DOT/FAA/NR-91/8

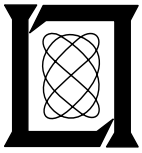
Project Report
ATC-182

Storm Tracking for TDWR: A Correlation Algorithm Design and Evaluation

E. Chornoboy

14 July 1992

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS



Prepared for the Federal Aviation Administration,
Washington, D.C. 20591

This document is available to the public through
the National Technical Information Service,
Springfield, VA 22161

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
ATC-182	DOT/FAA/NR-91/8	

4. Title and Subtitle	5. Report Date
Storm Tracking for TDWR: A Correlation Algorithm Design and Evaluation	14 July 1992
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Edward S. Chornoboy	ATC-182

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Lincoln Laboratory, MIT P.O. Box 73 Lexington, MA 02173-9108	11. Contract or Grant No. DTFA01-89-Z-02033

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
Department of Transportation Federal Aviation Administration Systems Research and Development Service Washington, DC 20591	Project Report
	14. Sponsoring Agency Code

15. Supplementary Notes

This report is based on studies performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology under Air Force Contract F19628-90-C-0002.

16. Abstract

Storm Movement Prediction (SMP) is a proposed (future) product for Terminal Doppler Weather Radar (TDWR), aiding controllers by tracking storms approaching and passing through the terminal environment. Because the scan strategy (data acquisition) of TDWR has been critically designed to meet the needs of its primary function, which is the detection of hazardous low-altitude wind shear, there is the question of whether reliable storm tracking can be obtained from the TDWR data set. The objectives of storm tracking involve a scope (spatial range) much larger than that required for the wind-shear algorithms where volume coverage is confined (in off-airport sited radars) to a sector covering the important approach and departure corridors and the only 360-degree scans are near-surface scans for gust-front detection.

This report examines the application of a correlation based method of detecting storm motion, testing the notion that reliable storm motion can be inferred from existing TDWR data. In particular, storm motion derived from an analysis of the TDWR Precipitation product (PCP) is studied. A summary description of the algorithm is presented along with an analysis of its performance using data from MIT Lincoln Laboratory's TDWR testbed operations in Denver (1988) and Kansas City (1989).

The primary focus of the present analysis is on the reliability of tracking, since the algorithm is expected to operate in an autonomous environment. Some attention is given to the idea of prediction, in the form of storm extrapolation, considering 15, 30, and 60 minute predictions. Specific areas for improvement are identified, and application of the algorithm track vectors for long-term prediction (30-60 minutes) is discussed with reference to example PCP images.

17. Key Words		18. Distribution Statement
TDWR	Storm Motion	This document is available to the public through the National Technical Information Service, Springfield, VA 22161.
Algorithm	Correlation Tracking	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	94	

ABSTRACT

Storm Movement Prediction (SMP) is a proposed (future) product for Terminal Doppler Weather Radar (TDWR), aiding controllers by tracking storm systems approaching and passing through the terminal environment. Because the scan strategy (data acquisition) of TDWR has been critically designed to meet the needs of its primary function, which is the detection of hazardous low-altitude wind shear, there is the question of whether reliable storm tracking can be obtained from the TDWR data set. The objectives of storm tracking involve a scope (spatial range) much larger than that required for the wind-shear algorithms where detailed volume coverage is confined (in off-airport sited radars) to a sector covering the important approach and departure corridors and the only 360 degree scans are near-surface scans for gust-front detection.

This report examines the application of a correlation based method for detecting storm motion, testing the notion that reliable storm motion can be inferred from existing TDWR data. In particular, storm motion derived from an analysis of the TDWR Precipitation product (PCP) is studied. A summary description of the algorithm is presented along with an analysis of its performance using data from MIT Lincoln Laboratory's TDWR testbed operations in Denver (1988) and Kansas City (1989).

The primary focus of the present analysis is on the reliability of tracking, since the algorithm is expected to operate in an autonomous environment. Some attention is given to the idea of prediction, in the form of storm extrapolation, considering 15, 30, and 60 minute predictions. Specific areas for improvement are identified, and application of the algorithm track vectors for long-term prediction (30-60 minutes) is discussed with reference to example PCP images.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS
(Continued)

LIST OF TABLES

1. BACKGROUND

1.1 Introduction

Terminal Doppler Weather Radar (TDWR) is the FAA sponsored *fully automated* system that combines sophisticated analysis algorithms with Doppler radar surveillance in an attempt to improve the detection and classification of hazardous weather events in the airport terminal environment [1p7]. Increasing the safety of aircraft departure and arrival is perhaps the most important motivation for its development, but also the likelihood that TDWR will favorably impact the planning and operational efficiency of air-traffic control is another reason for its existence. Hence, a number of "augmenting" products have been suggested for TDWR (see Turnbull et al., [2]), among which is Storm Motion Prediction (SMP), defined by Turnbull et al. [2] as "predict[ing] positions of significant storm cells using a cell-tracking algorithm." A prototype TDWR SMP algorithm, based on a correlation processor, has been developed at MIT Lincoln Laboratory.

1.2 Design Strategy

1.2.1 TDWR Data

The hazardous modes of TDWR (microburst, gust-front detection) define and thereby restrict the TDWR data available. One of the primary objectives of this report is to determine whether a viable storm tracking algorithm can operate from such a data base. The TDWR scan strategy [3] allows for both on-airport and off-airport siting of its radar. Off-airport siting has the obvious advantage, for microburst detection, of allowing sectored scanning—concentrating attention directly over the airport. This can be perceived, however, as a disadvantage for storm tracking because sectored scanning does not provide a vertical sampling uniform in direction from the airport, and therefore it is not optimal for the larger scale objectives of storm tracking. Only low-elevation 360 degree tilts, which are needed for gust-front detection, are assured by TDWR scanning. Although weather channel data may become available from an airport's surveillance radar (providing an integrated vertical profile), this report will not make that assumption and instead focus on whether robust autonomous tracking can be obtained from the low elevation tilts available from TDWR.

In TDWR, storm-reflectivity data will be presented to ATC and TRACON[1] personnel in the form of the Precipitation product (PCP)[2]: a "graphic display of precipitation ... in terms of

---

[1]Airport Traffic Control and Terminal Radar Approach Control

[2]As planned, the PCP product consists of high-resolution and low-resolution maps. The low-resolution map will be a 1 $km \times 1$ km Cartesian resampling of a single 360 deg reflectivity tilt reduced to the six-level National Weather Service (NWS) reflectivity scale. The elevation of this tilt is selected to provide reflectivity maps covering the important airspace corridors (along, and

six levels of reflectivity" [2]. The notion of "storms", their motion, and prediction of their future locations therefore requires—for the end users—a consistency with that representation. Clearly the SMP product should complement (i.e., serve as an overlay to) the graphical PCP display. This desired association, plus a consideration for the computational requirements of storm tracking, suggests that the PCP product itself might serve as input for a storm-motion algorithm. Clearly, valid concerns are the limited vertical extent of the PCP product and its reduced resolution (only the low-resolution 1 km PCP map is currently planned for algorithm input and this map is updated only at 5 minute intervals). Preliminary studies have indicated that the resulting tracking resolution (see Section 2.2.1) would not degrade the algorithm's functioning below that of its intended use as a *planning aid* for air-traffic control.

1.2.2 Correlation Tracking

In a study comparing tracking algorithms for NEXRAD application, Brasunas [4] could not (unfortunately) determine one algorithmic form (centroid, correlation, etc.) to be best when given the variety of weather situations in his data base. In particular, the centroid tracker he examined did poorly when tracking extended storms. His correlation tracker performed well with extended storms but only tracked storm translation and not (as was desired) the vector sum of storm translation and propagation (see Section 1.2.3 below). Brasunas [4] hypothesized that a binary form of correlation tracker might succeed where his failed. Given that extended structures can be expected at some time in most locations (and also considering the coarse resolution of the PCP product), a binary correlation tracker was determined most appropriate for this prototype study.

1.2.3 Storm Tracking vs Prediction

Because the present algorithm design is based solely on the use of PCP map input (i.e., no ancillary data), expectations for algorithm success should focus on its performance as a storm-tracking algorithm. Consequently, the "predictive" function is reduced to one of extrapolation. In this context, it is important to differentiate among

1. the tracking of (existing) cell translation,
2. the tracking of propagative (growth/decay) trends, and
3. the prediction of new storm cell formation.

Storm propagation, as described by Brasunas [4], can be categorized into steady-state and non-steady-state forms of growth/decay (see Chappell [5] for specific analysis). While individual

---

projecting from, the runways) used for aircraft approach and departure. The low-resolution map has a range extent compatible with the TDWR range requirement. The high-resolution map will use a 0.25 km sampling resolution and extend only to a range of 5 km.

cells typically translate as per the steering-level winds, the propagative growth of new cells can impart an apparent motion to a storm's envelope. Propagation can occur in a uniform fashion, in which case it is hypothesized that regular sampling should be capable of tracking the "steady-state" vector sum of translation and propagative motion for the storm envelope. This notion of storm movement "effected" by steady-state propagative growth is illustrated in Figure 1. Hence,

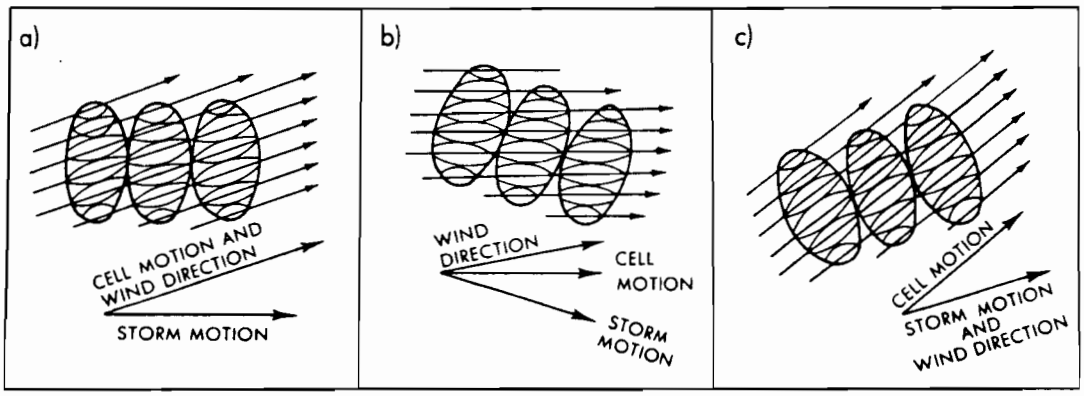


Figure 1. Examples of propagative storm motion (from Brasunas [4]).

for situations that fit the models of Figure 1, there is the interpretation that extrapolation will follow regular growth and therefore "predict." The goals of the present algorithm are therefore directed at the above listed items *1* and *2*, but not *3*. A separate Convection Initiation product (Turnbull et al., [2]), intending to provide predictions of new thunderstorm cell formation, has been proposed but integration of this product with SMP is considered a subject for future study.

1.2.4 Output Format

A strict format for presentation of the SMP output to controllers has yet to be defined. One important question in this regard is whether the graphical display should explicitly illustrate predicted storm location (e.g., by using translations of contour overlays). In view of the above implied algorithm limitations regarding storm-cell growth and decay, there is valid concern that controllers would attempt to use the graphically extrapolated storms to vector aircraft. To be conservative, track-vectors, showing direction and speed, should be sufficient and most convenient (leaving the process of extrapolation implied, i.e., mental extrapolation by controllers). It is clear from experience with other TDWR products that controller input, after trial usage, can best focus

the output design. The present algorithm maintains flexibility by supporting multiple forms of output.

1.2.5 Algorithm Structure

The SMP algorithm actually has been constructed as two separate algorithms: one executing a tracking function and the other, extrapolation. The extrapolator combines tracking information with symbolic-storm representations—here, contours derived from threshold analysis of the PCP maps—to produce display envelopes that delineate future storm location. A functionally separate extrapolation algorithm is favored because, in addition to output-format concerns (see above), it facilitates the creation of a flexible and interactive display. For the most part, this report will focus on evaluation of the tracking algorithm; the extrapolation algorithm will only be used to explore the predictive consequences of the tracking output.

2. ALGORITHM DESCRIPTION

2.1 Introduction

The SMP algorithm is not organized around a meteorological analysis of the PCP maps, as would occur with, for example, centroid [6], peak-cell [7], or even some earlier attempts at correlation tracking (see, for example, Blackmer et al. [8]). Hence, it does not explicitly track storm cells from one scan to the next. Instead, the algorithm determines motion using a well-known computer-vision method for motion detection (see, for example, Glazer [9]) whereby small local image windows are tracked between time frames, defining a uniform grid of measurement vectors representing the apparent motion (or "optic flow") between source images. A similar method was successfully applied by Vesecky et al. [10] to develop an *automated* system that tracks sea-ice dynamics. Also, Rinehart [11] has considered the method for the tracking of internal storm motion, however in a non-automated setting. How well such a method can work in an automated (meteorological) setting and how well the PCP-derived optic flow corresponds to actual storm motion are issues that are addressed here.

The essential features of the tracking concept are illustrated in Figure 2. There, a hypothetical weather-radar reflectivity-channel image, Panel *a*, is illustrated as displayed using the National Weather Service six-level threshold scale. Two regions of storm activity are illustrated; one is an extended storm containing multiple cells. Panel *b* illustrates how the image is viewed by the Cross-Correlation Tracker (described in Section 2.2). Only reflectivity levels 2 or greater (greater than 30 dBz) are tracked, and each above-threshold pixel is given equal weight (1.0). The image is partitioned into smaller overlapping sub-regions known as correlation boxes. Five such correlation boxes are outlined in the panel, where a 50% areal overlap between adjacent correlation boxes has been illustrated. The binary-pixel image contained within a correlation box is tracked from one time-frame to the next, and a displacement vector is computed and assigned the spatial coordinate corresponding to the box center. A uniform grid of displacement vectors is thereby derived corresponding to the locations marked by the cross hairs in Panel *b*. Panels *c* and *d* illustrate the original image with two possible output displays provided by the SMP algorithm. In Panel *c*, velocity vectors (with storm speeds in knots) have been provided for significant storms and, in the case of extended storms containing cells, significant storm cells. Centroid locations are determined using a Binary Storm Object algorithm; velocity values at the centroid locations are determined by interpolating the grid values in Panel *b*. Panel *d* illustrates a storm-envelope contour (corresponding to level-2 and above weather) which has been extrapolated to indicate storm location 10–15 minutes into the future.

The algorithm can be segmented into four functionally disjoint components organized as illustrated in Figure 3:

1. detect raw motion,

2. perform (in parallel to 1) an independent meteorological analysis,

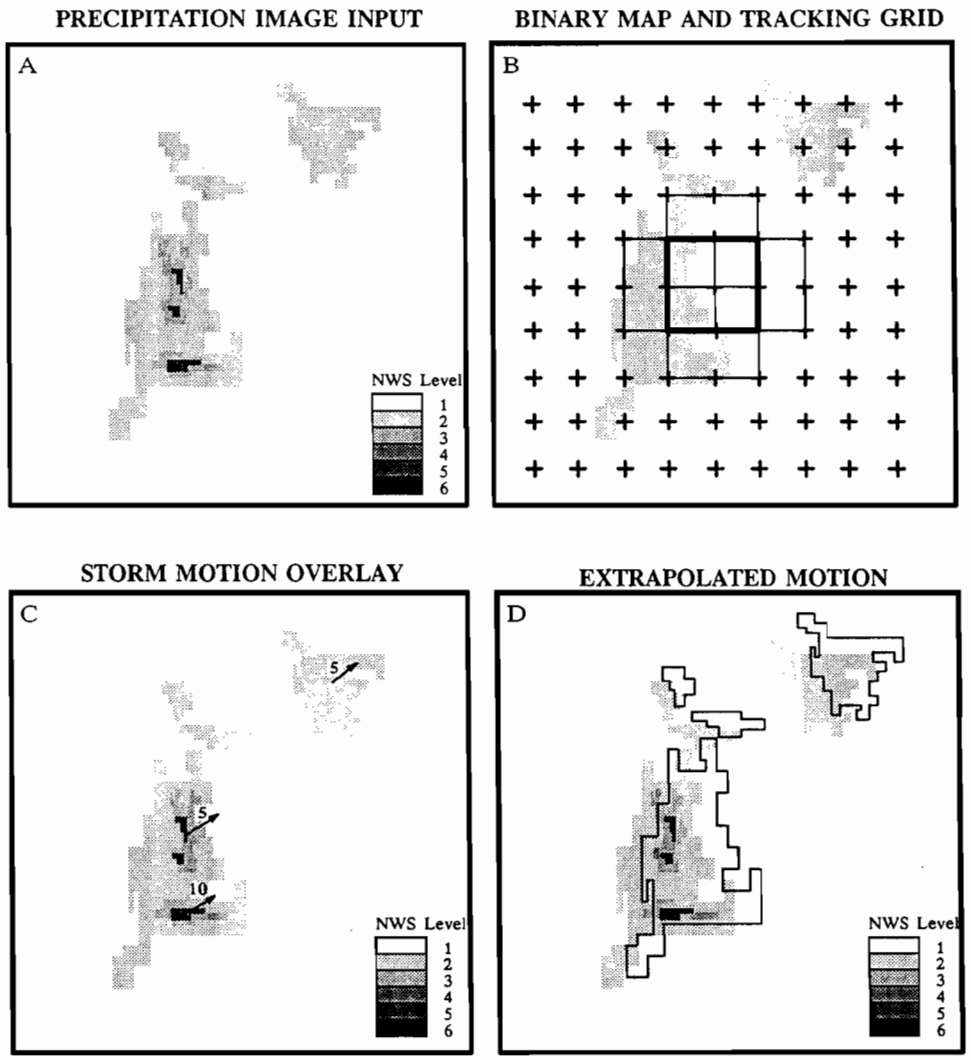


Figure 2. The Storm Movement Prediction (SMP) algorithm: processing steps.

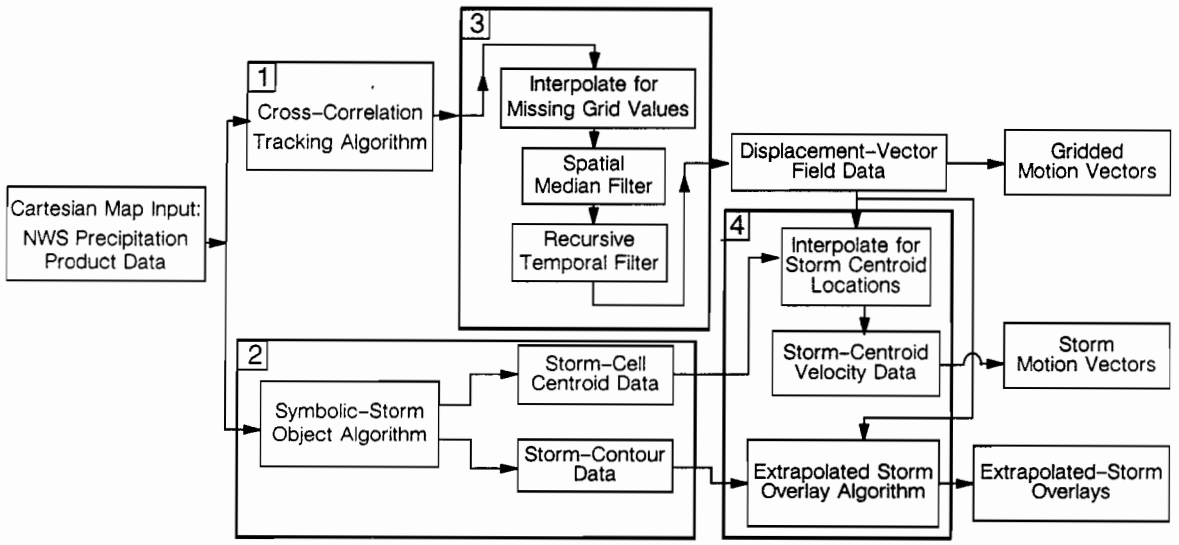


Figure 3. Storm Movement Prediction: functional organization.

3. refine raw motion estimates and reconcile with physical constraints, and
4. integrate motion information with weather-oriented objectives.

Input (Cartesian) PCP maps are processed in successive pairs by the Cross-Correlation Tracking (CCT) algorithm which estimates image motion between time frames in the form of a field of displacement vectors [Figure 2(b)]. A series of smoothing procedures is applied next to the CCT output to improve continuity. In parallel, the Cartesian maps are also processed by a Binary Storm Object (BSO) algorithm which provides a single NWS-level[3] storm characterization of the radar image. These binary-storm representations are used in two ways. First, they are used to compute centroids that define locations for SMP-output motion vectors, where output motion vectors are derived by interpolating displacement-vector-field data to these centroid locations [see Figure 2(c)]. Second, BSO representations are used by the Extrapolated Storm Overlay (ESO) algorithm which creates, from the current weather display, a storm-contour overlay extrapolated to predict future storm extent [Figure 2(d)]. The following subsections provide further details for each of the component algorithms.

[3]The correspondence between NWS levels and dBz values is as follows: level-1 = 18 to 30 dBz, level-2 = 30 to 41 dBz, level-3 = 41 to 46 dBz, level-4 = 46 to 50 dBz, level-5 = 50 to 57 dBz and level-6 = above 57 dBz.

2.2 Cross-Correlation Tracking

This algorithm, being the most substantial (regarding code length and execution time) of the component algorithms, provides the "raw data measurements" of image motion that, along with the output from the BSO algorithm, is the basis for most of the ensuing processing steps.

