Facilitating the Use of the MODIS Aqua L2 SST Dataset

Peter Cornillon

Graduate School of Oceanography, University of Rhode Island, Narragansett, RI 02882, USA

GHRSST XXIII

Barcelona, Spain

28 June 2022

Acknowledgments

- NASA grant # 80NSSC18K0837 via a subcontract from the Farallon Institute.
- NASA grant # NNH18ZDA001N-RST.
- NASA grant #20-TWSC20-2-0003 for the Openscapes project.

Outline

1 Motivation

2 The Problems And Their Solutions

3 Putting it All Together

4 Summary

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:
  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...

- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.

- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.

- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.

  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.

- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...

- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.

- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.

- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.

  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.

- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...

- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.

- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.

- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.

  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.

- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.

- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:
  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...

- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.

- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.

- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.

  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.

- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.

- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:
  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...

- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32 $\times$ 32 pixel regions.

- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3 $\times$ 3 pixel kernel.

- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:
  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:
  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:
  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32 $\times$ 32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3 $\times$ 3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:

  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...

- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.

- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.

- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

SST Fronts and Gradients

- Features associated with SST fronts are of significant interest in oceanography.

  They are important:
  - In the up-scale and down-scale flux of energy in the upper ocean.
  - As boundaries of water masses.
  - In a broad range of biological processes.
  - ...
- I study features with SST fronts identified with JF Cayula's edge detection algorithm

  A population-based algorithm operating on 32×32 pixel regions.
- And with gradients obtained from the Sobel Gradient operator

  Operating on a 3×3 pixel kernel.
- My preference is to use Level 2 (L2) as opposed to L3 or L4 data.
  - L2 data preserve the spatial relationship of pixels.
  - Higher levels of processing do not.
- I'm interested in applying these algorithms to long, global, high-resolution SST datasets.
- But such datasets (L2, global) can be hard to use for a number of reasons.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite

- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite

- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite

- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

The Dataset

- The longest, global, 1 km datasets are those obtained from MODIS.
- The primary MODIS L2 SST datasets available in GHRSST format are the R19 version:

  GHRSST Level 2P Global Sea Surface Skin Temperature from the MODIS on the NASA Aqua/Terra satellite
- These datasets
  - Were processed by the OBPG at NASA's GSFC
  - Using the retrieval algorithm developed by the Remote Sensing Group at the UMiami.
- The Aqua dataset consists of
  - Approximately 2 million 5-minute granules.
  - From mid-2002 to present.
  - Global coverage approximately twice daily.
  - Spatial resolution approximately 1 km at nadir.
  - 10 detectors per mirror rotation.

OBPG - Ocean Biology Processing Group
GSFC - NASA's Goddard Space Flight Center

5-minute granule corresponds to the data collected by the data in 5 minutes.

Outline

1. Motivation
2. The Problems And Their Solutions
3. Putting it All Together
4. Summary

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags e.g., low quality pixels flagged as good but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags e.g., low quality pixels flagged as good but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags e.g., low quality pixels flagged as good but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags e.g., low quality pixels flagged as good but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags
      e.g., low quality pixels flagged as good
      but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags e.g., low quality pixels flagged as good but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags
      e.g., low quality pixels flagged as good
      but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags e.g., low quality pixels flagged as good but I will not address these issues here.

Problems Impacting the Derivation of Fronts and Gradients Datasets

- Three basic problem areas related to fronts & gradients with the MODIS R19 datasets:
  - Granularization
  - Bow-Tie Effect
  - Quality Mask
    - Reference Temperature
    - High Gradient Regions
    - There are other issues with the quality flags
      e.g., low quality pixels flagged as good
      but I will not address these issues here.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32 $\times$ 32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32 $\times$ 32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32×32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32 $\times$ 32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32×32 pixel regions.
    - These regions are referred to as windows.
    - It begins by locating frontal candidates in the window based on pixel populations.
    - It then drops to the pixel level and contour follows gradients from the candidates.
        - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
    - It requires a minimal overlap between adjacent granules of one window.
        - Which means augmenting the granules [illegible]
    - Following location of fronts, frontal segments must be joined across the granule seams.
        - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32×32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32 $\times$ 32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32×32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32 $\times$ 32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32×32 pixel regions.
  - These regions are referred to as windows.
  - It begins by locating frontal candidates in the window based on pixel populations.
  - It then drops to the pixel level and contour follows gradients from the candidates.
    - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
  - It requires a minimal overlap between adjacent granules of one window.
    - Which means augmenting the granules before working on them.
  - Following location of fronts, frontal segments must be joined across the granule seams.
    - It turns out that this is not straightforward.

Granularization – The Problem

- MODIS orbits are divided into 5-minute granules.
- The Cayula-Cornillon edge detection algorithm operates on 32×32 pixel regions.
    - These regions are referred to as windows.
    - It begins by locating frontal candidates in the window based on pixel populations.
    - It then drops to the pixel level and contour follows gradients from the candidates.
        - Fronts may extend beyond the window edges!
- For these reasons matching fronts between granules is problematic.
    - It requires a minimal overlap between adjacent granules of one window.
        - Which means augmenting the granules before working on them.
    - Following location of fronts, frontal segments must be joined across the granule seams.
        - It turns out that this is not straightforward.

Granularization – The Solution

- Granules are merged into complete orbits.
- Orbits are started with the nadir pixel nearest 78° S on the descending part of the orbit.
  - Choosing this latitude minimizes the ocean area where orbits begin.
    i.e., it reduces the seam problem at the beginning/end of an orbit.
- Orbits overlap by 100 pixels.
  - This allows the user to join frontal segments between orbits.
    This is only an issue for a small portion of the Southern Ocean.

Granularization – The Solution

- Granules are merged into complete orbits.
- Orbits are started with the nadir pixel nearest 78°S on the descending part of the orbit.
  - Choosing this latitude minimizes the ocean area where orbits begin.
    i.e., it reduces the seam problem at the beginning/end of an orbit.
- Orbits overlap by 100 pixels.
  - This allows the user to join frontal segments between orbits.
    This is only an issue for a small portion of the Southern Ocean.

Granularization – The Solution

- Granules are merged into complete orbits.
- Orbits are started with the nadir pixel nearest 78°S on the descending part of the orbit.
  - Choosing this latitude minimizes the ocean area where orbits begin.
    i.e., it reduces the seam problem at the beginning/end of an orbit.
- Orbits overlap by 100 pixels.
  - This allows the user to join frontal segments between orbits.
    This is only an issue for a small portion of the Southern Ocean.

