN POINT BASED (PB) AND CORRELATION BASED (CB) TRACKING RESULTS FOR MAGNITUDE AND DIRECTION

| Cloud No. | Magnitude correlation (CB) | Cloud No. | Direction correlation (PB) |
|---|---|---|---|
| Cloud 1 | 0.522 | Cloud 1 | 0.412 |
| Cloud 2 | 0.699 | Cloud 2 | 0.397 |
| Cloud 3 | 0.289 | Cloud 3 | 0.314 |
| Cloud 4 | 0.614 | Cloud 4 | 0.692 |
| Cloud 5 | 0.333 | Cloud 5 | 0.245 |
| Cloud 6 | 0.718 | Cloud 6 | 0.462 |
| Cloud 7 | 0.377 | Cloud 7 | 0.215 |
| Cloud 8 | 0.516 | Cloud 8 | 0.302 |
| Cloud 9 | 0.388 | Cloud 9 | 0.502 |
| Cloud 10 | 0.439 | Cloud 10 | 0.367 |

in magnitude and direction between the point based for each cloud grouping is shown in Table IV. The methods are more highly correlated in magnitude (average correlation of 0.49) than in direction (average correlation of 0.39). Due to the ability of the point based method to track high curvature points, the point based method can track a given high curvature point more reliably, which decreases the false detections that lead to errors in computing the direction of motion.

We have implemented the area morphology operations and the clustering algorithms on a SUN Ultra 10 with 64 MB RAM. We utilize a fast pyramidal implementation of the AOC operation [1] that is 1000 times faster than the standard level set implementation. For all images used in this paper, the fast AOC algorithm requires less than 1 s of processing time. For the fuzzy $c$-means based classification technique, the execution time needed to produce Figs. 3 and 5 is less than 2 s, while the execution time is less than 1 s for Fig. 4. With the current implementation of establishing a correspondence between the source and destination contour curvature points for cloud tracking, the execution time is less than 2 s. In comparison, the correlation algorithm requires approximately 3 s each for Figs. 3–5, which is comparable to the overall expense of the scale space classification method. Given a search area of $S$ pixels and a template size of $S$ pixels, the correlation approach will increase in complexity as $O(S^2)$. The fast AOC operation, however, has an $O(S)$ complexity [1]. In cases of high-resolution imagery with large search areas, the contour-based tracking method provided by the scale space classification will improve algorithm efficiency.

## VI. CONCLUSIONS

The paper presents an integrated approach to cloud tracking based on scale space classification. The area based morphological operator generates scale space representation of the image where unnecessary image details are eliminated and large scale segments of similar intensity characteristics are preserved. In images with objects of similar intensity and differing scale, the scale space classifier is able to distinguish between the objects where the traditional fixed scale classifier fails, producing classification errors. The results demonstrate that the scale space approach is superior to the fixed scale correlation approach at a comparable computational expense.

The process of tracking presented here relies on a set of high curvature points that have important invariant geometric properties. The tracking algorithm utilizes a point-based approach and, hence, is not constrained by the rotational and scaling invariance of intensity correlation-based approach. The path and shape coherence constraints embedded in the cost function are applicable to deformable shapes, such as clouds. The complexity of the cost function minimization problem is reduced by the use of the ordering constraint. An important future extension would involve the application of a predictor–corrector scheme for the tracking algorithm, over a number of images from geostationary satellite image sequences.

## REFERENCES

[1] S. T. Acton, "Fast algorithms for area morphology," in *Digital Signal Processing*. New York: Academic, 2001, vol. 11, no. 3.

[2] S. T. Acton and A. C. Bovik, "Piecewise and local image models for regularized image restoration using cross validation," *IEEE Trans. Image Processing*, vol. 8, pp. 652–665, May 1999.

[3] J. C. Bezdek, *Pattern Recognition With Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.

[4] P. Bouthemy and A. Benveniste, "Modeling of atmospheric disturbances tracking in satellite images," in *Image Sequence Processing and Dynamic Scene Analysis*, T. Huang, Ed. Berlin, Germany: Springer-Verlag, 1983, pp. 580–593.

[5] I. Cohen and I. Herlin, "Tracking meteorological structures through curves matching using geodesic path," in *Proc. 6th Int. Conf. Computer Vision ICCV'98*, Bombay, India, Jan. 1998, pp. 396–401.

[6] J. Daniels, C. Velden, W. Bresky, and A. Irving, "Status and development of operational GOES wind products," in *Proc. 5th Int. Winds Workshop*, Lorne, Australia, Feb. 28–Mar. 3, 2000.

[7] G. Dew and K. Holmlund, "Investigations of cross-correlation and euclidean distance target matching techniques in the MPEF environment," in *Proc. 5th Int. Winds Workshop*, Lorne, Australia, Feb. 28–Mar. 3, 2000.

[8] R. M. Endlich and D. E. Wolf, "Automatic cloud tracking applied to GOES and METEOSAT observations," *J. Appl. Meteorol.*, vol. 20, no. 3, pp. 309–319, 1981.

[9] EUMETSAT, "Workshop on wind extraction from operational meteorological satellite data," in *Proc. 4th Int. Wind Workshop*, Saanenmöser, Switzerland, Oct. 20–23, 1998.

[10] ——, "Workshop on wind extraction from operational meteorological satellite data," in *Proc. 5th Int. Wind Workshop*, Lorne, Australia, Feb. 28–Mar. 3, 2000.

[11] F. Mitchell-Christie, *Practical Weather Forecasting*. London, U.K.: William Luscombe, 1977.

