

UNIVERSITÉ DE STRASBOURG

RAPPORT INTERMÉDIAIRE DE STAGE MASTER

Développement d'un outil d'analyse automatique de longues séries temporelles de mouvements du sol et de détection d'anomalies : application à des produits issues de données satellitaires Sentinel-1 et Sentinel 2

ETUDIANT :
PAMBOU MOUBOGHA EDDY
ANNÉE SCOLAIRE 2020-2021

ENCADRANT :
JEAN-PHILIPPE MALET

Table des matières

1	Introduction	3
2	Présentation de l'organisme d'accueil	4
3	Contexte du stage	5
3.1	Débruitage des séries temporelles	5
3.2	Détection des glissements de terrain	5
4	Zone d'étude et jeux de données	7
4.1	Zone d'étude	7
4.2	Acquisition des images optiques	8
4.3	Exploitation des images optiques	8
4.3.1	Corrélation d'images	8
4.3.2	MPIC-OPT	9
4.4	Données	9
5	Débruitage des séries temporelles	10
5.1	Décomposition en composantes indépendantes	10
5.1.1	Problème et définitions	10
5.1.2	Algorithme	11
5.1.3	Comparaison avec l'analyse en composantes principales	11
5.2	Détermination du nombre optimal de sources	12
5.3	Sélection des sources	13
6	Détection des glissements	14
6.1	Identification des profils de déplacement	14
6.2	Filtrage des données	15
6.2.1	Filtrage des vitesses	15
6.2.2	Test de significativité des vitesses moyennes	15
6.2.3	Elimination des séries chaotiques	16
6.2.4	Filtrage morphologique	16
6.3	Clustering	16

6.3.1	Clustering par densité	17
6.3.2	DBSCAN	17
6.3.3	HDBSCAN	18
7	Résultats et Discussion	20
7.1	Débruitage	20
7.2	Filtrage et clustering	20
7.3	Discussion	22
7.3.1	Débruitage	22
7.3.2	Détection	23
	Bibliographie	24

Chapitre 1

Introduction

Les glissements de terrain apparaissent lorsqu'une masse de terre descend sur un plan de glissement, provoqués par les activités anthropiques ou des phénomènes climatiques, géologiques ou géomorphologiques. Ces déplacements peuvent être lents (quelques millimètres par an) ou rapides (quelques centaines de mètres par jour).

Ces mouvements peuvent être à l'origine de catastrophes naturelles causant des pertes humaines et des dommages importants sur les infrastructures. Au plan mondial, les mouvements de terrain causent chaque année la mort de 800 à 1000 personnes. En France, ce risque concerne environ 7000 communes et présente, pour un tiers d'entre elles, un niveau de gravité fort.

Pour prévenir ces catastrophes, il faut surveiller les mouvements du sol. Cette surveillance est de plus en plus réalisée en utilisant les techniques de télédétection. En observation de la Terre, on distingue deux techniques de télédétection : l'imagerie radar et optique. L'imagerie radar mesure la projection 3D de la déformation dans la ligne de visée du satellite. Elle possède une résolution millimétrique et est adaptée pour mesurer les déplacements lents. Pour mesurer les déplacements rapides, l'imagerie optique est plus utilisée. Cette technique possède une résolution métrique et est sensible aux composantes Nord-Sud et Est-Ouest de la déformation.

on peut exploiter : a) des ondes émises émises par le soleil puis réfléchies par la Terre et enregistrées par un capteur embarqué dans un satellite b) des ondes émises par un émetteur artificiel placé sur le satellite puis réfléchies par la surface de la Terre et enregistrées par un capteur placé sur ce même satellite. Dans le premier cas, on parle de télédétection passive et les données acquises sont des images optiques, dans le second cas de télédétection active et d'images radar. La première méthode est sensible aux composantes Nord-Sud et Est-Ouest de la déformation et est plus adaptée pour la détection des glissements rapides. La deuxième méthode ne mesure que le projeté du déplacement 3D dans la ligne de visée du satellite et est adapté pour l'étude des glissements lents.

Il existe plusieurs algorithmes pour exploiter les images satellitaires. Ils s'appuient tous sur la corrélation d'images. Parmi eux, on trouve l'algorithme MPIC-OPT, récemment développé par les chercheurs de l'équipe Déformation Active de l'EOST. Les séries temporelles de déplacement calculés par MPIC-OPT, encore très peu exploités, offrent la possibilité de développer de nouveaux outils pour la détection des glissements de terrain.

Afin de valoriser ces données, ce stage propose d'appliquer la décomposition en composantes indépendantes (ICA) pour débruiter les signaux dérivés par MPIC-OPT et de mettre en place un workflow de détection des glissements de terrain en utilisant les statistiques et le clustering. Les différentes approches mises au point seront testées sur le glissement de La Valette.

Chapitre 2

Présentation de l'organisme d'accueil

L'Ecole et Observatoire des Sciences de la Terre (EOST) est une école d'ingénieurs en géophysique créé en 1935 par Edmond Rothé. Comme son nom l'indique l'EOST est également un observatoire. L'objectif premier de l'UMS830 est de faciliter l'observation pérenne des phénomènes naturels et de rendre accessible les données recueillies à la communauté scientifique. Ses tâches d'observation entrent dans le cadre des Services Nationaux d'Observation (SNO) labellisés par l'Institut National des Sciences de l'Univers du CNRS. Au total, l'EOST est impliqué dans dix services d'observation dont il est pilote au plan national ou partenaire actif. Ils concernent le domaine "Terre solide" et le domaine "Surfaces et interfaces continentales".

Ce stage s'effectue au sein du pôle Recherche dans l'équipe Déformation Active. Les travaux de l'équipe portent sur la déformation lithosphérique, le fonctionnement des failles sismiques et la déformation de sub-surface. Elle est fédérée autour de plusieurs disciplines, telles que la géodésie, la tectonique active, la géomorphologie et la paléosismologie. Elle appuie fortement sa recherche sur les données acquises par les observatoires de l'EOST gérés par ses membres et sur des chantiers régionaux, principalement situés en Méditerranée et en Afrique.

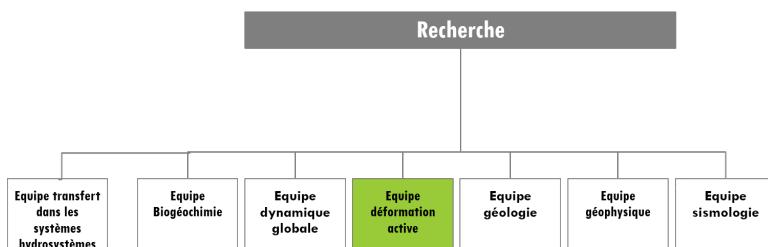


FIGURE 2.1 – Organigramme simplifié du pôle Recherche de l'EOST.

Chapitre 3

Contexte du stage

3.1. Débruitage des séries temporelles

Les séries temporelles dérivées par MPIC-OPT sont très bruitées et donc difficiles à interpréter. Une méthode statistique appelée **décomposition en composantes indépendantes** (ICA) a été appliquée à ces signaux afin d'éliminer le bruit. La méthode consiste à décomposer un ensemble de signaux mélangés par une transformation linéaire en un ensemble de signaux purs. Pour appliquer ICA, il est nécessaire de connaître le nombres de sources (paramètre) et de disposer d'une heuristique pour sélectionner les sources d'intérêt si on l'utilise comme outil de débruitage.

La méthode a été testée sur les glissements de La Valette et d'Aiguilles. Le nombre de sources a été fixé aléatoirement et les sources pouvant correspondre à du signal gravitaire ont été sélectionnées manuellement en se basant sur les connaissances géologiques des sites étudiés. Les premières observations montrent que ICA est plus adapté aux déformations linéaires (site de La Valette). En présence de variations de vitesses importantes (site d'Aiguilles), la méthode donne de moins bons résultats. Dans les deux cas, les séries temporelles débruitées varient en fonction du nombre de sources et la sélection des sources gravitaires nécessite l'intervention d'un expert.

