Multiscale Storm Identification and Forecast

V Lakshmanan

September 6, 2002

V Lakshmanan, lakshman@nssl.noaa.gov

Image identification methods

The operational way of identifying storms from radar images involves the use of multiple thresholds and counting runs of values above a threshold along a radial.

The centroids are used as proxy for the storms. "Storm Cell Identification and Tracking" (SCIT).

Short-term forecast methods

Short-term forecast methods

The common ways of forecasting storm locations include:

1. Find storm cores in each frame (volume scan) and then match cores across frames (e.g: WSR-88D/NSSL SCIT, NCAR Titan). Extrapolate change in position.

2. Use rectangular sub-grids and find maximum correlation within search radius. (e.g: TREC, MIT/LL Growth and Decay Tracker)

3. Use sub-grids ranging in size from entire image to small 16km x 16km grids and compute motion estimates, apply continuity criteria to derive wind-speed. (e.g: various wavelet methods, Czech COTREC method)

V Lakshmanan, lakshman@nssl.noaa.gov

Strengths/weaknesses

The storm core technique is suitable for small scale storms i.e. for short-range forecasts.

The large scale features and cross-correlation technique is suitable for longer forecasts, but with loss of detailed motion estimates. Also, assumption is that storms are of the scale of the sub-grid, not larger.

The multiscale estimation is suitable also for large scale forecasts, with less precise detailed motion estimates.

All correlation techniques rely on reverse projection, so there needs to be wind speed at the spot where the storm is moving to.

The image template methods assume that all pixels with a grid are moving together.

V Lakshmanan, lakshman@nssl.noaa.gov

Our hybrid approach

Motion estimates are made for groups of storms, but at various scales.

The motion estimate for a storm cell is the movement that minimizes the mean-absolute-error between the current frame and corresponding pixels in the previous frame.

The template is not a sub-grid of the image, but is instead the actual shape of the storm cell.

Rather than matching storm cells across frames, motion estimates are made via an image-analysis manner.

V Lakshmanan, lakshman@nssl.noaa.gov

Major stages

1. Find storms at different scales.

2. Estimate motion at the various scales.

3. Forecast for different periods using motion at different scales.

V Lakshmanan, lakshman@nssl.noaa.gov

Identifying storms

A K-Means clustering technique is used to identify components in vector fields such that the components found at different scales are nested.

Why nested partitions?

V Lakshmanan, lakshman@nssl.noaa.gov

Identifying storms

A K-Means clustering technique is used to identify components in vector fields such that the components found at different scales are nested.

Why nested partitions? Because storms structures are strictly hierarchical.

Cluster image values (reflectivity/infrared temperature, etc.) in neighborhood of pixel on two opposing criteria:

- Belong to same cluster as your neighbors.

- Belong to cluster whose mean is closest to your value.

V Lakshmanan, lakshman@nssl.noaa.gov

How

We can incorporate hierarchical segmentation into a K-Means clustering technique by steadily relaxing inter-cluster distances.

V Lakshmanan, lakshman@nssl.noaa.gov

Technique

1. Associate a vector of textural measurements with each pixel.

2. Requantize image into K levels using K-Means Clustering.

   (a) Initialize the k means somehow – we simply divided up the measurement space into equal intervals.
   (b) Assign the closest mean to each pixel.
   (c) Start iterating on the clustering scheme by reassigning pixels based on the Markov assumption.
   (d) Iterate until stable

3. Obtain most detailed segmentation through region growing, morphology.

4. Use inter-cluster distance to get coarser segmentation results.

V Lakshmanan, lakshman@nssl.noaa.gov

Neighborhood

When computing statistical texture, we compute the statistics in the neighborhood of a pixel.

How big a neighborhood? We would expect that with a tight neighborhood, we would get a large number of small regions, and that with a large neighborhood, we would get a fewer number of larger regions. That is what happens.

V Lakshmanan, lakshman@nssl.noaa.gov
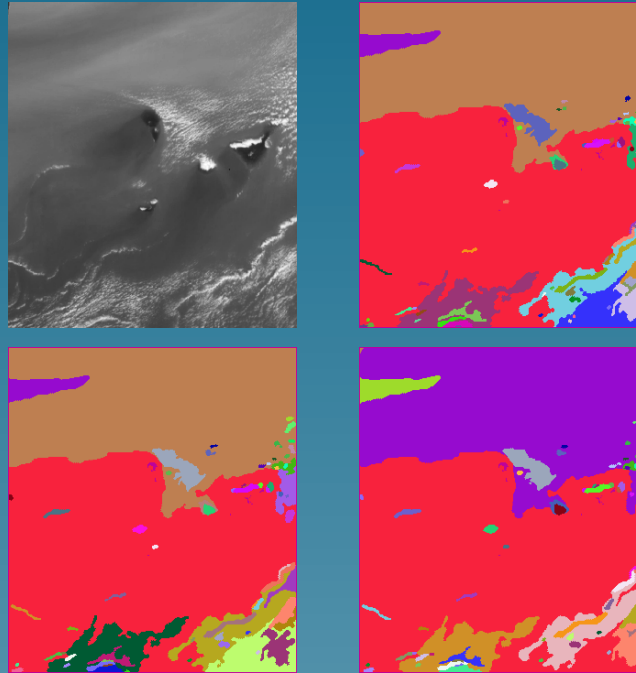
Figure 1: Segmenting a satellite image using statistics computed in neighborhoods of varying sizes (3,5 and 7)

back

V Lakshmanan, lakshman@nssl.noaa.gov

What is K?

K is not the number of regions in the final segmented output.

It is the number of central vectors about which we do the clustering.

The number of regions is determined by the spatial location.

As the number K increases, the clusters cover a smaller range in the texture space.

In case the number of regions is not known a priori, a very high value of K may be chosen. The most detailed segmentation may have too many regions, but coarser levels might yield the desired result. This is one advantage of using a hierarchical technique.

back

V Lakshmanan, lakshman@nssl.noaa.gov

Markov assumption

1. Take as candidates all the labels in the 8-neighborhood of the pixel.

2. Compute the contiguity distance $d_c(k)$ that would result if the label were changed to k.

3. Compute the distance between the mean of the kth cluster and the pixel's texture vector, $d_m(k)$.

4. Assign to the pixel the label for which E(k) is minimum. If this causes a change in the label of the pixel, another pass through the image is required.

We iteratively move pixels minimizing

$$E(k) = \lambda d_m(k) + (1 - \lambda) d_c(k) \qquad 0 \leq \lambda \leq 1 \qquad (1)$$

V Lakshmanan, lakshman@nssl.noaa.gov

where the distance in the measurement space is:

$$d_m(k) = \parallel \mu_k^n - T_{xy} \parallel \qquad (2)$$

and the discontiguity measure is::

$$d_c(k) = \sum_{ij \epsilon N_{xy}} (1 - \delta(S_{ij}^n - k)) \qquad (3)$$

a   b
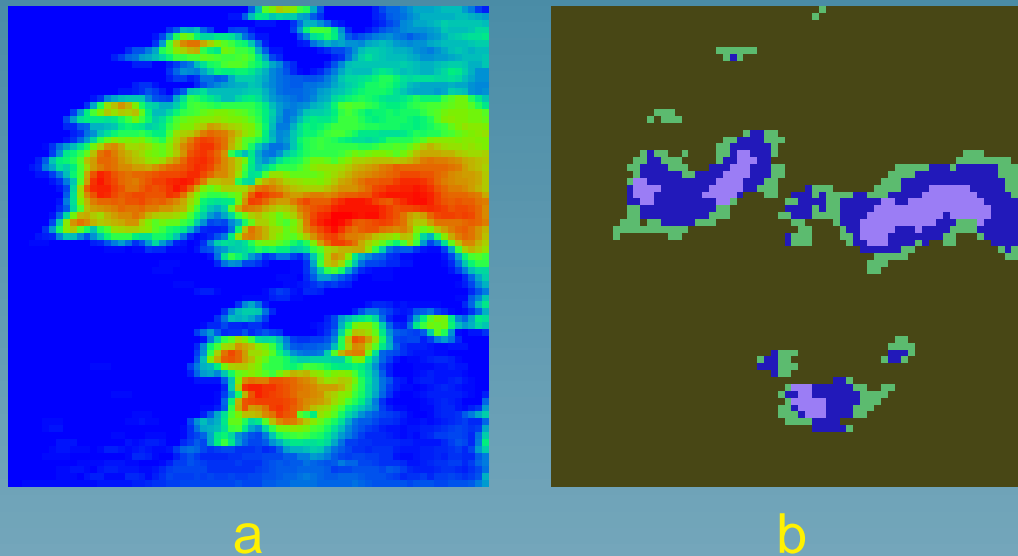
Figure 2: (a) reflectivity image (b) K=4 clustering back

V Lakshmanan, lakshman@nssl.noaa.gov

Most detailed segmentation

A region growing algorithm is employed to build a set of connected regions, where each region consists of 8-connected pixels that belong to the same K-Means cluster.

If a connected region is too small, then its cluster mean (the mean of the texture vectors at each pixel in the region) is compared to the cluster means of the adjoining regions and the small region is merged with the closest mean.

