#### Evaluation

Peter Cornillon Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

# Evaluating HYCOM Using the Surface Frontal Field

### Peter Cornillon, Penelope Howe and David Ullman

Graduate School of Oceanography/University of Rhode Island

HYCOM Meeting 24-26 April 2007

▲□▶▲□▶▲□▶▲□▶ □ のQ@

# Overview

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

# Objective

To evaluate HYCOM as configured for the  $\frac{1}{12}^{\circ}$  North Atlantic runs.

▲□▶ ▲□▶ ▲ 三▶ ▲ 三▶ - 三 - のへぐ

# Overview

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

# Objective

To evaluate HYCOM as configured for the  $\frac{1}{12}^{\circ}$  North Atlantic runs.

# Approach

Compare the probability density of finding a surface SST front in 16x16 pixel regions in MODIS data with that of finding a surface front in HYCOM output.

▲□▶▲□▶▲□▶▲□▶ □ のQ@

# Overview

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

# Objective

To evaluate HYCOM as configured for the  $\frac{1}{12}^{\circ}$  North Atlantic runs.

### Approach

Compare the probability density of finding a surface SST front in 16x16 pixel regions in MODIS data with that of finding a surface front in HYCOM output.

### Assumption

That the surface frontal field is a good proxy for upper ocean dynamics.

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

# Outline

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

#### Edge Detection

Basic Idea Cloud Problem Solution

### Edge Detection

- Basic Idea
- Cloud Problem

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

Solution

# Outline

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

# Edge Detection Basic Idea Cloud Problem

Solution

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

#### Evaluation

- Peter Cornillon, Penelope Howe and David Ullman
- Edge Detection Basic Idea Cloud
- Problem Solution

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - Histograms are generated for overlapping 32 x 32 plast windows.
      - The coheston of the distributions establishes the existence of a front.
    - Plasis at the warmest local minimum between the two distributions define hontal series values.
  - The local level:
    - Beginning with the seed values, a contour following algorithm finds frontailers and states
    - A contour ends when the contour changes director by more than 80° in 5 plasts.

・ ロ ト ・ 雪 ト ・ 雪 ト ・ 日 ト

э.

A frontal segment must have 10 pixels

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

### Edge Detection Basic Idea

Problem

• We use an edge detection algorithm developed at URI by Jean-François Cayula

# • The algorithm operates at 2 levels

- The region level:
  - Histograms are generated for overlapping 32 × 32 pixel windows.
  - The presence of a warm and a cool distribution suggests the existence of a front.
  - The cohesion of the distributions establishes the existence of a from
  - Pixels at the warmest local minimum between the two distributions define frontal seed values.

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

### The local level:

- Beginning with the seed values a contour following algorithm finds frontal segments.
- A contour ends when the contour changes direction by more than 90° in 5 pixels.
- A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

### Edge Detection Basic Idea

• We use an edge detection algorithm developed at URI by Jean-François Cayula

# The algorithm operates at 2 levels

### • The region level:

- Histograms are generated for overlapping 32 × 32 pixel windows.
- The presence of a warm and a cool distribution suggests the existence of a front.
- The cohesion of the distributions establishes the existence of a front.
- Pixels at the warmest local minimum between the two distributions define frontal seed values.

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

### The local level:

- Beginning with the seed values a contour following algorithm finds frontal segments.
- A contour ends when the contour changes direction by more than 90° in 5 pixels.
- A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

### Edge Detection Basic Idea

Cloud Problem Solution

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - $\bullet~$  Histograms are generated for overlapping 32  $\times$  32 pixel windows.
    - The presence of a warm and a cool distribution suggests the existence of a front.
    - The cohesion of the distributions establishes the existence of a front.
    - Pixels at the warmest local minimum between the two distributions define frontal seed values.