L'objectif de cette partie est donc de proposer une méthode pour déterminer le nombre de sources optimal et d'automatiser la sélection des sources gravitaires.

3.2. Détection des glissements de terrain

Un glissement de terrain correspond au déplacement d'une masse de terrains meubles ou rocheux le long d'une surface de rupture (plane, circulaire ou quelconque). L'observation des glissements a deux finalités : la compréhension du phénomène et la gestion du risque (dimensionnement, surveillance).

On trouve plusieurs approches dans la littérature sur la détection de glissements de terrain. Parmi ces approches, on peut citer les méthodes se basant sur le filtrage des vitesses. Ce type de méthode propose de rechercher les glissements dans les zones ayant des taux de déplacement relativement importants [1]. Cependant, elle suppose que les déplacements sont linéaires au cours du temps. En réalité, la vitesse d'un glissement varie en fonction des saisons et peut être affectée par certains événements géologiques. En conséquence, l'hypothèse des déplacements linéaires est souvent peu vérifiée mais elle reste une base de travail importante [7]. L'autre inconvénient est que la vitesse n'est pas un paramètre suffisant pour étudier et comprendre la dynamique d'un glissement de terrain.

Pour essayer de dépasser cette limite, certaines méthodes utilisent directement les séries temporelles de déplacement. Par exemple, les auteurs (quel article ?) proposent une classification des séries temporelles de déplacement pour cartographier les glissements de terrain dans l'Appennin du Nord en Italie [7]. Leur méthode consiste à rechercher des régimes de vitesses différents dans les séries temporelles en effectuant une série de tests statistiques. Cependant, les tests statistiques mis en oeuvre dépendent

des profils de déplacement identifiés manuellement dans les données. Quoique difficile à généraliser, leur approche permet d'associer les régimes de vitesses identifiés à des évènements géologiques. Ce qui facilite l'interprétation des clusters trouvés.

Chapitre 4

Zone d'étude et jeux de données

4.1. Zone d'étude

Notre zone d'étude est la vallée de l'Ubaye qui est caractérisée par un haut de versants façonné dans les formations très résistantes avec des altitudes allant de 1900 à 2100 m et des pentes supérieures à 45°. Plusieurs glissements ont été recensés dans cette zone. L'un des glissements les plus grands est celui de La Valette situé sur la rive droite de l'Ubaye, dans le bassin du torrent de La Valette. Il s'est activé en mars 1982. Il fait une longueur d'environ 2000 m et sa zone de départ est située vers 2000 m d'altitude. C'est le 3e plus grand glissement répertorié au niveau européen. Le glissement n'est pas homogène dans l'ensemble. En amont, on distingue des ravinements sur le bord des niches tandis qu'à l'aval le glissement se termine par une coulée de boue.



FIGURE 4.1 – Zone d'étude. Le polygone en blanc délimite la zone d'étude. Les polygones en noir représentent les glissements repérés.



FIGURE 4.2 – Glissement de La valette.

4.2. Acquisition des images optiques

L'imagerie optique est une technique de télédétection qui mesure les ondes électromagnétiques émises par le soleil et réfléchies par la surface terrestre. Les ondes sont enregistrées par un capteur optique installé dans un satellite.

En présence de conditions météorologiques dégradées (brouillard, pluie, nuages) ou lorsque la luminosité est trop faible (p. ex., la nuit), le satellite ne peut réaliser aucune mesure. En conséquence, la quantité d'images exploitables est diminuée et **les données finales ne sont pas régulièrement échantillonnées**.

Parmi les satellites optiques les plus utilisés, on peut citer **Sentinel 2** (résolution 10 m), LandSat (résolution 15 m) et SPOT 5 (résolution 2.5 m).

4.3. Exploitation des images optiques

Il existe plusieurs algorithmes pour exploiter les images optiques. Nous nous limiterons à présenter le principe de la corrélation d'images et l'algorithme MPIC-OPT.

4.3.1 Corrélation d'images

La corrélation d'images est une méthode qui permet de mesurer les déplacements en deux dimensions entre deux images en se basant sur l'intensité des pixels. Il existe plusieurs algorithmes de corrélation d'images. Cependant, la résolution des images de base est le facteur le plus déterminant dans la qualité des résultats.

La présence de végétation ou la modification de l'apparence du glissement peut créer des problèmes de décorrélation entre deux images.

4.3.2 MPIC-OPT

MPIC-OPT (Mutiple Pairwise Image Correlation of OPTical image Time-series) est un algorithme développé par les chercheurs de l'équipe Déformation Active de l'EOST. Il propose de comparer plusieurs paires d'images et d'utiliser la redondance des données pour réduire le bruit. Il est composé de trois modules principaux : a) le module corrélation, b) le module correction c) le module d'analyse spatio-temporelle.

Le **module correlation** réalise la catégorisation des pixels (suppression des pixels situés sous les zones nuageuses) ; il définit la taille de la fenêtre glissante pour comparer deux images et il fournit une première grille des déplacements bruts dans les directions Nord-Sud et Est-Ouest.

Le **module correction** est chargé d'appliquer diverses corrections sur les déplacements bruts calculés par le module corrélation (p. ex., ortho-rectification, filtrage des régions plates).

Le **module d'analyse spatio-temporelle** permet de calculer les vitesses moyennes d'un intervalle donné. Il permet également de calculer la cohérence des vecteurs déplacements de chaque pixel. Enfin, il permet d'inverser les séries temporelles de chaque pixel sur les deux composantes.

4.4. Données

Notre jeu de données est composé des champs de déplacement de 87016 pixels calculés par l'algorithme MPIC à partir des images prises par le satellite copernicus 2. Chaque pixel est localisé par ses coordonnées géographiques (latitude et longitude) et comporte les données suivantes : les séries temporelles des déplacements dans les directions Nord-Sud et Est-Ouest, les vitesses moyennes associées (calculées par regression linéaire) et la cohérence qui est une grandeur comprise entre 0 et 1 et qui permet d'estimer la constance de la direction du vecteur déplacement d'un pixel au cours du temps. Les séries temporelles sont composées de 87 observations prises entre le 27 décembre 2015 et le 06 septembre 2020. L'acquisition d'images optiques étant difficile lorsque les conditions météorologiques sont dégradées, les séries temporelles ne sont pas régulièrement échantillonnées. Dans la direction Nord-Sud, les taux de déplacement varient entre -0.0094 et 0.0133 m/jour ; dans la direction Est-Ouest, entre 0.0115 et 0.0138 m/jour.

Chapitre 5

Débruitage des séries temporelles

Les glissements de terrain résultent de la conjugaison de plusieurs facteurs (géologiques, anthropiques, etc.). De plus, la dérivation des séries temporelles de déplacement comporte des étapes entachées d'incertitudes (p. ex., orthorectification, les limites de la corrélation d'images). Toutes ces erreurs se combinent aux signaux que l'on essaie de détecter. Dans cette partie, le débruitage des données sera posé comme un problème de séparation de sources et sera résolu à l'aide d'une méthode statistique appelée **décomposition en composantes indépendantes** (ICA).

5.1. Décomposition en composantes indépendantes

5.1.1 Problème et définitions

ICA est souvent présenté en utilisant le *cocktail party problem*. Lors d'une telle soirée, on dispose P microphones dans une salle dense, où N personnes discutent par groupes de tailles diverses. Chaque microphone enregistre la superposition des discours des personnes à ses alentours et le problème consiste à retrouver la voix de chaque personne. Mathématiquement, ce problème est exprimé par

$$X = AS,$$

où \mathbf{X} est la matrice $(p \times p)$ représentant les signaux mélangés (les observations), \mathbf{A} est la matrice de mélange $(p \times p)$, \mathbf{S} est la matrice des signaux sources.