The result of the K-Means segmentation, region growing and region merge steps is the most detailed segmentation of the image.
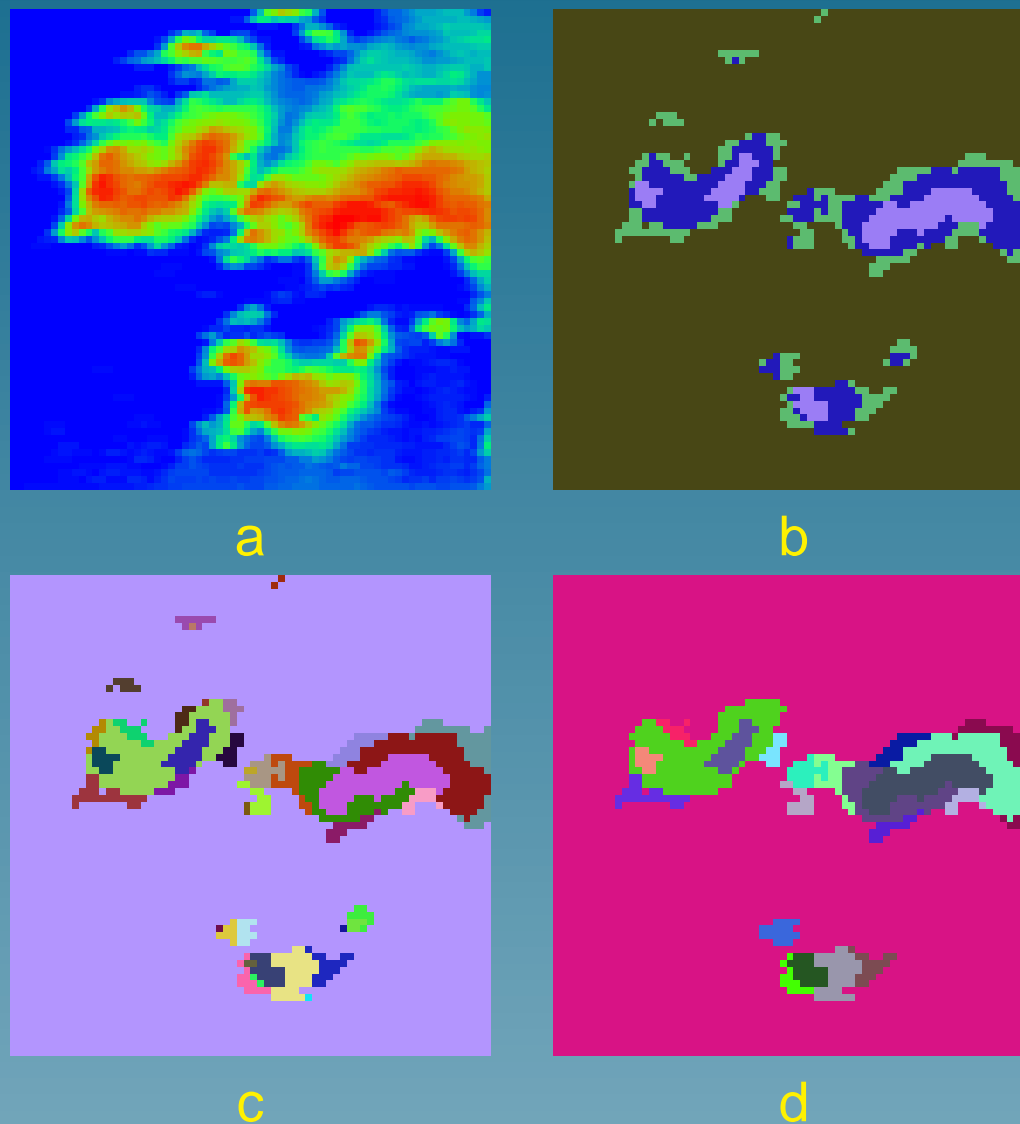
Figure 3:   (a) reflectivity image (b) K=4 clustering (c) Region growing (d) Morphological post-processing (back)

V Lakshmanan, lakshman@nssl.noaa.gov

Hierarchical segmentation

The inter-cluster distances of all adjacent clusters (or regions) in the image are computed.

A threshold is set such that half the pairs fall below this threshold. If a pair of clusters differ by less than this threshold, they are merged and cluster means updated.
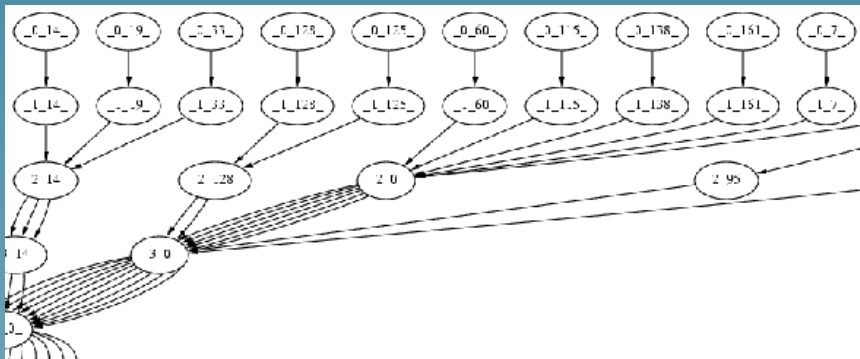
Continue until no two adjacent regions are closer in cluster space than the threshold. When this process is complete, we have the

V Lakshmanan, lakshman@nssl.noaa.gov

next coarser scale of the segmentation.

Repeat this process until no changes happen.
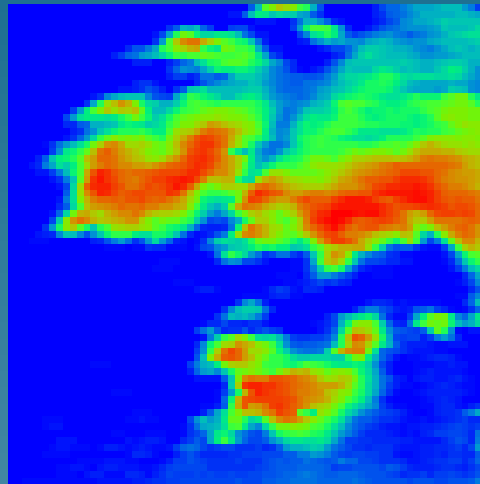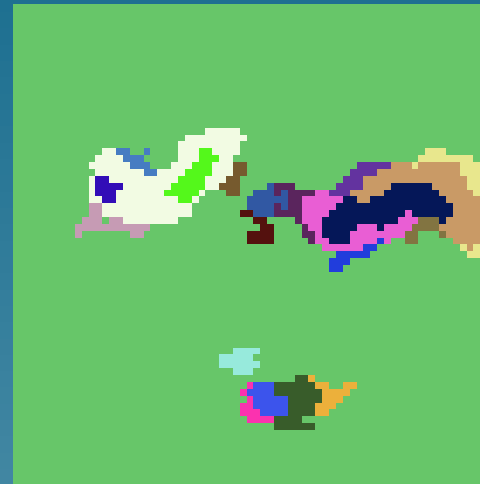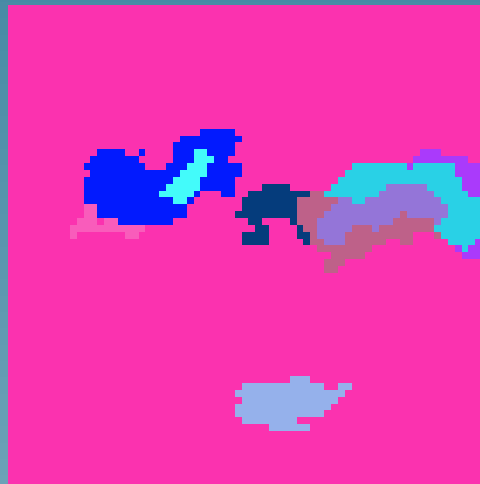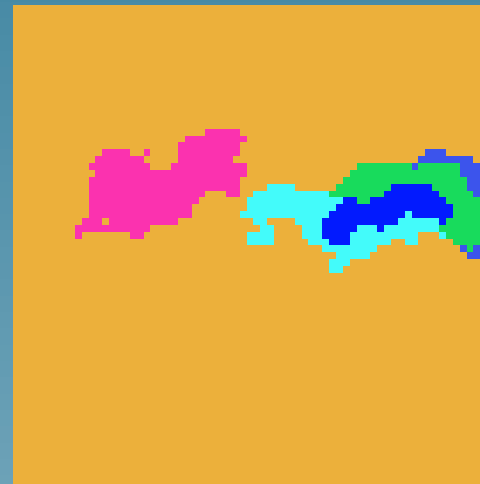
Figure 4: (a) reflectivity image (b) Most detailed (c,d) coarser levels

V Lakshmanan, lakshman@nssl.noaa.gov

For more details: Lakshmanan, V. 2001: A Heirarchical, Multiscale Texture Segmentation Algorithm for Real-World Scenes. Ph.D. thesis, U. Oklahoma, Norman, OK.

V Lakshmanan, lakshman@nssl.noaa.gov

Storm identification in radar imagery

This is the original radar reflectivity image from Fort Worth on Apr. 20,

1995.

