



WORLD OCEAN CIRCULATION

ALGORITHM THEORETICAL BASIS DOCUMENT FOR OCEAN FRONTS (THÈME 2)

customer	ESA/ESRIN
ESA contract	ESA Contract No. 4000130730/20/I-NB
document reference	WOC-ESA-ODL-NR-009_ATBD_T2_fronts_V2.0
Version/Rev	120
Date of issue	14/01/2022.

Issued by	Lucile Gaultier (OceandataLab)
Approved by	Gilles Larnicol (OceanDataLab/Magellium)
Approved by	Marie-Hélène Rio (ESA)

Distribution List

	Name	Organization	Nb. copies
Sent to :	M.H. rio	ESA/ESRIN	ESA-STAR
Internal copy :	Project Manager	OceanDataLab	1 (digital copy)

Document evolution sheet

Ed.	Rev.	Date	Purpose evolution	Comments
1	0	15/07/2021.	Creation of document	
	2	12/05/2022	Version 2	

Contents

1 Introduction	4
1.1 Products summary	4
1.2 Scope & Objectives	4
1.3 Document structure	4
1.4 Applicable & Reference documents	4
1.5 Terminology	5
2 Ocean fronts	7
2.1 Overview	7
2.2 SIED (Single Image Front Detection)	8
2.2.1 Step 1: Histogram analysis	8
2.2.2 Step 2: Compute directions of frontal structures	10
2.2.3 Step 3: Contour Following	11
2.2.4 Step 4: Filtering and flagging	12
2.3 MIED (Multiple Images Edge Detection)	13
2.3.1 Overview	13
2.3. Algorithms	14
2.3.1 Persistent fronts	14
2.4 Product validation	15
2.4.1 Front validation database	15
2.4.2 Qualitative validation	15
2.4.3 Metric to compare front	15
2.4.3 Results from the quantitative analyses	19

List Of Images

- Figure 1: Example of a map of probability of having a front (between 0 and 1, in grey) plotted over the Sea Surface temperature. Probabilities that are null have been masked.
- Figure 2: The red arrows represent the direction that maximizes the operator and the corresponding value of the operator is plotted in color. High values of the operator corresponds also to high probability of having a front. The operator has been normalized and is thus between 0 and 1
- Figure 3: Fronts are plotted over the SST, fronts on the left image are outputs from Step 3 with many loops, fronts on the right image are output from Step 4, after filtering and flagging

List Of Tables

-

1 Introduction

1.1 Products summary

This document provides the algorithm theoretical basis for the products developed by the OceanDataLab in the framework of the ESA World Ocean Circulation project. The product provides added value detection of frontal structures using remote sensing Sea Surface Temperature (SST). Data from two sensors are available, fronts have been derived from hourly SEVIRI level 4 SST near the Gibraltar Strait and daily level 4 Microwave SST over a wide portion of the North Atlantic Ocean.

1.2 Scope & Objectives

This document is the Algorithm Theoretical Basis Document (ATBD) which is dedicated to the description and justification of the algorithms used in the generation of:

- WOC-FRONTS

1.3 Document structure

In addition to this introduction, this document includes the following chapters:

Chapter 2: Ocean Front (WOC-FRONTS_1D)

1.4 Applicable & Reference documents

- [RD-1] ESA WOC2019: <http://woc2019.esa.int/index.php>
- [RD-2] Synthesis of the WOC2019 User Consultation Meeting recommendations http://woc2019.esa.int/files/WOC2019_summary_synthesis.pdf
- [RD-3] Cayula, J., & Cornillon, P. (1992)., Edge Detection algorithm for SST images, *Journal of Atmospheric and Oceanic Technology*, 9, 67-80

- [RD-4] W.T. Freeman and E.H. Adelson, "The design and use of steerable filters" - IEEE Trans. On Pattern Anal. Machine Intell., vol. 13, no. 9, pp 891-906, 1991.
- [RD-5] M. Donias et P. Baylou, "Un opérateur directionnel multicritère pour la détection et le suivi de lignes fines", 1997
- [RD-6] Cayula, J., & Cornillon, P. (1995). Multi-Image Edge Detection for SST Images, *Journal of Atmospheric and Oceanic Technology*, 12(4), 821-829.

1.5 Terminology

ACCUA	Analisi della dinamica della Corrente Circumpolare Antartica
AMSR2	Advanced Microwave Scanning Radiometer 2
ADT	Absolute Dynamic Topography
AI	Artificial Intelligence
AIL	Action Items List
AIS	Automatic Identification System
ASAR	Advanced Synthetic Aperture Radar
ASCAT	Advanced SCATterometer
ATBD	Algorithm Theoretical Basis Document
AVHRR	Advanced Very High Resolution Radiometer
CCD	Contract Closure Document
CCI	Climate Change Initiative
CCMP	Cross-Calibrated Multi-Platform
CFOSAT	Chinese-French Oceanography Satellite
CIESM	Mediterranean Science Commission
CMEMS	Copernicus Marine Environment Monitoring Service
CNES	Centre National d'Etudes Spatiales
CNR	Consiglio Nazionale delle Ricerche
CTD	Conductivity, Temperature and Depth
DP	Data Pool
DTU	Danmarks Tekniske Universitet
DUACS	Data Unification and Altimeter Combination System
EBUS	Eastern Boundary Upwelling System
ECCO	Estimating the Circulation & Climate of the Ocean
ECMWF	European Centre for Medium-Range Weather Forecasts
EFARO	European Fisheries and Aquaculture Research Organisations
EMB	European Marine Board
ENVISAT	Environmental Satellite
EO	Earth Observation
EPB	European Polar Board
EOEP-5	5th Earth Observation Envelope Programme (2017-2021)
ERA	ECMWF Reanalysis