Connaissant les signaux x_1, x_2, \dots, x_N de X , le but de ICA est d'estimer l'inverse de la matrice de mélange A . Pour estimer cette matrice, ICA se base sur les hypothèses suivantes : l'indépendance statistique et la non-gaussianité des sources.

La première hypothèse stipule que les sources que l'on cherche à extraire doivent être statistiquement indépendantes. On dit que deux variables x et y sont statistiquement indépendantes si x n'apporte aucune information sur y et vice-versa. **L'indépendance statistique ne doit pas être confondue avec la décorrélation.** Deux variables sont corrélées si leur covariance est nulle. La décorrélation est une version faible de l'indépendance statistique : deux variables statistiquement indépendantes sont nécessairement décorrélées mais l'inverse n'est pas toujours vrai.

La seconde hypothèse découle de considérations mathématiques. La non-gaussianité permet de mesurer de combien une distribution s'éloigne d'une distribution gaussienne. On peut la mesurer en utilisant l'entropie négative ou le kurtosis (coefficient d'aplatissement).

5.1.2 Algorithme

ICA construit la matrice \mathbf{W} (inverse de la matrice A) en estimant itérativement chacune des ses lignes \mathbf{w}^T . Le vecteur \mathbf{w}^T est construit de manière à maximiser la non-gaussianité du signal $\mathbf{w}^T \mathbf{X}$ sous la contrainte $\|\mathbf{w}\| = 1$. L'algorithme est le suivant :

Algorithm 1: ICA

Input: Données X , nombre de sources N , nombre d'itérations max_iter , tolérance tol

Output: Sources dé-mixées S

- 1 Centrer les données $X = X - E\{X\}$
 - 2 Blanchir les données $z = V X$
 - 3 Choisir une fonction non-linéaire g
 - 4 Initialiser aléatoirement w avec $\|w\| = 1$
 - 5 Mettre à jour $w \leftarrow E\{zg(w^T z)\} - E\{g'(w^T z)\}$
 - 6 Normaliser $w \leftarrow \frac{w}{\|w\|}$
 - 7 Répéter les étapes (5) et (6) max_iter fois tant que le critère de convergence est non satisfait
 - 8 Répéter l'étape (7) N fois
 - 9 **return** $S = WX$
-

Les étapes (1) et (2) sont des étapes de pré-processing. L'étape (2) transforme, par une opération linéaire, le vecteur centré résultant de l'étape (1) en un vecteur dont les composantes sont décorrélées deux à deux et de variance unité. L'initialisation aléatoire de w à l'étape (4) est à l'origine de l'absence d'ordre des composantes indépendantes. En effet, pour des runs différents, l'initialisation change et donc **l'ordre dans lequel les composantes sont calculées varie**. A l'étape (7), la convergence est atteinte lorsque w est orthogonal. Il est important de noter que **la solution trouvée à l'étape (7) est un minimum local**. Par conséquent, les sources extraites dépendent aussi de l'initialisation et ne sont donc pas nécessairement significatives.

5.1.3 Comparaison avec l'analyse en composantes principales

Le problème de séparation de sources peut aussi être résolu par l'ACP mais les sources extraites par les deux méthodes sont différentes. Ceci s'explique par les différences entre les deux approches. L'ACP cherche à trouver les axes qui expliquent le mieux la dispersion des données en se basant sur la variance expliquée par chaque axe. ICA cherche à séparer les données en maximisant la non-gaussianité de chaque source. L'ACP présente l'avantage de pouvoir trier les sources par ordre d'importance : les axes les plus importants sont ceux qui expliquent le plus la variance des données. Contrairement à l'ACP, les composantes de ICA ne sont pas ordonnées.

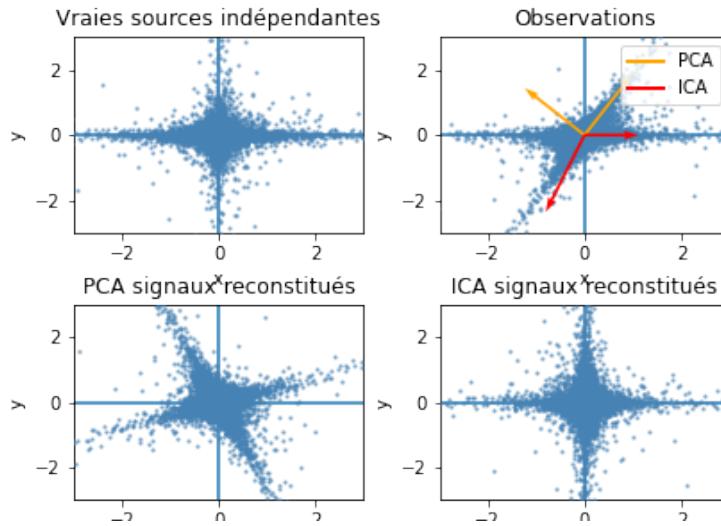


FIGURE 5.1 – Simulation d'un problème de séparation de sources avec ICA et ACP. Les vraies sources sont deux processus extrêmement non-gaussiens. Les sources reconstituées par ICA sont plus proches des vraies sources. (source : scikit-learn.org)

5.2. Détermination du nombre optimal de sources

La détermination du nombre de sources est une étape cruciale avant d'utiliser ICA. En effet, extraire très peu de sources peut conduire à des signaux non purs, alors que calculer trop de sources peut sur-décomposer les sources intéressantes et introduire du bruit.

On trouve plusieurs approches dans la littérature pour essayer de résoudre ce problème. En faisant varier le nombre de sources, on observe une convergence des signaux reconstitués. On peut alors choisir un nombre de sources qui permet d'atteindre la convergence suivant un critère donné. Dans notre cas, cette méthode est coûteuse en temps et en mémoire parce que nous travaillons avec un grand nombre de séries temporelles. Si on dispose d'un modèle de prédiction, on peut alors définir le nombre optimal de sources comme celui pour lequel le modèle donne les meilleurs résultats.

L'approche que nous allons utiliser s'inspire de la méthode de sélection du nombre de sources implémentée logiciel Icasso [9] et d'une variante utilisée dans (...). Elle repose sur le fait que le nombre optimal de sources permet d'avoir des sources qui varient très peu avec l'initialisation. Autrement dit, si on fixe un nombre de sources et on applique ICA plusieurs fois sur le même jeu de données mais avec des initialisations différentes, on peut regrouper les sources qui se ressemblent à l'aide d'un algorithme de clustering. Chaque cluster trouvé représente alors une source dont la significativité peut-être estimée par la compacité et l'isolation du cluster. Les premières étapes de Icasso sont décrites dans l'algorithme ci-dessous (les étapes concernant la visualisation ont été ignorées).