Granularization – The Solution

- Granules are merged into complete orbits.
- Orbits are started with the nadir pixel nearest 78°S on the descending part of the orbit.
  - Choosing this latitude minimizes the ocean area where orbits begin.
    i.e., it reduces the seam problem at the beginning/end of an orbit.
- Orbits overlap by 100 pixels.
  - This allows the user to join frontal segments between orbits.
    This is only an issue for a small portion of the Southern Ocean.

Granularization – The Solution

- Granules are merged into complete orbits.
- Orbits are started with the nadir pixel nearest 78° S on the descending part of the orbit.
  - Choosing this latitude minimizes the ocean area where orbits begin.
    i.e., it reduces the seam problem at the beginning/end of an orbit.
- Orbits overlap by 100 pixels.
  - This allows the user to join frontal segments between orbits.
    This is only an issue for a small portion of the Southern Ocean.

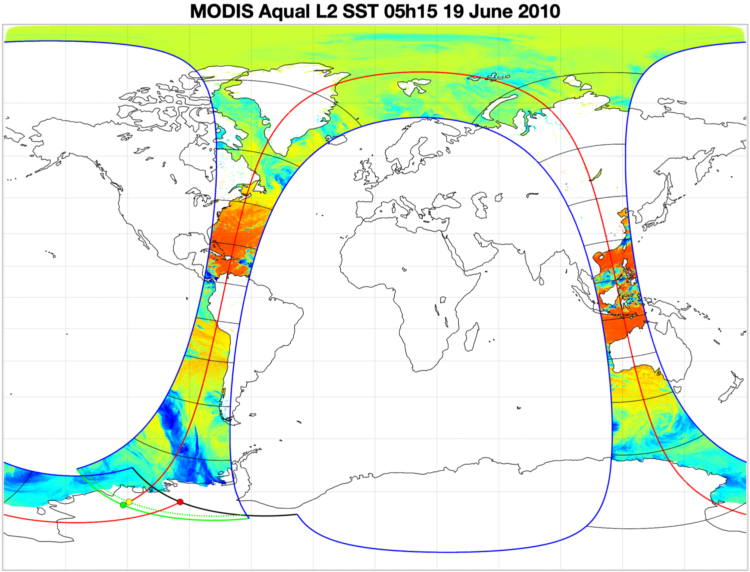
Granularization – The Solution

- Granules are merged into complete orbits.
- Orbits are started with the nadir pixel nearest 78°S on the descending part of the orbit.
  - Choosing this latitude minimizes the ocean area where orbits begin.
    i.e., it reduces the seam problem at the beginning/end of an orbit.
- Orbits overlap by 100 pixels.
  - This allows the user to join frontal segments between orbits.
    This is only an issue for a small portion of the Southern Ocean.

Granularization – The Solution

- Granules are merged into complete orbits.
- Orbits are started with the nadir pixel nearest 78°S on the descending part of the orbit.
  - Choosing this latitude minimizes the ocean area where orbits begin.
    i.e., it reduces the seam problem at the beginning/end of an orbit.
- Orbits overlap by 100 pixels.
  - This allows the user to join frontal segments between orbits.
    This is only an issue for a small portion of the Southern Ocean.

Granularization – The Solution – 20 Granules ⇒ 1 orbit



Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir
    with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir
    with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
    - The along-scan size of a pixel increases substantially
    - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
    - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
    - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
    - The along-scan properties remain the same, however
    - The along-track separation of pixels depends on position in the detector group.
        - Detector groups consist of 10 detectors for MODIS
    - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

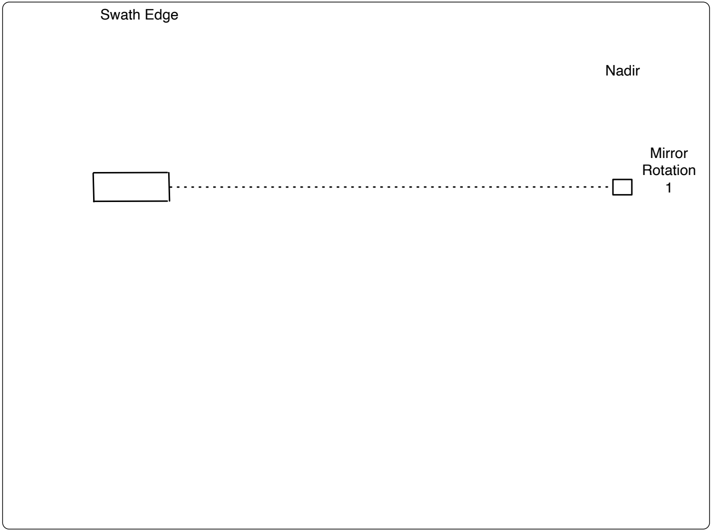
Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

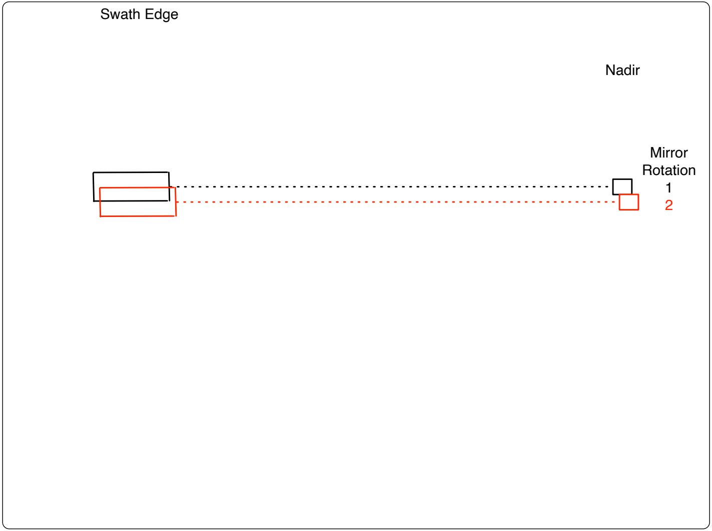
Bow-Tie Effect – The Problem

- As the distance from nadir increases the size & shape of pixels on a scan line changes:
  - The along-scan size of a pixel increases substantially
  - The along-track size of a pixel increases slightly
- For a single detector scanner (e.g., AVHRR) this is not a major problem:
  - The along-scan separation of pixels increases away from nadir with adjacent along-scan pixels touching.
  - The along-track separation of pixels is independent of distance from nadir with some overlap of adjacent along-track pixels.
- For multi-detector scanners this is a problem.
  - The along-scan properties remain the same, however
  - The along-track separation of pixels depends on position in the detector group.
    - Detector groups consist of 10 detectors for MODIS.
  - This effect becomes significant less than 1/2 way from nadir to the swath edge!

