UCLouvain

epl

École polytechnique de Louvain

Mesures de débit en rivière à partir de techniques d'imagerie

Développement d'une application

Auteurs: **Célestin Dufromont, Pierre Laurent**
Promoteur: **Sandra Soares-Frazão**
Lecteurs: **Bruno Dehez, Pierre-Yves Gousenbourger**
Année académique 2021–2022
Master [120] : ingénieur civil des constructions, Master [120] : ingénieur civil mécanicien

Résumé

Vouloir connaître le débit d'un cours d'eau n'est pas quelque chose de nouveau. En effet, depuis l'Antiquité, l'Homme se questionne sur la vitesse des cours d'eau qui l'entourent. Plusieurs instruments permettant de calculer le débit ont alors vu le jour au cours du temps. Cependant, leur utilisation n'est pas adaptée à des conditions exceptionnelles telles que des crues ou des assèchements.
Dans un contexte dicté par le changement climatique, où s'alternent périodes de crues et d'étiages, il est nécessaire de pouvoir compter sur un autre type de mesure afin de connaître le débit des cours d'eau en toutes circonstances.

C'est dans ce contexte que sont nées les techniques de mesure non-intrusives et avec elles, une application en mesure d'approximer le débit réel lorsqu'une vidéo et une bathymétrie lui sont fournies.

Bien que conçue pour être la plus intuitive possible, l'application nécessite une certaine connaissance pour obtenir des débits convenables. Un guide utilisateur complet a donc été mis au point pour garantir la compréhension de l'application. L'influence de certains paramètres sera également analysée afin de déterminer des réglages optimaux.

Une seconde partie sera dédiée à la réalisation de mesures sur le terrain et à l'expérimentation d'un kit permettant de les faciliter.

Enfin, afin de permettre l'amélioration et l'évolution de l'application par tout un chacun, la dernière partie de ce document s'intéressera à la mise en Open Source du code et décrira les tenants et les aboutissants de ce processus.

Abstract

Wanting to know the flow of a river is not something new. Indeed, since the Antiquity, the Man wonders about the speed of the rivers that surround him. Several instruments allowing to calculate the flow have then been created over time. However, their use is not adapted to exceptional conditions such as floods or droughts.
In a context dictated by climate change, during which periods of high and low water alternate, it is necessary to be able to count on another type of measurement to know the flow of the rivers in all circumstances.

It is in this context that non-intrusive measurement techniques were born and with them, an application able to approximate the real flow when video and bathymetry are provided.

Although designed to be as intuitive as possible, the application requires some knowledge to obtain suitable flow rates. A comprehensive user guide has been developed to ensure that the application is understood. The influence of certain parameters will also be analyzed in order to determine optimal settings.

A second part will be dedicated to the realization of measurements on the field and to the experimentation of a kit allowing to facilitate them.

Finally, in order to allow the improvement and the evolution of the application by everyone, the last part of this document will focus on the Open Source of the code and will describe the ins and outs of this process.

Remerciements

Nous tenons à adresser nos remerciements les plus sincères à toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail de fin d'études, et plus particulièrement :

Notre promotrice, la Professeure Sandra Soarez-Frazão, pour son accompagnement et ses conseils avisés qui ont orienté notre travail depuis le début,

Pierre-Yves Gousenbourger, assistant de recherche à l'UCLouvain, pour son aide plus que précieuse et les compétences qu'il a su nous transmettre, et dont l'énergie nous a motivé quand nous en avions besoin,

Le Pofesseur Bruno Dehez pour avoir accepté de faire partie de notre jury

Firas Ghali pour l'aide apportée sur le terrain et dans l'analyse des données récoltées,

Charlotte Everaert pour l'aide à la prise en main de l'application,

Rotchild Louis pour la qualité des vidéos qu'il est parvenu à nous transmettre,

Les scouts de Luingne (MO006) pour l'aide apportée lors la construction du kit et les scouts de Libramont (LU005) sans qui prendre des mesures à Lissoir n'aurait pas été possible,

Mr. Sébastien Adam, expert au LTTO, pour son temps et son savoir à propos de l'Open Source,

Enfin, nos proches pour les moments qu'ils nous ont fait vivre pendant les cinq années de nos cursus.

Table des matières

Introduction **1**

I État de l'art 3

1 Mesures du débit **5**
1.1 Techniques intrusives 6
1.1.1 Techniques non-basées sur la mesure de vitesse 6
1.1.2 Techniques basées sur la mesure de vitesse 10
1.2 Techniques non-intrusives 15
1.2.1 Particle Image Velocimetry 15
1.2.2 Large Scale Particle Image Velocimetry 17

2 Concepts utiles à la compréhension de l'application **18**
2.1 Stabilisation 18
2.2 Orthorectification 19
2.3 Analyse PIV 21
2.3.1 Création du champ de vecteurs 21
2.3.2 Validation 23
2.4 Coefficient de surface 25
2.5 Exploitation du champ de vecteurs 26

II État de l'application 27

3 Structure de l'application et guide utilisateur **28**
3.1 Structure de l'application et informations générales 28
3.2 Un guide utilisateur 30
3.2.1 Récupération du projet 30
3.2.2 Vérification du contenu du dossier 30
3.2.3 Un IDE ***Python*** 31

3.2.4 Un environnement virtuel 31
3.2.5 Données nécessaires à l'utilisation de l'application 32
3.2.6 Remarques générales applicables à l'entièreté de l'application 32
3.2.7 Tutoriel de première utilisation 32
3.2.8 Lancement de l'application 33
3.2.9 Choix de la bathymétrie 34
3.2.10 Choix de la vidéo 38
3.2.11 Stabilisation 40
3.2.12 Calibration 42
3.2.13 Orthorectification 45
3.2.14 Détermination du niveau d'eau 48
3.2.15 Région d'intérêt et masques 53
3.2.16 Filtres 55
3.2.17 Analyse PIV 58
3.2.18 Section d'intérêt 61
3.2.19 Détermination du débit 63

4 Problèmes rencontrés avec l'application 65
4.1 Problèmes de transmission 65
4.2 Problèmes de compilation 66
4.3 Problèmes de fonctionnement 67
4.4 Problèmes encore persistants 69
4.4.1 Problèmes/manquements de nature : Bug 69
4.4.2 Problèmes/manquements de nature : Ergonomique 71
4.4.3 Problèmes/manquements de nature : Affichage 73
4.4.4 Problèmes/manquements de nature : Fonctionnalité 74
4.5 Approximations/valeurs sans explication 76

III Développements relatifs à l'application 78

5 Mesures sur le terrain 79
5.1 Protocole de prises de mesures 79
5.2 Résultats des mesures à l'étranger 80
5.3 Nouvelles prises de vidéos 82
5.3.1 Noirhat - Dyle - 12 juillet 2022 82
5.3.2 Suzeryl - Thyle - 13 juillet 2022 82
5.3.3 Lissoir - Lesse - 10 au 19 juillet 2022 83
5.3.4 Limelette - Dyle - 21 juillet 2022 83
5.3.5 Bilan des mesures 84

6 Kit de mesure 85
6.1 Cahier des charges 85
6.1.1 Élaboration des contraintes principales 85

6.1.2 Test vidéo avec l'utilisation du kit 86
6.1.3 Élaboration de contraintes secondaires 89
6.2 Bilan 89

7 Influence de certains paramètres 91
7.1 Paramètres liés à la prise de mesure 91
7.1.1 Résolution d'image 92
7.1.2 Stabilisation 93
7.1.3 Utilisation du kit 95
7.2 Paramètres liés à l'utilisation de l'application 98
7.2.1 Aire d'interrogation 98
7.2.2 Nombre de sous-sections 102
7.2.3 Placement de la section de mesure 104
7.2.4 Rapport signal/bruit - Signal to Noise Ratio - SNR 107

8 Mise en Open Source 112
8.1 L'Open Source : Pourquoi ? 112
8.2 Qu'est ce que l'Open Source ? 113
8.2.1 Un brin d'histoire... 113
8.2.2 Caractéristiques de l'Open Source 114
8.3 Types de licences 116
8.4 Choix de la licence appliquée 116

9 Pistes d'améliorations 119

Conclusion 120

Bibliographie 123

Annexes 128
.1 RiverApp - Instructions première utilisation 128
.2 Données première utilisation 138
.3 Protocole de prises de mesures 139
.4 Récapitulatif des tests effectués sur le terrain 146
.5 Tableau de compatibilité des licences Open Source 148
.6 Liste des licences des packages importés 149

Introduction

Au moment d'écrire cette introduction, la Belgique (et plus largement toute l'Europe de l'Ouest) fait face à une sécheresse installée depuis des mois. "Sur les bases d'une année historique", évoquent même les experts. La situation des cours d'eau provoque des inquiétudes, comme à Sauheid où le débit de l'Ourthe a été estimé à 3 $[m^3/s]$ à la mi-août. La gestion de l'étiage implique donc de prendre des mesures fortes : le barrage de la plate Taille, par exemple, doit relâcher une partie de sa capacité dans la Sambre afin de maintenir un tirant d'eau suffisant permettant la navigation, les pompages sur la Meuse ont été plafonnés, etc.[34].

L'été dernier, au milieu du mois de juillet, la Belgique (et plus largement une partie de l'Europe de l'Ouest déjà) connaissait le scénario inverse et était frappée par des inondations dévastatrices qui demeurent encore aujourd'hui, la catastrophe naturelle la plus meurtrière de l'histoire du pays[7]. Par comparaison, le débit de l'Ourthe à Sauheid le 15 juillet 2021, était estimé à plus de 1090 $[m^3/s]$.

S'il est impossible de lier directement ces catastrophes avec le changement climatique, le GIEC estime qu'il accroît la probabilité et l'intensité de ce type d'évènements et que cette tendance risque de se poursuivre.

Lors de ces épisodes extrêmes, les techniques classiques de mesure de débit ne sont pas adaptées. En effet, les stations fixes ne sont pas calibrées pour de tels écoulements, et les valeurs obtenues ne reflètent parfois pas la réalité, quand les stations ne sont pas tout simplement détériorées ou détruites par la crue. Quant aux méthodes nécessitant l'intervention humaine sur le cours d'eau, elles deviennent alors terriblement dangereuses et sont de facto exclues. D'autre part, dans des parties du monde moins riches que nos pays occidentaux, ces systèmes ne sont juste pas présents ou pas en nombre suffisant, faute de moyens.

C'est pour répondre à ces problématiques, qu'une application de mesure du débit utilisant des vidéos est créée en 2020 par Simon Renard et améliorée en 2021 par Charlotte Everaert et Edouard de Grand Ry. C'est leurs travaux que nous avons poursuivis dans ce travail de fin d'études.

Nous avons, dans un premier temps, pris en main l'application et rapidement constaté une certaine difficulté à réellement la mettre en route pour obtenir des débits corrects. Forts de cette expérience, nous avons conçu un guide utilisateur qui permettra à tous les usagers futurs de posséder toutes les informations nécessaires à la bonne marche du logiciel.

En parallèle, nous avons étoffé le set de vidéos utilisables en procédant à des mesures sur des sites proches de Louvain-la-Neuve, ce qui nous a de surcroît permis de tester un prototype de kit facilitant lesdites mesures. Nous avons également établi un contact avec des chercheurs d'Haïti, de Madagascar et de Tunisie et leur avons fourni des protocoles de mesure. Par après, ceux-ci ont tenté de nous envoyer des enregistrements de leurs cours d'eau locaux.

Utilisant ces nouvelles vidéos ainsi que celles déjà en notre possession, nous avons pu tester et analyser l'influence de certains paramètres sur les résultats fournis par l'application.

Enfin, pour permettre à plus de contributeurs de développer l'application, nous avons entamé le processus de mise en Open Source en plaçant le code source de l'application sur la forge de l'UCLouvain et en gérant les différentes licences liées aux librairies utilisées.

Nous terminerons par une synthèse des résultats obtenus et des pistes d'amélioration pour d'éventuels travaux futurs.

Première partie

État de l'art

Préambule : comment calculer un débit ?

Pour bien comprendre le cheminement effectué par l'application, il convient d'expliquer brièvement comment un débit est calculé.

Un débit est un volume d'écoulement par unité de temps. Physiquement, il se calcule comme la vitesse de l'écoulement multipliée par la surface traversée par ce dernier (Figure 1) :

$$Q[m^3/s] = V[m/s] \cdot A[m^2] \tag{1}$$

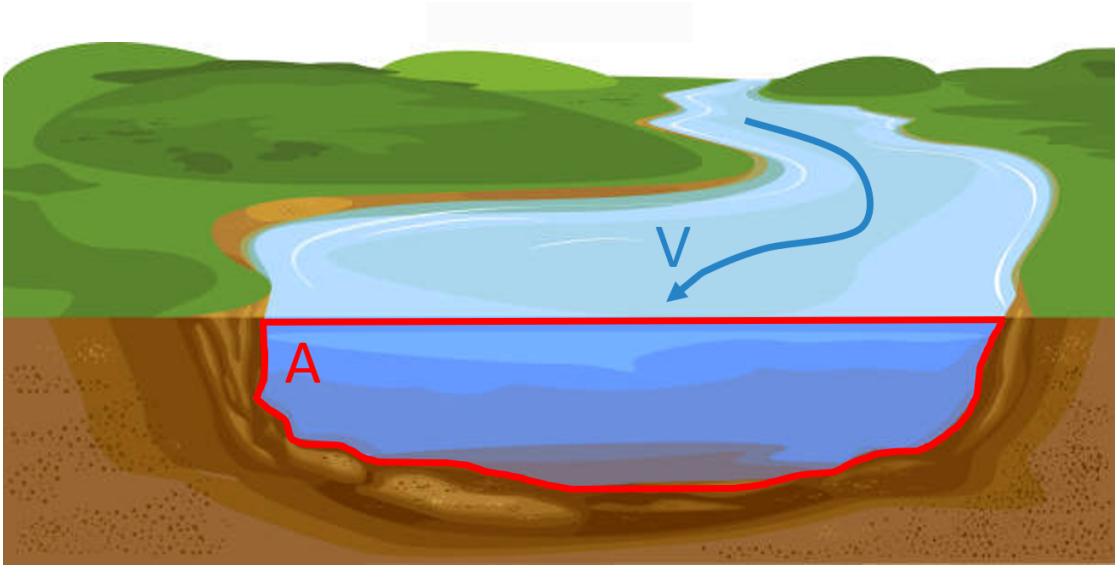


FIGURE 1 – Vitesse et aire mouillée d'une rivière.

Bien sûr, cette équation est la plus simple qu'il est possible d'écrire pour obtenir un débit. Dans la réalité, la détermination du débit résulte de formules plus complexes, liées au fait que ces données ne sont pas uniformes spatialement. Toutefois, cela nous permet d'introduire les données nécessaires à l'application : la vitesse de l'écoulement et l'aire mouillée

Les techniques de mesure de l'aire traversée ne seront pas présentées dans ce document puisque notre travail s'est surtout porté sur l'obtention de la vitesse de surface et du débit. D'autres mémoires portant sur le sujet, comme ceux de Lebon L. et Willemaert E. (2021)[21], Breels N. et Marneffe S. (2020)[22] ou Renard S. (2020)[61] évoquent plus en détails les techniques d'obtention de l'aire mouillée.

Chapitre 1

Mesures du débit

Déterminer un débit peut se faire à travers de nombreuses méthodes qui dépendront des caractéristiques du cours d'eau. Parmi celles-ci, on peut énoncer entre autres [67] :

- L'ordre de grandeur du débit
- La configuration du site
- La nature de l'écoulement
- Le personnel disponible
- La précision voulue

Les méthodes que nous reprendrons ici sont les plus courantes ou celles qui n'ont pas encore été évoquées par nos prédécesseurs. Ces méthodes peuvent être séparées en deux catégories, comme cela a été fait dans le mémoire de de Grand Ry E. et Everaert C.[11] : les techniques intrusives et non-intrusives.

L'intrusivité est ici décrite comme la nécessité pour l'appareil de mesure d'établir un contact avec la surface ou le cours d'eau.

1.1 Techniques intrusives

1.1.1 Techniques non-basées sur la mesure de vitesse

Mesures limnimétriques et courbes de tarage

Les stations limnimétriques constituent la grande majorité des stations des réseaux Wacondah [1] et Aqualim [2].

Un limnimètre est un équipement qui permet la lecture, l'enregistrement et la transmission de la hauteur d'eau en un point. Il est constitué d'une règle graduée depuis le fond sur laquelle peut être lue la hauteur d'eau (Figure 1.1).

FIGURE 1.1 – Échelle limnimétrique de la station de Noirhat.

Si celui-ci est équipé d'un dispositif d'enregistrement, on parle alors de limnigraphe. Les trois principaux types de limnigraphes sont :

- Les limnigraphes à flotteur utilisent un flotteur et un contrepoids pour inscrire le niveau sur un tambour rotatif créant des diagrammes appelés limnigrammes. Ils sont les plus anciens et leur autonomie est restreinte à trois mois maximum.

1. Le système WACONDAH (WAter CONtrol DAta system for Hydrology and water management) est géré et développé par la Direction de la Gestion Hydrologique (DGH) du SPW Mobilité et Infrastructures, notamment pour la prévision des crues et des étiages ainsi que la gestion des voies hydrauliques et des réservoirs.[15]

2. Aqualim est un réseau de mesure en continu des hauteurs et débits géré par la Direction des Cours d'Eau non navigables et concernant les cours d'eau de catégorie 1.[15]

- Les limnigraphes à pression ou "bulle à bulle" mesurent les variations de pression causées par les changements de niveau d'eau. Un débit d'air constant sous pression est envoyé au fond de la rivière. Dans un tube relié à une bonbonne de gaz comprimé, et via un manomètre à mercure, on mesure la pression de l'air qui est proportionnelle à la hauteur d'eau au-dessus de la prise installée dans la rivière.
- Les limnigraphes électroniques à capteur piézo-électrique utilisent des condensateurs munis d'une partie mobile induisant des variations de tension mesurables.

Ces derniers remplacent peu à peu les deux autres types en raison de leur faible entretien. De fait, il ne faut pas changer régulièrement les tambours ou les bonbonnes d'air comprimé.

En mesurant le débit correspondant à différentes hauteurs lues au limnimètre, il est possible d'établir une relation liant le débit à la hauteur (Figure 1.3). La courbe ainsi obtenue est appelée une courbe de tarage, et l'opération de mesure du débit créant un point de la courbe est appelée un jaugeage.

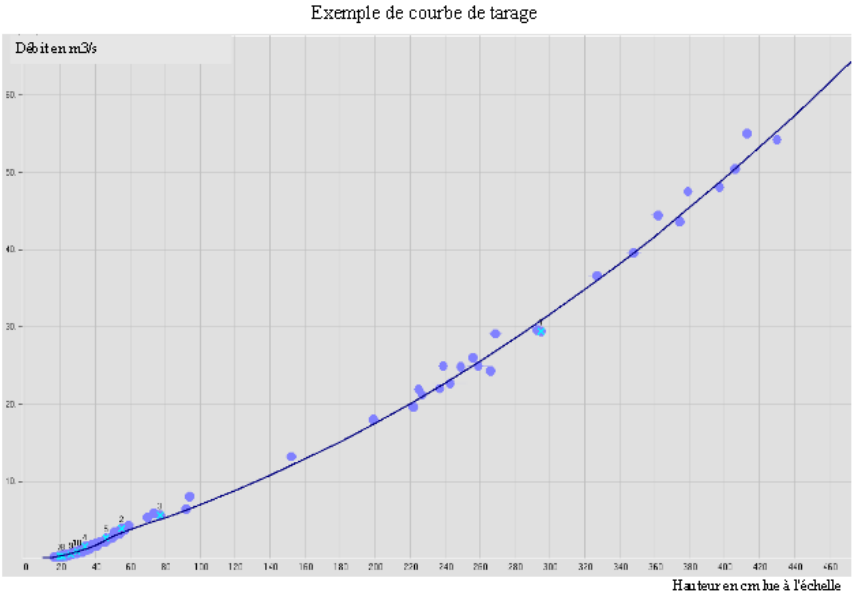


FIGURE 1.2 – Exemple de courbe de tarage et des points de jaugeage issu de [14].

Dès lors que cette relation existe pour un endroit de mesure, le lien est direct et les limnigraphes sont en mesure de transmettre le débit. Il est à noter que dans certains cas, la relation hauteur-débit n'est pas univoque (une hauteur ne correspond pas à un seul débit), cette technique n'est donc pas possible. Cela survient aux endroits ou le débit est contrôlé par des ouvrages (écluses, barrages,etc.).

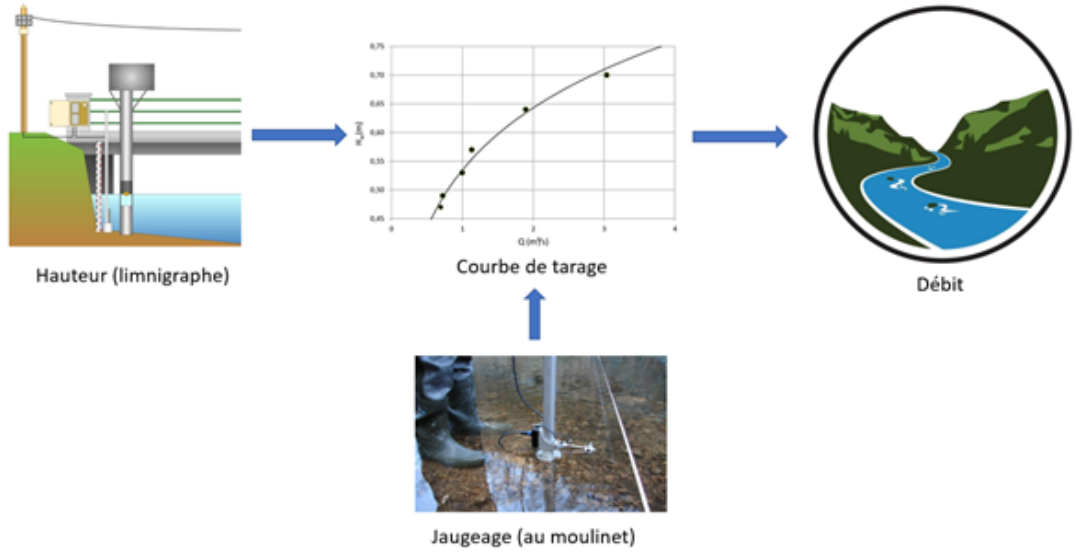


FIGURE 1.3 – Illustration de l'utilisation d'une courbe de tarage. Image issue de [16].

Les jaugeages doivent être effectués régulièrement afin de vérifier la validité des débits transmis. En période d'étiage, une petite modification de la bathymétrie [3] aura un impact sur la hauteur d'eau au point de mesure, altérant ainsi le débit fourni. En outre, peu de jaugeages ayant lieu en période de crues, les valeurs de la courbe de tarage pour les grandes hauteurs d'eau sont donc extrapolées et peuvent être éloignées de la réalité. Ce type de mesure est donc adapté aux débits moyens et au suivi régulier de l'hydrométrie. [2, 3, 14, 15, 35, 64, 67, 68]

Jaugeage par dilution de NaCl

Cette méthode, adaptée aux cours d'eau de faibles débits, s'effectue en injectant une quantité connue de chlorure de sodium dans l'écoulement et en mesurant la concentration du nuage salin via la conductivité électrique étant donné l'existence d'une relation linéaire entre ces deux grandeurs (Figure 1.4).

3. Dans ce mémoire, par bathymétrie est entendu l'ensemble des coordonnées (x,y) de la section de mesure.

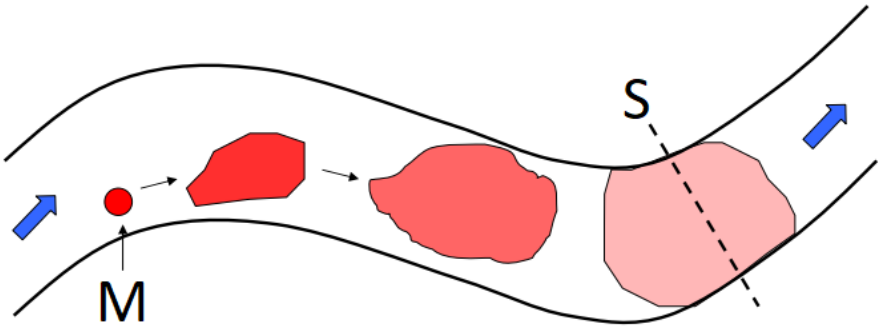


FIGURE 1.4 – Schéma illustrant un jaugeage au sel. Image issue de [12].

Si l'on établit que ΔC_m est l'augmentation de la concentration liée à la manipulation, M la masse de NaCl introduite et V le volume dans lequel cette masse s'est dissoute, on sait que :

$$\Delta C_m = \frac{M}{V} \tag{1.1}$$

De plus, si T est le temps de passage du nuage salin, on a :

$$Q = \frac{V}{T} \Leftrightarrow V = Q.T \Leftrightarrow \Delta C_m = \frac{M}{Q.T} \tag{1.2}$$

Le débit étant la valeur recherchée, on trouve :

$$Q = \frac{M}{\Delta C_m.T} \tag{1.3}$$

ou, si l'on dispose de mesures de concentration suffisamment proches dans le temps :

$$Q = \frac{M}{\int_{t_1}^{t_2} \Delta C_m.dt} \tag{1.4}$$

Il est possible d'utiliser d'autres composés que du chlorure de sodium, moyennant un appareillage de mesure adapté (colorimètre, radiomètre ,fluorimètre) à la substance employée (resp. Na_2CrO_4, or radioactif, Rhodamine par exemple).

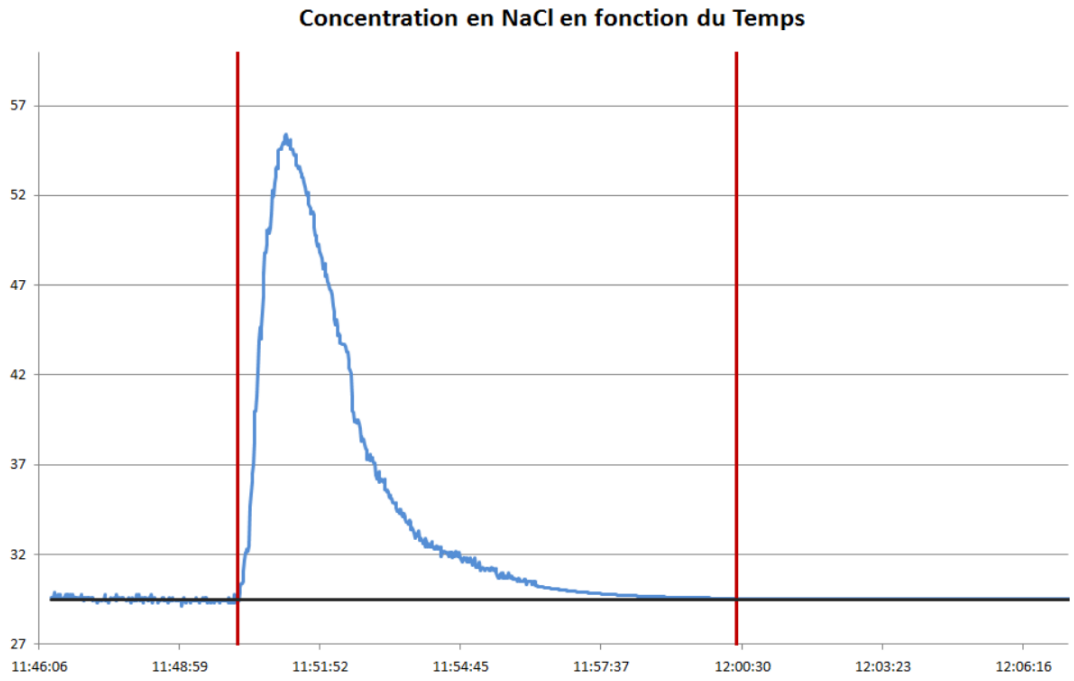


FIGURE 1.5 – Exemple de suivi de la concentration du nuage salin. Image issue [12].

Ce procédé nécessite de choisir très consciencieusement le tronçon de mesure, qui doit être accessible, ne pas présenter trop de recirculations et être d'une longueur adéquate (de 50 à 100 mètres) pour laisser le temps au sel de se diluer de façon homogène mais ne pas en perdre trop sur le trajet. De plus, l'écoulement doit être suffisamment turbulent pour permettre un brassage suffisant du traceur.

Cette technique est précise mais nécessite du temps et des moyens humains. L'impact écologique peut également se révéler non négligeable. [12, 20, 56, 35]

1.1.2 Techniques basées sur la mesure de vitesse

Flotteurs

Ce procédé consiste à poser un flotteur à la surface de l'eau et à mesurer le temps que celui-ci met pour effectuer le trajet entre deux points dont l'inter-distance est connue. Les flotteurs utilisés peuvent être de quelque nature que ce soit, mais pour éviter l'influence de facteurs extérieurs (vent, etc), il est conseillé d'utiliser des flotteurs conçus adéquatement. Les mesures doivent être réalisées plusieurs fois afin de pouvoir établir une moyenne (Figure 1.6).

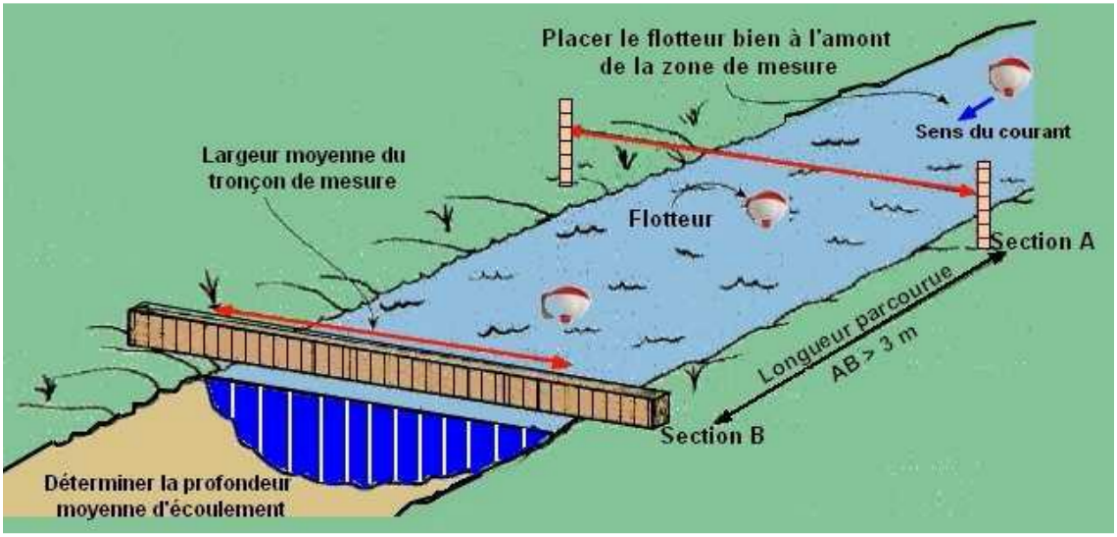


FIGURE 1.6 – Illustration de la méthode des flotteurs. Illustration tirée de [35].

On établit que d est la distance entre les points d'injection et de récupération des flotteurs, et t le temps mis pour parcourir la distance d. Une correction peut éventuellement être apportée pour tenir compte de la différence d'angle entre la trajectoire et la normale à la section. On peut alors aisément calculer la vitesse moyenne :

$$V_{moy} = \frac{d}{t} \tag{1.5}$$

Le débit se calcule alors comme :

$$Q = k.V_{moy}.A \tag{1.6}$$

où $k(< 1)$ est un coefficient de débit reliant la vitesse de surface à la vitesse moyenne sur la profondeur (voir Section 2.4) et A l'aire mouillée.

Cette méthode est simple, bon marché et rapide et fournit de bonnes approximations selon les applications voulues. On recommande son utilisation quand la mesure au moulinet, plus précise, est impossible en raison d'une profondeur ou une vitesse inadaptée ou de la présence de matériaux en suspension. [14, 27, 35, 67]

Courantomètre mécanique

FIGURE 1.7 – Moulinet OTT C31 utilisable jusqu'à 10 [m/s].

Cet appareil fonctionne en utilisant la relation entre la vitesse de rotation d'une hélice et la vitesse de l'écoulement. Les hélices doivent être choisies et calibrées en fonction de la vitesse maximale que l'utilisateur s'attend à mesurer. Chaque mesure fournit la vitesse en un point du cours d'eau et est effectuée sur une verticale. Les résultats obtenus sont alors transformés en vitesses moyennes sur ces mêmes verticales via des formules adaptées (Tableau 1.1).

Nombre de points de mesure	Profondeur des points de mesure	Vitesse moyenne sur la verticale de mesure
1	$0.6d$	$V_{moy} = V_{0.6}$
2	$0.2d, 0.8d$	$V_{moy} = (V_{0.2} + V_{0.8})/2$
3	$0.2d, 0.6d, 0.8d$	$V_{moy} = (V_{0.2} + 2V_{0.6} + V_{0.8})/4$

TABLE 1.1 – Tableau liant les vitesses ponctuelles et vitesses moyennes issu de [6].