Algorithm 2: Estimation du nombre de sources

Input: Données D , nombre de sources minimum M_{min} , nombre de sources maximum M_{max} , nombre d'itérations K

Output: Nombre de sources optimal

- 1 Appliquer K fois ICA sur D avec M sources et des **initialisations différentes**
 - 2 Clusteriser les $M \times K$ composantes indépendantes calculées en M clusters
 - 3 Pour chaque cluster C_k trouvé, calculer sa stabilité $I_q(C_k)$
 - 4 Calculer la stabilité moyenne du clustering $S(M)$
 - 5 Répéter les étapes précédentes pour chaque M compris entre M_{min} et M_{max}
-

A l'étape (2), l'algorithme de clustering est laissé au choix de l'utilisateur (Icasso utilise le clustering hiérarchique). Les sources sont comparées en utilisant la mesure de dissimilarité suivante : $1 - |r_{ij}|$ où r_{ij} représente le coefficient de corrélation de Pearson ou Spearmann entre deux sources i et j . I_q et $S(M)$ sont calculés de la manière suivante :

$$I_q(C_k) = \frac{1}{|C_k|^2} \sum_{i,j \in C_k} |r_{ij}| - \frac{1}{|C_k| \sum_{l \neq k} |C_l|} \sum_{i \in C_k} \sum_{j \neq C_k} |r_{ij}|$$

$$S(M) = \frac{1}{M} \sum_k I_q(C_k)$$

Le premier terme de I_q correspond à la similarité intra-cluster (compacité) et le second à la similarité inter-cluster (isolation). Si toutes les sources du cluster C_k sont parfaitement corrélées entre elles ($|r_{ij}| = 1$) et complètement décorrélées des autres sources ($|r_{ij}| = 0$) alors le clustering est parfait et $I_q = 1$. $S(M)$ est la moyenne des stabilités pour un clustering donné.

5.3. Sélection des sources

Chapitre 6

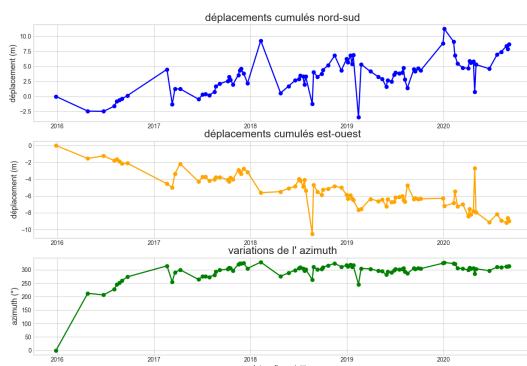
Détection des glissements

Notre approche pour détecter les glissements de terrain comporte deux grandes étapes : le filtrage et le clustering des données. Le filtrage permettra de réduire considérablement le bruit dans les données et sera réalisé à l'aide d'outils statistiques. Le clustering permettra de former les régions susceptibles d'être des glissements de terrain et sera mis en oeuvre en utilisant le partitionnement par densité.

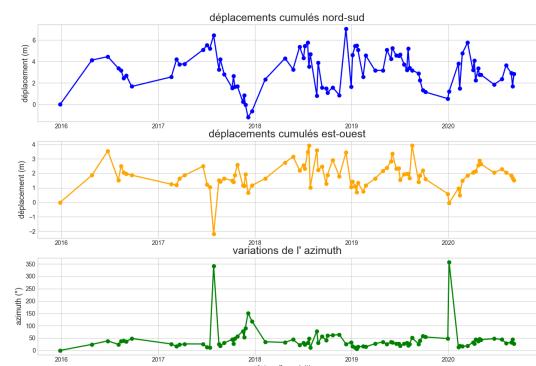
6.1. Identification des profils de déplacement

Les mouvements du sol dépendent de plusieurs paramètres et peuvent varier d'une zone à une autre. De manière générale, un profil de déplacement présentant une tendance linéaire (eventuellement à laquelle peuvent s'ajouter des variations saisonnières) a de fortes chances d'être un glissement. Comme nos séries temporelles sont géoréférencées, il est possible de les inspecter manuellement. On peut afficher chaque pixel dans Google Earth. Grâce à cette inspection, nous avons pu identifié les profils de déplacement suivants : a) les déplacements linéaires au cours du temps avec des variations de vitesse ; b) les déplacements montrant un caractère périodique ; c) les déplacements erratiques ; d) les déplacements constants.

La majorité des déplacements linéaires sont situés dans le glissement de La Valette. On en trouve aussi dans certaines zones couvertes par la végétation. Les déplacements avec de grandes fluctuations peuvent correspondre à des mouvements gravitaires ; on les observe principalement sur des pentes instables (mouvement d'éboulis ou de blocs de roches). Les déplacement erratiques semblent correspondre à des pixels immobiles.

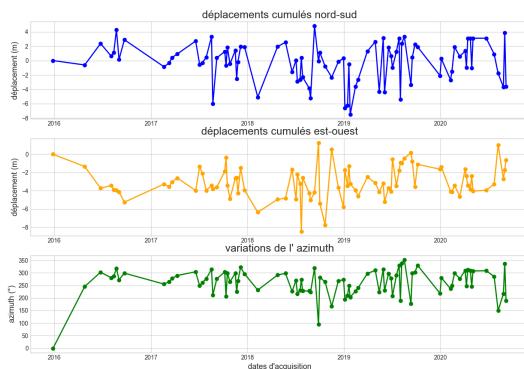


(a) type linéaire.

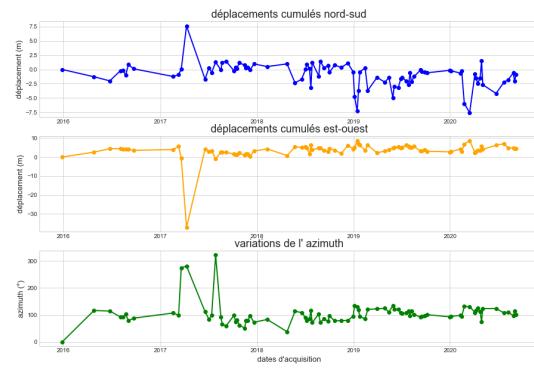


(b) type periodique.

FIGURE 6.1 – Les différents types de profil de déplacement observés dans les données.



(a) type aléatoire.



(b) type constant.

FIGURE 6.2 – Les différents types de profil de déplacement observés dans les données (suite).

6.2. Filtrage des données

6.2.1 Filtrage des vitesses

Le filtrage des vitesses consiste à supprimer les pixels ayant des taux de déplacement faibles. En effet, de tels pixels ont peu de chance d'appartenir à un glissement. Si on note σ l'écart-type des vitesses moyennes, n est un entier naturel et v la vitesse moyenne d'un pixel, le filtrage s'écrit simplement :

$$v > n \times \sigma$$

Les vitesses utilisées pour calculer σ sont celles dérivées par MPIC-OPT. Si on applique des transformations sur les séries temporelles, il faut recalculer les vitesses moyennes dans les deux directions.

6.2.2 Test de significativité des vitesses moyennes

Les vitesses moyennes étant estimées par régression linéaire, il apparaît important de s'interroger sur leur significativité statistique. On réalise le Test de Fischer pour évaluer la significativité des coefficients d'une régression linéaire. Dans le cas d'une régression linéaire simple, seule la pente β est concernée. Comme tout test statistique, le test de Fischer comporte une hypothèse nulle H_0 et une hypothèse alternative H_1 :

$$H_0 : \beta = 0$$

$$H_1 : \beta \neq 0$$

L'hypothèse H_0 est rejetée si la p-valeur associée à la statistique de Fischer est inférieure à un seuil α prédéterminé, $\alpha = 5\%$ dans la plupart des cas. Dans notre cas, le rejet de H_0 signifie que la vitesse moyenne calculée n'est pas significative. Autrement dit, la série temporelle des déplacements ne suit pas une évolution linéaire significative. L'hypothèse alternative H_1 est acceptée si l'hypothèse nulle H_0 est rejetée.