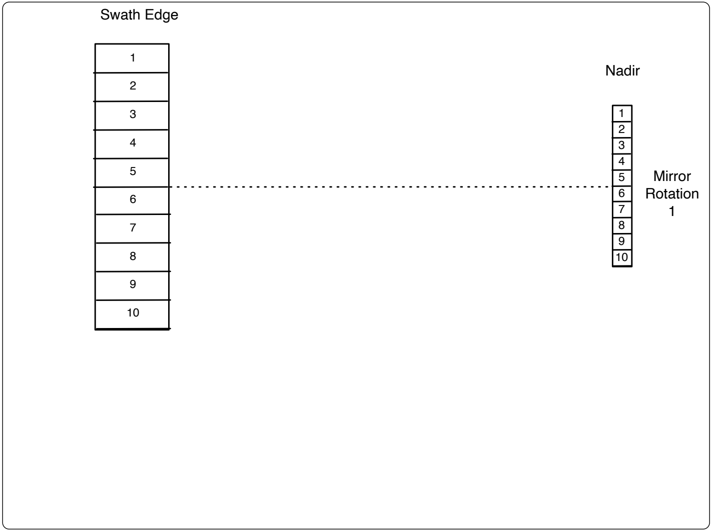
Bow-Tie Effect - Graphically - Single Detector 1^{st} Mirror Rotation



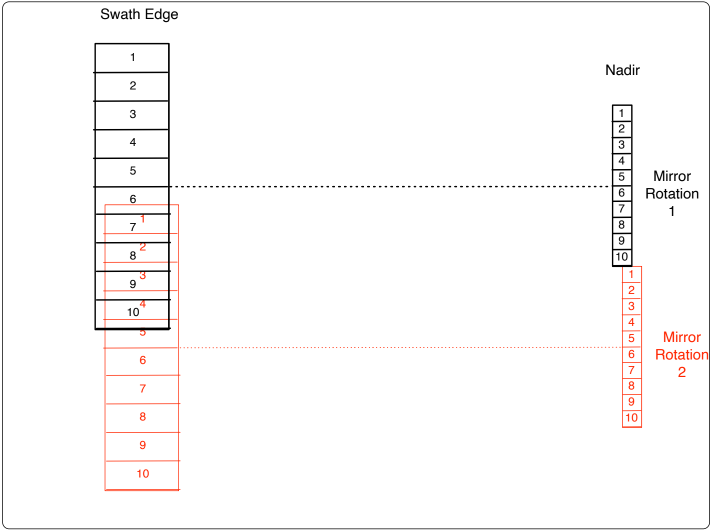
Bow-Tie Effect - Graphically - Single Detector 1^{st} & 2^{nd} Mirror Rotation



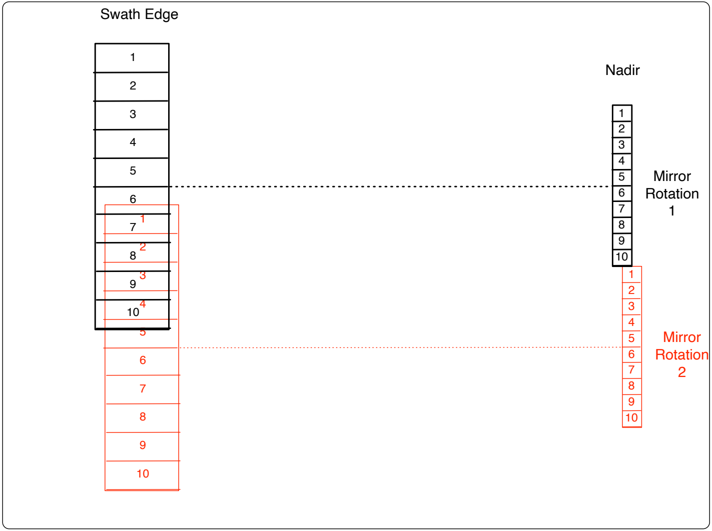
Bow-Tie Effect - Graphically - 10-Detector 1^{st} Mirror Rotation



Bow-Tie Effect - Graphically - 10-Detector 1^{st} & 2^{nd} Mirror Rotation

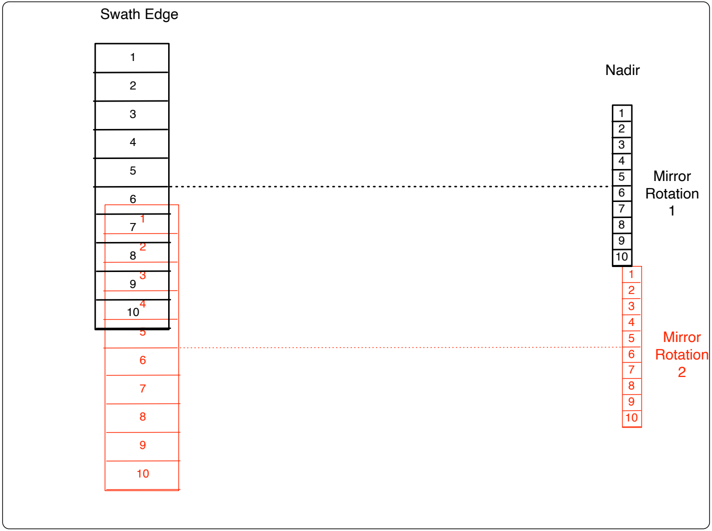


Bow-Tie Effect



This clearly plays havoc on the calculation of along-track gradients.

Bow-Tie Effect



This clearly plays havoc on the calculation of along-track gradients.

And the problem is not just at nadir.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, $\approx$ 5 minutes per full orbit & there are $\approx 100,000$ orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in $\approx$ 7 seconds.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, $\approx$ 5 minutes per full orbit & there are $\approx$ 100,000 orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in $\approx$ 7 seconds.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, ≈ 5 minutes per full orbit & there are ≈ 100, 000 orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in ≈ 7 seconds.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, ≈ 5 minutes per full orbit & there are ≈ 100, 000 orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in ≈ 7 seconds.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, $\approx$ 5 minutes per full orbit & there are $\approx$ 100, 000 orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in $\approx$ 7 seconds.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, $\approx$ 5 minutes per full orbit & there are $\approx 100,000$ orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in $\approx$ 7 seconds.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, ≈ 5 minutes per full orbit & there are $\approx 100,000$ orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in ≈ 7 seconds.

