



Projet sargalert

Fusion of satellite derived Sargassum velocity and numerical models.



Organisation du script

classes_drift.py :

Contient toutes les classes permettant le calcul des déplacements
classe principale : Modis_Goes_Translation

Fonction_utile.py :

Contient toutes les fonctions annexes utiles lors du calcul des déplacements mais ne nécessitant pas de paramétrage (normalement pas besoin d'y toucher)

Validation_modis.py :

Scripts permettant de généré les jeux de données de pointage

Validation_goes.py :

manuel pour la validation des données goes et modis

github

 sargalert_translation Private

 main  1 Branch  0 Tags

 Go to file  Add file  Code

 About

 jeremscript-beta Add files via upload a1a04f1 · 6 minutes ago  2 Commits

 Goes_validation.py Add files via upload 6 minutes ago

 README.md Initial commit 6 minutes ago

 classes_drift.py Add files via upload 6 minutes ago

 fonction_utile.py Add files via upload 6 minutes ago

 points_info.xlsx Add files via upload 6 minutes ago

 points_info_goes.xlsx Add files via upload 6 minutes ago

 points_info_modis.xlsx Add files via upload 6 minutes ago

 validation_modis.py Add files via upload 6 minutes ago

WP3 translation project

 Readme

 Activity

 0 stars

 1 watching

 0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Paramètres pre-utilisation :

paramètres :

chemins :

```
#=====
#====Chemins=====
path_img_mod = r"C:\Users\DELL\Desktop\goes\modis\mod\2022"
path_img_myd = r"C:\Users\DELL\Desktop\goes\modis\myd\2022"
path_img_goes = r"C:\Users\DELL\Desktop\goes\goes_download"
mask_output_goes = r"C:\Users\DELL\Desktop\goes\masque_goes_2022.tif"
mask_output_modis = r"C:\Users\DELL\Desktop\goes\masque_modis_2022.tif"
#=====
#Output=====
output_path = r"C:\Users\DELL\Desktop\Jupyter\Projet_Sargalert\Notebooks\plot"

#Mask
seuil_goes_inf = 0.0002 #goes threshold for sarg mask
seuil_goes_sup = 0.1 #goes threshold for sarg mask

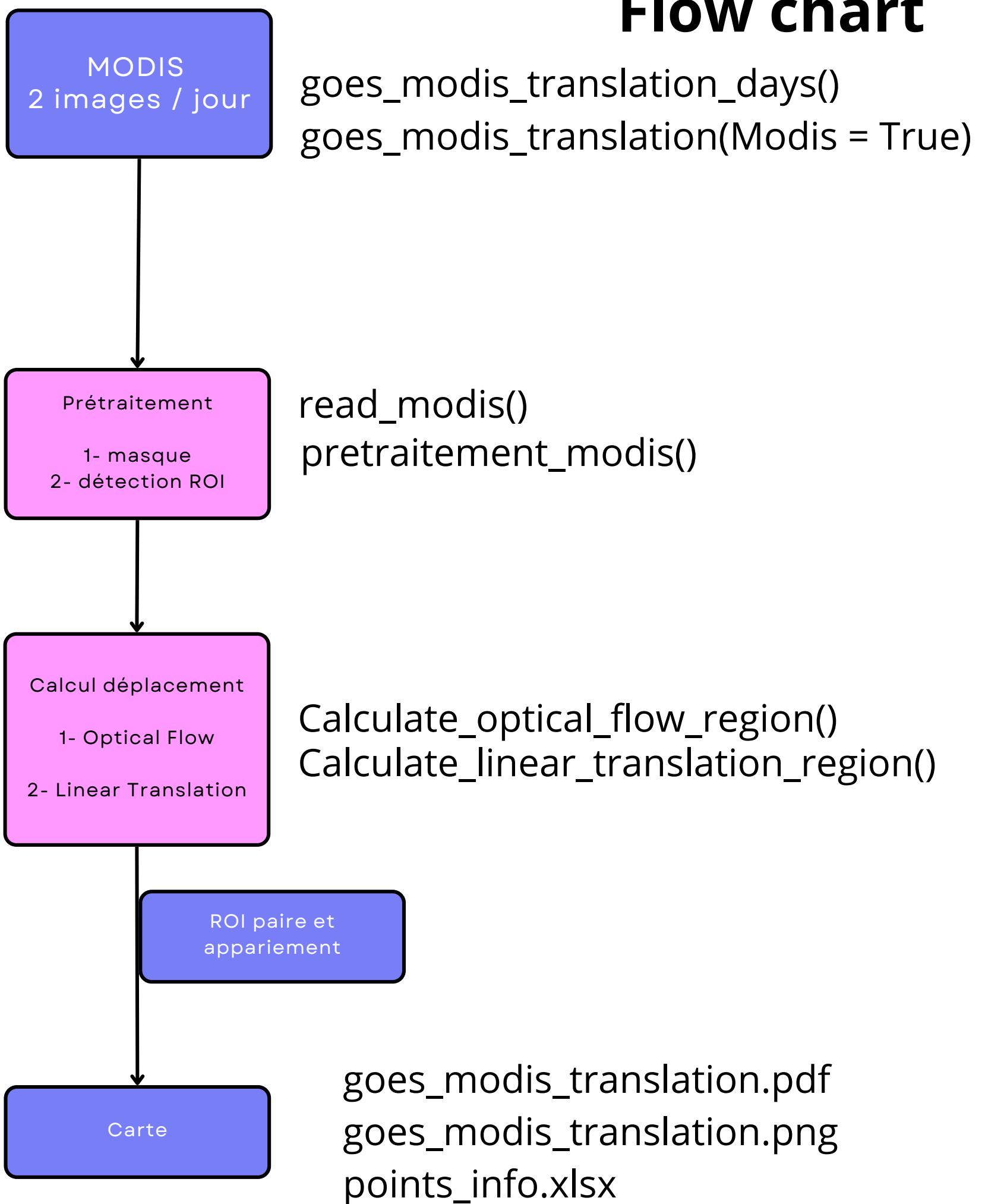
#Extention of ROI for more information (30 => 30 pixel more in each direction)
extend = 30

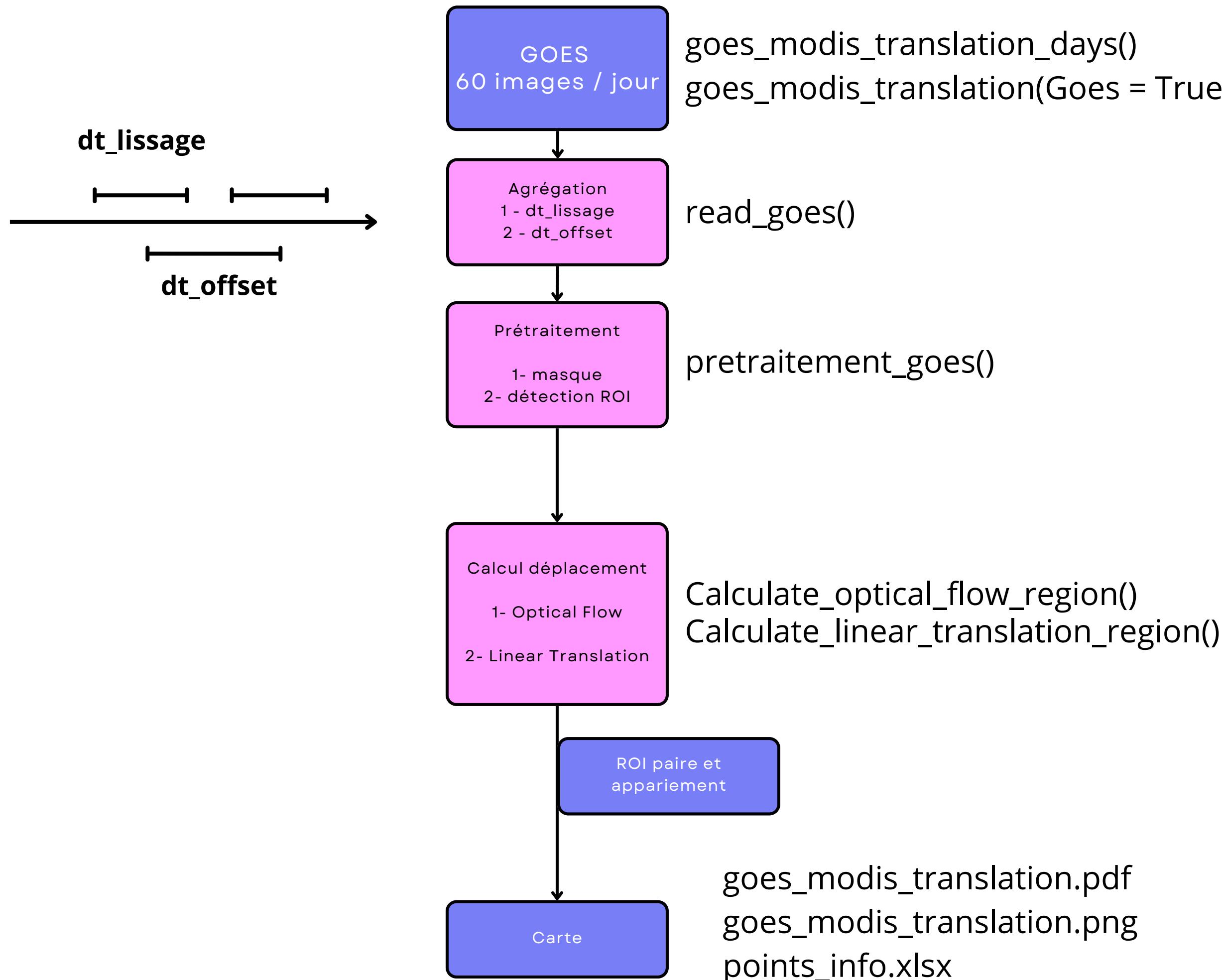
#Threshold for agreement between Optical flow and Linear translation
angle_threshold = 25 #in degrees
speed_threshold = 0.2 #in % (speed x % Less or more)

#Threshold for the number of matches minimum for linear_translation
matches_threshold = 3
#Pixel resolution in km
pixel_res = 1
#Maximol speed for sargasse in m/s
max_sarg_speed = 1

#Time period for goes : 1st date = 1st file
#Smoothing => 10h à 13h et 16h à 19h
#Time delta => 3h
#by shifting time you can shift the smoothing and the time delta for goes utilisation
start_time_mod = '2022-05-01T10:00:00.000000000'
end_time_mod = '2022-05-01T13:00:00.000000000'
start_time_myd = '2022-05-01T13:00:00.000000000'
end_time_myd = '2022-05-01T16:00:00.000000000'
delta_time_goes = 3 #time difference for goes
delta_time_modis = 3 #time difference for goes
```