Le débit total est ensuite calculé en faisant la somme des débits de chaque colonne calculés en connaissant la profondeur de chaque verticale (Figure 1.8). Cette méthode d'obtention du débit s'appelle l'exploration du champ de vecteurs.

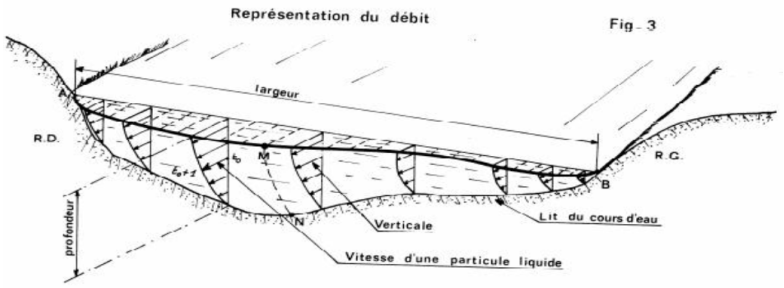


FIGURE 1.8 – Illustration d'un champ de vecteurs exprimé sur les verticales.

Le moulinet hydrométrique peut être monté sur une perche ou descendu dans le cours d'eau à l'aide d'un treuil. Il doit ainsi être placé parallèlement au sens du courant et peut donc produire des erreurs quand des courants de retour se produisent. Cet instrument a l'avantage de pouvoir s'utiliser dans un grand nombre de cas, d'être léger et facilement déplaçable-et il est de plus peu coûteux. Cependant, les résultats obtenus étant ponctuels, les mesures peuvent prendre un certain temps. [6, 13, 14, 35]

Courantomètre électromagnétique

FIGURE 1.9 – Moulinet OTT MF pro utilisable jusqu'à 6 [m/s].

Le courantomètre électromagnétique permet, comme le moulinet mécanique, de fournir la vitesse en un point. L'appareil génère un champ magnétique ce qui va générer une différence de tension, conformément à la loi de Faraday.

Cette tension est mesurée par les électrodes de la sonde. Les résultats ponctuels sont ensuite traités comme ceux issus d'un vélocimètre mécanique. Ce dispositif est robuste et n'est pas influencé par la présence de végétation. Toutefois, il doit être tenu à l'écart des éléments métalliques. À l'instar du moulinet hydrométrique, les mesures peuvent prendre un certain temps. [8, 14, 26, 66]

Acoustic Doppler Current Profiler - ADCP

FIGURE 1.10 – Exemple de capteur ADCP.

Cette technique utilise l'effet Doppler et des ondes acoustiques générées par des transducteurs piézoélectriques. Un signal ultrasonique de fréquence comprise entre 300 [kHz] et 3000[kHz] est émis dans le flux du liquide. Lorsque ce signal est réfléchi par des particules solides ou des bulles d'air, sa fréquence se modifie proportionellement à la vitesse du fluide. Les signaux sont alors rétrodiffusés et parviennent à nouveau au capteur, lui permettant de calculer les trois composantes spatiales de la vitesse.

Cet appareil peut être utilisé en continu et monté sur une embarcation pour effectuer les mesures sur une section par exemple et permet de mesurer la bathymétrie (Figure 1.11).

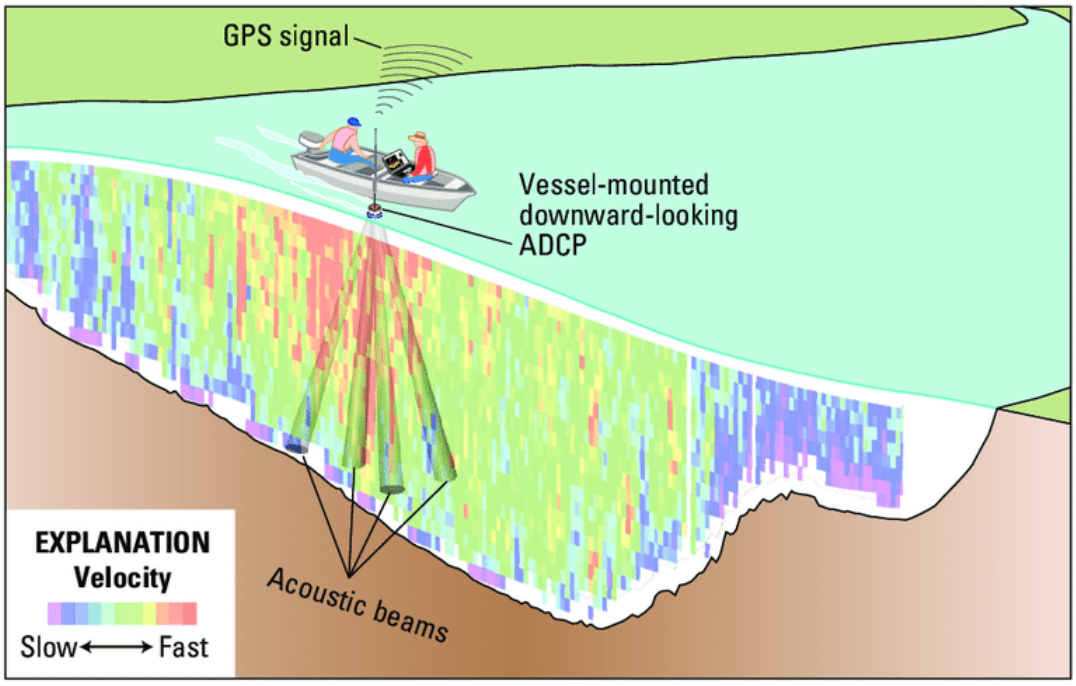


FIGURE 1.11 – Mesure du champ de vitesse via ADCP depuis une embarcation

Toutefois, il possède des zones aveugles près de la surface (de 5 à 100 [cm]) et près du fond (jusqu'à 5m). De plus, bien que la gamme de fréquences utilisée ait été identifiée comme ne causant pas trop de problèmes, l'ADCP génère de la pollution acoustique. [8, 24, 27, 41, 65]

1.2 Techniques non-intrusives

1.2.1 Particle Image Velocimetry

Définition et origine

Adrian R.J.[57] affirme que le terme de *Particle Image Velocimetry* est apparu dans littérature en 1984 [58], mais souligne que Meynart R.[55] évoque une technologie similaire sous le terme de *laser speckle velocimetry* déjà en 1982.

Toutefois, estimer la vitesse d'écoulement d'un fluide en observant la vitesse de particules présentes en surface est probablement la méthode la plus intuitive qui

soit. Dès lors, on peut imaginer que des personnes aient approximé la vitesse de l'eau de rivières en examinant le déplacement de petits débris bien avant que cela soit documenté. Il existe également une technique de mesure du débit utilisant ce concept (Section 1.1.2).

Si la PIV utilise bien ce principe, elle est décrite plus particulièrement par Raffel et al(2018)[19] comme une technique de mesure permettant de capturer des champs de vitesse en fractions de seconde. Adrian(2005)[57], lui, la définit comme la mesure précise et quantitative des vecteurs de vitesse du fluide en un très grand nombre de points simultanément. L'idée n'est donc pas de se focaliser sur les écoulements en rivière mais d'étudier les flux de fluides de manière générale. Tous les auteurs soulignent que les progrès effectués récemment en informatique et en imagerie ont permis à la PIV de se développer et d'étudier des écoulements de plus en plus complexes.

Méthodologie

La méthode se décompose en plusieurs étapes décrites ci-dessous. Toutes ne sont pas indispensables mais leur bonne application mène à de meilleurs résultats.

- **Ensemencement :**
  La surface des fluides étudiés étant souvent homogène, l'ajout de traceurs permet de faire apparaître la vitesse de surface. La taille de ces traceurs ayant un impact sur le résultat obtenu, il convient de les choisir avec certaines caractéristiques (taille, couleur, répartition,etc.)[28]. L'introduction de marqueurs peut être passée si l'écoulement est assez turbulent et possède des remous suffisamment visibles.
- **Prise des images :**
  La prise d'images doit s'effectuer en connaissant le temps séparant deux images consécutives. Les vidéos répondent à ce critère si l'on connaît la fréquence de prise d'images (le nombre de frames par seconde [fps]).
- **Prétraitement :**
  Avant de passer à l'analyse PIV en elle-même, il est conseillé de procéder à un prétraitement des images récoltées, via l'application de filtres accentuant la visibilité des traceurs et de masques ôtant les zones dénuées d'intérêt.
- **Analyse PIV :**
  Après avoir choisi les paramètres adéquats, on peut effectuer l'analyse PIV produisant alors un champ de vecteurs. Cette étape sera décrite plus en détails à la section 2.3
- **Calibration :**
  On établit dans cette étape la relation pixels/mètres nécessaire pour pouvoir exprimer les résultats en tant que vitesse. De plus, cela permet de se débarrasser d'éventuelles déformations inhérentes à la caméra utilisée.

- **Post-traitement :**
  Les vecteurs jugés non-pertinents sont ici enlevés et remplacés par des vecteurs interpolés.

1.2.2 Large Scale Particle Image Velocimetry

La Large Scale Particle Image Velocimetry est littéralement une PIV à plus grande échelle. En pratique, la LSPIV est l'application de la PIV sur des cours d'eau non-contrôlés en laboratoire, et donc en conditions réelles. Cette distinction implique des adaptations pour permettre l'analyse de l'écoulement.

Tout d'abord, il faut s'assurer de préparer le terrain, éventuellement effectuer des repérages et dans certains cas placer des balises.

Sur le terrain, il est beaucoup plus compliqué de réguler la luminosité qu'en laboratoire, et ce paramètre peut varier au cours d'une même mesure. La taille des traceurs doit évidemment être augmentée mais il faut également tenir compte de la nature de ces derniers : il n'est pas question de jeter n'importe quels objets dans le cours d'une rivière. En outre, il sera parfois compliqué de prendre des images tout à fait perpendiculairement au plan d'eau. Les vidéos devront donc être retravaillées par orthorectification afin de pouvoir effectuer l'analyse PIV (Section 2.2).

C'est Fujita et al.[18] qui sont les premiers à appliquer la technologie PIV à une rivière et à utiliser le terme LSPIV en 1997. Depuis lors, la technologie n'a cessé d'évoluer notamment sur l'utilisation de vidéos amateurs et le recours à des dispositifs de prises de vues aériennes (hélicoptères ou drones).

Chapitre 2

Concepts utiles à la compréhension de l'application

Les différentes sous-sections qui vous seront présentées ci-dessous représentent les concepts utilisés lors de l'application ainsi que les paramètres que nous avons fait varier lors de nos tests.

2.1 Stabilisation

Selon une des définitions apportées par le *Larousse*, stabiliser consiste à fixer quelque chose dans un certain état, l'empêcher de varier [37]. C'est exactement l'effet qui est recherché lorsque l'on applique la stabilisation à une vidéo. De fait, lorsqu'une vidéo est prise depuis un drone ou par une caméra tenue par un être humain, il est impossible que le cadre de la vidéo soit immobile. Il est alors intéressant d'essayer de réduire les mouvements parasites occasionnés par le vent, le corps humain, etc. et donc de fixer le cadre de la vidéo pour ne pas perturber le champ de vecteurs obtenu.

Le fichier `stabilization.py`, présent dans le dossier de l'application, permet à celle-ci de proposer à l'utilisateur de procéder ou non à une stabilisation. Si cette dernière est requise, l'utilisateur est alors amené à sélectionner sur la première image de la vidéo, par un rectangle, des points qui restent fixes par rapport à l'environnement filmé. Cela peut être des GCPs [1] ou des éléments naturels tels que des pierres ou des grosses branches, tant que ces repères restent immobiles.

1. Ground Control Points

Cette stabilisation est ici possible grâce à un algorithme bien précis de la librairie OpenCV, dénommé KCF (Kernerlized Correlation Filters). Le principe d'un tel algorithme est de repérer le centre des éléments sélectionnés dans chaque image de la vidéo et d'enregistrer les coordonnées de ceux-ci. Une autre fonction *Python* est ensuite appelée pour effectuer une transformation de perspective entre les coordonnées des centres repérés dans chaque image de la vidéo et les coordonnées des centres des éléments sélectionnés sur la première image. L'avantage de cet algorithme réside dans sa capacité à être rapide tout en restant précis. Cependant, si un des centres des éléments sélectionnés venait à disparaître de l'image, l'application crasherait alors.

Une fois la stabilisation terminée, la fonction crée une version stabilisée de la vidéo initiale qu'elle nomme *NomDeLaVideo-stabVersion*. Ce nouveau fichier pourra être utilisé pour des utilisations ultérieures.[38]

2.2 Orthorectification

Lorsque l'on mène une campagne de mesures, il arrive que toute les prises de vues de ne soient pas permises. Il peut dès lors être compromis de filmer la partie de la rivière qui nous intéresse avec un angle de 90°. Cependant, il est essentiel que les images analysées soient remises perpendiculaires à l'angle de vue. En effet, l'analyse PIV mesurant la distance entre les positions d'un groupe de particules sur deux images différentes de la vidéo, il faut être capable de mesurer ces distances avec précision. Or, si les images sont prises d'une manière différente qu'à la verticale, il est impossible d'obtenir une distance qui reflète la distance réelle. De fait, parmi l'ensemble des transformations géométriques existantes, la transformation la plus courante dans les situations rencontrées est l'homographie. Celle-ci a pour particularité de ne garder que les lignes droites, ce qui implique donc que les distances au sein de l'image modifiée ne correspondent plus à celles de l'image originale[4, 25, 29].

Le phénomène d'orthorectification permet cette remise à niveau à l'aide d'une homographie. L'objectif est donc de trouver, via une transformation H, la position du pixel (x', y') dans le repère orthogonal à la prise de vue à partir de sa position (x, y) dans le repère initial. Ceci peut également s'écrire comme :

$$(x', y') = H(x, y) \tag{2.1}$$

Cette transformation peut alors être réécrite sous forme matricielle :

$$\begin{bmatrix} ax' \\ ay' \\ a \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2.2}$$

Ce qui donne après linéarisation :

$$\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y & -x' \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y & -y' \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{2.3}$$

La résolution du système permet alors de trouver les coordonnées du pixel dans le repère final. Il convient évidemment de résoudre ce système pour les quatre points fixes qui délimiteront la région à orthorectifier et qui seront alors placés de façon stricte (Figure 2.1), et où leurs coordonnées pourront êtres trouvées grâce aux équations suivantes 2.4 :

$$\begin{cases} L_1^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2 \\ L_2^2 = (x_3 - x_2)^2 + (y_3 - y_2)^2 \\ L_3^2 = (x_4 - x_3)^2 + (y_4 - y_3)^2 \\ L_4^2 = (x_4 - x_1)^2 + (y_4 - y_1)^2 \\ L_5^2 = (x_3 - x_1)^2 + (y_3 - y_1)^2 \\ L_6^2 = (x_4 - x_2)^2 + (y_4 - y_2)^2 \end{cases} \tag{2.4}$$

On se retrouve donc avec huit inconnues pour six équations dont une est redondante. Il faut donc réduire le nombre d'inconnues au nombre de cinq. Pour ce faire, le point 4 est considéré comme l'origine et ses coordonnées sont alors simplement $(x_4, y_4) = (0, 0)$. Ceci nous amène alors à six inconnues. Afin de retirer encore une inconnue et donc une coordonnée, le point 1 est fixé verticalement au point 4 et donc $x_1 = 0$.

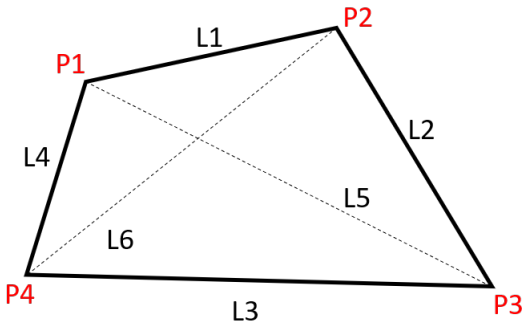


FIGURE 2.1 – Placement des GCPs

2.3 Analyse PIV

La librairie qui effectue l'analyse PIV et la validation s'appelle *OpenPIV*.

2.3.1 Création du champ de vecteurs

L'analyse PIV est le coeur de l'application, comparant chaque paire d'images et générant le champ de vecteurs utile au calcul du débit. La méthode d'analyse statistique utilisée ici et décrite dans les lignes qui suivent s'appelle la corrélation croisée.

Une grille de calcul, que nous appellerons grille d'interrogation est apposée sur l'image 1 de la figure 2.2. Sur chacun des noeuds de cette grille sera calculé un vecteur de déplacement. L'espacement de cette grille est appelée *Overlap* dans l'application.

La suite du processus nécessite de définir deux notions, deux zones carrées qui seront définies dans l'application par la longueur de leurs côtés :

- **L'aire d'interrogation (AI) :** L'aire d'interrogation est un carré extrait **de l'image 1** et centré sur les noeuds de la grille d'interrogation.
- **L'aire de recherche (AR) :** L'aire de recherche est une zone **de l'image 2**, également centrée sur les noeuds de la grille, qui sera parcourue par le centre de l'aire d'interrogation.

Dans l'application, ces deux paramètres ont la même valeur, ce qui signifie en réalité que l'aire inspectée par l'algorithme sur l'image 2 est quatre fois plus grande que ces zones. Cela se comprend en observant la figure 2.2 : quand le centre de la zone de recherche parcourt dans le carré rouge intérieur, la zone analysée correspond en fait au carré extérieur quatre fois plus grand.

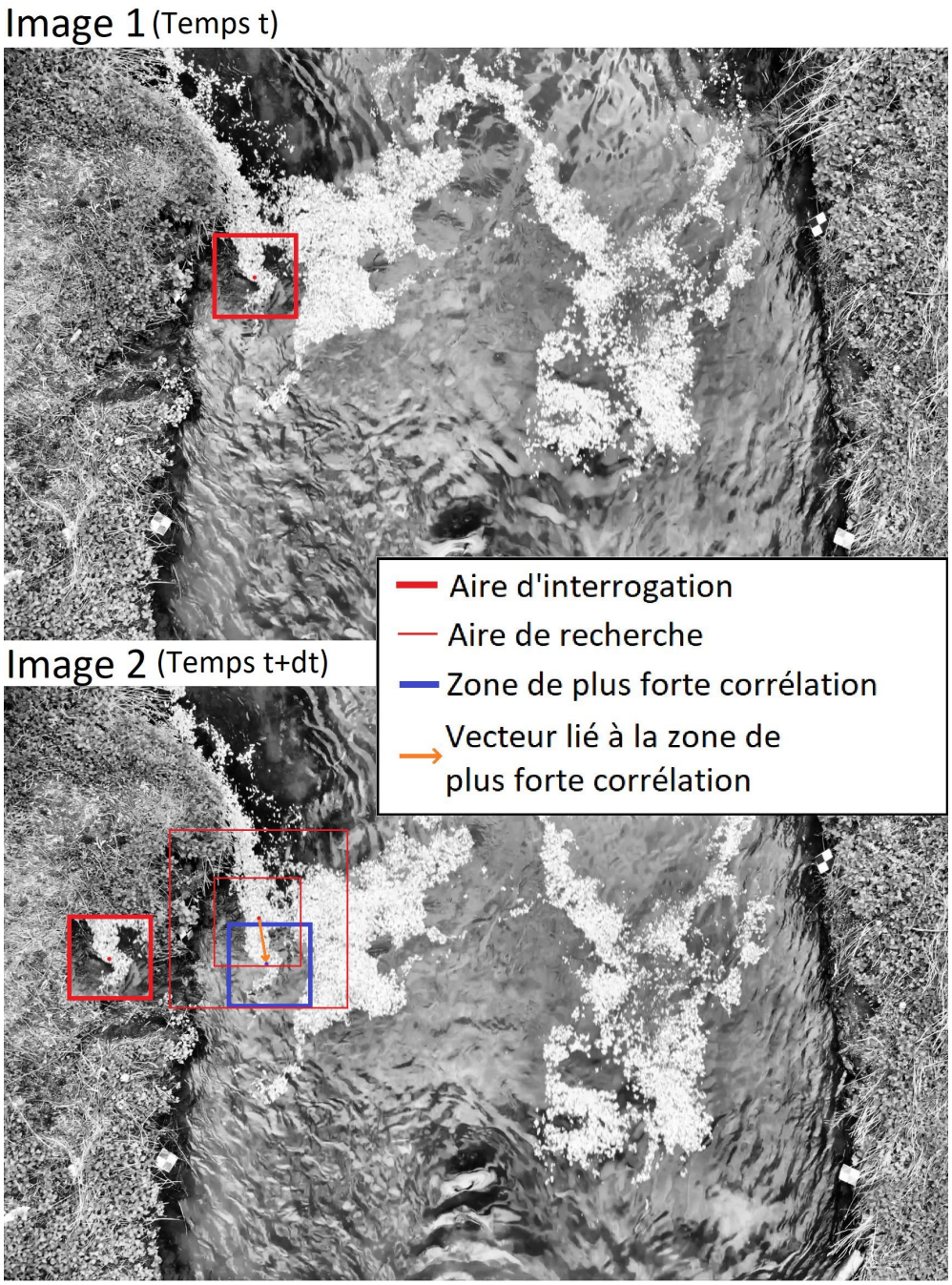


FIGURE 2.2 – Illustration de la création d'un vecteur par l'analyse PIV. L'aire de recherche a été copiée sur la berge de la deuxième image afin de mieux percevoir les similitudes. De même, pour plus de visibilité, les images ont été extraites à plusieurs secondes d'écart de la vidéo, et non quelques frames comme c'est habituellement le cas dans l'application.

L'application compare donc l'aire d'interrogation avec toutes les portions de l'aire de recherche étendue et établit pour chaque comparaison un coefficient de corrélation, que l'on peut considérer comme un indicateur de ressemblance entre les deux zones testées. Une transformée de Fourier rapide (FFT) est utilisée à ce stade pour réduire le temps de calcul. Après avoir effectué cette opération sur toute la zone de recherche, l'algorithme dispose d'une matrice des coefficients de corrélation. Dans la littérature, cette matrice est souvent représentée en 3D (Figure 2.3).

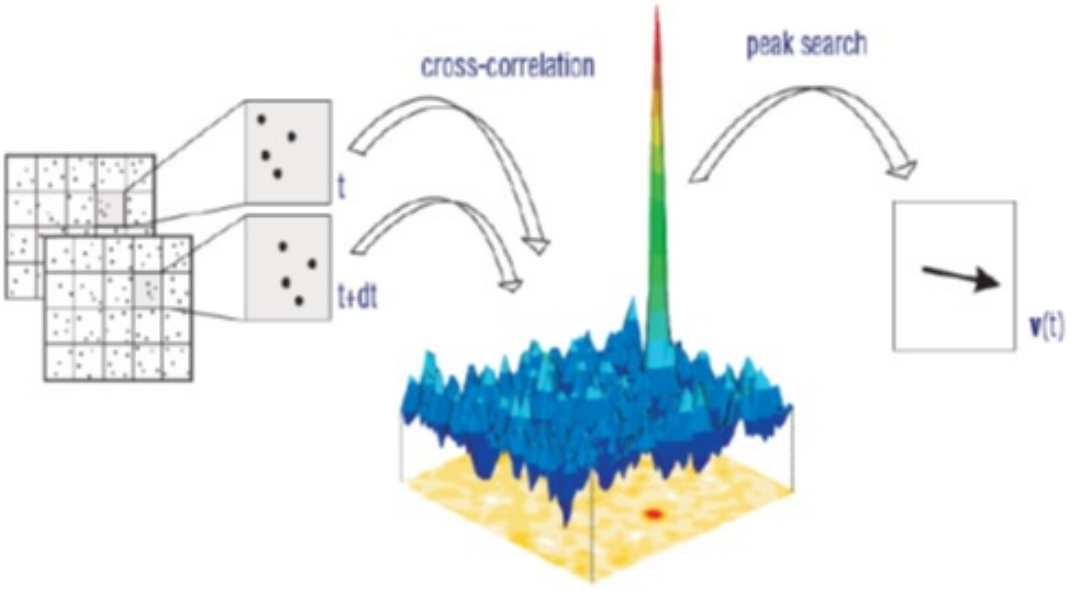


FIGURE 2.3 – Schématisation de l'analyse effectuée pour trouver un vecteur. Image issue de [5].

Un vecteur est alors créé entre le noeud de la grille d'interrogation considéré et la position du coefficient maximum de cette matrice. L'algorithme passe alors à la zone centrée sur le noeud suivant, et décale donc de la distance entrée dans le champ *Overlap*.

2.3.2 Validation

L'étape de validation permet d'écarter les vecteurs considérés comme trop peu fiables et de les remplacer par une moyenne des vecteurs environnants.

Dans l'application, le processus de validation passe par l'établissement d'une mesure - le rapport signal/bruit, évaluant la fiabilité d'un vecteur - et l'application d'un seuil en dessous duquel ledit vecteur est considéré comme erroné et donc remplacé.

Il existe d'autres méthodes de validation, établissant les valeurs fausses via des écarts à la moyenne ou des intervalles de vitesses. Toutefois, dans leur mémoire, Charlotte Everaert et Edouard de Grand Ry établissent, après avoir testé ces différentes approches :"[...]nous observons à nouveau de meilleurs résultats lorsque la méthode de validation `sig2noise_val()` est utilisée. [...] c'est cette méthode qui restera implémentée dans l'application"[11].

Le rapport signal/bruit que nous appellerons ici SNR est une valeur associée à chaque vecteur calculé lors de l'analyse PIV[44]. Il est calculé sur base de la matrice de corrélation établie pour chaque vecteur et peut être évalué via deux méthodes :

- Peak to peak : Le SNR est ici calculé comme le rapport de la taille du pic correspondant au point choisi sur la taille du deuxième pic le plus grand.
- Peak to mean : Le SNR est pris comme le rapport de la taille du pic correspondant au point choisi sur la moyenne de toutes les autres valeurs de la matrice de corrélation (voir figure 2.4).

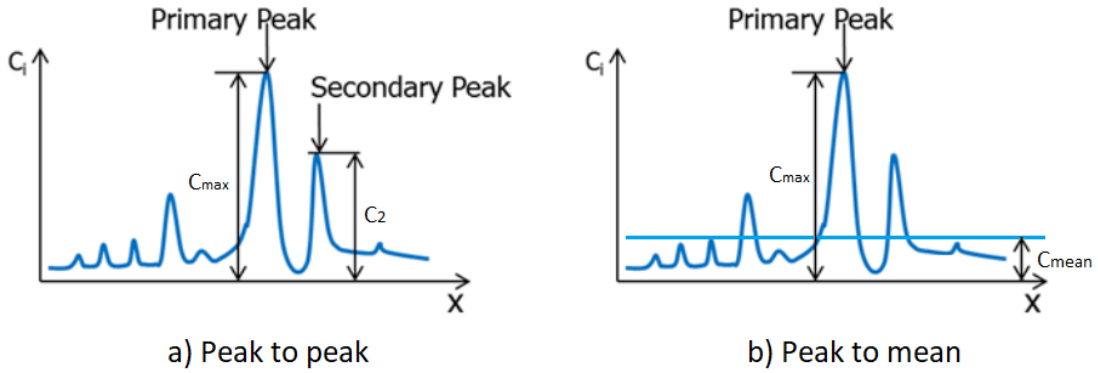


FIGURE 2.4 – Illustration des méthodes de calcul du SNR. La matrice est ici représentée en 2D. Image adaptée de [71]

On comprend bien que les 2 méthodes ne donnent pas les mêmes résultats. Les SNR attendus via "Peak2peak" sont plus faibles, de par la nature de la méthode de calcul, que via "Peak2mean". Dans notre cas, nous avons préféré l'approche "Peak2mean" qui limite l'influence du seuil introduit dans l'application.

Moyenne des vitesses

Une fois que cette opération a été effectuée entre toutes les paires d'images successives, une moyenne est effectuée sur tous les vecteurs trouvés au même endroit entre des images différentes, permettant de créer un champ de vecteurs moyens.

Bia T. et Somers H.[23] évoque le fait qu'effectuer cette moyenne sur les matrices obtenues à la place des vecteurs pourrait donner de meilleurs résultats.

D'autre part, Firas Ghali a étudié le fait d'utiliser la médiane des valeurs des vecteurs vitesse obtenu au lieu de la moyenne. Si nous n'avons pas eu accès à ses résultats détaillés, les analyses préliminaires semblaient encourageantes.

Ces différents façons de synthétiser la somme des valeurs obtenus pour une vidéo en une seule image sont intéressantes et mériteraient qu'un travail ultérieur s'y attarde.

2.4 Coefficient de surface

Dans la dernière fenêtre avant d'accéder au débit final, l'application requiert d'introduire le coefficient de surface. Ce coefficient sert à lier la vitesse moyenne dans la section et la vitesse de surface. En effet, la vitesse de l'écoulement n'est pas uniforme le long de la hauteur d'eau. Près du fond, en raison des forces de frottement, la vitesse est nulle et augmente avec la hauteur jusqu'à atteindre un maximum en surface (Figure 2.5).

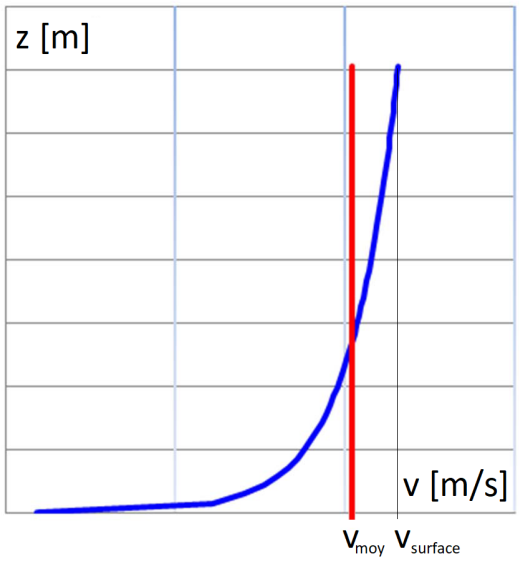


FIGURE 2.5 – Illustration du profil de vitesse dans une section. Image adaptée du cours LGCIV2053 - Hydraulique Fluviale.

Les vitesses trouvées doivent donc être pondérées par un paramètre oscillant entre 0.8 et 0.9. Durant nos expérimentations, ce paramètre a été fixé à 0.85.

2.5 Exploitation du champ de vecteurs

Après que l'utilisateur ait choisi les points de la section où il désire que le débit soit calculé et le nombre de sous-sections n, l'application sélectionne n points de la grille parmi les plus proches de la section introduite. Les vecteurs moyens associés à ces points sont alors multipliés par un facteur nommé le *facteur d'échelle*. Ce facteur d'échelle permet de passer de vitesses en $[px/s]$ à des vitesses exprimées en $[m/s]$ utilisables dans le calcul du débit.

Le facteur d'échelle est calculé lors de l'étape d'orthorectification où l'algorithme dispose de la distance en pixels entre les points et les distances réelles associées introduites par l'utilisateur. Si cette étape est ignorée, le facteur d'échelle est calculé sur base de la distance entre les points de la section et les distances réelles disponibles dans la bathymétrie.

L'aire de chaque sous-section est ensuite calculée via la règle trapézoïdale comme :

$$A_i = \frac{H_i + H_{i+1}}{2}.L_i[m^2] \tag{2.5}$$

où A_i est l'aire de la section i, H_i et H_{i+1} les hauteurs des verticales entourant la sous-section et L_i sa largeur.

Le débit total est enfin calculé comme la somme des débits de chaque sous-section :

$$Q_{tot} = \sum_{i=n}^{n} Q_i = \sum_{i=n}^{n} A_i.V_i[m^3/s] \tag{2.6}$$

où V_i est la composante de la vitesse i perpendiculaire à la section introduite.

Deuxième partie

État de l'application

Chapitre 3

Structure de l'application et guide utilisateur

Les prochaines lignes et prochaines pages décriront la structure de l'application et son utilisation telle que nous l'avons reçue. Bien que certains défauts aient été trouvés lors de nos premières utilisations, que certaines fonctionnalités aient été notées comme absentes ou améliorables du point de vue ergonomique, esthétique, etc, ou encore que certaines lignes, packages ont été ajoutés ou supprimés, la structure et le schéma de l'application sont restés quasi-intactes. Cette partie se veut donc clarificatrice pour l'utilisateur quant à l'utilisation de l'application.

3.1 Structure de l'application et informations générales

L'application consiste en une série de lignes de codes, réparties en plusieurs fonctions qui elles-mêmes sont stockées dans plusieurs fichiers. Le langage choisi pour le développement de l'application est le langage ***Python***. De notre côté, et comme nos prédécesseurs, nous utilisons l'environnement de développement ***Spyder*** disponible sur le navigateur ***Anaconda***, pour lancer et exécuter l'application.

Afin de comprendre au mieux les différentes fonctionnalités de l'application, un flowchart est représenté en figure 3.1 décrivant son fonctionnement.