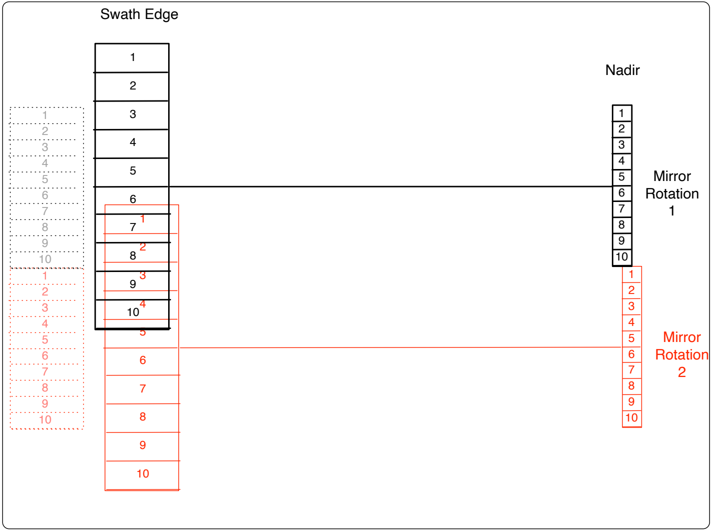
Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, $\approx$ 5 minutes per full orbit & there are $\approx 100,000$ orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in $\approx$ 7 seconds.

Bow-Tie Effect – The Solution

- I use Matlab's *griddata* function to regrid the data in the along-track direction. Specifically,
  - At each along-scan pixel location
    - Calculate the separation of the 5^{th} detector in adjacent mirror rotations.
    - Determine the new along-track pixel locations for the 10 detectors.
  - Regrid from the original array to the new array using the *linear interpolation* scheme.
  - This is *very* slow, $\approx$ 5 minutes per full orbit & there are $\approx 100,000$ orbits.
    - Construct a look-up table based on the griddata reconstruction to speed it up.
    - The result allows for an orbit to be regridded in $\approx$ 7 seconds.

Bow-Tie Effect – Graphically



Quality Mask

- The quality of each pixel is flagged from 0 to 5 based on a variety of criteria.
- One of the criteria is based on the difference between the retrieved SST and a reference SST.
    - This threshold is exceeded in regions of high dynamic variability and large changes in SST.

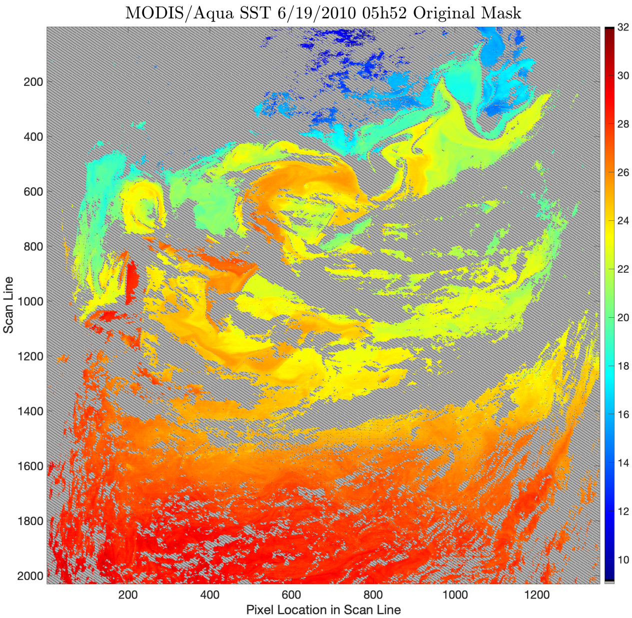
Quality Mask

- The quality of each pixel is flagged from 0 to 5 based on a variety of criteria.
- One of the criteria is based on the difference between the retrieved SST and a reference SST.
  - This threshold is exceeded in regions of high dynamic variability and large changes in SST.

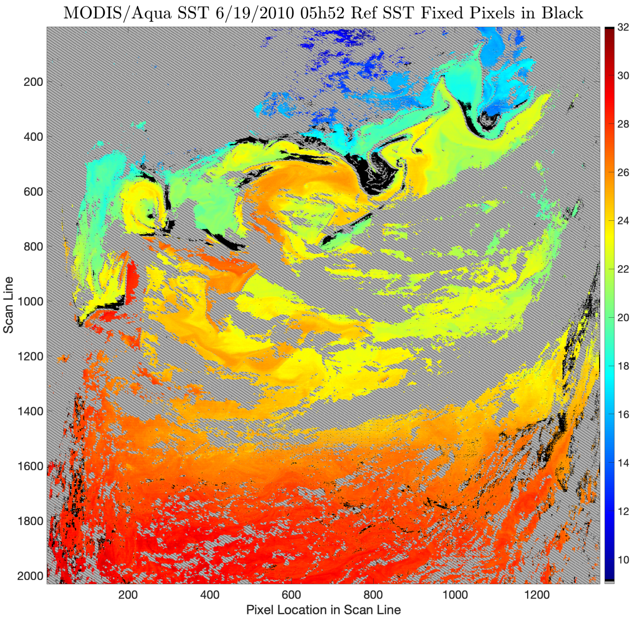
Quality Mask – Reference Temperature – The Problem

- The quality of each pixel is flagged from 0 to 5 based on a variety of criteria.
- One of the criteria is based on the difference between the retrieved SST and a reference SST.
  - This threshold is exceeded in regions of high dynamic variability and large changes in SST.