Flow chart



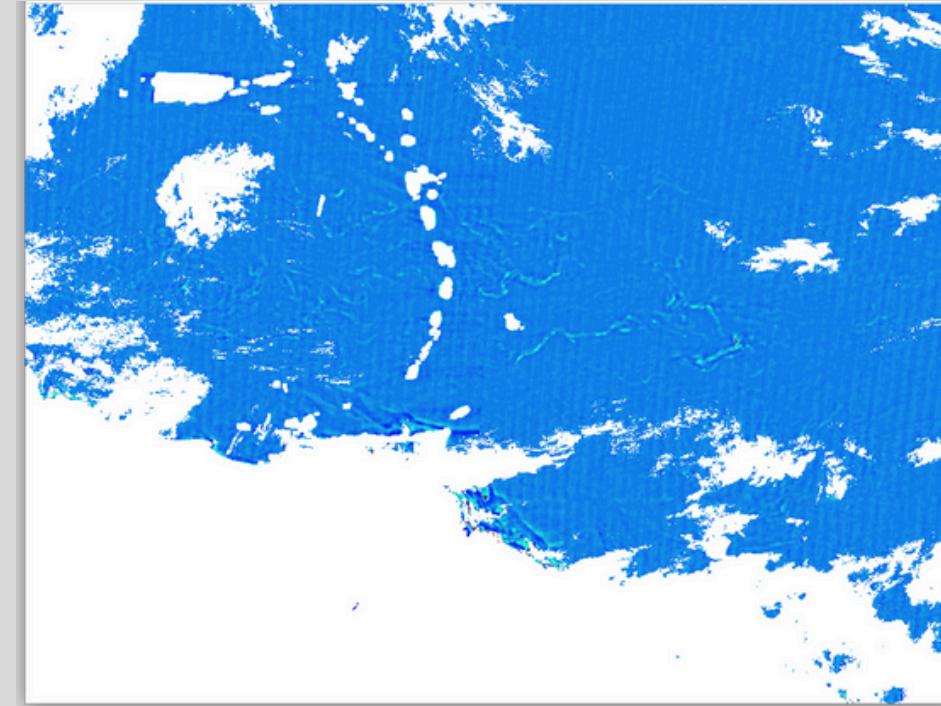


Données

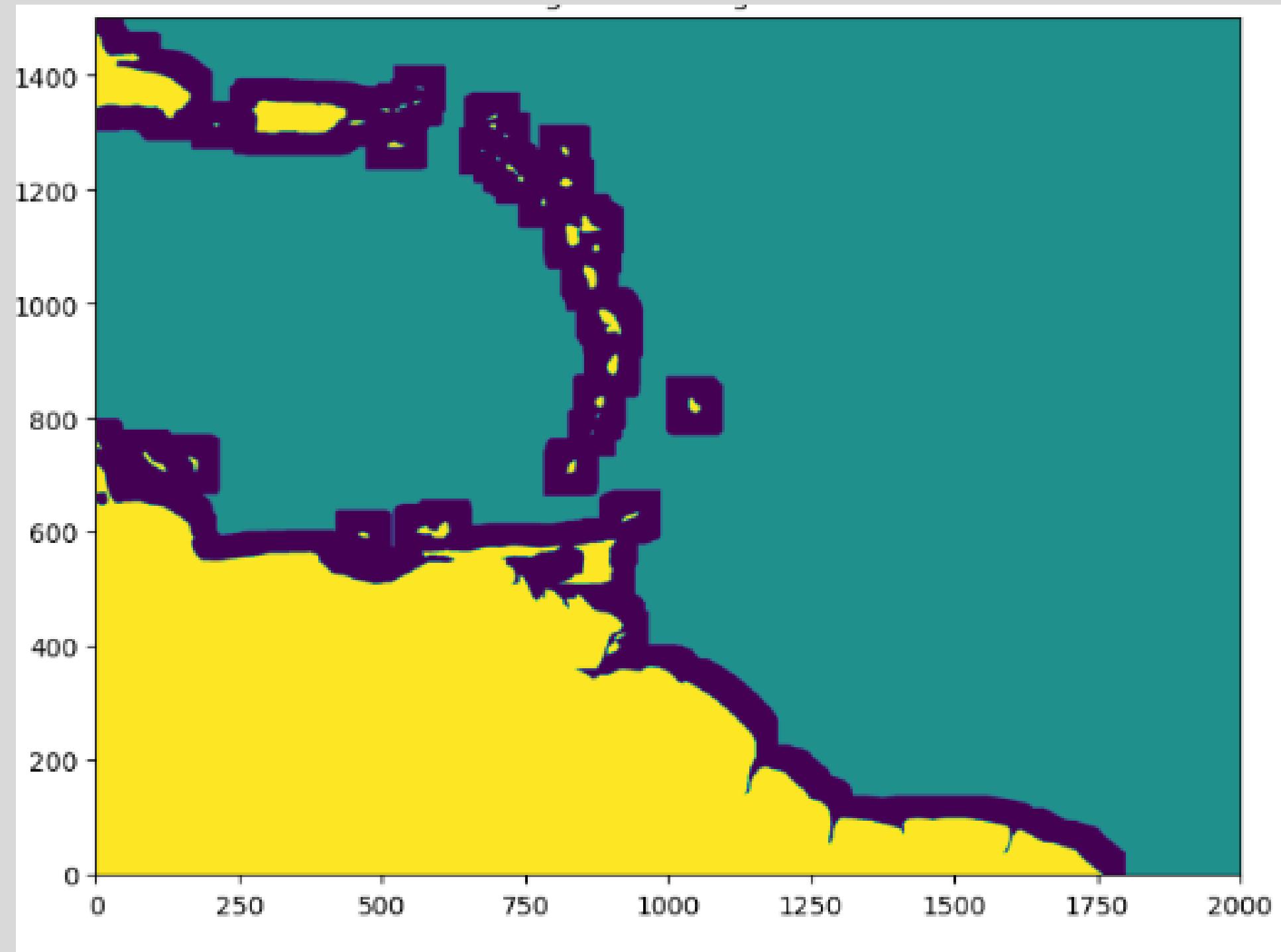
Images satellites

zone

	Indice	resolution spatiale	revisite	période
MODIS	AFAI	1 km	~ 3 h	mai 2022
GOES (CLS)	FAI	1 km	10 min	mai 2022



Masque



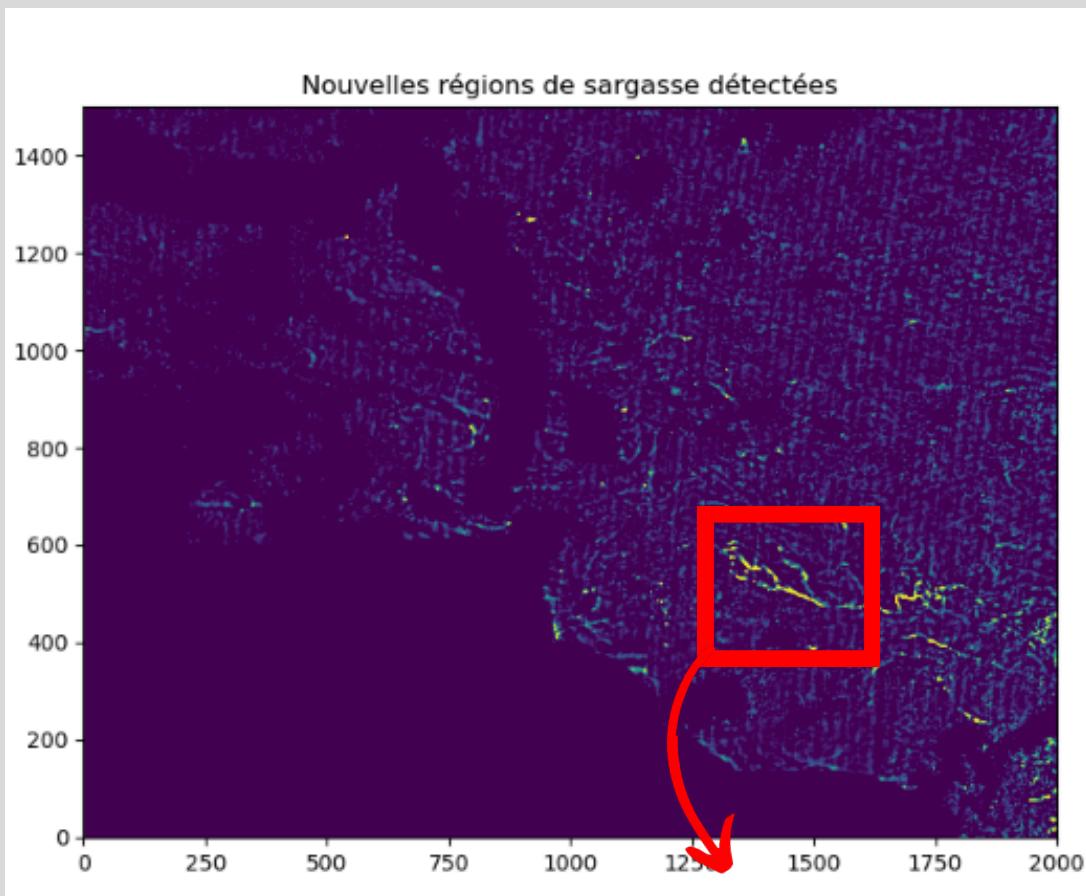
Masque terre

Masque : terre + 75km autour des cotes

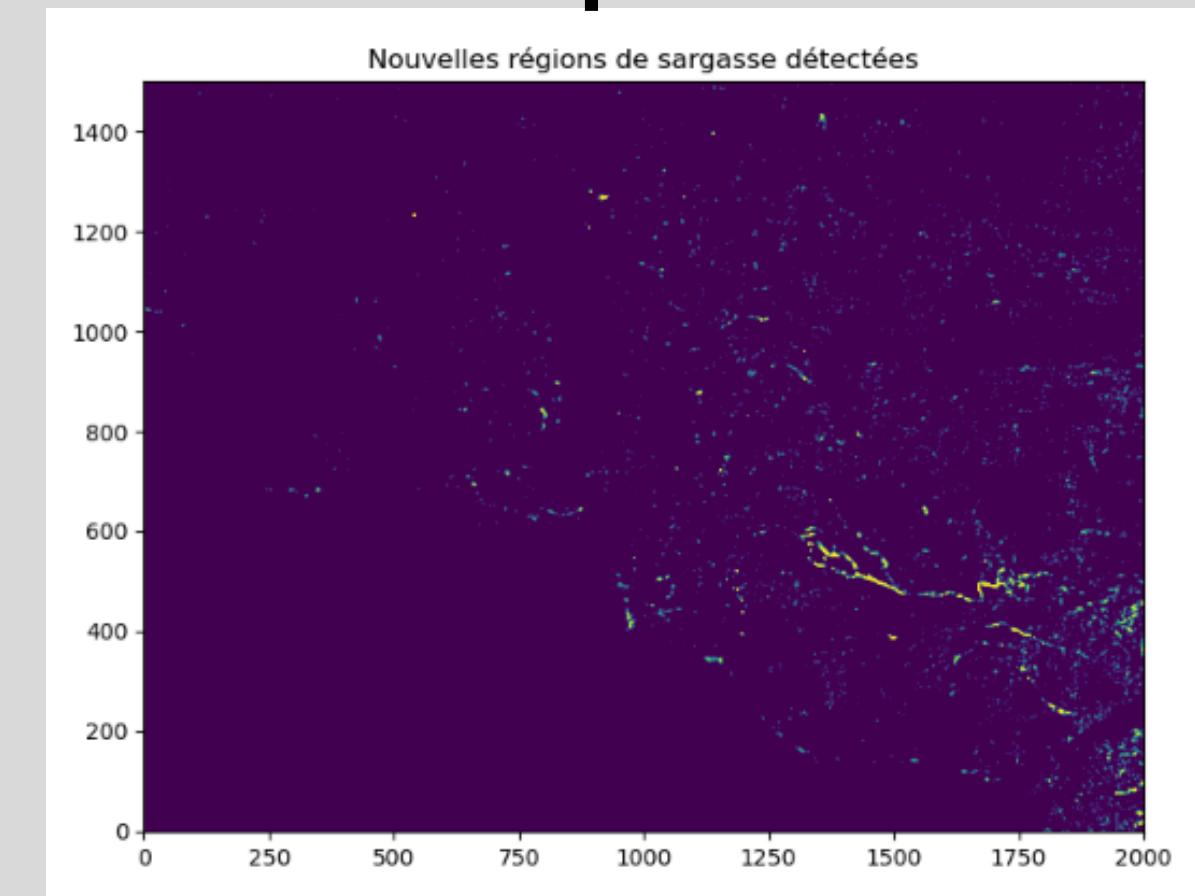
Masque

Masque sargasse

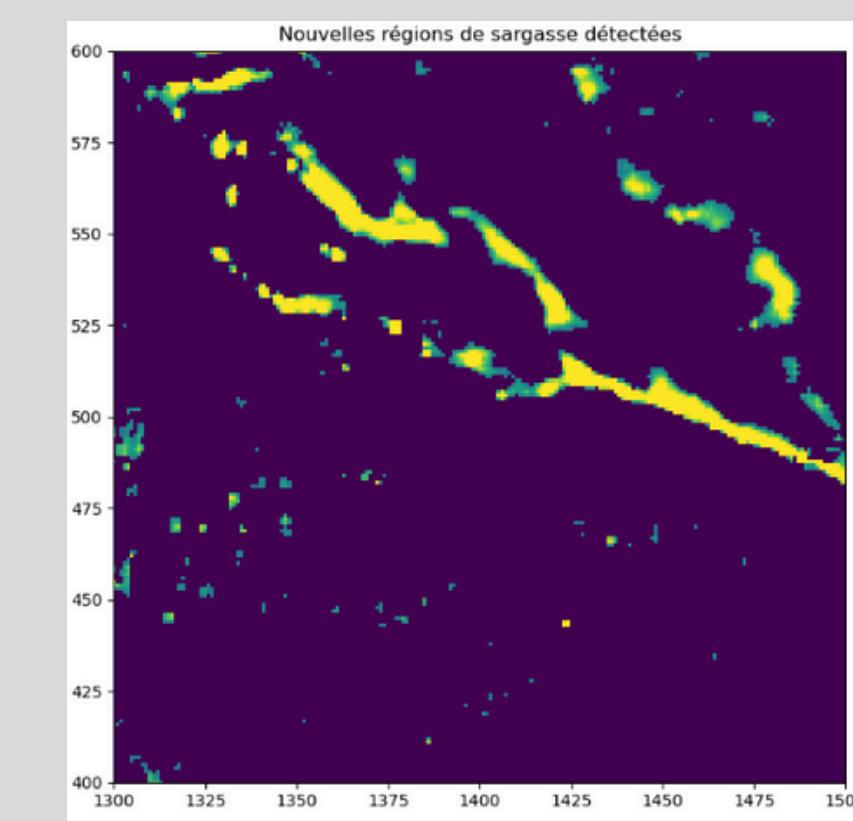
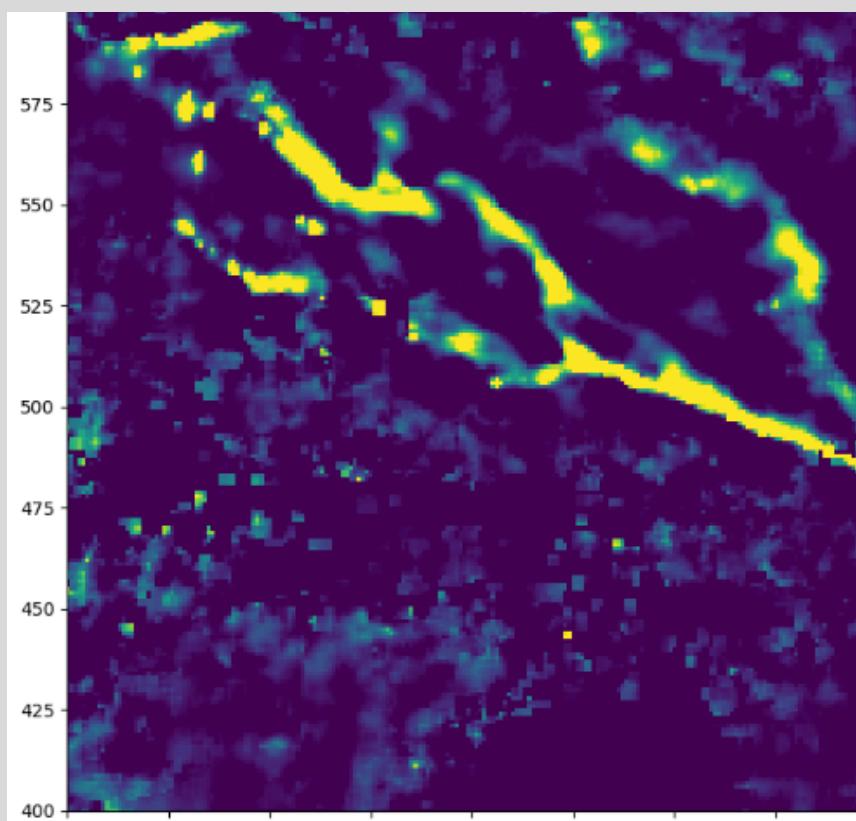
Avant