V Lakshmanan, lakshman@nssl.noaa.gov
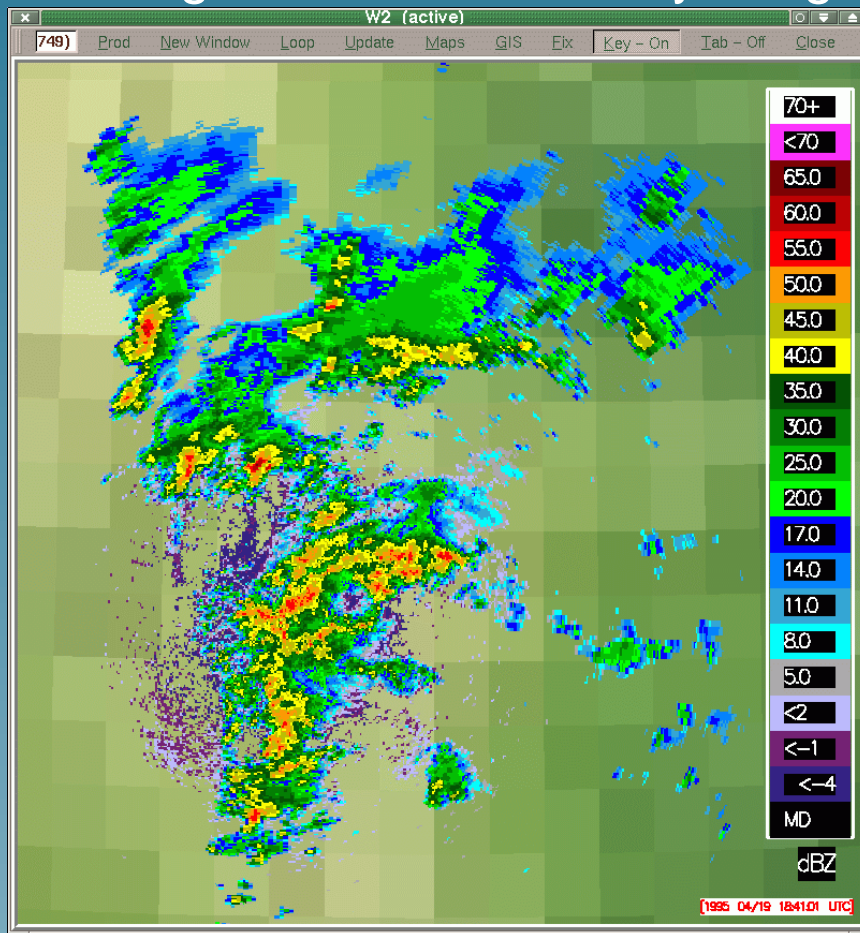
Detail

These are the detected storms at the most detail:
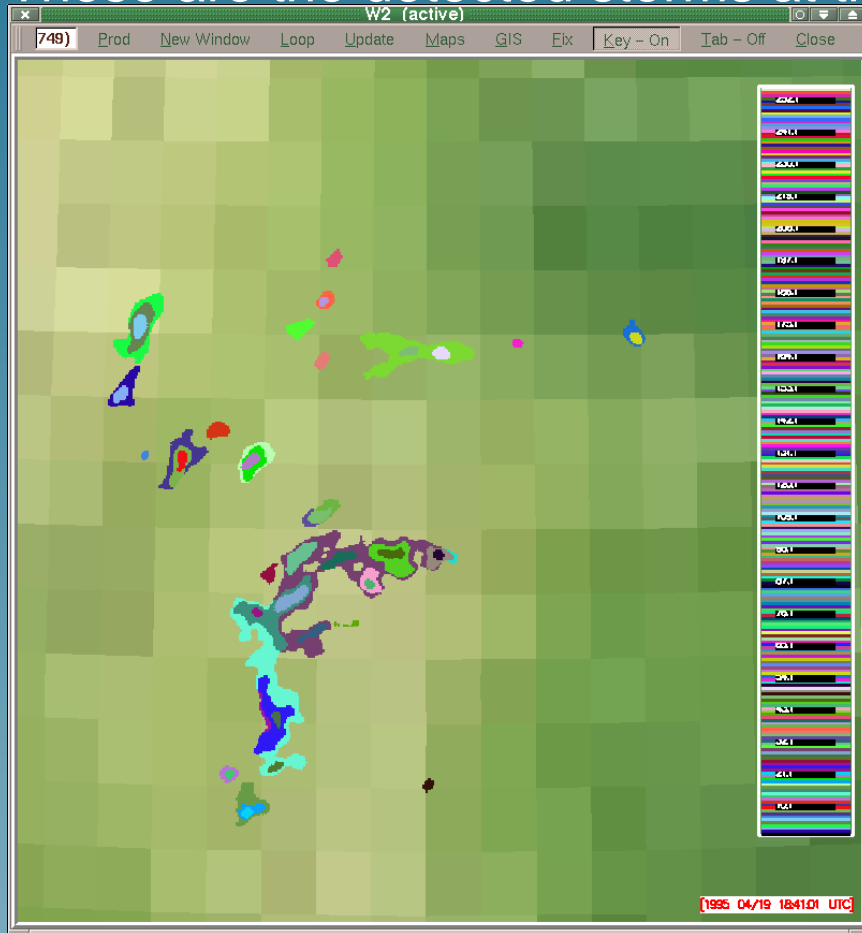
These storms will be used as

V Lakshmanan, lakshman@nssl.noaa.gov

template for forecasting less than 30 min.

V Lakshmanan, lakshman@nssl.noaa.gov

Less Detail

These are the detected storms at lesser detail:

These storms will be used as

V Lakshmanan, lakshman@nssl.noaa.gov

template for forecasting 60 - 90 min .

V Lakshmanan, lakshman@nssl.noaa.gov

Storm identification in satellite imagery

This is the original satellite infrared window channel image on Oct. 9,

2001.

V Lakshmanan, lakshman@nssl.noaa.gov

Detail

These are the detected storms at the most detail:

V Lakshmanan, lakshman@nssl.noaa.gov
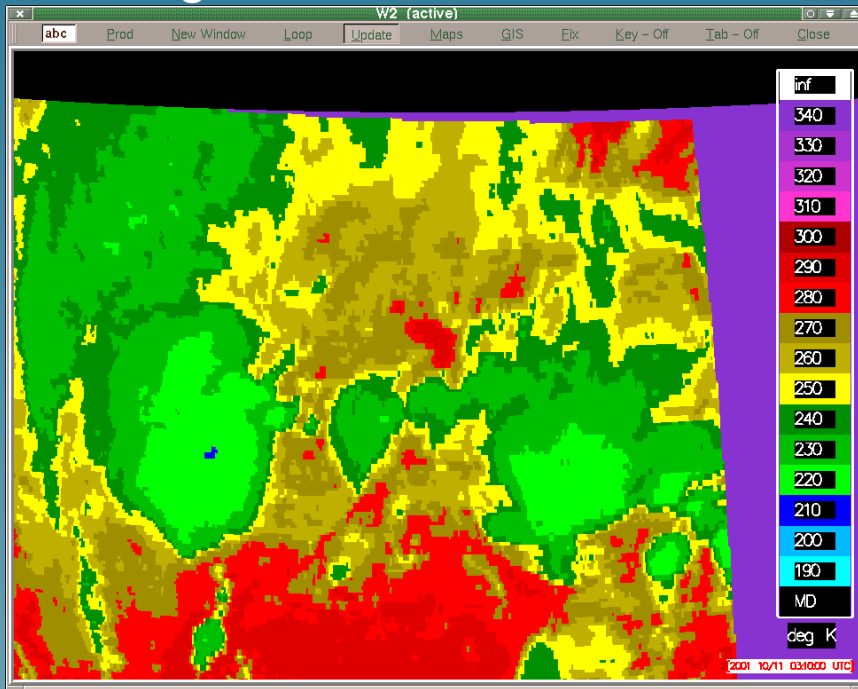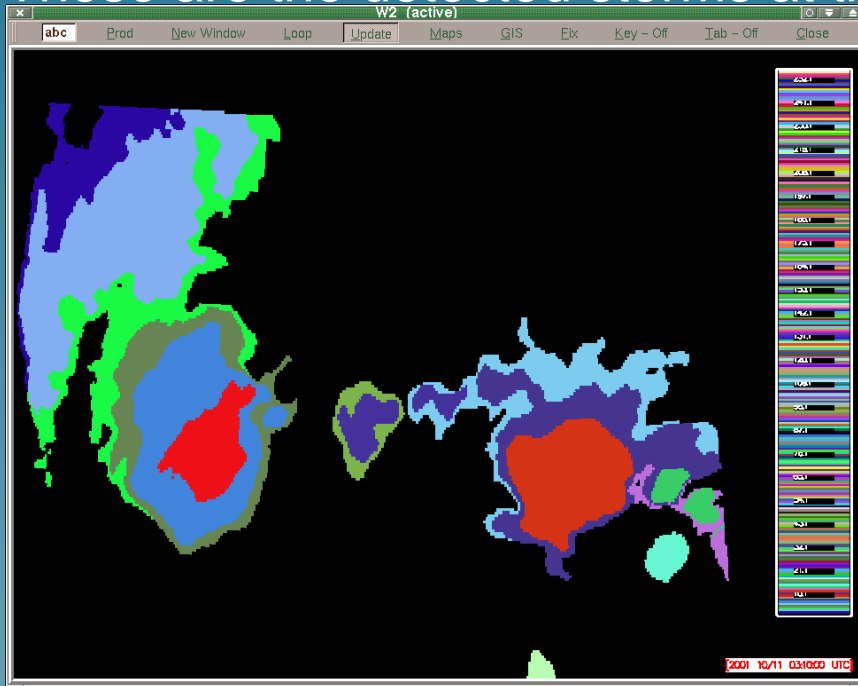
Motion Estimation

Using these storms as a template, the movement that minimizes the absolute-error between two frames is computed.

