TRACKING FUZZY STORM CENTERS IN DOPPLER RADAR IMAGES

D. Cheng$^1$, R. E. Mercer$^1$, J. L. Barron$^1$ and P. Joe$^2$

$^1$Dept. of Computer Science
The University of Western Ontario
London, Ontario, N6A 5B7

$^2$King City Radar Station
Atmospheric Environmental Services
Toronto, Ontario M3H 5T4

ABSTRACT

In this paper, we describe an automatic storm tracking system to help with the forecasting of severe storms. The concepts fuzzy point, fuzzy vector, fuzzy length of a fuzzy vector, and fuzzy angle between two non-zero fuzzy vectors are first examined. We use a region splitting algorithm with dynamic thresholding to determine storm masses in Doppler radar intensity images. We represent the center of an hypothesized storm using a fuzzy point. These fuzzy storm centers are tracked over time using an incremental relaxation algorithm. The algorithms are tested on actual radar images obtained from the Atmospheric Environment Service radar station at King City, Ontario, Canada. The algorithms are capable of producing storm tracks which closely match human perception.

1. INTRODUCTION

Because of the devastation caused by severe storms, the forecasting of severe storm movement is one of the most important tasks facing meteorologists. To help with this task we have been developing an automatic storm tracking system. Tracking past storm movement is a prerequisite to forecasting future storm movement.

Tracking severe storms in data sets obtained from an operational radar in southern Ontario poses interesting problems. Storm systems move too great a distance and their properties change too much over time (for instance, shapes deform and average intensities change) to allow the use of correlation-based tracking techniques. Other problems encountered include aliasing, storm path crossovers, and the birth and death of storms.

Kreuzski et al. [3] report on a relaxation algorithm that we have developed to overcome these problems. The results, although promising, showed some problems which can be attributed to the deformability and changing intensity values in the storm. The representation of the location of a storm in an image (which is required by the relaxation algorithm) is a point.

The method used to determine this point (center of mass calculation) is sensitive to the properties of the storm (deformability, changing intensities). Although the storm moves in one direction, the algorithm can actually record a movement in the opposite direction due solely to the somewhat arbitrary location of the center of mass of the storm in the subsequent image. This undesirable movement of the point representing the storm affects the performance of the relaxation algorithm. To obtain results that are close to reality with the methodology described in Kreuzski et al., we had to set thresholds in the relaxation algorithm at levels that allowed incorrect tracks to be generated.

To diminish the effect of the arbitrariness of the storm location we have modified the representation of the location of a storm to be a fuzzy point. A fuzzy point is a circle whose inner region represents the uncertainty of the location of a targeted point. Our current work [1] uses fuzzy points, fuzzy vectors, fuzzy lengths of fuzzy vectors, and fuzzy angles between two non-zero fuzzy vectors in the relaxation framework to produce more realistic storm tracks. Note that our interpretation of a fuzzy point is not based on the theory of fuzzy sets [6]; rather, it is in the spirit of fuzzy geometry [4, 5].

2. FUZZY POINT ALGEBRA

A fuzzy point $P = \langle c, r \rangle$ is a circle with center $c = (c_x, c_y)$ and radius $r$. It represents a region where a targeted point can arbitrarily be located. We denote the set of all fuzzy points as $\mathbb{P}$. We define the distance $\delta$ between two fuzzy points $P_1 = \langle c_1, r_1 \rangle$ and $P_2 = \langle c_2, r_2 \rangle$ as:

$$\delta(P_1, P_2) = ||c_1 - c_2||_2 + |r_1 - r_2|,$$

where $(\mathbb{P}, \delta)$ forms a metric space. If we consider $P$ as a subset of $\mathbb{R}^2$, then a fuzzy vector $\overrightarrow{P_1P_2}$ from fuzzy point $P_1$ to fuzzy point $P_2$ is the set of all displacement vectors from a point in $P_1$ to a point in $P_2$. The fuzzy length of this fuzzy vector is then defined as the set of lengths of all displacement vectors in $\overrightarrow{P_1P_2}$. This
set can be represented as the real interval \([d_{\text{min}}, d_{\text{max}}]\), where \(d_{\text{min}}\) and \(d_{\text{max}}\) are respectively the least and greatest distance between any two points in the circles representing the fuzzy points \(P_1\) and \(P_2\). Any fuzzy vector with fuzzy length \([0, d], d \geq 0\) is considered as a zero fuzzy vector. The fuzzy angle subtended by a non-zero fuzzy vector \(P_1P_2\) relative to a non-zero fuzzy vector \(P_1P_3\) is defined as the set of angles subtended by any displacement vector in \(P_2P_3\) relative to a displacement vector in \(P_1P_3\) having touching heads and tails, respectively. The angle between the two displacement vectors, \(\theta\), can be determined by the dot product. Like the fuzzy length of a fuzzy vector, the fuzzy angle between two non-zero fuzzy vectors is a real interval \([\theta_{\text{min}}, \theta_{\text{max}}] \subseteq [0, \pi]\). However, determining the two endpoints \(\theta_{\text{min}}\) and \(\theta_{\text{max}}\) is not trivial. Currently, we use an \(O(n)\) search algorithm to approximate the two endpoints.

3. STORM HYPOTHESIS

A region splitting algorithm with dynamic thresholding is used to determine storm masses in a radar intensity image. This algorithm replaces the one in [3] that uses Horowitz and Pavlidis’ merge-split algorithm [2]. The merge-split process in [3] is determined by a threshold on the difference between the maximum and minimum intensity levels of adjacent regions. We have observed that this criterion is sensitive to outliers in the data. Besides, in order to achieve good results, it requires the threshold to be appropriately and manually chosen for each image. In our new algorithm, we use the standard deviation \(s\) of the average intensity level of a region to govern the splitting process. We find that thresholding using \(s\) is very robust.

A square image \(S\) is initially at level 1 of a quadtree and is divided into four equally sized subnodes \(S_1, S_2, S_3\) and \(S_4\) which correspond to the north-western, north-eastern, south-eastern and south-western regions respectively.

For each subnode \(S_n\), where \(n \in \Lambda^+\) and \(\Lambda = \{1, 2, 3, 4\}\), we compute the mean, \(\bar{I}\), and the standard deviation, \(\sigma\), of the intensity levels of all pixels in \(S_n\). The splitting criterion of the subnode is based on the value of \(s\). The value of \(s\) indicates how well the value of \(I\) represents the average intensity level of all pixels in \(S_n\). We currently set a threshold \(T_s\) on \(s\) to 10.

- If \(s_2\) is large, as indicated by \(s_2 \geq T_s\), then the intensity levels of most pixels in \(S_n\) differ significantly from \(I\) and therefore, the subnode \(S_n\) is split into four smaller subnodes \(S_{n1}, S_{n2}, S_{n3}\) and \(S_{n4}\) at the next level.

- On the other hand, if \(s_2\) is small, as indicated by \(s_2 < T_s\), then we consider \(I\) to be representative of the intensity levels of most pixels in \(S_n\) and we compare the value of \(I\) to the dynamic threshold \(T_s\). \(T_s\) is based on the mean, \(\bar{I}\), and standard deviation, \(\sigma\), of the intensity levels of a subset of pixels in the image with intensity levels greater than \(\sigma_{\text{min}} = 16\). It is computed as \(T_s = \bar{I} + k\sigma\), where \(k = -0.5\). If we have \(I \geq T_s\), then the subnode \(S_n\) will be considered as a part of a potential storm and will be marked for further processing.

The above splitting process continues recursively until no subnode can be further split. Neighbouring subnodes that are marked for further processing will be grouped into a single region,

\[ R = S_{k1} \cup S_{k2} \cup \cdots \cup S_{kn} \]

if they are connected. We say that a subnode \(S'\) is connected to a subnode \(S\) with a distance \(d\) if all of the following conditions hold,

\[
\begin{align*}
I_x + d & \geq u_x' \\
I_y - d & \leq u_y' \\
I_x + d & \geq I_x' \\
I_y - d & \leq I_y'
\end{align*}
\]

where \((I_x, I_y)\) and \((u_x', u_y')\) denote the upper-left and lower-right corners of \(S\), respectively; \((u_x', u_y')\) and \((I_x', I_y')\) denote the upper-left and lower-right corners of \(S'\), respectively; and \(d\) is a threshold currently set to 2 pixels.

