**Arborescence SDB : (*Read me)***

***Racine :***

***SDB*** >

***my***\_***packages***/ >

My\_Geoprocess.py >

Package personnel regroupant les scripts de fonctions

***OldVersion***/ >

My\_Geoprocessold.py >

Ancienne version du package de fonctions perso.

***Data\_Shom***/ >

***Global***/ >

Regroupe la donnée SHOM

***ScriptsHorsChaine***/ >

Regroupe l’ensemble des scripts tests avant insertion dans la chaine

***Dossier\_Applications***/ >

Regroupe les scripts et produits issus des différentes applications.

Procédure

Idéale : peu de valeur sans erreurs ni bruit.

***Application : Poe***

***Arborescence de dossiers***

***Poe***/ >

***Figures***/ >

Regroupe les figures de l’article

***Recap***/ >

Regroupe les PowerPoint et illustration des présentations

***Résultats***/ >

***Bathymetries***/ >

***Nom\_Image\_Sentinel***/>

Regroupe les fichiers raster de bathy estimé avec l’image correspondant au dossier

***Classifications***/ >

Regroupe les fichiers de classif de types de fonds, anciennement utilisés.

***Scripts***/ >

***Classification***/ >

GetKmeansClassif.ipynb >

Fichier jupyter produisant un raster de classification du type de fonds (Plus utilisé dans la methodo actuelle)

***InSituDataManagement***/ >

GetCleanInSituSurvey.ipynb >

Fichier jupyter de conversion des fichiers de données terrain brutes (.RAW) en fichiers .txt regroupant : Heure, Lon(decimales), Lat (decimales), Profondeur.

Tide\_Correction.ipynb >

Fichier jupyter de correction des données terrain a partir de l’instrument de mesure de la marée.

***Modeling***/ >

GetBathy\_Lyzenga.ipynb >

Fichier jupyter de traitement des données Shom intra-pixel, de calibration des modèles Lyzenga et de production d’un raster de bathymétrie estimées.

GetBathy\_Lyzenga\_WithFieldData.ipynb >

Fichier jupyter de traitement des données terrain intra-pixel, de calibration des modèles Lyzenga et de production d’un raster de bathymétrie estimées.

GetBathy\_Stumpf.ipynb >

Fichier jupyter de traitement des données Shom intra-pixel, de calibration des modèles Stumpf et de production d’un raster de bathymétrie estimées.

GetBathy\_Stumpf\_WithFieldData.ipynb >

Fichier jupyter de traitement des données terrain intra-pixel, de calibration des modèles Stumpf et de production d’un raster de bathymétrie estimées.

GetBathy\_Stumpf\_Illustration.ipynb >

Copie de GetBathy\_Stumpf\_WithFieldData pour générer les illustrations de l’article.

***OpticalDataManagement***/ >

***Old\_tests\_DII***/ >

Regroupe les fichiers test de scriptage de la production de DepthInvariantIndex

Netcdf2Tiff.ipnyb >

Fichier jupyter de convertion NetCDF to Tiff (Conversion imprécise…)

ConvertCetcdf2Tif.py >

Script python de convertion NetCDF to Tiff (Conversion imprécise…)

Copyjp200MT2NcIntiff.py >

Script python de d’ extraction de la geotransform des métadonnées d’un raster de même empreinte et de même résolution au format jpeg2000 pour la conversions d’un NetCDF vers un Tiff (Conversion utilisé…)

TileProcessing.ipnyb >

Génère les différentes empreintes de l’image (sol, ombre de nuages…) par seuillage sur les tuiles de l’image.

GetX.ipynb >

Fichier jupyter de génération des descripteurs des modèles Lyzenga

GetL\_Ratios.ipynb >

Fichier jupyter de génération des descripteurs des modèles Stumpf.

GetL\_DII.ipynb >

Fichier jupyter de génération des DII pour classification (plus utilisé).

GetL\_HomogeneityMap.ipynb >

Fichier jupyter de creation de carte d’homogeneité des fonds et des propriétés optiques (essai en test).

***VisualisationScripts***/ >

Regroupe des scripts de visualisation 3d. Pas utilisé car trop lourd à générer.