2.2.1 Binary Cross-Correlation

The CCT tracker is a binary correlation algorithm: motion is detected from the analysis of binary images created by thresholding the input Cartesian maps at NWS level-2. Because only (binary) storm *envelopes* are tracked, presumably the algorithm will follow the vector sum of translation and steady-state growth (see Section 1.2.3).

Correlation Boxes: Local Image Windows. Figure 2(b) illustrates how binary Cartesian maps are partitioned into small local image windows, or "correlation boxes." The pattern of binary pixels within each correlation box constitutes a local feature of the image, and the CCT algorithm determines motion by tracking features from one time frame to the next. In the absence of noise and assuming translational motion only, a correlation box size of 5-7 pixels per side has been recommended as a general minimum [9]. Preliminary work, however, has indicated the need for a larger box size in the context of PCP processing; hence, the CCT algorithm currently implements a 28×28 pixel correlation box. Ideally, correlation boxes should be centered one per lattice point defined by the digital PCP image. This would provide the maximum amount of redundancy and ensure that all possible configurations of image features would be examined. Such an oversampling is desired to reduce the effects of outlier measurements, as are likely to result from correlation boxes containing weak image features. Computational limits make it necessary to compromise this sampling plan. Hence, correlation boxes are defined such that neighboring boxes overlap, at present by 50%. The above settings result in a displacement-vector field having a 14 km sampling interval. An operational example, illustrating the complete velocity vector grid is shown in Figure 4.

Feature Displacement and Velocity Measurement. The CCT algorithm tries to track each time-1 correlation box to its most likely position in the time-2 frame as follows. For each time-1 correlation box, a set of feasible displacement vectors is delimited by a radius corresponding to a plausible maximum storm velocity. For each feasible displacement, the correlation between the translated correlation-box image and the underlying time-2 map is computed. The correlation used is a variance normalized version of binary correlation (see Burt et al. [12]). The set of correlation values, indexed by feasible displacements, is said to generate a correlation surface. The time-1 to time-2 box displacement is taken to be that displacement which maximizes the correlation coefficient over all feasible displacements. Velocities are computed by normalizing the displacement with respect to the time elapsed between frames.

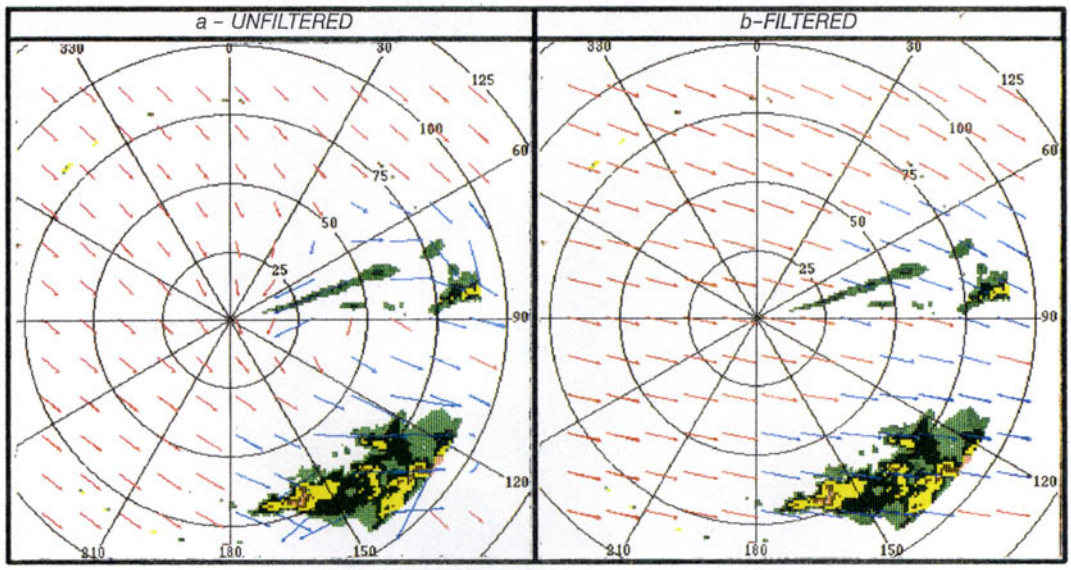


Figure 4. Storm motion internal velocity grids: a. unfiltered, b. filtered.

Confidence Weighting. The selection of optimum displacement by maximizing the correlation coefficient is not robust: unimodal correlation surfaces are not guaranteed (nor are global maxima) and maxima can be masked easily by image noise. In addition, broad surface peaks (in the correlation surface) contribute to displacement measurements having large variability. Some form of confidence weighting is usually required to prune outlier measurements that result from inadequate feature content or poor feature correlation. Two such checks are internal to the CCT algorithm. First, before a correlation box is tracked, it must satisfy a criterion for a minimum number of "on" pixels. This is a check that the correlation-box feature is of sufficient size such that it is expected to exist at time-2. Second, a displacement measurement is accepted only if the correlation maximum surpasses a minimum threshold. This is to ensure that there is sufficient match of image detail before a displacement is considered likely. That these two measures alone are inadequate is evident in Figure 4(a) where errors due to edge effects and second trip weather are clearly evident (blue arrows represent the measurement points). Additional checks and constraints for the correlation process are clearly desirable, but must be left as a topic for future investigation.

Missing Measurements. Generally only a subset of correlation boxes produce displacement measurements for any given time frame, often forming but a sparse sampling. In Figure 4, these vectors are illustrated in blue. However, displacements (velocities) are desired for all points of the correlation box lattice in order to support subsequent spatial and time-domain filtering. The present algorithm substitutes missing values with values derived from interpolation, shown as red vectors in Figure 4. In the event that the CCT algorithm cannot compute a displacement vector for any correlation box, the SMP algorithm treats this case as if there were no measurements for this time frame, identical to that of a gap in the input-map sequence, and the resulting output of the SMP algorithm depends on whether time-weighted filtering has been selected (see below).

Intrinsic Measurement Resolution. Output motion track vectors are scaled (as storm velocities) in knots. Because the PCP product has a spatial resolution of 1 $km \times 1$ km and is updated at the nominal rate of once per five minutes, velocity estimates from raw displacement measurements have an intrinsic resolution of ± 3.24 knots (i.e., a motion due east is determined as $6.48n \pm 3.24$, $n = \ldots, -1, 0, 1, \ldots$ knots). This should be sufficient since current display conventions call for storm velocities quantized in 5 knot increments. Temporal filtering can improve the accuracy of this speed estimate as described below (Section 2.4.3).

2.3 Symbolic Storm Identification

This portion of the SMP algorithm provides an analysis of image weather content and distills that information into a form that can be united with the motion information provided by the CCT algorithm. Symbolic identification is currently provided by the Binary Storm Object (BSO) algorithm which simply characterizes storms using single fixed-level interval thresholding. A "storm" is defined to be any region of contiguous pixels with reflectivity values between given upper *and* lower threshold bounds. The algorithm is used in different ways to provide two forms of SMP output: motion vectors associated with significant storm elements and storm contours showing predicted storm location.

2.3.1 Output Vector Location

For the purpose of tagging storms with motion vectors [ref. Figure 2(c)], it is recommended that motion information be provided for level-2 (and above) storms, assuming the storm also satisfies a minimum size criterion. This leads to difficulties in extended-storm situations where a region of level-2 reflectivity may contain a number of isolated higher-level reflectivity cells. In these cases, the association of a single motion vector with one large extended storm does not seem reasonable. At the same time, it is not desirable to eliminate the tagging of isolated, but significant, level-2 (exclusive) storms. To resolve this issue, an adaptive storm cell definition has been adopted to define SMP output vector locations.

Output vectors are located as follows. A list of centroids for motion-vector output is generated by a descending threshold analysis of the input map. First, level-6 "storms" are identified using the BSO algorithm. Centroids for these level-6 exclusive regions are computed and added to the output list if they satisfy a minimum size criterion. The analysis is repeated at level-5 and centroids are added to the output list for all significant level-5 regions that do not already contain a significant level-6 region. This procedure is repeated until isolated level-2 storms have been added to the list of output centroids. As a final measure, the list is pruned using a threshold for a minimum separating distance between centroids. This last step limits output vectors in the event that there is clustering among what otherwise appear as distinct cells. A limit on the total number of output vectors can be specified, in which case locations are taken from the top of the list.

2.3.2 Storm Envelopes for Graphic Display of Extrapolations

As with output storm-motion vectors, there is the need to keep the display of extrapolation contours to an efficient minimum. Extrapolation contours are generated only for level-2 thresholded storms. These symbolic storms are passed to a contouring algorithm that generates a list of contour segments circumscribing the level-2 weather [ref. Figure 2(d)]. Velocities, from the gridded field, are associated with the storms in order that the ESO algorithm may position the storm contours and illustrate the desired prediction.

2.4 Smoothing Procedures

The CCT algorithm does not provide reliable estimates of motion under all operating conditions: image evolution and threshold noise strongly influence the occurrence of erratic (i.e., outlier) motion estimates. Figure 4(a) illustrates the nature of the variability. (Particular to the case in Figure 4 are errors due to storms crossing the radar field of view and the presence of second trip weather). There are many options that could be explored to introduce (continuity) constraints during calculation of motion vectors. This is considered an area for future work. For the present analysis, it was necessary to require some form of smoothing, both spatially and temporally. The effect of these measures on the image of Figure 4(a) are shown in Figure 4(b).

Three stages of post-processing have been incorporated to achieve smoothing of the displacement measurements. The stages are applied to the CCT algorithm displacement-vector field in the following order.

2.4.1 Spatial Interpolation

In practice, the (CCT algorithm) displacement measurements always form an incomplete sampling of the two-dimensional correlation box grid (i.e., not all correlation boxes can be tracked). Missing measurement values are estimated by direct[4] interpolation using inverse-distance-squared weighting. Inverse-distance-squared weighting has been used commonly in the geophysical sciences with good results, particularly when data are sparse. The primary objections to using this form of interpolation (see Watson and Philip [13]) are not considered significant in the present application.

Except in some extended storm situations, the displacement measurements almost always do represent a sparse sampling of the correlation-box grid. Outlier values can be re-enforced (i.e., spread their influence) by the interpolation process. This is most serious when an outlier is spatially isolated. If isolated, the interpolator essentially replicates the bad value in bordering windows, and the effect will be to pass the outlier value through the subsequent nonlinear spatial filter. The algorithm therefore relies on time-domain filtering to reduce the influence of these errors.

2.4.2 Spatial Smoothing

A nine-point median[5] filter is applied independently to north-south and east-west displacement components. Median filtering was selected for its ability to remove outlier measurements while minimizing spatial blurring. This works well when a suspect vector is neighbored by other displacement vector measurements. However, spatially isolated outliers are not handled well by this procedure as discussed above.

2.4.3 Time-Weighted Smoothing

Velocity Update: New Measurements. Recursive geometric time-weighted smoothing is performed component-by-component and independent of any spatial filtering. Time smoothing is essential for two reasons. First, it ensures that a sequence of track vectors exhibiting temporal

---

[4]By "direct" it is meant that the interpolation function takes the value of the sample points at the sample point locations.

[5]The displacement-vector set is not an ordered set; therefore, the median filter described is not a true median filter. In fact, the output of the filter as implemented here does not need to be, nor is it often, an element of the input set of vectors. Nevertheless, this method appears to yield better results than if an output vector is selected based on the median of an ordered product of the displacement-vector field.

continuity is produced. The previously mentioned measures for improving reliability and removing outlier measurements are only partially effective in this respect. Second, time-weighted averaging will alleviate the quantization error of reported storm velocities. A simple geometric time weighting with forgetting factor α is used:

$$\hat{v}_t = \left[\sum_{i=0}^{t} \alpha^i\right]^{-1} \sum_{i=0}^{t} \alpha^i \, v_{t-i}, \tag{1}$$

where v_{t-i} represents velocity measurement from the CCT algorithm at time $t-i$ and $\hat{v}_t$ represents the time-filtered output. This filtering method is simple to implement in real-time, and by selecting α less than 1, the algorithm can track accelerations. If measurement quantization is modeled as an (independent identically distributed) additive noise component with uniform distribution in the interval $[-q, q]$, then the quantization variance for the filtered estimate will be reduced by a multiplicative factor equal to

$$r(\alpha) = \left[\sum_{i=0}^{t} \alpha^i\right]^{-2} \sum_{i=0}^{t} \alpha^{2i}.$$

Figure 5 plots $r(\alpha)$ for selected α values, illustrating the claim that quantization error can be improved to a limit specified by the bandwidth for motion acceleration.

Velocity Update in the Absence of New Measurements. During the course of operation, gaps in the input product sequence or input products that do not generate displacement measurements are likely to occur. For these cases, it is necessary to have a model that will propagate the motion estimates forward in time until measurements resume. The filtering scheme of Equation (1) is therefore combined with a constant-velocity model resulting in the update formula

$$\hat{v}_t = \begin{cases} \hat{v}_{t-1} + \epsilon_t & \text{if there are measured displacements and} \\ \hat{v}_{t-1} & \text{otherwise,} \end{cases} \tag{2}$$

where

$$\epsilon_t = \left(\sum_{i=0}^{t} \alpha^i\right)^{-1} (v_t - \hat{v}_{t-1}).$$

2.5 Storm Extrapolation

The Extrapolated Storm Overlay (ESO) algorithm produces predictions by applying the filtered motion-vector estimates of the CCT algorithm to the storm object contours derived from

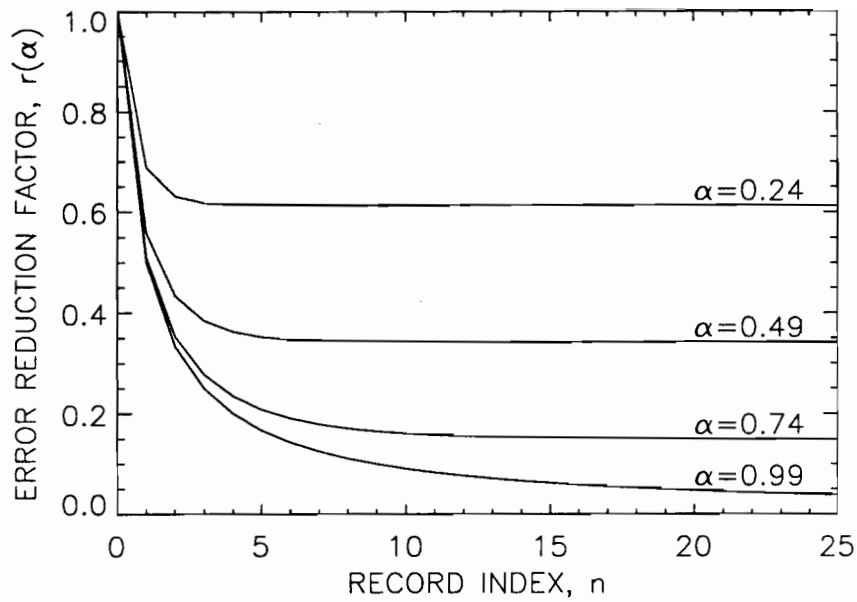


Figure 5. Reduction of quantization error with geometric time weighting. The reduction in error $r(\alpha)$ is plotted vs. record number n for three values of α: 0.24, 0.74, and 0.99.

the BSO algorithm output. Storm contours are extrapolated *en masse* in accordance with a storm velocity that is computed for the storm centroid. At present, extended level-2 storms are extrapolated as one element. (The algorithm does not at present use the velocity field to differentially extrapolate extended storm structures).

Ideally, the ESO algorithm should work in conjunction with user display software to provide a flexible tool that can be used to produce envelopes of predicted storm regions. An example display is depicted in Figure 2(d). Here, the predicted storm positions (shown by bounding contour lines) are presented as an overlay to the current PCP product display.

There are no current plans to implement an extrapolated contours product for ATC use. The ESO algorithm described here is currently used only for evaluation purposes, as in Section 4.

3. PERFORMANCE EVALUATION

Because the present focus regarding TDWR SMP is that of a tracking algorithm used to extrapolate storm location, evaluation of the algorithm logically focuses first on how well it tracks weather. Only as a secondary objective is an analysis of the "predictive" capabilities considered.

Tracking performance focuses on a scoring of the SMP motion-vector output against an equivalent[6] set of truth vectors: vectors derived from an off-line *a posteriori* analysis of the weather radar data. Scoring is founded on a qualitative measure of agreement (defined below), but quantitative characterizations will also be used to substantiate conclusions. The algorithm's predictive capabilities are evaluated by a subjective analysis using specific weather examples.

3.1 Track Vector Scoring: Prediction-Step Errors

Track vector evaluation is based on an analysis of vector-prediction error as follows. Assuming a constant-velocity model as expressed by Equation (2), the SMP output vectors (when scaled by selected time intervals) predict storm displacements. If the selected time intervals are multiples of the nominal sampling rate, n-step predictors of storm displacements—conditioned on the motion history up to and including the current-time measurement—are obtained from the algorithm. The truth vectors are estimates as well, but they are computed using data up to and including that obtained at the n-step sampling time. The truth vectors are viewed as the optimal n-step storm displacements and are used to gauge the predicted (SMP algorithm) displacements. A one-step prediction-error analysis is the basis for an evaluation of tracking performance.

3.2 Vector Characterization

Qualitatively, output motion vectors are classified into one of three categories. Motion vectors that agree (in a proximal sense that is described below) with their companion truth vectors are considered to be "accurate" (within the limits of the input-product resolution); they are referred to as "hits." Vectors that are not hits are "misses." Misses are classified into two categories: those that, in a general sense, conform to the truth-vector field—but do not satisfy the proximal criterion—and those that are of gross error. The class of "one-pixel-error" prediction vectors, defined in the following section, is used to distinguish between the two types of misses.

3.3 Classification of Hits: Proximal Agreement

The criterion for scoring a test vector is illustrated in Figure 6. As discussed in Section 2.2.1, each vector has an associated intrinsic resolution. In Figure 6, displacement vectors are illustrated

[6]By an equivalent set, it is meant that truth vectors are computed in one-to-one correspondence with an archived list of SMP output vectors.

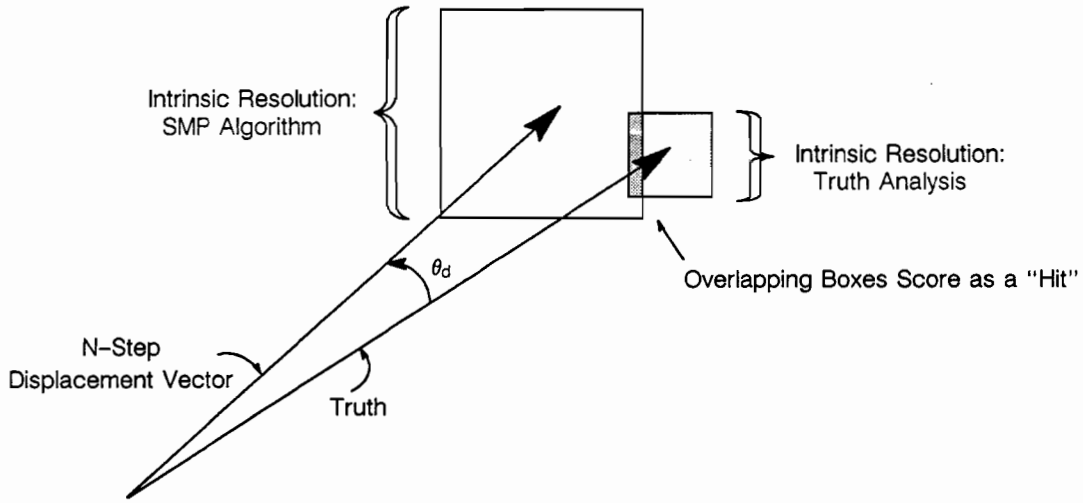


Figure 6. Scoring a test motion vector.

for a given prediction interval, and displacement-vector resolutions (determined by input image resolution and sampling interval) are indicated by bounding boxes for both test and truth vectors. The proximate scoring criterion accepts (scores as a hit) a test vector if there is any overlap between the bounding resolution boxes for test and companion truth vectors. Proximate-scoring results are summarized by Probability of Detection (POD) statistics. Vector agreement (or lack thereof) is further characterized by a quantitative analysis of directional, θ_d, and speed, ϵ_v, discrepancies between test and truth.

3.4 Classification of Misses: One-Pixel Errors

Although scored as a miss, vectors are deemed to conform to the truth-vector field if a quantitative analysis (see below) finds them to be of the one-pixel-error type. Figure 7 illustrates the class of one-pixel-error prediction vectors. A vector, with terminus equivalent to that of the truth vector, is allowed to vary ± 1 pixel (using the test resolution) in any direction. The eight vectors (illustrated with dashed lines) derived by such one-pixel shifts are referred to as one-pixel-error vectors. Maximum and minimum one-pixel-error directional and speed errors can be computed for a given truth speed as the truth vector direction varies from 0 to 180 degrees.

3.5 Classification of Misses: Gross Errors

Vectors that score as a miss and that are not of the one-pixel-error class are "gross" errors. These errors are considered unacceptable as output at any time during operation.