Après



seuil utilisé : fai_anomaly = 0.0002



Lissage

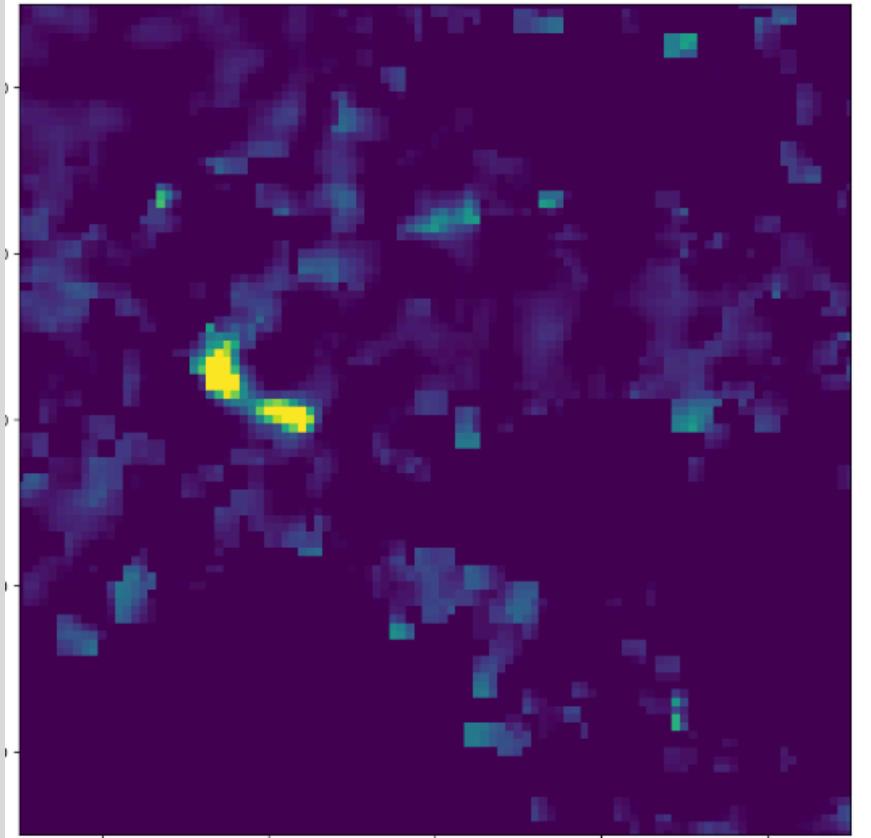
dt_lissage

3h

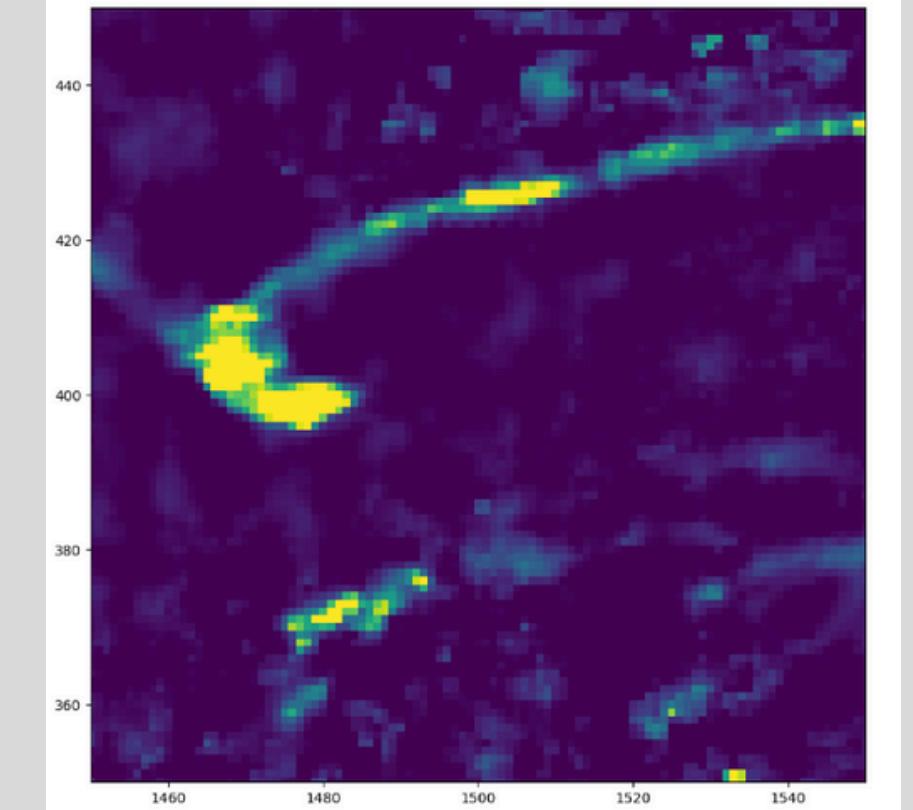
3h



dt_offset

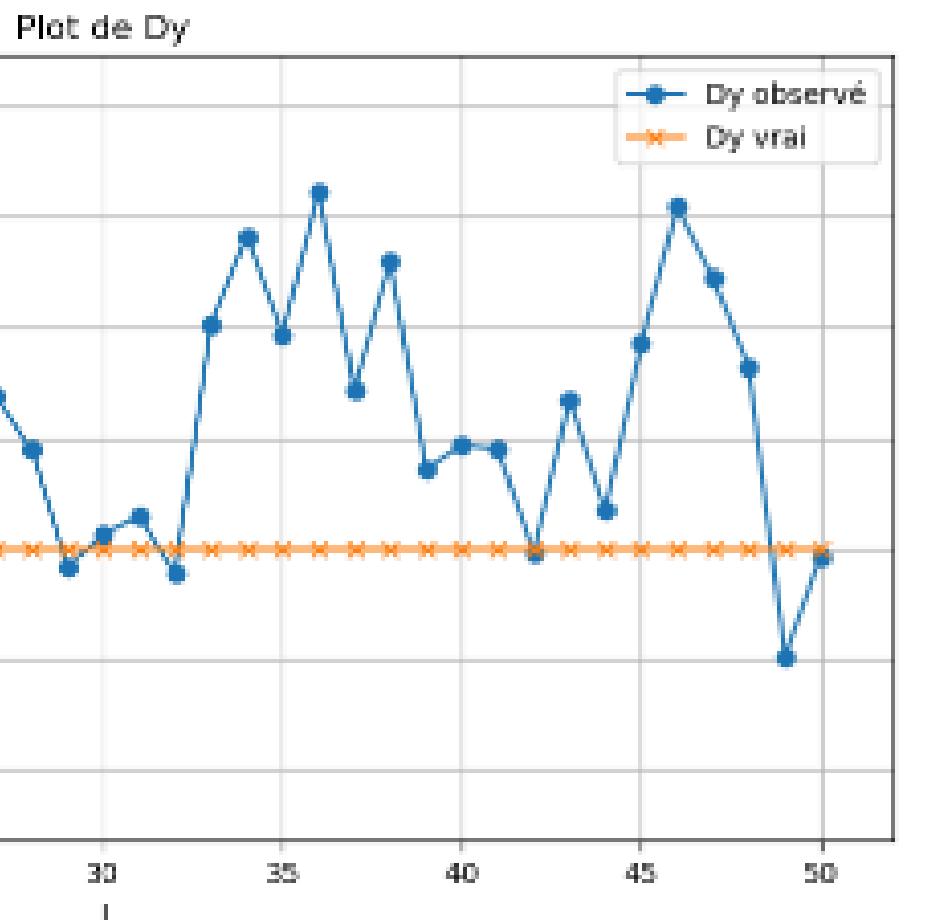
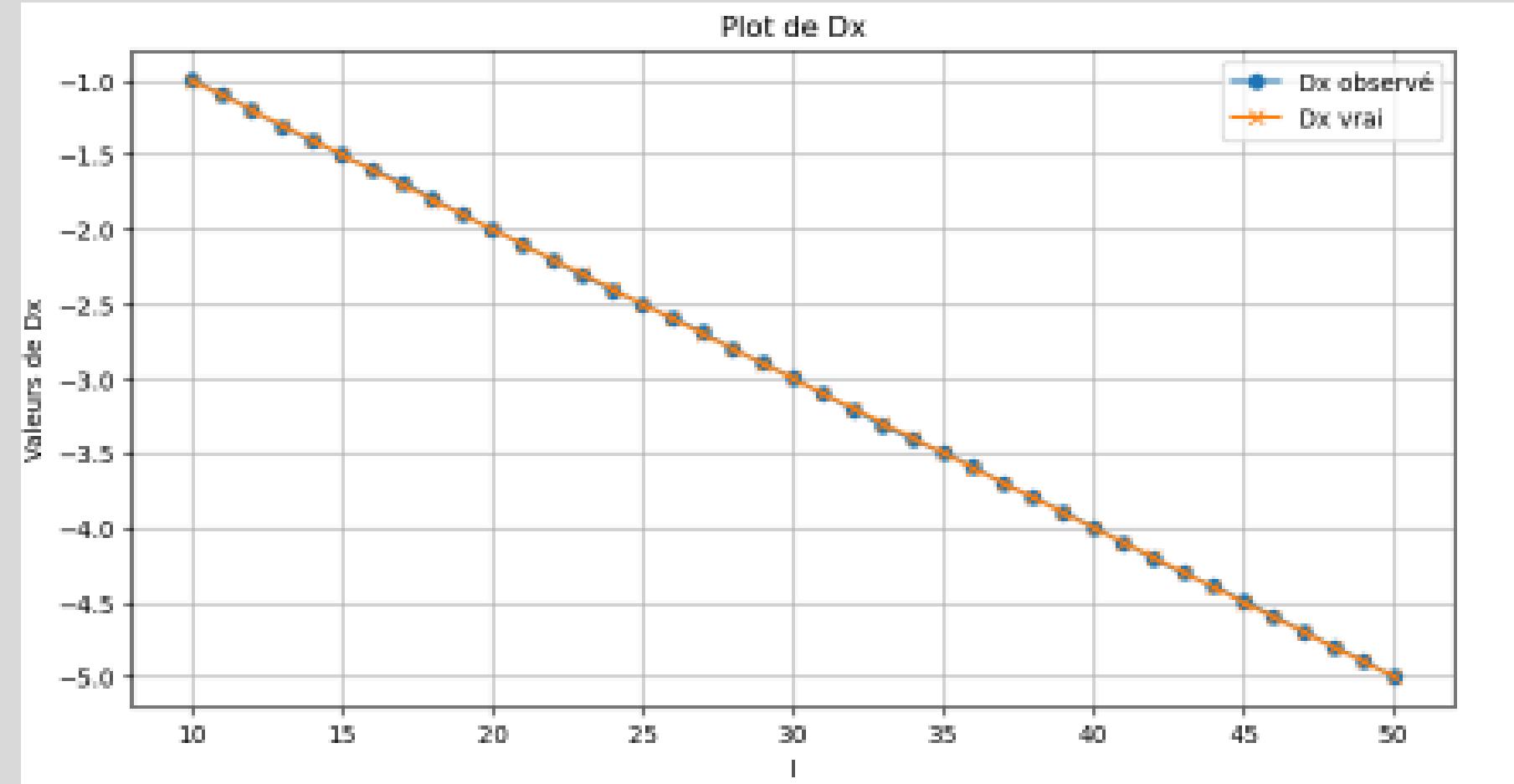