ESA	European Space Agency
ESF	European Science Foundation
EU	European Union
EuroGOOS	European Global Ocean Observing System
FAO	Food and Agriculture Organization of the United Nations
FR	Final Report
FSLE	Finite Size Lyapunov Exponent
GCM	Global Circulation Model
GOCE	Gravity Field and Steady-State Ocean Circulation Explorer
GOOS	Global Ocean Observing System
GMI	Global precipitation monitoring Microwave Imager
GMM	Gaussian Mixture Models
HYCOM	Hybrid Coordinate Ocean Model
IAR	Impact Assessment Report
ICCAT	International Commission for the Conservation of Atlantic Tunas
ICSU	International Council for Science
IGPB	International Geosphere-Biosphere Programme
IUGG	International Union of Geodesy and Geophysics
IFREMER	Institut Français de Recherche pour l'Exploitation de la Mer
IOC	Intergovernmental Oceanographic Commission
ITCZ	InterTropical Convergence Zone
mEOF-r	Multivariate Empirical Orthogonal Functions reconstruction
MoM	Minutes of Meeting
NATL3D	North Atlantic 3D Ocean Currents
NEMO	Nucleus for European Modelling of the Ocean
NOAA	National Oceanic and Atmospheric Administration
NCC	Norwegian Coastal Current
NwAFC	Norwegian Atlantic Front Current
NwASC	Norwegian Atlantic Slope Current
OLCI	Ocean and Land Color Imager
OSSE	Observing System Simulation Experiment
OC	Ocean color
OSCAT	Oceansat-2 SCATterometer
PD	Product Delivery
PM	Project Manager
PMP	Project Management Plan
PUB	Publication
PUM	Product User Manual
QUID	Quality Information Document
RB	Requirement Baseline
REMSS	Remote Sensing Systems

ROMS	Regional Oceanic Modeling System
RTOFS	Real-Time Ocean Forecast System
S3	Sentinel 3
SAR	Synthetic Aperture Radar
SEVIRI	Spinning Enhanced Visible and InfraRed Imager
SCOR	Scientific Committee on Oceanic Research
SIED	Single Image Edge Detection
SKIM	Sea surface KInematics Multiscale monitoring
SLSTR	Sea & Land Surface Temperature Radiometer
SOCIB	Sistema d'observació i predicció costaner de les Illes Balears
SODA	Simple Ocean Data Assimilation Ocean/sea ice reanalysis
SSH	Sea Surface Height
SSS	Sea Surface Salinity
SST	Sea Surface Temperature
SoW	Statement of Work
TN	Technical Note
TOPAZ	Tracers of Phytoplankton with Allometric Zooplankton
TUOC	Total Upper Ocean Currents
UCL	Use Case Library
UCM	User Consultation Meeting
UCPC	Upper-layer ocean Circulation Processes e-Catalogue
UI	Upwelling Index
UN	United Nations
URD	User Requirement Document
VR	Validation Report
VT	Visualization Tool
WBS	Work Breakdown Structure
WOC	World Ocean Circulation

2 Ocean fronts

2.1 Overview

The following product of frontal structures is retrieved from Sea Surface Temperature (SST) remote sensing observation. We consider here the frontal structures as barriers of transport and thus the fronts are detected using population histogram. A front is present if two distinct populations are detected in the processed window.

To run the front analysis, the image is splitted into small windows (whose size depends mainly on the resolution of the remote sensing observation).

The analysis can be either performed on a Single Image (Single Image Edge Detection, section 2.2) or on an ensemble of Images (Multiple Image Edge Detection, section 2.3).

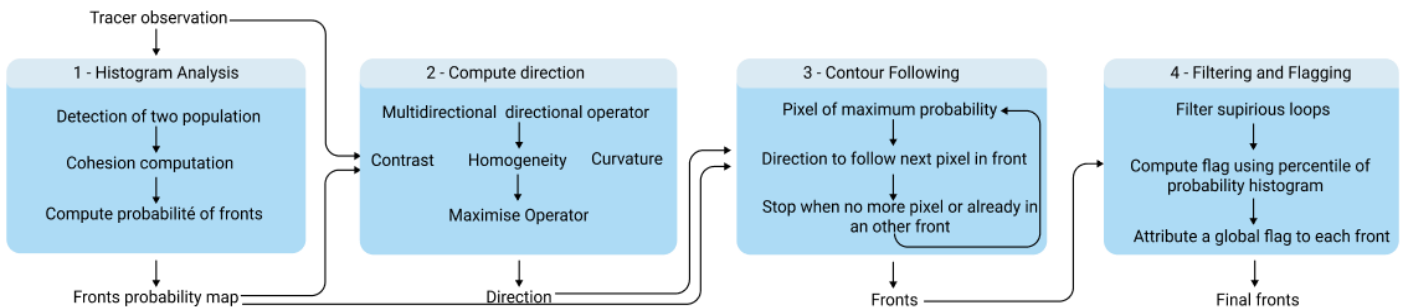
The strategy to compute the probability of fronts is very similar to the one implemented by Cayula and Cornillon (1992, 1994). The crest line following to retrieve a 1d front as a vector from the probability of front maps is a specific algorithm that relies on the property of the image in the neighborhood of the front (contrast, homogeneity, curvature).

2.2 SIED (Single Image Front Detection)

This algorithm relies on the fronts detection algorithm SIED (Single Image Edge Detection) detailed in [RD-3]. The input is a tracer image, and the output are vectors representing frontal structures.

The algorithm is splitted into 4 parts.

- Histogram analysis
- Contour following
- Compute direction of fronts
- Flagging and smoothing



2.2.1 Step 1: Histogram analysis

This part consists in retrieving edge pixels, i.e pixels that belong to a frontal structure.

Moving windows of several sizes are scanning the tracer image. The size of the moving window will highly depend on the scale of the structure we want to retrieve. For High resolution data, the histogram analysis is performed with windows of different sizes to catch large and small scale frontal structures.

For each window, an histogram of the tracer value is computed. Noisy values are removed by filtering out tracer values that are too different from the mean of the window. Pixels are then splitted into two populations, later referred to as A and B. The method to separate the two populations is detailed in [RD-3]. Considering a threshold τ between the two populations, the

total variance can be expressed as the sum of the variances within each population ($J_e(\tau)$) and the contribution to the total variance resulting from the separation of the two clusters ($J_b(\tau)$). If the histogram shows two distinctive peaks, the last function ($J_b(\tau)$) will be more impacted by a pixel categorized in the wrong category. Therefore the following function is maximized to find the frontier between the two populations $\tau_{optimal}$:

$$J_b(\tau) = \frac{N_A(\tau) * N_B(\tau)}{(N_A(\tau) + N_B(\tau))^2} * (\mu_A(\tau) - \mu_B(\tau))^2$$

with N_A (resp N_B) the number of pixels in the population A (resp B), and μ_A (resp μ_B) the mean of the tracer value for the population A (resp B).

We can compute $J_b(\tau_{optima}) = argmax(J_b(\tau))$ and assess the separation of the two populations. To do so, we will study the cohesion within each population.

We will refer later as θ for the maximum of the variance resulting from the separation of the two clusters. If the θ value is too low, the window can be dropped as no front has been found, but it is easier to assess the presence of a front using the cohesion algorithm presented thereafter.