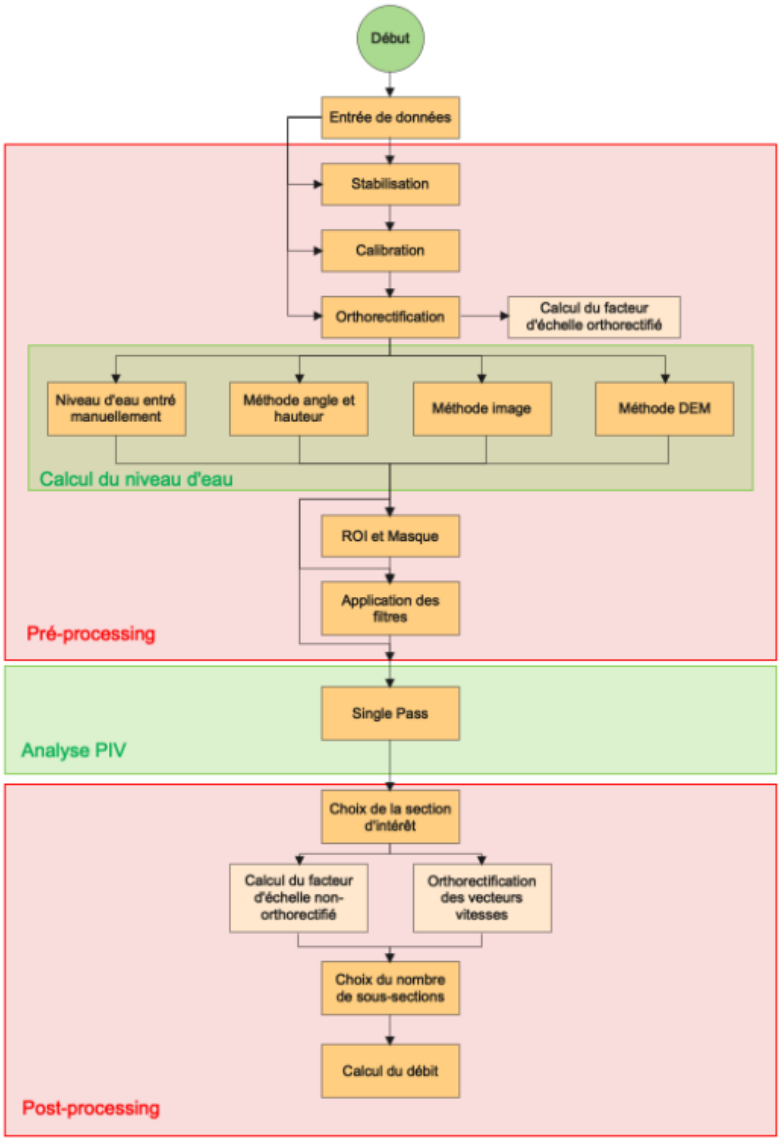


FIGURE 3.1 – Flowchart de l'application [11]

On peut y apercevoir les étapes clés de l'utilisation de l'application à savoir le pré-processing, l'analyse PIV et le post-processing. Ces étapes consistent respectivement en l'encodage des données afin de rendre la lecture de la vidéo possible et plus claire, l'analyse PIV de la vidéo selon les paramètres fixés et le calcul final du débit sur la section sélectionnée.
En plus de ces trois grandes étapes, on peut également voir ressortir le plus gros challenge de l'application : le calcul du niveau d'eau. En effet, on voit qu'il existe plusieurs méthodes de mesure du niveau d'eau dont certaines ne sont pas encore au point voire ne fonctionnent tout simplement pas.

3.2 Un guide utilisateur

3.2.1 Récupération du projet

Avant toutes choses, il paraît évident de récupérer le code source de l'application. Pour ce faire, aller sur `https://forge.uclouvain.be/hydraulics-simulations/riverapp` et télécharger le dossier en `.zip` comme montré à la figure 3.2.

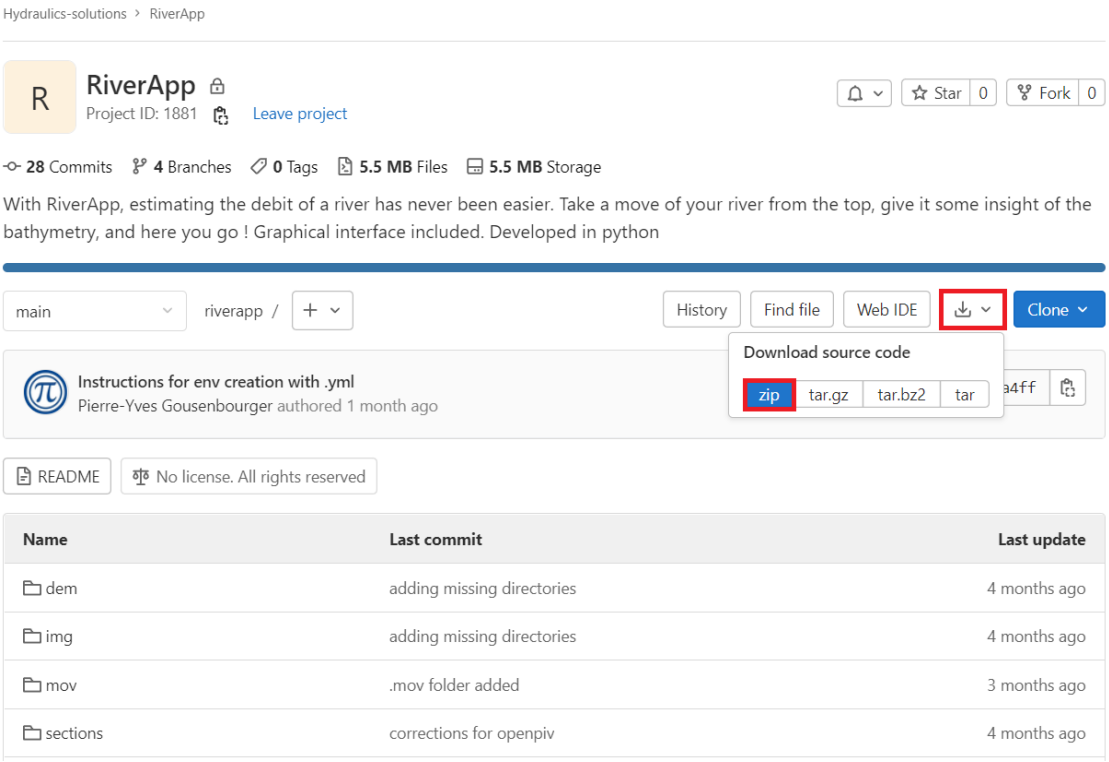


FIGURE 3.2 – Téléchargement du dossier contenant le code source sur *Forge*

Positionner et extraire le dossier à l'endroit souhaité. Ensuite, renommer celui-ci ***riverapp***.

3.2.2 Vérification du contenu du dossier

Une fois le dossier extrait, vérifier qu'aucun élément n'est manquant et que les dossiers imbriqués sont bien présents. Le nombre d'éléments dans le dossier devrait s'élever à 24 et les dossiers suivants devraient apparaître : ***img***, ***mov***, ***dem*** et ***sections*** lui-même composé des dossiers ***files*** et ***img***. Ces dossiers contiendront respectivement les images affichées dans l'application, les vidéos utilisées, des fichiers `.tiff` obligatoires pour une des méthodes d'obtention du niveau d'eau et tout ce qui est nécessaire pour les bathymétries : les images dans le dossier ***img*** et les fichiers `.txt` contenant les bathymétries dans le dossier ***files***. Une fois

l'organisation terminée, l'agencement du dossier de l'application devrait ressembler aux figures 3.3 et 3.4 :

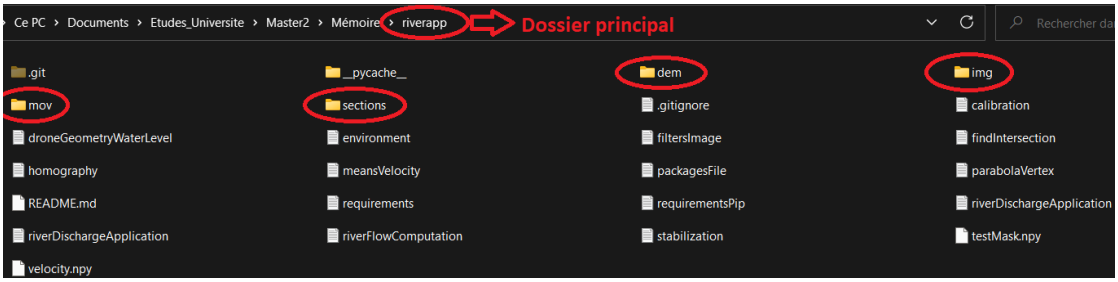


FIGURE 3.3 – Organisation des fichiers et dossiers de l'application

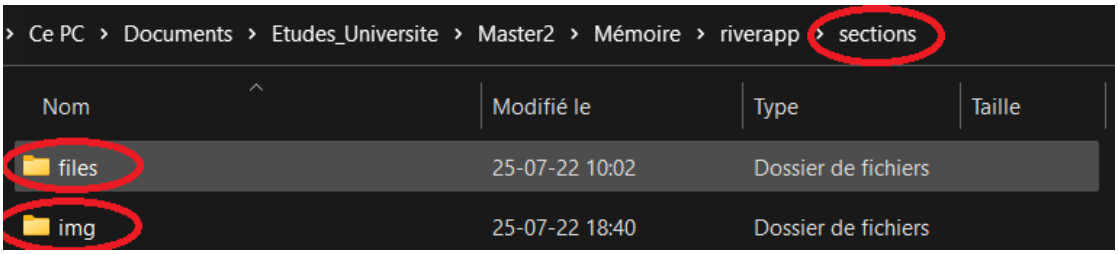


FIGURE 3.4 – Organisation des dossier dans le dossier sections

3.2.3 Un IDE *Python*

Afin de pouvoir lancer l'application, il est essentiel de disposer d'un IDE[1] ***Python***. ***Spyder*** est l'IDE qui a été utilisé pour la conception de l'application mais de nombreux logiciels sont utilisables. En effet, l'IDE utilisé n'est pas important dans le lancement et l'utilisation de l'application.

3.2.4 Un environnement virtuel

L'environnement virtuel, lorsqu'il est activé, permet d'utiliser une version du langage et des paquets bien précise. De cette manière, un utilisateur peut utiliser une version bien particulière d'un package dans un projet tout en gardant la dernière version de ce même package pour un autre projet.

Installer l'environnement virtuel lié à l'application, soit en utilisant le fichier ***environment.yml*** et en suivant la démarche décrite ici https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html [1], soit en utilisant le fichier ***requirements.txt*** et en suivant la démarche décrite ici https://docs.python.org/fr/3/tutorial/venv.html [54]

1. Un environnement de développement intégré, ou IDE, est un logiciel de création d'applications, qui rassemble des outils de développement fréquemment utilisés dans une seule interface utilisateur graphique (GUI).[31]

Remarque 1. *Il est possible de se passer de cette étape, mais pour une question de gain de temps et pour éviter des problèmes de compilation dus à d'éventuelles versions non compatibles, il est vivement conseillé d'installer l'environnement plutôt que d'installer chaque package et sa version associée un à un.*

Ouvrir le prompt de l'IDE utilisé ou du logiciel qui l'héberge et activer l'environnement en insérant ***conda activate "nom de l'environnement"***.

3.2.5 Données nécessaires à l'utilisation de l'application

L'utilisation de l'application exige de posséder certaines données. Il est donc nécessaire d'être en possession d'une vidéo relativement stable où quatre points fixes sont visibles ainsi que des distances les séparant, de la bathymétrie au droit de la section de mesure et de la profondeur au point le plus bas de la section.

3.2.6 Remarques générales applicables à l'entièreté de l'application

Remarque 2. *Toutes les virgules dans les nombres décimaux devront être encodés à l'aide d'un point.*

Remarque 3. *Lors d'un encodage avec plusieurs valeurs dans une même case, ces valeurs seront obligatoirement séparées par une virgule.*

Remarque 4. *Certaines étapes nécessitent un ordre particulier d'exécution. Il convient donc d'être attentif.*

Remarque 5. *Durant le processus, il est presque toujours possible de revenir en arrière ou d'annuler à l'aide des boutons* ***Go Back*** *et* ***Cancel****.*

3.2.7 Tutoriel de première utilisation

Un kit de première utilisation a été mis en place. Ses objectifs sont doubles. D'une part, vérifier que le programme ne génère pas d'erreurs empêchant le lancement du programme. D'autre part, cela permet à l'utilisateur novice de découvrir les différents panneaux de l'application et de se familiariser avec son fonctionnement via une mesure de bout en bout sur une vidéo de la Dyle à Noirhat.
Ce kit est un dossier, nommé ***RiverApp - Première utilisation***, qui comprend (Figure 3.5) :

- Un document PDF, ***RiverApp - Instructions première utilisation***, détaillant précisément les étapes à suivre pour lancer l'application, effectuer

les mesures ainsi que les résultats attendus à chaque étape. Ce document est présent en annexe .1.
- La bathymétrie sous forme de fichier `.txt`.
- Toutes les données à introduire pendant la mesure reprises dans un fichier `.txt` permettant le copier-coller. Ce fichier se trouve en annexe .2
- La vidéo sur laquelle opérer la mesure.

Tous ces éléments étant repris dans le même dossier, cela permettra de ne pas embrouiller l'utilisateur dans les fichiers à sélectionner.

Remarque 6. *Il convient de respecter la place des fichiers (bathymétrie et vidéo) lors du lancement de l'application (voir section 3.2.2).*

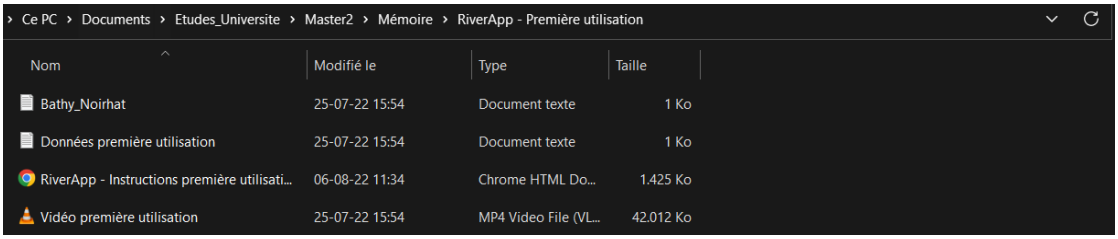


FIGURE 3.5 – Dossier reprenant tous les éléments essentiels à la première utilisation

La différence entre le guide utilisateur et le kit de première utilisation réside dans une explication plus brève et moins précise des étapes dans le cas du kit mais qui accompagne l'utilisateur dans chaque étape pour obtenir les résultats de l'exemple concret, fourni dans le dossier.
De cette manière, à l'aide du kit, l'utilisateur pourra rapidement vérifier que l'application est bien installée sur son ordinateur. Dans le cas du guide utilisateur, il pourra contrôler qu'il entreprend les bonnes actions et ce peu importe le type d'analyse qu'il désire accomplir.

3.2.8 Lancement de l'application

Remarque 7. *C'est votre première utilisation ? Il est vivement conseillé de d'abord effectuer le tutoriel de première utilisation. Retourner à la section 3.2.7*

Pour lancer l'application, exécuter le programme python contenu dans le fichier `riverDischargeApplication.py`, soit via son IDE (s'il en offre la possibilité), soit via le terminal Python. Quelques lignes s'afficheront dans la console avant que la fenêtre pop-up suivante fasse son apparition (Figure 3.6).

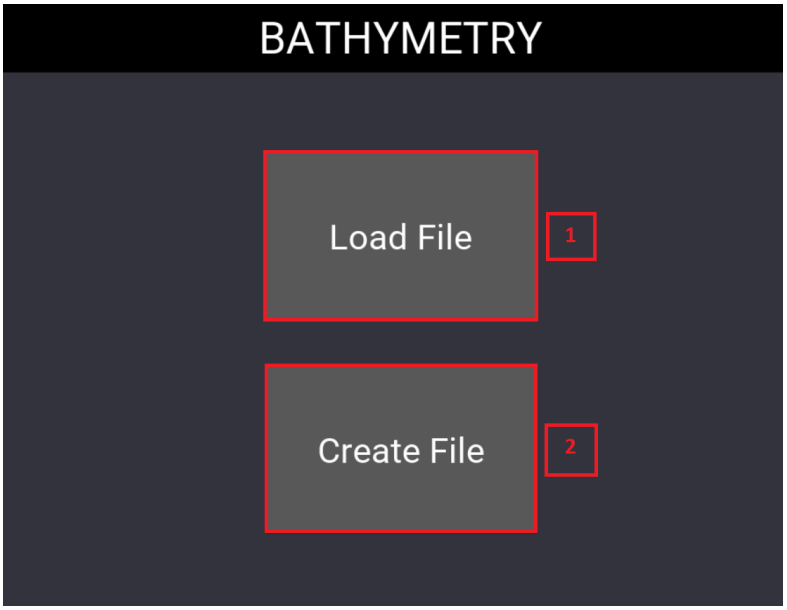


FIGURE 3.6 – Première fenêtre d'apparition

3.2.9 Choix de la bathymétrie

Si le fichier contenant la bathymétrie existe, sélectionner ***1. Load File*** (Figure 3.6).
Si seuls quelques points du profil sont connus ou si le profil respecte une forme géométrique simple, sélectionner ***2. Create File*** (Figure 3.6).
Si aucune des deux options ne correspond, veiller à posséder les éléments nécessaires à l'utilisation de l'application (Section 3.2.5).

1. Load File (Figure 3.6)

Choisir le fichier contenant la bathymétrie et sélectionner ***Load*** (Figure 3.7). Une fois sélectionné, le profil désiré apparaîtra (Figure 3.8).

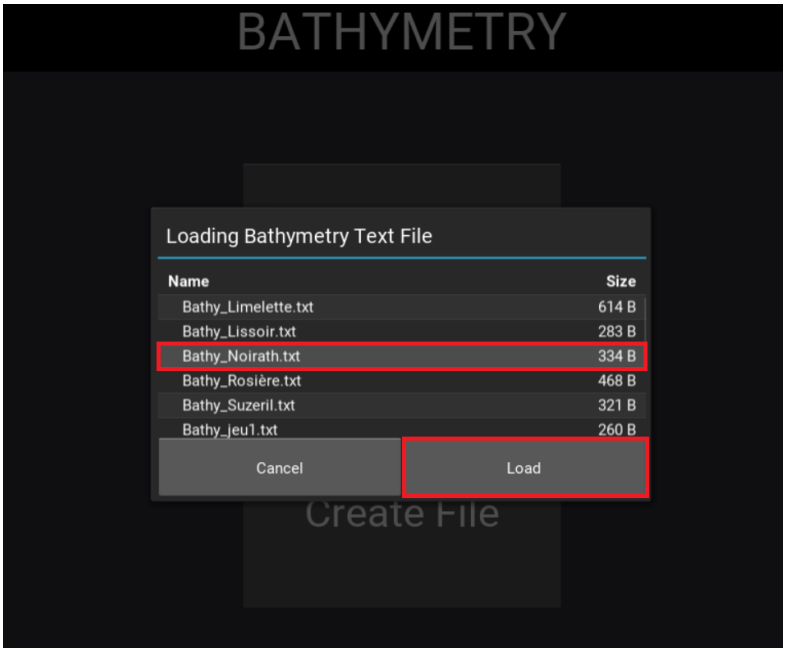


FIGURE 3.7 – Choix de la bathymétrie

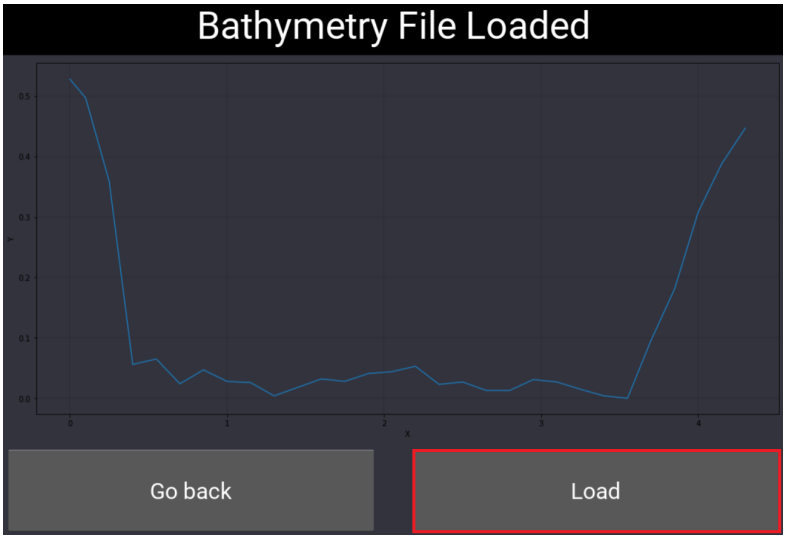


FIGURE 3.8 – Affichage du profil du cours d'eau choisi

Si l'image correspond au profil attendu, sélectionner ***Load*** (Figure 3.8).

Remarque 8. *Les coordonnées de la bathymétrie doivent être encodées dans un fichier* ***.txt*** *où chaque ligne contient deux nombres décimaux (correspondant aux abscisses et aux ordonnées) séparés par une virgule. L'origine des abscisses se situe sur la rive gauche (attention au sens de la vidéo, filmée vers l'amont ou l'aval). La première abscisse sera donc égale à* 0*. L'origine des ordonnées, quant à elle, se situera au point le plus profond du profil. (Figure 3.9)*

FIGURE 3.9 – Exemple d'un fichier contenant la bathymétrie

Remarque 9. *Il convient de vérifier que la bathymétrie est placée dans le bon dossier (****sections*** → ***files****) (Section 3.2.2).*

2. Create File (Figure 3.6)

Choisir le profil qui se rapproche le plus du profil réel. Les choix possibles sont un profil rectangulaire, trapézoïdal, parabolique ou libre. Le profil libre correspond au cas où les coordonnées de plusieurs points sont connues (Figure 3.10).

Si une forme géométrique est choisie, encoder le nom souhaité (***1***) (ce nom sera celui donné au fichier `.txt` qui contiendra la future bathymétrie) (Figure 3.11), les dimensions demandées, en mètre, (***2***) ainsi que les coordonnées du point rouge (***3***). Cliquer ensuite sur ***Load*** pour continuer. Un fichier `.txt` contenant la nouvelle bathymétrie sera immédiatement créé dans le dossier pourvu à cet effet.

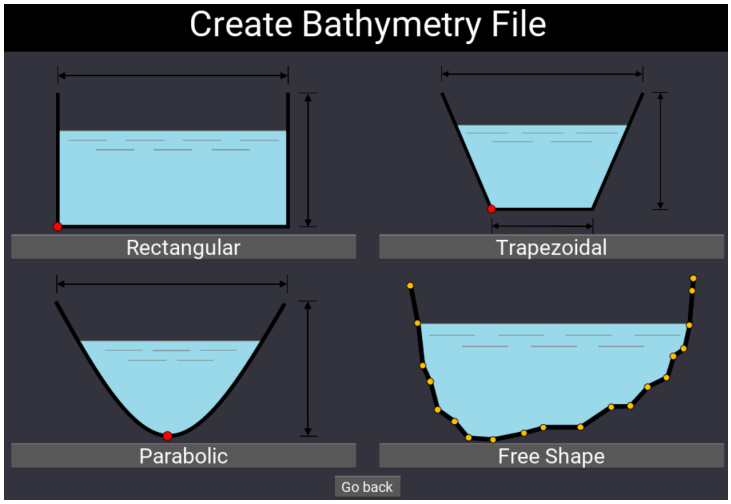


FIGURE 3.10 – Choix de la forme géométrique la plus proche du profil réel

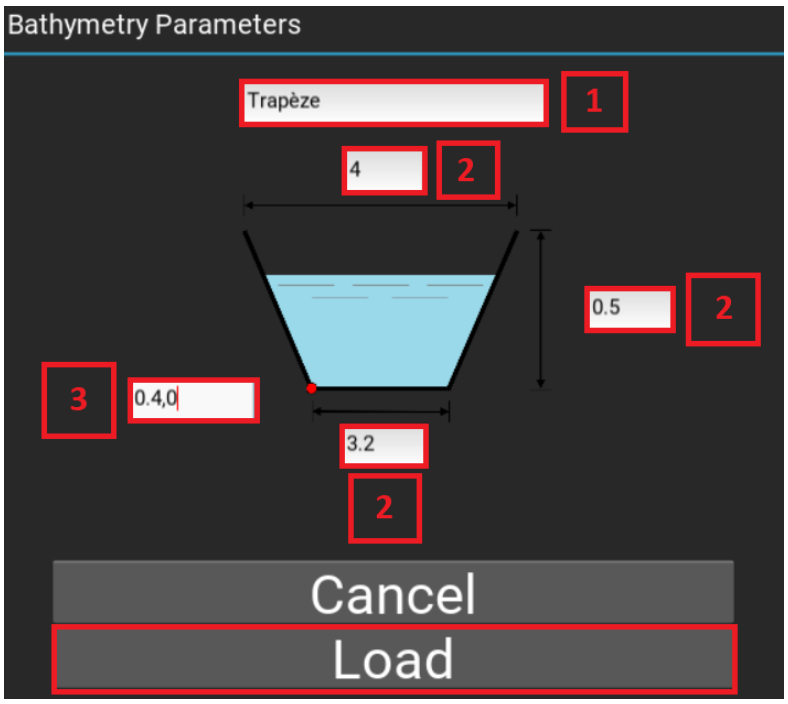


FIGURE 3.11 – Exemple d'encodage avec une forme trapézoïdale

Pour visualiser le profil créé, appuyer sur ***Go Back*** lors de l'étape suivante (Figure 3.13) et refaire la procédure de la bathymétrie en optant cette fois-ci pour ***1. Load File*** (Figure 3.6) et chercher le fichier `Trapèze.txt` lors du choix de la bathymétrie (Figure 3.7). Le profil montré à la figure 3.12 devrait apparaître.

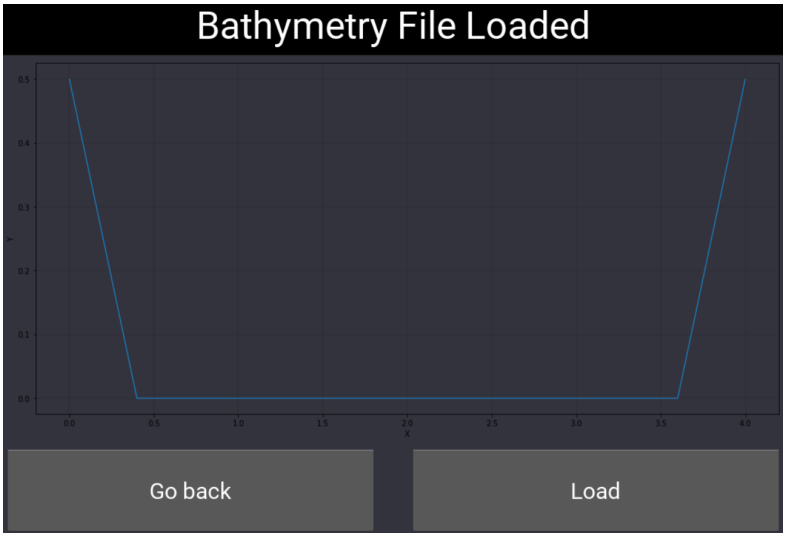


FIGURE 3.12 – Profil trapézoïdale suite à la commande ***Create File***

Dans le cas où le profil serait plus aléatoire mais que plusieurs coordonnées sont connues, opter pour ***Free Shape*** (Figure 3.10).
Encoder alors les différentes abscisses et ordonnées dans leurs cases respectives selon le format rappelé aux remarques 2 et 3.

3.2.10 Choix de la vidéo

Une fois le profil du cours d'eau chargé, l'écran représenté à la figure 3.13 devrait apparaître. Cliquer alors sur la vidéo désirée dans le menu déroulant.

Remarque 10. *On peut vérifier que c'est bien la bonne vidéo en vérifiant le temps de vidéo qui apparaît dans la case inférieure droite. (Figure 3.13)*

Une fois la vidéo choisie et afin d'accélérer l'analyse PIV, il convient de cibler la partie intéressante de la vidéo, aussi appelée ***range***, ainsi que l'écart ou ***step*** entre les images extraites.
Pour ce faire, remplir ***1. Step between frames*** par un nombre entier. Pour des vidéos en 24 ou 30 images par seconde, la valeur standard est 3. Pour les vidéos en 60 images par seconde, il est recommandé d'augmenter ce paramètre à 5 ou 6 sous peine d'augmenter le temps de calcul.
Compléter également les cases ***2. Range of video in seconds from*** et ***3. to*** respectivement avec la seconde par laquelle l'étude de la vidéo commencera et celle par laquelle elle se terminera. Il convient de choisir le moment où les traceurs commencent à **quitter** la zone d'intérêt afin que ladite zone soit remplie par les traceurs durant tout l'intervalle choisi. Veiller également à choisir un intervalle contenant le moins de zones dénuées de traceurs possible.(Figure 3.14)

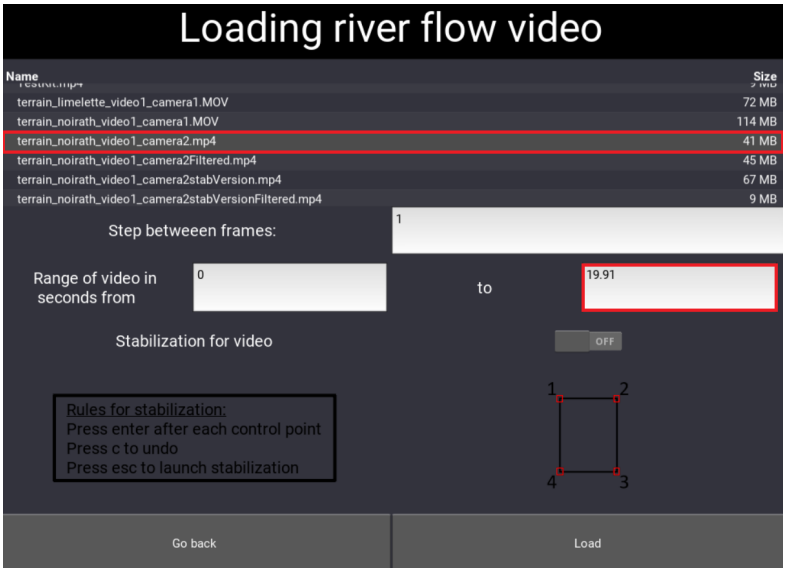


FIGURE 3.13 – Choix de la vidéo

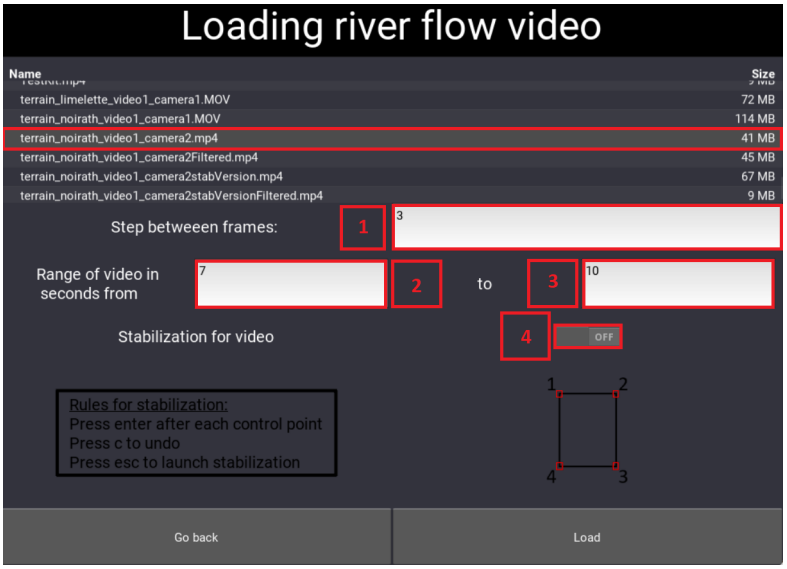


FIGURE 3.14 – Modification des paramètres d'analyses

Remarque 11. *Augmenter sans raison le* ***step*** *conduira à des erreurs de précision.*

Remarque 12. *Si aucun paramètre n'est modifié, la vidéo commencera, de facto, à la seconde* 0 *et se terminera à la fin de la vidéo avec* 1 *comme valeur du step. En somme, l'analyse PIV se produira sur l'entièreté de la vidéo et sur l'entièreté des images ce qui prendra beaucoup de temps.*

Avant d'appuyer sur ***Load*** pour poursuivre, il convient d'activer ou non la stabilisation (Figure 3.14).
Si une stabilisation n'est pas nécessaire, laisser l'option ***4. Stabilization for video*** sur ***Off***.
Si, au contraire, une stabilisation est requise, tourner l'option ***4. Stabilization for video*** sur ***On*** et suivre les étapes de la stabilisation (Section 3.2.11).

3.2.11 Stabilisation

Voici la première étape du Pre-processing. Pour rappel, c'est maintenant que les paramètres référents à l'analyse (stabilisation, calibrage, orthorectification) vont pouvoir être modifiés ou non.