lissage 30min



lissage 3h

Lissage



`modis_translation_lissage_region()`

Pas d'effet important trouvé

Détection ROI

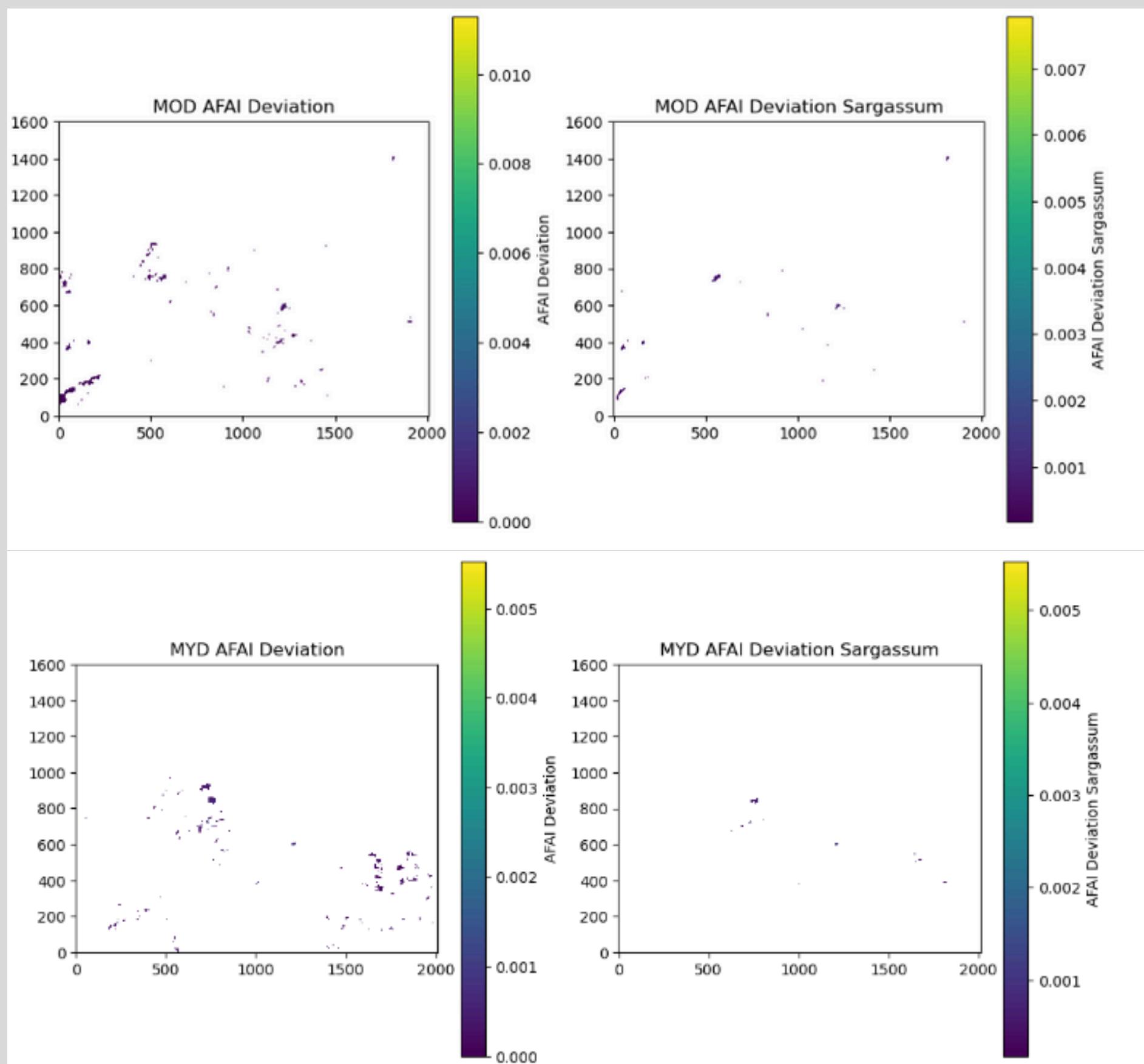


fig 1 : détections MOD et MYD

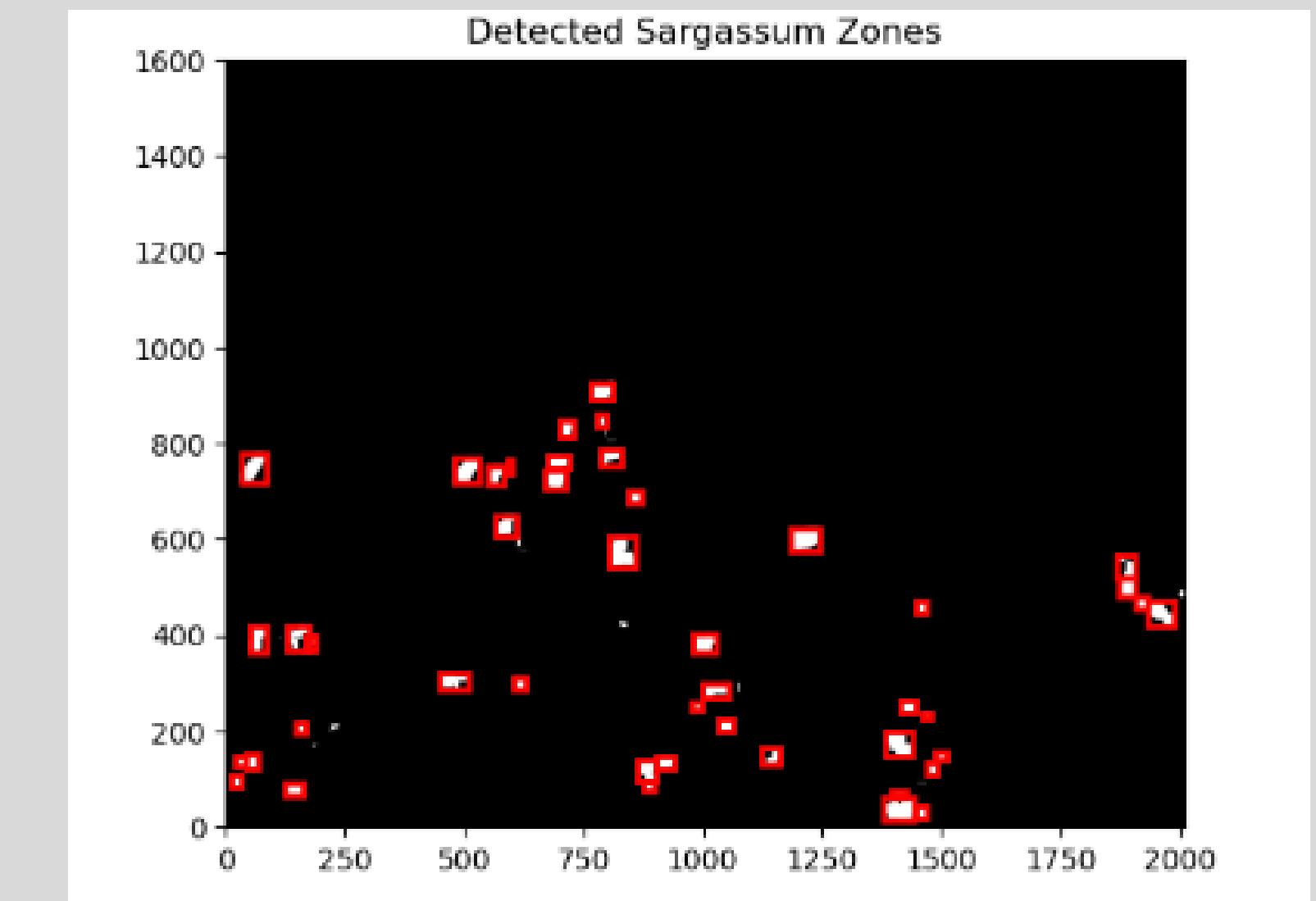
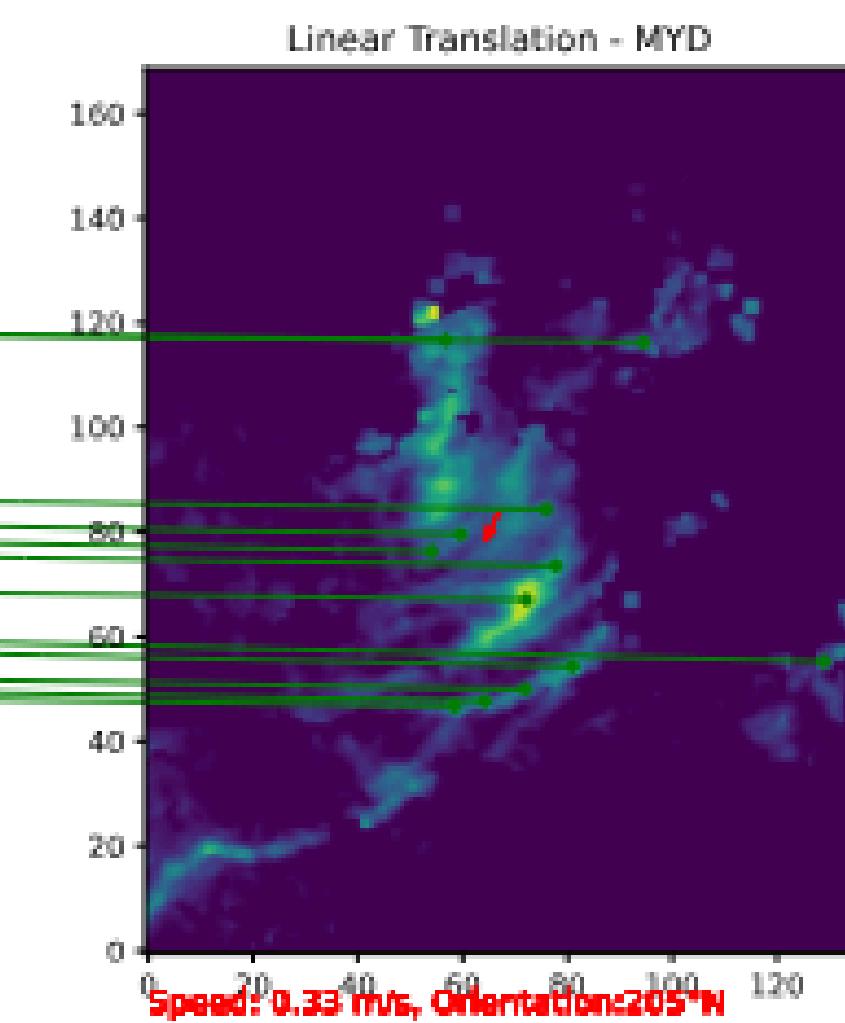
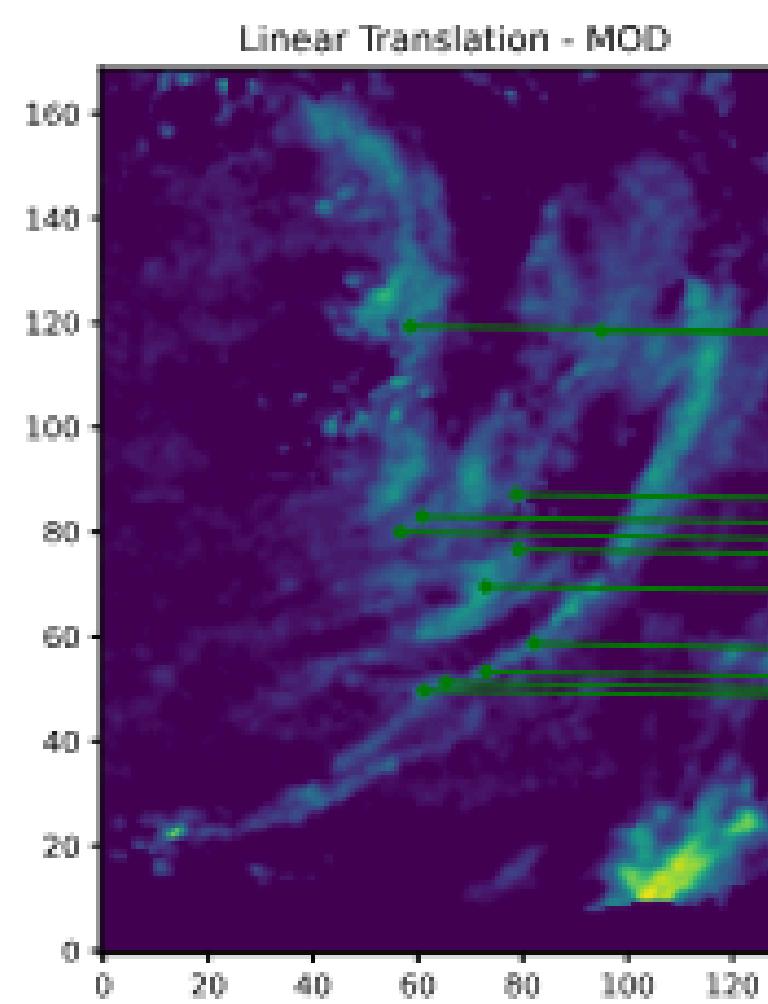
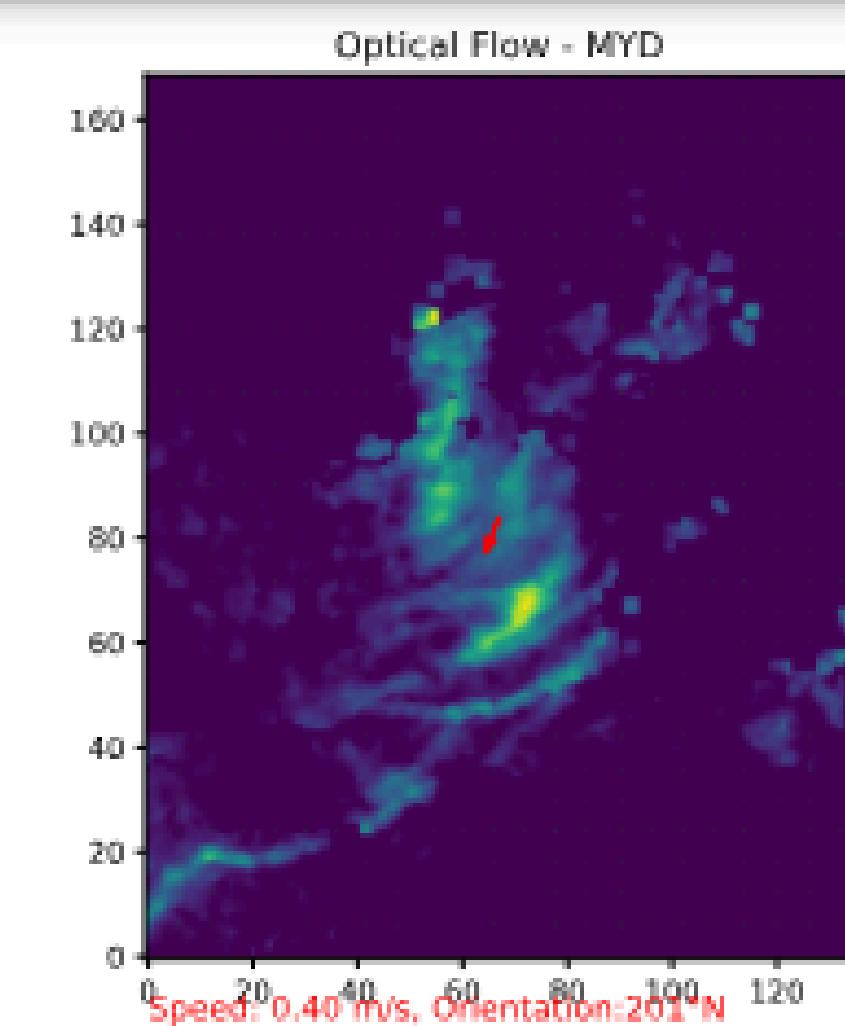
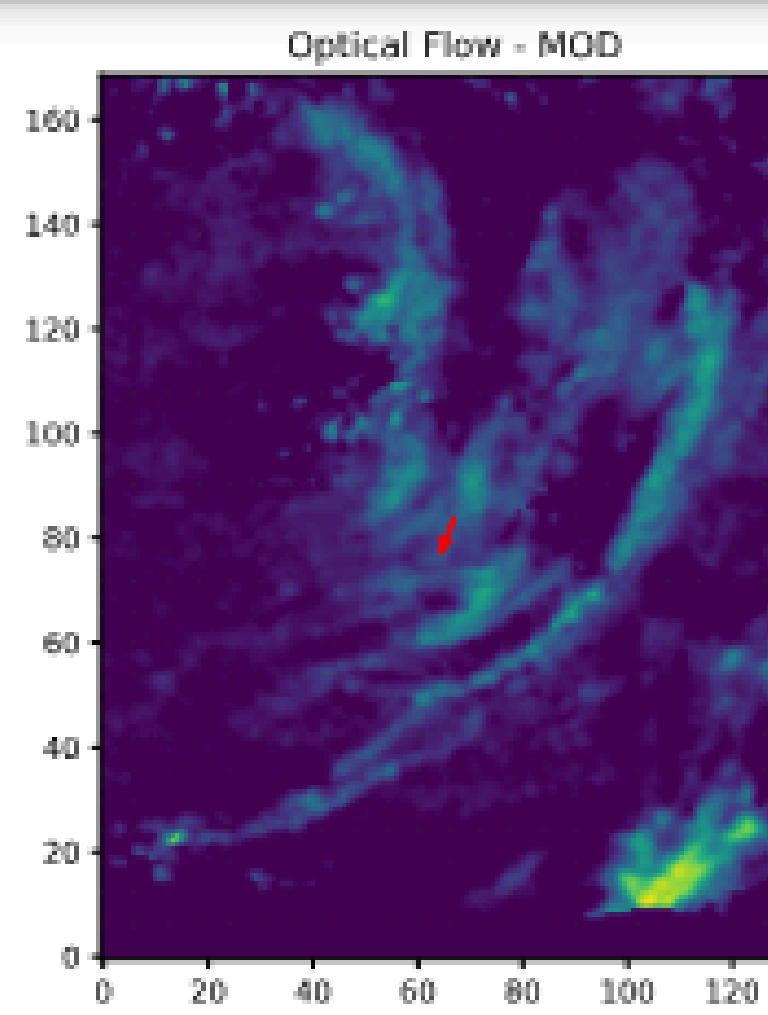


