

Edge Detection with WIM and WAM

1 Introduction

The location of fronts in the sea-surface temperature (SST) images provides information on a variety of processes in the ocean. Automated objective methods to detect SST fronts have been in the development for many years. Typical edge-detection methods such as the Sobel operator are discrete approximations to the gradient. Gradient-based edge detectors produce spurious responses when applied to noisy data. These so-called local operators use often produce a multitude of fake fronts when applied to noisy data (Holyer and Peckinpaugh, 1989). The single-image edge detection (SIED) method of Cayula and Cornillon (1992) is one of several more advanced edge detection methods and has been shown to be superior in tests (Cayula et al., 1991). The method has been used extensively by Peter Cornillon's group at URI (Ullmann and Cornillon 1999, 2000, 2001) and others (Kahru et al. 1995, Diehl et al. 2002) for objective mapping of SST fronts.

The basic idea of the SIED method is to use overlapping windows to investigate the statistical likelihood of an edge by (1) calculating the histogram and detect bimodality of the histogram, and (2) detecting the cohesiveness of the potential edge. A modification to the SIED method proposed by Diehl et al (2002) uses variogram analysis in two directions to find the best window size in both horizontal and vertical directions instead of a fixed window size.

The SIED method has been implemented in WIM and WAM. You can create edge (front) images interactively with WIM but for creating the edge images and their means for a series of images it is recommended to use WAM programs *wam_edge* and *wam_edge_accumulate*.

2 Methods

The SIED method as modified by Diehl et al. (2002) is implemented in WIM and WAM. SST data are available from multiple sensors and in multiple time composites. The best sources to use for edge detection are the MODIS-Terra and MODIS-Aqua daily, 8-day, and monthly 4 km datasets (available from 2000 to present) and the AVHRR Pathfinder version 5 (PF5) daily, 7-day and monthly data (available from 1981 to present). While the MODIS data may have higher accuracy, the AVHRR Pathfinder data have the advantage of a longer time period. Both MODIS and PF5 data are distributed in a form that includes pixels at variable quality levels.

I found that the Diehl et al (2002) method is very sensitive and detects large numbers of small-scale fronts. As our task is to detect the seasonal and inter-annual variability in the locations of the major fronts, I made various modifications.

The threshold value of the between-cluster variance to the within-cluster variance was increased from 0.76 (of Diehl et al., 2002) to 0.8.

Instead of using the variable window method of Diehl et al (2002) you can experiment with larger fixed size window (e.g. 20 x 20; the original Cayula and Cornillon, 2002 used 32 x 32 pixel window).

The daily and even 8-day composites are often missing large areas due to clouds. It is therefore tempting to use monthly or other composited images for front detection. However, composited images as opposed to uncomposited images may create artificial fronts and may be missing the

real fronts that become undetectable due to smoothing over time. It is therefore best not to use composited images for front detection.

3 Prerequisites

We assume that you are familiar with the basics of the Windows or DOS command line, i.e. how to open the command window, change directory, issue a command, etc. We also assume that you have a set of images that you are going to use. In our example we use screened 8-day MODISA daily SST images in the *Sat\MODISA\L3\8day\SST_4* folder.

4 Screening the images

If the images have pixels of various quality and the quality index then the first task is to screen the data and keep the best quality (quality 0 for MODIS and 7 for PF5). The pixels with lower quality are usually contaminated by clouds and would interfere with edge detection although the SIED method performs additional extra steps to eliminate suspect pixels. You can use *wam_screen_sst_ocpg* for screening the MODIS SST data produced by the Ocean Color Processing Group and *wam_screen_pf* for screening AVHRR Pathfinder data. Please see the corresponding section in the WAM Manual ([WAM.pdf](#)). In our example we use images that are already screened.

5 Picking the sub-area

You are probably interested in a certain area and not in the whole global ocean. You can create subset images with another tool (e.g. *wam_series*) and then run *wam_edge* on those. In this example we will create the subset images with *wam_edge* itself by cutting out a rectangle from the global equal angle image.