***Images***/ >

Regroupe les images non-traité sous dossier .SAFE/

***Acolite***/ >

***Nom\_Image\_Sentinel***/>

Image sentinel après correction atmospheric

***Shapes***/ >

Regroupe l’ensemble des shapefile d’empreintes

***CompressedData***/ >

Regroupe les descripteurs des différents modèles sous-forme de donnée compressée

***Terrain***/ >

***Echantillonage\_Fonds***/ >

Regroupe les photos des types fonds, prises lors de la sortie terrain

***MesuresBathy***/ >

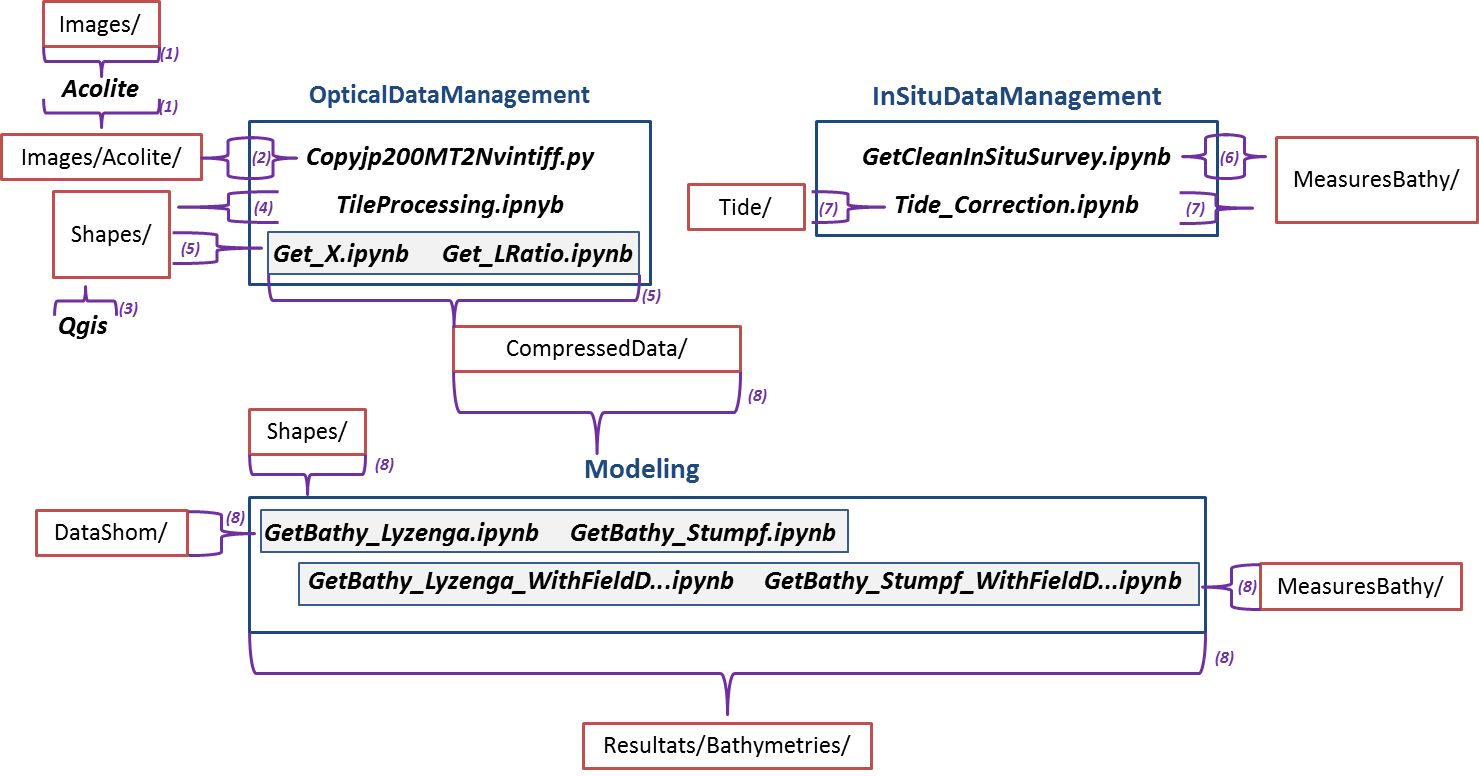
Regroupe les données bathy acquises par le sondeur lors de la mission terrain.