Stabiliser une vidéo est utile lorsqu'il y a une volonté d'effacer les mouvements parasites, qui pourraient fausser les résultats, produits lors de la prise de la vidéo. En effet, il est impossible que la personne ou le drone ayant filmé soit resté parfaitement immobiles lors de l'enregistrement.
Cependant, cette stabilisation n'est pas obligatoire. En effet, si la vidéo est prise depuis un trépied, il n'est pas nécessaire de procéder à une stabilisation. De plus, il faut savoir que la stabilisation crée une nouvelle vidéo et donc réduira la résolution de l'image. Entreprendre une stabilisation sur une vidéo stabilisée usera donc du temps précieux et de la mémoire sans raison.

Si une stabilisation est souhaitée, mettre au préalable l'option ***Stabilization for video*** sur ***On*** et finalement sélectionner ***Load*** (Figure 3.15).

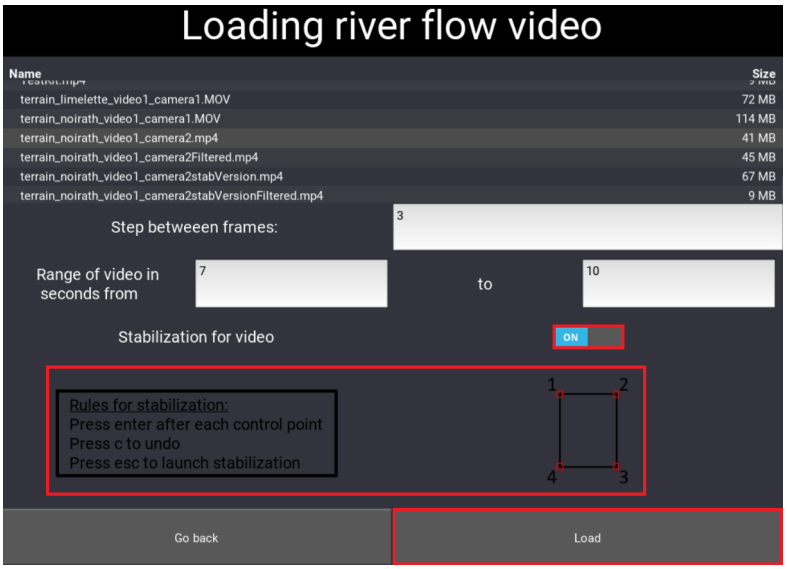


FIGURE 3.15 – Ajout de la stabilisation à l'analyse

Remarque 13. *Les règles encadrées auront toute leur importance pour la suite. Il convient de les retenir pour pouvoir les appliquer.*

En appuyant sur ***Load***, une nouvelle fenêtre contenant une image de la vidéo devrait s'ouvrir (Figure 3.16).
Sélectionner alors quatre points visibles tout au long de la vidéo et qui n'ont pas bougé lors de la vidéo (des pierres, de la végétation s'il n'y a pas de vent,etc.). Pour ce faire, effectuer un mouvement diagonal avec la souris de sorte qu'un carré bleu apparaisse (Figure 3.16). Une fois satisfait de la sélection, relâcher la souris et appuyer sur la touche ***ENTER*** du clavier et passer au point suivant.

FIGURE 3.16 – Sélection d'un point fixe

Remarque 14. *A chaque nouvelle sélection, la précédente s'efface. C'est normal !*

Une fois les quatre points sélectionnés, appuyer sur la touche ***ESC*** du clavier pour lancer la stabilisation.

Remarque 15. *Les quatre sélections réapparaîtront alors en vert et la vidéo se mettra alors en route. Ces carrés de couleurs resteront également présents pour le reste de l'analyse.*

Une fois la stabilisation terminée, une version stabilisée de la vidéo sera créée et enregistrée dans le dossier ***mov***. Une redirection vers la calibration devrait alors avoir lieu (Figure 3.17).

3.2.12 Calibration

La calibration permet de corriger les distorsions occasionnées par la caméra et d'obtenir les paramètres intrinsèques et extrinsèques de la caméra. Il sera donc possible, à ce point-ci, de choisir de calibrer ou non la caméra utilisée lors de l'enregistrement (Figure 3.17).
S'il existe un fichier `.npy` contenant les paramètres de calibration, choisir ***1. Load file*** (Figure 3.17).
Si aucune calibration n'a été effectué auparavant, opter pour ***2. Perform calibration*** (Figure 3.17).
Enfin, si les paramètres spécifiques à une calibration peuvent être extraits depuis la caméra, sélectionner ***3. Compute parameters*** (Figure 3.17).

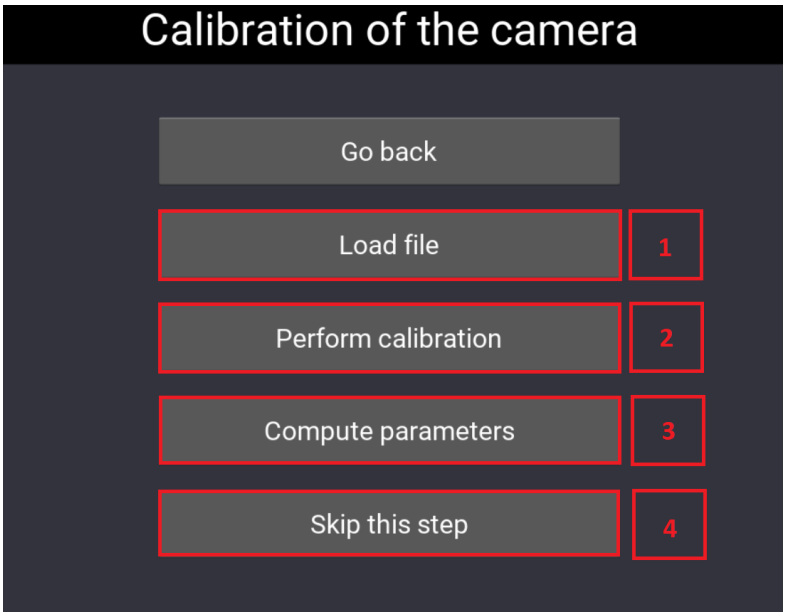


FIGURE 3.17 – Choix du procédé de calibration

1. Load file (Figure 3.17)

Pour appliquer cette calibration, choisir le fichier `.npy`, contenant les informations de calibration, dans le menu déroulant. Ensuite, sélectionner ***Load*** (Figure 3.18).

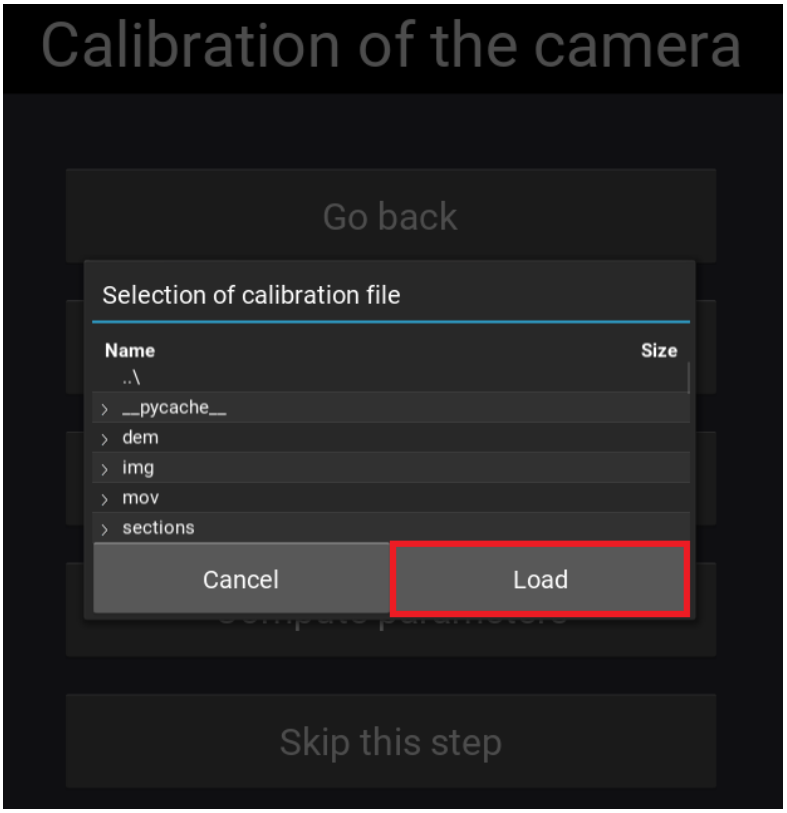


FIGURE 3.18 – Calibration à partir d'un fichier `.npy`

2. Perform calibration (Figure 3.17)

Pour procéder à la calibration, choisir ***1. Select video file for calibration*** et sélectionner la vidéo filmant le damier (Figure 3.19).
Ensuite, écrire à côté de ***2. Name of the file that will contain calibration parameters*** le nom du fichier qui contiendra les paramètres de la calibration. C'est ce fichier qui pourra être réutilisé ultérieurement à l'aide de la méthode de calibration décrite au point ***1. Load File*** (Figure 3.17).
Encoder les différents données relatives au damier comme les dimensions du damier à côté de ***3. Dimensions of the board*** séparées d'une virgule, le nombre d'images extraites à côté de ***4. Number of image you want to extract*** sur lesquelles sera procédée la calibration et finalement la taille des cases du damier en millimètres à côté de ***5. Size of the square of the checkerboard in mm***.
Une fois la fenêtre complétée, sélectionner ***Load*** (Figure 3.19).

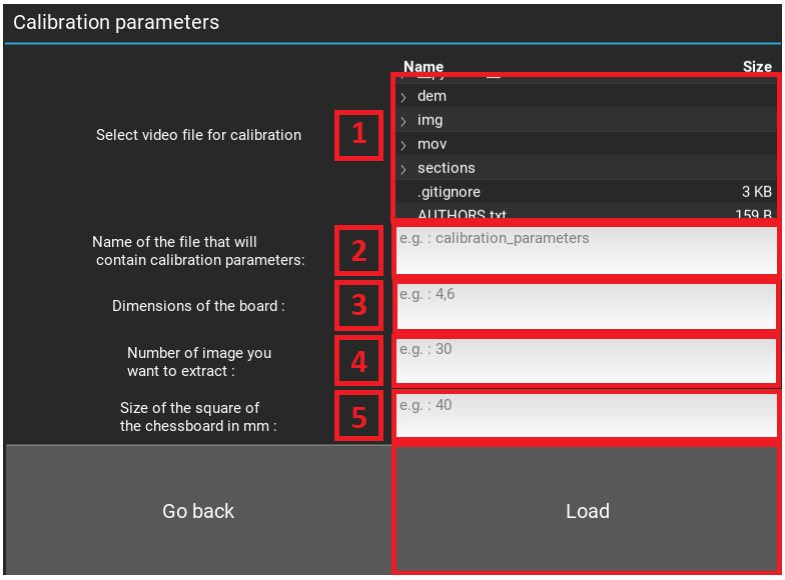


FIGURE 3.19 – Calibration à partir d'une vidéo d'un damier

3. Compute parameters (Figure 3.17)

Compléter les paramètres de la fenêtre à l'aide des informations trouvées dans la caméra (Figure 3.20).

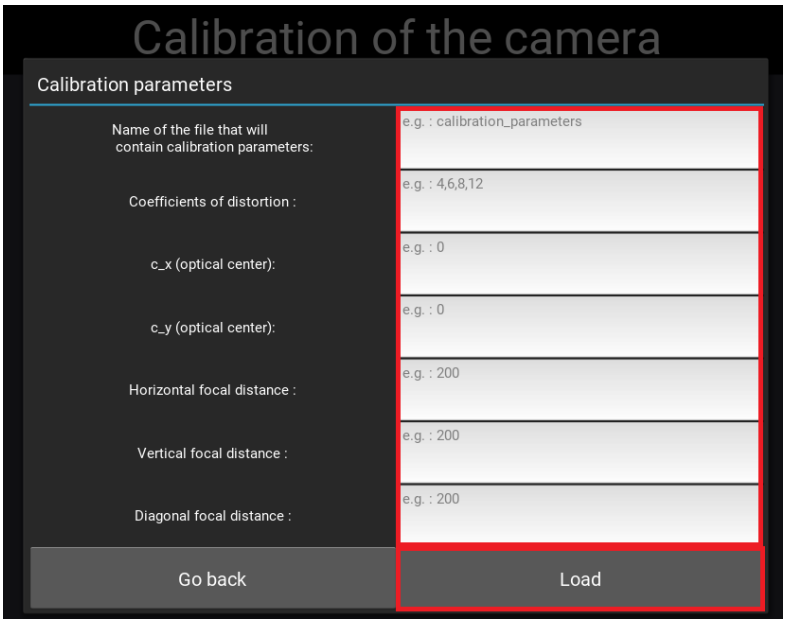


FIGURE 3.20 – Calibration à partir des paramètres de la caméra

4. Skip this step (Figure 3.17)

Étant donnée que cette étape n'est pas obligatoire, il est tout à fait possible

d'ignorer celle-ci. Sélectionner alors ***4. Skip the step*** (Figure 3.17).

3.2.13 Orthorectification

Pour procéder à l'analyse PIV correctement, l'application a besoin d'une image prise avec un angle de vue de 90°. Si une vidéo filmée depuis un drone respecte cette condition, ce n'est pas le cas de toutes les prises de vues. En effet, la plupart des vidéos soumises à l'analyse sont souvent prises d'une berge ou d'un pont.
L'orthorectification permet de redresser l'angle de prise de la vidéo vers un angle d'incidence de 90°.

Afin d'accomplir cette tâche, l'application a besoin de quatre points de référence fixes (souvent les mêmes que ceux sélectionnés lors de la stabilisation (Section 3.2.11)) ainsi que les distances qui séparent chaque point des trois autres.
Dès lors, sélectionner ***Select Ref Points*** sur la fenêtre d'orthorectification (Figure 3.21).

Cliquer ensuite, à l'aide de la souris, sur les quatre points qui seront associés aux points repères.
Une fois la sélection terminée, choisir ***Register*** (Figure 3.21).

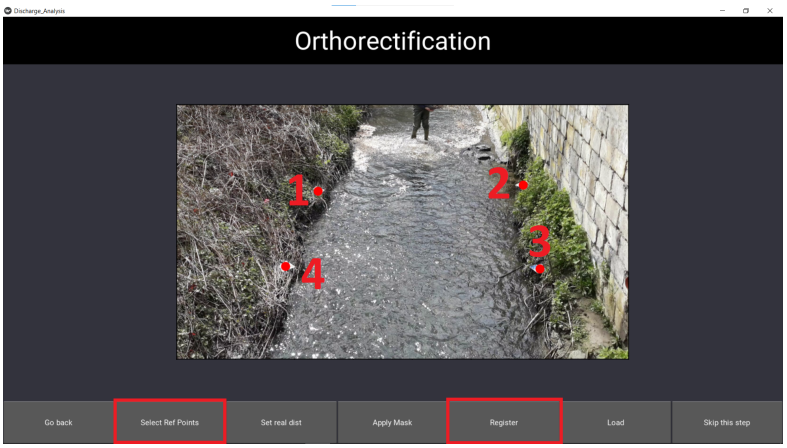


FIGURE 3.21 – Sélection des points de référence

Remarque 16. *L'ordre de sélection a son importance. Il faudra donc respecter l'ordre suivant : En haut à gauche, en haut à droite, en bas à droite, en bas à gauche.*

Remarque 17. *Lors de la sélection, les points apparaissent en rouge. Lorsque* ***Register*** *aura été sélectionné, ces points passeront au bleu (Figure 3.22).*

Ensuite et afin d'encoder les distances séparant les quatre points choisis. Sélectionner ***Set real dist*** (Figure 3.22).

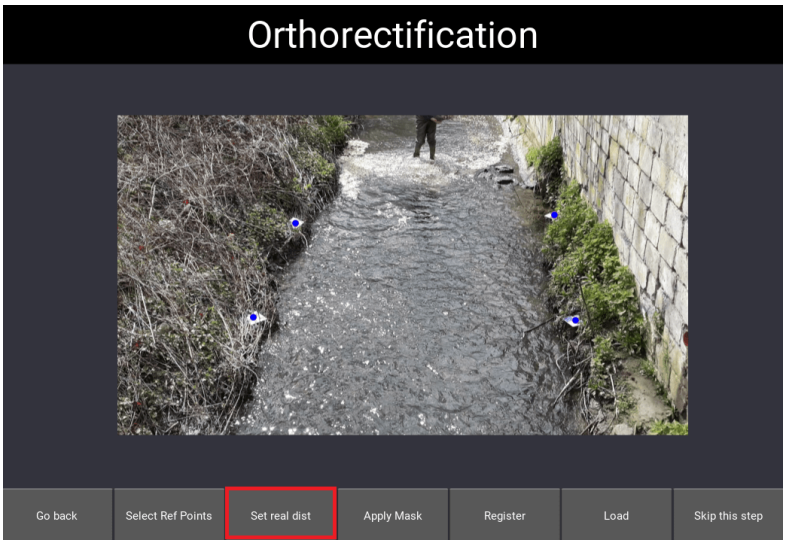


FIGURE 3.22 – Choix d'encodage des distances

Une fenêtre pop-up devrait faire son apparition afin de remplir l'unique case avec les distances préalablement mesurées sur le terrain.
Encoder les distances et sélectionner ***Load*** pour les enregistrer (Figure 3.23).

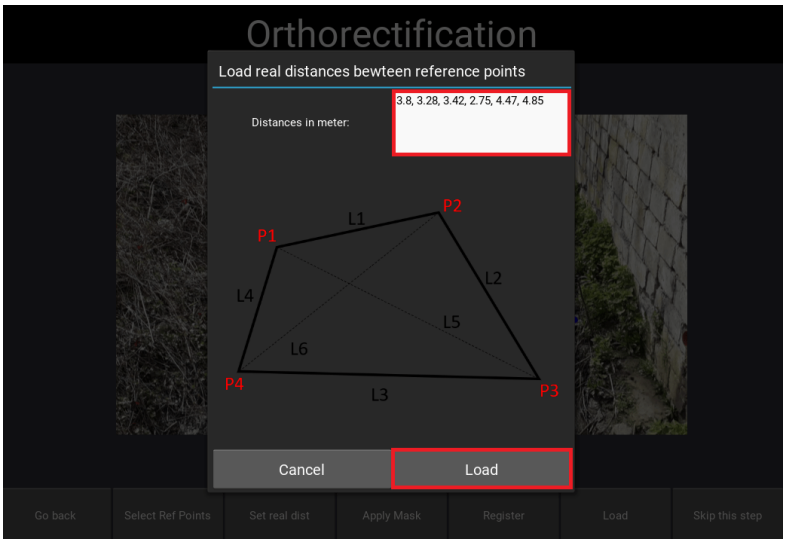


FIGURE 3.23 – Encodage des distances séparant les points

Remarque 18. *Comme pour la remarque 16, l'ordre d'encodage a toujours son importance. Il convient de remplir la case comme suit : L1, L2, L3, L4, L5, L6, où les distances L_i sont indiquées sur le schéma de la figure 3.23). Le copier-coller est possible et conseillé pour éviter les erreurs d'encodage.*

Sélectionner maintenant ***Apply Mask*** pour choisir la région où l'image sera orthorectifiée (Figure 3.24). Cela permettra une nouvelle fois d'accélérer le processus.

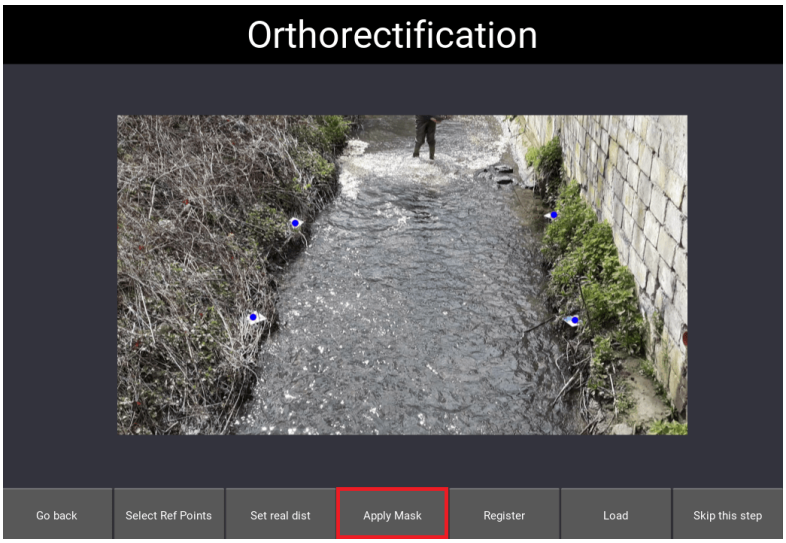


FIGURE 3.24 – Choix d'application du masque

Pour choisir la région, cliquer, à l'aide de la souris, une première fois pour lancer une ligne et une seconde fois pour terminer cette ligne. Une fois la région choisie, sélectionner ***Register*** (Figure 3.25).

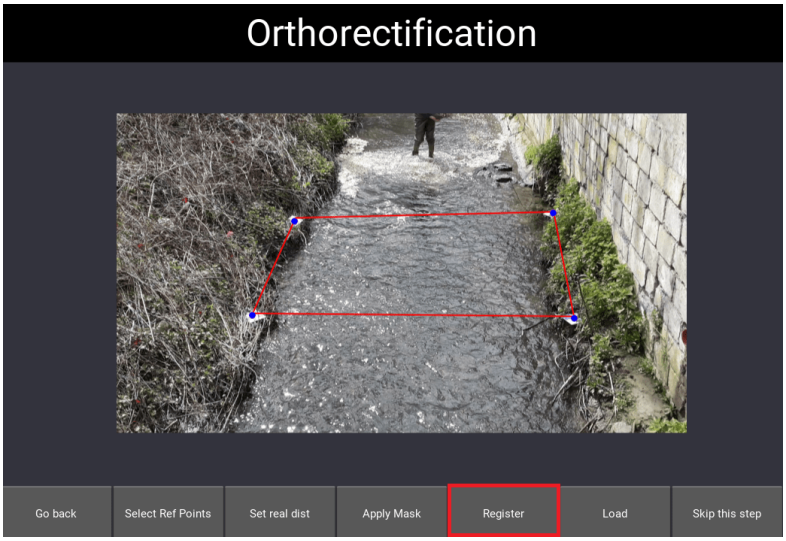


FIGURE 3.25 – Application du masque

Remarque 19. *A savoir que lors du deuxième clic, l'application lance une nouvelle ligne également. De ce fait, il ne faudra que trois clics pour faire deux lignes. Le phénomène est répété pour la troisième et quatrième ligne.*

Remarque 20. *Lors de l'ébauche de la région, les lignes apparaissent en rouge. Lorsque l'utilisateur aura sélectionné* ***Register****, ces lignes passeront au bleu (Figure 3.26).*

Une fois ces 3 étapes accomplies, passer à la fenêtre suivante (Figure 3.27) en sélectionnant ***Load*** (Figure 3.26).

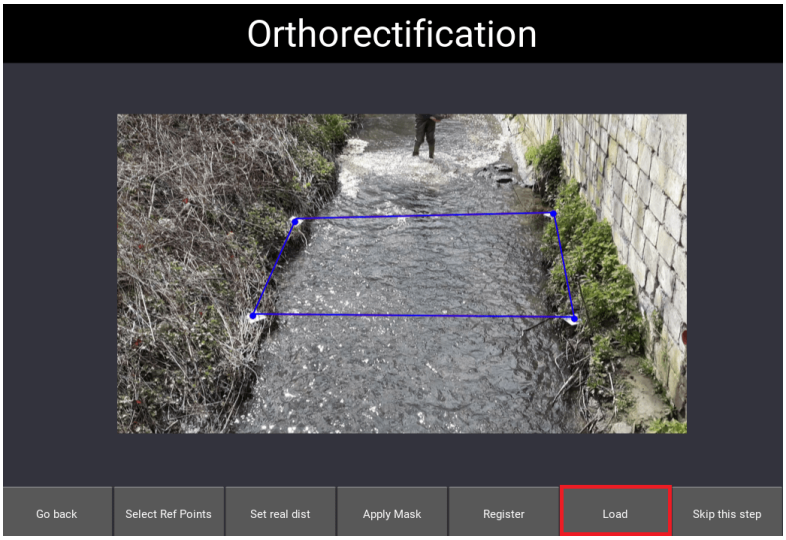


FIGURE 3.26 – Chargement des données d'orthorectification

L'orthorectification n'étant pas obligatoire (vidéo filmée par un drone par exemple), il est tout à fait acceptable de ne pas l'effectuer et de passer à l'étape suivante. Si c'est le cas, choisir ***Skip this step*** (Figure 3.21).

3.2.14 Détermination du niveau d'eau

La détermination du niveau d'eau est le challenge le plus important de cette application. En effet, c'est la donnée la plus compliquée à mesurer. Cependant, plusieurs techniques ont été mises en place pour obtenir cette valeur (Figure 3.27).

1. Set value (Figure 3.27)

Cette méthode sera utilisée si le niveau d'eau de la rivière analysée est connu. Après sélection, une fenêtre pop-up devrait apparaître (Figure 3.28).

Entrer le niveau d'eau en mètres dans la case pourvue à cet effet et continuer en cliquant sur ***Load***. L'écran principal de la région d'intérêt et des masques devrait apparaître (Figure 3.34).

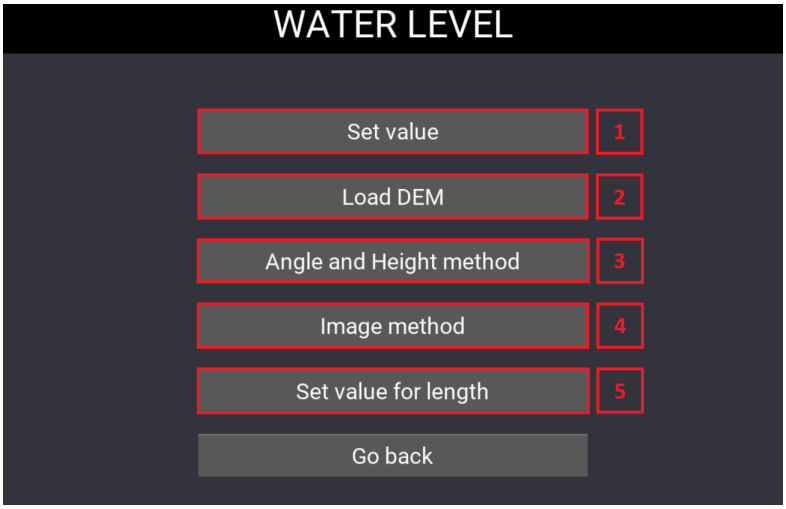


FIGURE 3.27 – Choix de la méthode de détermination du niveau d'eau

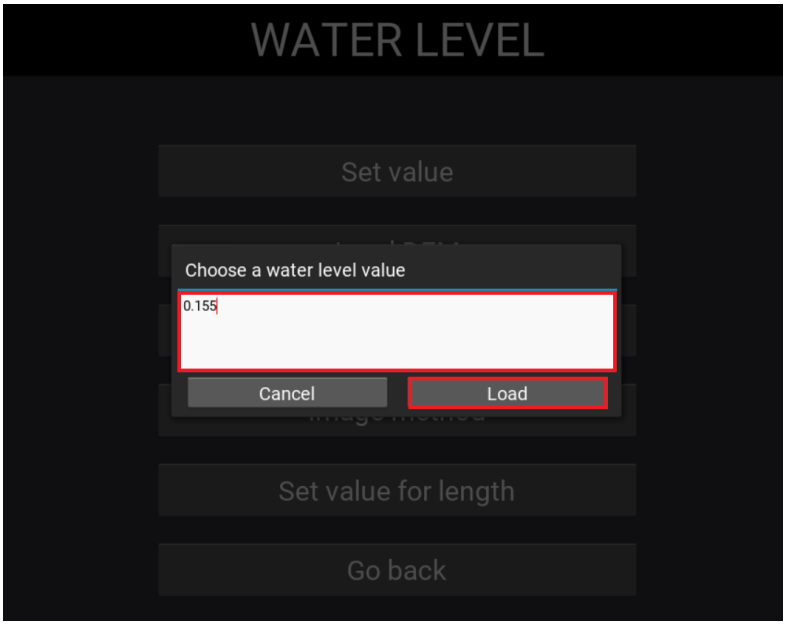


FIGURE 3.28 – Encodage manuel du niveau d'eau

Remarque 21. *La hauteur renseignée doit rester cohérente par rapport à la bathymétrie utilisée. La remarque 2 reste évidemment d'application.*

2. Load DEM (Figure 3.27)

Cette méthode peut être préférée si un fichier `.tiff` est disponible. Ce fichier contient la topographie de la rivière et de ses abords. Si l'on ouvre ce fichier, on pourra normalement apercevoir le relief du site accompagné d'une échelle de couleur pour les différentes hauteurs. L'objectif de cette méthode est de superposer

la bathymétrie et la courbe DEM [2]. Ces courbes devraient alors se rapprocher avant de diverger. Le point de rapprochement définit le niveau d'eau et permet donc d'obtenir la hauteur d'eau.

Sur l'écran qui apparaît à la figure 3.29, sélectionner en premier lieu, dans le menu déroulant, le fichier `.tiff` contenant le DEM ***1. Name***.

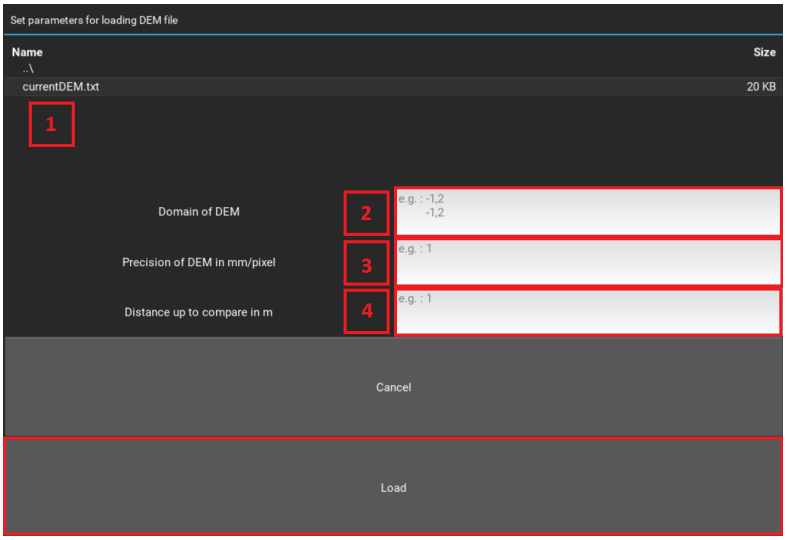


FIGURE 3.29 – Détermination du niveau d'eau à partir du DEM

Dans un second temps, compléter ***2. Domain of DEM*** avec les coordonnées, en mètres, de la région du DEM pour s'assurer que celui-ci et la bathymétrie soient dans le même repère (Figure 3.29).

Remarque 22. *La coordonnée x est celle en travers de la rivière tandis que la seconde est le long de la rive. L'intervalle x est donné dans la première ligne puis la plage y est donnée sur la deuxième ligne. La remarque 3 reste d'application.*

Remplir ***3. Precision of DEM in mm/pixel*** la précision du DEM en terme de millimètres par pixel.
Finalement, encoder ***4. Distance up to compare in m*** la distance, en mètres, à laquelle le DEM doit être comparé à la bathymétrie afin de limiter la recherche près de la rive.
Achever l'encodage et sélectionner ***Load***. Le graphe avec les deux courbes (figure 3.30) fait alors son apparition.

2. Digital Elevation Method

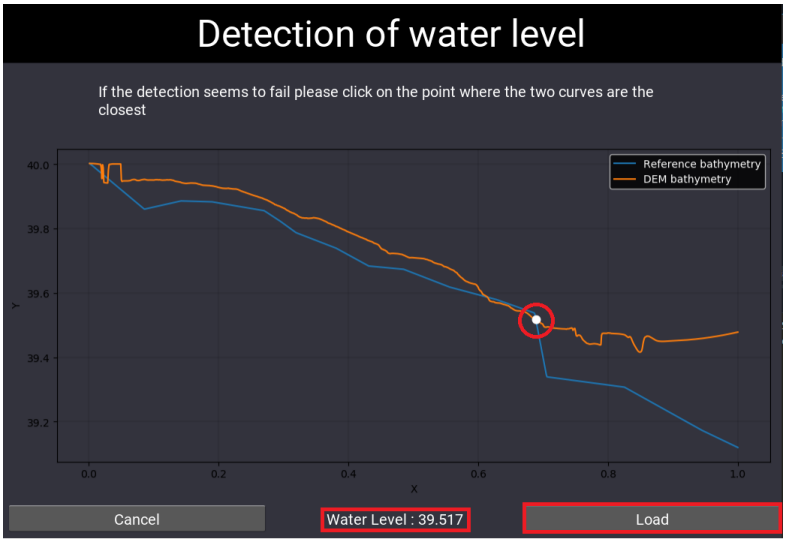


FIGURE 3.30 – Détermination du niveau d'eau à partir du DEM

Une proposition du point de rapprochement est faite par l'application. Si celle-ci convient, appuyer sur ***Load***. Si, au contraire, le point de rapprochement suivi d'une divergence n'est pas correctement placé, le replacer en cliquant à l'endroit idoine et appuyer sur ***Load***.