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

- The local level:
  - Beginning with the seed values a contour following algorithm finds frontal segments.
  - A contour ends when the contour changes direction by more than 90° in 5 pixels.
  - A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

### Edge Detection Basic Idea

Cloud Problem Solution

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - Histograms are generated for overlapping 32  $\times$  32 pixel windows.
    - The presence of a warm and a cool distribution suggests the existence of a front.
    - The cohesion of the distributions establishes the existence of a front.
    - Pixels at the warmest local minimum between the two distributions define frontal seed values.
  - The local level:
    - Beginning with the seed values a contour following algorithm finds frontal segments.
    - A contour ends when the contour changes direction by more than 90° in 5 pixels.

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

### Edge Detection Basic Idea

Problem Solution

- We use an edge detection algorithm developed at URI by Jean-François Cayula
  - The algorithm operates at 2 levels
    - The region level:
      - Histograms are generated for overlapping 32  $\times$  32 pixel windows.
      - The presence of a warm and a cool distribution suggests the existence of a front.
      - The cohesion of the distributions establishes the existence of a front.
      - Pixels at the warmest local minimum between the two distributions define frontal seed values.
    - The local level:
      - Beginning with the seed values a contour following algorithm finds frontal segments.
      - A contour ends when the contour changes direction by more than 90° in 5 pixels.

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

• A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

#### Edge Detection Basic Idea Cloud

- tion
- asic Idea oud oblem

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - Histograms are generated for overlapping 32  $\times$  32 pixel windows.
    - The presence of a warm and a cool distribution suggests the existence of a front.
    - The cohesion of the distributions establishes the existence of a front.
    - Pixels at the warmest local minimum between the two distributions define frontal seed values.

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

### The local level:

- Beginning with the seed values a contour following algorithm finds frontal segments.
- A contour ends when the contour changes direction by more than 90° in 5 pixels.
- A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

#### Edge Detection Basic Idea Cloud

Problem Solution

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - Histograms are generated for overlapping 32  $\times$  32 pixel windows.
    - The presence of a warm and a cool distribution suggests the existence of a front.
    - The cohesion of the distributions establishes the existence of a front.
    - Pixels at the warmest local minimum between the two distributions define frontal seed values.

### • The local level:

- Beginning with the seed values a contour following algorithm finds frontal segments.
- $\bullet\,$  A contour ends when the contour changes direction by more than  $90^\circ$  in 5 pixels.

◆□▶ ◆□▶ ▲□▶ ▲□▶ □ のQ@

• A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

#### Edge Detection Basic Idea Cloud Problem

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - Histograms are generated for overlapping 32  $\times$  32 pixel windows.
    - The presence of a warm and a cool distribution suggests the existence of a front.
    - The cohesion of the distributions establishes the existence of a front.
    - Pixels at the warmest local minimum between the two distributions define frontal seed values.

- The local level:
  - Beginning with the seed values a contour following algorithm finds frontal segments.
  - $\bullet\,$  A contour ends when the contour changes direction by more than  $90^\circ$  in 5 pixels.
  - A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

#### Edge Detection Basic Idea Cloud Problem

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - Histograms are generated for overlapping 32  $\times$  32 pixel windows.
    - The presence of a warm and a cool distribution suggests the existence of a front.
    - The cohesion of the distributions establishes the existence of a front.
    - Pixels at the warmest local minimum between the two distributions define frontal seed values.

- The local level:
  - Beginning with the seed values a contour following algorithm finds frontal segments.
  - A contour ends when the contour changes direction by more than  $90^\circ$  in 5 pixels.
  - A frontal segment must have 10 pixels.

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

#### Edge Detection Basic Idea Cloud Problem

- We use an edge detection algorithm developed at URI by Jean-François Cayula
- The algorithm operates at 2 levels
  - The region level:
    - Histograms are generated for overlapping 32  $\times$  32 pixel windows.
    - The presence of a warm and a cool distribution suggests the existence of a front.
    - The cohesion of the distributions establishes the existence of a front.
    - Pixels at the warmest local minimum between the two distributions define frontal seed values.

- The local level:
  - Beginning with the seed values a contour following algorithm finds frontal segments.
  - A contour ends when the contour changes direction by more than  $90^\circ$  in 5 pixels.
  - A frontal segment must have 10 pixels.