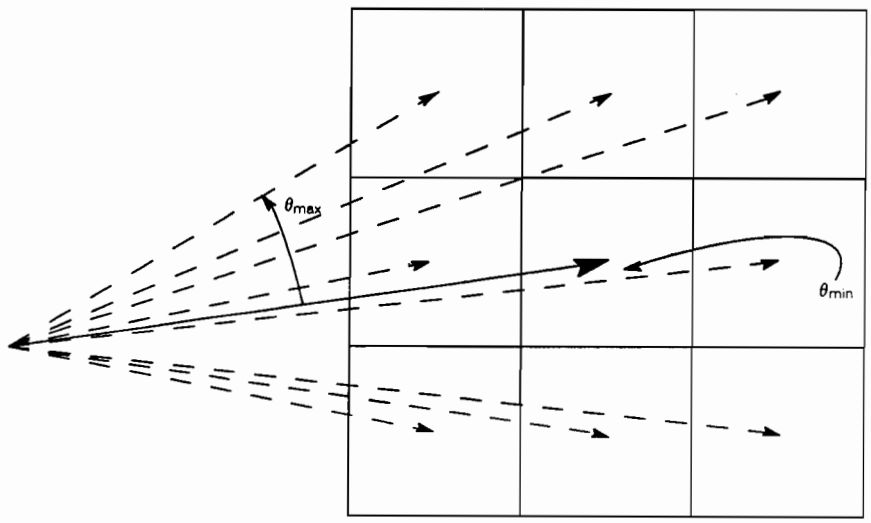


Figure 7. The class of one-pixel-error prediction vectors.

3.6 Quantitative Characterization

Further quantitative characterization is obtained by considering directional and speed errors, θ_d and ϵ_v. In scoring for numerical accuracy, however, two important issues should be kept in mind when interpreting results. First, truth does not represent an altogether independent observation of the same data set. As described in the section below, truth is in part generated from an analysis of the same radar data (although at a higher resolution and using a layered product) by algorithms similar to those producing the test vectors. Second, the inherent resolution of the truth vectors places a lower limit on the confidence limits, and usefulness (in an absolute sense), of quantitative error analysis. Nevertheless, given that the truth data base satisfies the objective criteria of a human observer/editor (see below), a numerical characterization is useful for classifying the type and extent of deviations within the test data base.

3.7 Reference "Truth"

A data base of truth, for scoring the SMP algorithm, was obtained by combining an off-line[7] tracking algorithm with human-observer data-base editing. The major steps of truth generation and product scoring (for a one-step prediction example) are illustrated in Figure 8. The genesis of truth

[7]Here, off-line refers to a non real-time, floating-point version of the correlation tracking algorithm.

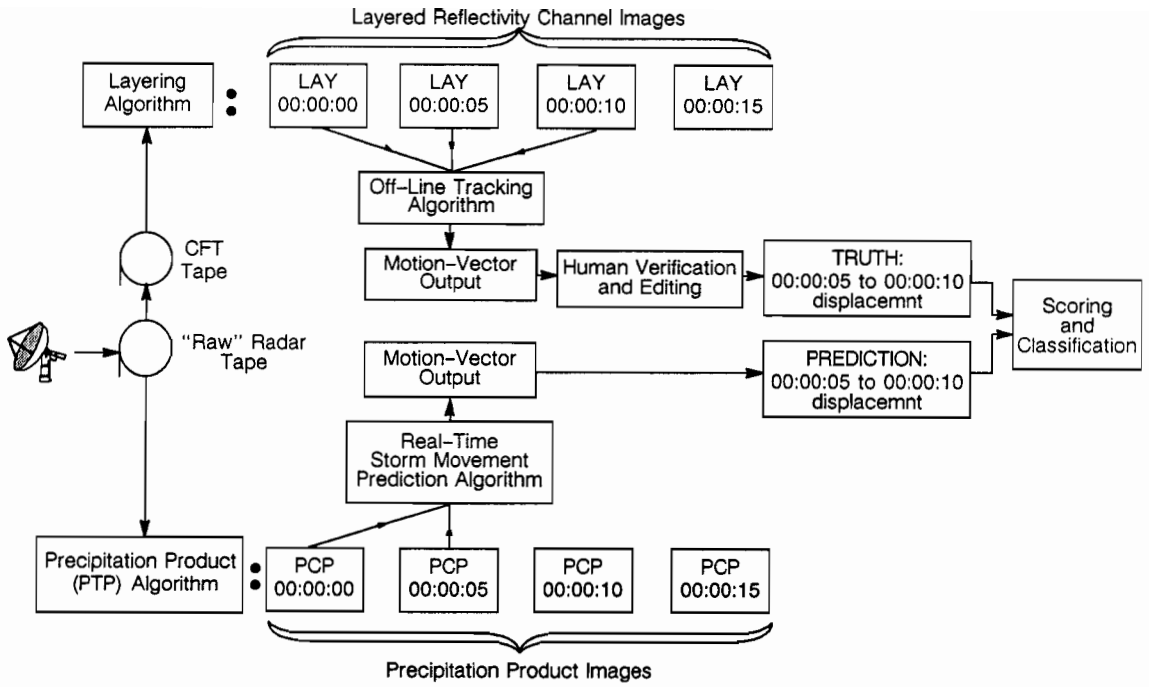


Figure 8. Truth genesis and scoring.

parallels that of the SMP motion vectors but with the following important differences. First, truth is based on the analysis of improved-quality image data: layered reflectivity images obtained using a finer (spatial) sampling resolution. Second, the off-line tracking algorithm used for truth genesis is a floating-point correlation algorithm providing improved discrimination of correlation maxima and localization of displacement measurements. Third, the off-line tracker is used to produce an *a posteriori* analysis of storm displacements. That is, in contrast to the SMP algorithm which provides predictions, the off-line tracker uses the correlation image processing to compute actual displacements (i.e., truth is obtained as the product of a smoothing algorithm). Last, the truth data base is subject to human verification/editing. This final step ensures that the truth data base provides, at the very least, a subjective portrayal of storm motion.

The assertion of "truth" is founded primarily on the observations that truth

1. is an *a posteriori* analysis of storm motion and that

2. human verification/editing ensures a subjective analysis consistent with the objectives of TDWR SMP.

The increased quantitative accuracy of the truth product injects meaning into a quantitative characterization of test vectors against truth.

4. RESULTS FROM 1988/89 FL-2 OPERATION

4.1 Introduction

Eighteen days from the 1988 FL-2 TDWR test-bed operation in Denver, Colorodo, and eight days from the 1989 operation in Kansas City, Kansas, were selected for study. In both cases, the operational dates[8] span a range of months covering the important spring and summer storm seasons of both regions. Denver storms tended to be smaller and slower moving in comparison to those recorded in Kansas City. Denver operations were also significantly influenced by nearby mountains. Hence, the two data sets offer somewhat complimentary evaluations.

The PCP product was not available during the evaluation period. Hence, raw radar data tapes were processed off-line to simulate real-time operation with emulation of PCP images. Tables 1 and 2 record the dates and times of operation, the number of SMP tilt records processed, and the number of motion-track vectors output for each day.

4.2 Quantitative Summary

4.2.1 Proximate Scoring Results

Tables 3 and 4 contain POD scoring results for the proximate scoring criterion discussed in Section 3.3. Of the total 6546 vectors in Denver, 5732 vectors agreed sufficiently with truth to be classified as hits, yielding an overall POD of 0.87. Fourteen of the eighteen days scored near or better than 90% agreement between the truth- and test-vector data bases. The exceptions—Apr24, Jun25, Aug21 and Sep10—are examined later in closer detail inasmuch as they underscore some algorithm limitations. In Kansas City, 9287 of 9931 vectors agreed with truth yielding an overall POD there of 0.93. POD scores below 90% were obtained for the days May14, May21, and Jun12, but there were no major departures as noted above for the Denver data set.

In both Tables, scoring statistics for an SMP algorithm without time-domain filtering are presented also (Denver POD = 0.46; Kansas City POD = 0.61). These poor results with unfiltered track vectors illustrates the (interim) need for processing the CCT algorithm output. For the remainder of this report, the focus will be on an SMP algorithm that includes post-processing (i.e., filtering) of the CCT algorithm vectors.

---

[8]As there are no overlapping days from the two locations days will be referenced solely by date. Tables 1 and 2 can be used to identify the site if needed.

TABLE 1

Summer 1988 (DEN) days selected for storm motion analysis

Date (1988)	Begin Time (Universal)	End Time (Universal)	Hours of Operation	Number of Tilts	Number of Vectors
Apr 24	18:23:50	20:24:05	2.00	23	38
May 10	18:13:24	20:26:59	2.23	26	107
May 18	19:30:22	00:14:44	4.74	55	513
Jun 9	18:24:06	00:52:44	6.48	78	906
Jun 10	21:08:14	23:52:54	2.74	33	120
Jun 11	23:13:01	01:43:54	2.51	30	164
Jun 15	19:31:26	23:54:43	4.39	49	297
Jun 21	18:50:30	00:26:31	5.60	68	566
Jun 22	21:20:30	00:05:11	2.75	29	399
Jun 25	18:28:25	23:17:55	4.83	58	959
Jun 29	22:36:15	23:46:34	1.17	15	51
Jul 4	18:05:31	00:59:16	6.90	84	664
Jul 7	17:34:56	01:22:56	7.80	94	550
Jul 11	18:56:49	00:01:33	5.08	61	167
Jul 16	18:05:12	00:54:42	6.83	83	476
Aug 12	18:09:35	00:50:36	6.68	81	242
Aug 21	18:08:30	01:09:57	7.02	85	255
Sep 10	21:32:28	23:35:45	2.05	25	72
Total	-	-	81.80	977	6546

TABLE 2

Summer 1989 (KC) days selected for storm motion analysis
(A "*" indicates non-continuous operation)

Date (1989)	Begin Time (Universal)	End Time (Universal)	Hours of Operation	Number of Tilts	Number of Vectors
May 14	18:18:35	21:17:02	2.97	25	212
May 21	14:51:52	18:27:01	3.59	58	2308
Jun 12	11:56:03	23:04:39	4.39*	53	215
Jun 24	18:32:38	02:01:57	6.49	83	622
Jun 26	20:13:36	02:35:12	6.36	77	2317
Jul 1	15:43:06	02:26:49	9.73	108	2380
Jul 10	20:05:30	02:51:54	5.77	82	1168
Jul 12	04:58:37	02:53:54	7.98*	112	709
Total	-	-	45.32	598	9931

4.2.2 Characterization of Hits

Figures 9 and 10 contain the results of a quantitative analysis (i.e., comparison against truth) of the accurate motion vectors (the "hits"). Histogram distributions of speed and directional errors are presented in Panels *a* and *b*, respectively. Scatter plots of speed and directional error versus truth speed are shown in Panels *c* and *d*, respectively. A scatter plot of speed error versus directional error is contained in Panel *e*. In general, both days shown similar results.

In each figure, Panel *a* shows that speed errors for hits are, as expected, sharply distributed and within the limits of a one-pixel excursion (i.e., ± 6.28 knots). In fact, the majority of errors measure less than that of a 1/2-pixel excursion (the bin-width resolution of each figure's histogram is 1.39 knots). Also, as evident from Panel *c*, the magnitude of a speed error has no apparent correlation with true storm speed.

The distribution of directional errors is more broad (Panel *b*) with a number of vectors deviating from truth by 30 degrees or more. However, large directional errors (> 30 degrees) are confined to very slow moving storms (0-1 pixel excursions $\leftrightarrow$ storm speed ≤ 6.28 knots) as can be seen in Panel *d*. In Panel *d* a significant, but not surprising, inverse relation between directional error and storm speed is clearly evident. Three curves illustrating the maximum, mean, and minimum one-pixel directional errors (see Section 3.2 and Figure 7) are included also in Panel *d*. For the most part, accurate track vectors cluster at or below the level representing a mean one-pixel directional error.

TABLE 3

Denver POD scores for motion vector proximity analysis

Date (1988)	Total Vectors	Avg. Vectors per Tilt	Speed (knots) Mean (Maximum)	Number of Hits (POD)	
				Filtered	Unfiltered
Apr 24	38	1.65	7.47 (10.93)	21 (0.55)	12 (0.32)
May 10	107	4.12	2.49 (4.69)	98 (0.92)	77 (0.72)
May 18	513	9.33	15.13 (18.22)	500 (0.97)	211 (0.41)
Jun 9	906	11.62	9.43 (13.15)	828 (0.91)	461 (0.51)
Jun 10	120	3.64	7.47 (12.23)	113 (0.94)	51 (0.43)
Jun 11	164	5.47	5.72 (8.41)	163 (0.99)	61 (0.37)
Jun 15	297	6.06	9.11 (12.88)	261 (0.88)	126 (0.42)
Jun 21	566	8.32	5.69 (8.31)	547 (0.97)	410 (0.72)
Jun 22	399	13.76	1.90 (10.17)	394 (0.99)	325 (0.81)
Jun 25	959	16.53	17.95 (24.12)	554 (0.58)	415 (0.43)
Jun 29	51	3.50	11.96 (13.95)	50 (0.98)	20 (0.39)
Jul 4	664	7.90	11.95 (17.27)	647 (0.97)	160 (0.24)
Jul 7	550	5.90	6.28 (10.24)	518 (0.94)	188 (0.34)
Jul 11	167	2.74	4.82 (9.61)	167 (1.00)	59 (0.35)
Jul 16	476	5.73	2.76 (6.08)	476 (1.00)	329 (0.69)
Aug 12	242	3.00	10.15 (15.19)	217 (0.90)	68 (0.28)
Aug 21	255	3.00	14.70 (26.58)	149 (0.58)	58 (0.23)
Sep 10	72	2.88	24.57 (25.86)	29 (0.40)	30 (0.42)
Total	6546	-	-	5732 (0.87)	3061 (0.47)

TABLE 4

Kansas City POD scores for motion vector proximity analysis

Date (1989)	Total Vectors	Avg. Vectors per Tilt	Speed (knots) Mean (Maximum)		Number of Hits (POD)			
					Filtered		Unfiltered	
May 14	212	9.52	6.58	(10.41)	177	(0.84)	NA	(-)
May 21	2308	39.79	23.75	(31.24)	1971	(0.85)	1314	(0.57)
Jun 12	215	4.06	17.22	(21.36)	186	(0.87)	134	(0.62)
Jun 24	622	7.49	16.68	(20.73)	564	(0.91)	297	(0.48)
Jun 26	2317	30.09	11.19	(29.10)	2194	(0.95)	1343	(0.58)
Jul 1	2380	22.04	4.65	(10.88)	2358	(0.99)	NA	(-)
Jul 10	1168	14.24	13.42	(16.81)	1155	(0.99)	969	(0.83)
Jul 12	709	6.33	6.42	(14.91)	682	(0.96)	454	(0.64)
Total	9931	-	-		9287	(0.93)	4511	(0.61)

In Panel *e*, there is no evidence of any unusual relation between speed and directional errors outside of the slight negative correlation required by the scoring criterion (i.e., the speed errors must be coupled with the directional errors and vice versa to pass the proximate criterion).

4.2.3 Characterization of Misses

The above characterization of the "accurate" vectors underscores the resolution limitations of the algorithm (and scoring procedure) and provides a useful reference for an analysis of the misses. Figures 11 and 12 contain an equivalent speed- and directional-error description for the misses. In Panel *a*, most of the speed errors are within the range corresponding to a one-pixel excursion, and virtually none exceed that of a two-pixel excursion. The distribution of directional errors for misses (Panel *b*) has a distribution similar to that for hits but with less of a skew toward zero directional error. The scatter plot distributions (Panels *c* and *d*) also are similar to their counterparts for hits: most errors are within the one-pixel error range (i.e., below the maximum directional error curve). The scatter plot in Panel *e* (speed error versus directional error) clearly shows that most misses border on being classified as hits and that the occurrences of gross errors were limited. This analysis shows that "hits" and "misses" form a continuum with relatively few examples of gross error which cannot be characterized within the resolution limitations of the tracking algorithm. Figures 13 and 14 illustrate a clear data trend that associates these gross errors with either the

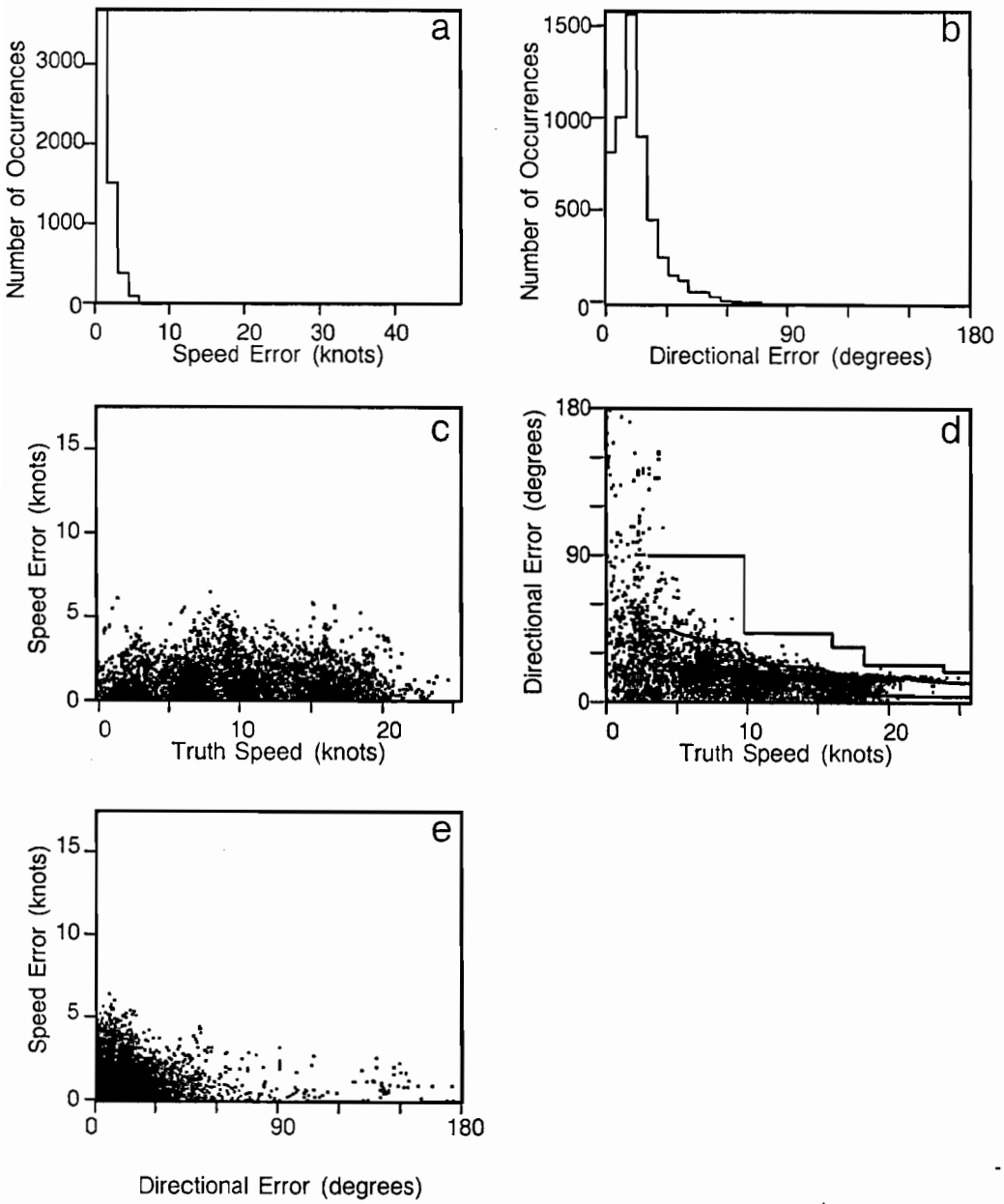


Figure 9. Characterization of hits: Denver. (a) Storm speed error distribution. (Bin width is 1.39 knots). (b) Directional error distribution. (Bin width is five degrees). (c) Scatter plot of speed error versus truth speed. (d) Scatter plot of directional error versus truth speed. Solid lines: maximum, mean, and minimum possible directional errors. (e) Scatter plot of speed error versus directional error.

