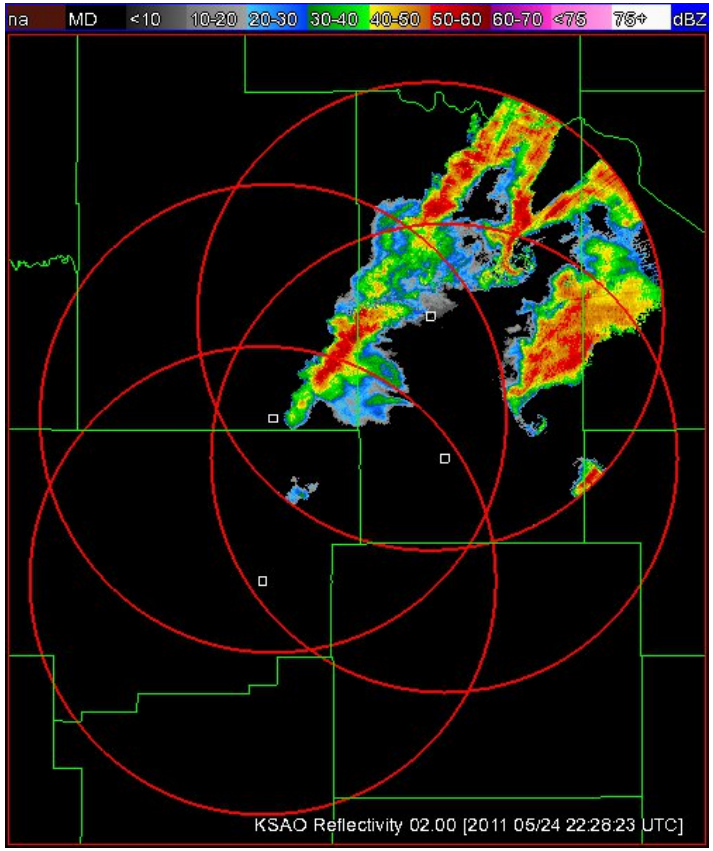
Unsupervised Detection of Bow Echoes in Dual Polarization Radar Data

Matthew Wiesner, Joseph Hardin, V. Chandrasekar

Project Motivation

- Facilitate historical radar data lookup
  - Search by feature not by date
- Adaptive Radar Sensing
- Prior work separates radar segmentation from feature identification
  - This work attempts to integrate both fields

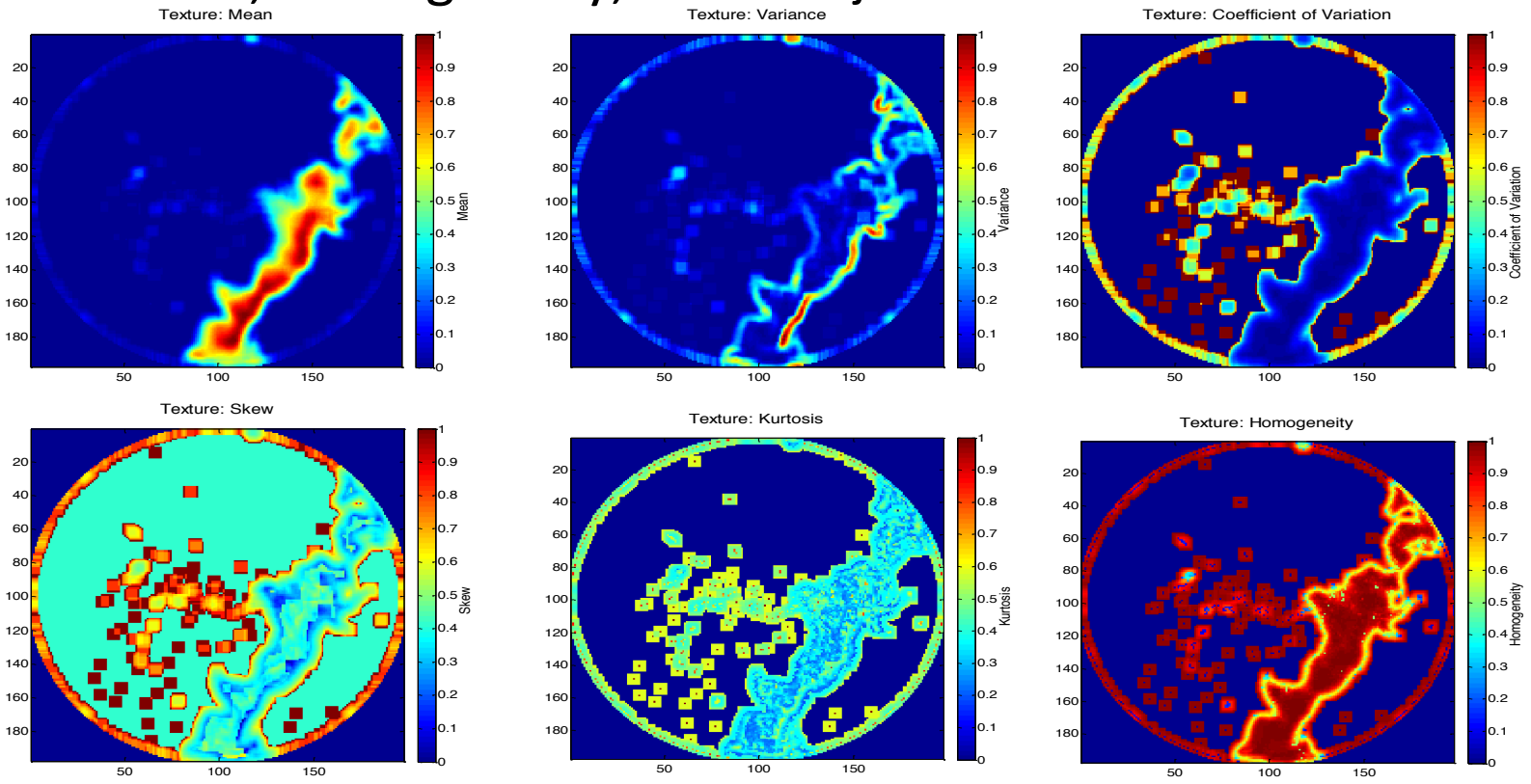


Methodology: Outline

- Segmentation (Lakshmanan) + Feature Detection
  1. Grid Radar Data (Image)
  2. Compute texture vectors at each image pixel
  3. Use k-means algorithm and flood-fill for hierarchical segmentation
  4. Principal Components Analysis (PCA) and 2nd order polynomial for feature detection