First we will experiment with the interactive edge detection in WIM. Open a sample global SST image with WIM and select your area of interest, i.e. select a rectangle with the left mouse button down, and then click on the *Cut* icon (*Scissors*) on the toolbar. Note and write down the Upper-Left and Lower-Right pixel coordinates – you will need them later. Try to use an image size with a horizontal dimension that is divisible with 4. Then select *Edge-SIED* from the WIM menu. Now select again the previous image (click on the SST image “*CutImg*”) and overlay the edges on top of it with the *Overlay* tool in the toolbar. You can repeat the same with different window sizes and note the differences. You may find that some edges correspond to coastlines. We will eliminate those by creating an extended land mask image that will limit the edges to the open sea area only. We pick our area of interest in the Gulf of Mexico and adjoining areas in the Caribbean and Central America with the upper-left and lower-right coordinates, respectively, 1952, 1396 and 2552, 1836. You can extract this sub-area by selecting the *Cut* (scissors) tool from the toolbar without selecting a rectangle and typing in the corresponding pixel coordinates. Now create the coastlines for this sub-image with *Geo-Get Map Overlay* with pixel value 255. Use the high-resolution coastlines, e.g. *coast_full.b*. Fill land areas with pixel value 1 using *Edit-Draw-Fill*. Be careful not to let the “paint” overflow into the ocean areas from coastlines that don’t touch the image edge. In these cases extend the coastlines to the image edge with a line or pen tool and pixel value 255. The result should look something like Fig. 1.



Fig. 1. Land mask of the area of interest.

Now expand the land mask using *Edge-Dilate-1*. The result is shown in Fig. 2.

Dilation of the land mask prevents detecting edges along coastlines. Remember to stretch the colors from 0 – otherwise the pixels below the start value may be black and invisible. The idea of the mask is that pixels with value 0 will be used in the subsequent edge detection analysis and nonzero pixels will be excluded. As you can see, we have expanded the land areas and excluded ocean pixels immediately next to the coast. Save the dilated land mask to a HDF file, e.g. *landmask_gom_dilated1x.hdf* in *C:\Sat\MODISA\L3\8day*. Now we are ready to create the monthly edge images.

If you know the latitude-longitude range of your area of interest then you can create your land mask in the following way. Say, our area of interest is 16-42 N and 140-100 W, and we are using a global image *global.hdf*. We then use *wam_ll2xy* to find the respective x and y (pixel coordinates) of the top left and lower right corner:

```
wam_ll2xy global.hdf 45 -140
```

```
wam_ll2xy global.hdf 16 -100
```

After that we cut the image:

```
wam_cut global.hdf 910 1024 1820 1683
```

Here 910; 1024 was the result from our first run of *wam_ll2xy* and 1820; 1683 from the second run. This creates a *Cut* image that we can edit.



Fig. 2. Dilated landmask of the area of interest.

6 Running *wam_edge*

Open a command window, change directory (*cd*) to a proper path (e.g. *cd C:\Sat\MODISA\L3\8day*) and issue the following command:

```
wam_edge SST_4A2004*screened.hdf 20 landmask_gom_dilated1x.hdf 1952 1396 2552 1836
```

Note that we are using all images in folder *SST_4* matching *A2004*screened.hdf*, a window size 20, landmask image *landmask_gom_dilated1x.hdf*, upper left corner 1952, 1396 and lower-right corner 2552, 1836. You can see the syntax (all possible arguments) by typing *wam_edge* without arguments and hitting *Enter*. If successful, you have created four subset images (**_Cut.hdf*) and corresponding edge images (**_Cut_Sied.hdf*) for each matching SST image. A sample edge image is shown in Fig. 3.

In our exercise we have 4 source images corresponding to 4 eight-day SST composites in 2004. We should now have 4 edge images like Fig. 3 (**Sied.hdf*) corresponding to each 8-day period of 2004.

Please note that the command created a whole set of subset images (**_Cut.hdf*) and edge images. The edge image names are created by adding a string “*_Sied.hdf*” to the SST image name. When you run the same command again you have to be careful **not** to match both the original global SST images and the subset (**_Cut.hdf*) images in the same command. When we keep the source image in a separate folder (e.g. *SST_4*) below the current folder where we run the command, the output will not be mixed with the input. If you do not give the four coordinate numbers in the command line then *wam_edge* will be applied to the whole image (e.g. previously extracted *Cut* subimage), e.g.

```
wam_edge A2004*_Cut.hdf 20 landmask_gom_dilated1x.hdf
```



Fig. 3. Edges detected for January 25-February 1, 2004 of MODISA SST. To reverse the colors (make the background white and fronts black) we have reversed-stretched the colors, i.e. set *Start=255* and *End=0* in *Color Definition*.