***Tide***/>

Regroupe les données mesurées par l’instrument lors de la mission terrain

***WeatherData***/ >

Regroupe les données metéo et le fichier jupyter permettant l’analyse en vue du choix du date optimal pour les prises de vue des images sat.

*** Procédure***

1. Correction atmosphérique des images par l’utilisation du logiciel Acolite (version antérieure)
2. Conversion des formats des rasters du NetCdf au tif
3. Création de l’empreinte de l’océan profond, de l’empreinte de la zone d’intérêt et des empreintes des zones de calibrations
4. Création de l’empreinte de sol et de vagues par seuillage sur la bande proche-infrarouge
5. Génération des données descripteurs grâce aux deux fichiers jupyter distincts. Enregistrement de ces données sous forme compressée.
6. Nettoyage des trames de mesures sondeur de terrains aux formats brutes. Association d’un point GPS pour deux à trois profondeur à la seconde. Conversion des coordonnées en décimales.
7. Traitement des données du profondimètre et correction de la marée sur les hauteurs d’eau mesurées.
8. Calibration des différents modèles grâce aux 4 fichiers jupyter distincts, sur la base des empreintes des zones de calibrations et respectivement avec les donnée SHOM et les données acquises lors de la mission terrain. Production des rasters de bathymétries estimées au format .tif, avec une seconde couche d’erreur interpolée par la méthode des plus proches voisins.

***Application : Petrie***

***Arborescence de dossiers***

***Petrie***/ >

***Résultats***/ >

***Bathymetries***/ >

***Nom\_Image\_Sentinel***/>

Regroupe les fichiers raster de bathy estimé avec l’image correspondant au dossier

***Classifications***/ >

Regroupe les fichiers de classif de types de fonds, anciennement utilisés.

***Scripts***/ >

***Modeling***/ >

GetBathy\_Stumpf\_RawData.ipynb >

Fichier jupyter de traitement des données Shom intra-pixel, de calibration des modèles Stumpf et de production d’un raster de bathymétrie estimées.

GetBathy\_Stumpf\_DeglintedData.ipynb >

Fichier jupyter de traitement des données Shom intra-pixel, de calibration des modèles Stumpf et de production d’un raster de bathymétrie estimées, le tout a partir d’image après Deglint.

***OpticalDataManagement***/ >

Netcdf2Tiff.ipnyb >

Fichier jupyter de convertion NetCDF to Tiff (Conversion imprécise…)

ConvertCetcdf2Tif.py >

Script python de convertion NetCDF to Tiff (Conversion imprécise…)

Copyjp200MT2NcIntiff.py >

Script python de d’ extraction de la geotransform des métadonnées d’un raster de même empreinte et de même résolution au format jpeg2000 pour la conversions d’un NetCDF vers un Tiff (Conversion utilisé…)

TileProcessing.ipnyb >

Génère les différentes empreintes de l’image (sol, ombre de nuages…) par seuillage sur les tuiles de l’image. Dans le Cas de Petrie, Génère une suite de raster supplémentaire après deglint.

GetX.ipynb >

Fichier jupyter de génération des descripteurs des modèles Lyzenga.

GetL\_Ratios\_Raw.ipynb >

Fichier jupyter de génération des descripteurs des modèles Stumpf.

GetL\_Ratios\_Deglinted.ipynb >

Fichier jupyter de génération des descripteurs des modèles Stumpf.

GetL\_DII.ipynb >

Fichier jupyter de génération des DII pour classification (plus utilisé).

GetL\_HomogeneityMap.ipynb >

Fichier jupyter de creation de carte d’homogeneité des fonds et des propriétés optiques (essai en test).

***BathyVisualisationAndProcessing***/ >

Regroupe des scripts de visualisation 3d. Pas utilisé car trop lourd à générer.

GetInterpolated.ipynb>

Produit un raster de bathymétrie estimée après l’interpolation des pixels sur corrigée par le processus du deglint.