Quality Mask



Quality Mask – Reference Temperature – The Problem



Quality Mask – Reference Temperature – The Solution

- The OBPG granules include a variable of the results for each of the threshold tests: *flags_sst*
  - This field is *not* included in the PO.DAAC version of the files ← an additional problem.
- Construct global, monthly $5^\circ \times 5^\circ$ grids of maximum ΔSST.
  - From nighttime fields for 2002-2019 generate 128×128 pixel regions $>$ 95% clear.
    - This reduced the chance of clouds in the 'clear' pixels.
  - For each monthly, $5^\circ \times 5^\circ$ grid location determine the 90% range of maximum ΔSST.
- Replace the quality flag based on

$$\text{pixel quality} = \begin{cases} \text{good}, & \text{if } |\text{retrieved SST} - \text{reference SST}| < 1.1 * \text{monthly, location threshold} \\ \text{bad}, & \text{if } |\text{retrieved SST} - \text{reference SST}| > 1.1 * \text{monthly, location threshold} \end{cases}$$

Quality Mask – Reference Temperature – The Solution

- The OBPG granules include a variable of the results for each of the threshold tests: *flags_sst*
  - This field is *not* included in the PO.DAAC version of the files $\leftarrow$ an additional problem.
- Construct global, monthly $5^\circ \times 5^\circ$ grids of maximum ΔSST.
  - From nighttime fields for 2002-2019 generate 128×128 pixel regions $>$ 95% clear.
    - This reduced the chance of clouds in the 'clear' pixels.
  - For each monthly, $5^\circ \times 5^\circ$ grid location determine the 90% range of maximum ΔSST.
- Replace the quality flag based on

$$\text{pixel quality} = \begin{cases} \text{good}, & \text{if } |\text{retrieved SST} - \text{reference SST}| < 1.1 * \text{monthly, location threshold} \\ \text{bad}, & \text{if } |\text{retrieved SST} - \text{reference SST}| > 1.1 * \text{monthly, location threshold} \end{cases}$$

Quality Mask – Reference Temperature – The Solution

- The OBPG granules include a variable of the results for each of the threshold tests: *flags_sst*
  - This field is *not* included in the PO.DAAC version of the files ← an additional problem.
- Construct global, monthly 5° × 5° grids of maximum ΔSST.
  - From nighttime fields for 2002-2019 generate 128 × 128 pixel regions > 95% clear.
    - This reduced the chance of clouds in the 'clear' pixels.
  - For each monthly, 5° × 5° grid location determine the 90% range of maximum ΔSST.
- Replace the quality flag based on

$$\text{pixel quality} = \begin{cases} \text{good}, & \text{if } |\text{retrieved SST} - \text{reference SST}| < 1.1 * \text{monthly, location threshold} \\ \text{bad}, & \text{if } |\text{retrieved SST} - \text{reference SST}| > 1.1 * \text{monthly, location threshold} \end{cases}$$

Quality Mask – Reference Temperature – The Solution

- The OBPG granules include a variable of the results for each of the threshold tests: *flags_sst*
  - This field is *not* included in the PO.DAAC version of the files $\leftarrow$ an additional problem.
- Construct global, monthly $5^\circ \times 5^\circ$ grids of maximum ΔSST.
  - From nighttime fields for 2002-2019 generate 128×128 pixel regions $>$ 95% clear.
    - This reduced the chance of clouds in the 'clear' pixels.
  - For each monthly, $5^\circ \times 5^\circ$ grid location determine the 90% range of maximum ΔSST.
- Replace the quality flag based on

$$\text{pixel quality} = \begin{cases} \text{good}, & \text{if } |\text{retrieved SST} - \text{reference SST}| < 1.1 * \text{monthly, location threshold} \\ \text{bad}, & \text{if } |\text{retrieved SST} - \text{reference SST}| > 1.1 * \text{monthly, location threshold} \end{cases}$$

Quality Mask – Reference Temperature – The Solution

- The OBPG granules include a variable of the results for each of the threshold tests: *flags_sst*
  - This field is *not* included in the PO.DAAC version of the files $\leftarrow$ an additional problem.
- Construct global, monthly $5^\circ \times 5^\circ$ grids of maximum ΔSST.
  - From nighttime fields for 2002-2019 generate 128×128 pixel regions $>$ 95% clear.
    - This reduced the chance of clouds in the 'clear' pixels.
  - For each monthly, $5^\circ \times 5^\circ$ grid location determine the 90% range of maximum ΔSST.
- Replace the quality flag based on

$$\text{pixel quality} = \begin{cases} \text{good}, & \text{if } |\text{retrieved SST} - \text{reference SST}| < 1.1 * \text{monthly, location threshold} \\ \text{bad}, & \text{if } |\text{retrieved SST} - \text{reference SST}| > 1.1 * \text{monthly, location threshold} \end{cases}$$

Quality Mask – Reference Temperature – The Solution

- The OBPG granules include a variable of the results for each of the threshold tests: *flags_sst*
  - This field is *not* included in the PO.DAAC version of the files $\leftarrow$ an additional problem.
- Construct global, monthly $5^\circ \times 5^\circ$ grids of maximum ΔSST.
  - From nighttime fields for 2002-2019 generate 128×128 pixel regions $>$ 95% clear.
    - This reduced the chance of clouds in the 'clear' pixels.
  - For each monthly, $5^\circ \times 5^\circ$ grid location determine the 90% range of maximum ΔSST.
- Replace the quality flag based on

$$\text{pixel quality} = \begin{cases} \text{good}, & \text{if } |\text{retrieved SST} - \text{reference SST}| < 1.1 * \text{monthly, location threshold} \\ \text{bad}, & \text{if } |\text{retrieved SST} - \text{reference SST}| > 1.1 * \text{monthly, location threshold} \end{cases}$$

Quality Mask – Reference Temperature – The Solution

- The OBPG granules include a variable of the results for each of the threshold tests: *flags_sst*
  - This field is *not* included in the PO.DAAC version of the files $\leftarrow$ an additional problem.
- Construct global, monthly $5^\circ \times 5^\circ$ grids of maximum ΔSST.
  - From nighttime fields for 2002-2019 generate 128×128 pixel regions $>$ 95% clear.
    - This reduced the chance of clouds in the 'clear' pixels.
  - For each monthly, $5^\circ \times 5^\circ$ grid location determine the 90% range of maximum ΔSST.
- Replace the quality flag based on

$$\text{pixel quality} = \begin{cases} \text{good}, & \text{if } |\text{retrieved SST} - \text{reference SST}| < 1.1 * \text{monthly, location threshold} \\ \text{bad}, & \text{if } |\text{retrieved SST} - \text{reference SST}| > 1.1 * \text{monthly, location threshold} \end{cases}$$