Texture

- Textures in Radar
  - Computed in a 7x7 neighborhood
  - T_{xy} = {mean, variance, coefficient of variation, skewness, kurtosis, homogeneity, contrast}

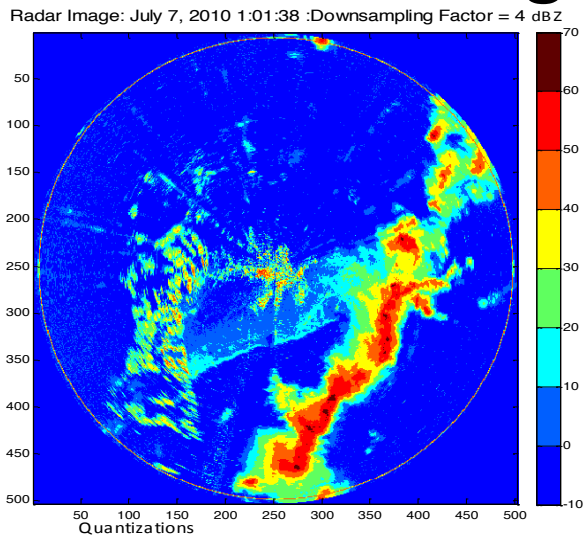


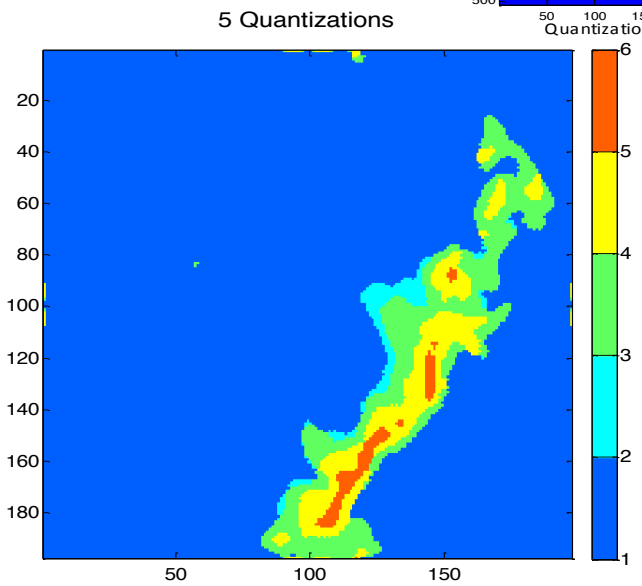
Texture Segmentation

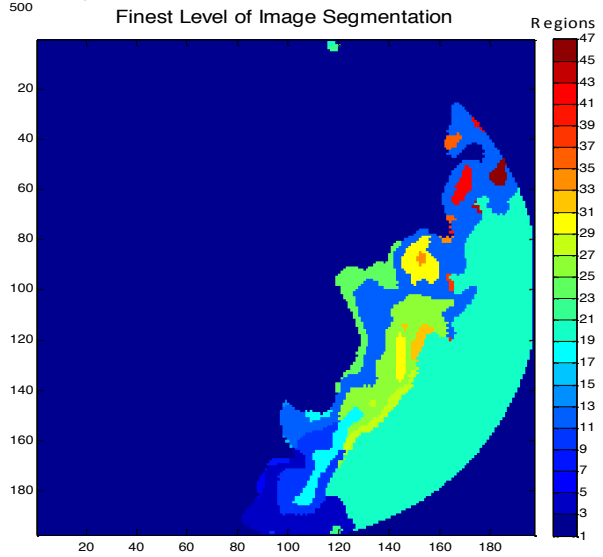
- Formal method of forming textures by k-means clustering (Lakshmanan):
  - Pixels should be clustered based on their texture values as done on the previous slide
  - Pixels should be clustered based on a Markov assumption that adjacent pixels will be members of the same texture
  - A pixel is assigned to the cluster which minimizes a cost function that accounts for both of the above