3.1. Construction of Fuzzy Storm Centers

After region splitting we have a set of regions \(\{R_1, R_2, \ldots, R_n\}\) in the Doppler radar intensity image which are hypothesized as storms. To represent the location of each storm region, \(R_j\), using a fuzzy point \(P = (c, r)\), we first compute the weighted averages, \(\bar{x}\) and \(\bar{y}\), of the centers of all subnodes forming the region, in the \(x\)- and \(y\)-direction, respectively; and also the corresponding standard deviations, \(s_x\) and \(s_y\). Then, the center of the fuzzy point \(P\) is taken as \(c = (\bar{x}, \bar{y})\); and the radius of the fuzzy point is determined by \(r = k_r \max(s_x, s_y)\), where \(k_r\) is a parameter to control the size of the fuzzy point. We currently use \(k_r = 0.5\). We interpret the above construction as fitting a quasi-circular Gaussian surface through the storm region. Since the data points in the storm region do not necessarily spread around the peak of a Gaussian surface, we could not employ a least-square method to perform a Gaussian fit. Since the fuzzy point represents the uncertain location of the storm center, we refer to the fuzzy point as the fuzzy storm center.
4. INCREMENTAL RELAXATION ALGORITHM

Once a set of storms has been hypothesized for the radar intensity image sequence, the correctness of these storms can be verified by tracking them over time. Our tracking algorithm is based on Krezowski's temporal relaxation algorithm with property coherence [3]. We have selected the size of a fuzzy storm center as a property to be coherent over time.

Let $S_k$ be an hypothesized fuzzy storm center in the $k$th image. A disparity represented by a fuzzy vector $S_j S_{j+1}$ is constructed from $S_j$ to $S_{j+1}$ if the infimum of the fuzzy length of the fuzzy vector is less than a threshold, $T_d$, which is set to a default value of 10 pixels; and concurrently, the two fuzzy storm centers have compatible sizes. We define a property function, $f_S$, to measure the size-compatibility of two fuzzy storm centers $S_1 = (c_1, r_1)$ and $S_2 = (c_2, r_2)$ as:

$$f_S(S_1, S_2) = \begin{cases} 1 - \frac{|r_1 - r_2|}{\max(r_1, r_2)} & \text{if } r_1 > 0 \text{ or } r_2 > 0, \\ 1 & \text{otherwise.} \end{cases}$$

A size-compatibility threshold, $T_{sc}$, is set to 0.5. Note that if $T_{sc}$ is set to 1, then a disparity will be constructed between two fuzzy storm centers only when they have exactly the same size. On the other hand, if $T_{sc}$ is set to 0, then the size-compatibility criterion is effectively removed. We measure the partial compatibility between two adjacent disparities using a weighted sum of three components: length compatibility $C_a$, angle compatibility $C_b$, and size compatibility $C_s$. The overall compatibility function is defined as:

$$C = w_d C_a + w_b C_b + w_s C_s,$$

where $w_d$, $w_b$, and $w_s$ are normalized weights such that $w_d + w_b + w_s = 1$. We currently use $w_d = 0.2$, $w_b = 0.2$ and $w_s = 0.6$. Two adjacent disparities are connected together if their compatibility value is greater than a threshold:

$$C \left( S_j S_{j+1}, S_{j+1} S_{j+2} \right) > T_c,$$

where $T_c$ is currently 0.2. When all qualified adjacent disparities have been linked together, the certainty of each disparity is refined iteratively by relaxation on the overall compatibility among its adjacent disparities. Consider a disparity $d = S_j S_{j+1}$. The initial certainty of the disparity, denoted as $p_0(d)$, is set to $f_S(S_j, S_{j+1})$. During each iteration, we apply both spatial and temporal consistency constraints to compute the supporting and contradictory evidence of the disparity using the compatibility values. Let $E_s$ and $E_c$ denote the supporting and contradictory evidence, respectively. Let $n_s$ and $n_c$ denote the number of supporting disparities and the number of contradictory disparities, respectively. The four quantities are reset to zero at the start of each iteration.