Remarque 23. *La hauteur calculée en fonction du point de rapprochement est affichée en bas de l'écran (Figure 3.30).*

3. Angle and Height method (Figure 3.27)

Remarque 24. *L'étape de calibration est obligatoire pour effectuer ce type de détermination du niveau d'eau. Si celle-ci n'a pas été performée, l'application le fera savoir et proposera d'utiliser une autre méthode ou de retourner à l'étape de calibration (Section 3.2.12).*

Sélectionner l'image de la rivière à l'altitude basse (***1. Select image at the lower altitude***) (Figure 3.31).
Répéter l'action pour l'image de la rivière à l'altitude haute (***2. Select image at the higher altitude***).
Encoder la différence de hauteur, en mètres, entre les deux images (***3. Delta of heigth between de two pictures in meters***) et cliquer sur ***Load*** pour poursuivre.

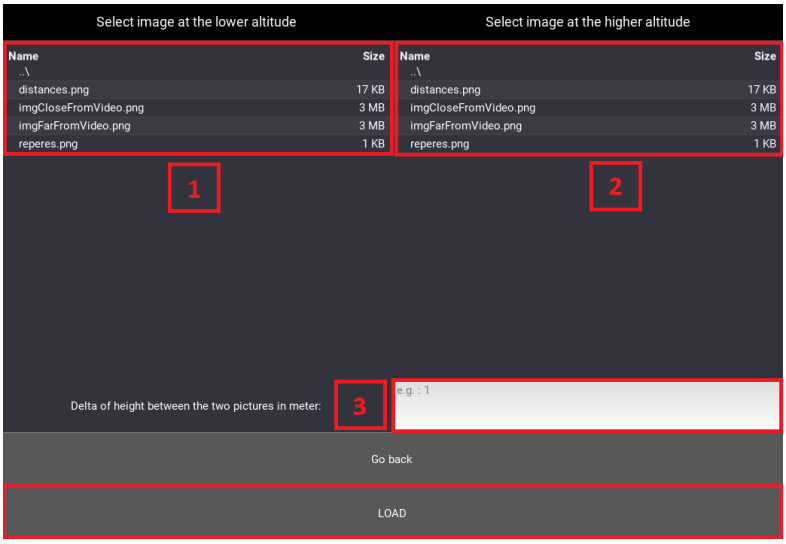


FIGURE 3.31 – Détermination du niveau d'eau à partir de la méthode angle-hauteur

4. Image method (Figure 3.27)

Remarque 25. *L'étape d'orthorectification est obligatoire pour effectuer ce type de détermination du niveau d'eau.*

Sélectionner ***Section Points*** puis choisir les extrémités du cours d'eau (Figure 3.32).
Une fois la sélection terminée, appuyer sur ***Register*** puis ***Load*** pour accéder à la fenêtre suivante (Figure 3.34).

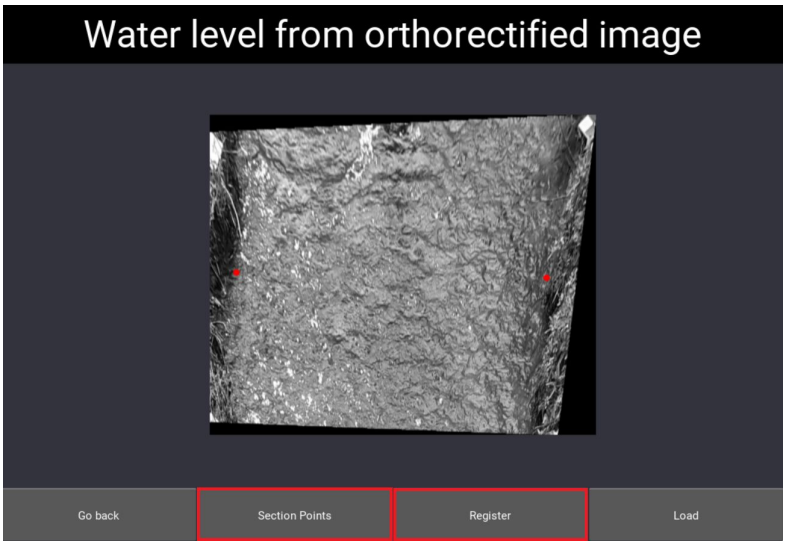


FIGURE 3.32 – Détermination du niveau d'eau à partir de l'image orthorectifiée

Remarque 26. *La remarque 17 est également d'application ici.*

5. Set value for length (Figure 3.27)

Entrer directement la largeur mesurée du cours d'eau puis valider avec ***Load*** (Figure 3.33).

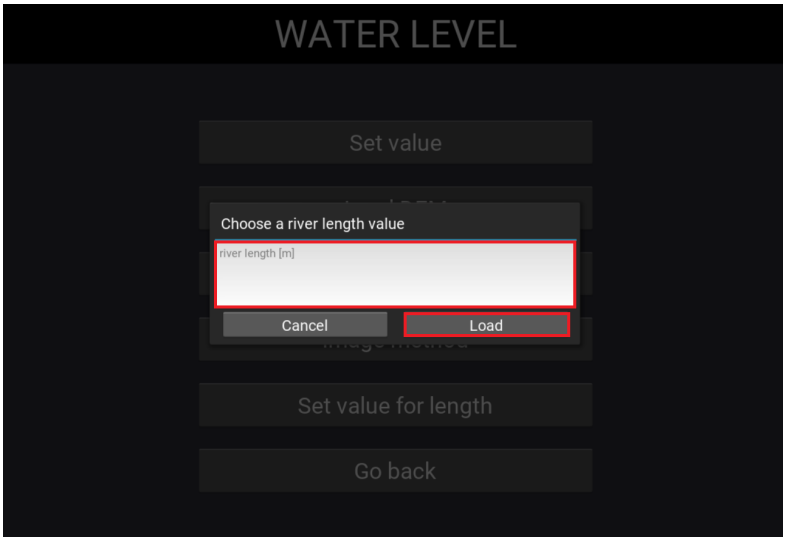


FIGURE 3.33 – Encodage de la largeur du cours d'eau

3.2.15 Région d'intérêt et masques

A ce stade-ci du pre-processing, il est possible de choisir une région d'intérêt et d'appliquer des masques. Cette région d'intérêt permettra à l'analyse PIV de se centraliser sur cette région en particulier. De cette manière, l'analyse se verra accélérée. Quant aux masques, ils permettront de retirer les parties inutiles de cette région d'intérêt. Généralement, on enlèvera les berges. En effet, un mouvement, due au vent par exemple, de la végétation présente sur les berges pourrait interférer lors l'obtention de la vitesse de surface.

Sélectionner ***Select ROI*** pour créer la région d'intérêt (Figure 3.34).
Ensuite, pour étendre la région d'intérêt, cliquer à un endroit en haut à gauche de l'image et entreprendre un mouvement diagonal vers en bas à droite pour dessiner un rectangle. Une fois la forme tracée, cliquer une nouvelle fois sur la souris et appuyer sur ***Register*** (Figure 3.34).

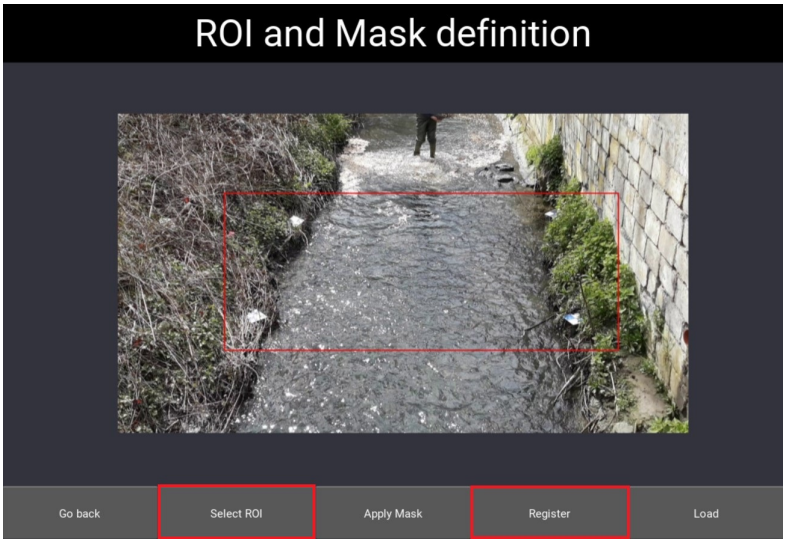


FIGURE 3.34 – Création de la région d'intérêt

Remarque 27. *Lors de l'ébauche de la région d'intérêt, cette dernière apparaîtra en rouge. Celle-ci passera au vert une fois que* ***Register*** *aura été sélectionné.*

Passer ensuite à l'esquisse des masques. Sélectionner ***Apply Mask*** (Figure 3.35).

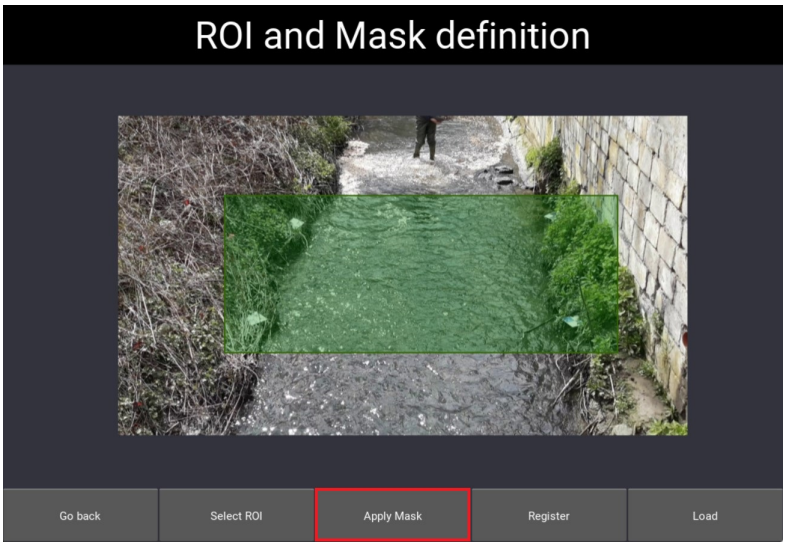


FIGURE 3.35 – Sélection de l'application des masques

Dessiner ensuite clic par clic les contours des parties à retirer de l'analyse PIV (les berges contenues dans la région d'intérêt par exemple). Une fois les contours d'une partie terminés, sélectionner ***Register*** (Figure 3.36).

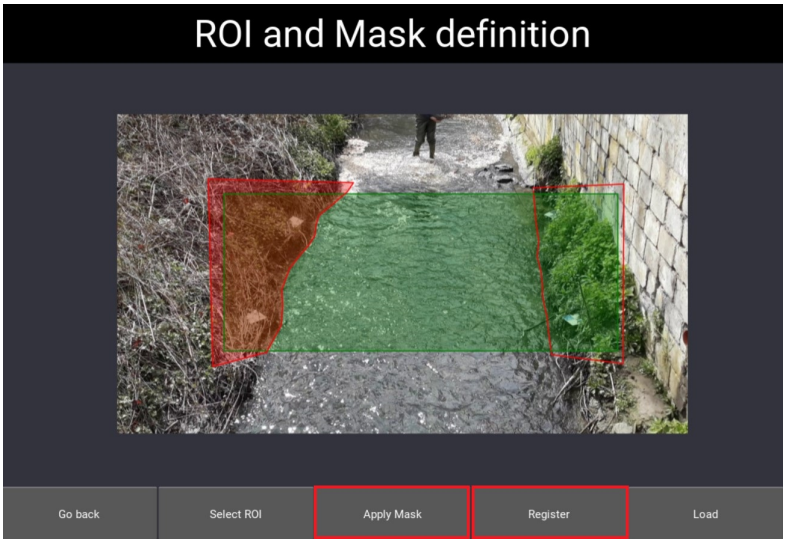


FIGURE 3.36 – Application des masques

Remarque 28. *L'étape est à reproduire autant de fois qu'il y a de parties à ôter.*

Remarque 29. *Identiquement à la remarque 27, seuls les contours apparaîtront en rouge lors de leur ébauche. L'intérieur de la partie entourée par ces contours virera au rouge plus clair lorsque* ***Register*** *aura été sélectionné.*

Pour passer au choix des filtres, choisir ***Load***.

3.2.16 Filtres

La dernière étape avant de procéder à l'analyse PIV est l'application de filtres. Celle-ci n'est pas obligatoire mais elle peut permettre d'accentuer le contraste entre les particules flottantes (les traceurs) et l'eau. Elle permet donc une meilleure identification des particules en mouvement lors de l'analyse PIV.

Pour ajouter un filtre, sélectionner ***Choice Filter***. Une liste des filtres applicables devrait apparaître au-dessus. De là, choisir le filtre le plus adapté à la vidéo. Pour plus d'informations au sujet d'un filtre, cliquer sur ***Information*** après avoir sélectionné ce filtre (Figure 3.37) et/ou consulter le tableau synthétique 3.1.

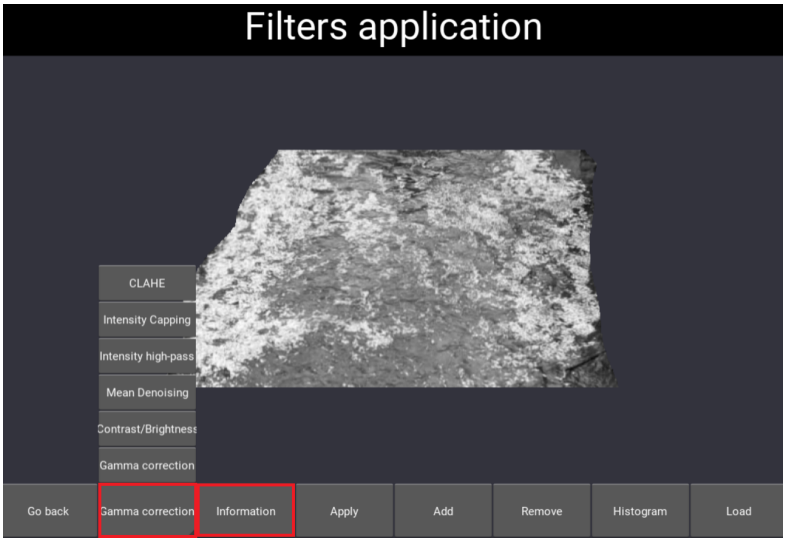


FIGURE 3.37 – Choix du filtre

Afin de d'encoder les paramètres nécessaires à l'application du filtre, appuyer sur ***Apply*** (Figure 3.38).
Une nouvelle fenêtre pop-up doit s'ouvrir avec les données à compléter. Encoder et charger, à l'aide de ***Load***, les valeurs demandées.
Consulter, si nécessaire, le tableau synthétique 3.1 pour connaître le rôle de ces paramètres.

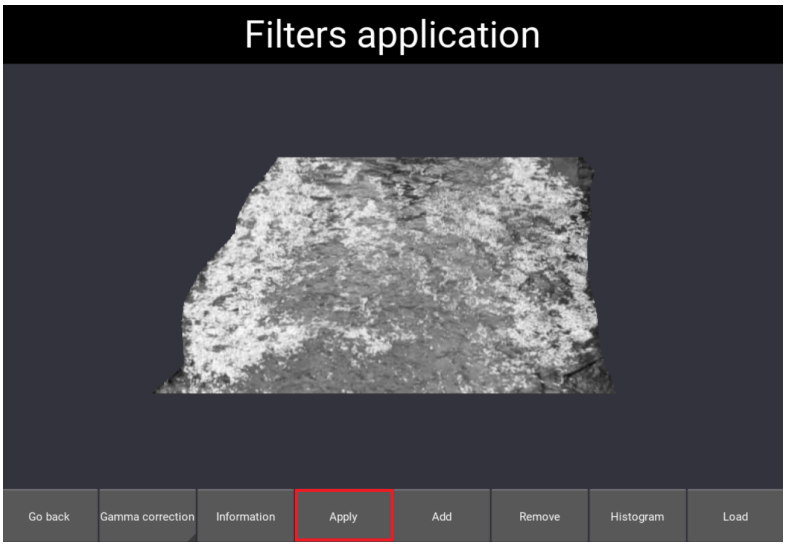


FIGURE 3.38 – Application du filtre

Filtre	Fonction	Paramètres nécessaires
CLAHE	Augmentation du contraste en étendant l'intensité des pixels à tout le spectre.	-clipLimit : Valeur de pixel sous laquelle l'histogramme d'intensité sera limité. -tileGridSize : Taille des carreaux par lesquelles l'image est divisée.
Intensity Capping	Ajout d'un plafond d'intensité à tous les pixels.	-Threshold : Valeur du plafond d'intensité.
Intensity High-pass	Mise en évidence des particules et uniformisation de l'arrière plan.	Non-fonctionnel.
Mean Denoising	Suppression du bruit numérique.	-Filter strenght : Puissance du filtre. -Template Window size : Taille de la fenêtre. -Search Window size : Taille de la fenêtre de recherche.
Contrast/ Brightness	Correction du contraste et de la luminosité.	-Alpha value : Correction du contraste. -Beta value : Correction de la luminosité.
Gamma Correction	Assombrissement ou éclaircissement de l'image.	Gamma value : $\gamma < 1$ éclaircissement, $\gamma > 1$ assombrissement.

TABLE 3.1 – Tableau synthétique des filtres

Remarque 30. *Pour plus d'informations sur les filtres, nous conseillons de se référer aux travaux rédigés par Renard Simon [61] et de Grand Ry Edouard et Everaert Charlotte [11]*

Ensuite et afin d'appliquer définitivement le filtre, appuyer sur ***Add*** (Figure 3.39).

Une fenêtre pop-up devrait faire son apparition disant que le filtre a bien été ajouté.

Il est également permis de supprimer les filtres ainsi que voir l'effet du filtre sur l'histogramme en sélectionnant respectivement ***Remove*** ou ***Histogram***.

Passer à l'étape suivante en cliquant sur ***Load*** (Figure 3.40).

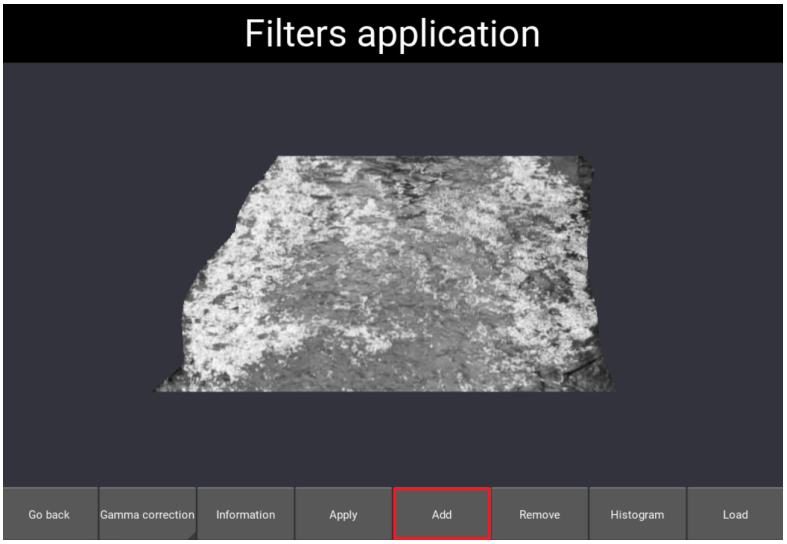


FIGURE 3.39 – Ajout définitif du filtre à la vidéo

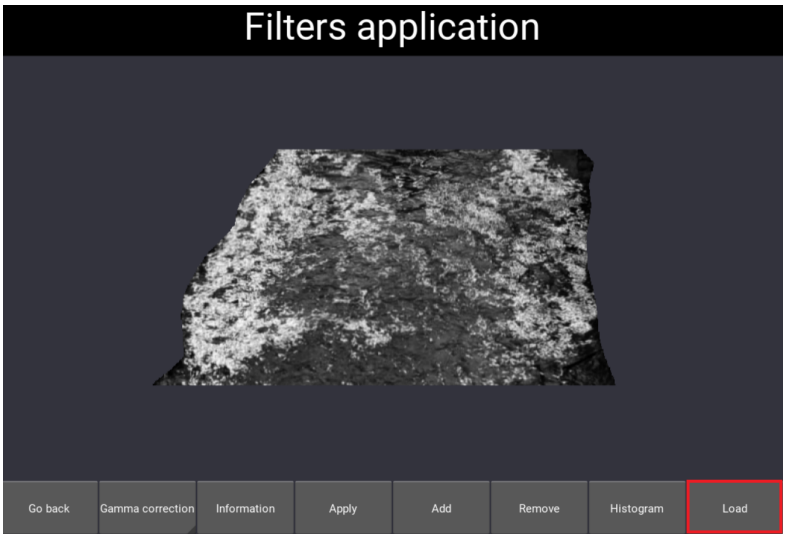


FIGURE 3.40 – Chargement des filtres

3.2.17 Analyse PIV

Le pre-processing étant maintenant terminé, l'analyse PIV va pouvoir être lancée. L'application va donc maintenant calculer les vitesses en surfaces de la vidéo stabilisée (voir section 3.2.11) et filtrée (voir section 3.2.16) et ce dans l'intervalle (voir section 3.2.10) et la région d'intérêt choisis (voir section 3.2.15).

Pour entreprendre l'analyse de manière classique, sélectionner ***1. Single pass analysis*** (Figure 3.41).

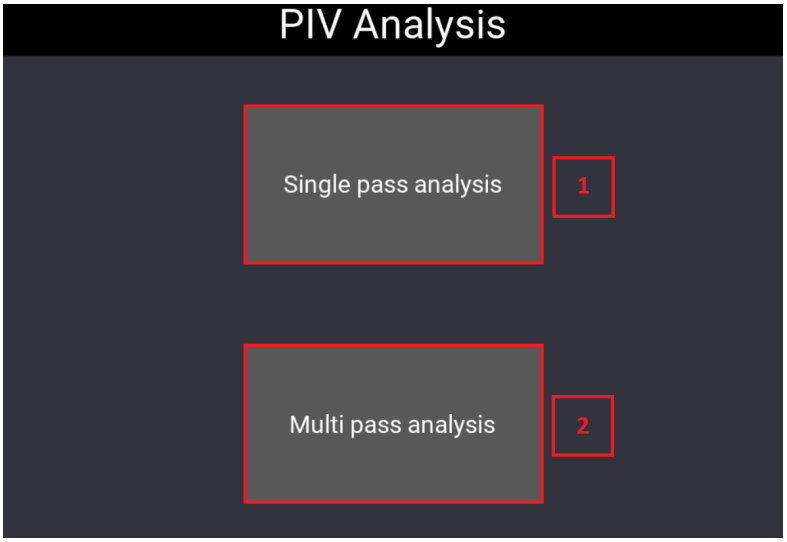


FIGURE 3.41 – Choix du type de l'analyse PIV

La fenêtre pop-up de la figure 3.42 devrait s'ouvrir, demandant alors d'entrer les paramètres utiles à l'analyse PIV. Ces paramètres sont :

- La taille de la fenêtre d'interrogation en pixels : le nombre introduit ici est utilisé pour assigner la taille de la fenêtre d'interrogation, mais aussi celle de recherche. Concrètement, augmenter l'aire d'interrogation fournira moins de vecteurs et prendra moins de temps, et vice-versa. Des valeurs standard sont 50 pour une vidéo Full HD(1920x1080) et 100 pour une vidéo Ultra HD(3840x2160), bien que cela dépende de la taille de la région d'intérêt intérêt.
- Le chevauchement (ou overlap) : ce paramètre exprime le nombre de pixels entre deux centres de zones d'interrogation. Diminuer l'overlap produira moins de vecteurs et prendra moins de temps, et vice-versa. Il est généralement fixé à la moitié du paramètre précédent.
- Le seuil de validation SNR : ce paramètre joue un rôle dans la validation des vecteurs extrêmes. Plus il est élevé, plus le nombre de vecteurs remplacés sera important. Étant donné son influence sur le champs de vecteurs produit, nous conseillons d'effectuer une mesure en fixant ce seuil à 1 (pas de remplacement) et d'augmenter ensuite légèrement si des zones présentent des vecteurs inconstants. Pour la plupart des vidéos, un seuil SNR supérieur à 1.2 altère fortement le champs de vitesses produit.

Une fois terminé, sélectionner ***Load***.

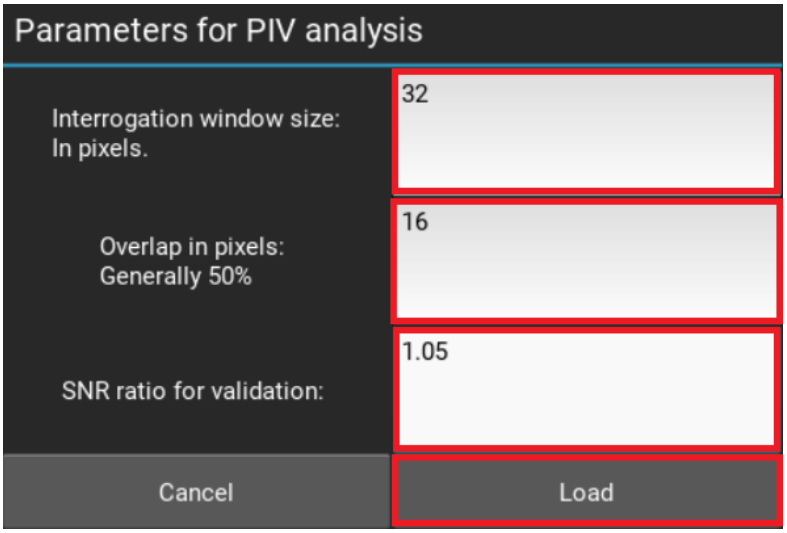


FIGURE 3.42 – Encodage des paramètres d'analyse

L'analyse devrait alors commencer et une barre de chargement devrait apparaître (Figure 3.43).

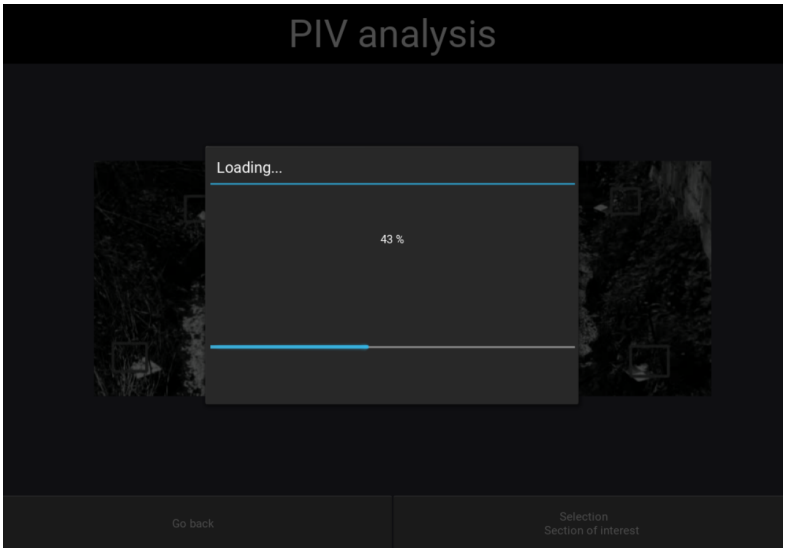


FIGURE 3.43 – Analyse PIV en cours

Lorsque l'analyse est terminée, l'application affiche une image avec les vecteurs vitesses moyens relatifs calculés. Ces vecteurs seront alors stockés dans un fichier `velocity.npy` dans le dossier principal. Pour continuer et sélectionner la section d'intérêt 3.2.18, appuyer sur ***Selection Section of interest*** (Figure 3.44).

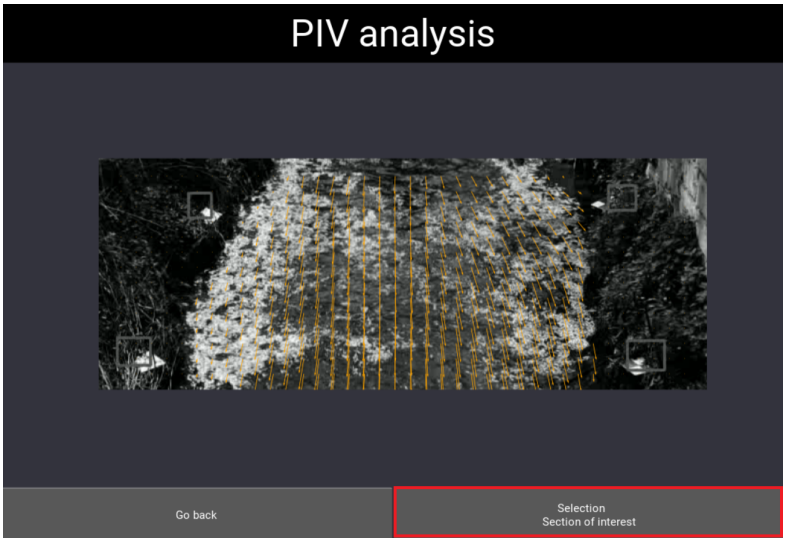


FIGURE 3.44 – Affichage des vecteurs vitesses moyens relatifs

3.2.18 Section d'intérêt

La section d'une rivière étant très rarement constante le long de celle-ci, il convient d'indiquer à l'application la section correspondant au profil du cours d'eau appliqué lors du chargement de la bathymétrie.
Pour choisir cette section, appuyer au préalable sur ***Section Points*** (Figure 3.45).

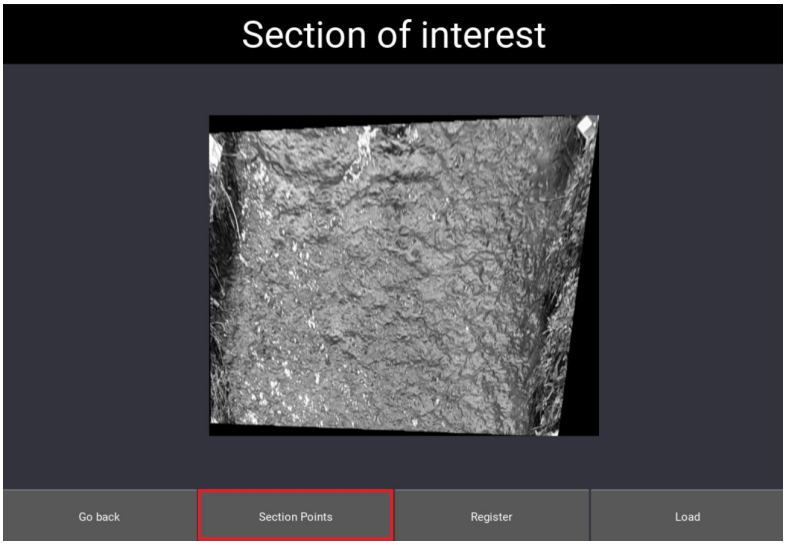


FIGURE 3.45 – Lancement de la sélection de la section d'intérêt

Cliquer maintenant sur l'image, qui est maintenant orthorectifiée, aux deux endroits qui représentent les extrémités de la section et sélectionner ***Register*** suivi de ***Load*** (Figure 3.46).

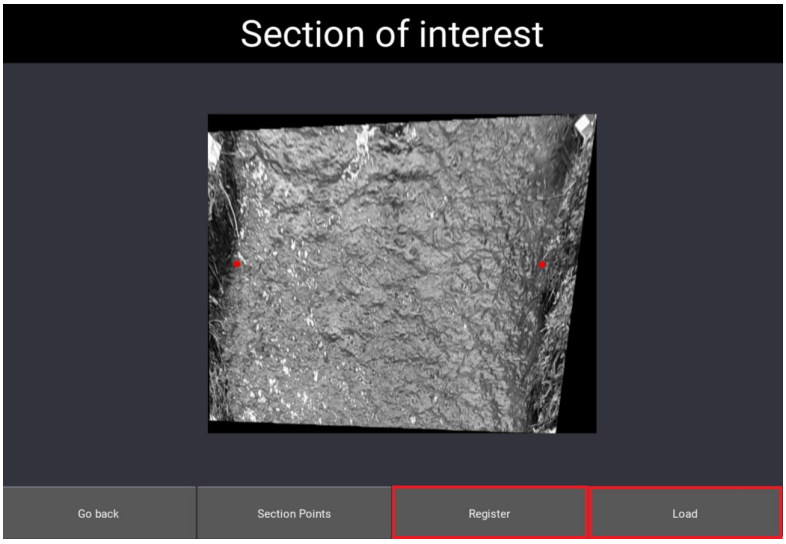


FIGURE 3.46 – Sélection de la section d'intérêt

Remarque 31. *Encore une fois la remarque 17 est d'application lors de la sélection des extrémités.*

Encoder le nombre d'intervalles par lequel la section d'intérêt sera divisée dans la fenêtre pop-up apparue (Figure 3.47). Pour chaque sous-section, un débit sera calculé et un vecteur vitesse moyen sera affiché.