For radar images, we used consecutive (5 min) frames. For satellite imagery, we used frames 400 seconds apart.

V Lakshmanan, lakshman@nssl.noaa.gov

Stage 2: Motion Estimate

Motion estimation is done by moving a template of the identified cluster at the appropriate scale around in the previous image.

A matrix of mean absolute error at the different positions is obtained.

V Lakshmanan, lakshman@nssl.noaa.gov

MAE Minimization
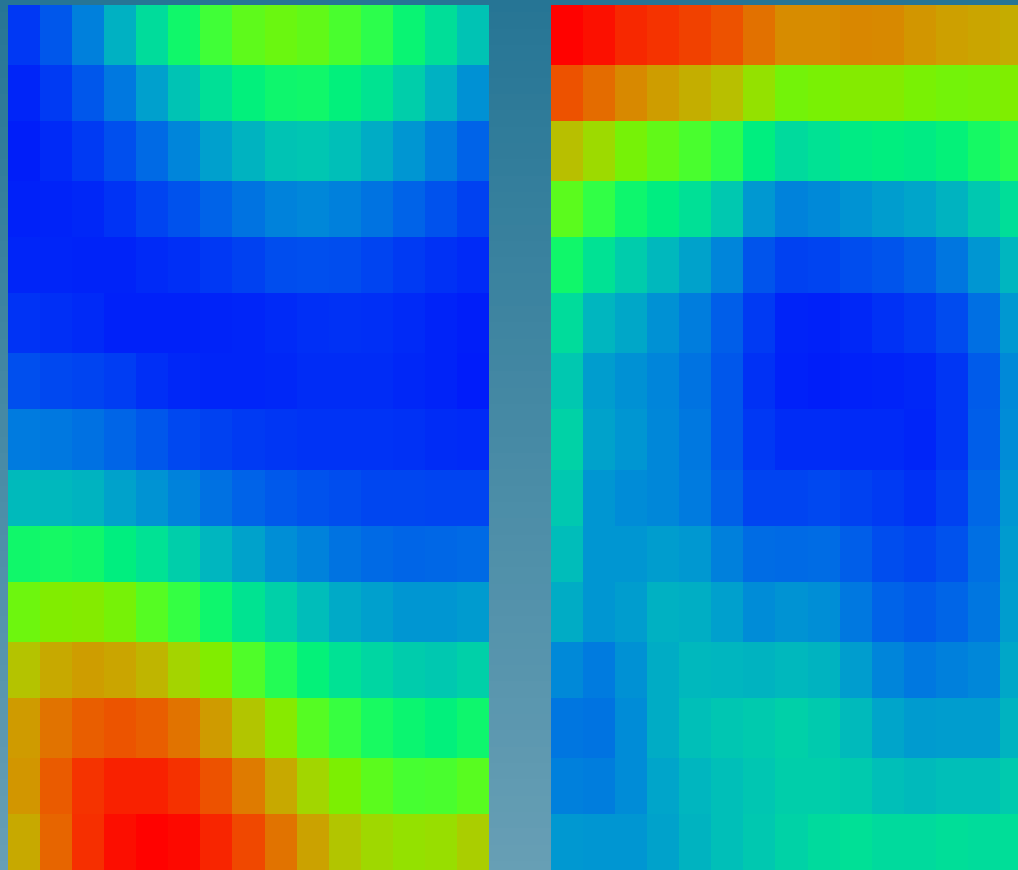
The field is minimized by weighting each pixel by how much it differs from the absolute minimum and finding the centroid.

V Lakshmanan, lakshman@nssl.noaa.gov

Example

This is the east component of the motion at the most detailed scale.
Note that for each storm template, we get a single motion estimate.
Red is moving east. Not shown: the north-south component.

V Lakshmanan, lakshman@nssl.noaa.gov
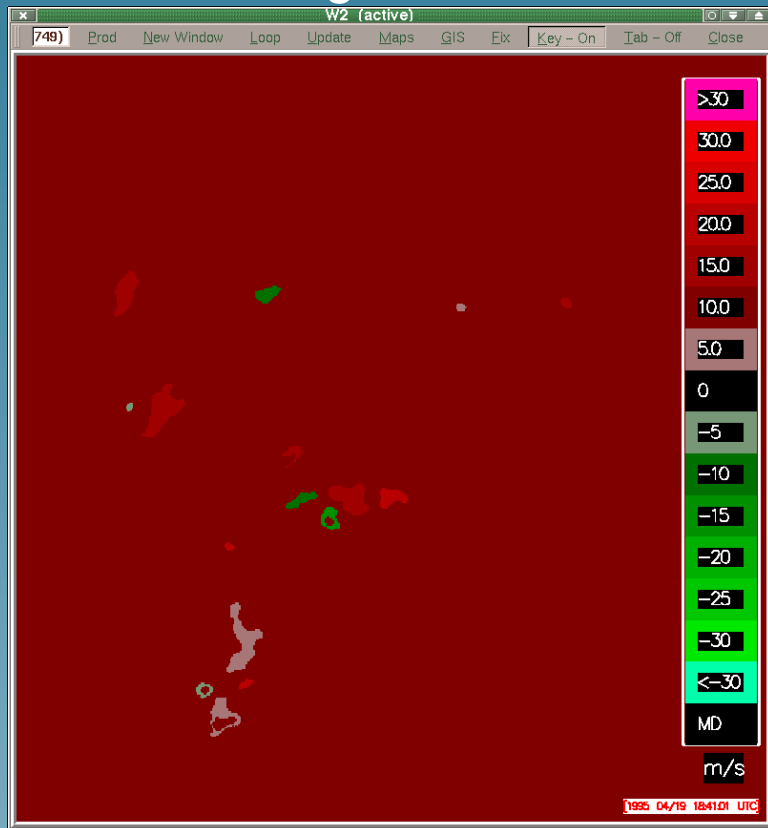
Growth/Decay Estimate

For each storm template, we also get a growth/decay estimate.
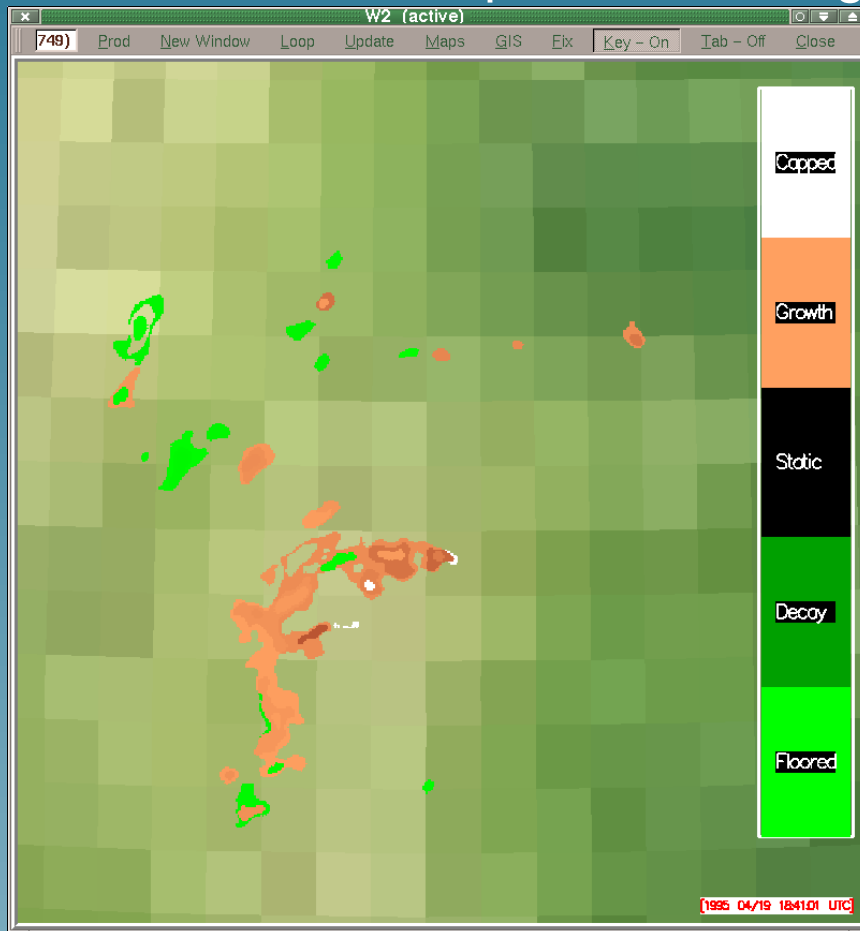
This is based on how much the

V Lakshmanan, lakshman@nssl.noaa.gov

average value inside the template changes.