fig 2 : Extraction des ROI
paramètres :
-érosion : 3x3
-dilatation : 10x10

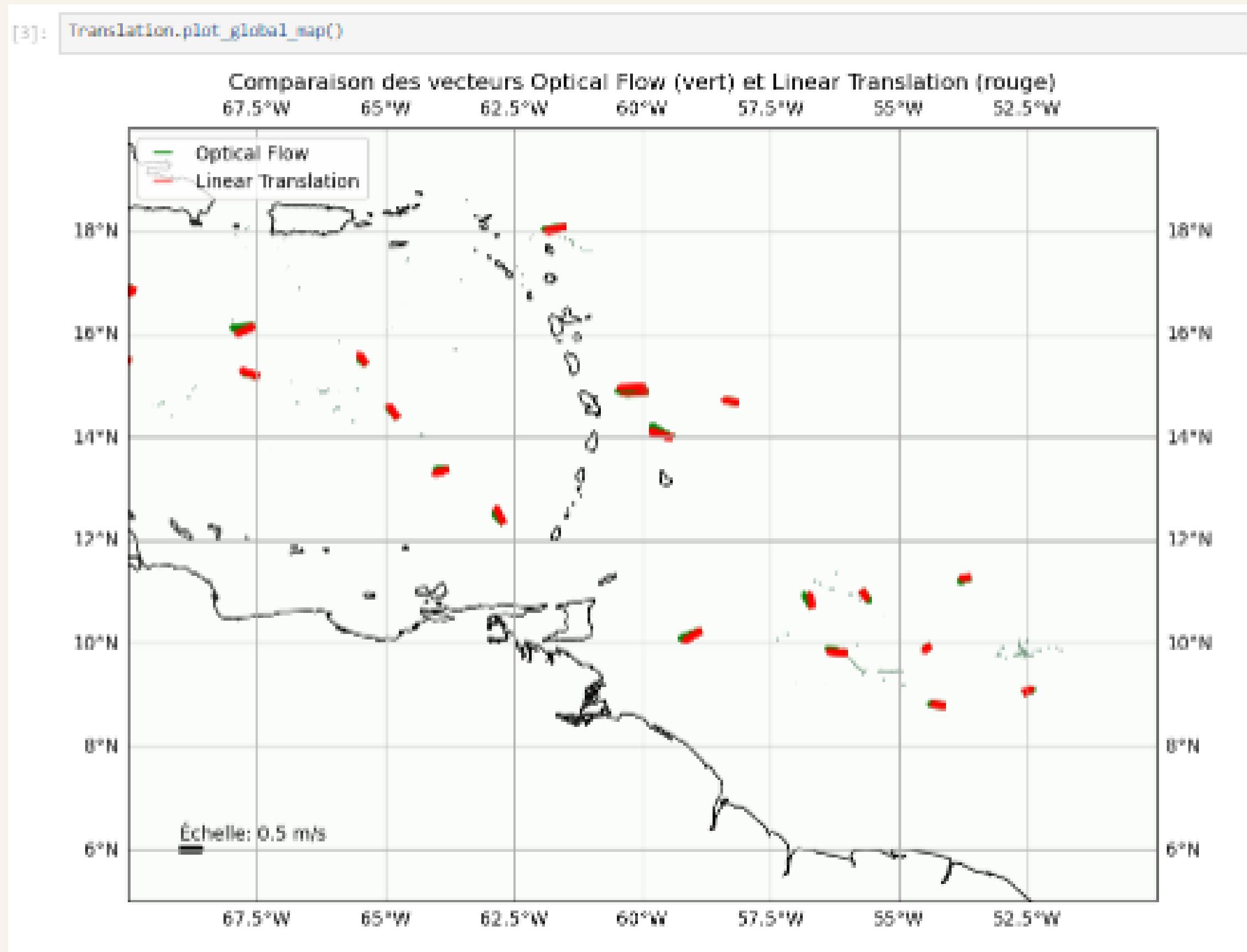
Résultats



Une ROI MODIS



Carte global



`plot_global_map()`

csv points_infos

	A	B	C	D	E	F	G	H	I	J
1	Date	Mod_Lat	Mod_Lon	Myd_Lat	Myd_Lon	u_OF	v_OF	u_LT	v_LT	Sat
2	20230602_11	9,96744635	-54,3860607	9,69398335	-54,3509316	-0,18934956	-0,12583265	-0,18083239	-0,15350293	Goes
3	20230602_11	8,79021069	-54,0883573	8,86527308	-54,0153584	-0,39060209	0,04480682	-0,33638342	0,0383994	Goes
4	20230603_11	12,316881	-62,6838105	12,3985813	-62,4415511	-0,24614266	0,26336175	-0,18628163	0,36779269	Goes
5	20230605_11	14,3595629	-64,7446699	14,3526064	-64,7403948	-0,2542752	0,27815181	-0,21428904	0,29865656	Goes
6	20230606_11	9,79466381	-56,0071771	9,63573853	-55,869094	-0,45676218	0,11401498	-0,43667458	0,05383421	Goes
7	20230608_11	10,67985	-56,6936716	10,5140097	-56,6508178	-0,18693964	0,33421695	-0,05046943	0,33889739	Goes
8	20230608_11	11,0406387	-55,733964	10,6549991	-55,4280103	0,18681965	-0,27663851	0,14318413	-0,23889899	Goes
9	20230609_11	9,03741669	-52,6022337	9,00959118	-52,5998969	0,26983511	0,0896655	0,2344005	0,10249322	Goes
10	20230609_11	13,3813554	-63,7600528	13,8261209	-63,6616465	-0,32023445	-0,00516792	-0,34304116	-0,1172791	Goes
11	20230609_11	15,4736016	-69,9364499	15,4707051	-69,9358542	-0,26212679	0,06593015	-0,22997602	0,03435406	Goes

exemple date : 20230601_1130_1430 => 1er juin 2023 mod : 11h30 et myd 13h30

Détection dans le script si il y a déjà un point pour ce satellite à cette date et ces coordonnée et dans ce cas là => pas écriture