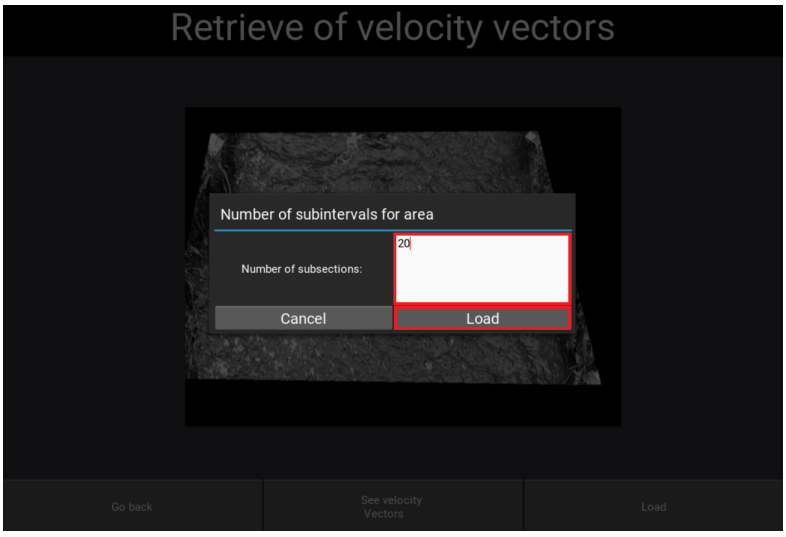


FIGURE 3.47 – Encodage du nombre d'intervalles de la section d'intérêt

Après avoir sélectionné ***Load***, les vecteurs vitesses le long de la section d'intérêt pourront être visualisés en appuyant sur ***See velocity Vectors*** (Figure 3.48).

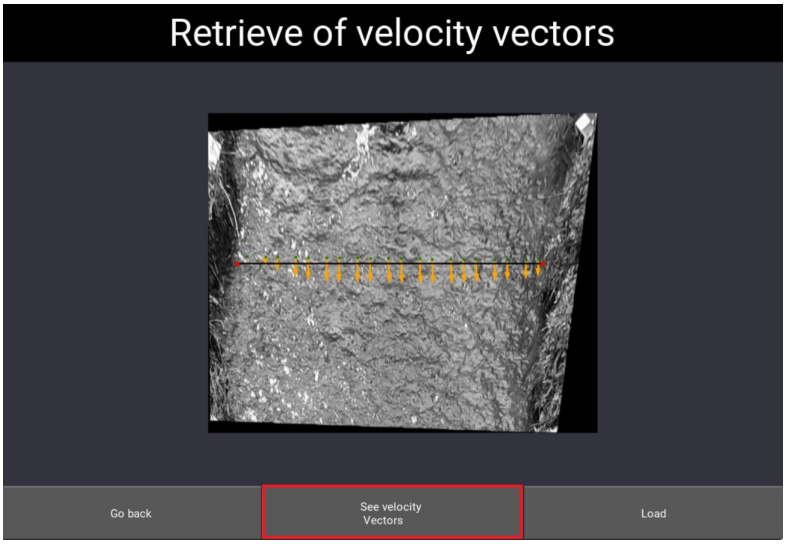


FIGURE 3.48 – Révélation des vecteurs vitesse

Pour passer à la dernière étape, appuyer sur ***Load***.

3.2.19 Détermination du débit

Jusqu'à maintenant, l'application a uniquement calculé des vitesses de surface. Or la vitesse de l'écoulement n'est pas constante sur la profondeur de la section. Ainsi on observera une vitesse plus faible au droit du lit de la rivière et une vitesse plus élevée en surface. On introduit donc un coefficient qui lie la vitesse moyenne et vitesse de surface dans la prochaine étape.

Encoder le coefficient de surface approprié (Figure 3.49). Ce coefficient est compris entre 0.7 et 1 selon la profondeur et le type de lit. La valeur standard est 0.85.

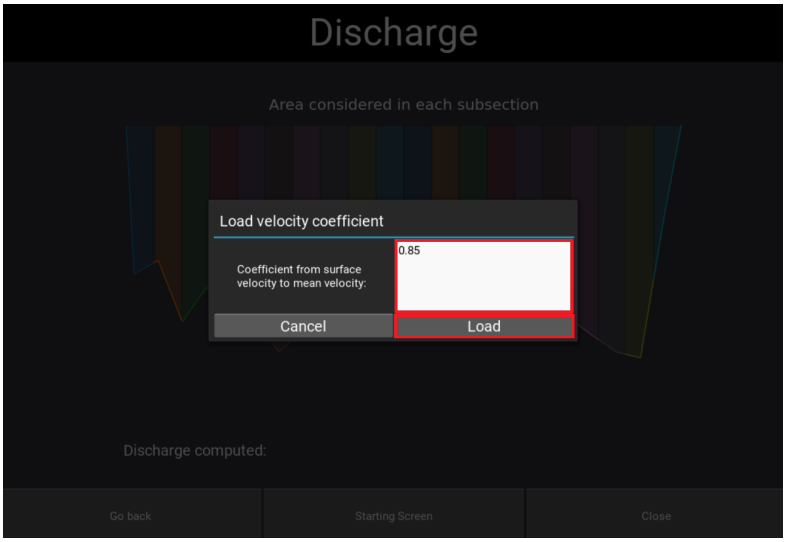


FIGURE 3.49 – Encodage du coefficient de surface

Les différents intervalles de la section d'intérêt ainsi que le débit calculé apparaîtront enfin !

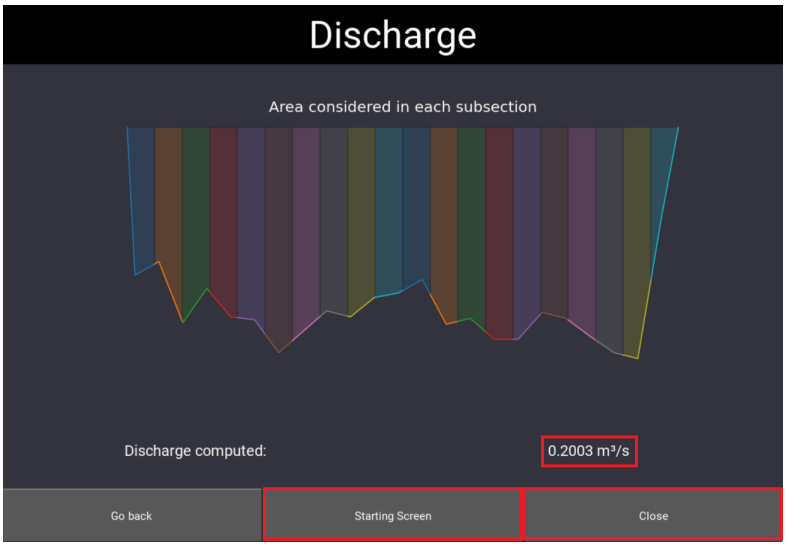


FIGURE 3.50 – Affiche du débit

Pour recommencer une analyse depuis le début ou fermer l'application, activer respectivement les boutons ***Starting Screen*** ou ***Close*** (Figure 3.50).

Chapitre 4

Problèmes rencontrés avec l'application

Dans cette section, nous allons évoquer et expliquer les différents types de difficultés auxquelles nous avons dû faire face avant de pouvoir utiliser l'application de manière fonctionnelle, efficace et précise. Elles seront présentées dans l'ordre dans lesquelles nous les avons rencontrées, et pour chacune nous expliquerons ce que nous avons mis en place et les éclairages que chaque situation a pu nous apporter.

4.1 Problèmes de transmission

Nous avons obtenu le code de l'application via *Transvol*, le service de transfert de fichiers volumineux de l'UCLouvain. Cependant, il est vite apparu que ce mode de transmission comportait plusieurs lacunes qu'il convenait de combler, afin d'assurer la pérennité et la robustesse de l'application.
En effet, un lien *Transvol* nécessite une action d'un détenteur du code source de l'application et n'est valable que 14 jours. La version ainsi transmise dépend entièrement de l'envoyeur et n'est donc pas obligatoirement la dernière ou ne dispose pas forcément de tous les fichiers nécessaires au bon fonctionnement du programme. Dans notre cas, certaines images présentes dans l'interface ne nous avaient pas été fournies, générant des erreurs d'affichage. On peut ainsi craindre un phénomène de téléphone arabe, où le code perdrait en qualité à chaque transfert. Tous ces inconvénients, qui affectent déjà un petit groupe de collaborateurs et utilisateurs, empêchent une bonne distribution à plus grande échelle.

Une solution pour pallier ce souci avait déjà été évoquée dans les mémoires précédents : l'Open Source. Ce mode de distribution sera détaillé à la section 8.

4.2 Problèmes de compilation

Nous avons donc reçu l'application sous forme d'un dossier contenant les fichiers Python correspondant aux différentes fonctions composant l'application. Cependant, en exécutant le fichier `riverDischargeApplication.py`, de nombreuses erreurs apparaissaient dans la console, empêchant l'application de se lancer. Dans un premier temps, nous avons tenté de débugger chaque erreur séparément, ignorant que le problème était plus global. Dès lors, à chaque résolution d'erreur, nous ne parvenions pas à faire démarrer le programme car de nouvelles erreurs apparaissaient ou d'anciennes déjà résolues refaisaient surface. Il nous a alors fallu chercher de manière plus générale les raisons pour lesquelles l'application, qui fonctionnait pourtant sur les ordinateurs de Edouard de Grand Ry et Charlotte Everaert, ne parvenait même pas à compiler sur les nôtres. Nous nous sommes alors rendu compte que la majorité des messages d'erreur ne provenaient pas du code en lui-même mais de l'importation des packages nécessaires à son bon fonctionnement.
Pour mieux comprendre, il convient d'expliquer le fonctionnement et la philosophie de Python, des librairies et des modules qui y sont associés.

Fonctionnement et philosophie *Python* : Une bibliothèque est un ensemble de classes, variables et fonctions liées à un thème ou un domaine spécifique. Le langage de programmation ***Python*** en possède une multitude, c'est d'ailleurs ce qui constitue une de ses principales forces. Ces librairies peuvent être issues du pack originel de ***Python***, auquel cas on les appelle des modules intégrés.
Les autres, les modules tiers, sont conçus par des développeurs et mis à disposition de la communauté permettant à tout un chacun de les utiliser. Ces packages, tout comme ***Python*** à proprement parler, sont régulièrement mis à jour pour régler des bugs ou ajouter de nouvelles fonctionnalités. De ce fait, un programme utilisant ces modules peut être affecté de diverses manières : une fonction peut prendre de nouveaux arguments, des parties de librairies peuvent devenir obsolètes, de nouvelles variables peuvent entrer en conflit avec celles d'autres librairies, etc.
Pour passer outre ces changements, il convient de figer la version utilisée par le logiciel pour que même si des mises à jour ont lieu dans les packages, le code utilise la version du module pour lequel il a été conçu. Il existe une fonctionnalité dédiée dans ***Python*** : les environnements. Il s'agit de "dossiers auto-suffisants qui contiennent une installation de *Python* pour une version particulière de *Python* ainsi que des paquets additionnels" [54]. Un environnement doit être généré via le terminal, et idéalement sur l'ordinateur sur lequel l'application a été créée et fonctionne normalement.

Dans notre cas, le lien avec les modules tiers était effectué via le fichier `PackageFiles.py` qui était appelé au lancement de l'application et importait lui-même tous les paquets nécessaires. Cependant, il n'était pas fait mention de la version de chaque librairie à appeler. De ce fait, la version effectivement importée

ne correspondait pas forcément à celle utilisée au moment de la conception, ce qui générait des erreurs à répétition. Dès lors que nous avons compris cela, nous avons contacté Edouard de Grand Ry et Charlotte Everaert et leur avons demandé d'extraire l'environnement dédié et de nous le transmettre. Nous avons ensuite pu l'installer et l'activer sur nos terminaux.
A contrario, la librairie OpenPIV, qui performe l'analyse PIV pour fournir les champs de vecteurs, avait été importée dans le code et était utilisée en local. Or, ce mode de fonctionnement contrevient aux bonnes pratiques. Nous avons donc changé cela et remis l'accès à la librairie OpenPIV.

4.3 Problèmes de fonctionnement

Une fois que nous avons réussi à compiler et démarrer le programme, nous avons voulu le faire fonctionner. Pour ce faire, nous disposions de vidéos issues des mémoires précédents, et pour lesquelles les informations de bathymétrie, de position des balises GCP, de profondeur et du débit de contrôle ($Q = 0.203 m^3/s$, issu d'Aqualim) étaient connues. Pour être certains d'utiliser l'application de la bonne manière, nous avons reproduit à l'identique, fenêtre après fenêtre, l'exemple qui avait été exposé par Edouard de Grand Ry et Charlotte Everaert lors de leur présentation finale sur une vidéo de la Dyle à Noirhat prise par leurs soins.

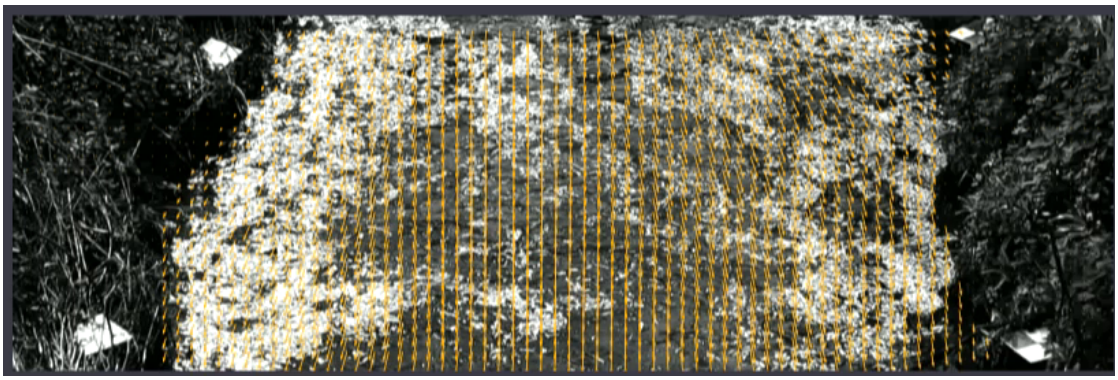


FIGURE 4.1 – Champ de vecteurs présenté par Edouard de Grand Ry et Charlotte Everaert dans leur vidéo, pris comme référence. Le débit trouvé ici par l'application est de 0.2162 $[m^3/s]$.

Malgré des paramètres identiques de part et d'autre, les résultats que nous obtenions étaient très éloignés des leurs d'une part, et totalement inconstants d'autre part, que ce soit au niveau du champ de vecteurs vitesse ou du résultat final. Les résultats que nous obtenions présentaient en effet des comportements totalement incohérents, avec des tourbillons, des parties vides de vecteurs ou des parties à contre-courant, alors que l'extrait de vidéo, lui, présentait un courant relativement rectiligne (Figure 4.2).

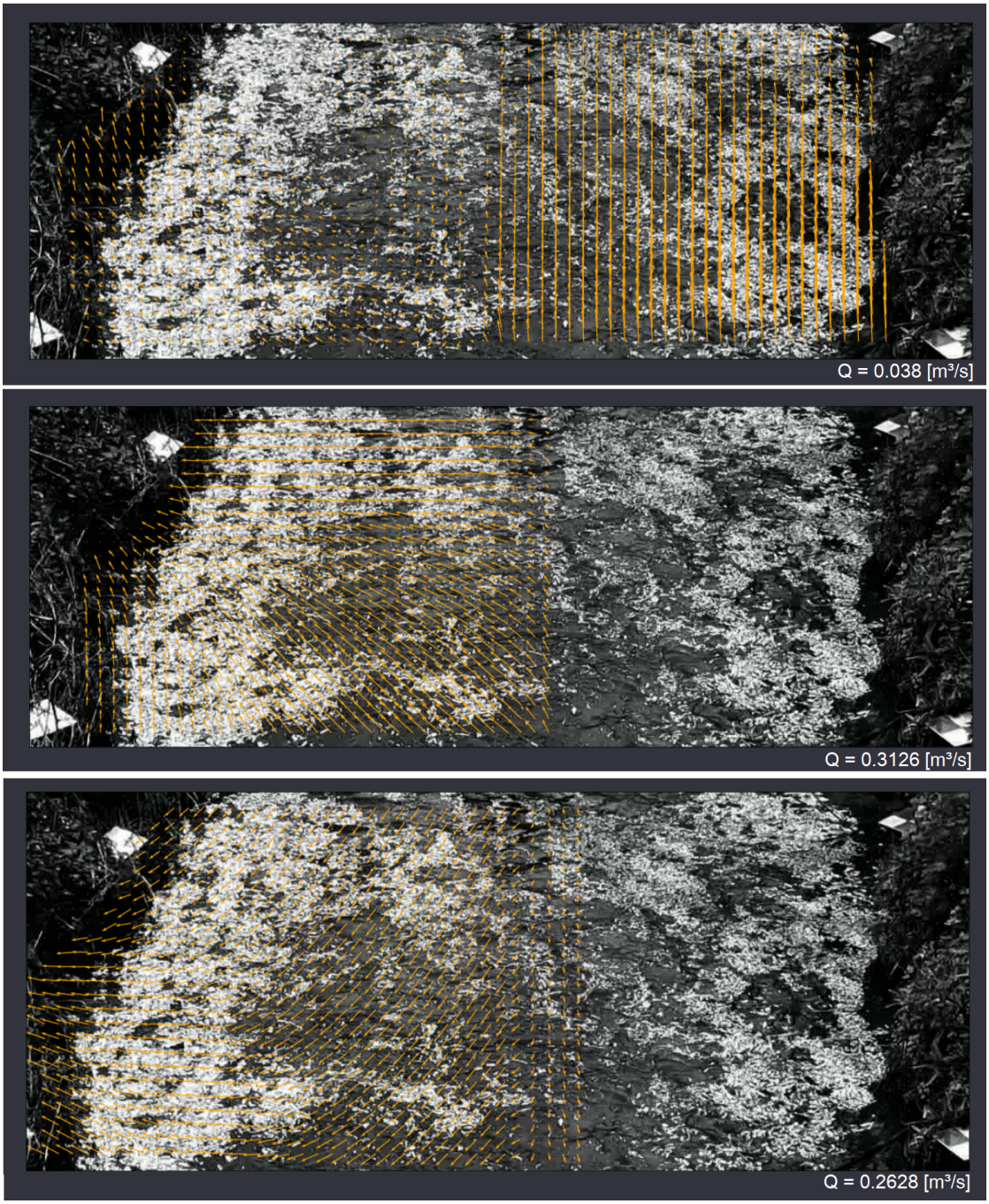


FIGURE 4.2 – Champs de vecteurs et résultats finaux obtenus

Nous nous sommes alors rendu compte que ce problème provenait d'une mauvaise valeur du Signal to Noise Ratio (SNR) qui faisait donc que l'application remplaçait la plupart des valeurs par d'autres intrepolés comme expliqué au point 2.3.2. Après voir résolu cela, nous étions enfin en mesure d'obtenir des champs de vecteurs et des débits cohérents.

4.4 Problèmes encore persistants

Bien que ces problèmes ne gênent pas l'utilisation de l'application et l'analyse complète des différents cours d'eau, ils alourdissent sensiblement chaque procédure ou empêchent le bon fonctionnement de ***Python***. Ces problèmes se présentent selon différents types et peuvent être de nature : Bug, Ergonomique, Affichage, Fonctionnalité. Nous entendons par ergonomiques des problèmes qui ont tendance à rendre l'utilisation de l'application peu pratique. Tous ces problèmes seront consignés et classés dans la forge de l'UCLouvain afin de rassembler le code et ses erreurs.

4.4.1 Problèmes/manquements de nature : Bug

- **Erreur lors de l'arrêt de l'application :**
  Lorsque l'analyse que l'on voulait performer est terminée, on peut aisément quitter l'application soit via la croix de fermeture de la fenêtre soit via le bouton *Close*. Dans les deux cas, l'application se fermera avec succès mais une erreur sera envoyée dans la console.
  Bien que l'erreur octroyée à l'arrêt de l'application ne gêne en rien son interruption, il serait important de savoir comment résoudre cette erreur. En effet, et on peut le voir à l'erreur suivante, celle-ci peut influencer le bon redémarrage de l'application pour une prochaine analyse.
- **Erreur lors du redémarrage de l'application :**
  Si le noyau n'a pas été redémarré après que l'erreur précédente soit survenue, une autre erreur surviendra au redémarrage de l'utilisation suivante, empêchant l'application de fonctionner. Un redémarrage du noyau permet de ne pas rencontrer ce problème mais il semblerait plus opportun de corriger l'erreur plutôt que constamment redémarrer le noyau. Ces deux erreurs semblent ne survenir que sur l'IDE *Spyder* et il est à noter que ces erreurs surviennent également après un crash de l'application quelle que soit la raison.
- **Choix de la vidéo :**
  Lors de la sélection de la vidéo, si l'utilisateur sélectionne une vidéo, il ne peut pas revenir au dossier contenant sa sélection sous peine de faire crasher l'application.
- **Menu déroulant capricieux :**
  Lors de certaines étapes, il arrive que l'utilisateur soit confronté à un menu déroulant, comme dans le choix de sa vidéo par exemple (voir section 3.2.10). Si l'utilisateur ne dispose pas de souris et qu'il souhaite utiliser le pavé tactile de son ordinateur, il pourra faire descendre le menu déroulant sans encombres. Néanmoins, il ne pourra pas utiliser son pavé tactile pour remonter dans le menu déroulant.
  Il est évident qu'il serait préférable de pouvoir utiliser le pavé tactile dans

les deux sens.

- **Bug de l'*Intensity High Pass* :**
  Une étape de l'application consiste à choisir et ensuite apposer un filtre sur les images de la vidéo (voir section 3.2.16). Parmi les filtres proposées, on peut notamment y trouver un dénommé ***Intensity High Pass***.
  Cependant, et bien que celui-ci soit disponible, son utilisation fera crasher l'application et l'utilisateur se verra dans l'obligation de la relancer.
- ***Ne répond plus* :**
  Une fois que l'utilisateur a choisi le ou les filtres qui convenaient à l'analyse de sa vidéo (voir section 3.2.16) et qu'il décide de passer à l'étape suivante, il arrive très souvent que l'application ne réponde plus. Cela ressemble à un crash banal de l'application sauf qu'après quelques secondes, celle-ci continue et passe à l'étape suivante.
  Nous ne savons pas ce qui induit cette latence lors de l'ajout de filtres. Bien que cela ait l'air sans conséquence dans un premier temps, cela pourrait cacher un problème plus profond qui pourrait resurgir plus tard lors d'améliorations.
- **Redressement de la section d'intérêt :**
  Lors de l'encodage du nombre de sous-sections, il arrive que l'application ne fonctionne plus. Nous pensons que cela est dû à l'endroit de sélection des extrémités de la section d'intérêt. En effet, si cette section venait à ne pas être horizontale, l'application remettra celle-ci à plat lors de l'encodage du nombre d'intervalles de calculs. Si une des extrémités, lors de la remise à plat, venait à sortir du cadre prévu dans l'application, celle-ci crashera.
- **Nombre de sous-sections :**
  Avant de calculer le débit via l'application (voir section 3.2.18), il est demandé à l'utilisateur d'entrer un nombre d'intervalles par lequel sera divisé sa section d'intérêt.
  Si le nombre de sous-sections introduit est trop grand par rapport au nombre de vecteurs générés sur la longueur, l'application crashe.
- **Mauvaise aire mouillée considérée :**
  Si l'utilisateur introduit une valeur de hauteur d'eau se situant dans un creux de la bathymétrie, l'application ne va considérer que cette partie lors du calcul du débit, menant à des débits sous-estimés (Figure 4.3).

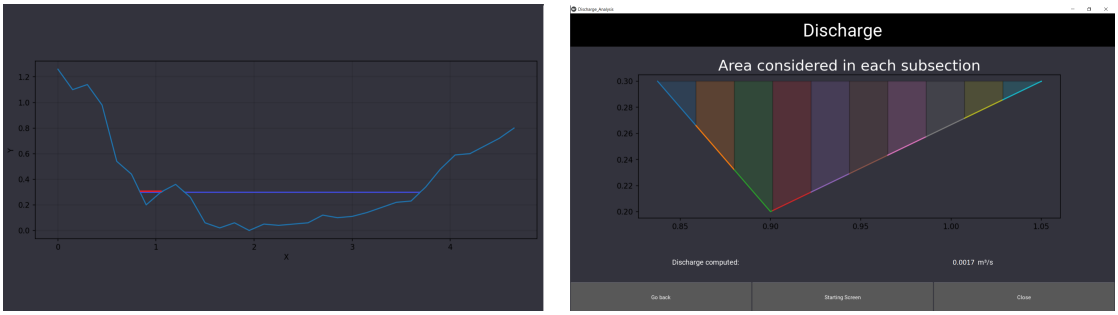


FIGURE 4.3 – Illustration du problème rencontré. L'application devrait considérer la ligne bleue mais elle considère la ligne rouge, menant au résultat représenté à droite.

4.4.2 Problèmes/manquements de nature : Ergonomique

- **Robustesse :**
  Quand un utilisateur prend en main l'application, il arrive que celui-ci ne soit pas encore familier avec l'application. Dès lors, l'encodage de certaines données peut paraître fastidieux et si le format demandé par l'application n'est pas respecté, cette dernière rencontrera plus que certainement un problème de compilation.
  On pourrait donc imaginer que l'application puisse prendre en compte différents types d'encodage ou que celle-ci reconnaisse que l'encodage n'est pas correct. Dans ce cas-là, on verrait apparaître une fenêtre d'avertissement informant l'utilisateur que sa saisie n'est pas sous le bon format.
  De cette manière, on éviterait de devoir recommencer tout le processus d'analyse pour une virgule écrite à la place d'un point lors de la dernière étape par exemple.
- **Explications pour le choix de la vidéo :**
  Au moment où l'utilisateur est amené à choisir la vidéo qu'il désire analyser, il lui est également possible de choisir un ***Step*** ainsi qu'un ***Range*** de lecture (Section 3.2.10).
  Afin de bien comprendre ce que représente chacun des deux paramètres et pour bien différencier leur utilité au sein de l'application, il serait intéressant de joindre une brève description sur ce qu'il convient d'encoder dans celles-ci. Cela pourrait apporter une meilleure clarté lors de cette étape.
- **Consignes de la stabilisation :**
  Au moment de performer la stabilisation, il convient de respecter certaines règles (touches, etc) (Section 3.2.11). Ces règles sont reprises de manière condensée à l'étape précédant celle de la stabilisation.
  Bien que cela soit utile et serve de bon rappel à l'utilisateur, il paraîtrait plus judicieux de fournir ce rappel lors de l'étape de stabilisation.

En effet, une information est vite oubliée et cela rendrait l'application beaucoup plus intuitive quant à son utilisation.

- **Consignes de l'orthorectification :**
  Dans la même optique que la remarque du dessus et comme l'orthorectification demande également un processus précis et fixe pour son établissement, il semblerait qu'un rappel condensé des règles à adopter, lors de cette étape, soit approprié.
- **Ajout de plusieurs filtres identiques :**
  Arrivé à l'étape des filtres (Section 3.2.16), l'utilisateur peut choisir ou non d'apposer un filtre à sa vidéo. Après avoir choisi son filtre ainsi qu'encodé les données relatives à celui-ci, l'utilisateur peut ajouter le filtre en sélectionnant ***Add Filter***. On peut donc apercevoir un changement de contraste entre l'image vierge de filtre et celle avec un filtre.

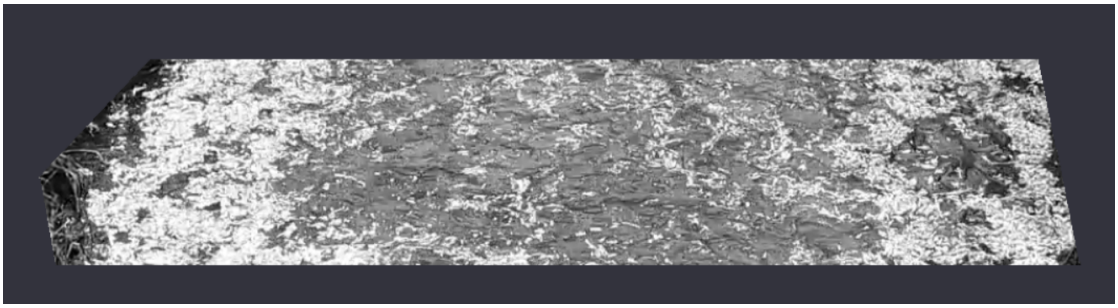


FIGURE 4.4 – Image de la vidéo sans filtre

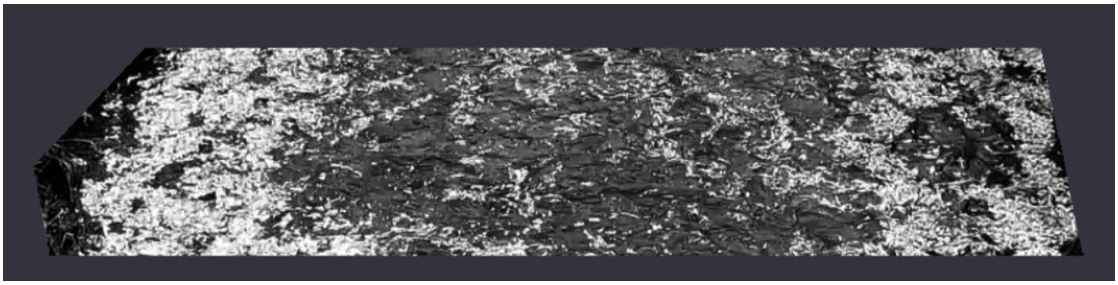


FIGURE 4.5 – Image de la vidéo avec une application du filtre ***Gamma***.

Bien que cela aurait peu de sens, l'utilisateur pourrait vouloir appliquer à nouveau le même filtre. Néanmoins, l'application du même filtre ne changera plus le contraste de l'image. En effet, l'image 4.5 persistera encore après des applications supplémentaires. On pourrait dès lors croire que l'ajout des filtres ne se fait pas correctement. Or, lorsque l'on poursuit l'analyse après l'analyse PIV (Section 3.2.17) l'image suivante apparaît :

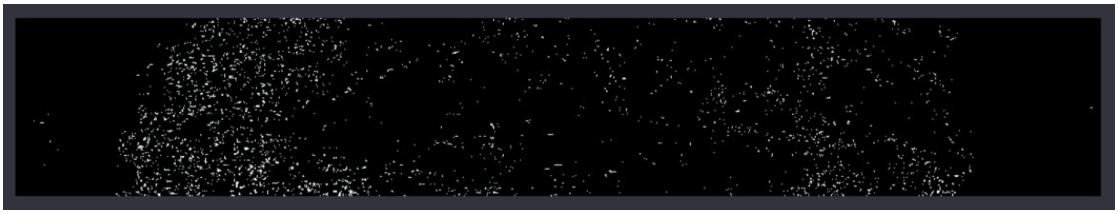


FIGURE 4.6 – Résultat de l'analyse PIV d'une vidéo avec dix applications du filtre ***Gamma***.

On aperçoit donc clairement que les filtres supplémentaires ont bien été appliqués.
Afin que l'utilisateur ne se trompe pas dans le nombre d'application d'un même filtre ou afin qu'il puisse voir l'effet escompté de ceux-ci, il serait préférable que l'évolution du contraste puisse se voir en temps réel et ce même si plusieurs application d'un même filtre sont faites.

4.4.3 Problèmes/manquements de nature : Affichage

- **Esthétique de l'application :**
  L'esthétique de l'application actuelle est assez rustique et banale. De fait, les couleurs ainsi que la forme des boutons sont très peu agréables. Nous trouverions intéressant d'améliorer cette esthétique sans pour autant obtenir quelque chose d'extravagant.
- **Taille des points :**
  Lors de l'orthorectification ou du choix de la section d'intérêt, il est demandé à l'utilisateur de pointer sur l'image des points de références. Ces points servent de base pour les distances à encoder par après afin d'orthorectifier l'image ou pour tracer la section.
  Après plusieurs analyses, nous avons aperçu que la grosseur de ces points variaient d'un ordinateur à l'autre et d'une analyse à l'autre. Bien que cela ne pose pas de soucis concernant la compilation de l'application, il pourrait y avoir un soucis de précision. Pour éviter cela ou pour simplement avoir une application identique sur chaque ordinateur, il serait préférable d'obtenir une taille constante pour tout le monde.
- **Disparition du chargement :**
  Pendant l'analyse PIV, une fenêtre s'affiche pour montrer l'état de progression, en pourcentage, de l'analyse. On peut donc voir l'évolution de l'analyse en direct.
  Le problème d'affichage décelé ici est que si l'utilisateur venait à cliquer sur sa souris pendant la progression, la fenêtre viendrait à disparaître. Même si la progression se poursuit, l'utilisateur n'est plus capable de suivre le chargement en temps réel.
  Il serait alors préférable que l'on puisse soit ne pas faire disparaître cette

fenêtre, soit de pouvoir la faire apparaître à nouveau afin de toujours pouvoir suivre la progression de l'analyse si on le souhaite.