V Lakshmanan, lakshman@nssl.noaa.gov

Stage 3: Forecast

Using all of this information, we can forecast how the field is going to look ... The original reflectivity field:

V Lakshmanan, lakshman@nssl.noaa.gov
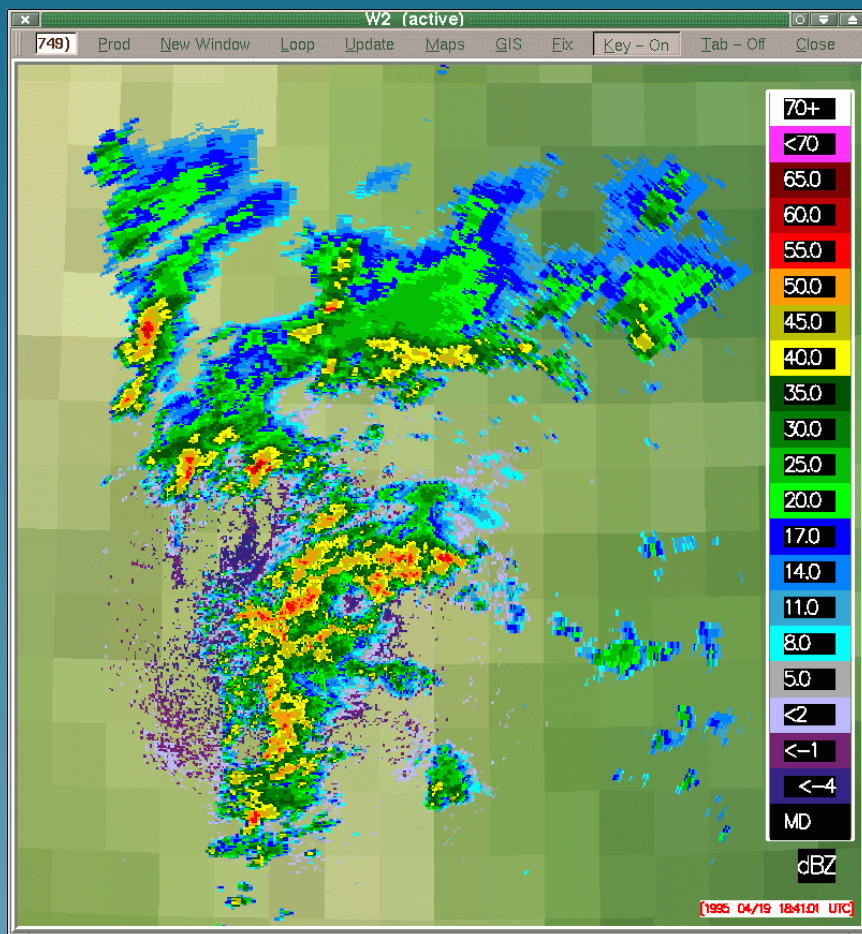
The forecast is done:

V Lakshmanan, lakshman@nssl.noaa.gov

1. Forward: project data forward in time to a spatial location given by the motion estimate at their current location and the elapsed time.

2. Define a background motion estimate given by the mean storm motion.

3. Reverse: obtain data at a spatial point in the future based on the current wind direction at that spot and current spatial distribution of data.

V Lakshmanan, lakshman@nssl.noaa.gov

15 minute forecast

Clustering happened in the range 30-60dBZ, so that is what gets

forecast:

V Lakshmanan, lakshman@nssl.noaa.gov
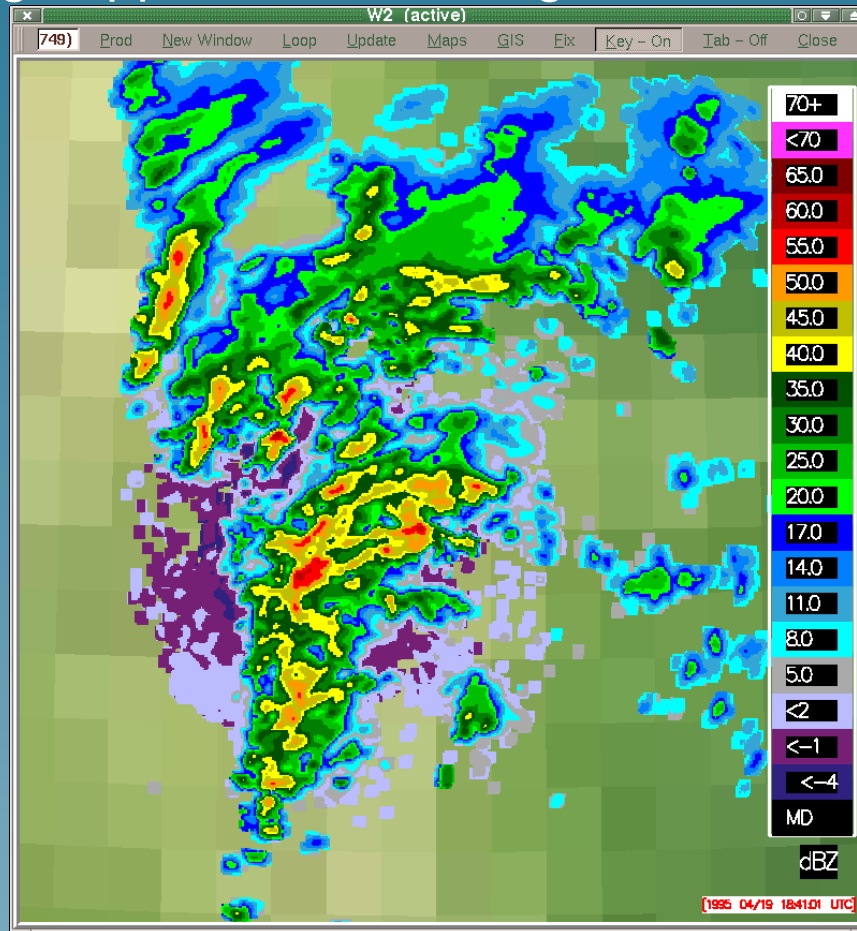
60 minute forecast

V Lakshmanan, lakshman@nssl.noaa.gov
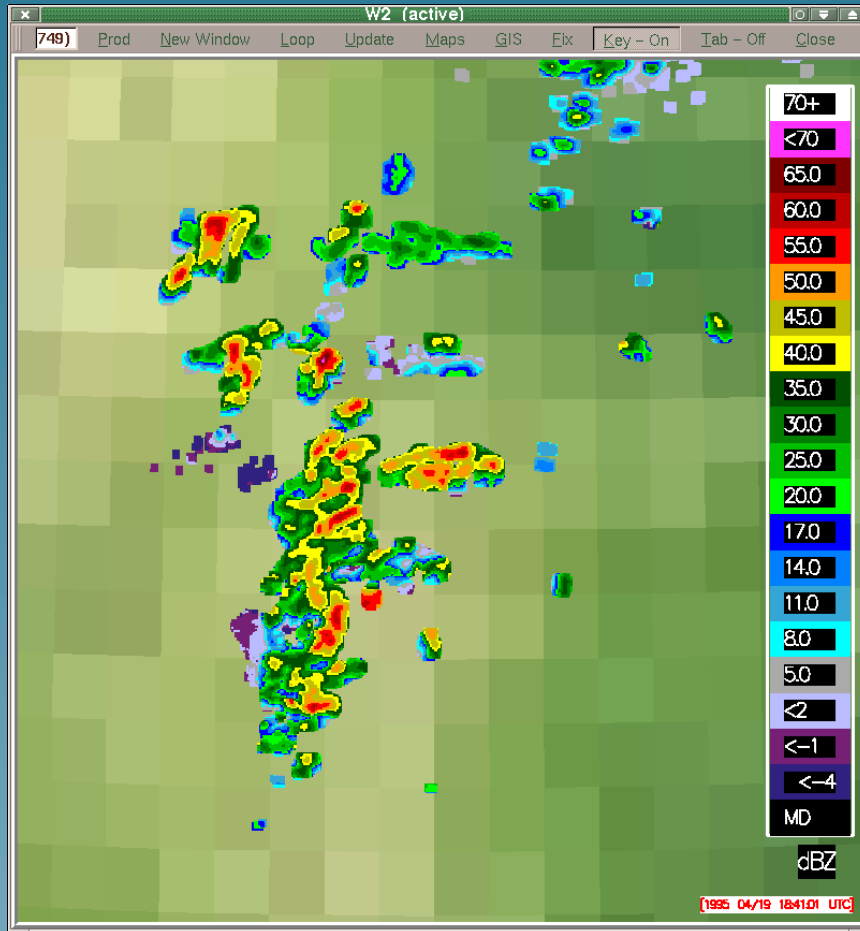
Choice of scale of comparision

From now on, we'll use the most detailed scale (scale 0) since that

gives the most errors ...