Texture Segmentation

- Results of k-means clustering for 5 textures

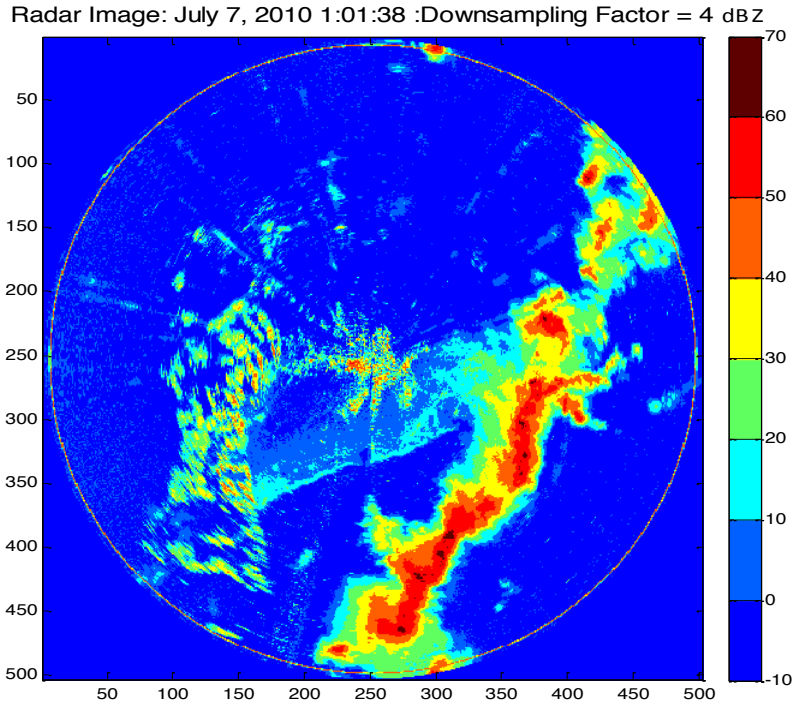


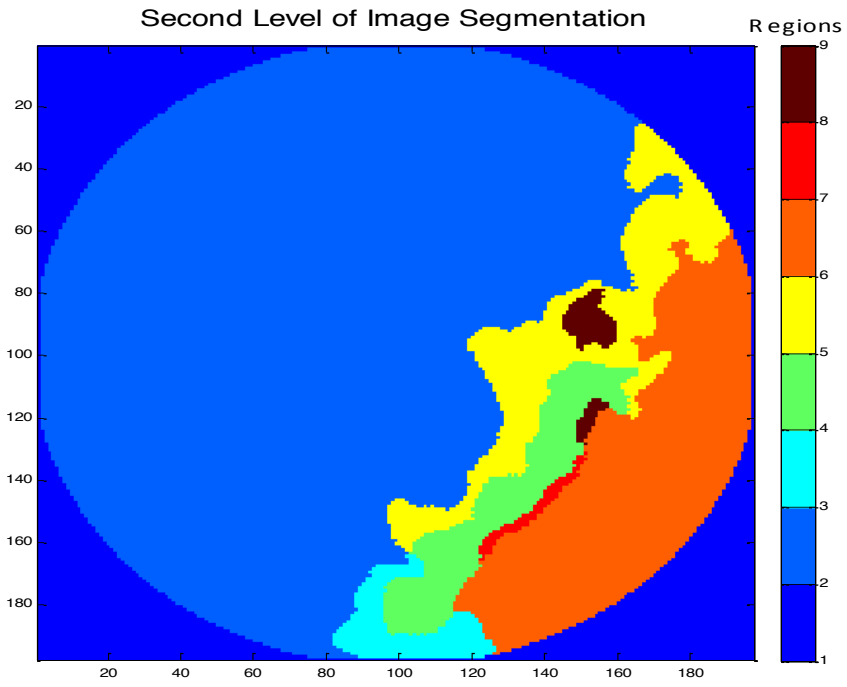




Segment Merging

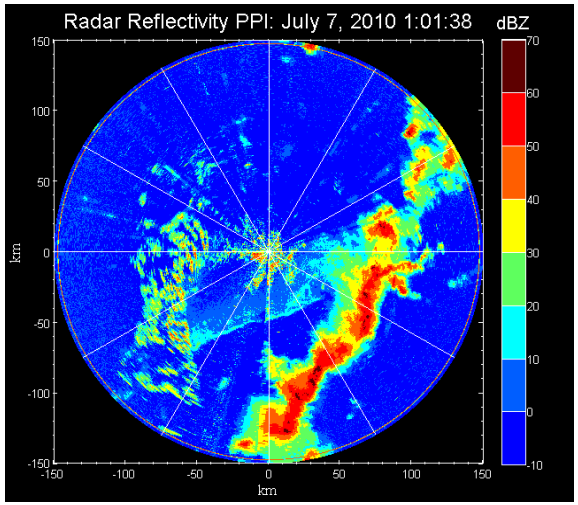
- Next highest level of segmentation hierarchy
- Merge Segments in order of texture space proximity

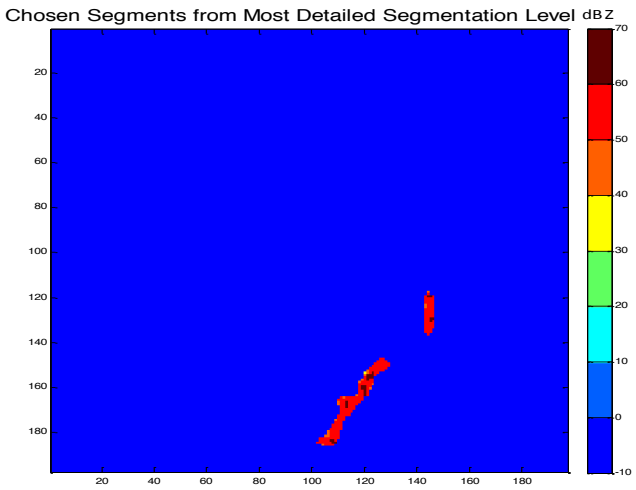


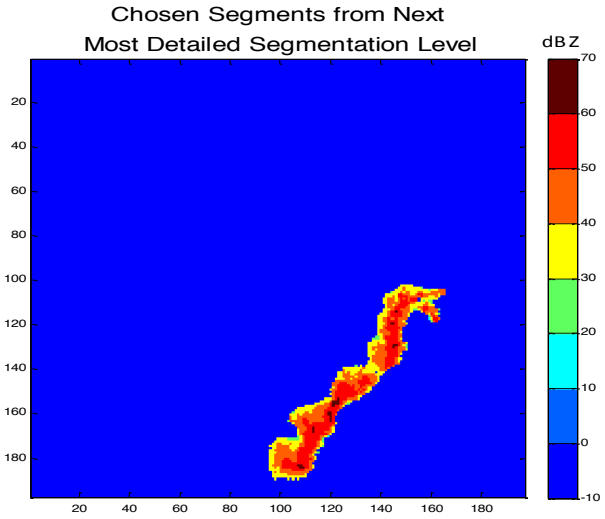


Candidate Segments

- Which segments are candidate bow echoes?
  - Segments should have a high mean reflectivity
  - Segments should not be too small
  - Examine each level in the hierarchical segmentation for candidate segments