***Images***/ >

Regroupe les images non-traité sous dossier .SAFE/

***Acolite***/ >

***Nom\_Image\_Sentinel***/>

Image sentinel après correction atmospheric

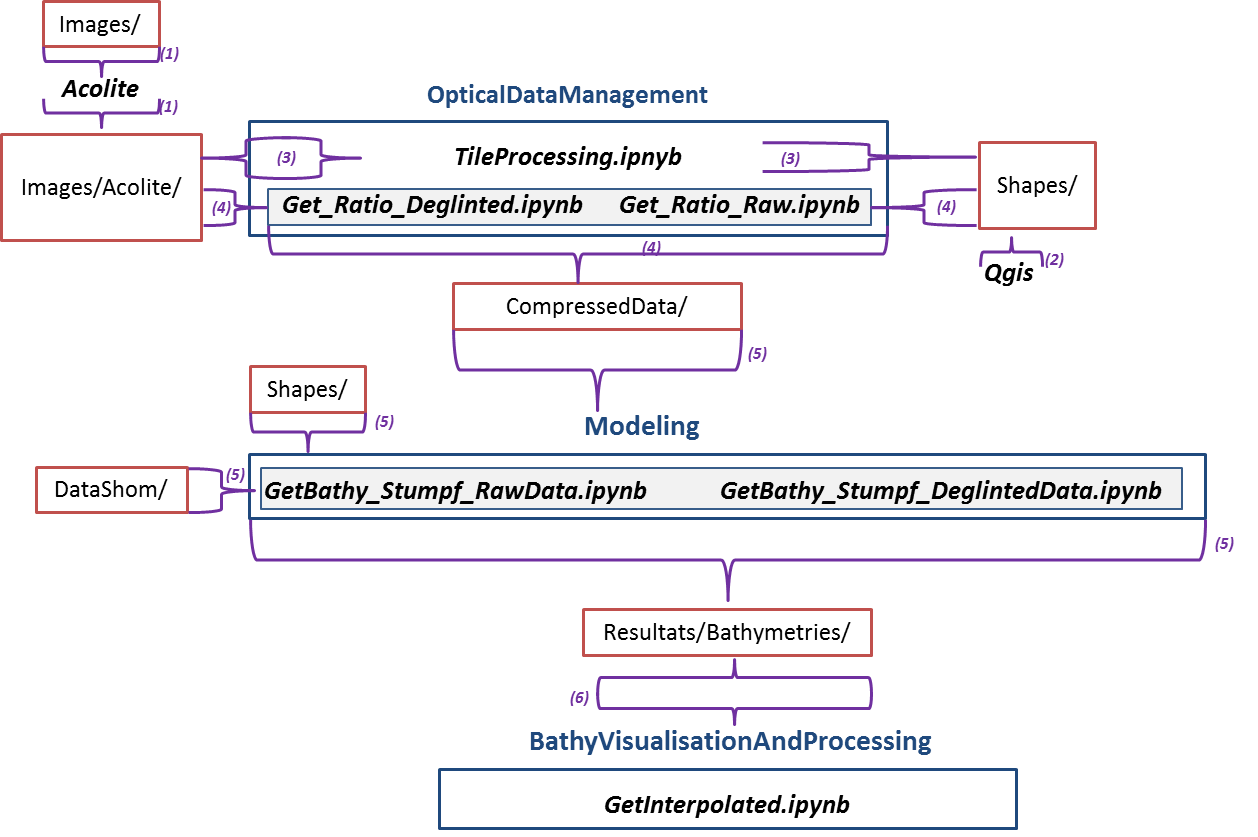
***Shapes***/ >

Regroupe l’ensemble des shapefile d’empreintes

***CompressedData***/ >

Regroupe les descripteurs des différents modèles sous-forme de donnée compressée

***Procédure***

******

1. Correction atmosphérique des images par l’utilisation du logiciel Acolite (Nouvelle version : réalise elle-même la conversion en tif et rogne convenablement l’image)
2. Création des empreintes de l’océan profond, de la zone d’intérêt, des échantillons à utilisés pour le déglint.
3. Création de l’empreinte de sol et de vagues par seuillage sur la bande proche-infrarouge et déglint des différentes bandes.
4. Génération des données descripteurs grâce aux deux fichiers jupyter distincts. Réalisation de deux sets : l’un brute, l’autre après déglint. Enregistrement de ces données sous forme compressée.
5. Calibration des différents modèles grâce aux 2 fichiers jupyter distincts, sur la base des deux sets de données optiques et avec les données SHOM. Production des rasters de bathymétries estimées au format .tif, avec une seconde couche d’erreur interpolée par la méthode des plus proches voisins.
6. Création d’un nouveau fichier raster sur la base de la bathymétrie estimée par la données déglinted, après interpolation des pixels sans solution, suite à une sur correction du déglint.