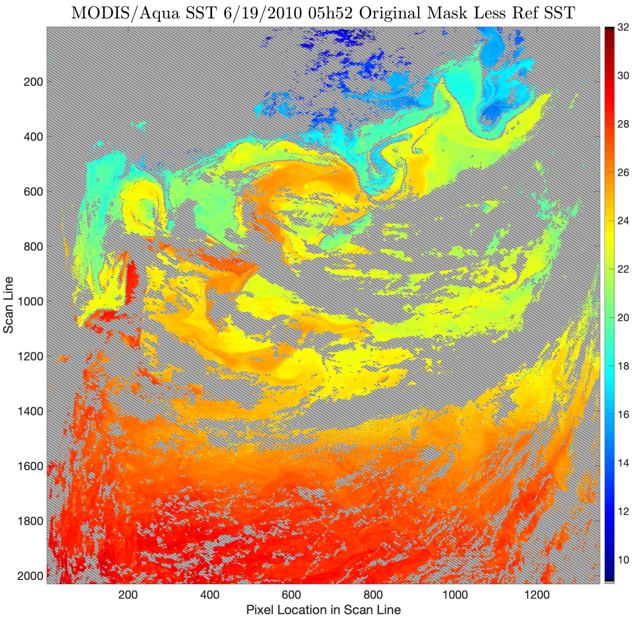
Quality Mask – High Gradient Regions – The Problem

- Another criterion is based on the range of brightness temperates in a 3 $\times$ 3 pixel square.
  - The problem is that this threshold is often exceeded in regions of high SST gradients.

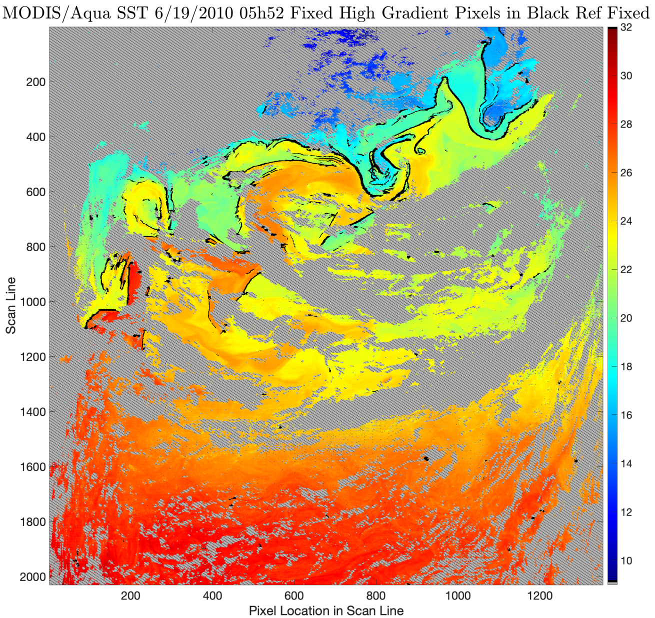
Quality Mask – High Gradient Regions – The Problem

- Another criterion is based on the range of brightness temperates in a 3 $\times$ 3 pixel square.
  - The problem is that this threshold is often exceeded in regions of high SST gradients.

Quality Mask – High Gradient Regions – The Problem

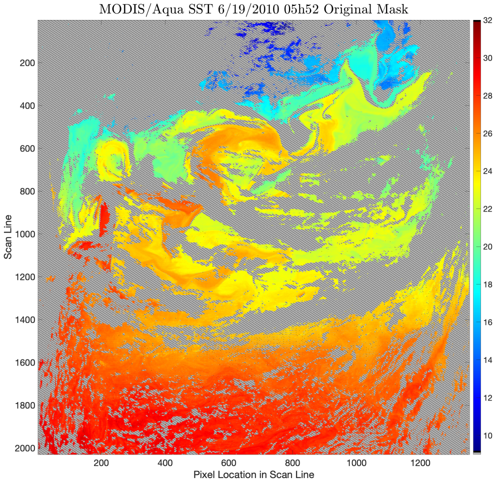


Quality Mask – High Gradient Regions – The Problem



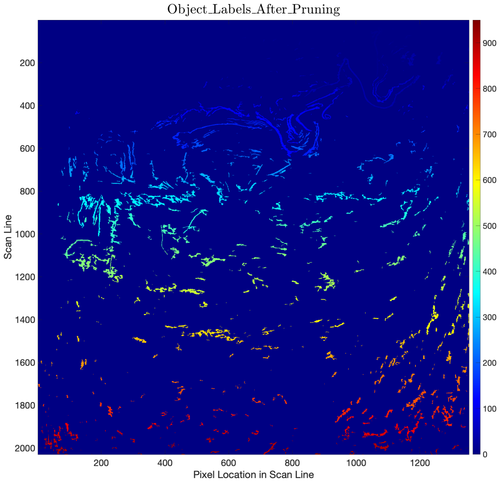
Quality Mask – High Gradient Regions – The Solution

- Find pixels exceeding the brightness temperature different threshold - from *flags_sst*.
  - Group these into objects.
  - Examine object characteristics
  - ⇒ flag object as good or bad.



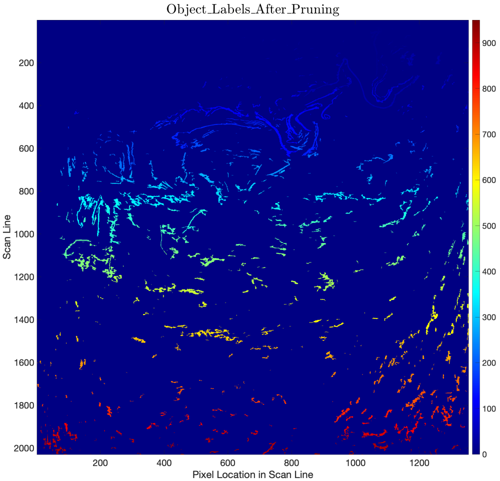
Quality Mask

- Find pixels exceeding the brightness temperature different threshold - from *flags_sst*.
  - Group these into objects.
  - Examine object characteristics
  - ⇒ flag object as good or bad.



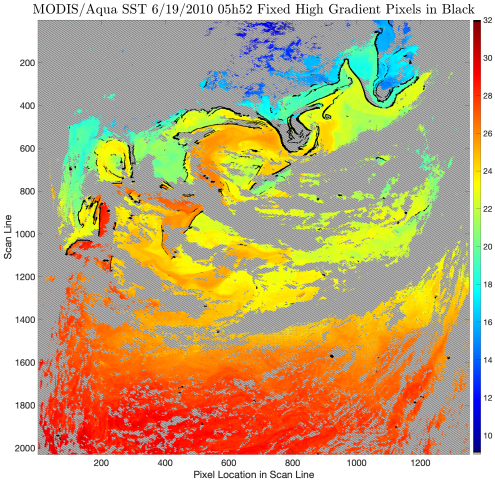
Quality Mask

- Find pixels exceeding the brightness temperature different threshold - from *flags_sst*.
  - Group these into objects.
  - Examine object characteristics
  - ⇒ flag object as good or bad.