- **Images résiduelles :**
  Après avoir performé une analyse, l'utilisateur à la possibilité de fermer l'application ou de recommencer une analyse en revenant à l'écran de départ. Lorsque celui-ci choisit la seconde possibilité, il s'avère que certaines sélections ou images de l'analyse précédente restent à l'image pendant quelques instants. Dans la mesure où il serait possible d'utiliser à nouveau ces sélections (voir ***Mémoire de l'application*** dans la section 4.4.4), cela serait utile. Mais dans le cas présenté ci-dessus, cela représente seulement une perte de temps et un problème d'affichage.
- **Plusieurs sélections de sections d'intérêts :**
  Lors de la sélection de la section d'intérêt, utile pour générer les vecteurs nécessaires au calcul du débit, l'utilisateur doit déterminer ses deux extrémités en cliquant sur son écran (Section 3.2.18). Il s'avère que l'utilisateur peut enregistrer plusieurs paires de points sur l'image. Cependant, seule la dernière sélection semble être considérée.
  Il serait plus judicieux que les sélections précédentes disparaissent lorsque l'utilisateur sélectionne ***Section points*** comme sur le figure 3.45 plutôt que d'entasser les couples de points sur l'image.
- **Nouvelle analyse :**
  A la fin de l'analyse, l'utilisateur peut, s'il le souhaite, se faire rediriger vers l'écran initial en sélectionnant ***Starting Screen***. Celui-ci peut dès lors procéder à une nouvelle analyse et ce, même en choisissant une vidéo différente de la précédente. Si cette vidéo possède une prise de vue zénithale, elle ne nécessite pas d'orthorectification. L'utilisateur peut alors passer cette étape. Cependant, en n'effectuant pas l'orthorectification, l'application garde en mémoire l'image orthorectifiée de la vidéo précédente. Bien qu'à ce stade-là, l'utilisateur est dans l'incapacité de s'en rendre compte, l'erreur s'affiche lors du choix de la section d'intérêt.
  L'utilisateur est alors forcé de choisir une section qui ne correspond pas à la vidéo qu'il désire analyser.
  Il paraît évident que cela pose un problème quant à l'examen de la vidéo pour l'obtention du débit. Même s'il semblerait que ce soit bien la bonne vidéo qui soit sélectionnée, il devient compliqué de choisir précisément la section d'intérêt, qui reste une étape fondamentale pour l'obtention du débit.

4.4.4 Problèmes/manquements de nature : Fonctionnalité

- **Mémoire de l'application :**
  Lorsque l'on analyse un cours d'eau ou une rivière, il arrive que l'on veuille tester plusieurs paramètres ou l'évolution d'un paramètre et ce sur une ou plusieurs vidéos. Cependant, certaines étapes, lors de l'utilisation de

l'application, restent inchangées et ce malgré les changements de paramètres. Il serait donc intéressant de pouvoir récupérer certaines informations venant de l'analyse précédente.
De plus, quelques étapes demandent à l'utilisateur de dessiner des régions spécifiques à l'aide de sa souris ou de déterminer avec plus ou moins de précisions le début et la fin d'une ligne. Les différences entre chaque analyse pourraient dès lors occasionner une mauvaise interprétation de l'évolution du ou des paramètres choisis par l'utilisateur.
Une touche supplémentaire, telle que ***Previous Selection***, simplifierait, affinerait et accélérerait des analyses consécutives.

- **La touche *TAB* :**
  Sur certaines fenêtres, il arrive qu'il y ait plusieurs cases à remplir. Dans ces cas-là, il est obligatoire de cliquer sur chaque case vide avant de pouvoir écrire à l'intérieur. Il serait plus avantageux et plus rapide que l'utilisateur puisse se servir de la touche ***TAB*** de son clavier et ainsi rendre plus fluide l'encodage d'une telle fenêtre.
- **La touche *ENTER* :**
  L'encodage de certaines fenêtres nécessite que l'utilisateur ne remplisse qu'une et une seule case. Afin de rendre plus rapide et plus fluide l'encodage de ce genre d'étape, il serait utile de pouvoir sélectionner la touche ***ENTER*** du clavier afin d'enregistrer la réponse. Ceci permettra dès lors à l'utilisateur de ne plus devoir reprendre sa souris et avancer directement à la fenêtre suivante.
- **Les touches *Flèches* :**
  Il arrive que l'utilisateur doive choisir un fichier ou autre parmi un menu déroulant. Toujours afin de rendre plus fluide le choix de ces fichiers. Il pourrait être intéressant que l'utilisateur puisse parcourir ce menu à l'aide des ***Flèches*** de son clavier.
- **Encodage direct :**
  Toujours dans l'optique de fluidifier et d'accélérer les encodages. Nous pensons qu'il serait opportun de pouvoir encoder directement les cases laissées vides et ce sans devoir cliquer sur celles-ci pour pouvoir les remplir. En effet, actuellement, il est impossible de se passer de sa souris pour remplir une case.
- ***Go Back* ou *Cancel* ? :**
  Lors de l'utilisation, il peut arriver que l'utilisateur se soit trompé dans un encodage ou ait été trop vite à l'étape précédente et souhaite revenir en arrière. Ceci est tout à fait possible grâce aux touches ***Go Back*** présentes dans de nombreuses étapes. Bien que le retour à l'étape précédente ait été effectué, toutes les informations encodées à l'étape précédente ont également été supprimées. L'utilisateur ne se trouve donc pas seulement une étape en arrière mais bien deux étapes en arrière. Nous ne voyons pas l'intérêt d'ôter ces données alors que le seul but recherché par l'utilisateur était de changer

certaines de ses sélections.
Il serait donc préférable d'ajouter une touche ***Cancel*** ainsi que de bien la différencier par rapport à ***Go Back***.
Ainsi ***Bo Back*** reviendrait effectivement à l'étape précédente mais ***Cancel*** permettrait d'annuler une saisie, un encodage, etc. Pour annuler l'étape précédente complètement ou partiellement, l'utilisateur se verrait alors revenir en arrière dans un premier temps et puis annuler chaque sous-étapes/sélections qu'il désire.

- ***Single pass*, *Multi pass* ou *Go Back* :**
  Lors du lancement de l'analyse PIV (voir figure 3.41), l'utilisateur a le choix de choisir l'analyse ***Single pass*** ou celle ***Multi pass***. Cependant, si celui-ci pense qu'il s'est trompé dans les informations fournies précédemment, il lui est impossible de revenir en arrière comme sur la plupart des autres étapes. Il serait donc judicieux d'ajouter l'option ***Go Back*** sur cette fenêtre afin de pouvoir changer quelconque données précédentes.
- ***"Are you sure you want to quit the app ?* :**
  Après quelques utilisations de l'application, il est aisé de se rendre compte qu'il est fastidieux de faire tout le processus d'analyse. Il serait dès lors dommage que, par mégarde, un utilisateur quitte l'application.
  Mettre en place une fenêtre pop-up, qui apparaîtrait lorsque l'utilisateur choisit de quitter l'application pour lui demander si c'est bien l'action qu'il désire entreprendre pour éviter de mauvaises manipulations, serait intéressant.
  De cette manière, on éviterait du temps perdu par inadvertance lors de l'utilisation de l'application.

Évidemment, certains problèmes ou manquements auraient pu être disposés dans plusieurs catégories.

4.5 Approximations/valeurs sans explication

Dès la première utilisation de l'application, il est apparu que plusieurs valeurs ont été fixées au sein même de l'application. Certaines de ces valeurs sont neutres, c'est-à-dire qu'elles nécessitent un encodage de la part de l'utilisateur avant de pouvoir passer à l'étape suivante.
D'autres valeurs, quant à elles, sont pré-encodées avec une valeur par défaut. C'est le cas, par exemple, de l'aire de recherche, de l'overlap et du SNR lors de l'analyse PIV (Section 3.2.17) ou encore du coefficient de surface (alpha) lors du calcul du débit (Section 3.2.19).

Bien que ces valeurs soient, pour certaines, les plus utilisées, nous étions en droit de nous poser la question de l'influence de ces paramètres sur l'analyse et si les

valeurs renseignées étaient constamment les plus appropriées.
Pour ce faire, nous avons décidé de faire des tests de sensibilité pour percevoir les limites des valeurs à utiliser ainsi que la véracité de celles déjà présentes (Section 7.2).
Afin de disposer d'un jeu de vidéo suffisant, pour pouvoir notamment procéder à ces tests de sensibilités, une campagne de prise de mesures s'est révélée utile (Section 5.1). Nous nous sommes donc rendus, avec l'aide précieuse de ***Firas Ghali***, sur les sites de Noirhat, Suzeryl et Limelette pour effectuer ces prises de mesures. Des mesures, moins précises, ont également été prises lors d'un camp Scout à Lissoir.

Troisième partie

Développements relatifs à l'application

Chapitre 5

Mesures sur le terrain

5.1 Protocole de prises de mesures

Comme expliqué précédemment, il nous était utile de procéder à des campagnes de prises de mesures pour enrichir notre base de données de vidéos, et ce afin d'accomplir des tests de sensibilité sur certains paramètres.

Étant donné que nous ne pouvions nous rendre ensemble lors de chacune des prises de mesures, il était important de s'assurer que les enregistrements pris par l'un comme par l'autre soient exploitables et suivent une démarche identique dans le but de comparer les images des sites différents entre elles.

De plus, nous avons été mis en contact par le biais de notre promotrice avec des chercheurs malgache (Zo Ramahaimandimby), haïtien (Rotchild Louis) et tunisien (Firas Ghali) afin que leurs équipes prennent des mesures et des vidéos sur diverses rivières locales. Nous devions donc nous assurer de recevoir des données utiles pour éviter un travail de mesure de leur part qui serait inexploitable de notre côté alors même que ces chercheurs et leurs équipes n'étaient pas du tout familiarisés avec l'application.

Ces raisons nous ont amené à créer un protocole d'acquisition des données, qui se voulait le plus précis possible. Le but de ce protocole est que nous puissions trouver un débit cohérent via l'application et le comparer à d'autres mesures juste avec ce document dûment complété et les vidéos demandées. Tout en s'assurant des vidéos exploitables, nous pouvons alors nous permettre de ne pas nous déplacer et les équipes de chercheurs, quant à elles, ne doivent jamais ouvrir l'application.

Ce protocole, présent en annexe .3, a été amélioré à plusieurs reprises et est encore destiné à être amélioré selon les retours des différentes équipes effectuant les mesures selon nos instructions.

Ce document est destiné à être lu en amont de la prise d'images afin de prendre connaissance du matériel nécessaire à la prise de mesures, des étapes à suivre pour obtenir des vidéos utilisables par l'application ainsi que des explications succinctes à propos de concepts clés liés à la mesure du débit par l'application. Ce texte reprend la procédure à suivre dans le cas idéal, c'est à dire un point de mesure doté d'un pont, d'une mesure de contrôle et de berges facilement accessibles, d'une luminosité suffisante,etc. mais nous avons conscience que cela n'est pas forcément le cas sur le terrain.

Ce document devra également être imprimé et complété au cours de la prise de mesures. En effet, il est demandé aux personnes chargées de l'obtention des données de remplir les champs vides. Les informations à y consigner permettent de regrouper la bonne vidéo à la bonne bathymétrie, au bon jeu de coordonnées GPS des balises GCPs, etc et également de limiter la perte d'informations pour les utilisateurs qui eux, n'étaient pas sur le terrain.

5.2 Résultats des mesures à l'étranger

Description

Firas Ghali et Zo Ramahaimandimby n'ont malheureusement pas pu effectuer de mesures sur terrain en raison de contraintes locales, mais le contact a été établi et ils ont reçu le protocole et aidé à l'améliorer. Rotchild Louis, quant à lui, a effectué trois vidéos sur la rivière Rouyonne à Léogane en Haïti. Nous sommes très satisfaits de la qualité de ces vidéos, ainsi que de la complétude des informations fournies entre autres via le formulaire. Nous avons été en mesure d'effectuer des tests sur ces vidéos sans problème et disposions de toutes les informations nécessaires sans nous être rendu sur place et sans devoir demander des données a posteriori.

FIGURE 5.1 – Images extraites des trois vidéos prises à Léogâne par Rotchild Louis

La hauteur d'eau y est de 25.1 [cm] et le débit mesuré au courantomètre MFPro était de 0.191 $[m^3/s]$. La luminosité est très homogène de sorte qu'aucun reflet ne gène la prise de mesures. De la végétation bloque la vue sur une petite portion de la rivière, comme cela est visible sur la figure 5.1.

Résultats

Le premier objectif des mesures à l'étranger était d'obtenir un jeu de données utilisables via notamment le protocole de mesure mis en place. Comme dit plus haut, cela est chose faite. Le deuxième objectif potentiel était de pouvoir les inclure dans nos différentes analyses.

Bien que non utilisées pour nos tests de sensibilité (Section 7), nous avons décidé d'effectuer des tests sur les trois vidéos fournies et pour potentiellement faire ressortir d'autres conclusions.

Les trois vidéos envoyées respectaient les directives demandées concernant l'angle de prise de vue. En effet, il avait été demandé de filmer une première fois de manière zénithale au cours d'eau, une deuxième fois de façon axiale mais pas nécessairement zénithale et une dernière fois obliquement à la rivière.
Nous avons dès lors testé les différentes vidéos selon une utilisation normale de l'application et avons obtenu les résultats représentés dans le tableau 5.1.

	Vidéo 1 : Prise de vue zénithale	**Vidéo 2 : Prise de vue axiale**	**Vidéo 3 : Prise de vue oblique**
Moyenne	0,232 $[m^3/s]$	0,166 $[m^3/s]$	0,431 $[m^3/s]$
Ecart type	0,021 $[m^3/s]$	0,059 $[m^3/s]$	0,029 $[m^3/s]$

TABLE 5.1 – Débits obtenus via l'application sur les vidéos prises par Rotchild à Haïti.

Il apparaît assez vite que c'est la vidéo prise perpendiculaire au cours d'eau par un drone, qui fournit les résultats les plus constants et les plus proches de la réalité. Cependant, les résultats des tests obtenus avec la vidéo axiale ne sont pas pour autant mauvais. De fait, même si ceux-ci présentent une variation élevée par rapport à la première vidéo, l'ensemble des valeurs obtenues oscillent autour de la valeur réelle du débit.
Pour ce qui s'agit du dernier enregistrement, les résultats ont l'air de converger vers une valeur précise mais qui malheureusement n'est pas la bonne.

Bien que ces tests soient un peu maigres pour tirer de vrais enseignements, il ressort quand même une tendance qui lie le degré d'orthorectification et les débits calculés

à l'aide de l'application. En effet, moins on orthorectifie plus les résultats sont prometteurs. Cela se laisse facilement observer lorsque l'on sait que la première vidéo n'a pas nécessité d'orthorectification et que la remise à niveau de la seconde était relativement faible.
On peut donc conclure que moins la vidéo nécessite d'être orthorectifiée plus les résultats seront précis et constants. Cette constatation sera également faite lors des tests sur le kit de mesure (Section 6).

5.3 Nouvelles prises de vidéos

D'autre part, nous avons nous aussi effectué des prises de mesures sur le terrain afin d'étoffer la base de données de vidéos utilisables pour l'application et faire varier des paramètres dont l'influence sera étudiée dans la section 7.1. Cela nous a aidé à comprendre l'autre pendant de la mesure de débit en rivières, au delà de l'utilisation de l'application, et permis de tester un prototype de kit (Section 6). Nous nous sommes rendus, comme nos prédécesseurs avant nous, aux endroits des stations Aqualim de Noirhat, Suzeryl et Limelette. L'un de nous a profité de son camp scout, situé aux abords de la Lesse, pour également prendre une série de vidéos. Bien que réalisée avec les moyens du bord, cette campagne était plus destinée à établir les limites de l'application.
Entre chaque campagne, nous avons pu apprendre de nos erreurs et approximations pour améliorer nos prises de mesure ultérieures et le protocole de mesure .3. Chaque campagne va être décrite succinctement ci-dessous.

5.3.1 Noirhat - Dyle - 12 juillet 2022

Cette campagne fût la première des quatre. Les vidéos ont été prises depuis un pont vers l'amont. La hauteur d'eau a varié entre 15 et 20 cm et le débit Aqualim entre 0.114 et 0.162 $[m^3/s]$ pendant la durée de la campagne. Pour ce qui est de la luminosité, il faisait très ensoleillé, ce qui a généré quelques reflets sur la surface.

5.3.2 Suzeryl - Thyle - 13 juillet 2022

Lors de la deuxième campagne, les vidéos ont été prises sur un trépied placé dans le cours de l'eau car la végétation obstruait la vue depuis le pont. La hauteur d'eau était de 45cm au point le plus profond et le débit Aqualim de 0.21 $[m^3/s]$. La végétation était abondante autour du site de mesure et se reflétait sur la surface qui était calme.

Malheureusement, nous nous sommes rendu compte après les premiers essais avec l'application que l'angle de vue étant très faible, la vidéo était très déformée lors de l'orthorectification ce qui empêchait totalement de trouver des débits pertinents.

5.3.3 Lissoir - Lesse - 10 au 19 juillet 2022

Cette campagne-ci était un peu différente des autres. En effet, elle ne s'est pas effectuée aux abords d'une station Aqualim comme les autres mais a été exécutée lors d'un camp Scout. Aucune information n'étant connue pour cet endroit, il a fallu procéder à la prise de mesure depuis le début, bathymétrie comprise.
Les vidéos ont alors été prises à l'aide d'un téléphone depuis la berge, où les points repères étaient représentés par des grosses pierres peintes en rouge. La hauteur d'eau était d'environ 30 cm. Le débit approximatif a, lui, été mesuré à l'aide d'une éponge et d'un chronomètre. La distance parcourue par l'éponge n'étant jamais constante, il a été très compliqué d'établir une valeur précise du débit. Il n'y avait aucune végétation présente sur le site mais les enregistrements n'ont jamais été effectués les même jours ni aux même heures, ce qui rend compliqué le résumé de la luminosité. Bien que dans l'ensemble la présence de reflets était assez rare.

5.3.4 Limelette - Dyle - 21 juillet 2022

Pour ce qui est de la dernière campagne de mesures, les vidéos ont été prises depuis un pont vers l'amont. La hauteur d'eau était de 50 cm et le débit Aqualim de $1.09[m^3/s]$. Ce jour-là, le temps était gris, de telle façon que les reflets sont moins flagrants et très limités.

FIGURE 5.2 – Photos des sites de mesure.

5.3.5 Bilan des mesures

On peut observer que les résultats obtenus avec l'application (visibles dans les sections suivantes) sont assez éloignés de la valeur réelle estimée par Aqualim, et souvent sous-estimés. Nous pensons que cette différence s'explique par plusieurs facteurs :

- Ayant déjà la bathymétrie des endroits susmentionnés via les travaux précédents, nous n'en avons pas effectué la mesure. Or, il est possible que le fond de ces cours d'eau ait été modifié sévèrement depuis lors, notamment lors les crues exceptionnelles de juillet 2021.
- Avec du recul, nous pensons que notre mesure de la profondeur n'a dans certains cas pas été assez rigoureuse, induisant des approximations alors que ce paramètre est très important dans le calcul du débit.
- Dans la majorité des cas, les vidéos prises étaient loin de présenter un point de vue orthogonal et nécessitaient une forte orthorectification, ce qui altère fortement les résultats.

Cependant, cela nous a permis de comprendre les enjeux liés à la prise de mesures sur le terrain et d'affiner notre approche, notamment lors de l'élaboration des différents protocoles présents en annexes. Pour l'analyse de l'influence de certains paramètres (Section 7), les valeurs ont donc été comparées entre elles et les vidéos plus anciennes correspondant aux bathymétries à notre disposition ont été utilisées.

Chapitre 6

Kit de mesure

Pour rappel, l'objectif recherché par le sujet de notre travail de fin d'études et véhiculé par l'application mise en place est de pouvoir mesurer le débit d'un cours d'eau de manière non-intrusive. Il est entendu par là de pouvoir se procurer le débit sans établir de contact avec la surface de l'eau.
Or à l'heure actuelle, si la vidéo n'est pas prise selon un point de vue orthogonal à la surface, le processus requiert de placer des points de référence et de mesurer les distances les séparant. Cela n'est évidemment pas idéal dans la mesure ou l'accès aux berges peut s'avérer difficile ou dangereux. Afin de pallier ce problème, l'idée d'un ***Kit de mesure*** a été imaginée sous la forme d'un dispositif rectangulaire qu'il serait possible de jeter dans l'eau. De cette manière, toutes les vidéos possèdent quatre points espacés de distances fixes et connues.

Ce chapitre relèvera le cahier des charges que doit remplir ce kit, les différents avantages et inconvénients d'un tel type de kit, notre première ébauche du kit et finalement l'analyse de vidéos utilisant celui-ci.

6.1 Cahier des charges

6.1.1 Élaboration des contraintes principales

Connaissant l'objectif recherché par la conception d'un kit, nous avons pu faire ressortir les contraintes principales devant être respectées par le kit :

- Les 6 distances entre les 4 points doivent être fixes et connues.
- Le kit doit pouvoir flotter à la surface de l'eau.
- Les coins du kit doivent se trouver le plus proche possible de la surface de l'eau

Connaissant ces caractéristiques, nous avons pu élaboré un prototype de ce dispositif répondant à tous les critères. Très simplement, nous avons monté un carré en bois de palette, sur lequel nous avons ajouté une diagonale afin de le rigidifier (Figure 6.1). Étant donné que nous avions prévu d'exécuter des mesures sur des rivières très modestes, nous avons estimé que 50 cm x 50 cm était une taille idéale. En effet, cela permet au cadre d'être léger et relativement peu encombrant (la diagonale mesure donc un peu plus de 70 cm). Les coins du dispositif étant représentatifs des points repères de l'orthorectification, ceux-ci ont été peints afin de les rendre plus visibles, mais cela restait optionnel. Lors de ses utilisations, le kit était retenu par une corde tenue par la personne qui insérait les traceurs dans le cours d'eau.

FIGURE 6.1 – Cadre servant de prototype au kit de mesure.

6.1.2 Test vidéo avec l'utilisation du kit

L'ébauche du kit répondant à toutes les contraintes principales, nous avons pu le tester en situation réelle6.2. Le kit s'est avéré être peu encombrant et peu lourd. Si son utilisation n'a pas réellement été mise à l'épreuve dans des conditions exigeantes (débits ou vitesses importants, vent soutenu, intempéries, etc.), nous n'avons rencontré aucune difficulté majeure. Tout au plus pouvons nous citer deux points qui ont retenu notre attention.

Toutes les vidéos tournées l'étaient vers l'amont. Une personne devait à chaque fois se tenir dans la rivière pour retenir le kit, ce qui est impensable en période de grande crue. Il serait bon d'imaginer une alternative dans ce cas de figure. De plus, à cause de ce hasard, le kit n'a jamais été testé tenu du pont ou de la berge, ce qui pourrait fournir de nouvelles conclusions.

D'autre part, il arrivait parfois que le kit gène le passage des traceurs, laissant une zone vide derrière lui. Il serait judicieux d'étudier l'impact de ces zones et de mettre en place des protocoles pour les limiter, par exemple via un placement judicieux.

FIGURE 6.2 – Kit en situation d'utilisation à Noirhat.

Nous avons ensuite effectué une série de tests sur les vidéos tournées lors de ces sorties à Noirhat, Suzeryl et Limelette [1].
En modifiant quelque peu les lignes de code, nous avons fixé la région d'orthorectification ainsi que le ROI. Ainsi, nous pouvions comparer l'orthorectification à l'aide du kit ou à l'aide des GCPs tout en gardant ces données identiques entre les deux analyses. Les tableaux 6.1, 6.2, 6.3 reprennent donc les tests effectués sur les trois stations, où chaque test représente une région d'orthorectification différente :

Noirhat		
Débit réel = 0,137 m^3/s	Sans Kit	Avec Kit
Test 1	0,1132 m^3/s	0,1003 m^3/s
Test 2	0,1182 m^3/s	0,1018 m^3/s
Test 3	0,0869 m^3/s	0,093 m^3/s
Test 4	1,5070 m^3/s	1,644 m^3/s

TABLE 6.1 – Tests effectués sur le site de Noirhat

1. Les vidéos utilisées sont la vidéo GX018582 de Noirhat (voir figure 2 de l'annexe .4), la vidéo X018625 de Suzeryl (voir figure 3 de l'annexe .4), la vidéo X019092 de Limelette (voir figure 5 de l'annexe .4).

Suzeryl		
Débit réel = 0, 19 m^3/s	Sans Kit	Avec Kit
Test 1	0,1240 m^3/s	-
Test 2	0,0947 m^3/s	0,0806 m^3/s
Test 3	0,1148 m^3/s	0,0383 m^3/s
Test 4	-	-

TABLE 6.2 – Tests effectués sur le site de Suzeryl

Limelette		
Débit réel = 1,086 m^3/s	Sans Kit	Avec Kit
Test 1	0,8462 m^3/s	-
Test 2	0,9298 m^3/s	0,8827 m^3/s
Test 3	0,7097 m^3/s	0,6907 m^3/s
Test 4	0,9121 m^3/s	0,9244 m^3/s

TABLE 6.3 – Tests effectués sur le site de Limelette

Après un regard sur les résultats, on peut faire une première constatation assez directe : dans certains cas d'utilisation du kit, l'orthorectification effectuée n'a pas permis le calcul du débit.
Une deuxième observation qui peut se faire est que les résultats oscillent beaucoup par rapport à la valeur réelle du débit. Ceci implique que la région d'orthorectification a plus d'importance que l'on ne pourrait le penser.
La dernière remarque que l'on peut émettre est que même si certains résultats de l'utilisation du kit sont prometteurs, il semble que les résultats obtenus à l'aide des GCPs, soient plus proches de la réalité. Cependant, cette différence, notamment observée sur les sites de Suzeryl et de Limelette, peut être expliquée par l'angle de la prise de mesure ou encore par une largeur plus imposante du cours d'eau. De fait, l'angle de prise de mesure étant tellement fermé sur le site de Suzeryl, le processus d'orthorectification est important et possède donc une plus influence. Sur le site de Limette, l'orthorectification n'est pas spécialement importante mais le kit semble très petit par rapport à la largeur de la rivière. Il est donc plus compliqué de sélectionner précisément les quatre repères et il arrive alors souvent que les dimensions ne correspondent pas à la sélection. Lorsque l'on utilise les GCPs, cela n'est pas fatal de se tromper de quelques pixels et donc de quelques centimètres au vu des relativement grandes distances qui séparent les GCPs. Mais si l'on utilise un kit de 50 cm de côté, se tromper de quelques pixels aura proportionnellement beaucoup plus d'impact.

Pour conclure, il paraît évident que la première ébauche du kit n'est pas optimale. Néanmoins, il s'avère au vu des résultats que son utilisation reste prometteuse. Il conviendra alors d'améliorer le kit actuel pour en parfaire son utilisation. Les

contraintes secondaires énoncées ci-après pourraient constituer, dans un premier temps, une amélioration considérable à de futurs prototypes.

6.1.3 Élaboration de contraintes secondaires

Les contraintes secondaires ne sont pas nécessaires à la bonne utilité du kit mais représentent un avantage certain pour son utilisation ou son transport :

- Le kit doit rester léger pour conserver un aspect pratique.
- Le kit pourrait se décliner en différentes dimensions selon la taille du cours d'eau ou du point de vue que l'utilisateur peut en avoir.
- Le kit pourrait être montable et démontable, voire même dépliable.
- Le kit pourrait résider dans une boîte de protection.
- Si une boîte est à disposition, elle permettra d'y ranger les différents éléments essentiels à une bonne prise de mesure (voir section 5.1).
- Étant donné que les dimensions du ou des kits sont connues, un bouton "***Kit***" pourrait être ajouté à l'application qui utiliserait alors des données pré-encodées, ce qui présenterait un gain de temps.
- On pourrait également imaginer que le kit dispose d'un dispositif pour mesurer la vitesse de surface et/ou le débit.

6.2 Bilan

En plus de la raison principale pour laquelle le kit a été pensé, il s'est révélé, au fur et à mesure des nombreuses utilisations de l'application, que celui-ci possède plus d'un avantage :

- L'utilisation d'un kit de mesure octroie en toute circonstance la possibilité de procéder à l'orthorectification.
- Même si le placement des GCPs est envisageable, le kit permet de se passer de la mise en place de ceux-ci ainsi que de la mesure des distances qui les séparent. C'était de loin l'étape la plus fastidieuse et la plus longue lors d'une prise de mesure.
- Sachant que le kit flotte et que celui-ci est peu épais, les quatre points repères seront au plus proche de la surface de l'eau, ce qui est une des conditions de pose des GCPs.

Même si le kit comporte des avantages considérables, son utilisation possède néanmoins quelques inconvénients :

- L'utilisation de matériel à même la surface de l'eau va l'encontre de la non-intrusivité que se veut véhiculer l'application.
- L'obtention complète du débit ne dépend plus seulement d'une application. En effet, les GCPs pouvant être des éléments naturels, nous possédions seulement l'application pour la mesure du débit. Avec l'emploi d'un kit, le matériel nécessaire devient un peu plus encombrant.
- Sa taille moindre étant un avantage par rapport à son transport, celle-ci se trouve être un inconvénient quant au placement des points de référence lors de l'orthorectification. En effet, une erreur de placement de quelques pixels aura beaucoup plus d'impacts que sur des distances de plusieurs mètres.

En conclusion, l'utilisation de ce prototype s'est avérée prometteuse mais requérant un soin plus grand que des mesures classiques basées sur les GCPs. L'étendue des cas où le kit fonctionnera est probablement un peu moins large que l'application en elle même mais il compense par une facilité d'utilisation accrue.

Chapitre 7

Influence de certains paramètres

Dans cette section, nous allons étudier l'influence que peuvent avoir certains paramètres, que ce soit au moment de la prise de vidéo ou de l'utilisation de l'application et ce afin d'établir quels sont les paramètres à même de fournir le meilleur champ de vecteurs et le débit le plus précis selon les caractéristiques de chaque vidéo.

Remarque 32. *Dans la suite de ce chapitre, la ligne orange, présente dans les graphiques, fait référence à la valeur supposée réelle du débit, extraite du réseau Aqualim.*

7.1 Paramètres liés à la prise de mesure

Les paramètres dépendant de l'utilisateur **au moment de la prise de vidéo** seront étudiés dans cette section. Les tests seront donc menés sur des vidéos différentes, ce qui induit de facto des biais. En effet, une seule caméra ne peut pas filmer deux fois le même endroit au même moment.

Dans certains cas, les vidéos ont donc été prises au même moment mais avec une caméra différente. Bien que nous ayons essayé de réduire la différence au niveau des réglages, nous ne pouvons pas exclure que les différences dans les résultats obtenus soient liés en partie à une différence inhérente aux caméras (un angle de vue légèrement différent, une mise au point en différents endroits, etc.).

Dans d'autres cas, les vidéos comparées ont été filmées avec la même caméra, mais successivement. Ici aussi, les biais pouvant survenir sont multiples : changement de luminosité, répartition différente des traceurs, etc.

Les résultats finaux de nos analyses sont donc à prendre avec des pincettes et à confirmer avec de futurs sets de vidéos supplémentaires.

7.1.1 Résolution d'image

Pour évaluer la différence dans les débits obtenus en fonction de la résolution de la vidéo, nous avons effectué trois séries de 10 tests sur deux vidéos de Limelette : une vidéo en FullHD(1920x1080) et une en UltraHD(3840x2160) testée avec deux aires d'interrogation différentes. Ces vidéos ont été prises au même moment.

La distinction entre les aires d'interrogation s'explique par la volonté d'inclure une comparaison à nombre de vecteurs égaux. Dès lors que les largeurs d'images étaient différentes d'un facteur 2, les aires d'interrogation devaient suivre ce même rapport pour produire un nombre de vecteurs similaire.