***Résultats et observations***

Ce site d’application ne contient qu’une très pauvre quantité de données. Cependant l’ensemble des mesures semblent ne provenir que d’une seule campagne ce qui pourrait améliorer les résultats vis-à-vis des application avec des mesure a forte hétérogénéité temporelle. En effet, si la zone comporte une dynamique sédimentaire constante sur l’ensemble de son empreinte, les différences entre les profondeurs actuelles et celles mesurées lors de la campagne pourraient être systématiques. Ce qui ne nuirait pas au processus de calibration, mais induirait un biais systématique, probablement faible mais impossible à détecter, sur les résultats.

Du fait de la faible densité de mesures, seul les pixels ne comportant qu’une seul mesure située aux extrémités de son empreinte, sont écartés. A noter également l’absence de toute mesure en profondeur intermédiaire d’environ 16 à 26 mètres rendant un peu plus incertaine la calibration de modèle performant dans cette gamme.

Malgré la présence de pixel « sur-corrigés », les modèles calibrés après « déglint » sont plus performant.

Dans les deux cas, les meilleurs modèles sont obtenus après pénalisation et calibration en 2 itérations.

Coefficient de déterminations :

|  |  |  |
| --- | --- | --- |
| Modèle | Deglint | Raw |
| MLS | 0.916 | 0.937 |
| 2MLS | 0.970 | 0.979 |
| RMLS | 0.917 | 0.935 |
| 2RMLS | 0.973 | 0.982 |

***Application : Grand Noumea***

***Arborescence de dossiers***

***GN***/ >

***Scripts***/ >

N’est plus utilisé ici.

***Images***/ >

Regroupe les images non-traité sous dossier .SAFE/

***Acolite***/ >

***Nom\_Image\_Sentinel***/>

Image sentinel après correction atmospheric

***Shapes***/ >

Regroupe l’ensemble des shapefile d’empreintes

***FootPrint\_(n°)***/ >

***Résultats***/ >

***Bathymetries***/ >

***Nom\_Image\_Sentinel***/>

Regroupe les fichiers raster de bathy estimé avec l’image correspondant au dossier

***Classifications***/ >

Regroupe les fichiers de classif de types de fonds, anciennement utilisés.

***CompressedData***/ >

Regroupe les descripteurs des différents modèles sous-forme de donnée compressée

***Scripts***/ >

***Modeling***/ >

GetBathy\_Stumpf\_RawData.ipynb >

Fichier jupyter de traitement des données Shom intra-pixel, de calibration des modèles Stumpf et de production d’un raster de bathymétrie estimées.

***OpticalDataManagement***/ >

Netcdf2Tiff.ipnyb >

Fichier jupyter de convertion NetCDF to Tiff (Conversion imprécise…)

ConvertCetcdf2Tif.py >

Script python de convertion NetCDF to Tiff (Conversion imprécise…)

Copyjp200MT2NcIntiff.py >

Script python de d’ extraction de la geotransform des métadonnées d’un raster de même empreinte et de même résolution au format jpeg2000 pour la conversions d’un NetCDF vers un Tiff (Conversion utilisé…)

TileProcessing.ipnyb >

Génère les différentes empreintes de l’image (sol, ombre de nuages…) par seuillage sur les tuiles de l’image. Dans le Cas de Petrie, Génère une suite de raster supplémentaire après deglint.

GetX.ipynb >

Fichier jupyter de génération des descripteurs des modèles Lyzenga.

GetL\_Ratios\_Raw.ipynb >

Fichier jupyter de génération des descripteurs des modèles Stumpf.