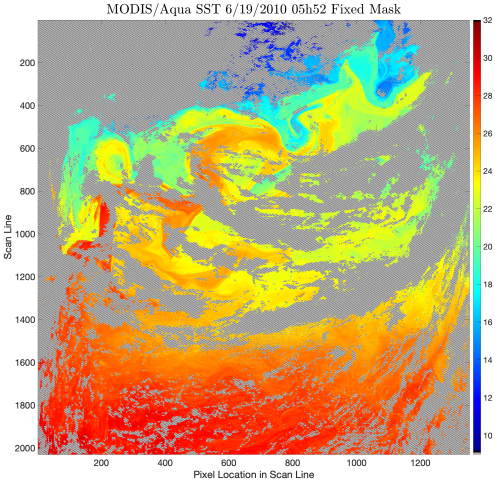
Quality Mask

- Find pixels exceeding the brightness temperature different threshold - from *flags_sst*.
  - Group these into objects.
  - Examine object characteristics
  - $\Rightarrow$ flag object as good or bad.



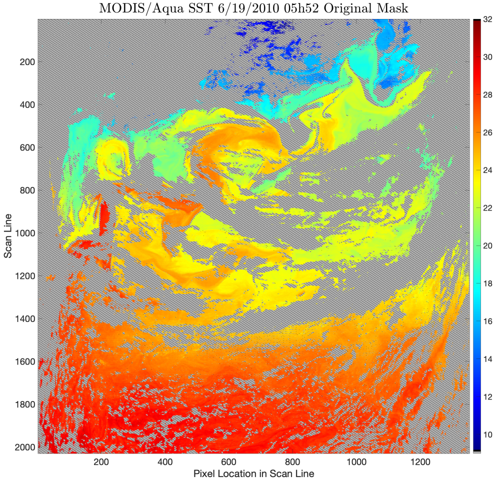
Quality Mask – The Final Mask

- A final 0-1 mask based on all tests is produced; 0 for good, 1 for bad.



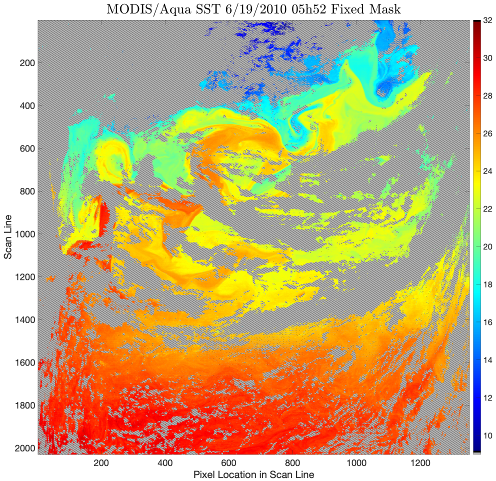
Quality Mask – The Final Mask

- A final 0-1 mask based on all tests is produced; 0 for good, 1 for bad.



Quality Mask – The Final Mask

- A final 0-1 mask based on all tests is produced; 0 for good, 1 for bad.



Outline

1. Motivation
2. The Problems And Their Solutions
3. Putting it All Together
4. Summary

The Basic Processing Steps

1. Assemble granules into complete orbits beginning at descending 78°S.
2. Regrid to address the bow-tie effect.
3. Generate a 0-1 quality mask for the regridded data.
4. Since we have the data all there, calculate Sobel gradients.

The Basic Processing Steps

1. Assemble granules into complete orbits beginning at descending 78°S.
2. Regrid to address the bow-tie effect.
3. Generate a 0-1 quality mask for the regridded data.
4. Since we have the data all there, calculate Sobel gradients.

The Basic Processing Steps

1. Assemble granules into complete orbits beginning at descending 78°S.
2. Regrid to address the bow-tie effect.
3. Generate a 0-1 quality mask for the regridded data.
4. Since we have the data all there, calculate Sobel gradients.

The Basic Processing Steps

1. Assemble granules into complete orbits beginning at descending 78°S.
2. Regrid to address the bow-tie effect.
3. Generate a 0-1 quality mask for the regridded data.
4. Since we have the data all there, calculate Sobel gradients.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
    - This is all done in Matlab
    - A basic function used to fix the mask may use 4 cores; it's hard to tell.
    - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
    - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
    - Modify the processing path to extract the necessary information from both datasets.
    - Process in us-west-2

  But, I'm new to the cloud.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

But, I'm new to the cloud.

Processing – More Problems And Some Solutions

- The previous steps, with fast regridding, take about 7 minutes/orbit/Intel Xeon W core.
  - This is all done in Matlab
  - A basic function used to fix the mask may use 4 cores; it's hard to tell.
  - So, it may require more than 7 minutes/intel core.
- There are $\approx 100,000$ orbits $\Rightarrow$ 16 core months of processing.
- And that assumes that everything goes smoothly.
- I would like to move the processing to the cloud.
- NASA is serving the MODIS Aqua L2 SST dataset from Amazon's us-west-2 cloud.

  But, the GHRSSTified MODIS L2 SST files at the PO.DAAC do not have the *flags_sst* field.
- The files with all of the fields needed are accessible from the OBPG at Goodard.

  But, these data are not available in the cloud.
- Solution
  - Extract *flags_sst* from the fields at Goddard and move them to us-west-2.
  - Modify the processing path to extract the necessary information from both datasets.
  - Process in us-west-2

  But, I'm new to the cloud.

Openscapes – Some More Help

- Openscapes is a NASA-funded project

  Designed to help researchers move their processing to the cloud near DAAC stores.
- Hope to move the MODIS L2 fix-it-up processing path to the Amazon cloud by end of June.
- Hope to have the new dataset available by mid to late September.
- Hope to serve the resulting dataset via the PO.DAAC
- I guess that that's a lot of hoping.

Openscapes – Some More Help

- Openscapes is a NASA-funded project

  Designed to help researchers move their processing to the cloud near DAAC stores.
- Hope to move the MODIS L2 fix-it-up processing path to the Amazon cloud by end of June.
- Hope to have the new dataset available by mid to late September.
- Hope to serve the resulting dataset via the PO.DAAC
- I guess that that's a lot of hoping.