[12] A. K. Jain and R. Dubes, *Algorithms for Clustering Data*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[13] J. Lee, R. Weger, S. Sengupta, and R. Welch, "A neural network approach to cloud classification," *IEEE Trans. Geosci. Remote Sensing*, vol. 28, pp. 846–855, Sept. 1990.

[14] J. Morel and S. Solimini, *Variational Methods in Image Segmentation*. Boston, MA: Birkhauser, 1995.

[15] D. P. Mukherjee, A. Zisserman, and M. Brady, "Shape from symmetry: Detecting and exploiting symmetry in affine images," *Phil. Trans. R. Soc. Lond. A*, vol. 351, pp. 77–106, 1995.

[16] F. T. Newland, A. Tatnall, and M. Brown, "Fuzzy object-based generation of cloud motion from sequences of Meteosat satellite imagery," in *Proc. 1st Amer. Meteorol. Soc. Meeting Artificial Intelligence*, Phoenix, AZ, Jan. 13–15, 1998.

[17] S. J. Nieman, W. P. Menzel, and C. M. Hayden *et al.*, "Fully automated cloud-drift winds in NESDIS operations," *Bull. Amer. Meteorol. Soc.*, vol. 78, no. 6, pp. 1121–1133, June 1997.

[18] G. S. Pankiewicz, "Pattern recognition techniques for identification of cloud and cloud systems," *Meteorol. Appl.*, vol. 2, pp. 257–271, Sept. 1995.

[19] J. Parikh and A. Rosenfeld, "Automated segmentation and classification of infrared meteorological satellite data," *IEEE Trans. Syst. Man. Cybern.*, vol. SMC-8, pp. 736–743, 1978.

[20] J. Peak and P. Tag, "Segmentation of satellite imagery using hierarchical thresholding and neural networks," *J. Appl. Meteorol.*, vol. 33, pp. 605–616, 1994.

[21] P. Salembier and J. Serra, "Flat zones filtering, connected operators, and filters by reconstruction," *IEEE Trans. Image Processing*, vol. 4, pp. 1153–1160, Aug. 1995.

[22] I. Sethi and R. Jain, "Finding trajectories of feature points in a monocular image sequence," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, Jan. 1987.

[23] R. A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*. New York: Academic, 1997.

[24] S. P. Smith and A. K. Jain, "Testing for uniformity in multidimensional data," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 73–81, Jan. 1984.

[25] B. Tian, A. Mukhtiar, R. Mahmood, T. V. Haar, and D. Reinke, "A study of cloud classification with neural networks using spectral and textural features," *IEEE Trans. Neural Networks*, vol. 10, pp. 138–151, Jan. 1999.

[26] S. E. Umbaugh, *Computer Vision and Image Processing: A Practical Approach Using CVIPTools*. Englewoods Cliff, NJ: Prentice-Hall, 1997.

[27] J. F. Vega-Riveros and K. Jabbour, "Review of motion analysis techniques," *Proc Inst. Elect. Eng.*, pt. 1, vol. 136, no. 6, Dec. 1989.

[28] I. Weiss, "Geometric invariance and object recognition," *Int. J. Comput. Vis.*, vol. 10, no. 3, pp. 201–231, June 1993.

[29] Q. X. Wu, "A correlation relaxation labeling framework for computing optical flow template matching from a new perspective," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 843–853, Sept. 1995.

[30] S. Yhann and J. Simpson, "Application of neural networks to AVHRR cloud segmentation," *IEEE Trans. Geosci. Remote Sensing*, vol. 33, pp. 590–604, May 1995.

**Dipti Prasad Mukherjee** received the B.E. degree from Jadavpur University, Calcutta in 1985, the M.S. degree from the University of Saskatchewan, Saskatoon, SK, Canada in 1989, and the Ph.D. degree from Indian Statistical Institute (ISI), Calcutta, SK, in 1996.

He is currently a Faculty Member with the Electronics and Communication Sciences Unit, ISI. He was a Visiting Assistant Professor at the Oklahoma Imaging Laboratory, School of Electrical and Computer Engineering at Oklahoma State University, Stillwater, in 1998–1999. He was UNDP Fellow with the Robotics Research Group at the University of Oxford, Oxford, U.K., in 1992. His research interests are in the areas of computer vision and graphics. He has published 25 peer reviewed journal papers and is the author of a textbook on *Computer Graphics and Multimedia*.

Dr. Mukherjee was the recipient of UNESCO-CIMPA fellowships to INRIA, France in 1991, 1993, and 1995 and fellowship to ICTP, Trieste, Italy in 2000.

**Scott T. Acton** (SM'99) received the B.S. degree in electrical engineering from Virginia Tech, Blacksburg, in 1988 as a Virginia Scholar. He received the M.S. degree in electrical engineering and the Ph.D. degree in electrical engineering from the University of Texas, Austin, in 1990 and 1993, respectively.

He is currently an Associate Professor, Department of Electrical and Computer Engineering, University of Virginia, Charlottesville. He has worked in industry for AT&T, the MITRE Corporation, and Motorola, Inc., and in academia for Oklahoma State University, Stillwater. His research interests include biomedical image analysis, multiscale signal representations, diffusion algorithms, active contours, video tracking, area morphology, image segmentation, and content-based retrieval.

Dr. Acton is the winner of the 1996 Eta Kappa Nu Outstanding Young Electrical Engineer Award, a national award that has been given annually since 1936. He also received the 1997 Halliburton Outstanding Young Faculty Award. He serves as Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING and is an active participant in the IEEE, SPIE, and Eta Kappa Nu.