On note p_{ns} et p_{ew} les p-valeurs respectives des régressions linéaires des composantes Nord-Sud et Est-Ouest d'un pixel. On considère qu'un pixel se déplace sous l'effet de la gravité si au moins une des composantes de son déplacement a une vitesse significative. On peut énoncer la condition de filtrage de cette manière :

$$(p_{ns} < \alpha) \vee (p_{ew} < \alpha)$$

6.2.3 Elimination des séries chaotiques

Les pixels ayant une comportement chaotique peuvent passer le filtre précédent. Ceci s'explique en partie par le fait que la régression linéaire est sensible aux outliers. Pour améliorer le filtrage de ce type de pixels, on peut supposer que leurs séries temporelles auront des distributions plus aplatis à cause des oscillations autour de 0. L'aplatissement d'une distribution est quantifiée par une grandeur statistique appelée le kurtosis ou coefficient d'aplatissement. Si on considère une variable aléatoire réelle X d'espérance μ et d'écart-type σ , son kurtosis est défini par :

$$\beta_2 = \mathbb{E} \left[\left(\frac{X - \mu}{\sigma} \right)^4 \right]$$

Si les coefficients d'aplatissement $\beta_2 ns$, $\beta_2 ew$ respectifs des composantes ns et ew sont simultanément négatifs, il s'agit probablement d'un pixel chaotique. Nous considérons qu'un pixel est chaotique si au moins une des composantes de son déplacement est à un kurtosis négative :

$$(\beta_2 ns > 0) \vee (\beta_2 ew > 0)$$

6.2.4 Filtrage morphologique

Le filtrage morphologique consiste à utiliser la pente topographique et son orientation par rapport au Nord (azimut) pour filtrer les pixels. Ces deux valeurs sont calculés à partir d'un modèle numérique de terrain (MNT).

La pente topographique α représente l'inclinaison d'un objet par rapport à l'horizontal. Elle varie entre 0° et 90° . Comme les glissements sont par définition localisés dans les zones abruptes, la pente topographique permet de supprimer les pixels qui sont situés dans les régions plates relativement à une pente minimum fixée α_{min} . Le critère de sélection est donc :

$$\alpha > \alpha_{min}$$

L'azimuth γ_{mnt} est mesuré depuis le Nord en degrés. Sa valeur est comprise entre 0° et 360° . Une valeur de 0° signifie que le mouvement est orienté dans la direction Sud-Nord ; une valeur de 90° dans la direction Ouest-Est. Il permet de vérifier que la direction moyenne d'un pixel γ_c se déplaçant sous l'effet de la gravité est proche ou égale à γ_{mnt} . Cette direction est calculé à partir des composantes de son vecteur déplacement en utilisant la relation.

$$\tan(\gamma_c) = \frac{ew[i]}{ns[i]} = \frac{v_{ew}}{v_{ns}} \quad \forall i \in [0; n]$$

Etant donnée qu'une série temporelle fournie plus d'informations, nous avons préféré dériver la variation de l'azimuth au cours du temps au lieu d'une simple valeur. En fixant une tolérance δ et pourcentage seuil p_s , on obtient le critère suivant :

$$|\gamma_{mnt} - \gamma_c| < \delta$$

Il est peu probable que tous les échantillons de la série temporelle azimutale vérifient la condition précédente. Pour cette raison, nous avons donc décidé de considérer un pixel comme non bruité si le proportion de points vérifiant la condition ci-dessus est supérieure à un certain seuil p_s .

6.3. Clustering

Le clustering est une méthode statistique qui permet de diviser un ensemble de données en des sous-ensembles partageant des caractéristiques communes en utilisant une mesure de similarité ou une distance pour comparer les objets entre-eux.

Il existe plusieurs familles d’algorithmes de clustering parmi lesquels on peut citer : les algorithmes centroides, les algorithmes hiérarchiques et les algorithmes à densité. Dans cette partie, nous nous focaliserons sur DBSCAN (density-based spatial clustering of applications with noise) et HDBSCAN (hierarchical DBSCAN) qui sont deux algorithmes appartenant à la dernière famille.

6.3.1 Clustering par densité

Le clustering par densité repose sur l’idée suivante : deux points appartiennent au même cluster si on peut créer un chemin pour passer de l’un à l’autre de proche en proche. Cette approche débouche sur les notions d'**epsilon-voisinage** et de **connexion par densité**.

Si on considère ϵ un réel strictement positif, on appelle epsilon-voisinage d’un point x l’ensemble des points du jeu de données qui sont situés à une distance inférieure à x :

$$N_\epsilon(x) = \{u \in X | d(u, x) < \epsilon\}$$

Si on considère un entier naturel n_{min} , on dit que deux points x et y sont connectés par densité si l’on peut passer de l’un à l’autre par une suite d’ ϵ -voisinage contenant chacun au moins n_{min} points :

$$|N_\epsilon(x)| \geq n_{min}$$

Grâce aux notions définies précédemment, les algorithmes de clustering par densité possèdent des avantages incontournables : le nombre de clusters n’est pas prédéfini et les clusters trouvés peuvent être non-convexes.

6.3.2 DBSCAN

DBSCAN est un algorithme de clustering par densité. Il a été proposé en 1996. Ses champs d’applications sont divers : analyse cartographique, analyse de donnée, segmentation d’image, détection d’anomalies, etc.

DBSCAN itère sur les points du jeu de données. Pour chaque point, il construit l’ensemble des points connectés par densité à ce point : il calcule l’ ϵ -voisinage de ce point, puis si ce voisinage contient plus de n_{min} points, les ϵ -voisinages de chacun d’eux pour pouvoir agrandir le cluster. Si un point a moins de n_{min} voisins, alors il est considéré comme du bruit.

Le choix des paramètres ϵ et n_{min} n’est pas évident. Pour que l’algorithme trouve les clusters pertinents, il faut les choisir de manière à avoir assez de points intérieurs. Ce qui revient à choisir n_{min} assez petit ou ϵ très grand. Cependant le fait de fixer ϵ ne permet pas à l’algorithme de trouver des clusters de densités différentes.

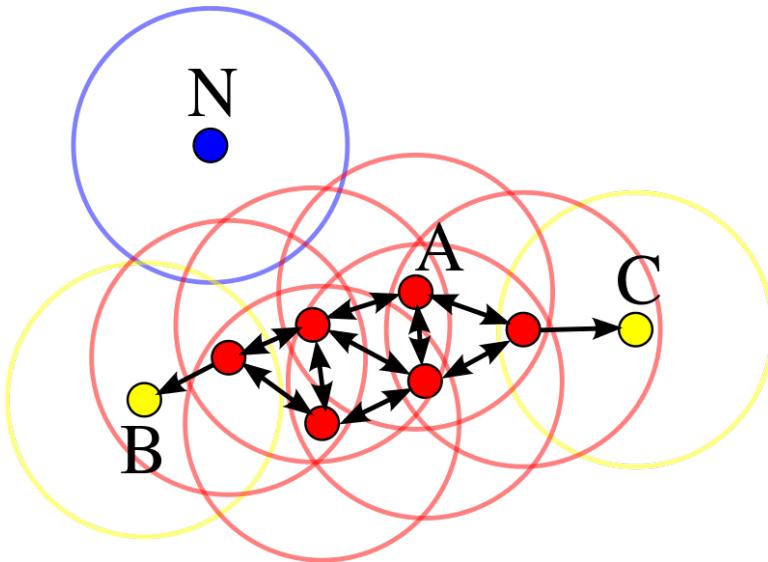


FIGURE 6.3 – Les points A sont les points déjà dans le cluster. Les points B et C sont atteignables depuis A et appartiennent donc au même cluster. Le point N est une donnée aberrante puisque son epsilon voisinage ne contient pas au moins n_{min} points. Source : wikipedia

Algorithm 3: DBSCAN

Input: epsilon-voisinage ϵ , nombre de points minimum n_{min}

Output: clusters, bruit

- 1 Prendre un point x qui n'a pas été visité
 - 2 Construire $N_\epsilon(x)$
 - 3 si $|N_\epsilon(x)| < n_{min}$ alors
 - 4 marquer x comme du bruit
 - 5 sinon
 - 6 Initialiser $C = \{x\}$
 - 7 agrandir le cluster C
 - 8 Ajouter C à la liste des clusters
 - 9 Marquer tous les points de C comme visités
 - 10 Répéter les étapes de (1) à (3) tant qu'il y a des points non visités
-

6.3.3 HDBSCAN

HDBSCAN est un algorithme hybride : il étend DBSCAN en le convertissant en un algorithme de classification hiérarchique, puis il utilise une technique basée sur la stabilité des clusters pour déterminer les regroupements les plus pertinents. Son fonctionnement étant complexe, nous nous limiterons à expliquer comment il parvient à obtenir des meilleures clusters que DBSCAN.