Populations A and B should differ enough so that they can be splitted with a front line. Cohesion coefficients are computed and the window is dropped if they don't meet defined thresholds. Three coefficients are computed : simple cohesion for population A (C_A), simple cohesion for population B (C_B) and global cohesion (C_{AB}):

$$C_A = \frac{\Gamma_{AA}}{\Gamma_{AA} + \Gamma_{AB}}$$

$$C_B = \frac{\Gamma_{BB}}{\Gamma_{BB} + \Gamma_{BA}}$$

$$C_{AB} = \frac{\Gamma_{AA} + \Gamma_{BB}}{\Gamma_{AA} + \Gamma_{AB} + \Gamma_{BB} + \Gamma_{BA}}$$

with Γ_{XY} the number of pixels from population Y which are neighbors to a pixel from population X.

The single cohesion coefficient C_A (resp. C_B) represents the number of pixels from population A (resp. B) that are close to their own population compared to the number of pixels from population A (resp. B) that are close to the other population B (resp. A). The global cohesion coefficient C_{AB} reflects the cohesion of population A and B at once.

If any of these coefficients are below a defined threshold, the window is dropped. Note that a filter is applied to the pixel before the cohesion computation to remove isolated pixels and homogenize populations.

The last step is to actually compute the probability of having a front. To do so, an edge pixel is defined as a pixel that belongs to population A (resp. B) but has a pixel from population B (resp. A) in the neighborhood.

For each pixel, the probability of a front is obtained by accumulating the θ value computed previously for each detection while the window is spanning. This probability is then normalized

by dividing by the number of times the pixel has been detected to retrieve a probability between 0 and 1.

Pixels with a probability but which are isolated are removed as they are considered as noise.

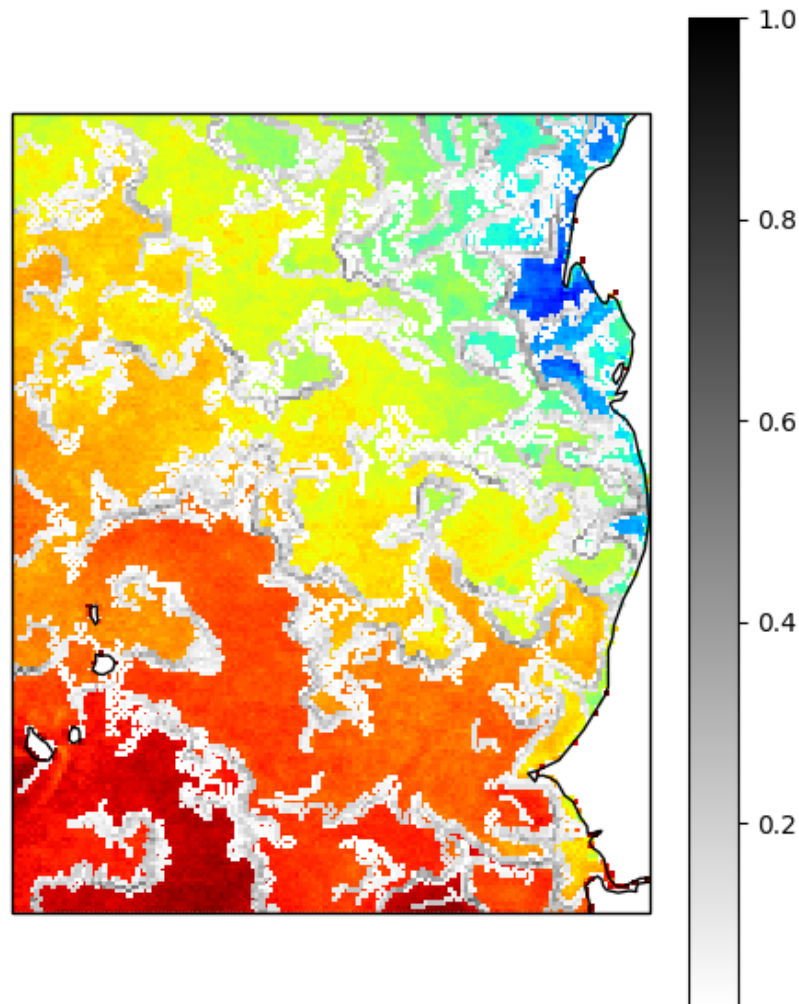


Figure 1: Example of a map of probability of having a front (between 0 and 1, in grey) plotted over the Sea Surface temperature. Probabilities that are null have been masked.

2.2.2 Step 2: Compute directions of frontal structures

Using the front probability map computed previously, the ridge line can be extracted by following the local maximum of probability. As the output of most contour following algorithm is quite noisy, a statistical method has been chosen to retrieve the front structure.

For each pixel where the probability is not null, an operator is computed to represent the direction of the ridge line, with its relevance ([RD-4], [RD-5]).

For each pixel p , one can consider a disk with N rays and R points linearly spaced on each ray. The front probability map is interpolated on the disk and is later referred to as $P_{disk}(r, n)$. Three parameters are computed on this disk:

$$C[p, n] = \sum_{r>0}^R \frac{P_{disk}(r, n)}{r}$$

- contrast:
- heterogeneity: $H[p, n] = \text{variance}_r(P_{disk}(r, n))$
(variability of probability values on the ray with direction n)

$$A[p, n] = \sum_{r>0}^R \min_k (P_{disk}(r, n) - P_{disk}(r, n + k))$$

- Curvature:
-

The radius R will vary with the resolution and the noise in the data. Generally, the size of the radius corresponds to the size of the smallest curvature rays that can be seen on a front. For SEVIRI, a radius of 3 pixels is optimal whereas for Sentinel-3 SLSTR, 5 pixels are necessary.