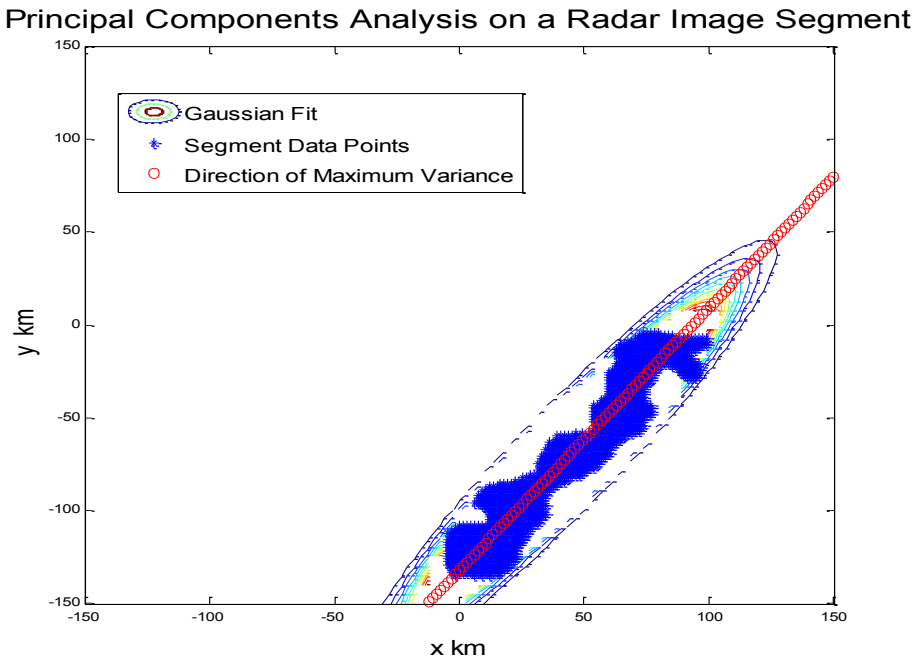
Bow Echo Conditions

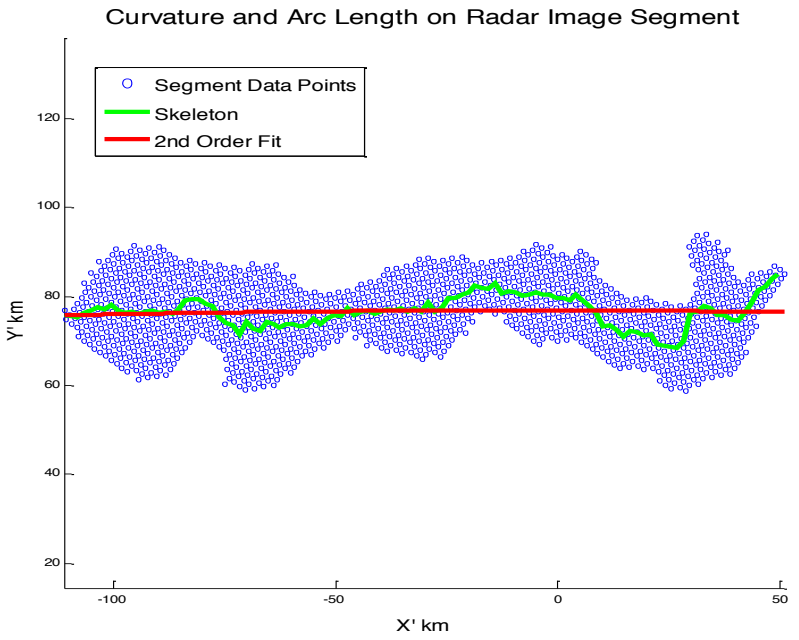
- Segment must be **LINEAR**
  - One large eigenvalue of covariance matrix (PCA)
- Segment must have noticeable **CURVATURE**
  - Relatively high weight of 2nd order term in quadratic fit
- Segment must have an **ARC LENGTH** longer than 20km
- The **THICKNESS** to length ratio of the segment must be low
  - A measure of “noise” around a skeleton of the bow echo
- If these conditions are met, a bow echo is detected

Bow Echo Detection

- Example of Linear, Non-Bow Echo Storm

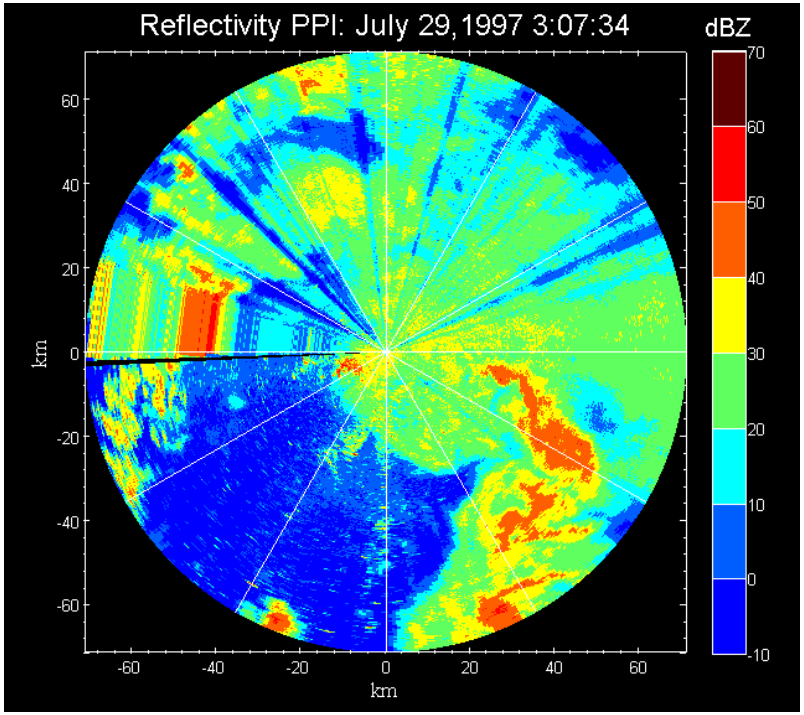
Curvature = 0.0162 Linearity = 0.94 arc length = 162.0088