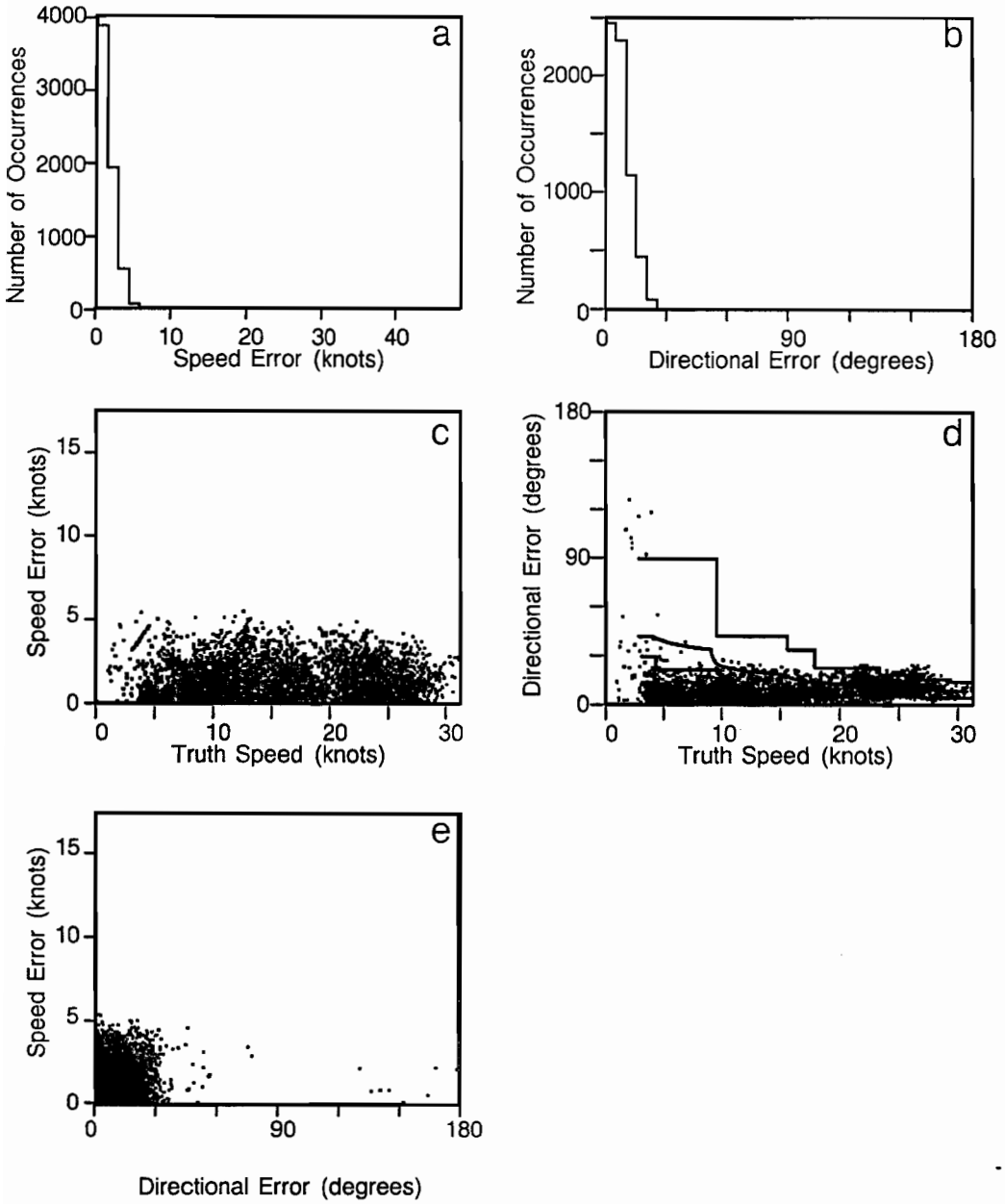


Figure 10. Characterization of hits: Kansas City. (a) Storm speed error distribution. (Bin width is 1.39 knots). (b) Directional error distribution. (Bin width is five degrees). (c) Scatter plot of speed error versus truth speed. (d) Scatter plot of directional error versus truth speed. Solid lines: maximum, mean, and minimum possible directional errors. (e) Scatter plot of speed error versus directional error.

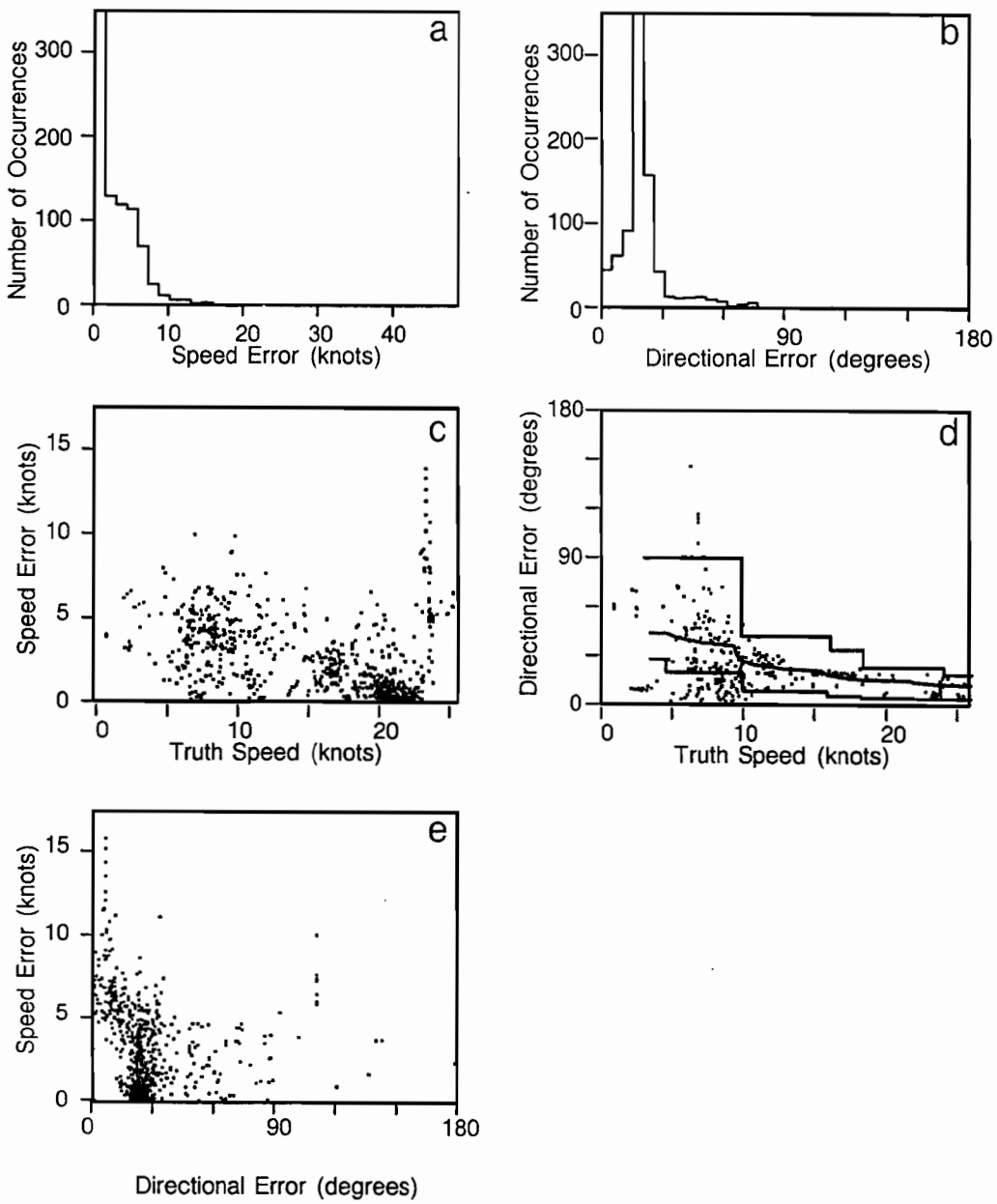


Figure 11. Characterization of misses: Denver. (a) Storm speed error distribution. (Bin width is 1.39 knots). (b) Directional error distribution. (Bin width is five degrees). (c) Scatter plot of speed error versus truth speed. (d) Scatter plot of directional error versus truth speed. Solid lines: maximum, mean, and minimum possible directional errors. (e) Scatter plot of speed error versus directional error.

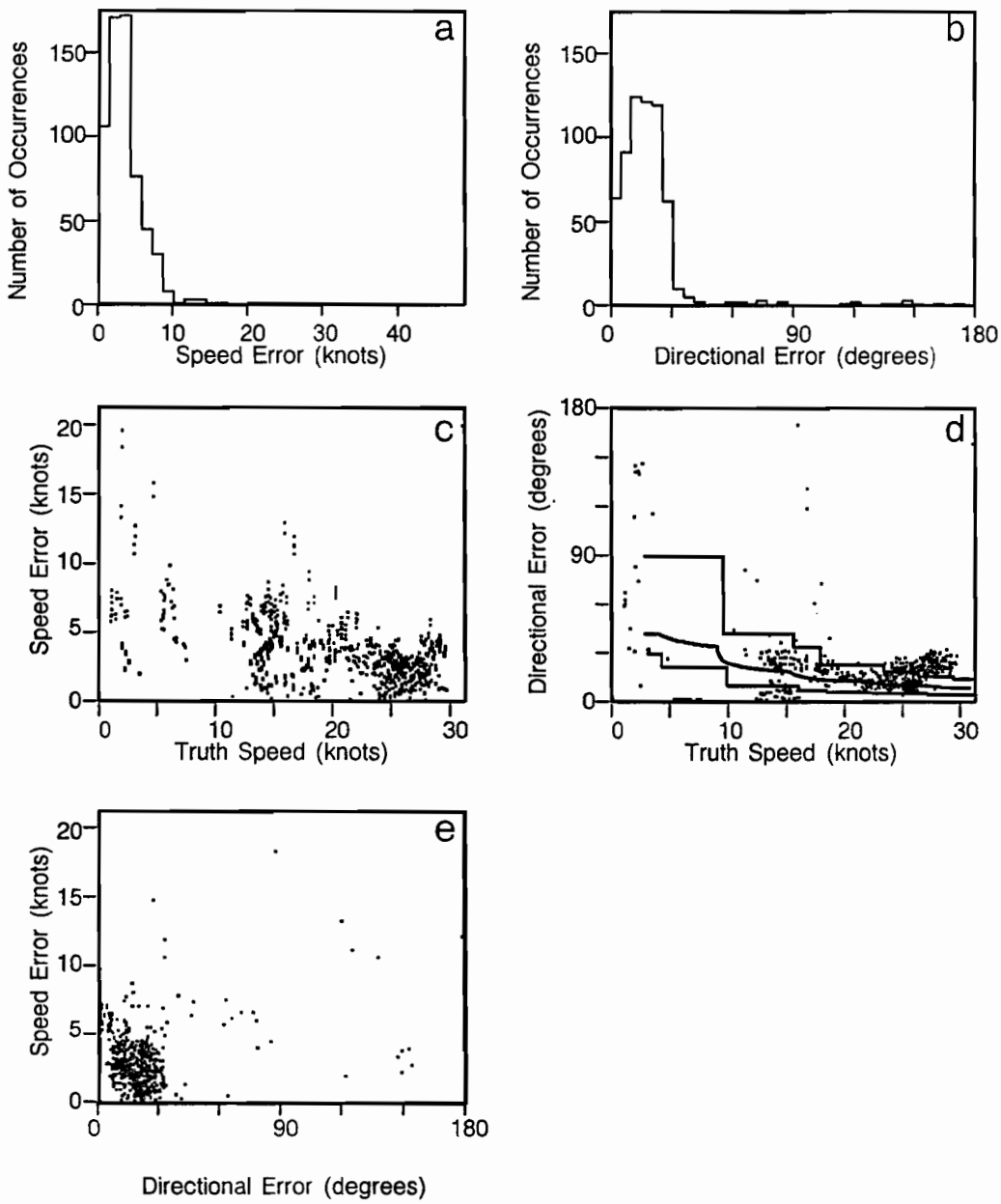


Figure 12. Characterization of misses: Kansas City. (a) Storm speed error distribution. (Bin width is 1.39 knots). (b) Directional error distribution. (Bin width is five degrees). (c) Scatter plot of speed error versus truth speed. (d) Scatter plot of directional error versus truth speed. Solid lines: maximum, mean, and minimum possible directional errors. (e) Scatter plot of speed error versus directional error.

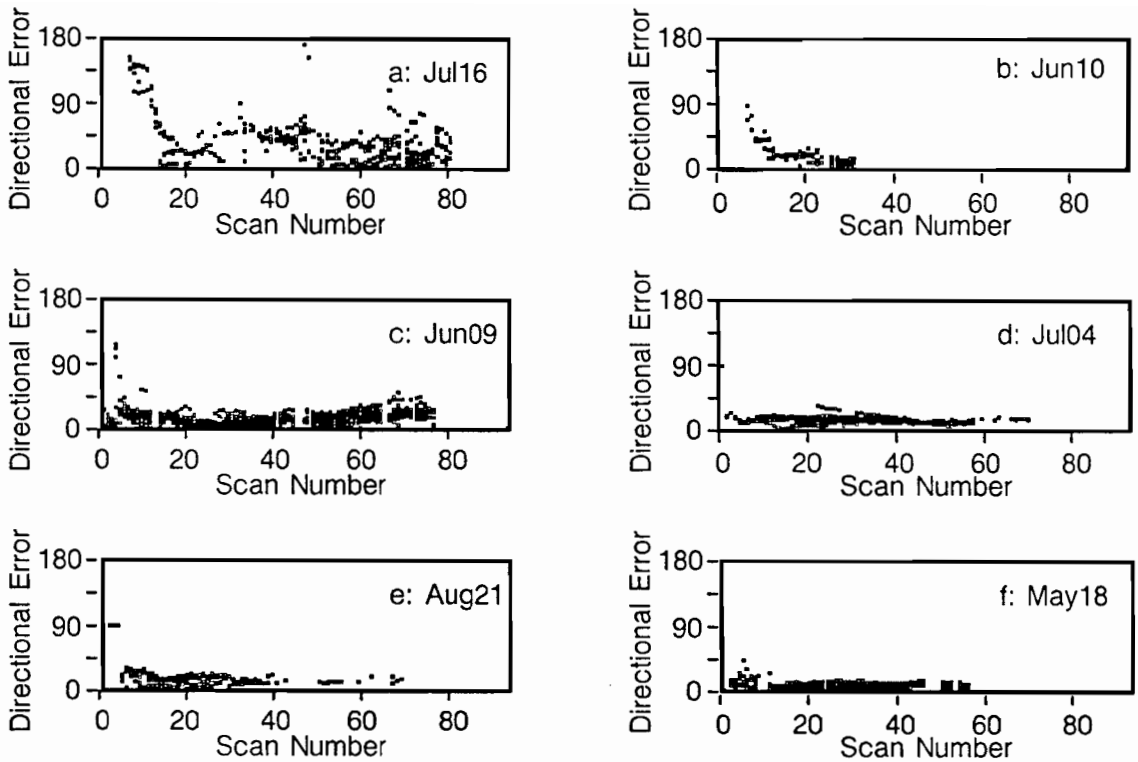


Figure 13. Directional error: effect of scan index and storm speed (Denver). Panels are arranged in order of increasing average storm speed. (a) Jul16, *2.8 knots (6.1 max). (b)* Jun10, *7.5 knots (12.2 max). (c)* Jun09, *9.4 knots (13.1 max). (d)* Jul04, *12.0 knots (17.3 max). (e)* Aug21, *14.7 knots (26.6 max). (f)* May18, *15.1 knots (18.2 max).*

initial-record segment of a day's operation or a day having very slow storm speeds. These figures also illustrate how directional error is modulated by storm speed.

In Figures 13 and 14, directional error for twelve days (six from each operational period) is plotted indexed by scan number. In each figure, the panels are arranged in order of increasing average storm speed (based on either the entire day's average, or a selected significant portion) beginning with the upper-left panel, proceeding down each column, and ending with the lower-right panel. In all twelve examples, the largest directional errors occur at the beginning of each day—the period before which time-domain smoothing becomes effective. Three exceptions, *Jul16* [Figure 13(a)], *Jul12* [Figure 14(a), first half of record], and *Jul01* [Figure 14(b), last third of record], exhibit gross directional error throughout. The performance for these three days is easily understood, though, after noting that the average storm speeds for *Jul16*, the first part of *Jul12*, and the last third of *Jul01* were 2.75, 2.65, and 2.42 knots respectively. Clearly, for storms moving below the resolution of the system it is unreasonable to expect good directional performance.

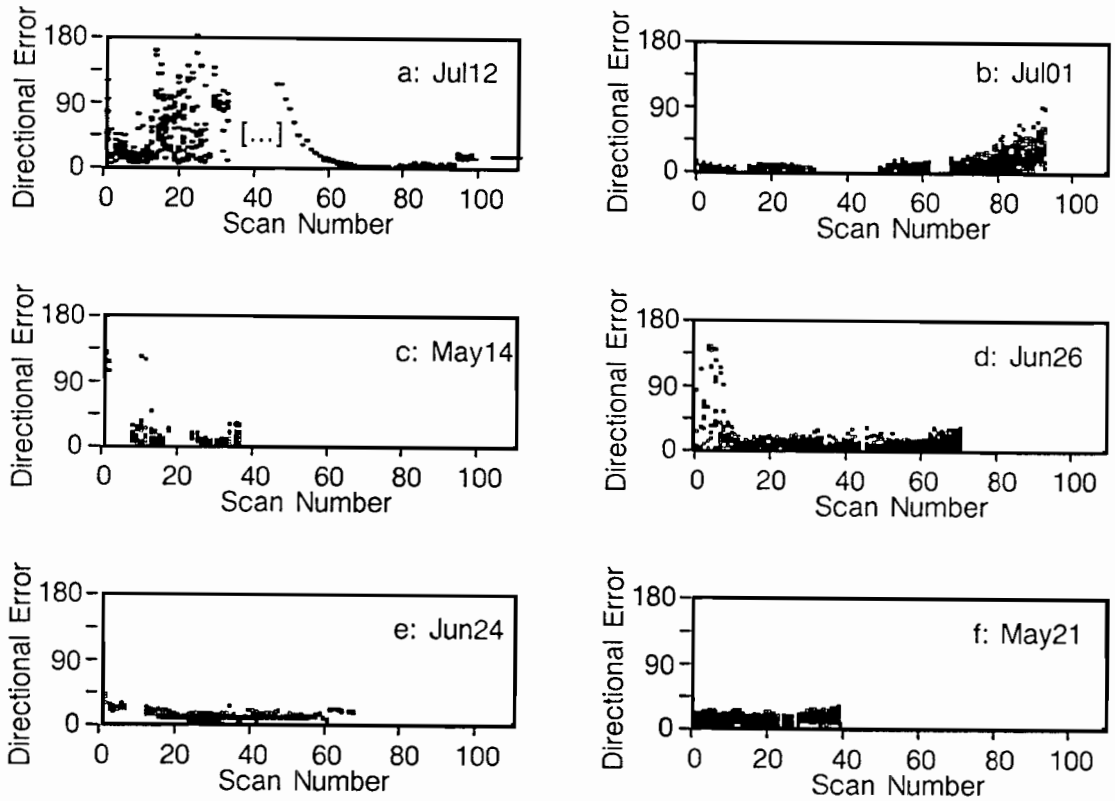


Figure 14. Directional error: effect of scan index and storm speed (Kansas City). Panels are arranged in order of increasing average storm speed. (a) Jul12, *6.4 knots (2.65 before break, 11.28 after, and 14.9 max overall). (b)* Jul01, *4.7 knots (8.64, 4.75, and 2.42 for 1st, 2nd, and 3rd segments, and 10.9 max overall). (c)* May14, *6.6 knots (10.4 max). (d)* Jun26, *11.2 knots (29.1 max). (e)* Jun24, *16.7 knots (20.7 max). (f)* May21, *23.8 knots (31.2 max).*

4.3 Performance and Meteorological Context

The previous section introduced one way in which algorithm performance is data dependent: performance was shown to improve with increasing storm speed. In this section, other correlations between algorithm performance and meteorological context are examined. Attention is focused on days that scored poorly (in overall POD) and days that illustrate specific algorithm limitations.

4.3.1 Denver.

Jun09. This day is noteworthy because significant storm activity was observed during nearly seven hours of continuous operation. Figure 15 contains a scan-by-scan record of tracker performance ("scan POD" versus scan) and selected precipitation profiles that highlight storm evolution (covering about a five hour period). Initially, isolated regions of convective activity appear west (Panels *a* and *b*) and track eastward at a moderately slow speed (9.43 knot average). Cells quickly develop cores above level-3 and level-4, and during this period the algorithm performs well (see scan POD record). Cells coalesce weakly to form a line structure (Panels *d* and *e*), and significant cell decay precedes the appearance of a preponderance of level-1 precipitation. (It is important to keep in mind that the algorithm does not "use" the level-1 weather for tracking). As areas of level-1 precipitation begin to dominate and as the areal distribution of level-1 regions corresponds less with active cell location and extent, the performance of the algorithm is seen to deteriorate (Panels *d*, *e*, and *f*). Misses appearing in the latter half of the record are all of the one-pixel class. Because there is the continued presence of cells with level-3 and level-4 activity throughout the recorded data, the tracker continues to perform reasonably well. This is in contrast to *Jun25* (see below) where few cells are present in the later scans, and algorithm performance deteriorates in a more precipitous fashion. The lower plot also illustrates the "storm size" of each contiguous region that is tracked within the image. As can be seen from that plot, deterioration in storm tracking is not correlated with a decrease in the amount (areal extent) of storm regions being tracked. Instead, this figure (and those that follow) clearly show that performance is less reliable during decay phases of storm evolution when there is a preponderance of (level-1) stratiform precipitation, which results in a "noisier" data environment for the tracker.

Jun25. The day *Jun25* is interesting because it is characterized by significant storm activity and a poor overall POD value. Roughly half of all recorded misses for the Denver data set are from *Jun25*. Figure 16 presents the tracker-performance (POD) record and selected precipitation profiles (covering about a four hour interval). Briefly, isolated convective activity begins to form to the east and north-east (Panel *a*), and all precipitation generally tracks in a WNW direction. As convective activity increases, a squall line forms containing level-3 and level-4 storm cells that continue to track WNW (Panel *b*). The line increases in size with some cells attaining level-5. In Panel *c*, a portion of the line begins to dissipate (area due north of radar) giving way to stratiform precipitation, but a major portion of the line is still intact.