Avec DBSCAN, il faut imperativement choisir le bonne valeur de ϵ pour obtenir les clusters les plus pertinents. Mais comme mentionné précédemment, cette valeur n'est pas évidente à fixer. Mieux encore, cette stratégie s'avère peu efficace lorsque la densité des clusters varie. Dans l'exemple ci-dessous, on peut fusionner les clusters bleu et jaune ou ne pas réussir à capturer le cluster rouge.

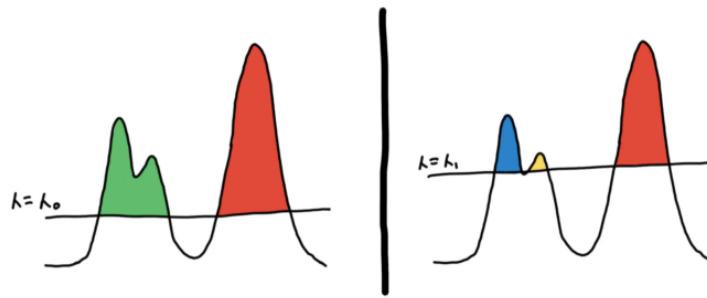


FIGURE 6.4 – Les clusters trouvés dépendent du choix de λ (équivalent de ϵ). Dans DBSCAN, cette valeur est fixée dès le début en paramètre.

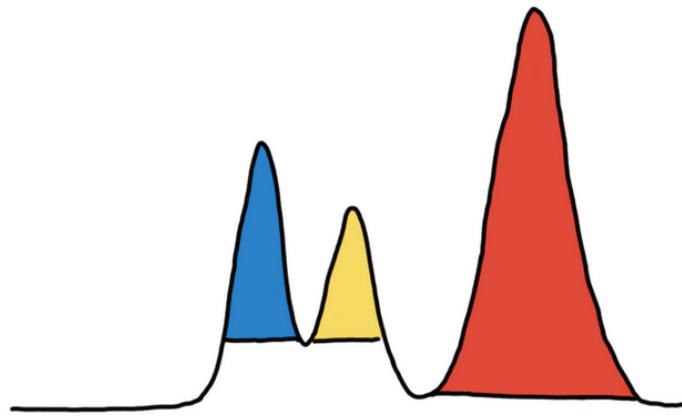


FIGURE 6.5 – Trois clusters avec des densités significativement différentes.

Pour résoudre ce problème, HDBSCAN utilise plusieurs valeurs de ϵ et cherche à déterminer quels sont les sommets qu'ils faut séparer ou fusionner. Pour ce faire, il utilise le critère d'aggrégation suivant : si la somme des volumes de deux sommets est supérieure au volume de leur base alors on peut les considérer comme des montagnes distinctes (on doit les séparer). Dans le cas contraire, il s'agit juste de deux sommets d'une même montagne (on doit les fusionner).

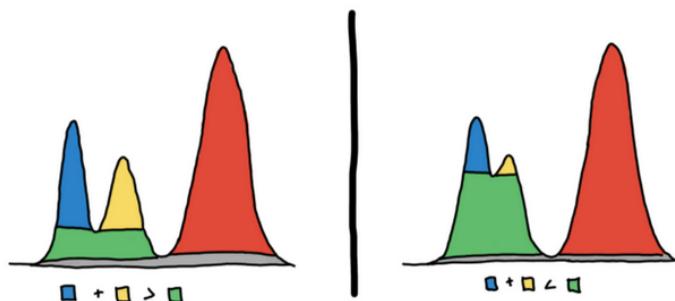


FIGURE 6.6 – Il semble avoir trois clusters (à gauche) et deux clusters (à droite). Le critère d'aggrégation est visible en dessous de chaque image.

Chapitre 7

Résultats et Discussion

7.1. Débruitage

ICA fonctionne avec plusieurs paramètres. Pour nos tests, nous avons utilisé les paramètres *n_components* (nombre de composantes indépendantes), *random_state* (initialisation), *tol* (tolérance contrôlant le critère de convergence) et *max_iter* (le nombre d'itérations).

Pour les tests, le nombre d'itérations de ICA a été fixé à 300 (la valeur par défaut est 200). La tolérance par défaut a été conservée (0.0001). On rappelle que si les paramètres *tol* et *max_iter* sont mal choisis, ICA ne converge pas et le résultat n'est pas fiable.

Pour la détermination du nombre optimal de sources, le nombre d'itérations a été fixé à 50 et le nombre de composantes indépendantes varie entre 2 et 30. ICA a donc été exécutée $50 * 29 = 1450$ fois. Le clustering des composantes indépendantes a été réalisé en utilisant le clustering hiérarchique. Le lien moyen a été choisi comme métrique d'aggrégation des clusters. Pour comparer les résultats, nous avons utilisé le coefficient de silhouette. Le coefficient de silhouette varie entre -1 et 1 et le coefficient défini entre 0 et 1. Pour les deux indices de validation, le nombre de clusters optimal est celui qui maximise la valeur de l'indice.

Les résultats indiquent que le nombre de clusters optimal pour les données de la direction Est-Ouest vaut 12 en utilisant le coefficient défini et 2 en utilisant le coefficient de silhouette. Dans la direction Nord-Sud, on obtient respectivement 19 et 2 clusters avec le coefficient défini et le coefficient de silhouette. On remarque que les valeurs du coefficient sont très faibles (< 0.6) et que celles du coefficient de silhouette décroissent et reste relativement élevées (> 0.7).

7.2. Filtrage et clustering

On adopte les notations suivantes pour des raisons pratiques : le test de significativité des vitesses moyennes sera noté $F_{regression}$, le filtre sur les variations azimutales F_{azimut} , le filtre sur la pente topographique F_{pente} , le filtre sur le kurtosis $F_{kurtosis}$ et le filtre sur les vitesses moyennes $F_{vitesse}$. Si F_1, F_2, \dots, F_n sont n filtres, on note $F_1 \wedge F_2 \wedge \dots \wedge F_n$ le filtre qui laisse passer les pixels qui vérifient les conditions de chaque filtre F_i .

Pour le filtrage des pixels, nous avons testé trois scénarios. Dans le premier scénario, nous avons appliqué individuellement les filtres $F_{regression}$, F_{azimut} et $F_{kurtosis}$. Dans le second scénario, nous avons utilisé le filtre $F_{kurtosis} \wedge F_{regression}$. Dans le troisième scénario, nous avons appliqué les filtres du premier scénario simultanément. Afin de pouvoir détecter les micro-glissements nous n'avons pas appliqué le filtre $F_{vitesse}$. Le filtre F_{pente} est inutile car il n'y a quasiment pas de zones plates.

Dans l'étape du clustering, nous avons travaillé uniquement avec les paramètres *min_cluster_size* et *min_samples* de HDBSCAN. Le paramètre *min_cluster_size* permet de fixer la taille minimum d'un cluster. Le paramètre *min_samples* permet de contrôler la densité des clusters ; plus il est grand plus

les clusters trouvés sont denses et plus on a des points catégorisés comme du bruit. Nous avons testé plusieurs valeurs de min_cluster_size , la valeur 60 a été retenue car elle donne les meilleurs résultats. Le paramètre min_samples a été fixé 1 pour réduire le nombre de pixels catégorisés comme du bruit. Pour comparer les séries temporelles, la distance euclidienne a été préférée à la distance élastique (DTW) pour des raisons de performance.