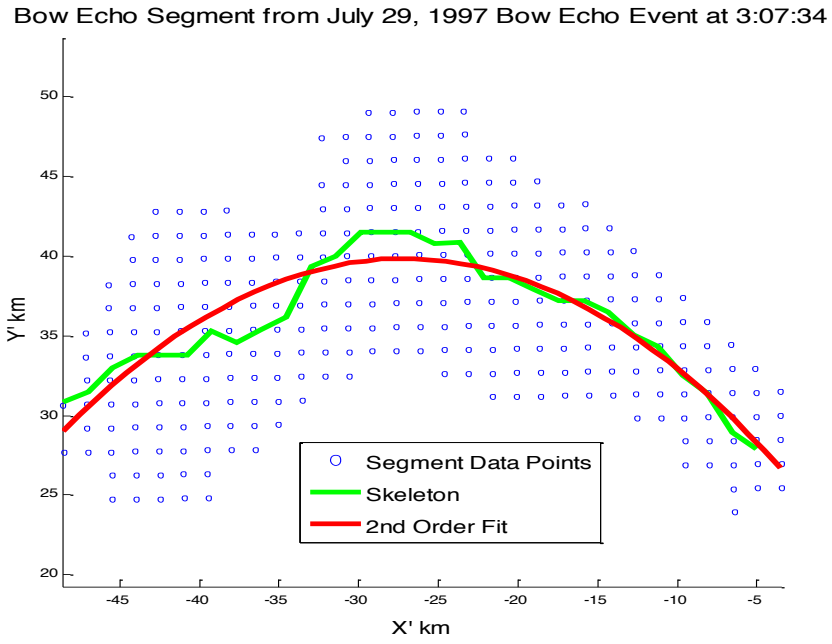




Bow Echo Detection

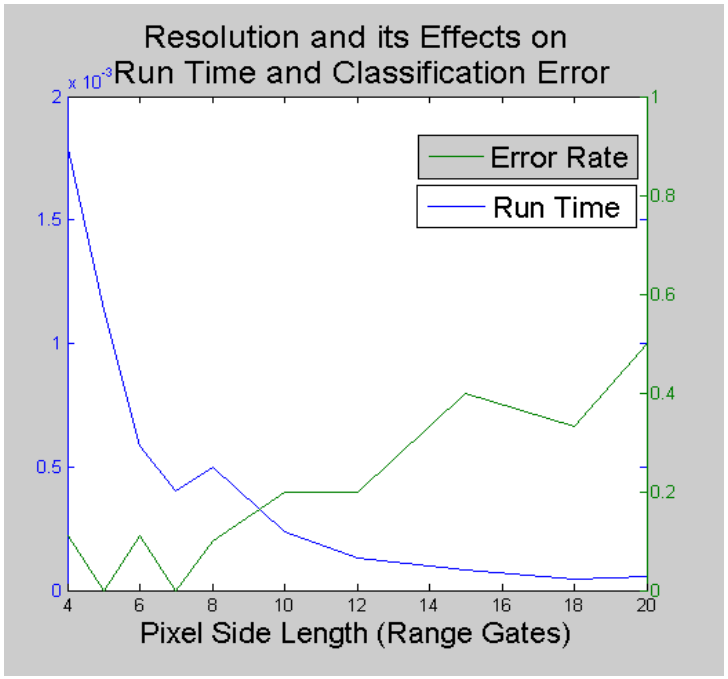
- A successfully detected bow echo meets all of the previous requirements as in the example shown below