V Lakshmanan, lakshman@nssl.noaa.gov
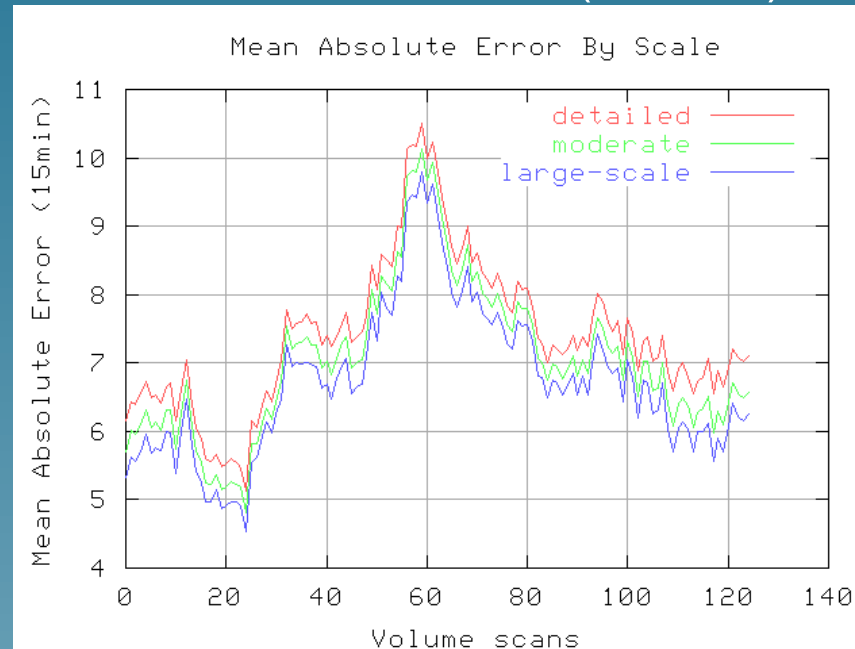
Performance Measures: Bias

The bias of the forecast fields at different scales. The time axis is over 750 minutes (12.5 hours). We seem to be over-estimating by about 2

dBZ.

V Lakshmanan, lakshman@nssl.noaa.gov
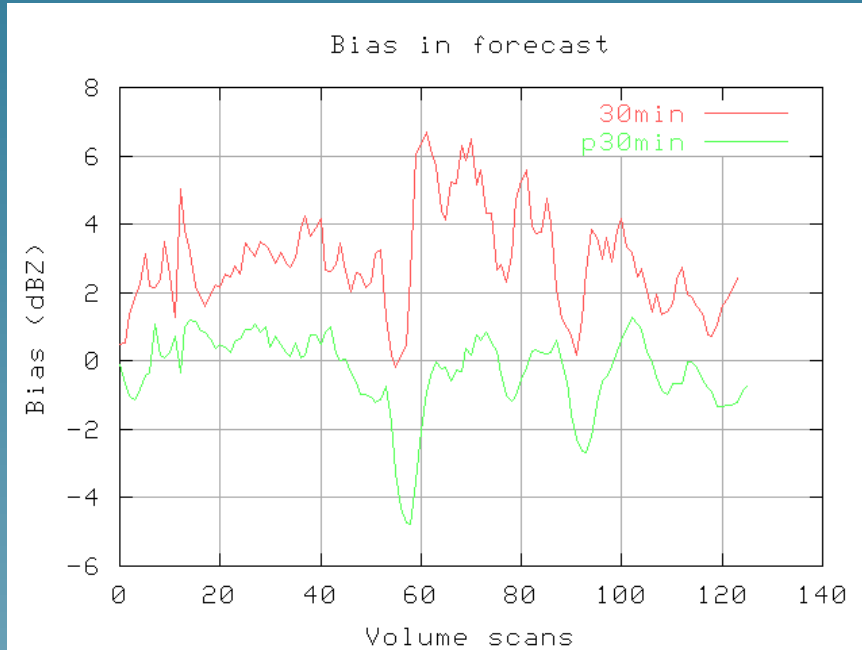
Performance Measures: CSI

Skill at predicting 30dBZ+ at the exact location (no spatial window
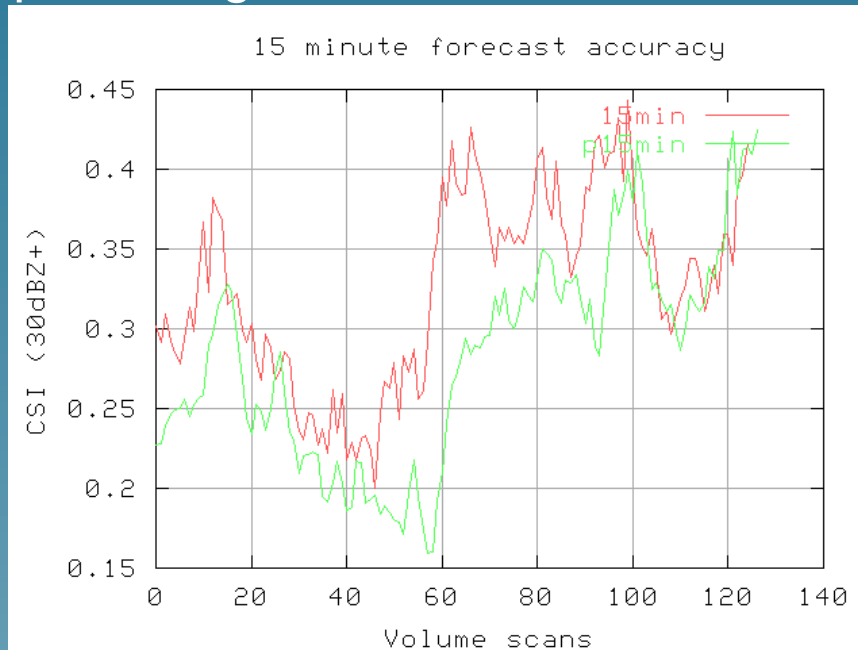
used):

We do slightly better than persistence in a 15 minute forecast ... notice the wierdness at the end of the sequence ...

V Lakshmanan, lakshman@nssl.noaa.gov

Performance Measures: Mean Absolute Error

Mean absolute error for the 15 minute forecast compared with a
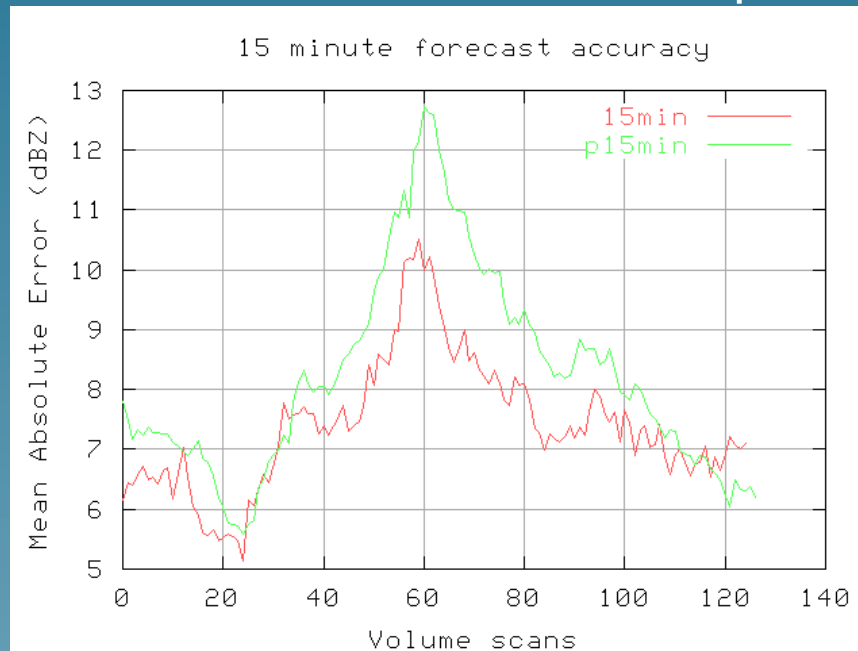
persistence forecast.

V Lakshmanan, lakshman@nssl.noaa.gov

CSI vs MAE

The CSI seems to indicate the technique performs a lot better than persistence. The MAE doesn't show much difference between the two.

Why?

V Lakshmanan, lakshman@nssl.noaa.gov

CSI vs MAE

The CSI seems to indicate the technique performs a lot better than persistence. The MAE doesn't show much difference between the two.

Why?

Because MAE takes into account actual reflectivity values. We are good at predicting storm location, but not so good at growth/decay.

Second question: Why does persistence as a strategy suddenly start doing better at the very end?

V Lakshmanan, lakshman@nssl.noaa.gov

Clutter!