Les résultats du scénario (1) montre qu'aucun filtre appliqué individuellement ne parvient à éliminer le bruit efficacement. Dans le scénario (2), le glissement de La Valette et un micro-glisement situé à l'Ouest de La Valette sont détectés. La détection du micro-glisement est intéressante car il est situé dans une zone végétalisée où la corrélation d'images donne de mauvais résultats (la végétation a pouvoir réflecteur très faible). De plus, la détection s'opère dès un seuil de confiance égale à 5%. L'augmentation du niveau de risque α ne permet pas d'avoir plus de pixels dans le glissement de La Valette. Au contraire, son augmentation introduit plus de bruit.

Dans le scénario (3), seul le glissement de La Valette est détecté. Dans les scénarios (2) et (3), on remarque que seule la partie supérieure du glissement de La Valette est détectée. En effet, dans cette partie du glissement, les déplacements sont quasi linéaires et les valeurs du kurtosis sont positives et très élevées. Dans la partie inférieure du glissement, les déplacements sont moins linéaires (plus chaotiques) et les taux de déplacements relativement faibles. Le filtre $F_{\text{regression}}$ est en grande partie responsable de la non-détection de cette partie.

Globalement, les clusters trouvés ne sont pas toujours spatialement cohérent. Ceci peut s'expliquer par le fait que le clustering est réalisé en domaine temporel et non spatial.

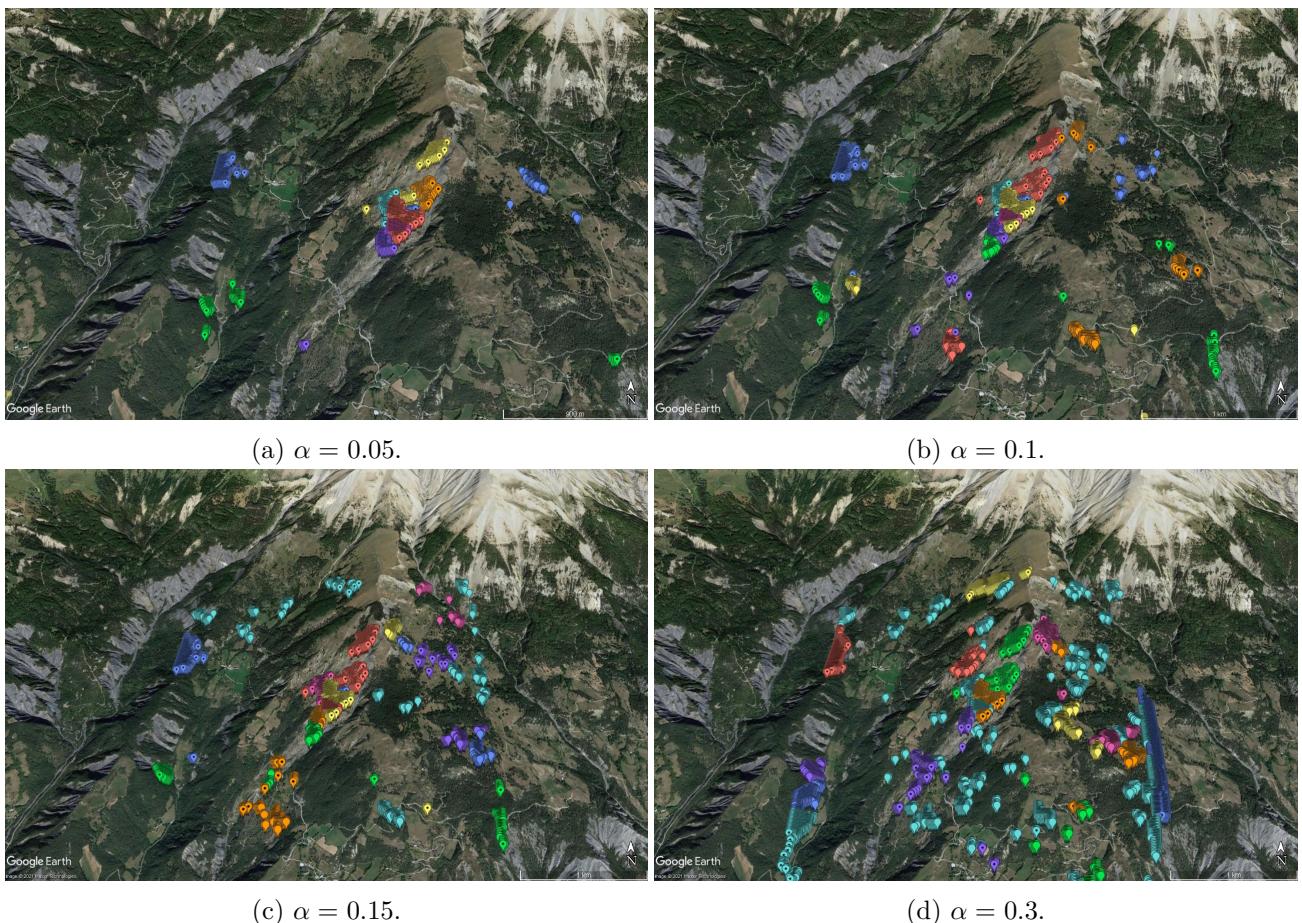
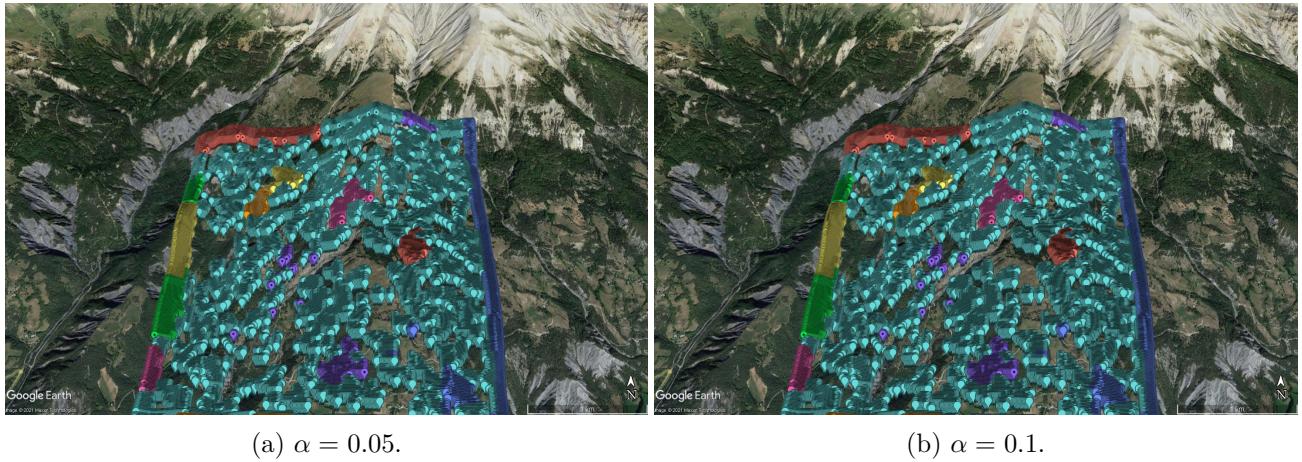
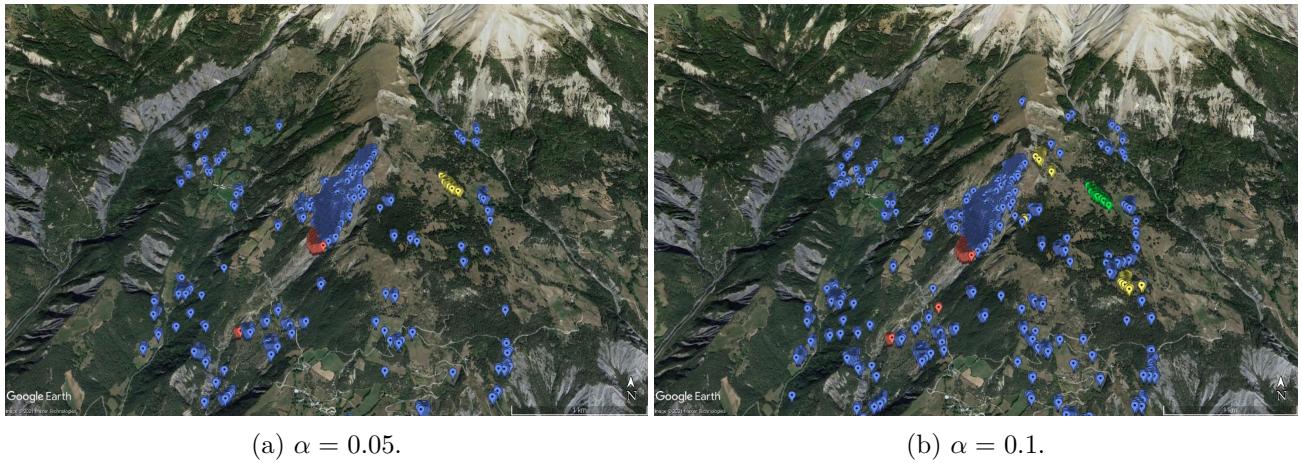