An operator is retrieved as the linear combination of the three parameters contrast, heterogeneity and curvature, weighted by three coefficients, α, β, γ on the data matrix. For each pixel, the main direction of the front is the one that maximises the operator.

$$\text{operator} = (\alpha * C - \beta * H + \gamma * A)$$

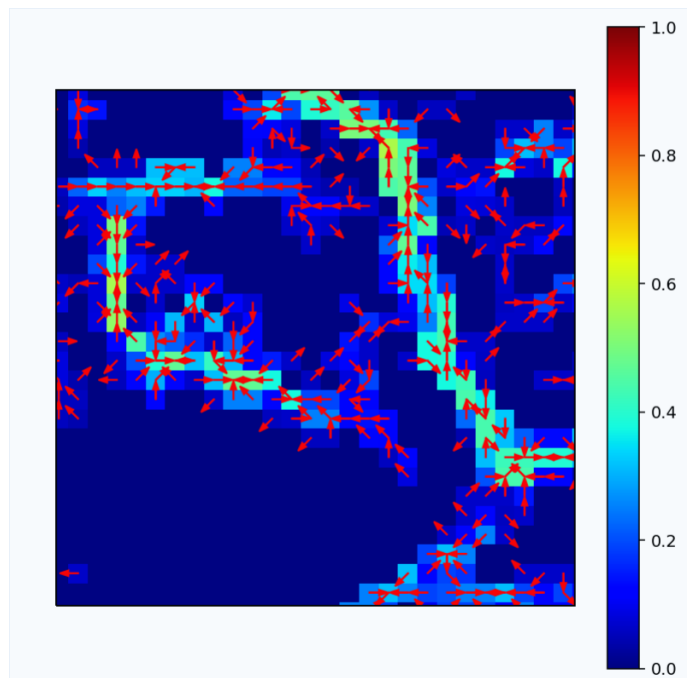


Figure 2: The red arrows represent the direction that maximizes the operator and the corresponding value of the operator is plotted in color. High values of the operator corresponds also to high probability of having a front. The operator has been normalized and is thus between 0 and 1

2.2.3 Step 3: Contour Following

The contour following relies on the previously computed probability map and direction. The starting pixel corresponds to the maximum probability of having a front among the pixels that are to be processed. Following the previously computed direction, the next pixel is selected and removed from the list of pixels to be processed. The contour is stopped when:

- one of the image boundary is reached
- no strictly positive probability of having a front is found in the possible direction
- The pixel or its neighbours already belong to a contour

From the starting point, the search for pixels is performed in the two possible directions.

While computing the main directions (section 2.2.2), several discretizations of the possible directions have been tested. In the algorithm either 8, 16 or 32 possible directions have been tested and in most cases 16 directions is the best compromise between accuracy and computing cost.

Is it possible from the method to calculate for each detected front a sort of "uncertainty" information? Like a probable front envelope?

2.2.4 Step 4: Filtering and flagging

When performing the contour following, loops can appear at the end of a contour when following the maximum probability values. These loops are removed if they meet these three criteria:

- Small length
- Small probability (as they are in fact on the edge of a front)
- Close edges

A flag is then attributed to each front to represent its relevance. It represents a percentile in the probability of front histogram. For this processing, fronts are sorted into 6 flags (6 percentiles). A flag value is attributed to each pixel of a front. A flag value is also attributed to the whole front by keeping the majority flag value within the pixel that belongs to the front. It enables users to filter front with their probability. For example, flag front with a value of 1 correspond to the 16.7 % value the most probable

The figure below illustrates the removal of these loops during the filtering step as well as the flagging.

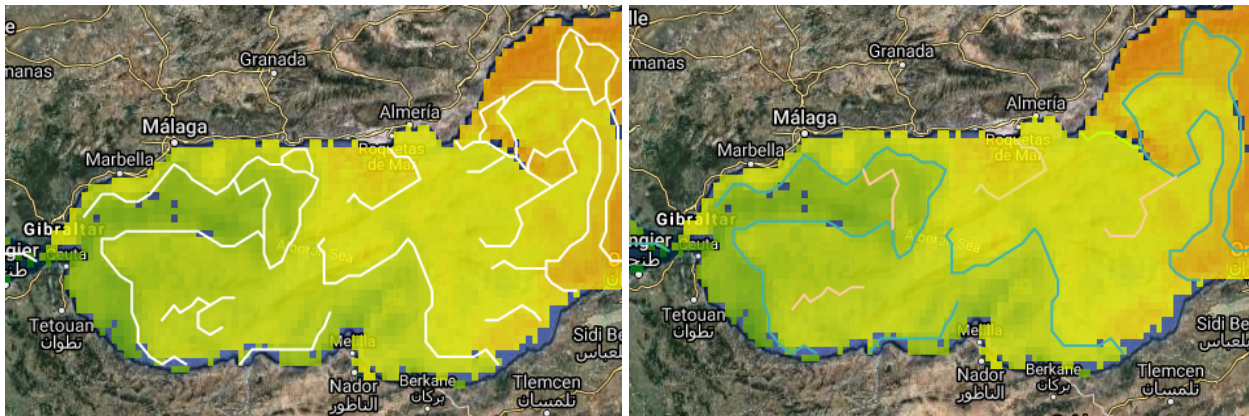
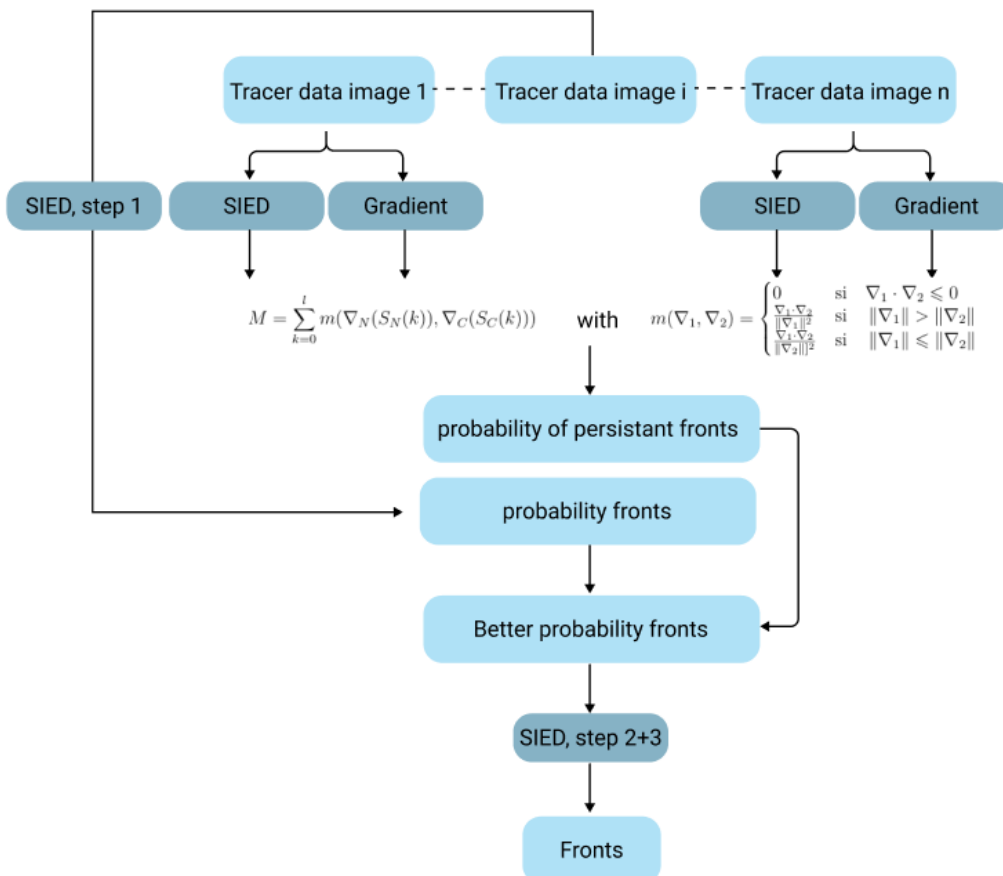


Figure 3: Fronts are plotted over the SST, fronts on the left image are outputs from Step 3 with many loops, fronts on the right image are output from Step 4, after filtering and flagging. The colors on the right images correspond to the flag value. Fronts in blue are highly probable (flag 1), pink fronts correspond to flag 2 fronts and light green to flag 3 fronts.