# The Probability Fields

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

- The statistic that we use to evaluate HYCOM is the probability of finding a front in a given region
- The frontal probability is defined by:

 $\frac{\sum_{i=1}^{N} \text{Front}_{i}}{\sum_{i=1}^{N} \text{Clear}_{i}}$ 

◆□▶ ◆□▶ ▲□▶ ▲□▶ ■ ののの

- i is over all pixels in a spatial and temporal region.
- Front, is 1 if the pixel is a front pixel, 0 otherwise and
- Clear<sub>i</sub> is 1 if the pixel is clear, 0 otherwise
- To obtain good statistics, probabilities are calculated for:
  - 16 × 16 pixel regions
  - For all fields in 2004; one per day.
  - I.e., N = 16 × 16 × 366 = 93,696.

# The Probability Fields

#### Evaluation

- Peter Cornillon, Penelope Howe and David Ullman
- Edge Detection Basic Idea Cloud Problem Solution

- The statistic that we use to evaluate HYCOM is the probability of finding a front in a given region
- The frontal probability is defined by:

$$\frac{\sum_{i=1}^{N} \text{Front}_{i}}{\sum_{i=1}^{N} \text{Clear}_{i}}$$

- *i* is over all pixels in a spatial and temporal region,
- Front, is 1 if the pixel is a front pixel, 0 otherwise and
- Clear<sub>i</sub> is 1 if the pixel is clear, 0 otherwise.
- To obtain good statistics, probabilities are calculated for:
  - 16 × 16 pixel regions
  - For all fields in 2004; one per day.
  - I.e., N = 16 × 16 × 366 = 93,696.

# The Probability Fields

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

- The statistic that we use to evaluate HYCOM is the probability of finding a front in a given region
- The frontal probability is defined by:

$$\frac{\sum_{i=1}^{N} \text{Front}_{i}}{\sum_{i=1}^{N} \text{Clear}_{i}}$$

- *i* is over all pixels in a spatial and temporal region,
- Front, is 1 if the pixel is a front pixel, 0 otherwise and
- Clear<sub>i</sub> is 1 if the pixel is clear, 0 otherwise.
- To obtain good statistics, probabilities are calculated for:
  - 16 × 16 pixel regions
  - For all fields in 2004; one per day.
  - I.e.,  $N = 16 \times 16 \times 366 = 93,696$ .

# Outline

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem We began by calculating frontal probability for HYCOM

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ ─臣 ─のへで

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

### We began by calculating frontal probability for HYCOM

▲□▶ ▲□▶ ▲□▶ ▲□▶ ▲□ ● のへぐ



Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

### We began by calculating frontal probability for HYCOM and MODIS 2004





Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem

### We began by calculating frontal probability for HYCOM and MODIS 2004



Some similarities, but big differences, especially in the center of the gyre.

▲□▶▲□▶▲□▶▲□▶ □ のQ@

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem

- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem

### • The cloud field varies significantly over this region.





### • We correct for cloudiness by dividing by the number of clear pixles,

- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem



- We correct for cloudiness by dividing by the number of clear pixles,
- But the edge detection algorithm relies on histograms
- To calculate a histogram
  - More than 10% of the pixels must be clear, and
  - At least 25% of the clear pixels must be in smallest population.
  - Many 16x16 pixel regions with some clear pixels will fail these tests.
- Clouds will result in fronts not being detected at clear pixel locations.
- The fronts found are still good, but the algorithm underestimates probabilities.

# Outline

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

# Edge Detection

Basic Idea

Cloud Problem

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへぐ

Solution

◆□▶ ◆□▶ ▲□▶ ▲□▶ □ のQ@

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

- Map MODIS clouds onto HYCOM SST images.
- Run the edge detector on the 'cloudy' HYCOM fields.
- Compute the front probability field from the results, *F<sub>Cloudy</sub>*.
- Compare F<sub>Cloudy</sub>

◆□▶ ◆□▶ ▲□▶ ▲□▶ □ のQ@

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

- Map MODIS clouds onto HYCOM SST images.
- Run the edge detector on the 'cloudy' HYCOM fields.
- Compute the front probability field from the results, *F<sub>Cloudy</sub>*.
- Compare F<sub>Cloudy</sub>