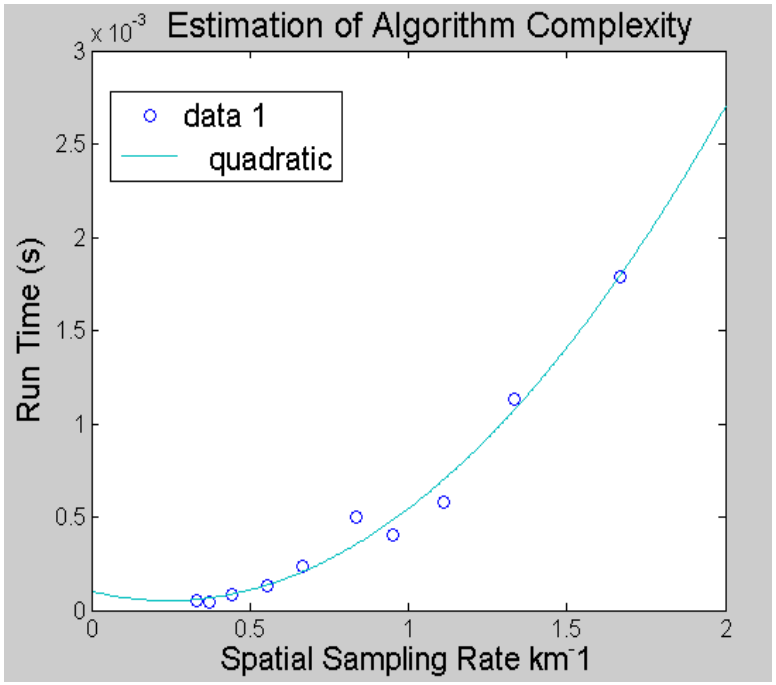


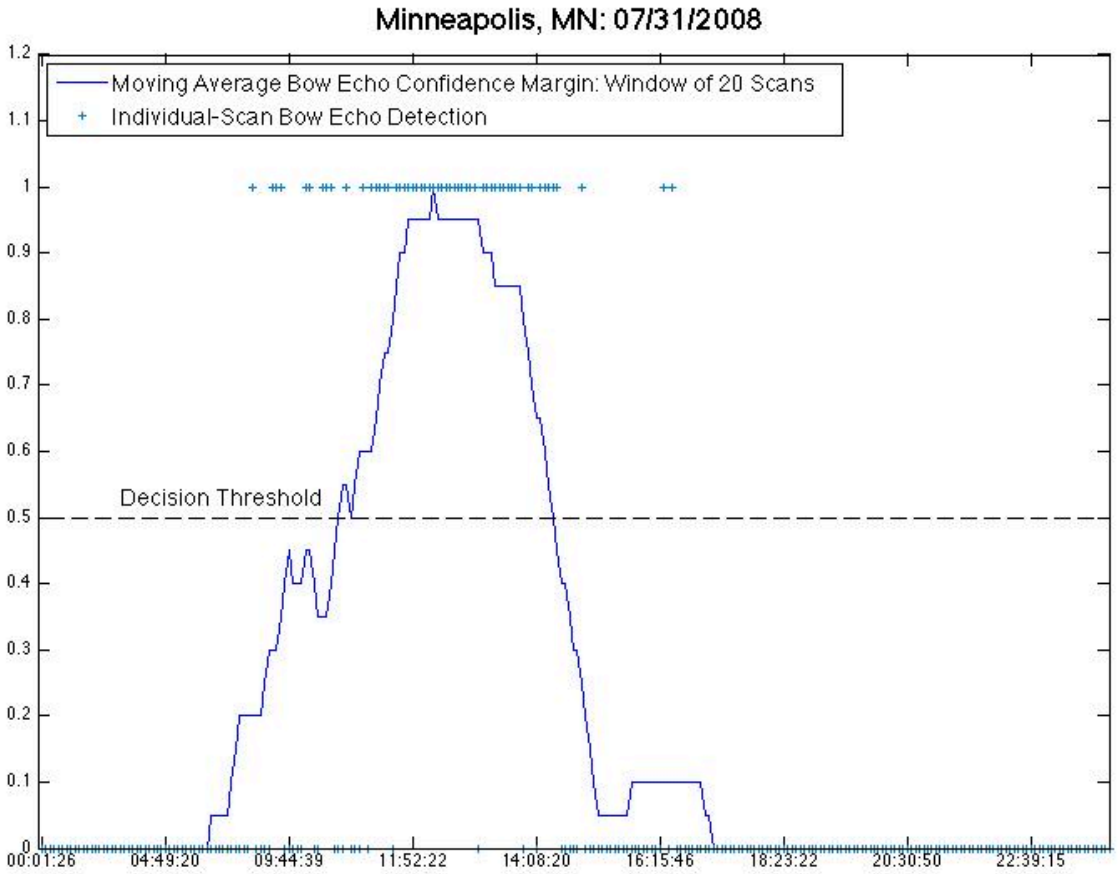


Performance

- Algorithm runtime scales poorly with resolution which limits accuracy
- Down-sampling factor depends on the size of the range gates (4 for 1 km, 15 for 0.25 km)
- Runtime/scan <= 1 min