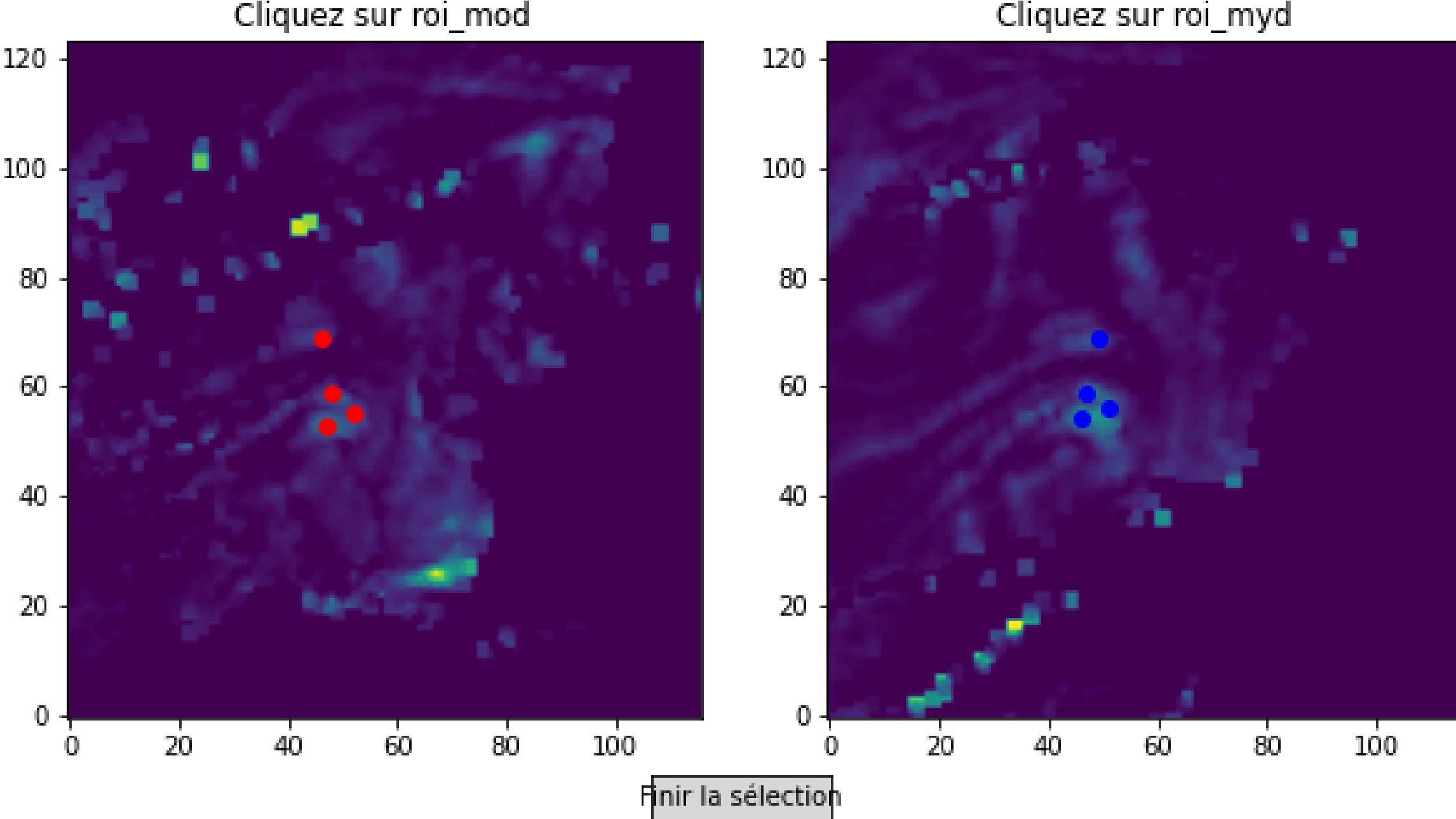
Validation

MODIS

Appariement manuel (clic)

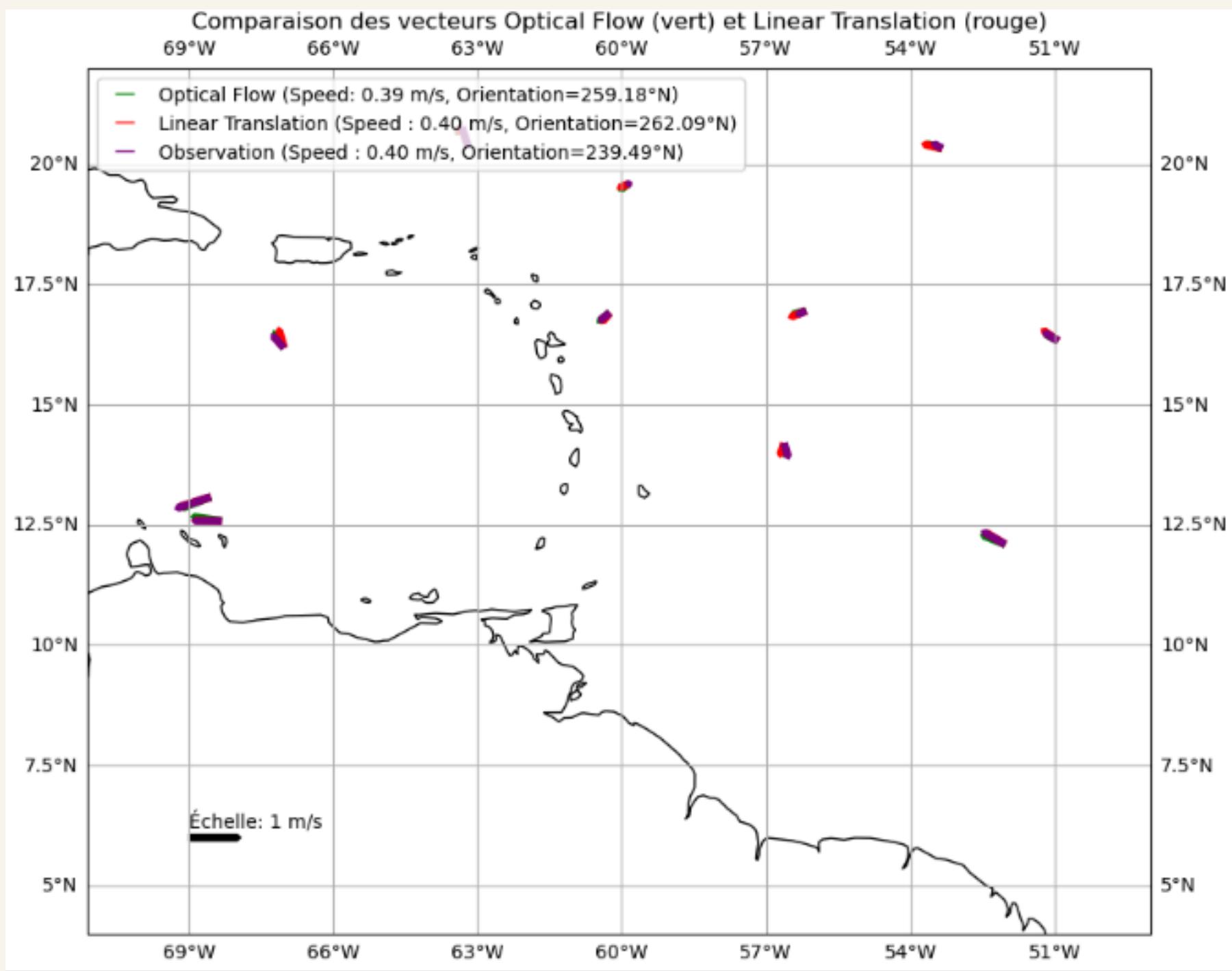
Script python :
Validation_modis.py

Stockage dans :
Points_info_modis.xlsx

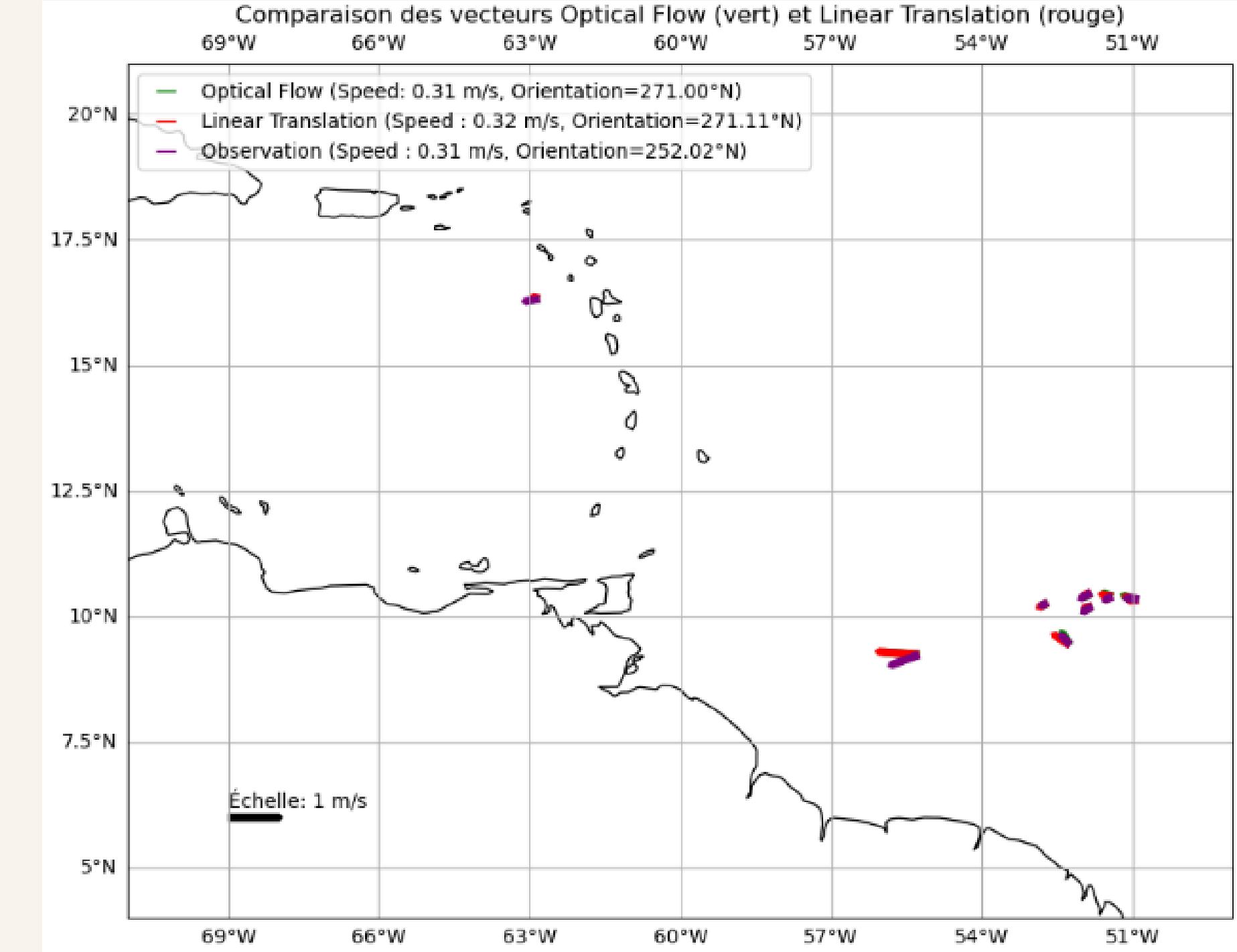


A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Date	Mod_Lat	Mod_Lon	Myd_Lat	Myd_Lon	u_m	v_m	Mod_Lat	Mod_Lon	Myd_Lat	Myd_Lon	u_OF	v_OF	u_LT	v_LT	
20230602_18,00072	18,00072	-59,8857	18,02752	-59,9081	-0,20691	-0,24819	18,08274	-60,3183	17,90393	-60,2721	-0,3293	-0,3282	-0,3518	-0,26378	
20230602_11,42768	11,42768	-50,253	11,42597	-50,2794	-0,24456	0,015809	11,31341	-50,5793	11,32116	-50,3878	-0,24558	0,163193	-0,32866	0,13292	
20230602_17,276	17,276	-60,2118	17,29463	-60,2366	-0,23039	-0,17248	16,83354	-60,5504	17,08389	-60,5335	-0,24979	-0,19893	-0,193	-0,22563	
20230603_16,65183	16,65183	-57,5989	16,67719	-57,6184	-0,18013	-0,23483	16,5149	-58,2096	16,44188	-58,0747	-0,33973	-0,22239	-0,30831	-0,27818	
20230603_15,4889	15,4889	-56,274	15,49016	-56,3194	-0,42103	-0,01161	15,27804	-56,4939	15,46537	-56,4165	-0,36594	0,309244	-0,41691	0,196302	
20230603_15,51359	15,51359	-58,4041	15,49635	-58,4216	-0,16235	0,159625	15,90767	-58,9203	15,17915	-58,9194	-0,3078	-0,01498	-0,29051	-0,12673	
20230603_14,22715	14,22715	-69,9416	14,27636	-69,9704	-0,26634	-0,45566	13,6846	-70,0517	14,24051	-70,0512	-0,35861	-0,34916	-0,2323	-0,40498	