Because there is clutter in the data sequence ...
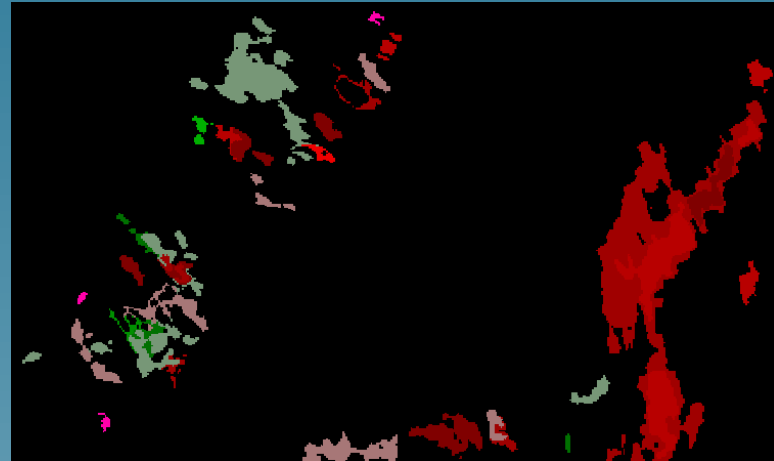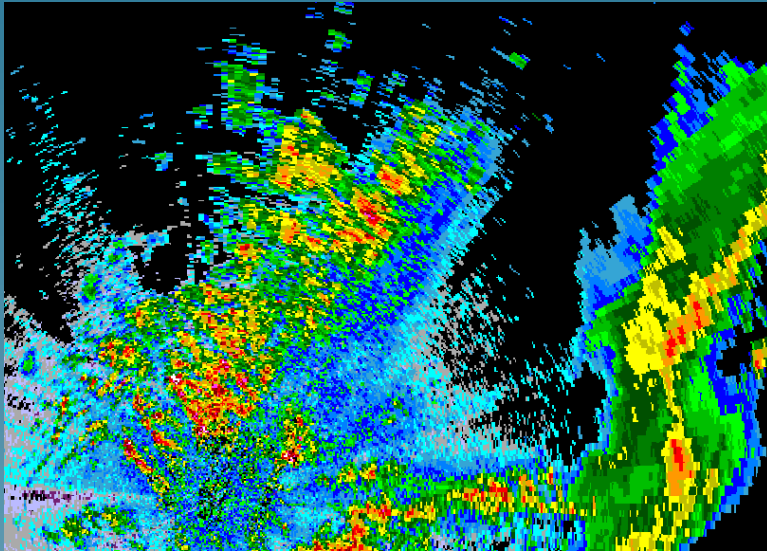
The clutter is badly segmented and so we get a poor motion estimate there.
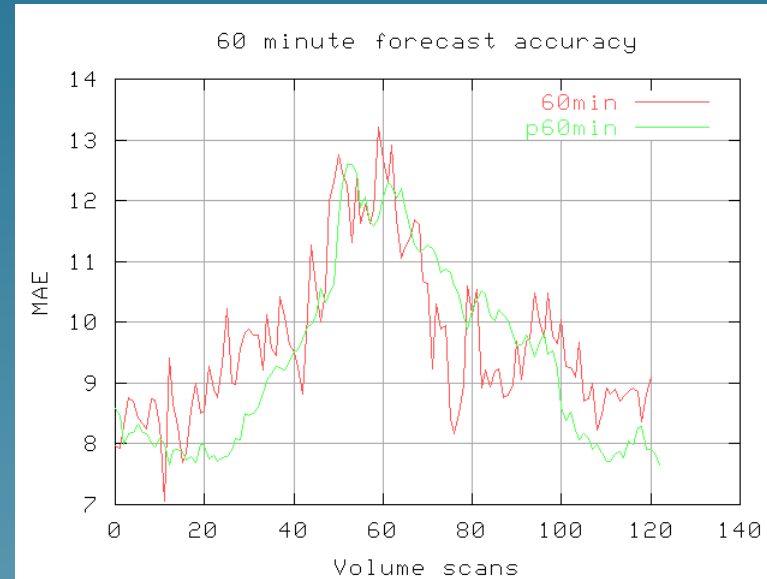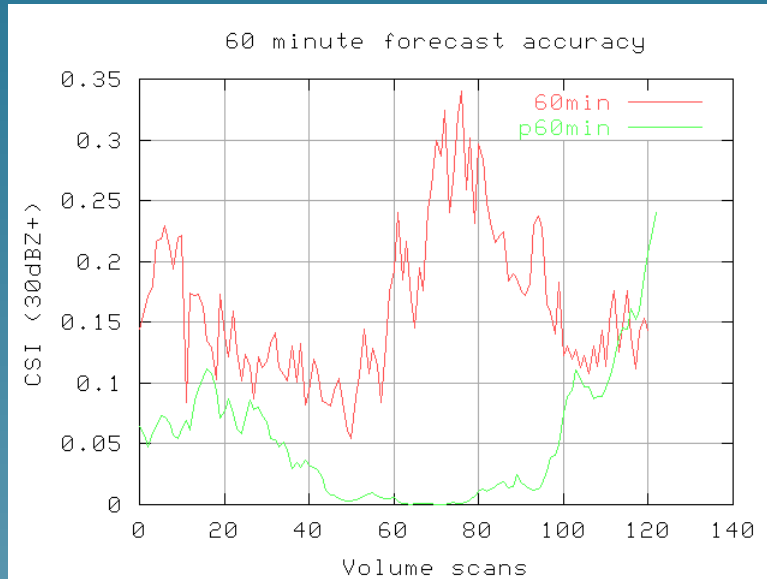
V Lakshmanan, lakshman@nssl.noaa.gov

How about longer forecasts?

As anticipated, persistence loses its charm beyond 15 minutes, but in the part of the sequence where there is a lot of clutter, our performance is still poor.

V Lakshmanan, lakshman@nssl.noaa.gov

15 and 30 minute forecast
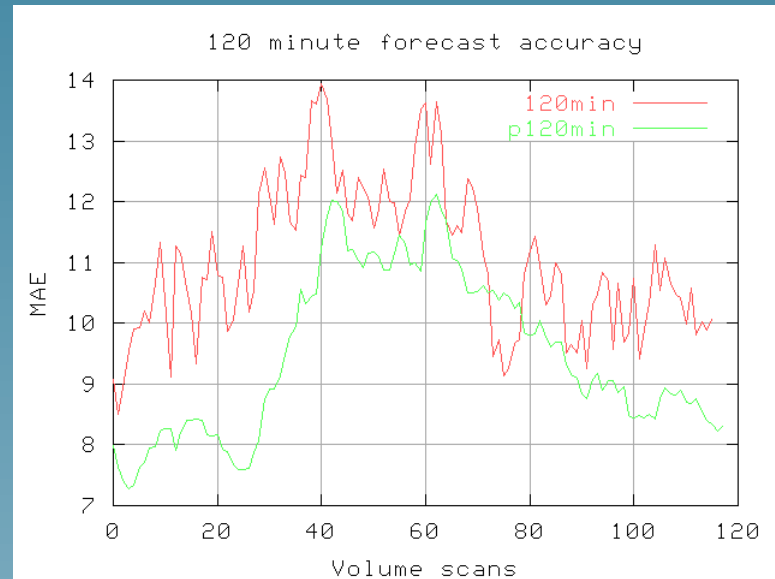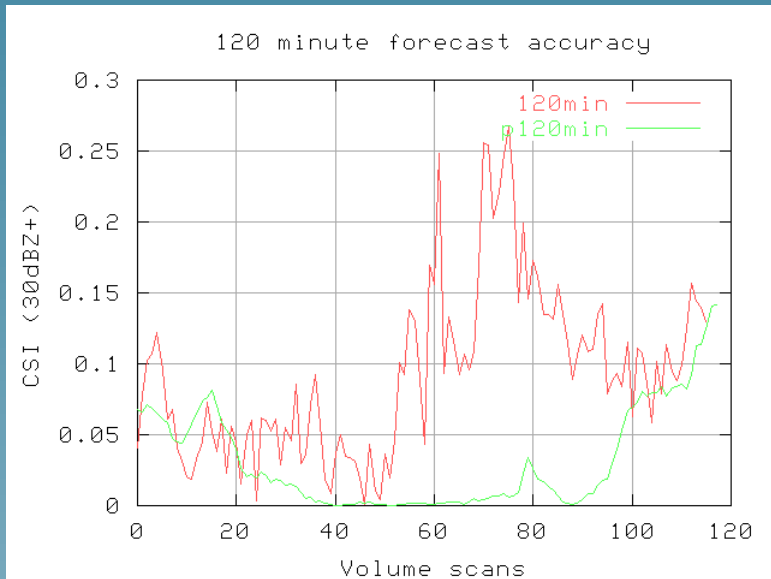
V Lakshmanan, lakshman@nssl.noaa.gov
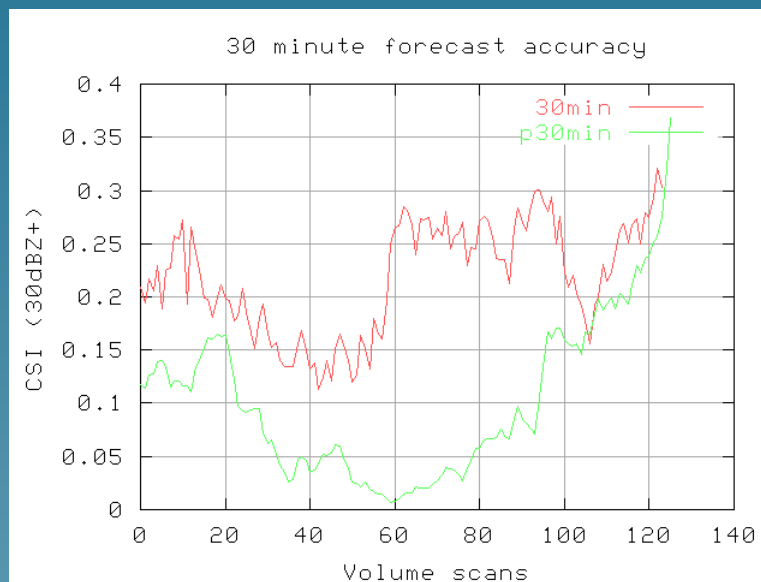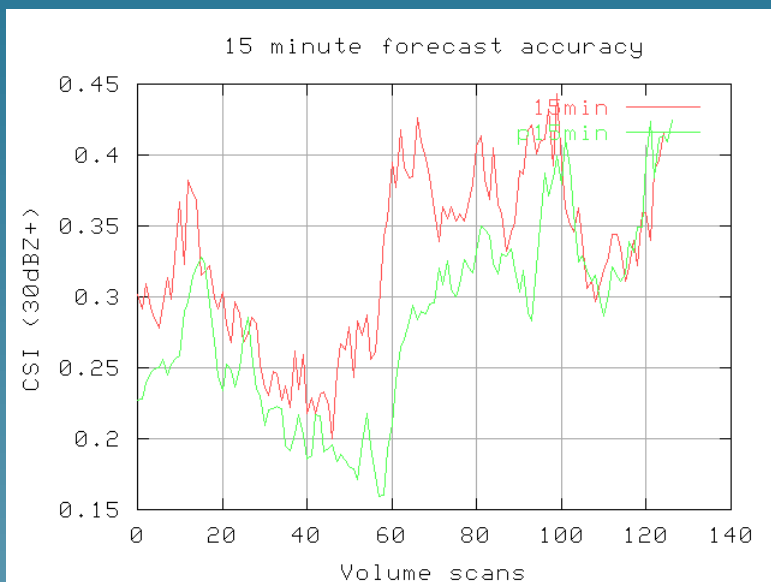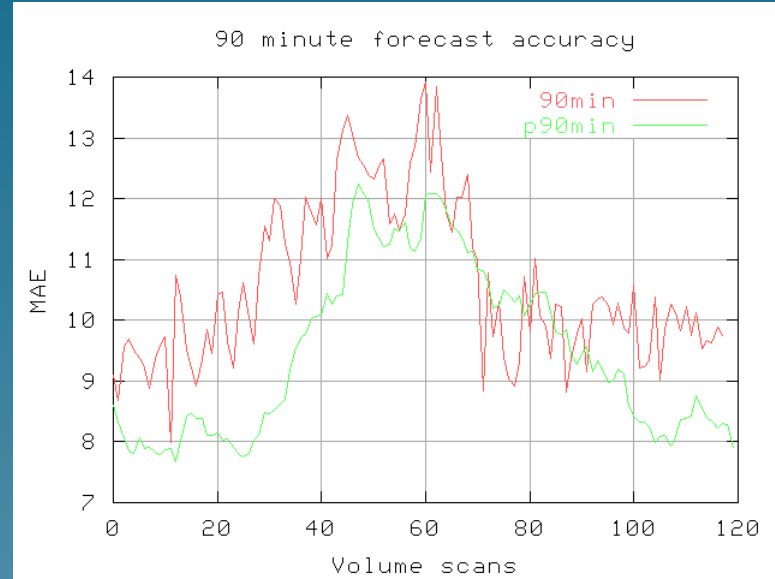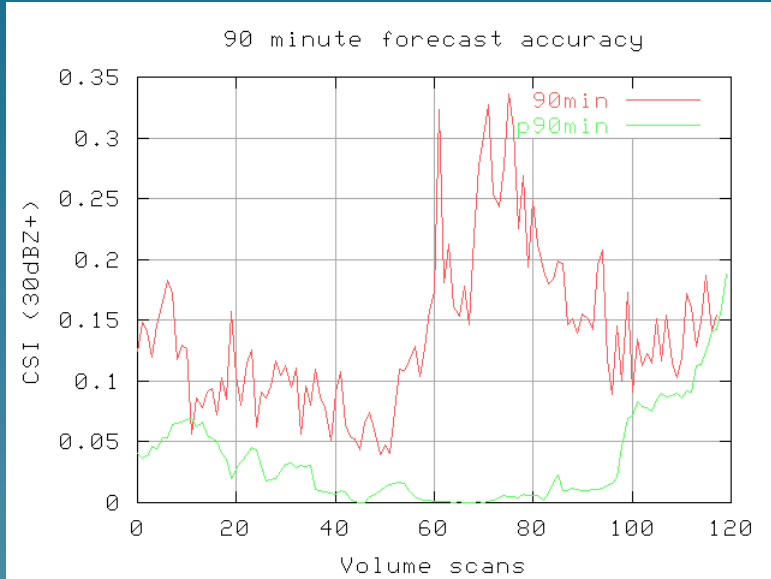
60 minute forecast