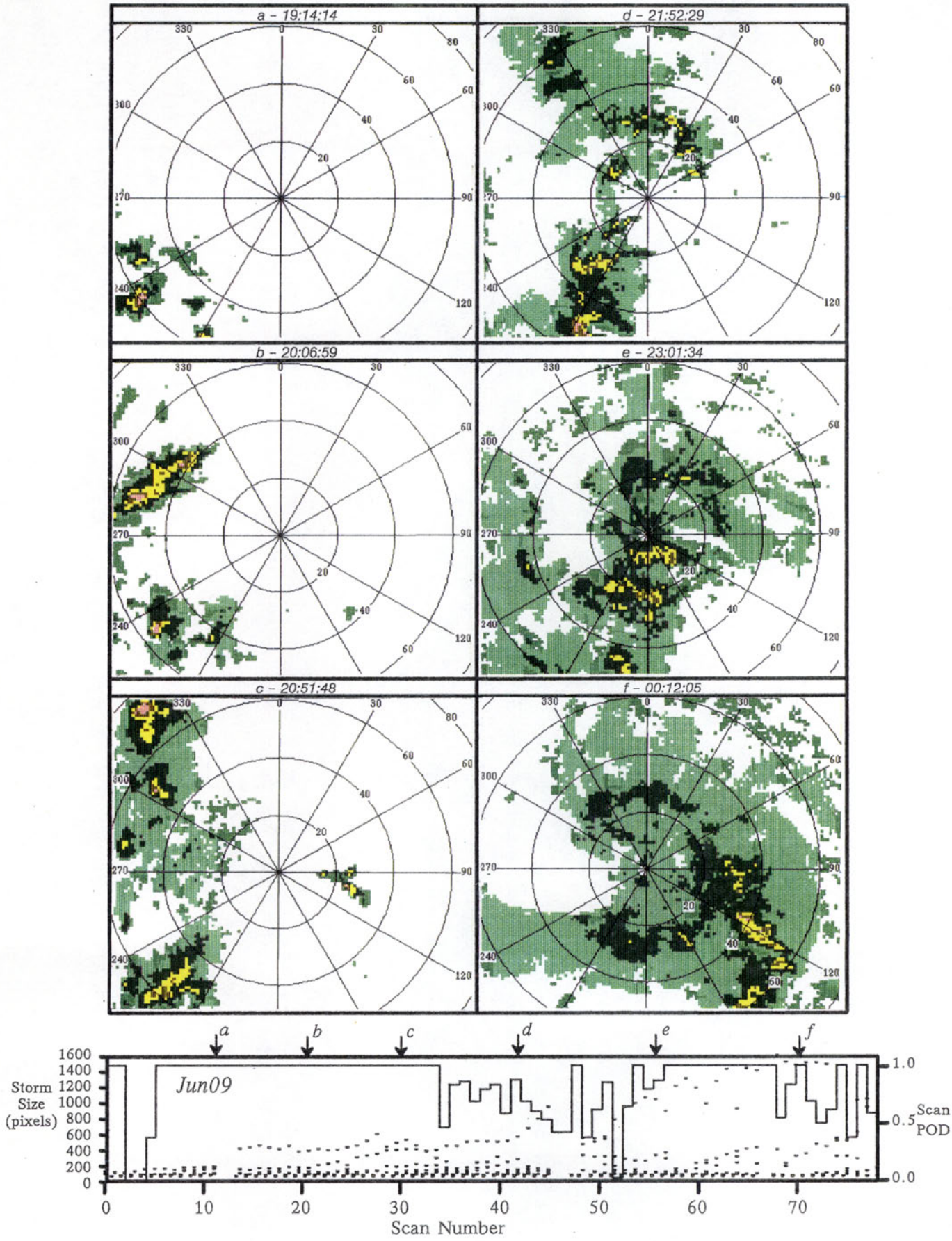


Figure 15. Tracking performance and storm evolution: Jun09.

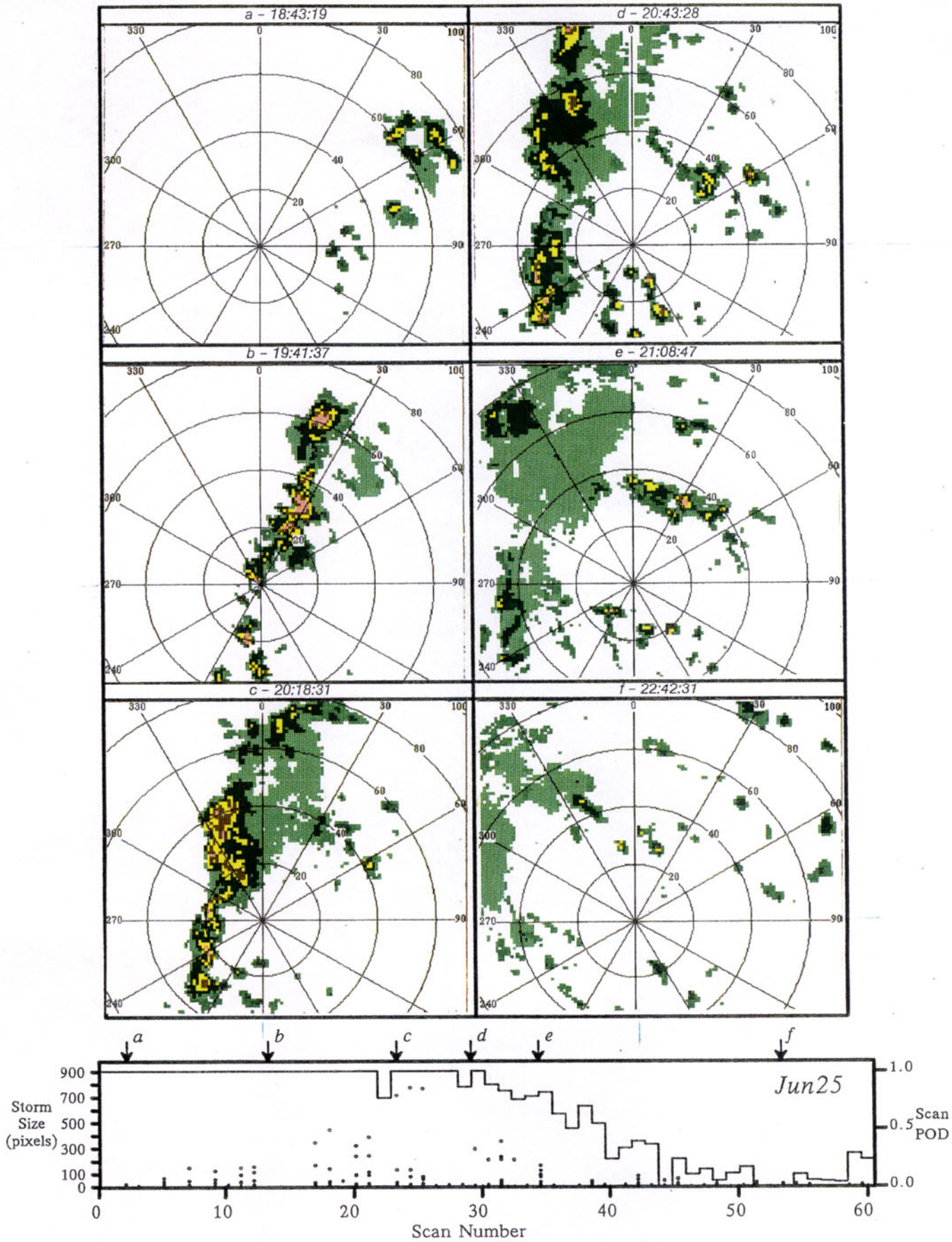


Figure 16. Tracking performance and storm evolution: Jun25.

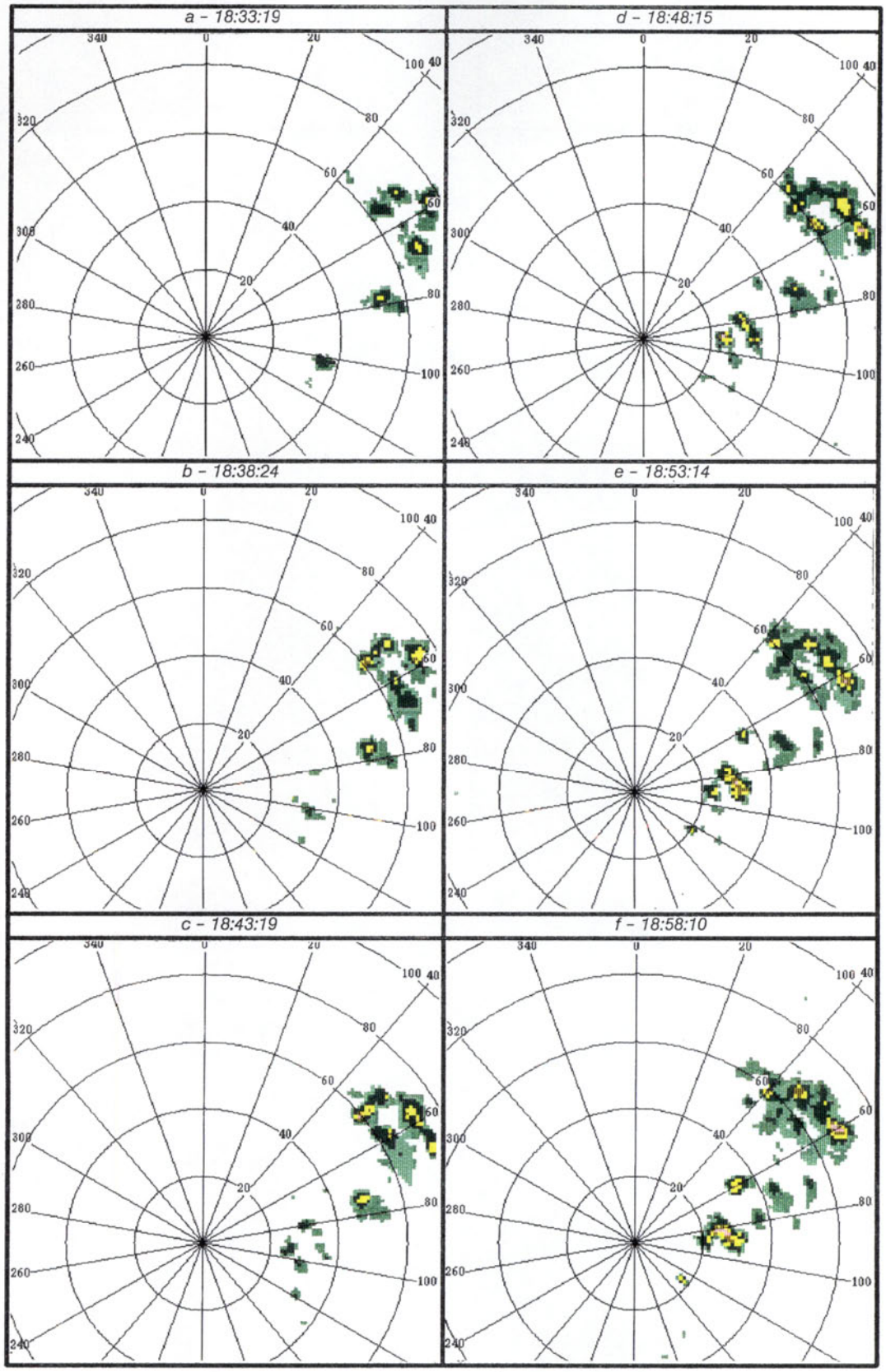


Figure 17. Tracking performance and storm evolution: Jun25, *early convective growth.*

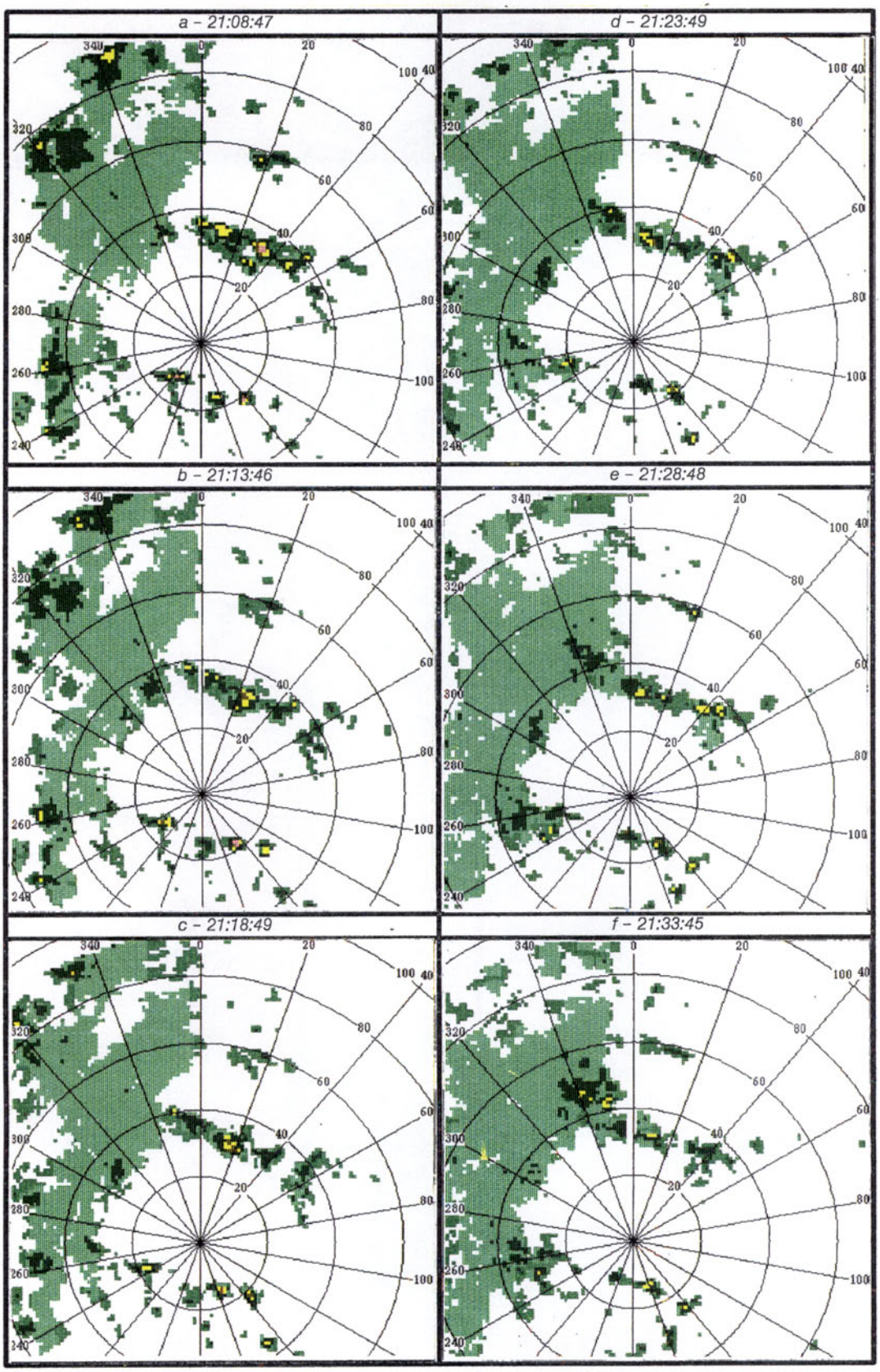


Figure 18. Tracking performance and storm evolution: Jun25, *later dissipative decay.*

Moving further west, the squall line begins to break apart although level-3 and level-4 cells are still apparent (Panel *d*). Also, isolated convective activity continues to form east of the 0-degree radial. By Panel *e* the squall line has given way to widespread stratiform precipitation although some convective activity continues to the east. Eventually, all activity dissipates to scattered precipitation at levels 1 and 2 only (Panel *f*).

The figure's scan-by-scan scoring summary shows a clear correspondence between dissipation of the squall-line and deterioration of tracker performance. Note that prior to the squall-line breakup, the algorithm performed with near 100% efficiency. Also, an analysis of the miss type showed that the misses for *Jun25* were clustered and of the one-pixel-error type. That is, the performance decline in the scan-by-scan scoring summary represents a degradation in the accuracy of the track vectors as opposed to the sudden occurrence of gross tracking errors (there where no gross errors for *Jun25*).

Level-2 (i.e., level-2 thresholded) storm size or the amount of "trackable" weather will clearly influence algorithm performance, but this does not appear sufficient to explain the decline in performance in Figure 16. Included with the scan-by-scan POD summary is a scatter plot of level-2 storm size versus scan. Although there is close agreement between storm size and POD value as the squall line dissipates, a similar correspondence between storm size and POD value is not manifest during the convective growth phase.

Figures 17 and 18 contain sequences (five minute interval sampling) of precipitation products corresponding, respectively, to the early convective growth phase and later dissipative phase of Figure 16. The most apparent difference is the appearance of widespread stratiform precipitation during the later storm phases. The implication for tracking is that there is a much greater variability in the above level-2 weather during storm dissipation because of the extensive presence of sub-threshold level-1 precipitation.

Jun15. The early part of this day (ref. Figure 19) was characterized by small isolated regions of convective activity that dissipated quickly, leaving behind only small regions of light precipitation. A few of the cells were large and/or contained significant level-3 and level-4 activity that resulted in satisfactory algorithm performance. Some smaller cells, however, that did not increase in size and that did not attain the higher precipitation levels (i.e., greater than level-2) contributed to a more variable algorithm performance in the first half of the day's activity (Panels *a*, *b* and *c*). Toward the middle of the day's data record, convective activity quickly developed into a squall line which the algorithm tracked satisfactorily (Panels *d* and *e*). In contrast to the days *Jun09* and *Jun25*, the decay of the squall line was quick and did not leave behind widespread level-1 precipitation (Panel *f*). Correspondingly, algorithm performance toward the end of the storm did not deteriorate at all during the recorded period (Panel *f*; compare scan POD).

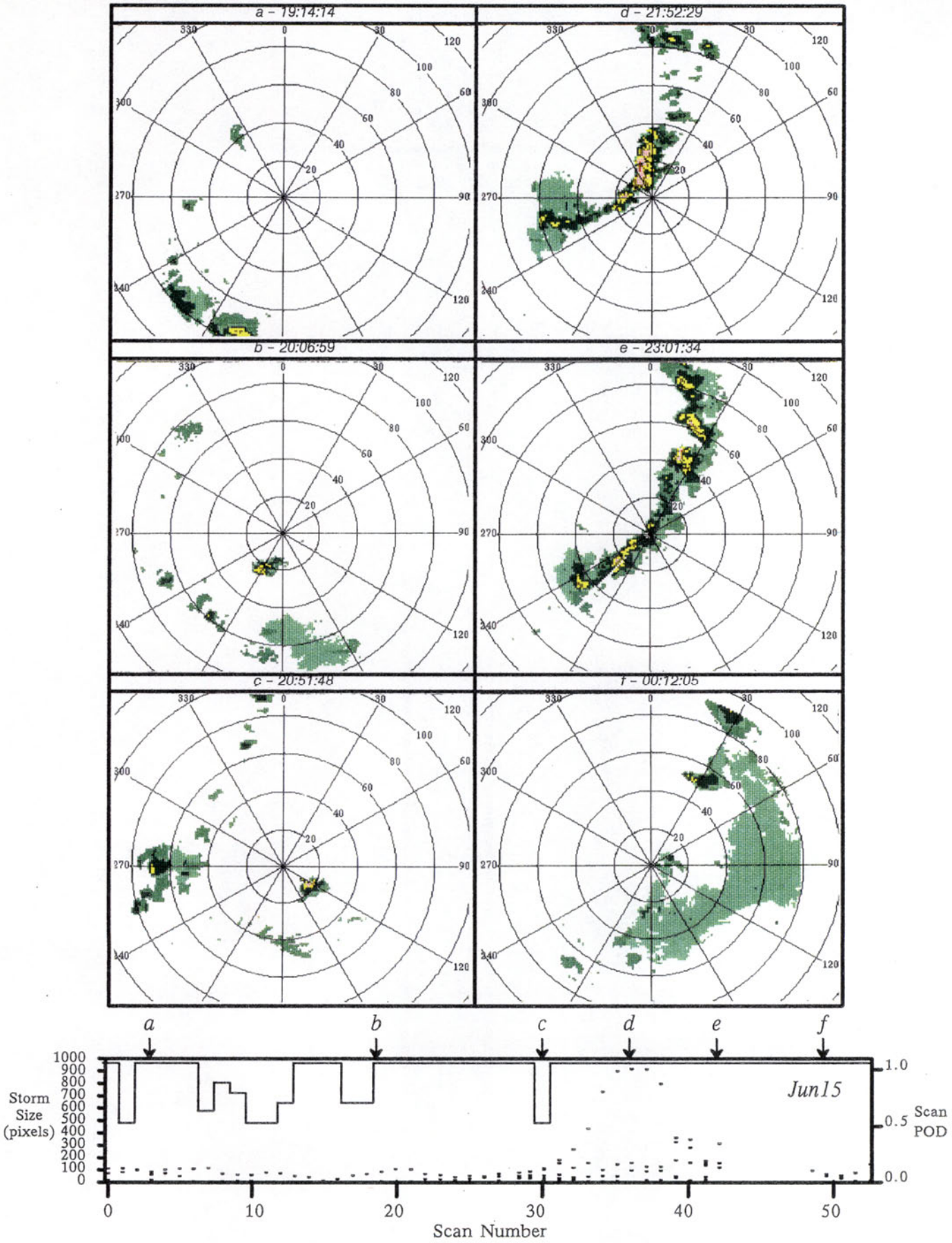
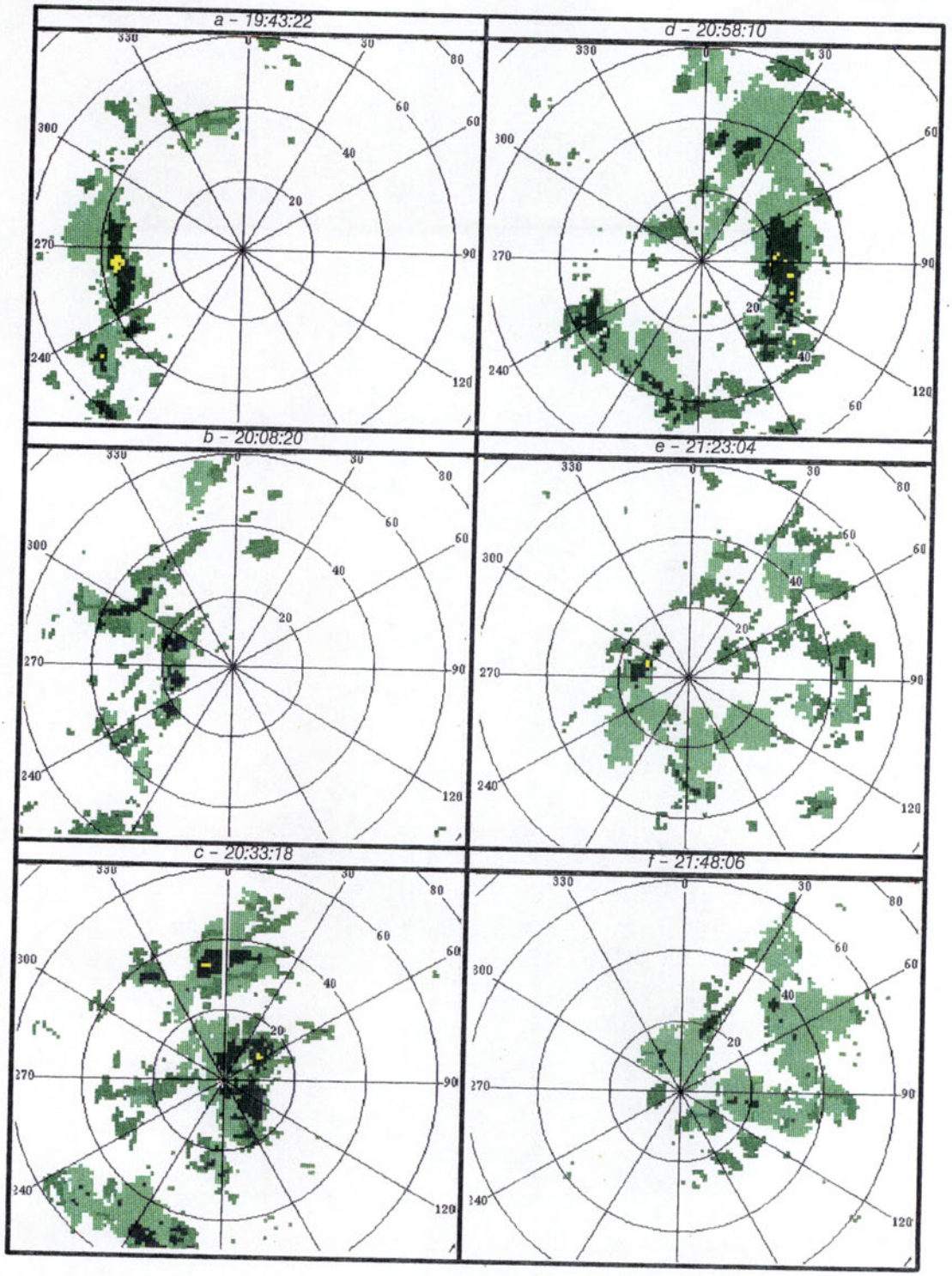