2.3 MIED (Multiple Images Edge Detection)

2.3.1 Overview

The multi-image algorithm aims at retrieving fronts that have been missed by the single image algorithm because of a cloud or noise on the single image processed. If a front is detected in other images close in time they can be added to the single image. If a front is detected in no other images, it will considerably lessen the probability of having a front and thus this front will be removed or flagged as highly improbable. The method implemented is similar to the one described in [RD-6].



2.3. Algorithms

The first step consists of running the algorithm previously described SIED on all available images. While running the SIED algorithm, the probability of having a front and the tracer gradient in each direction are stored for each front pixel.

2.3.1 Persistent fronts

The fronts previously computed are splitted in small segments of 10 pixels length. For example, if a front is 22 pixel long, 13 segments will be considered: $[n_0..n_9]$, $[n_{10}..n_{19}]$... $[n_{120}..n_{21}]$

For each of these segment, for each image, a we consider a possible translation of δ pixels of the front using a moving window of size $[-\delta, \delta] \times [-\delta, \delta]$.

The corresponding metric M is computed as follow for a pixel p using I images :

$$M[p] = \sum_{k=0}^l m(\nabla_N(S_N[k, p]), \nabla_C(S_C[k, p]))$$

with S_N , the segment on the image k, and S_C a similar segment on the other image, eventually even translated. ∇_N et ∇_C relates to the gradients of the image k and the gradient on the other image.

The m function is defined as follow:

$$\text{if } \nabla_1 \cdot \nabla_2 \leq 0 \quad m(\nabla_1, \nabla_2) = 0$$
$$\text{else } m(\nabla_1, \nabla_2) = \frac{\nabla_1 \cdot \nabla_2}{\max(\|\nabla_1\|^2, \|\nabla_2\|^2)}$$

The M function is computed on the δ window for each pixel but only the maximum value is kept.

This analysis is performed on all images and the new front probability is normalized to get a value between 0 and 1.

The contour following algorithm is similar to the one from SIED and is performed on the new front probability map as explained in section 2.2.2, 2.2.3, 2.2.4.

2.4 Product validation

As there is no ground truth for the frontal products, performing a quantitative analysis is not an easy task.

The validation can be performed qualitatively by looking at an ensemble of fronts vector over the corresponding SST image. This can be done easily using the Woc portal woc.oceandatalab.com.

To perform a more quantitative validation, the strategy is to build a database of frontal structures manually drawn using the Visualisation Tool SynTool and consider it as a baseline for the comparison.

2.4.1 Front validation database

SST are drawn using the SynTool web instance available at woc.oceandatalab.com. Drawing capabilities enable us to easily draw shapes and build synoptic maps. The shape can then be exported in a specific json file.

2.4.2 Qualitative validation

All the processed frontal structures have been included in the WOC portal woc.oceandatalab.com. The data that are available are derived from SEVIRI L3C OSISAF SST product and REMSS MWOI L4 SST product in 2018 for the North Atlantic area and 2019 for the Agulhas area).

2.4.3 Metric to compare front

To perform an automatic and quantitative analysis, one needs to measure the similarities between two curves coming from two different sources.

These similarities include the measurement of distance between these two curves (difference in position), but also the difference in terms of direction and curvature.

This quantitative analysis is here performed between the manually drawn and the automatically detected fronts database but it can also be used to measure the displacement of a front between two observations.

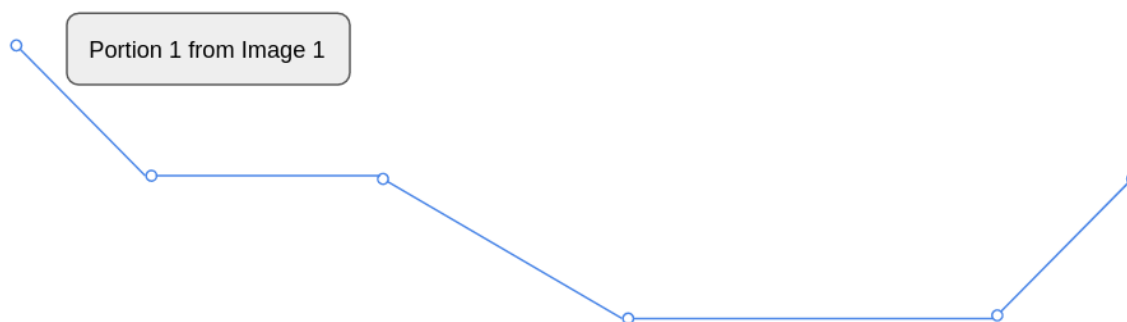
For each front in the first image, the algorithm has three steps:

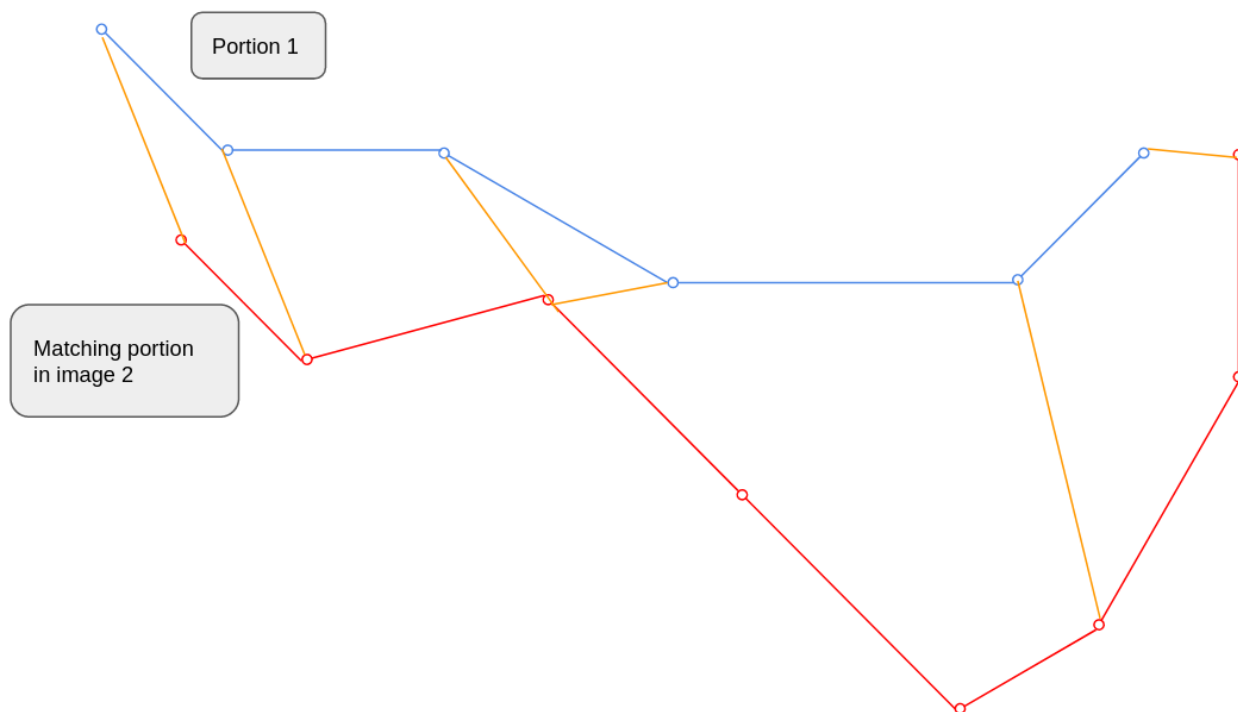
- find the closest front in the other image
- Match the points from the two curves
- compute the similarity between the two curves: mean distance, translation, direction, curvature

To perform the comparison, frontal structures are divided into small portions to ease the matching. Most of the time, 5 to 10 pixel length is adequate to perform the comparison.

The first step is to compute a distance matrix between all the pixel fronts in the images (image 1 and image 2).

The first processed front from image 1 is the one that contains the closest pixel in image 2:

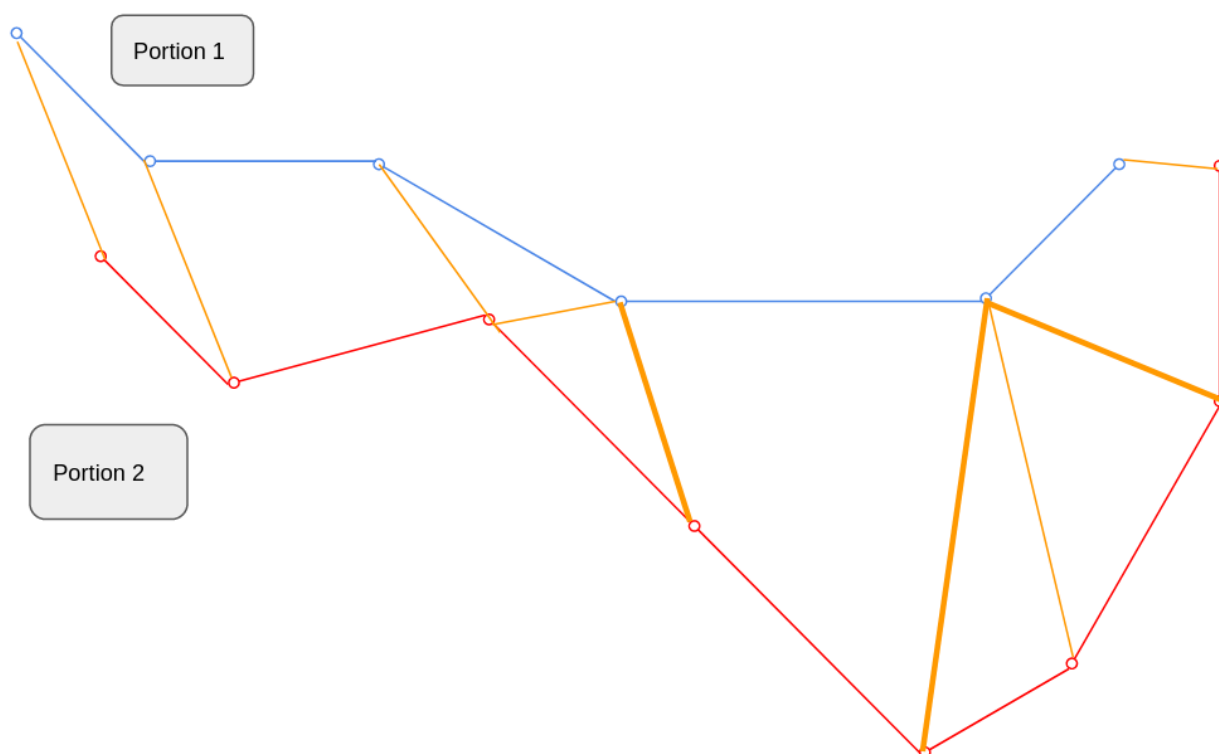




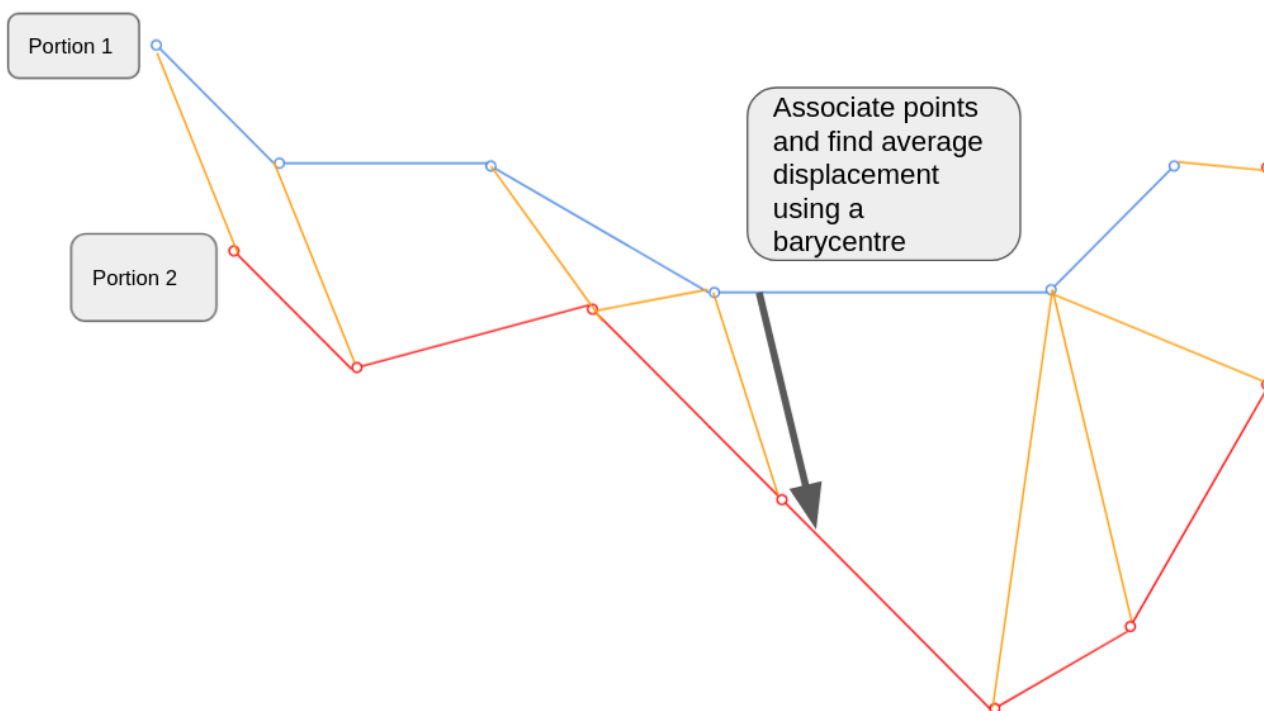
The next step is to match the pixels from the two curves using a Dynamic Time Warping (DTW) algorithm.

For each pixel from Portion 1, the closest pixel in Portion 2 is found. At this stage there are eventually few points that are not affiliated.

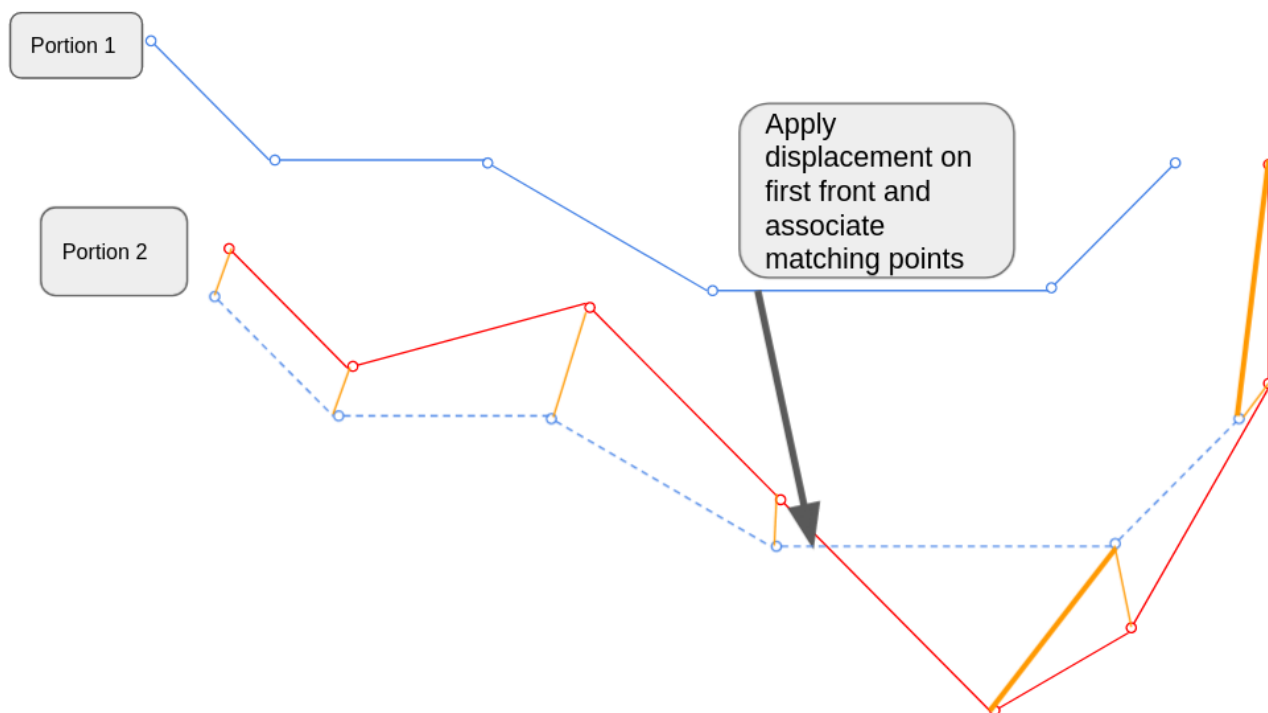
The non affiliated points from Portion 2 are affiliated with the closest points from Portion 1 (in bold in the image below):



Using all segment, one can compute the displacement between the two fronts (black arrow in the scheme below):