Note that our CSI remains high even at 60 minutes. Our mean absolute error performance is poor.

V Lakshmanan, lakshman@nssl.noaa.gov

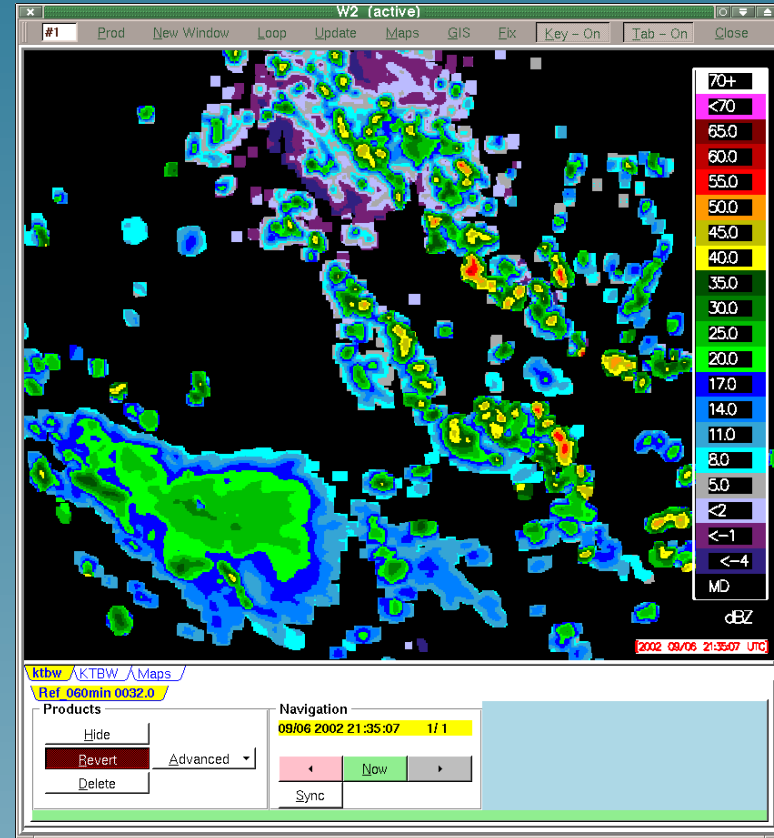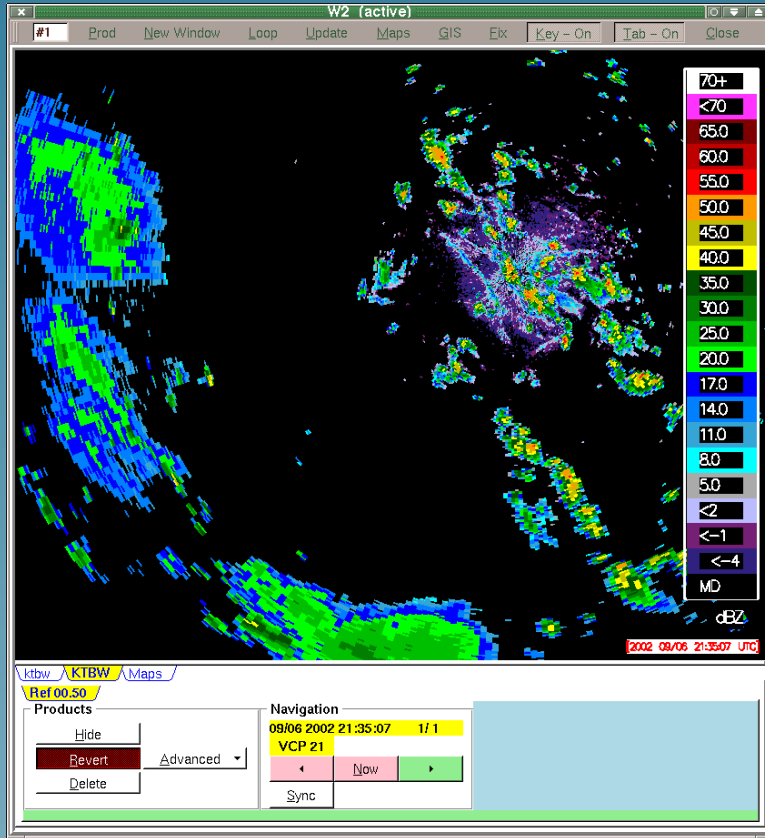# 90 and 120 minute forecast

V Lakshmanan, lakshman@nssl.noaa.gov

V Lakshmanan, lakshman@nssl.noaa.gov

Real-time data

Running real-time on Sep. 6, 2002 on KTBW (Tampa) LDM data feed.

V Lakshmanan, lakshman@nssl.noaa.gov

Opportunities

1. High bias – associated with splatting during forward projection.

2. Poor forecast of actual data values (high MAE), i.e. poor growth/decay estimate.

3. A better choice of scale for making forecasts?

4. Assimilation of mesoscale model wind speeds?

5. Use of Doppler radar velocity estimates?

6. Images look unrealistic beyond 60 minutes.

How can we improve upon this technique?