Figure 19. Tracking performance and storm evolution: Jun15.



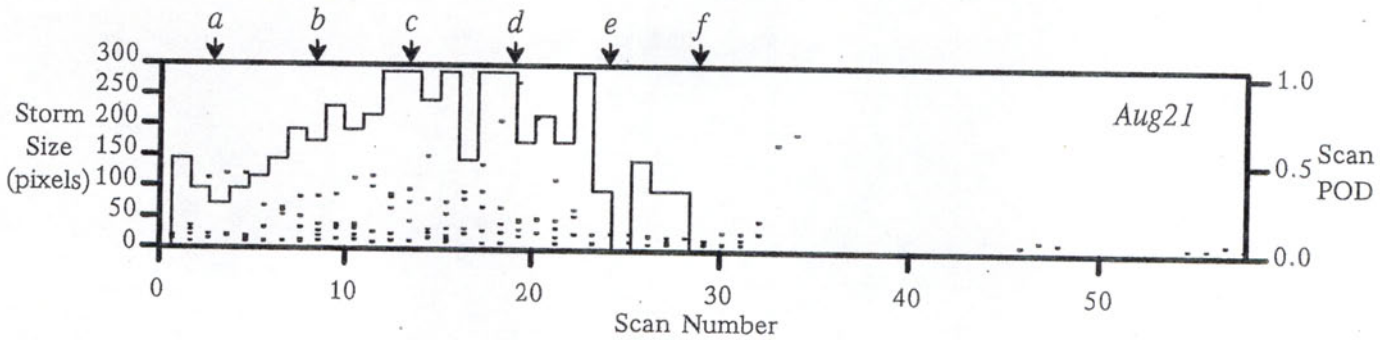


Figure 20. Tracking performance and storm evolution: Aug21.

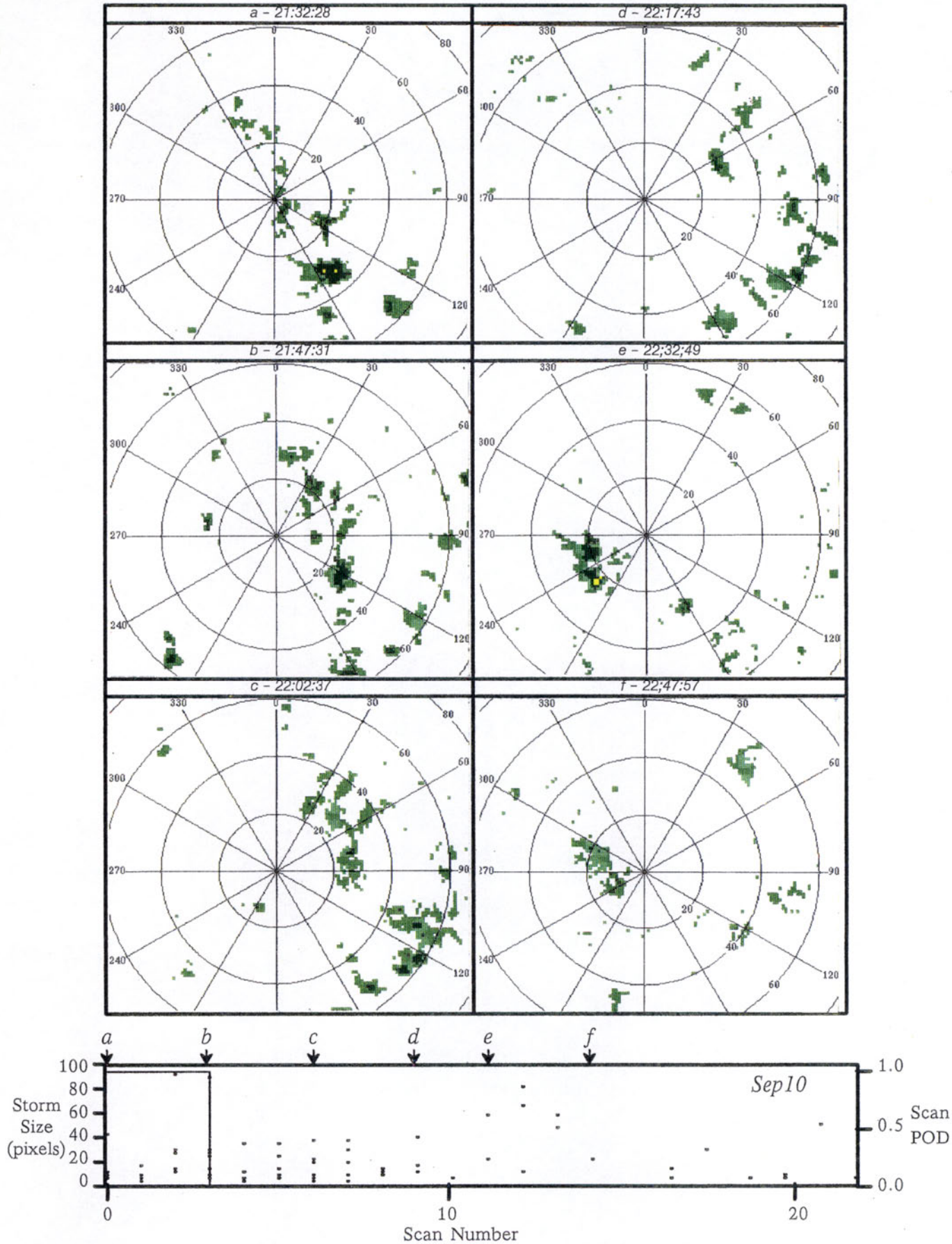


Figure 21. Tracking performance and storm evolution: Sep10.

Aug21. Storm activity for this day was characterized by only weak convective activity throughout the recorded period (Figure 20). Few cells can be observed at level-3 (or above), and cumulative scoring for the day is low at 0.58. Smaller level-2 storms embedded in regions of more widespread level-1 reflectivity contribute to a more variable tracker performance (all panels). Again, errors incurred during continuous operation of the tracker (i.e., after initial startup) are all of the one-pixel-error class. That is, the track vectors show general agreement with the truth vectors. There is a slight correlation between level-2 storm size and scan POD implying that level-2 (exclusive) storms of sufficient size can be adequately tracked.

Sep10 (and Apr24). The days *Sep10* and *Apr24* did not contain severe storm activity, but their analysis has been included for completeness, by showing that some of the recorded tracking errors are attributable to days with insignificant weather. Only the data of *Sep10* will be presented (Figure 21). These data are from short periods of system operation, and storm activity for each day only consisted of scattered weak convective activity with virtually no cell reaching level-3 or higher. Furthermore, level-2 storm size was low for all storms during the period of observation (see storm size vs. scan plot). (The data for *Apr24* was contaminated, in addition, by the presence of severe clutter "breakthrough").

4.3.2 Kansas City.

May21. This day had one of the weaker squall lines observed, with reflectivity generally staying in the level-2 to level-3 range (Figure 23). The most erratic performance for this day occurs when stratiform precipitation is at its greatest extent, as can be seen by comparison of Panels *a-c* with Panels *d-f*.

Jun24. It was unfortunate that the strong storm shown in Figure 22(a) tracked only near the edge of the PCP map. During the time of Panels *a-c*, tracking that storm was difficult because of edge artifacts. To the correlation tracker, objects crossing an image edge can appear to move in a direction parallel to that edge. Without a previous history or other nearby storms to observe (see *Jul10*, below), there is no current way to resolve this ambiguity. As more of the storm enters the field of view, tracking improves (Panels *d* and *e*), although the proximity of this storm to the upper left (image) corner resulted in less than perfect scoring. A later storm, entering from the south east (Panel *f*), also introduced an edge artifact, but as the storm moved through the image center tracking performance improved.

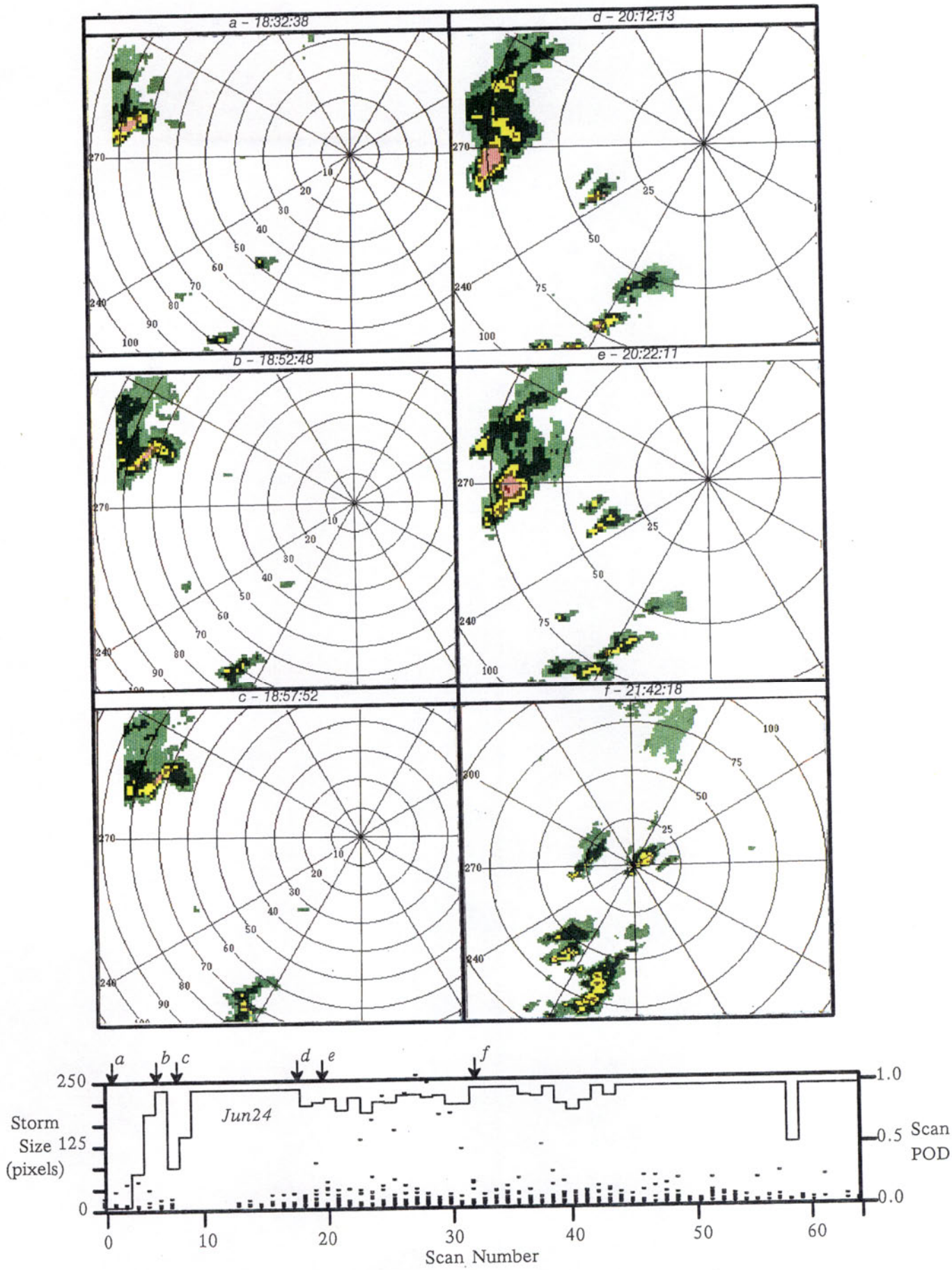


Figure 22. Tracking performance and storm evolution: Jun24.

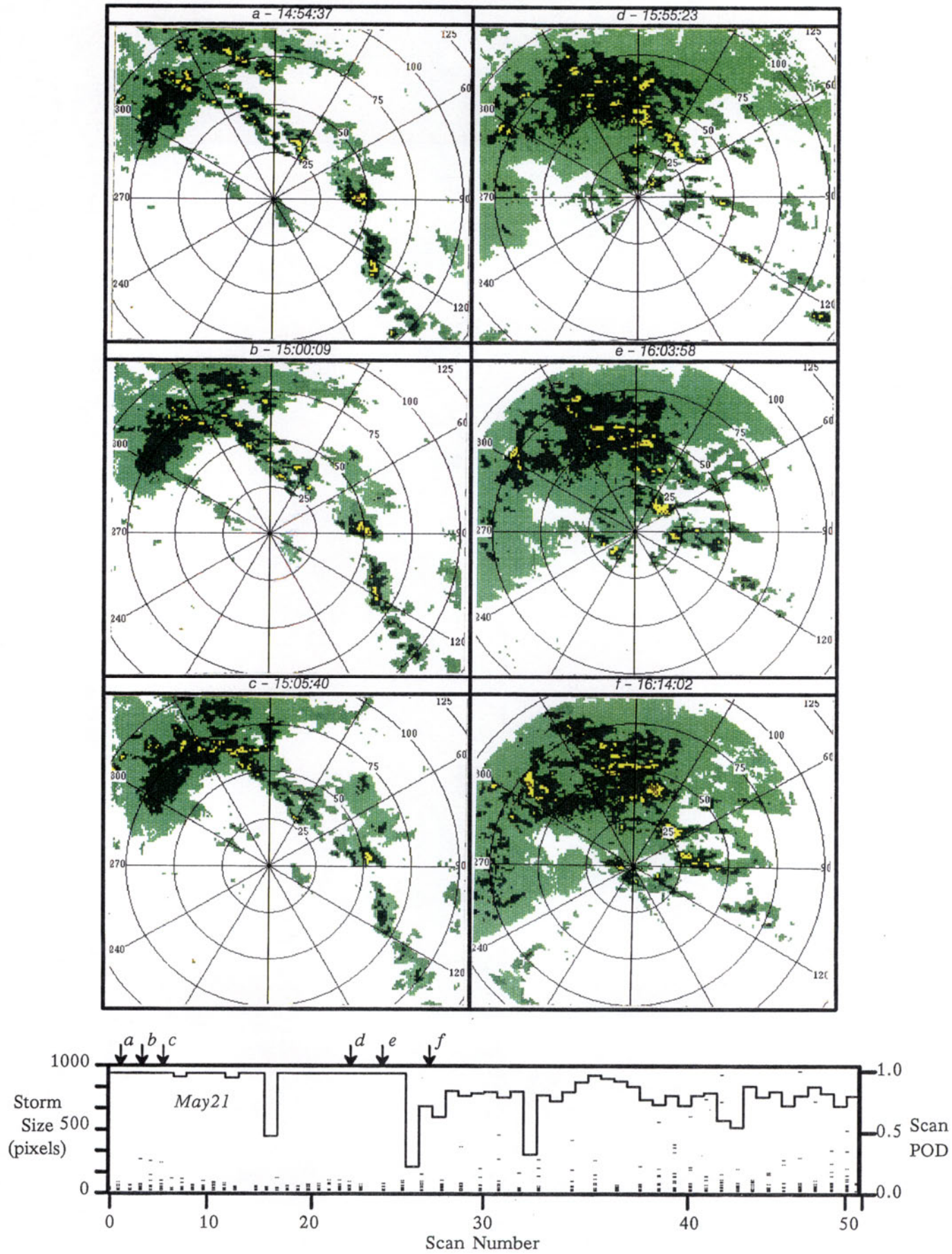
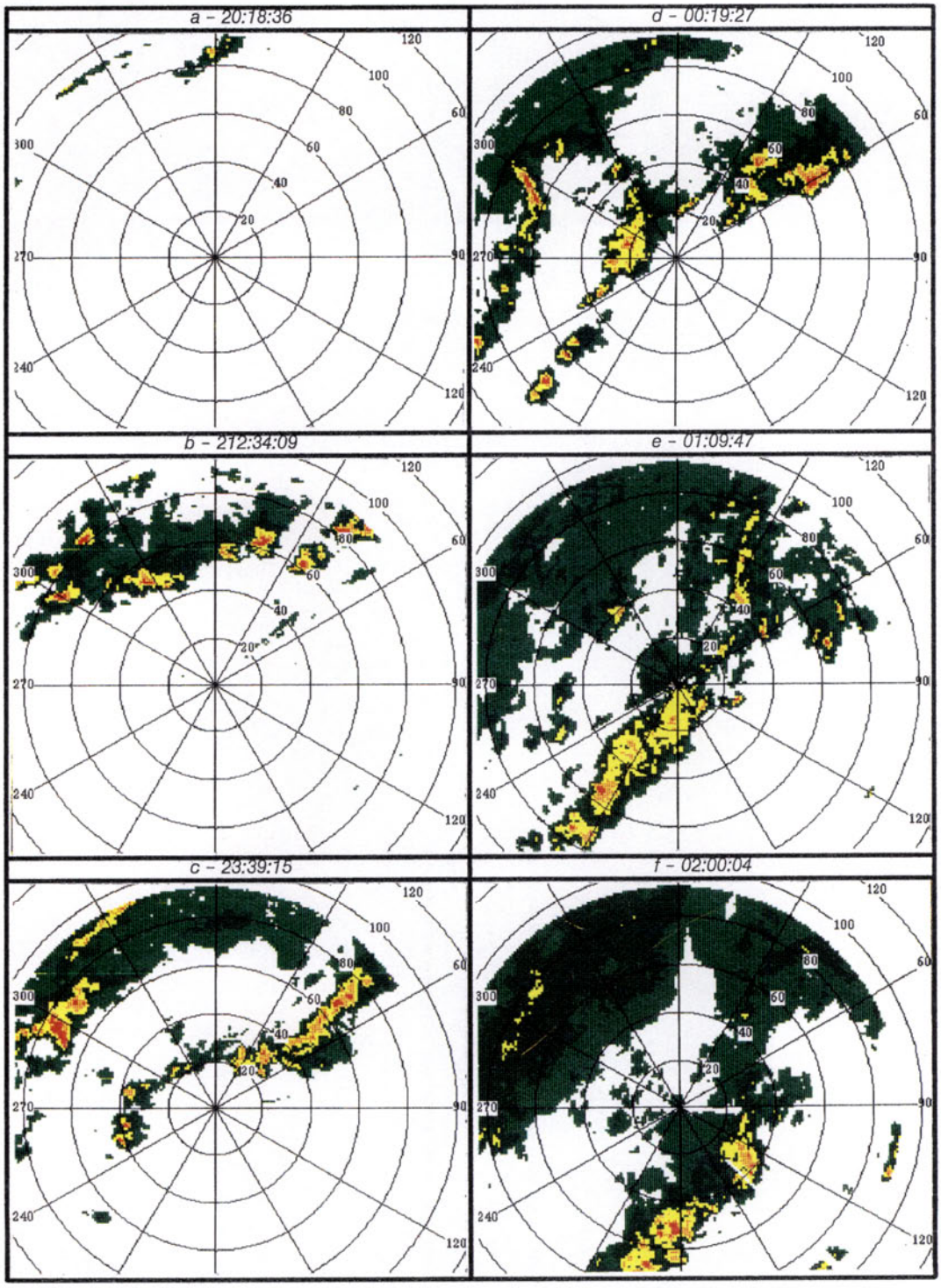


Figure 23. Tracking performance and storm evolution: May21.