Openscapes – Some More Help

- Openscapes is a NASA-funded project

  Designed to help researchers move their processing to the cloud near DAAC stores.
- Hope to move the MODIS L2 fix-it-up processing path to the Amazon cloud by end of June.
- Hope to have the new dataset available by mid to late September.
- Hope to serve the resulting dataset via the PO.DAAC
- I guess that that's a lot of hoping.

Openscapes – Some More Help

- Openscapes is a NASA-funded project

  Designed to help researchers move their processing to the cloud near DAAC stores.
- Hope to move the MODIS L2 fix-it-up processing path to the Amazon cloud by end of June.
- Hope to have the new dataset available by mid to late September.
- Hope to serve the resulting dataset via the PO.DAAC
- I guess that that's a lot of hoping.

Openscapes – Some More Help

- Openscapes is a NASA-funded project

  Designed to help researchers move their processing to the cloud near DAAC stores.
- Hope to move the MODIS L2 fix-it-up processing path to the Amazon cloud by end of June.
- Hope to have the new dataset available by mid to late September.
- Hope to serve the resulting dataset via the PO.DAAC
- I guess that that's a lot of hoping.

Outline

1. Motivation
2. The Problems And Their Solutions
3. Putting it All Together
4. Summary

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
    - Granules will be combined into orbit files.
    - The bow-tie effect will be addressed.
    - Quality mask will be modified to fix high gradient and reference SST problems.
    - Gradients will be calculated.
    - Results will be stored in orbit-based netCDF files

      And made available to the community.
- The dataset will be accessible by mid to late September.

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
    - Granules will be combined into orbit files.
    - The bow-tie effect will be addressed.
    - Quality mask will be modified to fix high gradient and reference SST problems.
    - Gradients will be calculated.
    - Results will be stored in orbit-based netCDF files

      And made available to the community.
- The dataset will be accessible by mid to late September.

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
  - Granules will be combined into orbit files.
  - The bow-tie effect will be addressed.
  - Quality mask will be modified to fix high gradient and reference SST problems.
  - Gradients will be calculated.
  - Results will be stored in orbit-based netCDF files

    And made available to the community.
- The dataset will be accessible by mid to late September.

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
  - Granules will be combined into orbit files.
  - The bow-tie effect will be addressed.
  - Quality mask will be modified to fix high gradient and reference SST problems.
  - Gradients will be calculated.
  - Results will be stored in orbit-based netCDF files

    And made available to the community.
- The dataset will be accessible by mid to late September.

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
  - Granules will be combined into orbit files.
  - The bow-tie effect will be addressed.
  - Quality mask will be modified to fix high gradient and reference SST problems.
  - Gradients will be calculated.
  - Results will be stored in orbit-based netCDF files

    And made available to the community.
- The dataset will be accessible by mid to late September.

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
    - Granules will be combined into orbit files.
    - The bow-tie effect will be addressed.
    - Quality mask will be modified to fix high gradient and reference SST problems.
    - Gradients will be calculated.
    - Results will be stored in orbit-based netCDF files

      And made available to the community.
- The dataset will be accessible by mid to late September.

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
  - Granules will be combined into orbit files.
  - The bow-tie effect will be addressed.
  - Quality mask will be modified to fix high gradient and reference SST problems.
  - Gradients will be calculated.
  - Results will be stored in orbit-based netCDF files

    And made available to the community.
- The dataset will be accessible by mid to late September.

Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
  - Granules will be combined into orbit files.
  - The bow-tie effect will be addressed.
  - Quality mask will be modified to fix high gradient and reference SST problems.
  - Gradients will be calculated.
  - Results will be stored in orbit-based netCDF files

    And made available to the community.
- The dataset will be accessible by mid to late September.

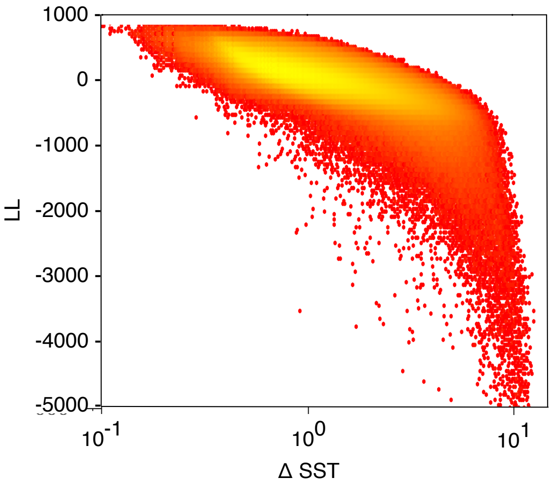
Summary

- The MODIS L2 Aqua SST dataset will be reorganized to facilitate its use for feature-related studies.
  - Granules will be combined into orbit files.
  - The bow-tie effect will be addressed.
  - Quality mask will be modified to fix high gradient and reference SST problems.
  - Gradients will be calculated.
  - Results will be stored in orbit-based netCDF files

    And made available to the community.
- The dataset will be accessible by mid to late September.

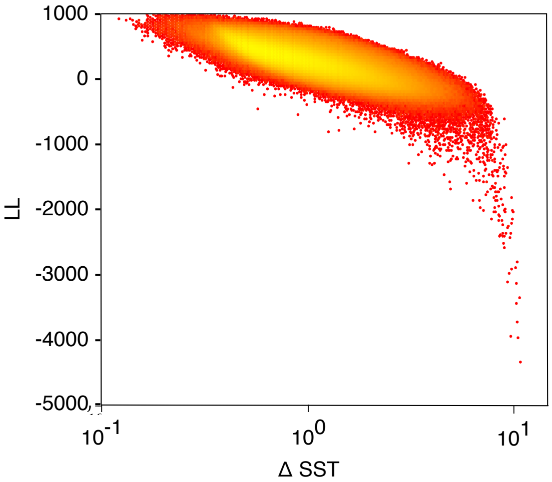

That's all Folks!
66

Quality Mask – The Problem – An Aside 95% Clear



Courtesy J.X. Prochaska. Also see: Madolyn Kelm's poster/ S1 highlights: *Using Deep Supervised Learning to Explore and Identify Patterns within Sea Surface Temperature Data*

Quality Mask – The Problem – An Aside 99% Clear



Courtesy J.X. Prochaska. Also see: Madolyn Kelm's poster/ S1 highlights: *Using Deep Supervised Learning to Explore and Identify Patterns within Sea Surface Temperature Data*