◆□▶ ◆□▶ ▲□▶ ▲□▶ □ のQ@

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

- Map MODIS clouds onto HYCOM SST images.
- Run the edge detector on the 'cloudy' HYCOM fields.
- Compute the front probability field from the results, *F<sub>Cloudy</sub>*.
- Compare F<sub>Cloudy</sub>

◆□▶ ◆□▶ ▲□▶ ▲□▶ □ のQ@

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

- Simply calculate fronts obtained from HYCOM with the MODIS clouds imposed on SST field, F<sub>Cloudy</sub>.
- Map MODIS clouds onto HYCOM SST images.
- Run the edge detector on the 'cloudy' HYCOM fields.
- Compute the front probability field from the results, *F<sub>Cloudy</sub>*.
- Compare F<sub>Cloudy</sub>

◆□▶ ◆□▶ ▲□▶ ▲□▶ □ のQ@

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

- Map MODIS clouds onto HYCOM SST images.
- Run the edge detector on the 'cloudy' HYCOM fields.
- Compute the front probability field from the results, *F<sub>Cloudy</sub>*.
- Compare F<sub>Cloudy</sub>



### Evaluation

Solution

 Simply calculate fronts obtained from HYCOM with the MODIS clouds imposed on SST field, F<sub>Cloudy</sub>.

- Map MODIS clouds onto HYCOM SST images.
- Run the edge detector on the 'cloudy' HYCOM fields.
- Compute the front probability field from the results,  $F_{Cloudy}$ .
- Compare F<sub>Cloudy</sub> with F<sub>MODIS</sub>





#### MODIS Front Probability 2004

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution  Simply calculate fronts obtained from HYCOM with the MODIS clouds imposed on SST field, F<sub>Cloudy</sub>.

- Map MODIS clouds onto HYCOM SST images.
- Run the edge detector on the 'cloudy' HYCOM fields.
- Compute the front probability field from the results, *F<sub>Cloudy</sub>*.
- Compare F<sub>Cloudy</sub> with F<sub>MODIS</sub>





HYCOM Front Probability 2004



MODIS Front Probability 2004

# Frontal Probability Differences

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution • Of more interest is the difference between the HYCOM frontal structure and that of MODIS.



HYCOM minus MODIS Front Probability 2004

◆□ ▶ ◆□ ▶ ◆三 ▶ ◆□ ▶ ◆□ ●

# Frontal Probability Differences

10

20

30

40

50

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution  Of more interest is the difference between the HYCOM frontal structure and that of MODiS. Median filtered.

HYCOM minus MODIS Front Probability (Median Filtered) 2004 0.025 0.02 10 0.015 20 0.01 0.005 30 0 40 -0.005 -0.01 50 -0.015 60 -0.02

・ ロ ト ・ 国 ト ・ 国 ト ・ 国 ・ つ へ ()

60

# Frontal Probability Differences

#### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution • Of more interest is the difference between the HYCOM frontal structure and that of MODiS. Median filtered.

HYCOM minus MODIS Front Probability (Median Filtered) 2004



# All Done

### Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

# The End

◆□▶ ◆□▶ ◆ □▶ ◆ □▶ ● □ ● ● ● ●

# Median Filtered HYCOM - MODIS Fronl Probability 2004

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution

Median Filtered HYCOM-MODIS Front Probability 2004 450 400 Mean = 0.0064 350 300 250 200 150 100 50 -0.03 -0.02 -0.01 0 0.01 0.02 0.03

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 のへぐ

# MODIS Front Probability 2004

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution



◆□▶ ◆□▶ ◆豆▶ ◆豆▶ □ のへで

# HYCOM Front Probability 2004

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution



▲□▶ ▲圖▶ ▲臣▶ ▲臣▶ ―臣 … のへで

# Filtered HYCOM-MODIS Front Probability vs Cloud Cover

Evaluation

Peter Cornillon, Penelope Howe and David Ullman

Edge Detection Basic Idea Cloud Problem Solution



◆ロト★園と★臣と★臣と「臣」のな(で)