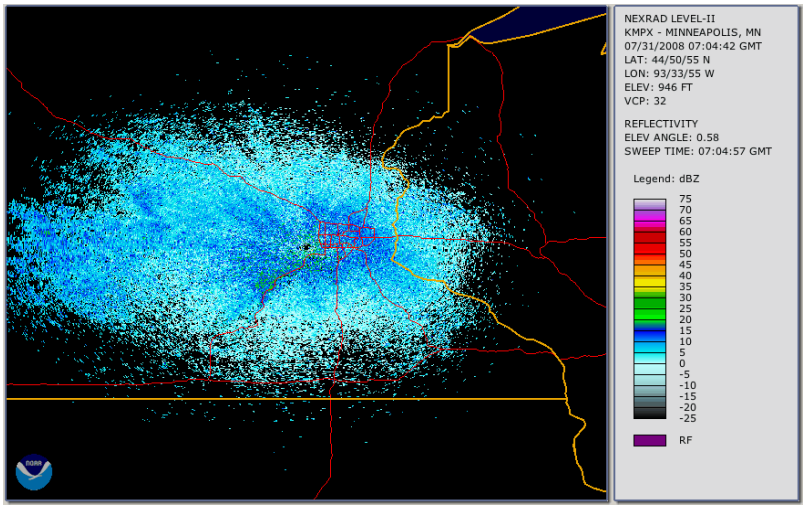


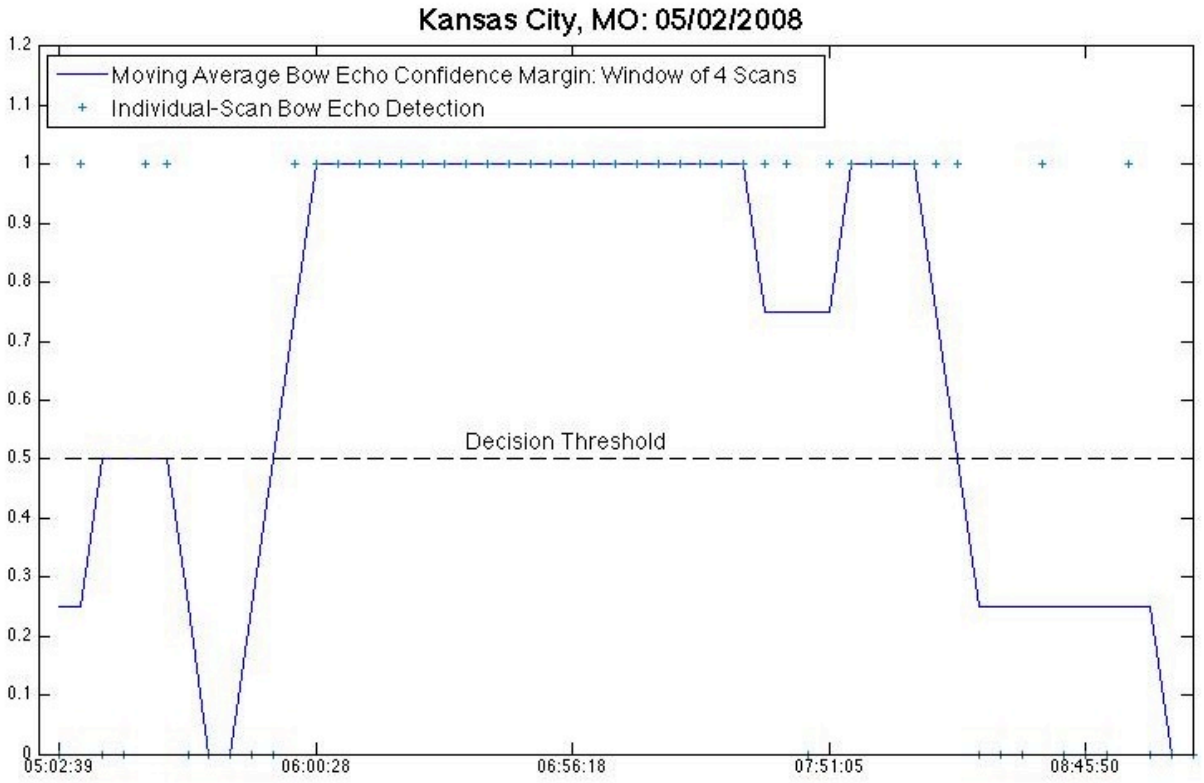


True Positive Rate: 89.6%

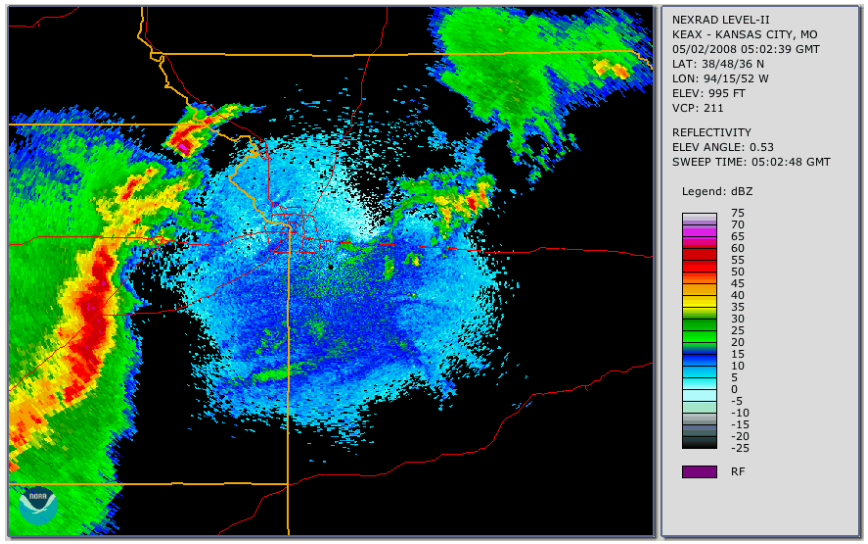
False Positive Rate: 6.13%

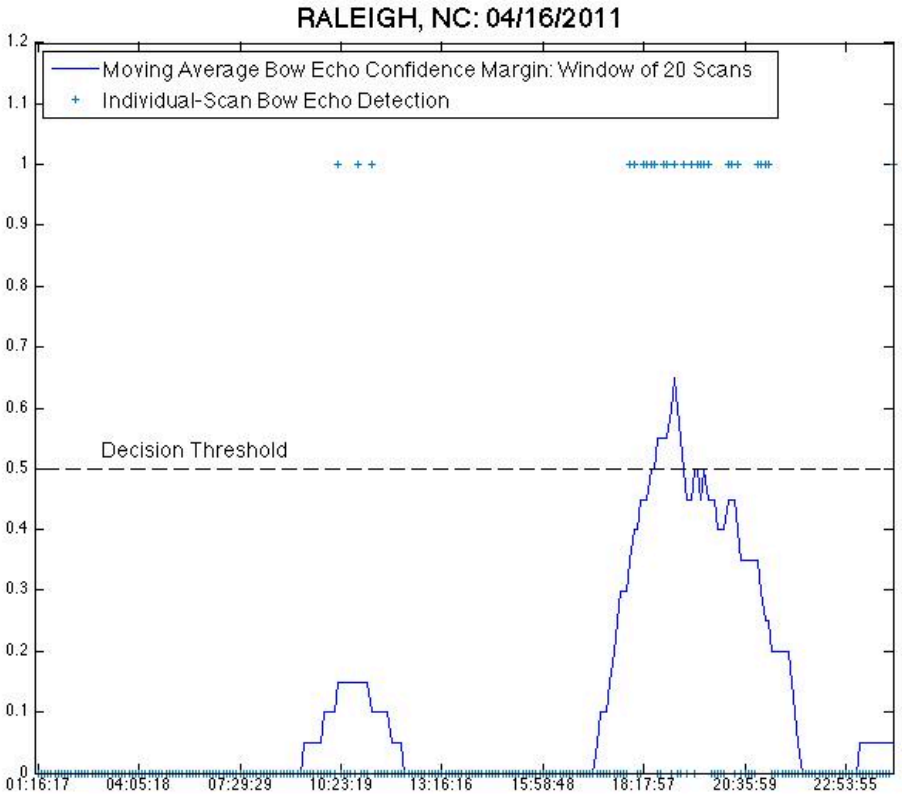
False Negative Rate: 1.04%





True Positive Rate: 78%
False Positive Rate: 3.07%
False Negative Rate: 2.19%

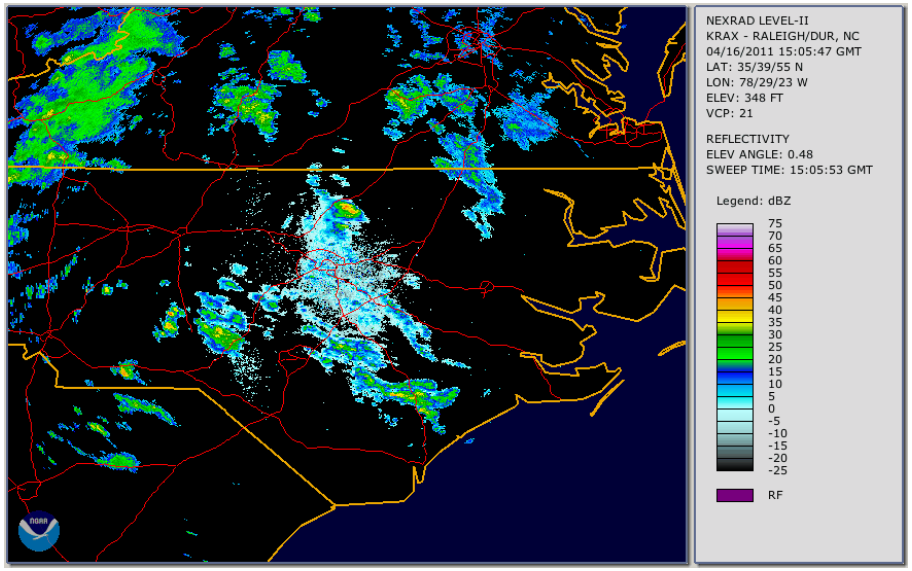




True Negative Rate : 89.8%

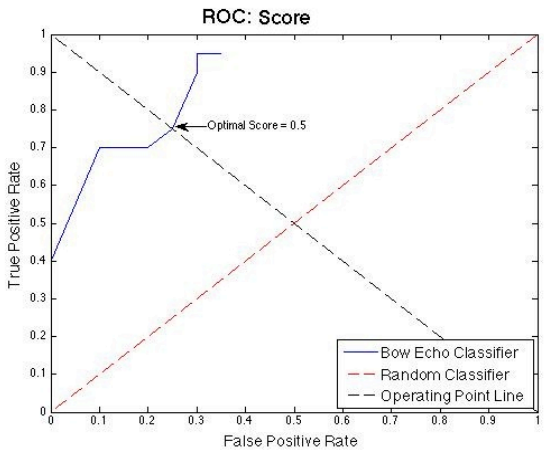
False Positive Rate : 10.2%

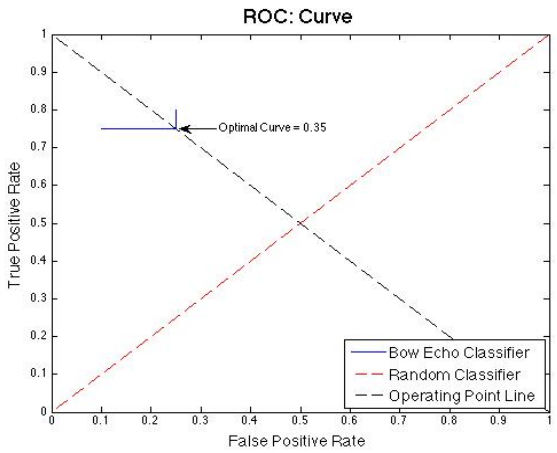
False Positive Rate in Thunderstorms: 28.2%

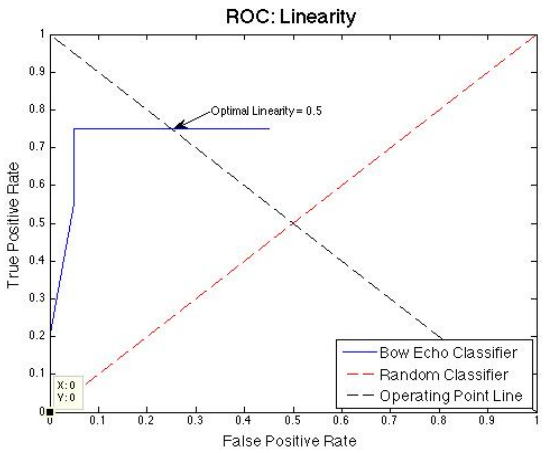


Performance

- Optimizing performance







- True Positive Rate is between 70 – 90% depending on parameter values
- False Positive Rate can be up to 30%
- Runtime per scan is <= 1 minute
- ROC curves indicate reasonable results for 1rst past classifier
- Temporal analysis yields more accuracy, and can be used to give confidence margins

Future Work

- Acceptable preformance but…
  - Improved detection with more sophisticated algorithms (Hough Transform for parabola detection)
  - Algorithm can be used to create a training corpus of bow echoes for use in supervised detection
  - Algorithm may run much quicker if optimized for speed