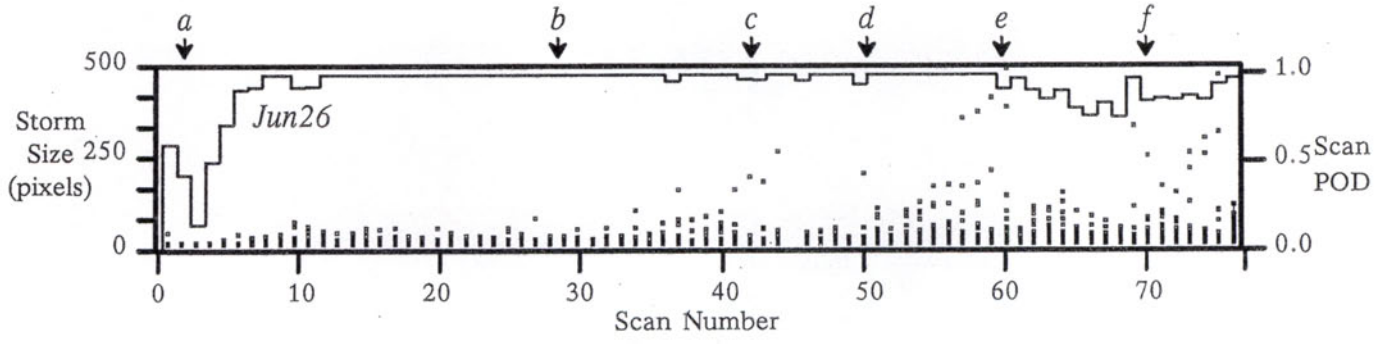


Figure 24. Tracking performance and storm evolution: Jun26.

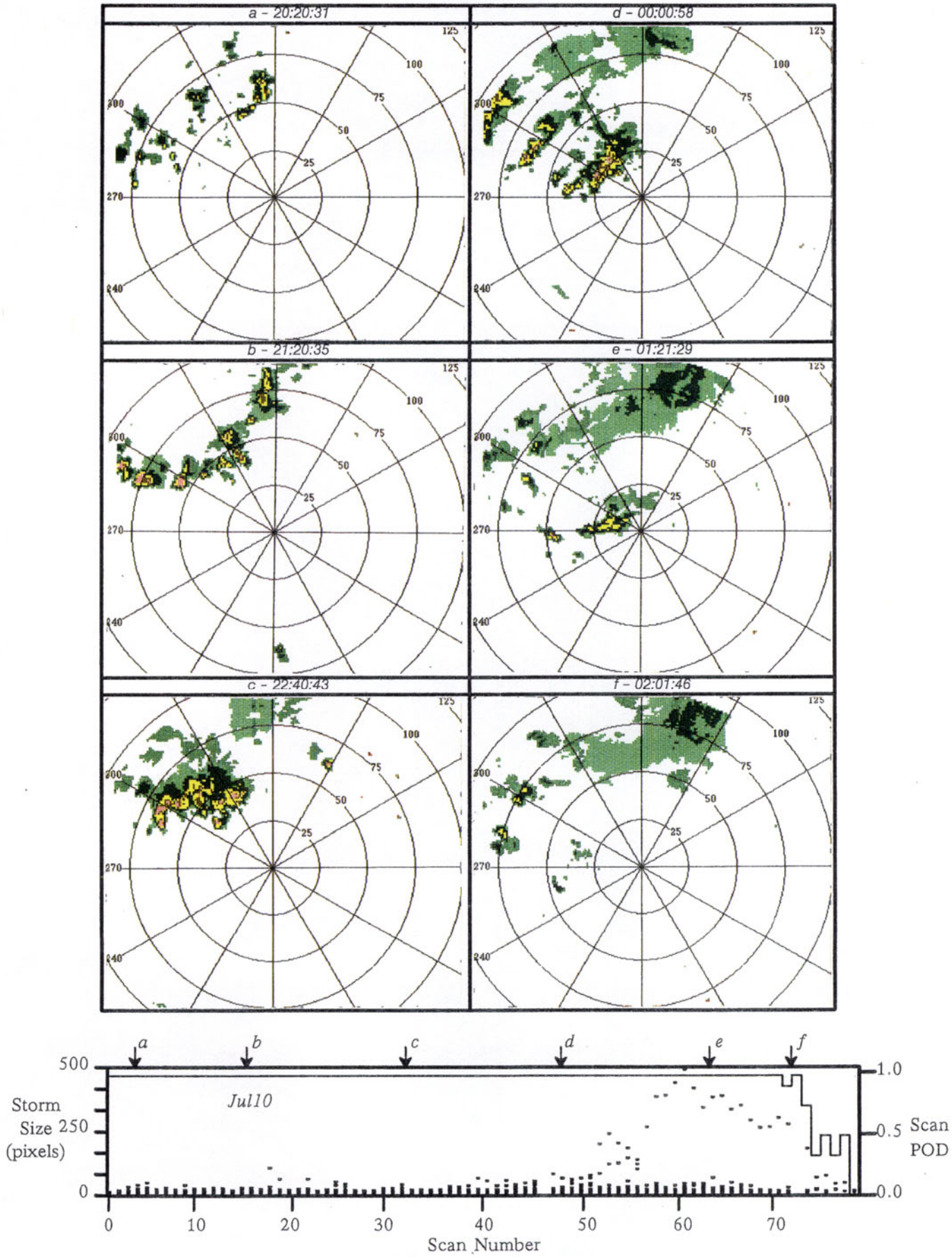


Figure 25. Tracking performance and storm evolution: Jul10.

Jun26. This day (Figure 24) had multiple squall lines and contained much convective activity. Initial misses at startup were due to the effect of storms crossing the image edge (Panel *a*). Generally, individual cells tracked in a north-east direction, although the formation of new cells (to the south east) resulted in apparent storm motion to the south east (compare Panels *d*, *e*, and *f*). To an extent, these panels follow the conceptual growth model of Figure 1(b). Some tracking error occurred as a result of this cell growth veering from the steering wind direction. (The ability to distinguish these two "types" of motion is examined again in Section 4.4). During periods of high convective growth, overall tracking performance was good (Panels *b-d*). As the amount of stratiform precipitation increased, some performance deterioration occurred (Panels *e* and *f*).

Jul10. In contrast to some of the above Kansas City days, convective activity began this day (Figure 25) within the field of view (i.e., the algorithm's first view of weather did not require an attempt to track across an image edge, Panel *a*). Tracking the developing line and, later, more mature cells presented no problems (Panels *b* and *c*). By the time of Panel *d* there were multiple lines. One line is clearly just entering into view (i.e., crossing the edge). This line, in an isolated situation, would have caused difficulties for the tracker, as with *Jun24* above. However, because this particular example contained two additional lines within the image interior and there was a history of their tracking, enough information existed such that the algorithm could handle the edge crossing problem. (It was also fortuitous in this case that motion was roughly orthogonal to the image edge, also alleviating the artifact). In Panels *e* and *f* storms decay without leaving an extensive amount of stratiform precipitation, and, consequently, algorithm performance does not deteriorate until after Panel *f* when most of the significant storm activity has subsided.

4.4 Extrapolation Performance

This section presents a few examples where extrapolated storm contours (predictions) are compared to the weather (PCP maps) later observed. Clearly, the notion of extrapolating current storm envelopes to obtain longer term (30-60 minute) predictions is too simple. Many factors can affect perceived performance, and for such a literal interpretation of prediction as many bad examples as good can be found in the data. The examples presented here are an attempt to identify some of the more important issues that bear on the SMP product and its use for long-term planning.

4.4.1 Isolated Storms

Figure 26 (*May18*) can be considered one of the better examples of performance for this data base. The figure contains a sequence of PCP maps at intervals of 0, 15, 30, and 60 minutes. Left column images contain storm contours derived from the time-0 image. Contours at 15, 30, and 60 minutes are extrapolations assuming a constant velocity model and the velocity profile of time 0. The right column repeats the presentation of the PCP maps with the relevant motion vector estimates for each time. (Older vectors are kept in each illustration so that by time 60, the illustrated vector field is a composite of old vectors and new).

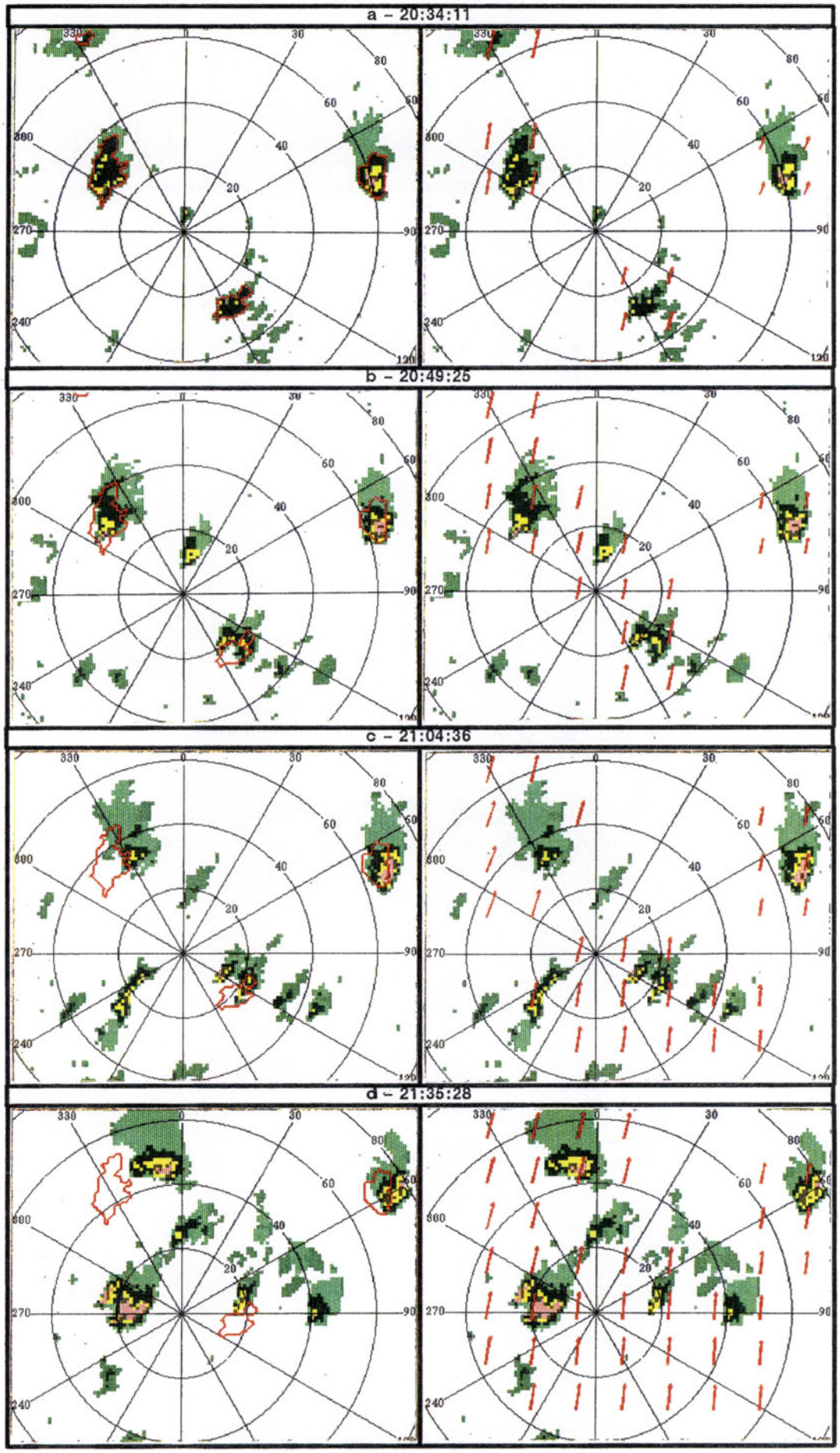


Figure 26. Extrapolation analysis: May18.

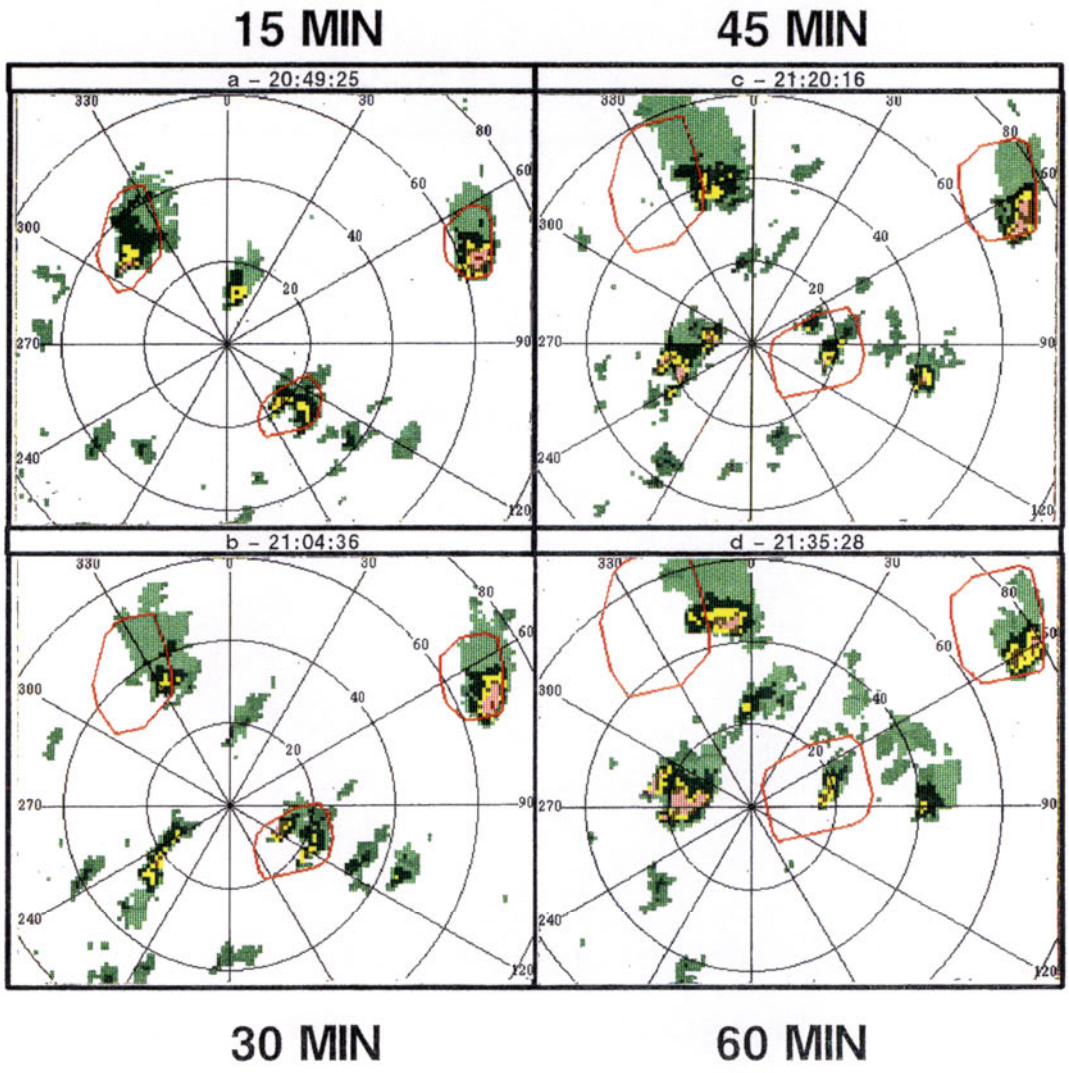


Figure 27. Extrapolation analysis: May18, *bounding error contours.*

In Figure 26, only three storms are identified at time zero. For 15 and 30 minute predictions, a close correspondence exists between extrapolated contours and moving storm areas. However, even at 15 minutes elapsed time there exist new regions of unaccounted storm activity. At 60 minutes elapsed time the correspondence between predicting contours and actual weather is slight. Of the three contours, two no longer overlap with any weather after 60 minutes. At 60 minutes, the unaccounted-for major storm due west of the radar is now clearly important.

The quick evolutionary nature of the PCP map is clearly evident in the sequence of Figure 26. The appearance and disappearance of new storm cells within 15-30 minutes (3-4 time frames) elapsed time clearly eclipses any hope of predicting the content of the 60 minute map from the content at time 0. A layered precipitation product might alleviate this to some extent.

The sequence of motion vectors (right column, Figure 26) indicates a steady (i.e., non-accelerating) motion, so that the vectors at time 0 continue to be representative of motion 60 minutes later (counter examples to this are presented below). The extrapolated contours track the mature storms albeit with a slight error bias, which naturally increases with extrapolation amount. As pointed out in Section 2.2.1, each motion vector is limited in its accuracy due to sampling resolution. If this uncertainty is accounted for when extrapolating contours, a representation as in Figure 27 results. In this figure, an error of ± 3.24 knots has been used to expand the contours during extrapolation. (To ease the implementation, convex hulls of the expanded contours are used). Now the extrapolated contours reflect the uncertainty in calculation of the motion vectors. When viewed in this context, two of the three storms are easily contained within the prediction contours out to 60 minutes. In both of these cases, the storm that was tracked was stable throughout the tracking period (mature with minimum nearby convective growth). The third system, located in the NW quadrant, does not appear to be successfully tracked. In contrast to the other two, this storm exhibited convective growth veering from the prevailing track direction. Evidence for this comes from examining the scan-by-scan images (not shown) and from the right column portion of Panel *c* in Figure 26. There, it can be seen that the local motion vector estimate has turned eastward (evidence of this turning is mitigated because time-domain filtering was in effect) in comparison to the vector estimates of Panels *a*, *b*, and *c*. Hence, the NW quadrant cell shown in panels dated 30 minutes and after is new, having replaced the one present in the original time-0 frame. This perturbation clearly appears as a non steady-state acceleration, which was not in evidence at time 0.

Figure 28 (*Jun24*) is a similar example taken from the Kansas City data. Again, the motion vector history is uniform in trend (no large accelerations) and the extrapolated contours follow storm translation well out to 30 minutes. If vector uncertainty is accounted for, then an analysis using expanded contours (as in Figure 27) shows good agreement out to 60 minutes. Major discrepancies result with new growth and a storm that enters from outside the field of view (SW quadrant at 60 minutes elapsed time). Also present in this example is an instance in which a storm has decayed completely, resulting in a "false alarm" prediction (NW quadrant).

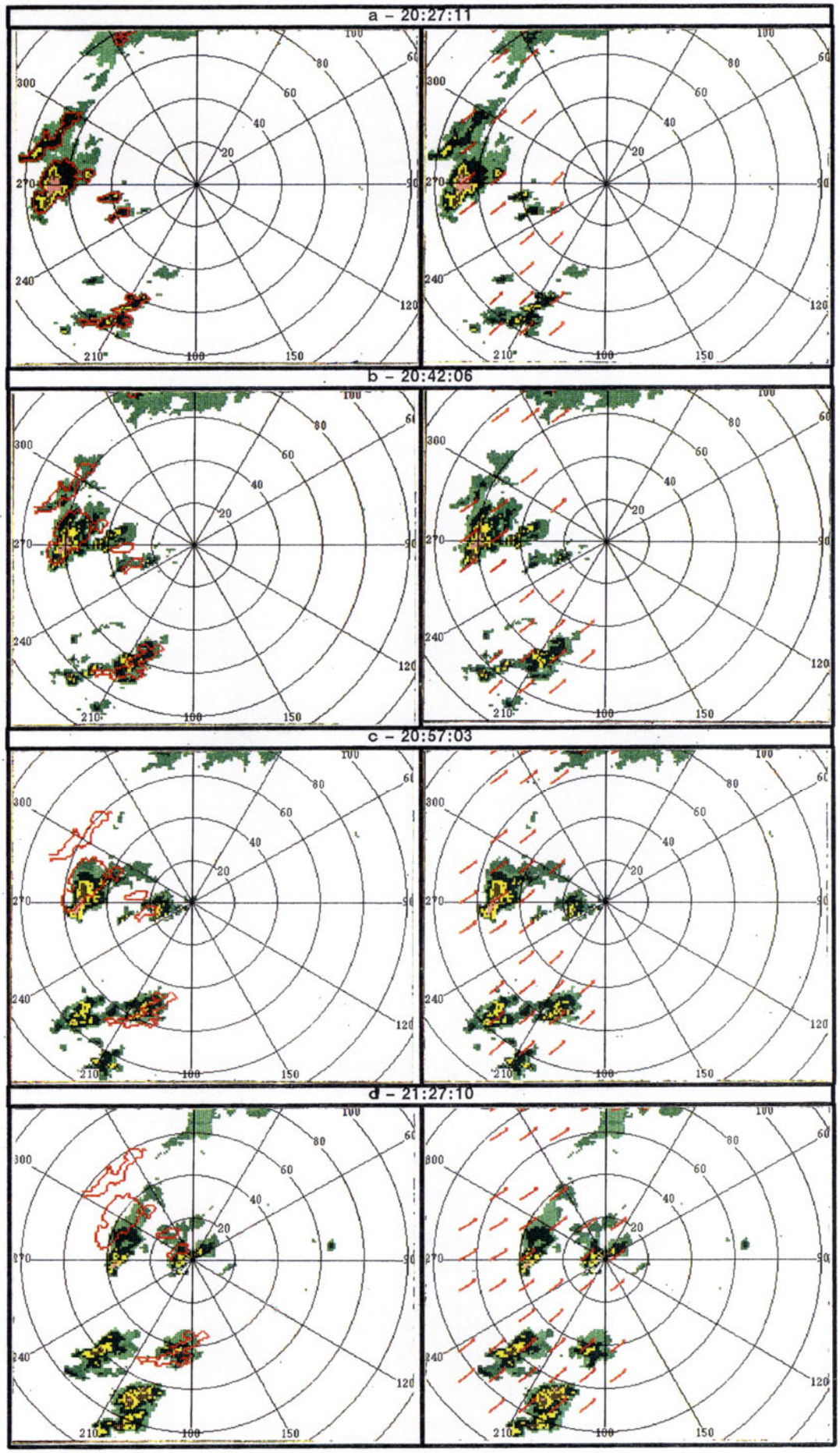


Figure 28. Extrapolation analysis: Jun24.