GetL\_DII.ipynb >

Fichier jupyter de génération des DII pour classification (plus utilisé).

GetL\_HomogeneityMap.ipynb >

Fichier jupyter de creation de carte d’homogeneité des fonds et des propriétés optiques (essai en test).

***BathyVisualisationAndProcessing***/ >

Regroupe des scripts de visualisation 3d. Pas utilisé car trop lourd à générer.

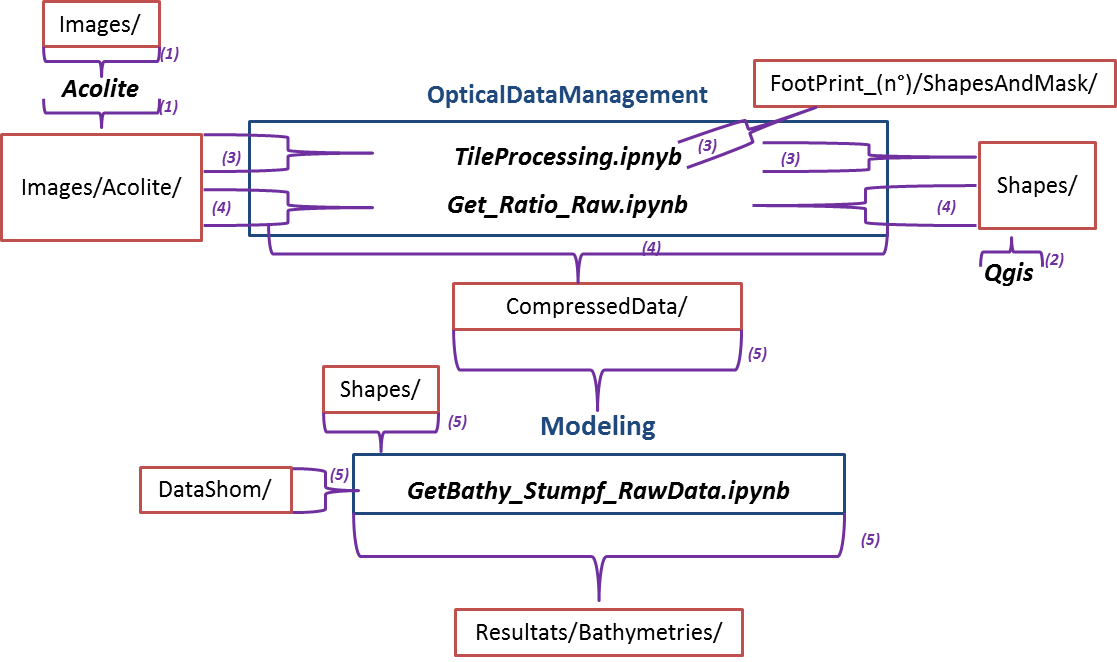
GetInterpolated.ipynb>

Produit un raster de bathymétrie estimée après l’interpolation des pixels sur corrigée par le processus du deglint.

***ShapesAndMask***/>

Regroupe un shape correspondant à l’empreinte de détecteur traité ainsi masque sous forme de raster boolean définissant la zone de travail.

***Procédure***

******

1. Correction atmosphérique des images par l’utilisation du logiciel Acolite (Nouvelle version : réalise elle-même la conversion en tif et rogne convenablement l’image)
2. Création des empreintes de l’océan profond, de la zone d’intérêt.
3. Création de l’empreinte de sol et de vagues par seuillage sur la bande proche-infrarouge sur l’ensemble de l’image (jugé suffisant). Création de l’empreinte définie par les pixels observés par un seul des detectecteurs.
4. Génération des données descripteurs grâce au fichier jupyter dédié à la donnée optique brute, sans déglint. Enregistrement de ces données sous forme compressée.
5. Calibration des différents modèles grâce au fichier jupyter dédié à la donnée optique brute, avec les données SHOM. Production des rasters de bathymétries estimées au format .tif, avec une seconde couche d’erreur interpolée par la méthode des plus proches voisins.

***Résultats et observations***

La projection des mesures de terrain dans l’espace formé par la variable profondeur et les ratios de signaux optiques montre un niveau très élevé de bruit. Cela induit une quantité considérable de mauvaise classification lors de la première itération des modèles de prédiction en deux étapes.