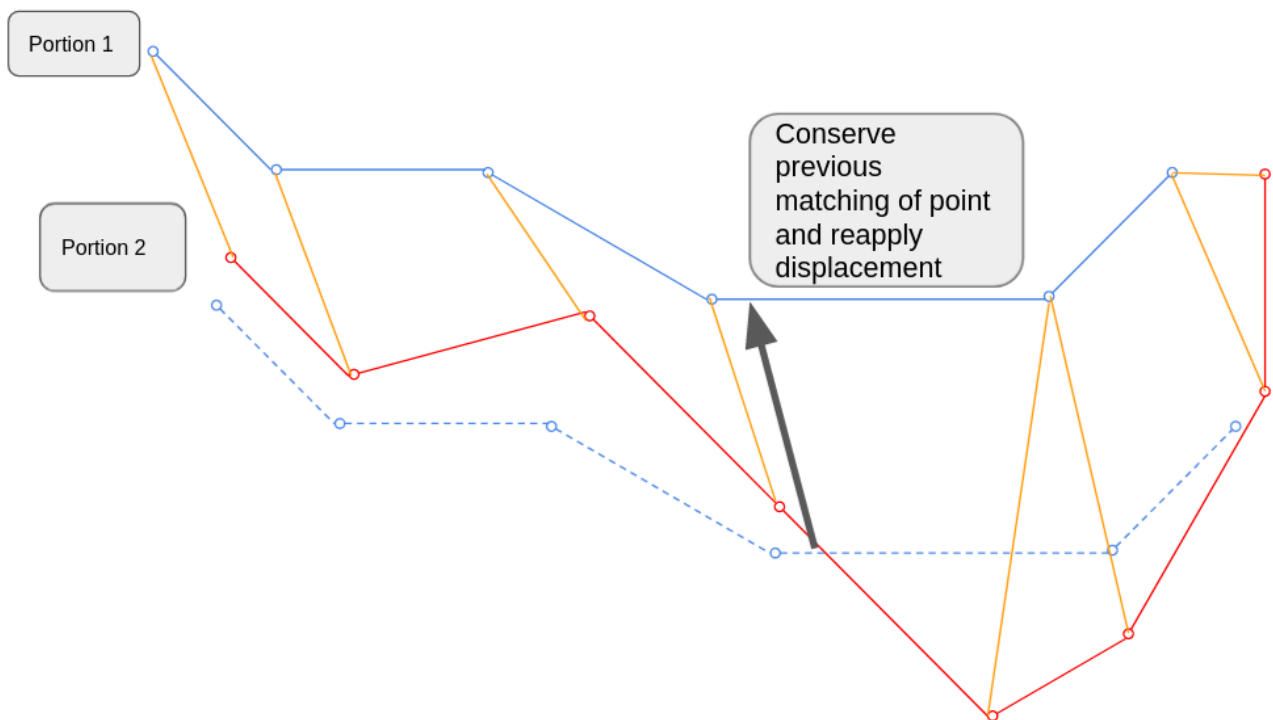


The matching between pixels from the Portion 1 and Portion 2 is then performed using the comparison between the portion 1 that has been displaced (dash line in the scheme below) and the Portion 2:



This way the matching takes into account a possible translation between the two fronts before matching the equivalent points.

The previous matching is kept and the Portion 1 at its original position is compared to the Portion 2.



At this stage, for each front, the mean distance is computed (mean length of the yellow segment) as well as the global translation between the two fronts (black arrow)

Two other metrics can be retrieved when the fronts are compared:

- similarity in term of curvature for each matching point that has been found
- similarity in term of direction

For each manually drawn front, if a matching front is found in the automatically processed database, the distance between the two fronts is measured. The fail rate corresponds to the percentage of fronts that could not be affiliated to another portion of fronts.

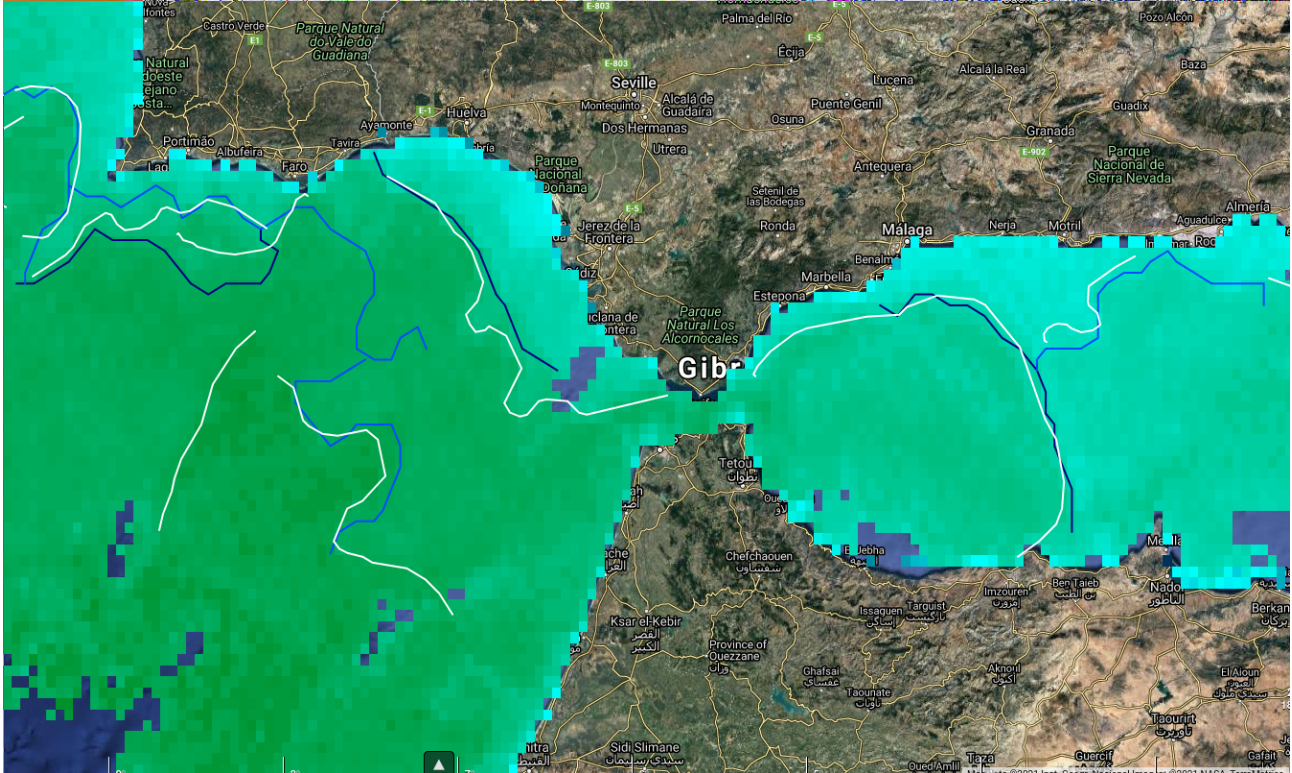
Both, the fail rate and the mean distance between two fronts are used for the analyses.

So far, 25 images have been analysed and only the most probable fronts are validated.

The fail rate is between 12 and 15% depending on the dynamic range in the processed tracer observation. The mean distance and displacement is 5 to 8 km for the SEVIRI fronts, which corresponds to the pixel resolution.

Could you show some examples? (of success and of failure)

It would also be interesting, to quantify the uncertainty of the manual front detection procedure, to have it done by more than one operators and check how the drawn fronts differ one from the other..



Example of automatic detection from SEVIRI for fronts flagged as 1 (dark blue) and 2 (light blue) compared with fronts manually drawn in white. It can be difficult for the naked eye to detect some frontal structures in very homogeneous area. But fronts can also be missed in the automatic database when they are too coastal, around noisy data or near clouds. When fronts are detected by the human and by the automatic detection they are usually located at the right place.