- To apply the temporal consistency constraint, for each adjacent disparity $d_k$ to $d$ of the form $d_k = S_{j-1} S_j$ or $d_k = S_{j+1} S_{j+2}$, we compute the compatibility at the $k$th iteration $(k > 0)$ between the two disparities as:

$$C_k(d, d_k) = \frac{w_1 C(d, d_k) + w_2 \left( \frac{p_{k-1}(d) + p_{k-1}(d_k)}{2} \right)}{w_1 + w_2},$$

where $w_1$ and $w_2$ are normalized weights that sum to 1. We currently use $w_1 = 0.4$ and $w_2 = 0.6$. If $C_k(d, d_k) > T_k$ (we use $T_k = 0.6$), then we add $p_{k-1}(d)$ to $E_s$ and increment $n_s$ by 1. Otherwise, we add $p_{k-1}(d)$ to $E_c$ and increment $n_c$ by 1.

- To apply the spatial consistency constraint, for each disparity $d_k$ which has the same head storm or tail storm as $d$, if $p_{k-1}(d) \geq p_{k-1}(d_k)$, then we add $p_{k-1}(d)$ to $E_s$ and increment $n_s$ by 1. Otherwise, we add $p_{k-1}(d)$ to $E_c$ and increment $n_c$ by 1.

- The certainty of the disparity $d$ at the $k$th iteration is modified by:

$$p_k(d) = \begin{cases} \frac{1}{2} \left( 1 + \frac{w_s E_s + w_c E_c}{w_s E_s + w_c E_c} \right) & \text{if } E_s \neq 0 \text{ or } E_c \neq 0, \\ 0 & \text{otherwise;} \end{cases}$$

where $w_s$ is the weight of the supporting evidence and is computed as $w_s = \frac{n_s}{n_s + n_c}$; and $w_c$ is the weight of the contradictory evidence and is computed as $w_c = \frac{n_c}{n_s + n_c}$.

- The iterative process stops at the $k$th iteration when the certainty of each disparity has converged to the desired level of confidence, say to $n$ decimal places (we use $n = 6$):

$$\varepsilon = |p_k(d) - p_{k-1}(d)| < 10^{-n},$$

for each disparity $d$ or the maximum number of iterations has been reached: $k \rightarrow T_k$, where $T_k$ is currently set to 20.

Once the relaxation process has converged, we construct a set of all longest tracks such that each disparity has a final certainty over a threshold $T_p$ (we use
$T_p = 0.85$). We choose a subset of these tracks, with the condition that no storm lies upon more than one chosen track.

We have changed the implementation of the algorithm from the processing of a complete image sequence of known length to an incremental process. Given $n$ images the hypothesized storm disparities are relaxed. When an $(n+1)^{th}$ image is added, the relaxation is restarted using the results for the first $n$ images plus the hypothesized storms of the $(n+1)^{th}$ image. We have observed empirically that the result is always the same as if all $n+1$ images had been initially relaxed. The difference in computation speed between the two methods is insignificant since relaxation converges within ten iterations in either mode. The incremental algorithm allows us to view the current storm tracks as the data become available.

5. EXPERIMENTAL RESULTS

Due to space limitations, we only show test results of our algorithms on two images from two image sequences: the 77-series and the 18-series. The storms are moving from left to right in the 77-series and the results obtained from our algorithms agree with this fact, as shown in Figure 1. Storm movements are more complicated in the 18-series. Some of the storms are moving from left to right, and some are moving from top to bottom. Our algorithms are capable of tracking these storm movements, as shown in Figure 2. To ensure the display of smooth tracks, we use B-splines to plot the tracks, with one or more applications of 3-point averaging of the fuzzy storm centers before the B-spline approximation is performed. By using the fuzzy storm center concept to obtain the tracks and the B-spline approximation with 3-point averaging for their display, we can now obtain smooth tracks that are long and smooth and which closely match human perception of a “motion picture” of a storm image sequence.

6. REFERENCES