Les mesures, sont presque intégralement contenues dans la partie nord de l’empreinte. Un petit amoncellement de mesures, au sud, est disponible lorsque la limite temporelle est fixée à 1970. Il y a donc un compromis très contraignant à faire entre une bonne distribution spatiale et une homogénéité temporelle des mesures de terrain. Les résultats issus d’un set de validation avec une limite temporelle fixée à 2010 risque de voir l’apparition d’un biais progressif à mesure que l’on tend vers la partie sud de l’image. L’observation de ce résultat semble également montrer qu’ils est biaisé sur la partie nord de l’image, puisque les profondeurs des platiers de l’aquarève et de l’ilot maitre semblent surestimées. A l’inverse, les profondeurs de la barrière de corail, au sud, semble sous-estimées à leur endroit intérieur. La couche résultant de 2010 semble donc montrer plusieurs biais.

Les deux résultats sont issus de modèles itératifs avec seuil. Le seuil en question, qui devrait être fixé à 24.5 mètres, doit être recontrôlé car lors de la conception du script l’inscription automatisé des seuils dans l’intitulé du fichier geotiff résultant n’était pas encore mise en place. De plus, malgré la métrique du R2, plus favorable à un seuil si élevé, ce n’est peut-être pas la solution la plus judicieuse du fait d’un nombre insuffisant de valeur de calibration au sein de la gamme de profondeurs supérieures qu’il décrit. A l’inverse un seuil plus bas, autour des 5 mètres, pourrait avoir des résultats favorables sur la métrique d’écart moyen, plus sensible, malgré l’augmentation de mauvaise classification initiale qui serait induit. La forte quantité de mesure, inférieure à 5 mètres, y serait favorable.

Notons que les deux rasters de bathymétrie estimée se sont vus rogné sur la part sud de leur empreinte, suite à un bug, corrigé depuis. Les deux raster restent toutefois acceptable puisque la partie rogné concerne des profondeur or du lagon. Les résultats des deux rasters reste toutefois satisfaisant au regard des donnée de départ et de la comparaison avec la carte marine du SHOM.

***Application : Voh-Koné-Pouembout***

***Arborescence de dossiers***

***VKP***/ >

***Scripts***/ >

N’est plus utilisé ici.

***Images***/ >

Regroupe les images non-traité sous dossier .SAFE/

***Acolite***/ >

***Nom\_Image\_Sentinel***/>

Image sentinel après correction atmospheric

***Shapes***/ >

Regroupe l’ensemble des shapefile d’empreintes

***FootPrint\_(n°)***/ >

***Résultats***/ >

***Bathymetries***/ >

***Nom\_Image\_Sentinel***/>

Regroupe les fichiers raster de bathy estimé avec l’image correspondant au dossier

***Classifications***/ >

Regroupe les fichiers de classif de types de fonds, anciennement utilisés.

***CompressedData***/ >

Regroupe les descripteurs des différents modèles sous-forme de donnée compressée

***Scripts***/ >

***Modeling***/ >

GetBathy\_Stumpf\_RawData.ipynb >

Fichier jupyter de traitement des données Shom intra-pixel, de calibration des modèles Stumpf et de production d’un raster de bathymétrie estimées.

***OpticalDataManagement***/ >

Netcdf2Tiff.ipnyb >

Fichier jupyter de convertion NetCDF to Tiff (Conversion imprécise…)

ConvertCetcdf2Tif.py >

Script python de convertion NetCDF to Tiff (Conversion imprécise…)

Copyjp200MT2NcIntiff.py >

Script python de d’ extraction de la geotransform des métadonnées d’un raster de même empreinte et de même résolution au format jpeg2000 pour la conversions d’un NetCDF vers un Tiff (Conversion utilisé…)

TileProcessing.ipnyb >

Génère les différentes empreintes de l’image (sol, ombre de nuages…) par seuillage sur les tuiles de l’image. Dans le Cas de Petrie, Génère une suite de raster supplémentaire après deglint.

GetX.ipynb >

Fichier jupyter de génération des descripteurs des modèles Lyzenga.