7 Running *wam_edge_accumulate*

After we have a series of edge images we want to somehow average them into a mean, composited image. Simple averaging is not appropriate as different areas may have different numbers of cloud-free pixels. This is especially relevant when using daily images that have large areas of missing data due to clouds. We will therefore need to count both the number of valid pixels and the number of detected fronts in order to get statistics for the fronts. Type:

```
wam_edge_accumulate A2004*_Cut.hdf 1
```

Please note that we are **not** matching directly the edge images (**Sied.hdf*) but the source (subset) images (**_Cut.hdf*). Both the source (e.g. **_Cut.hdf*) and the corresponding edge images (**_Sied.hdf*) have to be in the same folder. The number 1 in the command means that we are making 1 set of averages, i.e. the overall mean. We can use 12 instead of 1 and in that case we would be making monthly averages instead of the overall average. For example, if we use daily or 8-day images in edge detection we can make monthly averages by using “12” in the command. If successful, we now have three new types of HDF images (and the corresponding PNG images): *front count*, *valid SST* and *front frequency* (see Figs. 4, 5, 6 below).

Take a look at the content of each of the HDF files with WIM. Explain the meaning of pixel values of these images. A command option that we did not use is for reduction of the resulting images. In case of very large images you may want to reduce the size of the resulting images, e.g. by 2 or 3 times.

Both *wam_edge* and *wam_edge_accumulate* create attributes "Start Year", "Start Day", "End Year", "End Day" and save these attributes to all HDF files that they save. This makes it possible to create time series (e.g. with *wam_statist*) using the results from both programs. If you are using nonstandard input data (e.g. files that *wam_edge* cannot get the date from) then it may not work.

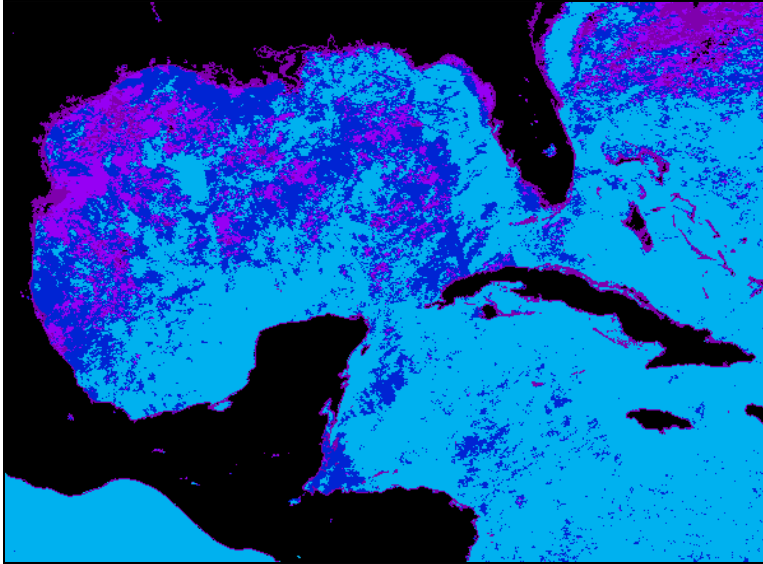


Fig. 4. Number of valid SST pixels used for edge detection. You can add the color scale manually with *View-Annotate*.