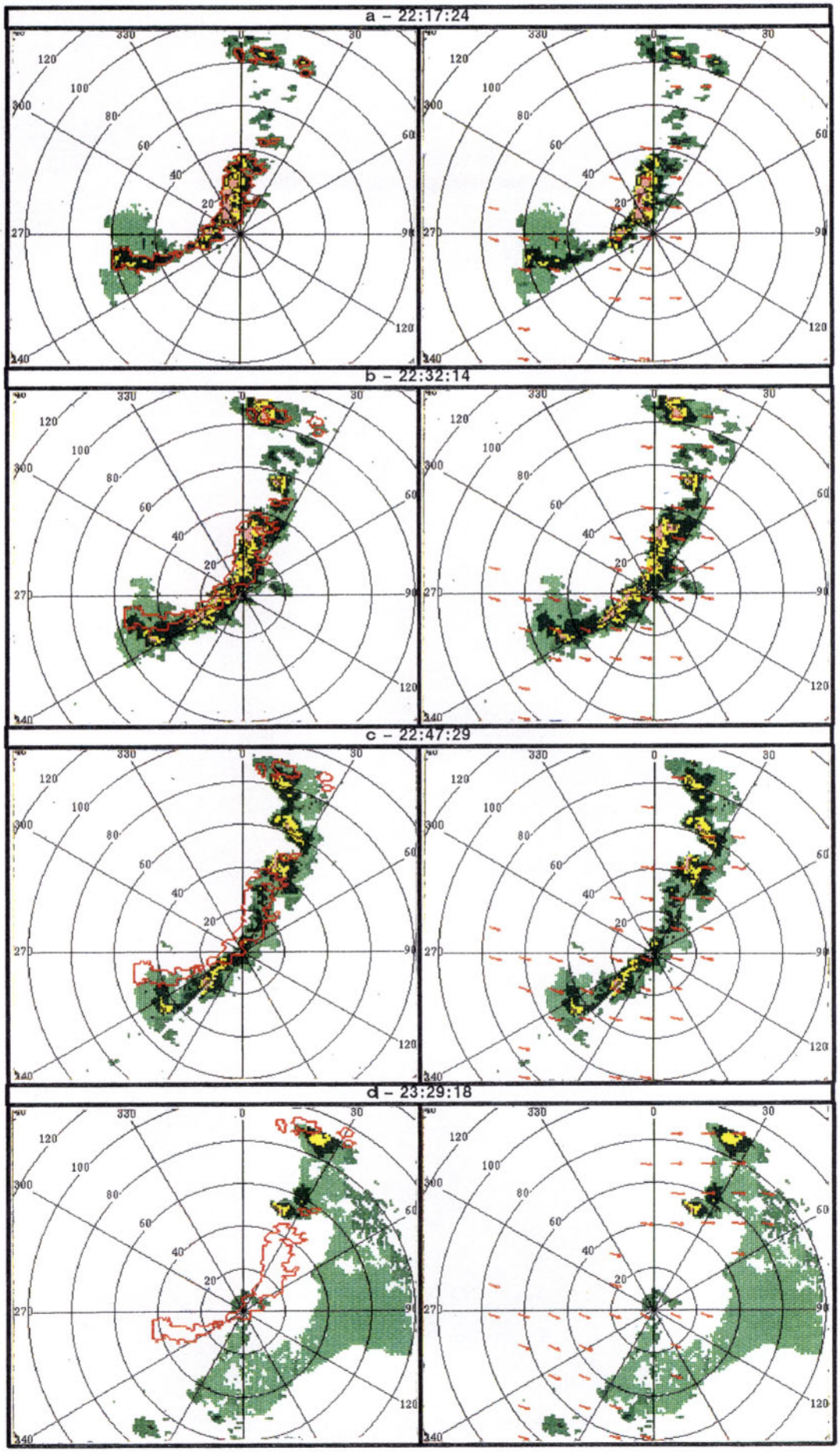


Figure 29. Extrapolation analysis: Jun15.

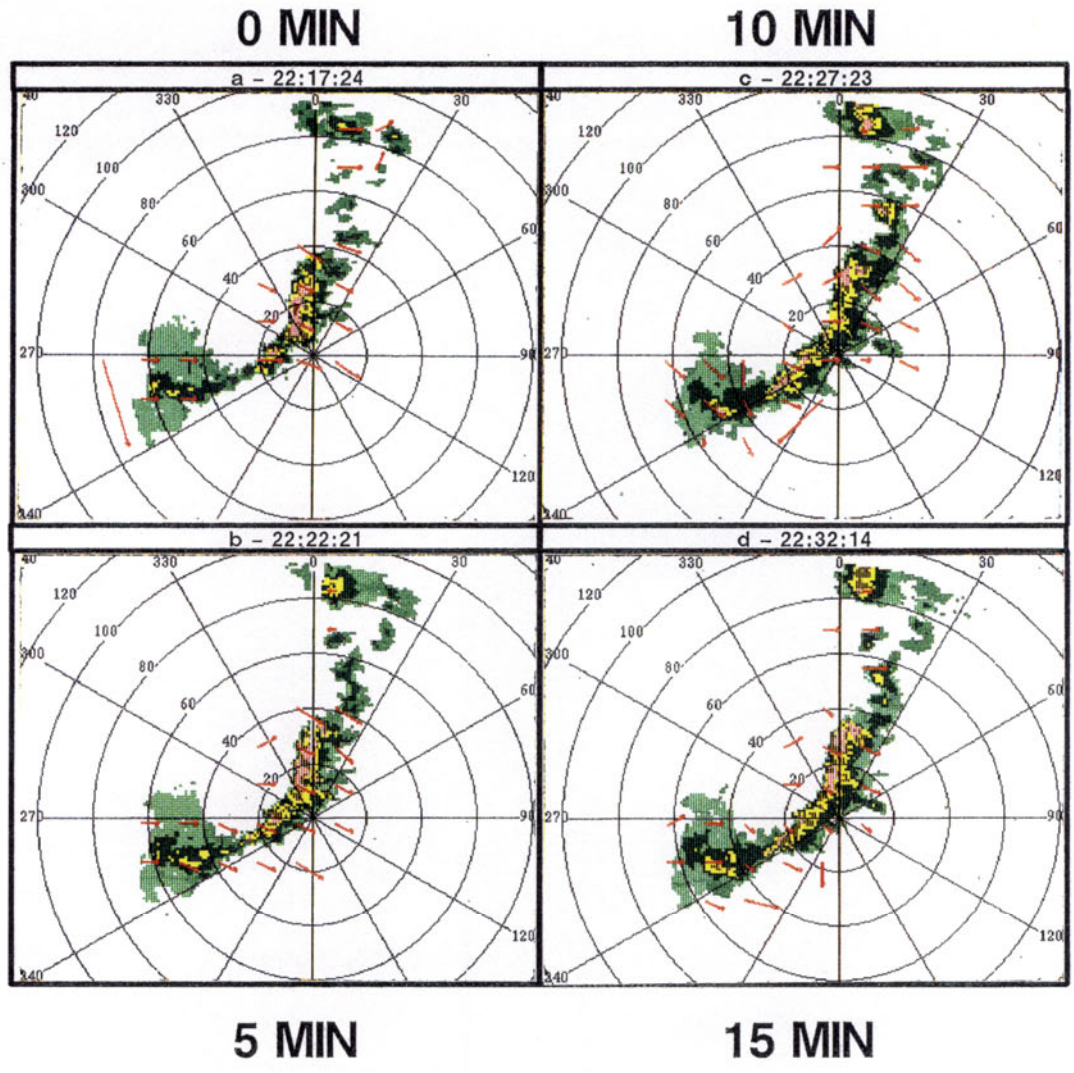


Figure 30. Extrapolation analysis: Jun15, *unfiltered vectors.*

4.4.2 Line Structures

Figure 29 shows a mature line storm from Denver (Jun15). There is some new growth "filling" the line in the NE quadrant, and the "tail end" located in the SW quadrant appears to undergo some convective growth differing (in direction) from the prevailing line translation. Extrapolation of the contour matches storm location well, except that because the line decays from 30 minutes and on, there remained little to predict at 60 minutes elapsed time. The motion vectors largely appear uniform during the period and do not indicate any motion attributable to (apparent) convective growth in the SW quadrant. Did the tracker not detect this growth or was the growth "weak" in the sense that the time-domain filtering masked its detection? Figure 30 contains the unfiltered motion vectors for the time frames between 0 and 15 minutes elapsed time. Comparison of the unfiltered and filtered vectors shows that the algorithm was able to detect local differences in motion, and that much of this evidence was lost when filtered to obtain the motion vectors in Figure 29 (note an acceleration at 10 minutes in Panel *c* of Figure 30). Figure 30 is an exceptional example for unfiltered measurements and, in general, filtering cannot be relaxed unless correlation processing can be improved (i.e., made more robust). As a consequence, small accelerations such as some due to convective growth are not reflected in the filtered output. In any event, extrapolation has been based on the assumption of a constant velocity model. Are there situations when a more or less steady-state acceleration could be measured and accounted for?

Figure 31 (Kansas City) is the final example and illustrates a fast moving line storm with significant convective growth veering from the steering direction. At 30 and 60 minutes elapsed time, large deviations between extrapolated contours and actual weather are evident. At 30 and 60 minutes (even at 15) large accelerations are present in the motion field (even though it is strongly filtered). However, because extrapolation was based on a constant velocity model, and used the velocities at 0 minutes, a prediction based on long-term extrapolation could not predict the turning motion even though it was clearly tracked by the algorithm. Such a vector turning was evident in this case for an extended period of time. Therefore, extrapolation including vector acceleration could have improved the extrapolation performance in this example.

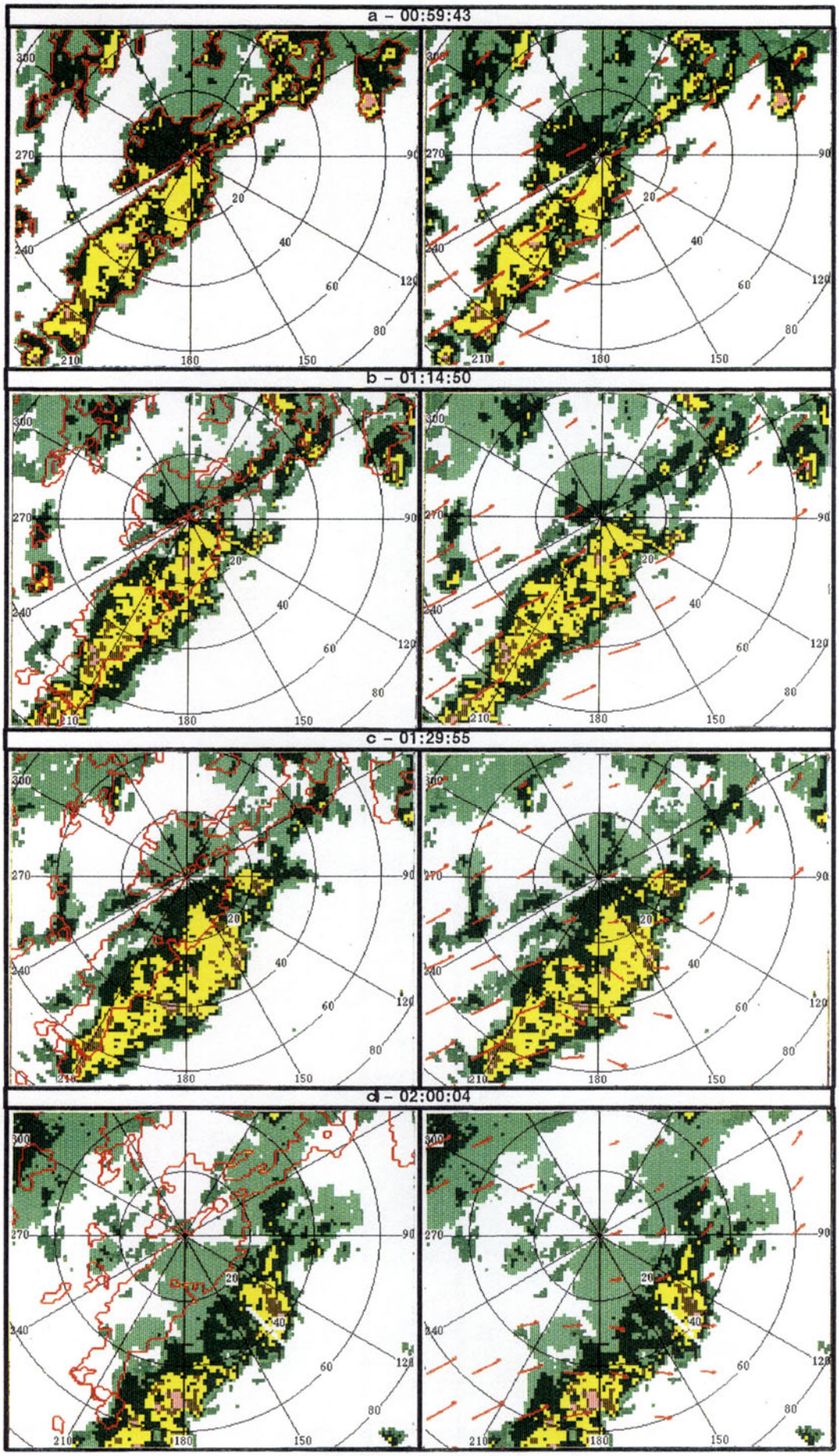


Figure 31. Extrapolation analysis: Jun26.

5. Conclusions

The overall results of this work should be encouraging. In general, the algorithm was found to perform within its expected capabilities. While the limited vertical extent and resolution of the presupposed PCP-map input do pose difficulties, neither prevented the development of a viable storm tracker. The algorithm operates mostly within an uncertainty range defined by the PCP sampling resolution. Instances of "gross" algorithm error were few and usually associated with the initial startup period, known (and therefore avoidable) edge effects, or storms moving so slow as to be below the resolving power of the algorithm. Moderate errors (one-pixel errors in displacement calculations) were shown to be strongly correlated with the decay phase of most storm systems. New developing storms and storms exhibiting moderately fast motion correlate with the best algorithm performance. An exceptional note applies to situations with highly convective storms and is discussed below.

It was verified that correlation processing can be used to track storm motion from a PCP map sequence. However, the resolution of the PCP maps is such that filtering of the correlation output is currently necessary for its stabilization. This does have an undesirable effect regarding the algorithm's ability to track accelerations. As expected, no difficulties were encountered in regard to tracking cell spilts/mergers. Binary correlation was chosen to explore whether tracking storm envelopes would successfully follow the vector sum of storm translation and storm propagation (see Section 1.2.3). Clear evidence exists to support a conclusion in the affirmative, that the correlation processor does track the vector sum. Unfortunately, the extensive filtering required for output stabilization smooths much of the detected accelerations, many of which are very short lived events (3-4 time frames). Hence, although the correlation processor appears capable of detecting "vector sum motion," algorithm output more often resembles the translational motion, which is most stable over longer periods of time.

Many of the deficiencies identified in this report are clearly approachable with additional work. There are two issues that bear strongly on the use of this product for long-term predictions. First, the variability of the PCP map, in particular its low vertical resolution, often does not provide adequate content for long-term prediction. In addition, as shown in some of the examples, the volatile nature of the PCP maps destabilizes processing. Second, propagative storm motion (whether "steady state" or not) often is detected as an acceleration in the storm locations. Under the current notion of using velocity vectors to indicate storm motion, there is required some form of approximation, matched to the user's time frame, if acceleration is to be accounted for. Otherwise, in the case of rapid convective growth, the algorithm's output will be perceived to be in large error.

6. Recommendations

The notion of a cross-correlation storm motion algorithm is appealing in the context of the TDWR data set, and work with this type of algorithm should continue. Below are listed specific recommendations directed at improving the algorithm's operation.

Because of the nature of the PCP map, raw correlation processing can be erratic and requires substantial filtering for stabilization. It is likely desirable that a layered reflectivity product (such as generated by NEXRAD) be used for the correlation input. Investigations substituting the ASR weather channel should be encouraged.

Also desirable would be additional work to develop a more robust correlation processor, possibly incorporating local or model based constraints during processing. This would alleviate the need for extensive filtering and allow the overall system to exhibit a greater sensitivity for acceleration (i.e., "steady-state" growth). Many processing artifacts (edge boundaries, clutter breakthrough, second trip weather, etc.) could be handled by an improved correlation processor as well.

In like fashion, improved correlation processing could also alleviate the processing deterioration that occurs with widespread stratiform precipitation. However, to an extent this difficulty is induced by the arbitrary use of level-2 weather thresholding. An examination of alternative thresholding (different from the NWS levels) may reveal a level better suited for storm tracking.

The possible use of "hybrid" maps, combining long-range portions of the low-PRF scan with the PCP maps, should also be considered as a possible means for reducing edge (and second trip) artifacts. These extended maps are also desirable because they increase the spatial scope of the weather observed (the notion of extrapolating weather 30-60 minutes into the future only makes sense if the algorithm can "see" weather within a 30-60 minute window [centered on the airport] of the present).

This algorithm appears capable of detecting (tracking) propagative growth. While storm translation appears to be steady over an extended period of time, "propagative growth" often appears short lived relative to the stated goal of 30-60 minute planning. Such propagative growth may be better viewed as an acceleration vector than as a velocity vector to be summed with a translational component. Can this information be used to improve the prediction of future storm location or conveyed to the end user in useful form? These are areas for consideration but having, as yet, no clear solution.

REFERENCES

1. The Federal Aviation Administration Plan for Research, Engineering, and Development, November 1988. Draft.

2. D. Turnbull, J. McCarthy, J. Evans, and D. Zrnić. The FAA Terminal Doppler Weather Radar (TDWR) program. In *Preprints: Third International Conference on the Aviation Weather System*, pages 414–419, 1989.

3. S.D. Campbell and M.W. Merritt. TDWR scan strategy requirements, revision 1. Project Report FAA-RD-DOT/FAA/PM-87-22, MIT Lincoln Laboratory, Lexington, Massachusetts, April 1990.

4. J.C. Brasunas. A comparison of storm tracking and extrapolation algorithms. Project Report DOT-FAA-PM 84-1, MIT Lincoln Laboratory, 1984.

5. C.F. Chappell. Quasi-stationary convective events. In P.S. Ray, editor, *Mesoscale Meteorology and Forecasting*, chapter 13, pages 289–310. American Meteorological Society, Boston, Massachusetts, 1986.

6. C.L. Bjerkaas and D.E. Forsyth. Operational test of a three-dimensional echo tracking program. In *19th radar meteorology conference*, pages 244–247, 1980.

7. R.K. Crane. Automatic cell detection and tracking. *IEEE Transactions on Geoscience Electronics*, GE-17(4):250–262, 1979.

8. R.H. Blackmer, Jr, R.O. Duda, and R. Reboh. Application of pattern recognition techniques to digitized weather radar data. Final Report SRI Project 1287, Stanford Research Institute, 1973.

9. F.C. Glazer. *Hierarchical Motion Detection.* PhD thesis, University of Massachusetts, 1987.

10. J.F. Vesecky, R. Samadani, M.P. Smith, J.M. Daida, and R.N Bracewell. Observation of sea-ice dynamics using synthetic aperture radar images: automated analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 26(1):38–47, 1988.

11. R.E. Rinehart. A pattern-recognition technique for use with conventional weather radar to determine internal storm motions. In *Recent Progress in Radar Meteorology*, number 13, pages 119–134. 1981.

12. P.J. Burt, C. Yen, and X. Xu. Local correlation measures for motion analysis: a comparative study. *Procedings Pattern Recognition and Image Processing*, pages 269–274, 1982.