Validation



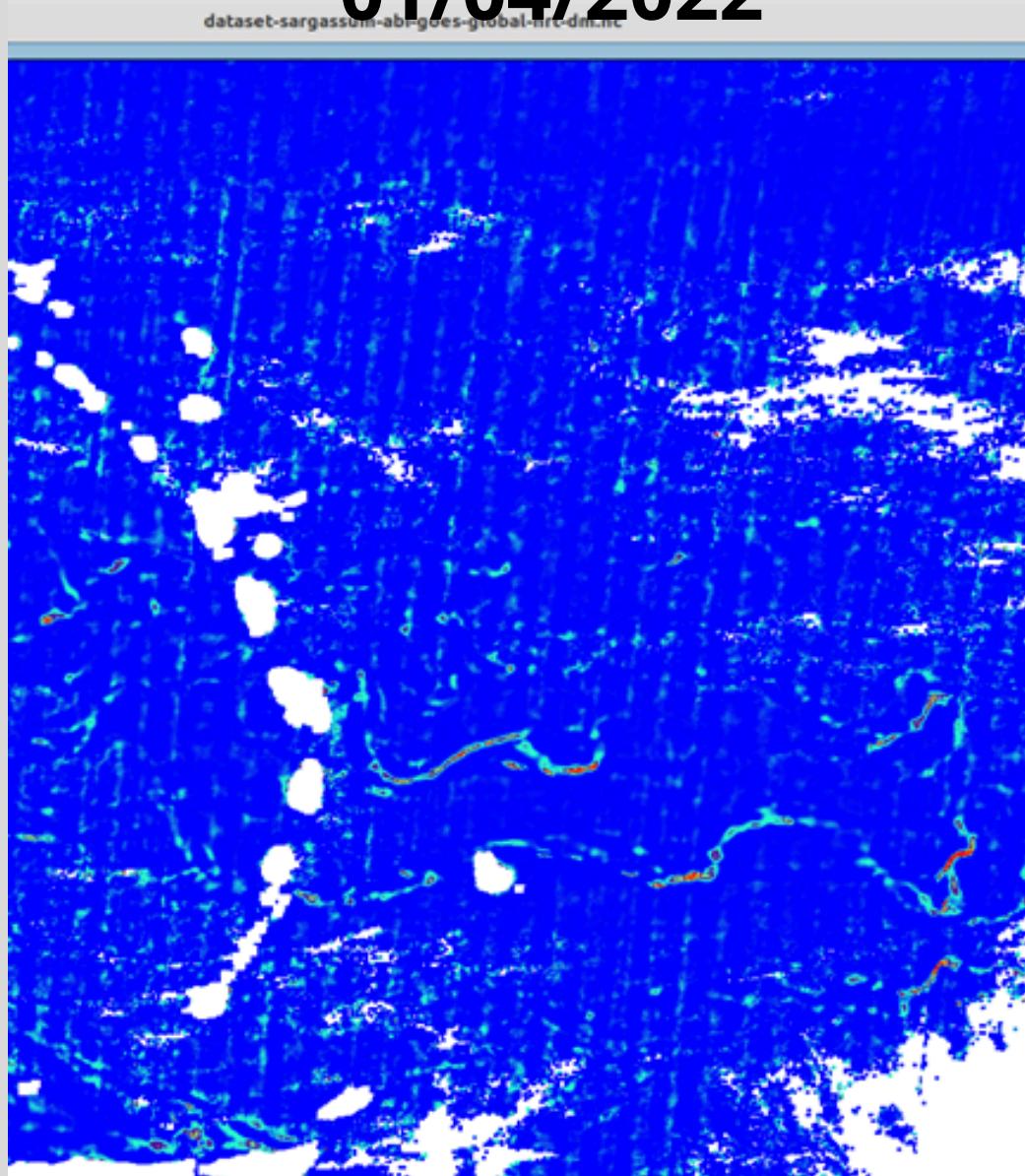
MODIS



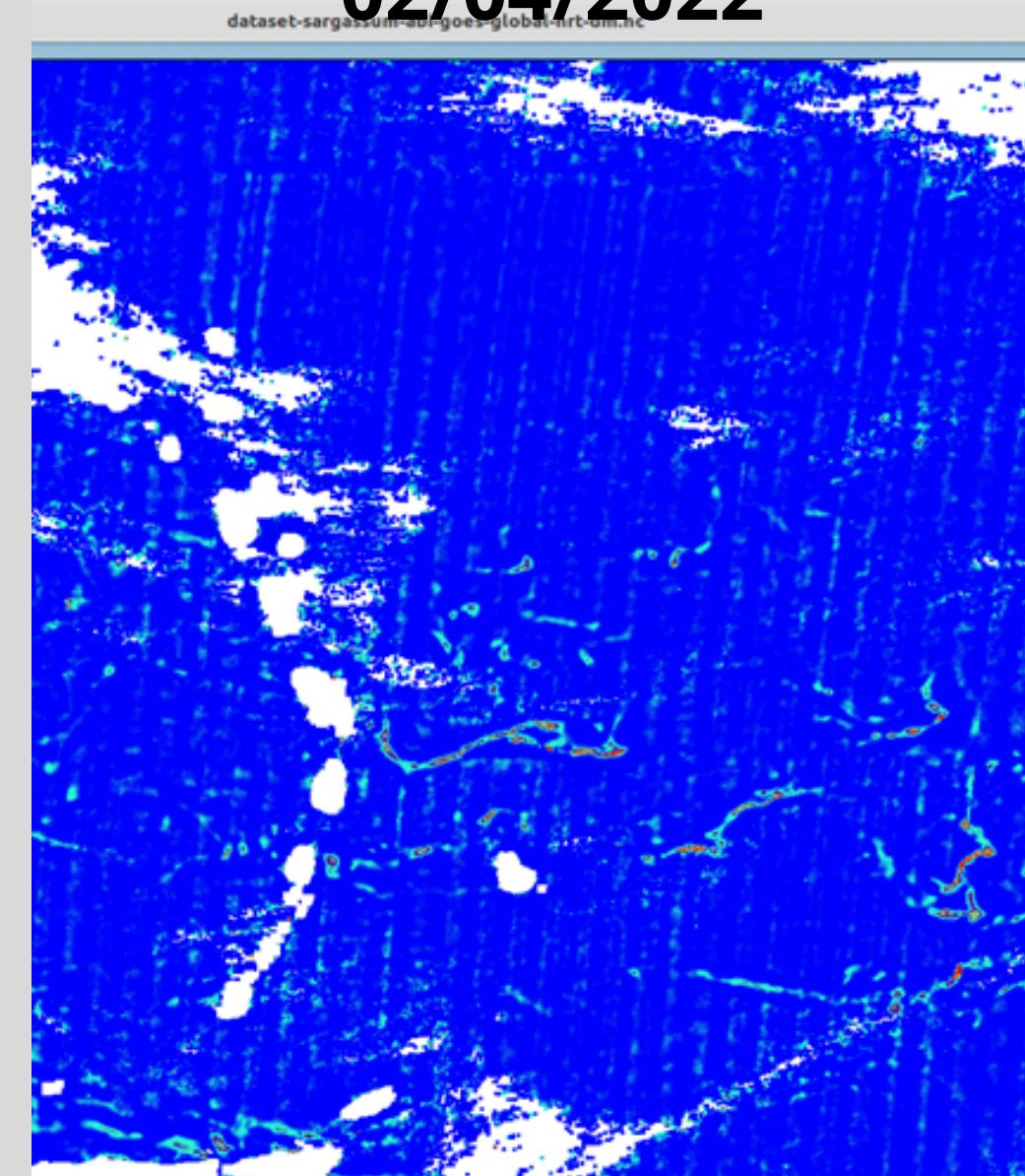
GOES

Daily version

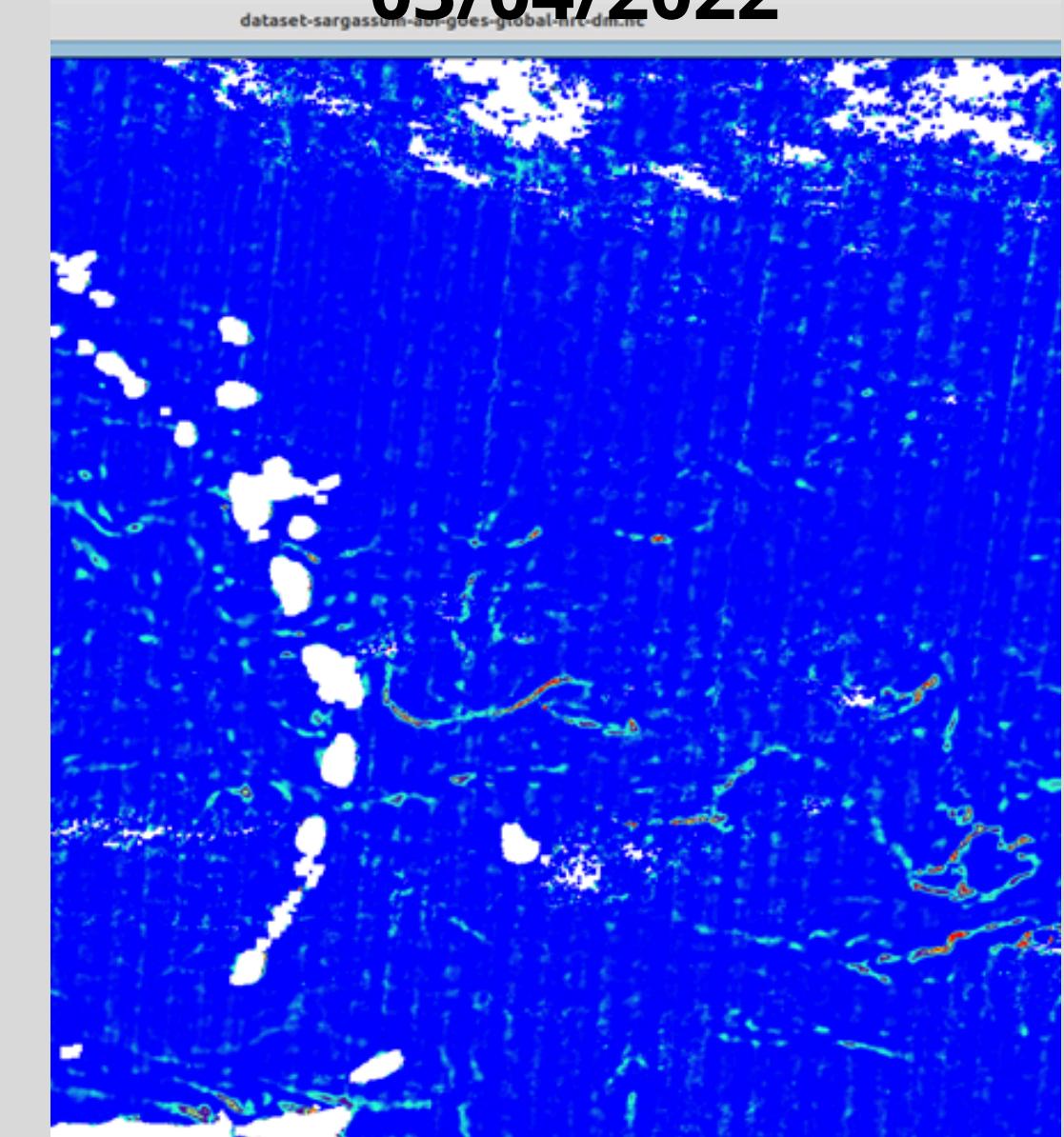
01/04/2022



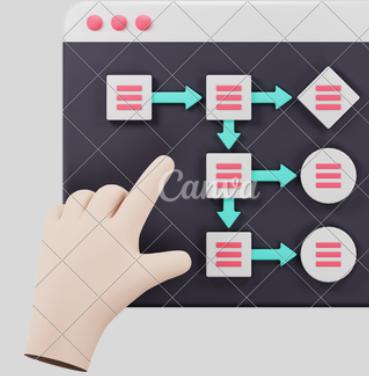
02/04/2022



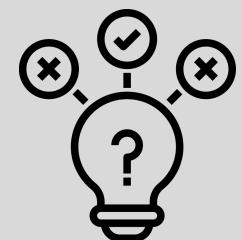
03/04/2022



Algorithmes

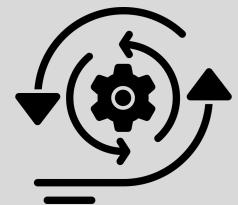


Optical flow



Hypothèses :

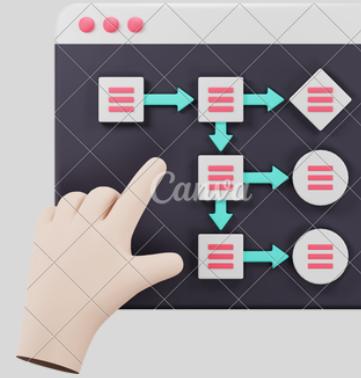
- Intensité des pixels constantes
- pixels voisins ont un déplacement identique



Principes :

- Reconnaissance des motifs
(Utilisation de polynôme quadratique pour approximer les pixels aux alentours puis comparaison de l'évolutions de ces polynômes entre les images pour déduire le champ de vitesse)