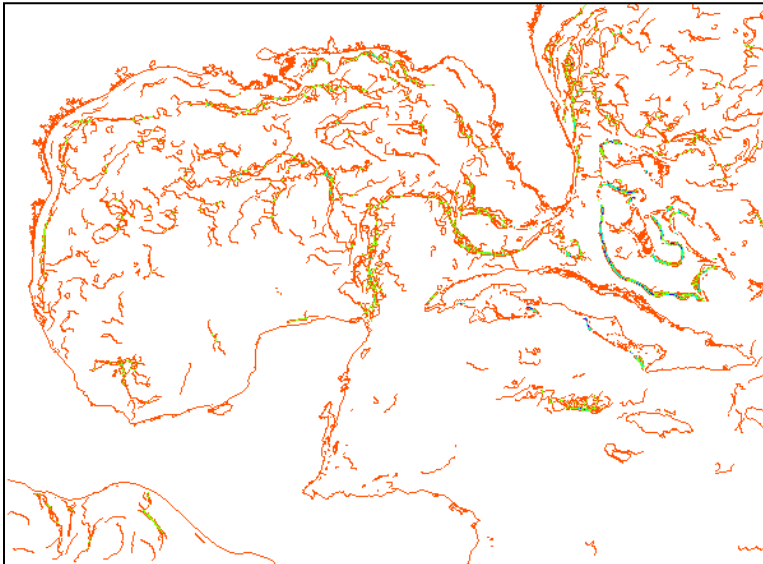


Fig. 5. Number of detected edges (for each pixel). You can add the color scale manually with *View-Annotate*.

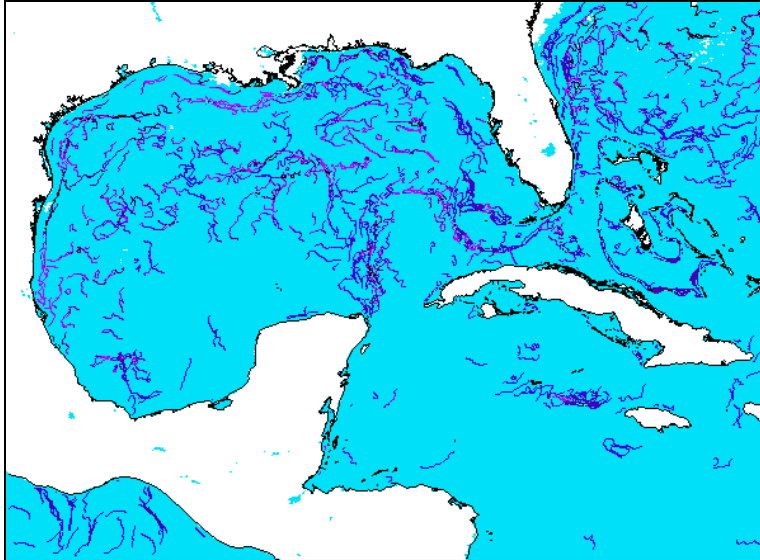


Fig. 6. Frequency of detected edges (i.e. the number of detected edges divided by the number of valid SST for each pixel).

References

- Cayula, J.-F., P. Cornillon, R. J. Holyer and S.H. Peckinpaugh (1991) Comparative study of two recent algorithms designed to process sea-surface temperature fields, *IEEE Trans. Geosci. Remote Sens.*, GE-29,175-177.
- Cayula, J.-F. and P. Cornillon (1992) Edge detection algorithm for SST images. *Journal of Atmospheric and Oceanic Technology* 9: 67-80.
- Diehl, Scott F., Judith W. Budd, David Ullman and Jean-Francois Cayula (2002) Geographic Window Sizes Applied to Remote Sensing Sea Surface Temperature Front Detection. *Journal of Atmospheric and Oceanic Technology*, 19(7): 1115-1113.
- Holyer, R. J. and S.H. Peckinpaugh (1989) Edge detection applied to satellite imagery of the oceans. *IEEE Transactions on Geoscience and Remote Sensing* 27: 46-56.
- Kahru, M., B. Håkansson, O. Rud, Distributions of the sea-surface temperature fronts in the Baltic Sea as derived from satellite imagery. *Cont. Shelf Res.*, 15(6): 663-679, 1995.
- Ullman, D.S., and P.C. Cornillon (1999) Surface temperature fronts off the East Coast of North America from AVHRR imagery, *J. Geophys. Res.*, 104(C10), 23459-23478.
- Ullman, D.S., and P.C. Cornillon (2000) Evaluation of front detection methods for satellite-derived SST data using in situ observations, *J. Atmos. Oceanic Tech.*, 17(12), 1667-1675.
- Ullman, D.S., and P.C. Cornillon (2001) Continental shelf surface thermal fronts in winter off the northeast US coast, *Cont. Shelf Res.*, 21(11-12), 1139-1156.