GetL\_Ratios\_Raw.ipynb >

Fichier jupyter de génération des descripteurs des modèles Stumpf.

GetL\_DII.ipynb >

Fichier jupyter de génération des DII pour classification (plus utilisé).

GetL\_HomogeneityMap.ipynb >

Fichier jupyter de creation de carte d’homogeneité des fonds et des propriétés optiques (essai en test).

***BathyVisualisationAndProcessing***/ >

Regroupe des scripts de visualisation 3d. Pas utilisé car trop lourd à générer.

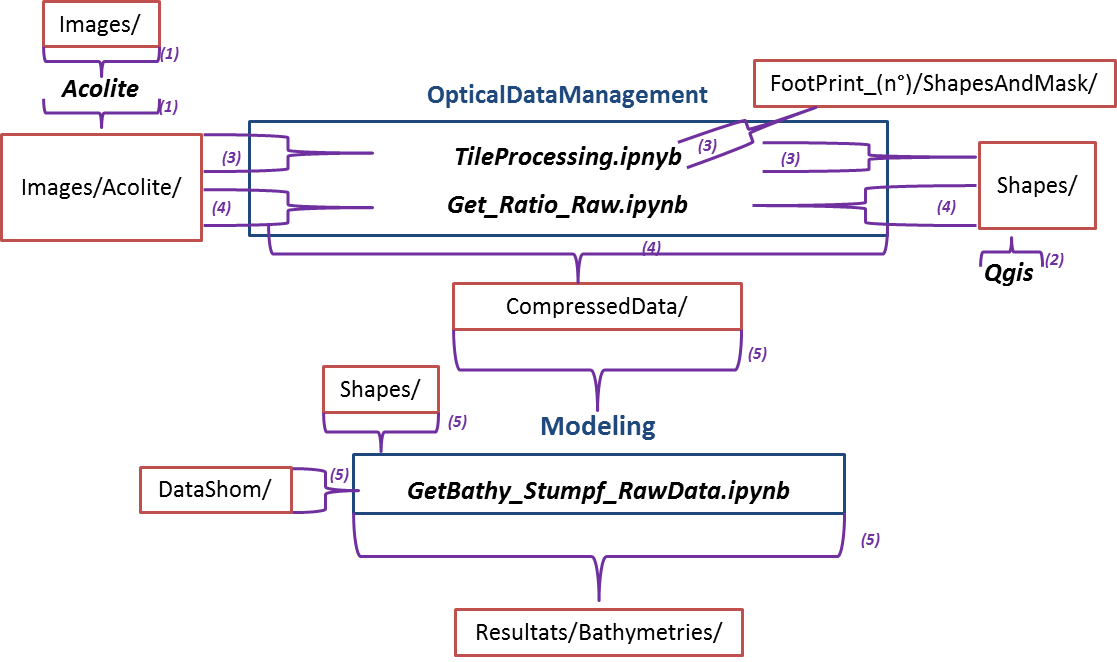
GetInterpolated.ipynb>

Produit un raster de bathymétrie estimée après l’interpolation des pixels sur corrigée par le processus du deglint.

***ShapesAndMask***/>

Regroupe un shape correspondant à l’empreinte de détecteur traité ainsi masque sous forme de raster boolean définissant la zone de travail.

***Procédure***

******

1. Correction atmosphérique des images par l’utilisation du logiciel Acolite (Nouvelle version : réalise elle-même la conversion en tif et rogne convenablement l’image)
2. Création des empreintes de l’océan profond, de la zone d’intérêt.
3. Création de l’empreinte de sol et de vagues par seuillage sur la bande proche-infrarouge sur l’ensemble de l’image (jugé suffisant). Création de l’empreinte définie par les pixels observés par un seul des detectecteurs.
4. Génération des données descripteurs grâce au fichier jupyter dédié à la donnée optique brute, sans déglint. Enregistrement de ces données sous forme compressée.
5. Calibration des différents modèles grâce au fichier jupyter dédié à la donnée optique brute, avec les données SHOM. Production des rasters de bathymétries estimées au format .tif, avec une seconde couche d’erreur interpolée par la méthode des plus proches voisins.