Algorithmes



Optical flow

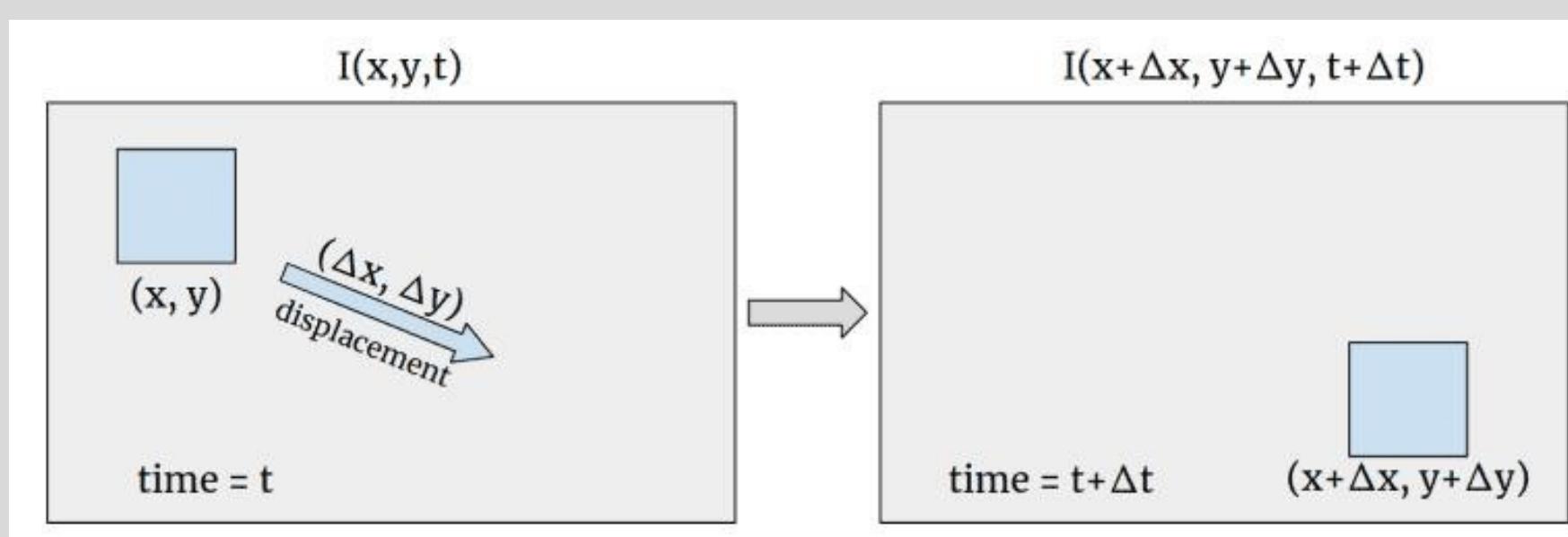


fig 3 : Principe de l'optical flow

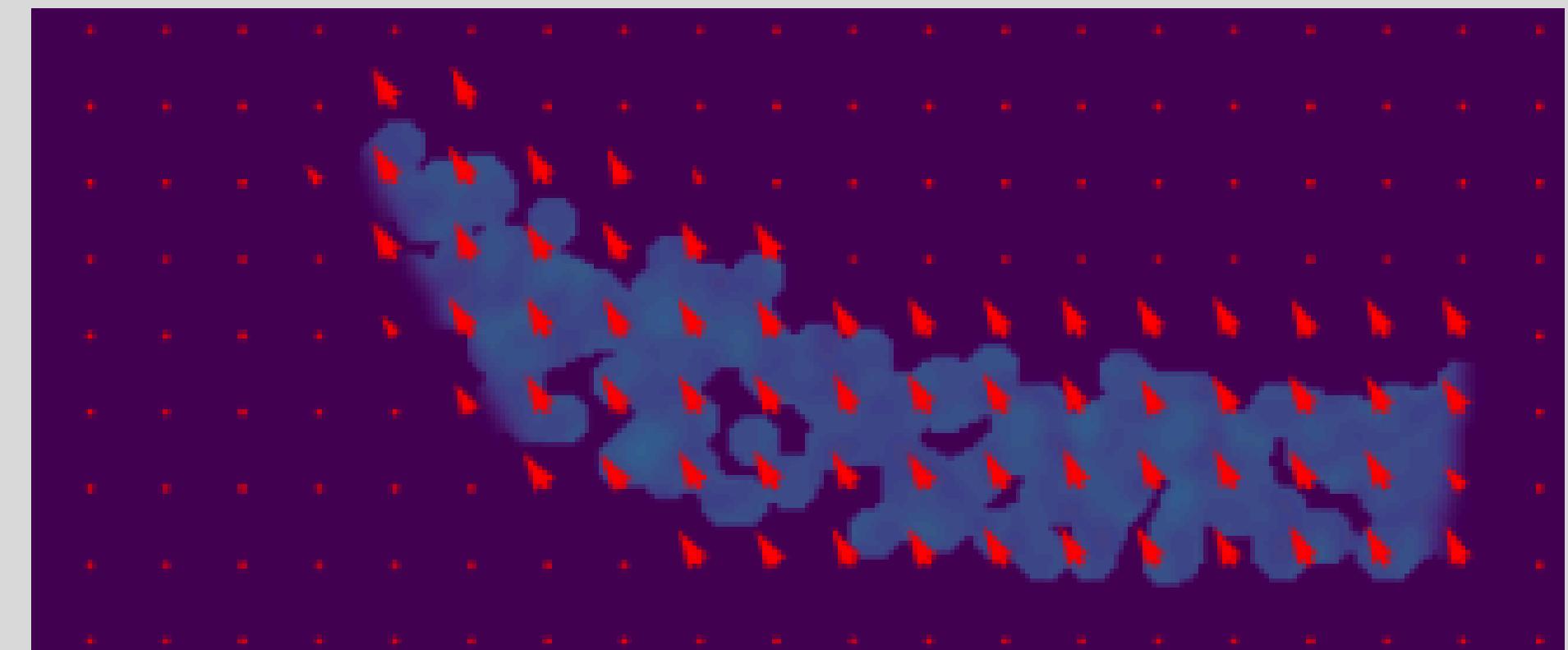
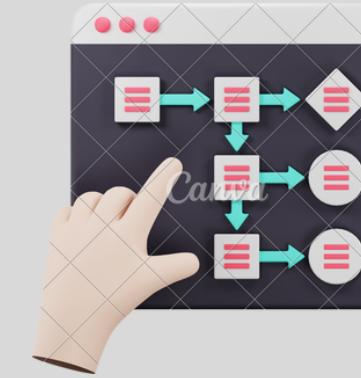


fig 4 : Résultats de l'optical flow sur un patch de sargasse artificiel au déplacement imposé

Algorithmes



SIFT

(Scale Invariant Feature Transform)

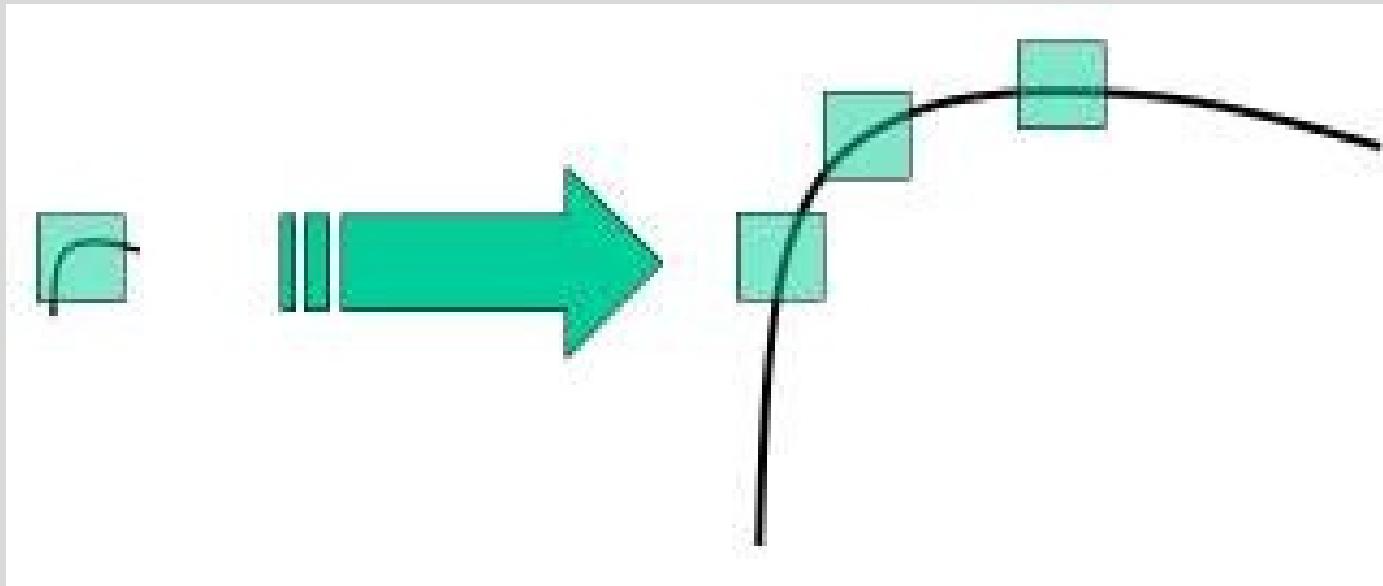
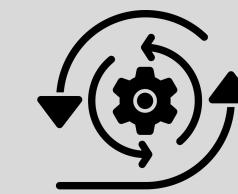


fig 5 : Un coin n'est plus un coin si il y a eu un redimensionnement

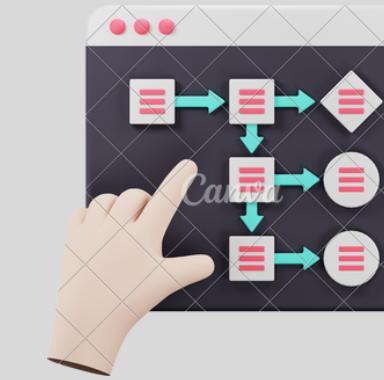


Principes :

- repérage de points clefs (max de gradient qui sont détectable dans les 2 images)
- appariement des points-clefs selon proches voisins (SIFT original) et selon le nombre d'appariement générés. (Algo LinearTranslation)

(Hyp : mouvement identique partout sur l'image)

Algorithmes



SIFT (Scale Invariant Feature Transform)

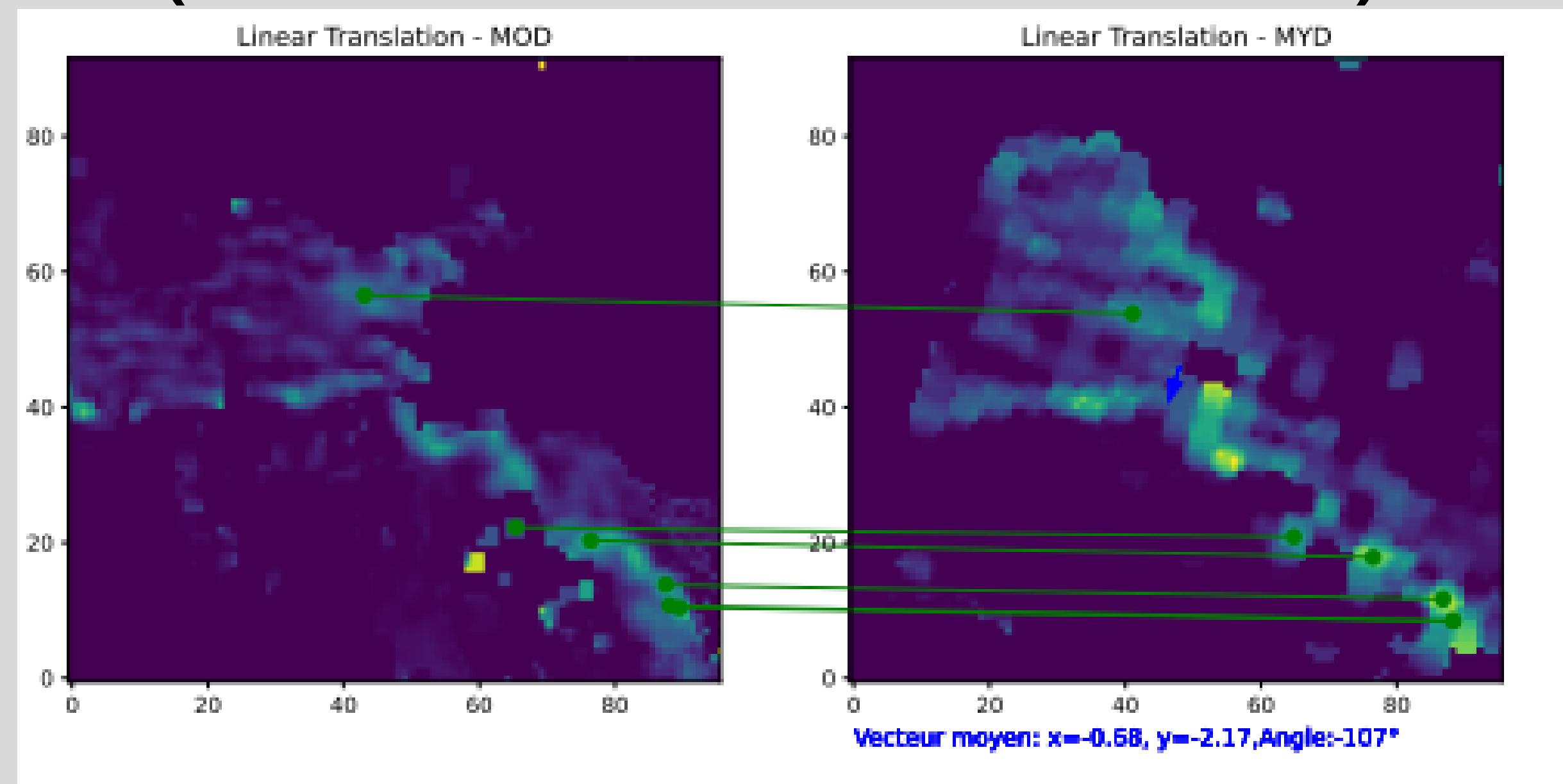


fig 6 : Exemple de l'utilisation de SIFT