FIGURE 7.1 – Filtre $F_{\text{regression}} \wedge F_{\text{kurtosis}}$ pour différentes valeurs du niveau de risque α .

FIGURE 7.2 – Filtre $F_{\text{regression}} \wedge F_{\text{azimut}}$ pour différentes valeurs de α , tol et p_r .FIGURE 7.3 – Filtre $F_{\text{vitesse}} \wedge F_{\text{regression}} \wedge F_{\text{kurtosis}}$ pour différentes valeurs de α et $|v| > \sigma$.

7.3. Discussion

7.3.1 Débruitage

Choix des paramètres

L'intervalle du nombre de sources peut poser problème dans cet algorithme. En effet, pour un nombre de sources donné, ICA peut ne pas converger si la tolérance est mal choisie. Dans notre cas, toutes les exécutions de ICA ont convergé en conservant la valeur par défaut de la tolérance. On insistera aussi sur le fait que le nombre d'itérations n'ait pas été choisi assez grand. La significativité des résultats n'est donc pas garantie.

Qualité du clustering

Les auteurs du logiciel ICASSO proposent d'utiliser la corrélation absolue ($1 - |r|$) pour comparer deux sources. Mais la présence de la valeur absolue dans la formule soulève naturellement une question : deux sources ayant une corrélation (Spearman ou Pearson) égale à -1 sont-elles identiques ? Comme l'ont montré les résultats, le coefficient prédéfini et le coefficient de silhouette sont en désaccord. Il apparaît donc important de choisir un indice de validation adapté aux données.

Les résultats ont aussi montré que le nombre de composantes indépendantes trouvés dans les deux directions ne sont pas nécessairement identiques. Ceci peut s'expliquer par le fait que le bruit n'est

pas le même dans les deux directions ou qu'il y a du mouvement dans une seule direction. Dans notre cas, il pourrait s'agir du bruit vu qu'il y a clairement du mouvement dans les deux directions.

7.3.2 Détection

Linéarité des déplacements

La p-valeur calculée lors du test de significativité d'une regression linéaire dépend des résidus du modèle. Les déplacements très bruités exhibant une tendance linéaire peuvent ne pas être détecté par le filtre. Pour prendre en compte cet aspect, nous avons fait varier le niveau de risque α qui est souvent fixé à 5%. La valeur du niveau de risque α doit permettre d'améliorer la détection. Etant donnée que les données sont très bruités, il n'est pas totalement impertinent de choisir des valeurs supérieure à 5%. Les tests réalisés avec des valeurs plus grandes (10%, 15%, 30%) ont permis de compter plus de pixels dans le glissement de La Valette mais elles ont en même temps augmenté le nombre de faux-positifs. Si les séries temporelles sont à la fois cycliques et linéaires, il faut nécessairement les désaisonnaliser avant d'appliquer un modèle de regression linéaire.

Combinaison de filtres

L'application simultanée de tous les filtres ne conduit pas nécessairement à de meilleurs résultats. La condition de filtrage globale est très contraignante et les filtres ne sont pas toujours en accord. Si on possède plusieurs filtres, il faudrait théoriquement tester toutes combinaisons possibles afin de trouver le meilleur filtre. Dans notre cas, nous n'avons testé que deux combinaisons conduisant chacun à des résultats différents.

Profils de vitesse moyenne

Le clustering est réalisé sur les profils de vitesse moyenne calculés à partir des déplacements Est-Ouest et Nord-Sud. Cette méthode a permis de passer de deux séries temporelles (les deux composantes du déplacement) à une seule mais elle repose sur l'hypothèse de la linéarité des déplacements qui n'est pas toujours vérifiée. Pour éviter d'émettre des hypothèses sur le régime de la déformation (linéaire, quadratique, etc.), on peut envisager d'effectuer directement le clustering sur chaque composante du déplacement (Il peut avoir du mouvement sur une composante et pas une autre). Autrement dit, clusteriser les séries temporelles de la composante Nord-Sud et ceux de la composante Est-Ouest séparemment. Il faudra ensuite définir une méthode pour unifier les clusters trouvés.

Bibliographie

- [1] Gokhan Aslan, Michael Foumelis, Daniel Raucoules, Marcello De Michele1, Severine Bernarde1and, Ziyadin Cakir : Landslide Mapping and Monitoring Using Persistent Scatterer Interferometry (PSI) Technique in the French Alps.
- [2] Provost Floriane, Michea David, Malet Jean-Philippe, Boissier Enguerran, Pointal Elisabeth, Stumpf Andre, Pacini Fabrizio, Doin Marie-Pierre, Lacroix Pascal, Bally Philippe : Terrain deformation measurements from optical satellite imagery : the MPIC-OPT processing services for geohazards monitoring.
- [3] Noélie Bontemps, Pascal Lacroix, Marie-Pierre Doin : Inversion of deformationfields time-series from optical images, and application to the long term kinematics of slow-moving landslides in Peru.
- [4] Pascal Lacroixa, Grégory Bièvrea, Erwan Pathiera, Ulrich Kniessb, Denis Jongmansa : Use of Sentinel-2 images for the detection of precursory motions before landslide failures.
- [5] Floriane Provost, Jean-Philippe Malet : Spatiotemporal ICA/PCA decomposition of optical displacement field stacks : perspective for landslide time series inversion.
- [6] Gabriella Milone and Germana Scepi : A Clustering Approach for Studying Ground Deformation Trends in Campania Region trough PS-InSAR Time Series Analysis.
- [7] M. Berti, A. Corsini, S. Franceschini, and J. P. Iannaccone : Automated classification of Persistent Scatterers Interferometry time series.
- [8] Ulykbek Kairov, Laura Cantini, Alessandro Greco, Askhat Molkenov, Urszula Czerwinska, Emmanuel Barillot1and Andrei Zinovyev : Determining the optimal number of independent components for reproducible transcriptomic data analysis.
- [9] Johan Himberg and Aapo Hyv arinen : ICASSO : SOFTWARE FOR INVESTIGATINGTHE RELIABILITY OF ICA ESTIMATES BY CLUSTERING AND VISUALIZATION.