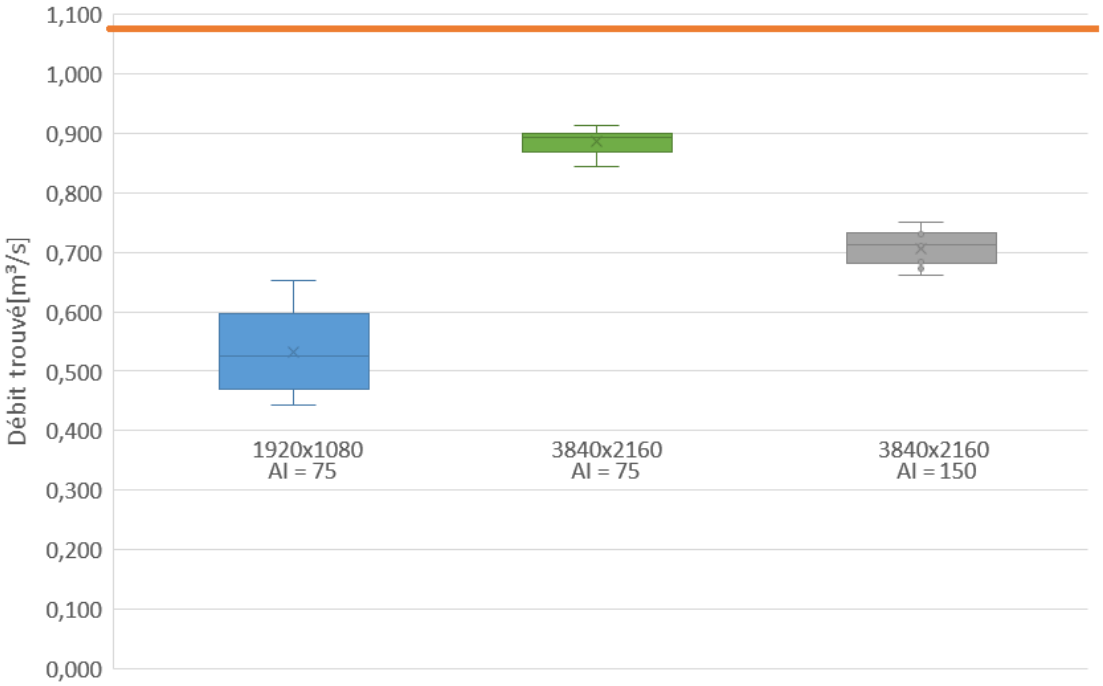


FIGURE 7.1 – Résultats obtenus pour des résolutions d'image différentes.

Les résultats (Figure 7.1) montrent clairement que la vidéo prise en moins bonne qualité donne des résultats moins bons et plus variables que celle en UltraHD. Il semble donc qu'une meilleure résolution fournisse un résultat meilleur, ou du moins plus élevé et ce, quelle que soit la taille de l'aire d'interrogation considérée.

D'autres vidéos comparatives ont été tournées à Noirhat et Suzeryl, mais elles sont inexploitables : dans le premier cas une différence de réglages entre les caméras

entrave la comparaison et dans le second l'orthorectification nécessaire corrompt les résultats (Section 5.3.2). D'autres tests s'avèrent nécessaires pour vérifier ce premier résultat.

Remarque 33. *Il semble y avoir une différence entre les deux séries calculées sur la même vidéo avec des aires d'interrogation différentes : la série avec ce paramètre plus bas montre un meilleur résultat. En ce sens, cette vidéo paraît différer de l'analyse menée au point 7.2.1.*

7.1.2 Stabilisation

Parmi les vidéos exploitables des sites de Noirhat et de Limelette, certains tests ont été filmés par deux caméras. Une était tenue en main alors que la deuxième reposait sur un trépied. Pour une même répartition des traceurs et une même luminosité extérieure, on disposait alors d'une vidéo stabilisée (celle sur le trépied) et d'une vidéo non-stabilisée (celle tenue en main).
Bien que certains paramètres peuvent encore différer entre les deux prises (angle de prise, paramètres intrinsèques de la caméra, etc), il est possible d'effectuer une série de tests afin de tester l'efficacité de la stabilisation via l'application.

Pour effectuer cette comparaison, nous avons exécuté une série de dix tests sur une vidéo stable ainsi que sur sa variante stabilisée à l'aide de l'application. Cette opération a été répétée sur un couple de vidéo de Noirhat et de Limelette. Les débits obtenus sont montrés aux graphes 7.2 et 7.3.

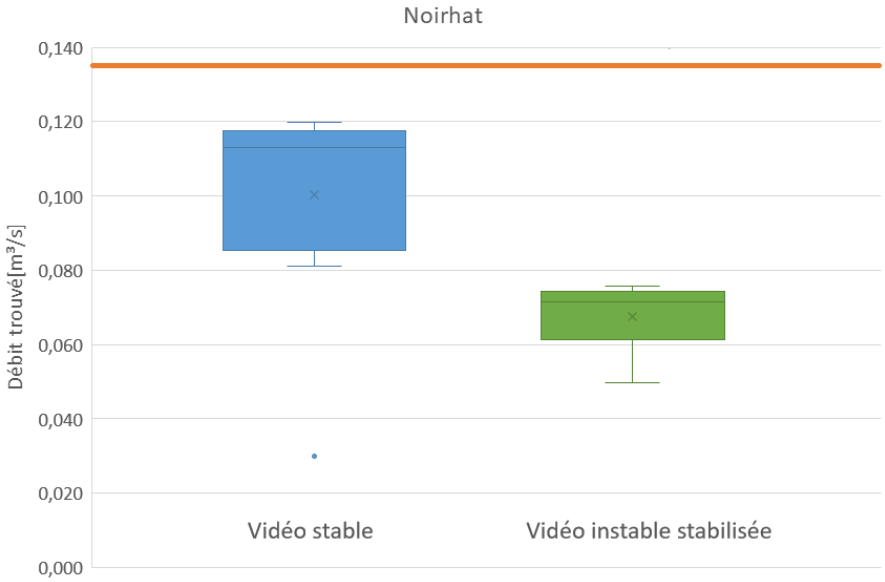


FIGURE 7.2 – Résultats obtenus pour une vidéo initialement stable (bleu) et une vidéo stabilisée à l'aide de l'application pour le site de Noirhat.

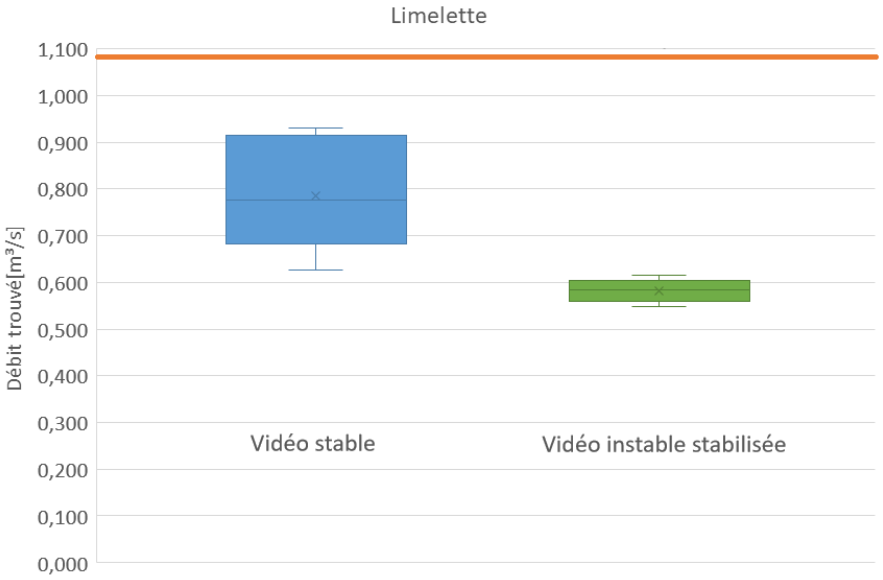


FIGURE 7.3 – Résultats obtenus pour une vidéo initialement stable (bleu) et une vidéo stabilisée à l'aide de l'application pour le site de Limelette.

Deux observations majeures peuvent être faites.
La première est que la vidéo stable initialement est plus encline à fournir des résultats proches du débit réel. A l'inverse, la vidéo stabilisée à l'aide de l'application fournit nécessairement des valeurs nettement inférieures à celle recherchées.

La seconde observation réside dans la faculté de fournir des résultats plus ou moins constants. En effet, on peut apercevoir que dans le cas de la vidéo stabilisée via l'application, les résultats varient indéniablement moins que dans le cas de la vidéo stabilisée antérieurement.

Une dernière observation, qui ne réside pas sur le graphique, peut également être faite. De fait, lorsque la stabilisation est entreprise à l'aide de l'application, une version stabilisée de la vidéo initiale est créée. Vidéo qui baisse alors fortement en résolution. Selon la conclusion faite à la section ci-dessus (Section 7.1.1), il est alors normal de constater des moins bons résultats pour ces vidéos.

Pour conclure, le processus de stabilisation semble fonctionner correctement vu la faible variation des résultats. Néanmoins, les modifications apportées à la vidéo, telle que la résolution, s'avèrent être rédhibitoires quant à l'obtention de résultats corrects.

7.1.3 Utilisation du kit

Le résultat d'une analyse performée avec un kit de mesure réside presque entièrement sur la qualité de l'orthorectification. En effet, étant le but premier du kit, il est évident que chaque changement de conception ou d'utilisation de celui-ci pourrait constituer une variation importante de la qualité de l'orthorectification et donc du débit calculé. Pour vérifier quelle utilisation du kit est la meilleure, nous avons comparé la position du kit sur les sites de Noirhat et Limelette.

Pour entreprendre ces mesures, nous avons pris différentes vidéos lorsque le kit se trouvait en haut de l'image, en bas de l'image ou centré par rapport à l'image. Les valeurs des débits obtenus sont représentées sur les graphes 7.4 et 7.5.

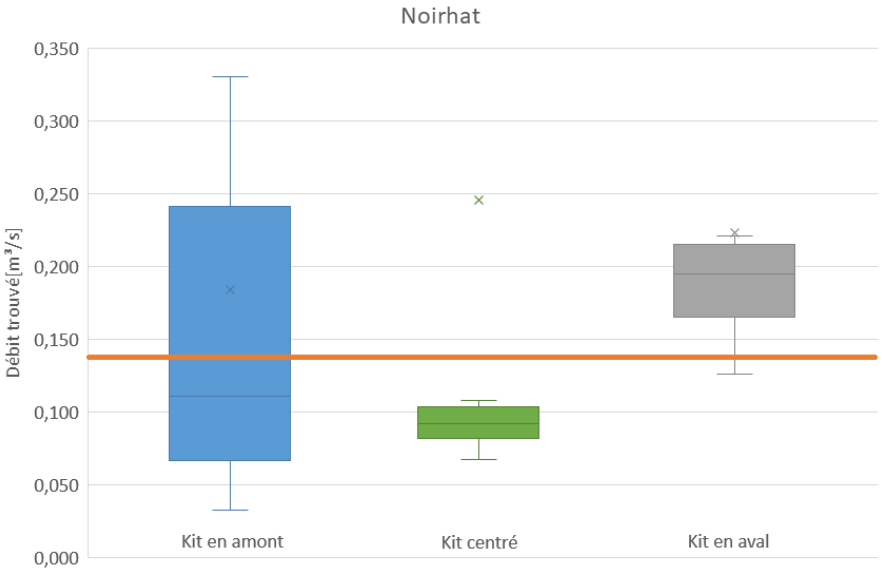


FIGURE 7.4 – Résultats obtenus en fonction de la position du kit sur le site de Noirhat.

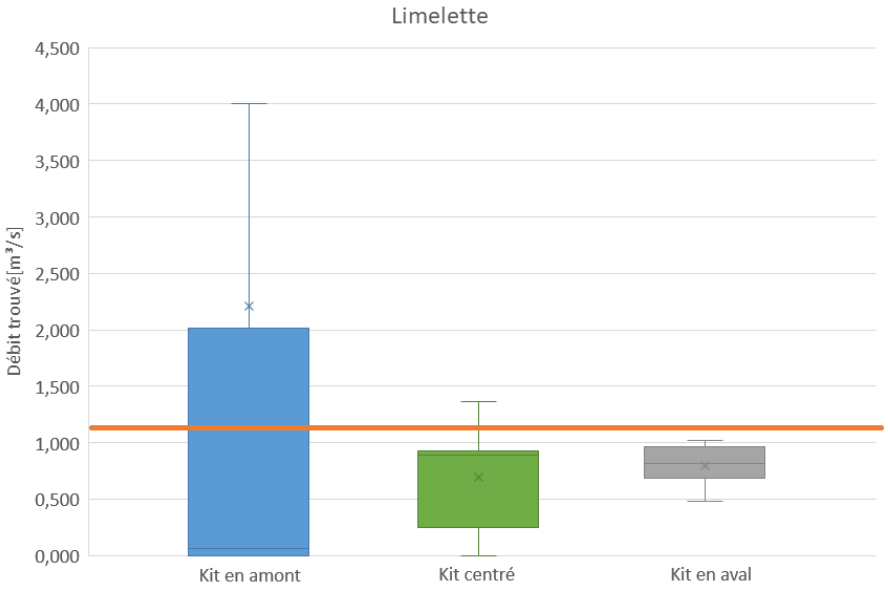


FIGURE 7.5 – Résultats obtenus en fonction de la position du kit sur le site de Limelette.

Tant sur le site de Noirhat que de Limelette, on peut apercevoir que la situation où le kit se trouve en amont de l'image fournit des résultats assez divergents voire inexistants. Cette disparité peut être expliquée par une orthorectification compliquée. De fait, le kit étant loin (et donc assez petit), le placement précis des points de référence est moins précis. De plus, le kit ne faisant que 50 cm de côté, une erreur de quelques pixels dans la sélection des points de repère représente vite une grosse différence en terme de distance et mène donc généralement à une mauvaise orthorectification (Figure 7.6) et par conséquent à un débit incorrect.

De manière générale, lorsque le kit est loin, les extrémités de celui-ci sont difficilement différentiables car il est presque filmé de profil. Cela demande donc à l'application d'effectuer une orthorectification très importante et ajoute dès lors un degré d'imprécision. C'est pourquoi on peut voir une nette amélioration en terme de régularité et de précision lorsque le kit descend sur l'image sur le site de Limelette (Figure 7.5).

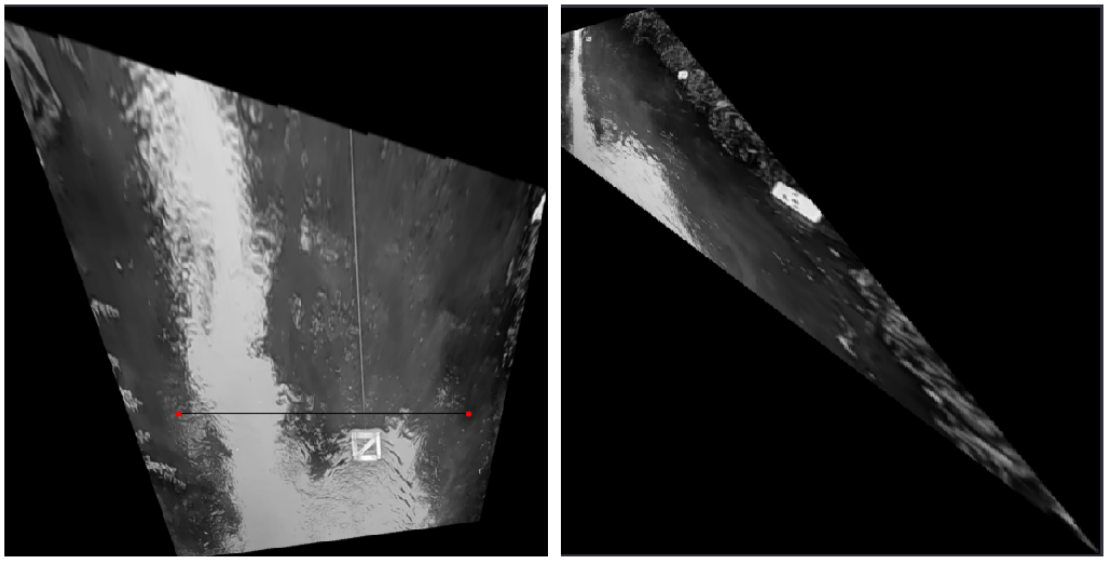


FIGURE 7.6 – Résultat de l'orthorectification pour un kit centré (à gauche) et pour un kit en haut de l'image (à droite)

On pourrait dès lors s'attendre à apercevoir le même phénomène sur le site de Noirhat, mais il n'en est rien. Dans ce cas-ci, c'est le cas du kit centré par rapport à l'image qui présente une meilleure régularité et précision (Figure 7.4). Ceci peut être expliqué par le fait que le milieu de l'image se trouve également être le milieu de l'aire orthorectifiée. L'aire entière se trouve alors moins déformée et donc les vecteurs également. Le débit est alors plus fiable à la réalité.

Pour résumer les observations faites ci-dessus, il semble qu'il y ait deux paramètres qui fassent varier la qualité de l'orthorectification via le kit.
Le premier est le degré d'orthorectification, dans le sens où plus l'image doit être modifiée ou déformée pour être remise à plat plus l'orthorectification est sujette à être erroné.
Le second est la proximité de l'aire orthorectifiée par rapport au kit. En effet, l'angle de vue pourrait correspondre parfaitement au kit. Cependant, si l'aire est loin du kit celle-ci subira quand même une orthorectification importante et apportera son lot d'imprécision lors du calcul du débit.

Dès lors, on ne peut pas affirmer qu'il y ait une meilleure position qu'une autre pour placer le kit de mesure. Néanmoins, il convient de trouver le juste milieu entre proximité de l'aire à orthorectifier et le degré d'orthorectification par rapport à l'angle de prise de vue afin d'obtenir les résultats les plus constants et proches de la réalité.

7.2 Paramètres liés à l'utilisation de l'application

7.2.1 Aire d'interrogation

Pour rappel, l'aire d'interrogation est un carré qui, lors de l'analyse PIV entre deux images, est issu de la première image, et dont la correspondance sera recherchée dans l'aire de recherche de la seconde image. L'aire de recherche correspond donc à une zone qui sera parcourue par l'aire d'interrogation afin de trouver la meilleure corrélation et ainsi établir un vecteur vitesse (voir section 2.3 pour des explications plus détaillées). Ces zones sont introduites dans OpenPIV via la longueur des côtés, et sont donc exprimées en pixels bien qu'étant des aires. Par défaut, la taille de la zone de recherche est la même que la zone d'interrogation :"default is the same interrogation window size and it is a fallback to the simplest FFT based PIV"[43]. Ceca est adapté car étant donné le court laps de temps entre deux images et la vitesse relativement faible d'un débit comparé à cette durée, on s'attend à ce que les éléments de la zone d'interrogation restent relativement proches. Le chevauchement ou overlap en anglais est le nombre de pixels entre les centres de zones d'interrogation. Il est généralement fixé à 50% du côté de l'aire d'interrogation.

Effets attendus

Faire varier l'aire d'interrogation aura donc pour conséquence de faire varier également l'aire de recherche. Une grande aire d'interrogation aura pour effet de créer moins de vecteurs. L'écoulement y est discrétisé plus grossièrement, et dès lors moins précisément. A contrario, une aire d'interrogation petite permet de créer beaucoup de vecteurs. Si cela est présumé plus précis, cela consomme également plus de temps de calcul. De plus, il est possible qu'avec une aire trop petite, les éléments de la zone d'interrogation de la première image soient sortis de la zone de recherche de la seconde image. Il sera alors impossible pour l'analyse PIV de trouver la bonne zone et de générer le bon vecteur. Ce phénomène est décrit par Bia T. et Somers H.[23].

Résultats

On observe en effet que prendre une zone d'interrogation trop petite génère des vecteurs aléatoires (Figure 7.7). Le temps de calcul est comme prédit décuplé, passant de quelques dizaines de secondes à quelques minutes voire quelques dizaines de minutes pour les mêmes extraits de vidéo.

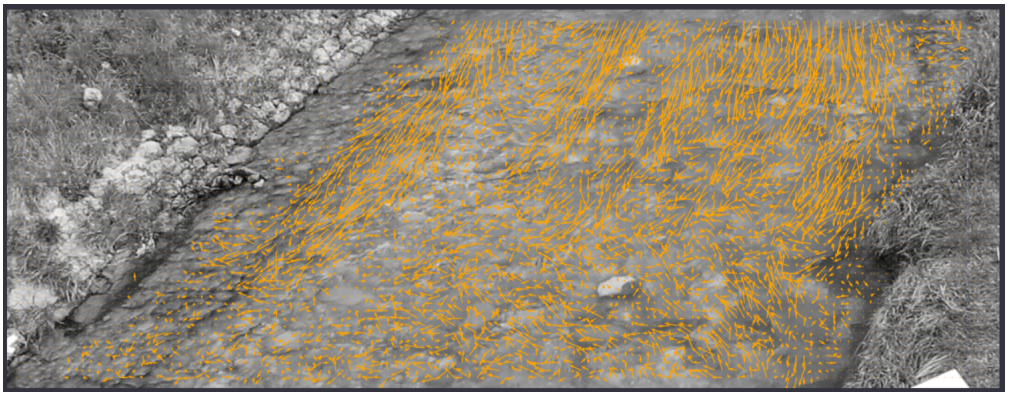


FIGURE 7.7 – Illustration du problème rencontré avec une petite aire d'interrogation.

Quand la zone est trop grande, un autre phénomène apparaît. Les vecteurs étant moins nombreux, il arrive que le nombre de sous-sections demandé par l'utilisateur (Section 2.5) soit supérieur au nombre de vecteurs effectivement présents sur la largeur. Dans ce cas de figure, l'application doit combler le manque de vecteurs et sélectionne un ou plusieurs vecteurs sur une ligne supérieure ou inférieure de la grille d'interrogation (Figure 7.8). On peut donc observer que pour des aires d'interrogation importantes (>7-8% de la taille de l'image), une partie des vecteurs considérés sont alors éloignés de la ligne noire représentant la section (Figure 7.8).

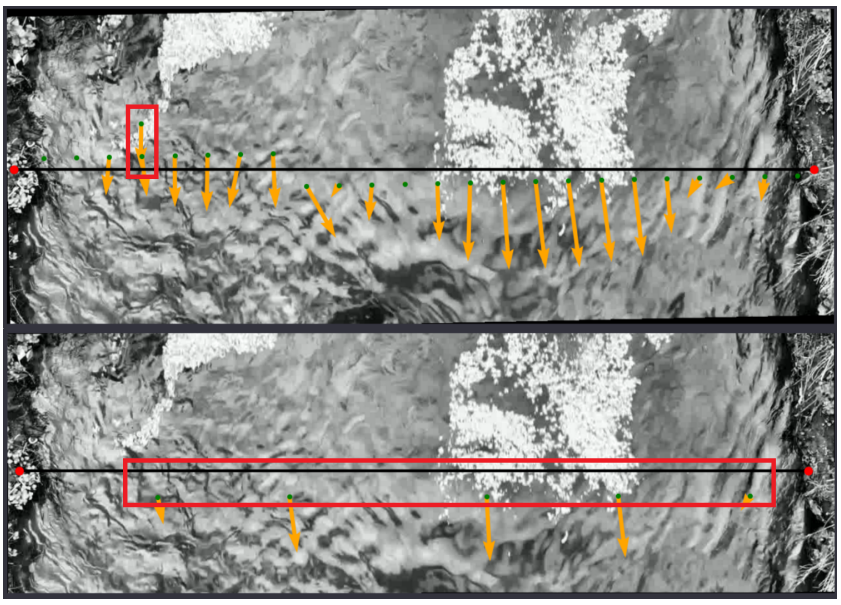


FIGURE 7.8 – Illustration des problèmes rencontrés avec une grande aire d'interrogation.

Cela peut constituer un problème car l'influence du placement de la section sur l'image n'est pas à négliger sur le résultat final. En outre, un trop faible nombre de vecteurs peut amener à faire crasher l'application si le nombre de sous-sections est trop important comparativement (Section 4.4.1).

Il y a donc un équilibre à trouver entre ces 2 configurations. Pour approcher cet équilibre, nous avons effectué des tests sur quatre vidéos issues de quatre cours d'eau différents : Noirhat, Limelette, la vidéo 7 du Video Globe Challenge et Lissoir. Pour cette dernière le débit réel n'était pas connu. Lors de ces tests, tous les autres paramètres, à savoir le step, les temps extraits de la vidéo, les points d'orthorectification et le masque associé, la ROI et le masque associé, les filtres, l'endroit de la section et le nombre de sous-sections étaient bloqués, seule l'influence de la zone d'interrogation influençait donc les différences dans les résultats.

Les débits obtenus sont représentés sous forme de graphiques à la figure 7.9.

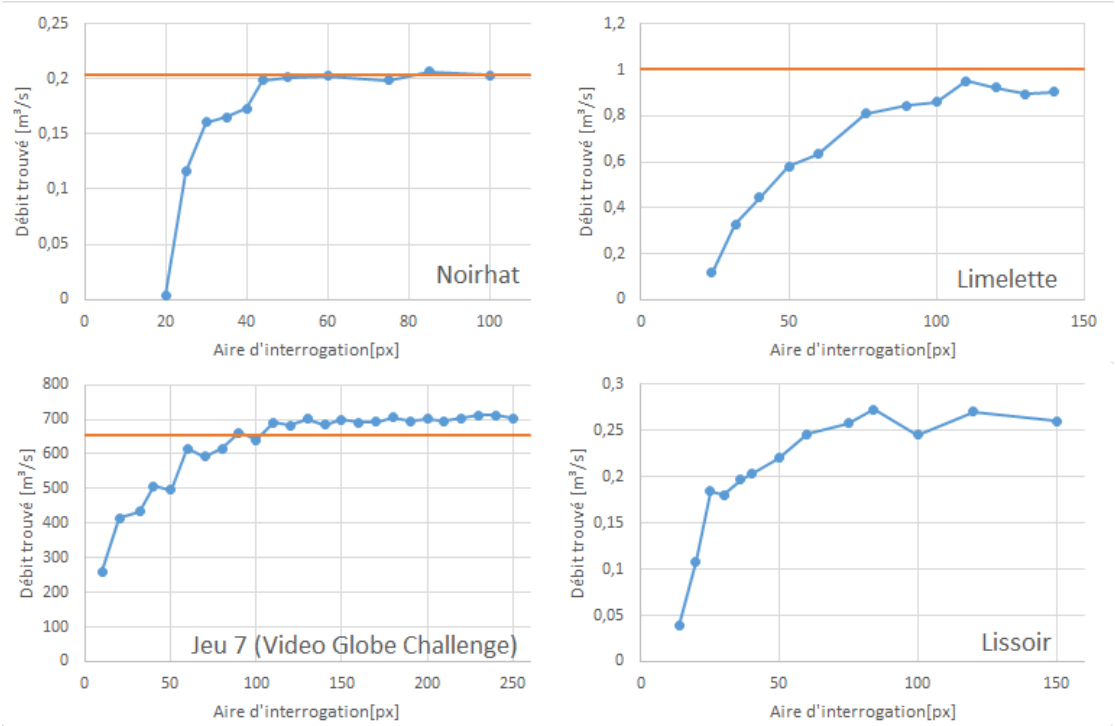


FIGURE 7.9 – Débits obtenus en faisant varier l'aire d'interrogation pour 4 cours d'eau. Pour rappel, la ligne orange représente le débit réel.

Au regard de ces graphes, une tendance se dégage clairement : pour les plus petites aires d'interrogation, le débit est fortement sous-évalué. Au fur et à mesure que l'on augmente ce paramètre, le débit augmente jusqu'à atteindre un plateau relativement proche du débit réel.

Contrairement à ce qui avait été attendu, augmenter le nombre de vecteurs ne fournit pas un résultat plus précis. Vouloir atteindre la plus grande précision en réduisant l'aire d'interrogation pour obtenir le maximum de vecteurs ne semble donc pas être la solution adéquate. A cela s'ajoute également la perte de temps considérable que cela induit.

La baisse de précision supposée pour de grandes zones d'interrogation n'a donc pas été observée ici : cela peut s'expliquer par la fixation du nombre de sous-sections à une valeur qui ne permet pas de discrétiser trop grossièrement la section (20 ou 25 sous-sections selon les vidéos).

Afin de mieux se rendre compte de cette réalité, nous avons divisé l'aire d'interrogation considérée par la largeur de la vidéo (en pixels). De même, les résultats obtenus par RiverApp ont été divisés par le débit réel (Figure 7.10).

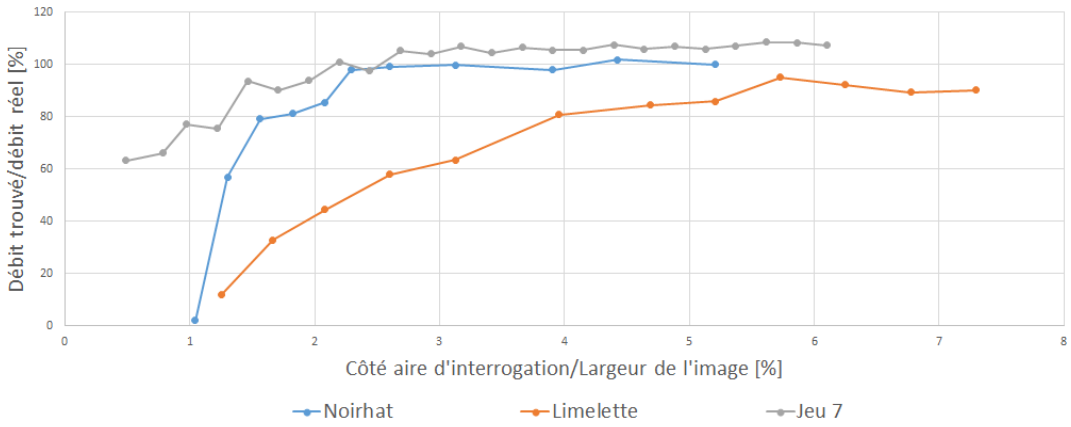


FIGURE 7.10 – Résultats ré-exprimés en pourcentages.

Sur base de l'analyse de ces trois vidéos, nous pouvons établir que le débit atteint 80 % du débit réel à partir d'une aire d'interrogation de côté correspondant à 4% de la largeur de l'image et même avant (environ 2%) pour deux vidéos. Pour information, 4% de la largeur de l'image correspondent respectivement à 77 pixels et 164 pixels pour des images larges de 1920 et 4096 pixels et qui sont des résolutions courantes pour des vidéos. La valeur par défaut étant de 32 pixels, nous pensons que cela pourrait induire les nouveaux utilisateurs en erreur (Section 4.5).

Par ailleurs, pour la vidéo de Lissoir, bien que nous n'ayons pas le débit réel pour ce site, l'allure du graphe obtenu reste la même que les autres, et affiche ainsi un plateau autour de la valeur de 0.26 $[m^3/s]$. En considérant cette valeur, la courbe correspondant à cette vidéo s'intègre dans le graphe en figure 7.11.

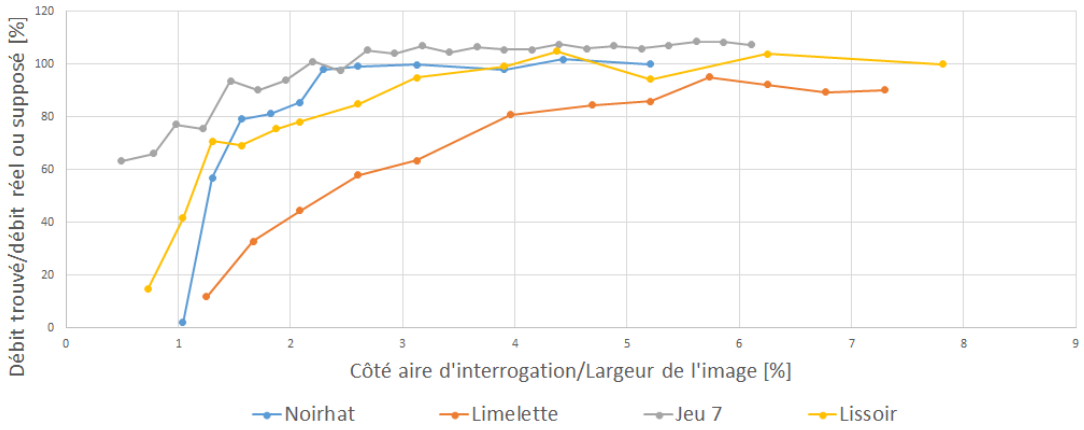


FIGURE 7.11 – Résultats exprimés en pourcentage, Lissoir a été rajouté avec un débit supposé de 0.26 $[m^3/s]$.

On peut en conséquence raisonnablement penser que 0.26 $[m^3/s]$ est une valeur qui approxime le débit sur cette vidéo. Sur base de notre expérience et en comparant avec d'autres cours d'eau à notre disposition, cela semble tout du moins ne pas être une valeur invraisemblable.

7.2.2 Nombre de sous-sections

Effets attendus

À l'image de l'aire d'interrogation, choisir le nombre de sous-sections revient à déterminer un niveau de discrétisation de la section (Section 2.5). On s'attend donc à ce qu'un faible nombre de sous-sections soit moins précis qu'une section discrétisée moins grossièrement. Cependant, un trop grand nombre de divisions par rapport à la quantité de vecteurs pourrait mener à un phénomène déjà décrit dans la section 7.2.1 : les vecteurs choisis pour représenter une subdivision du profil pourraient alors être sélectionnés relativement loin de la section choisie, diminuant ainsi la précision.

Résultats

Cette fois encore, les tests ont été conduits en bloquant tous les paramètres pour la vidéo et en ne faisant varier que le nombre de sous-sections.

L'aire d'interrogation a été choisie en fonction du test réalisé à la section précédente : le plus petit possible afin de pouvoir considérer un maximum de sous-sections, mais assez élevé pour donner un débit correct.

Le phénomène attendu pour une quantité importante se produit effectivement (Figure 7.12), toutefois un vecteur isolé n'aura que peu d'influence sur le débit final étant donné le nombre élevé de vecteurs proches de la section.

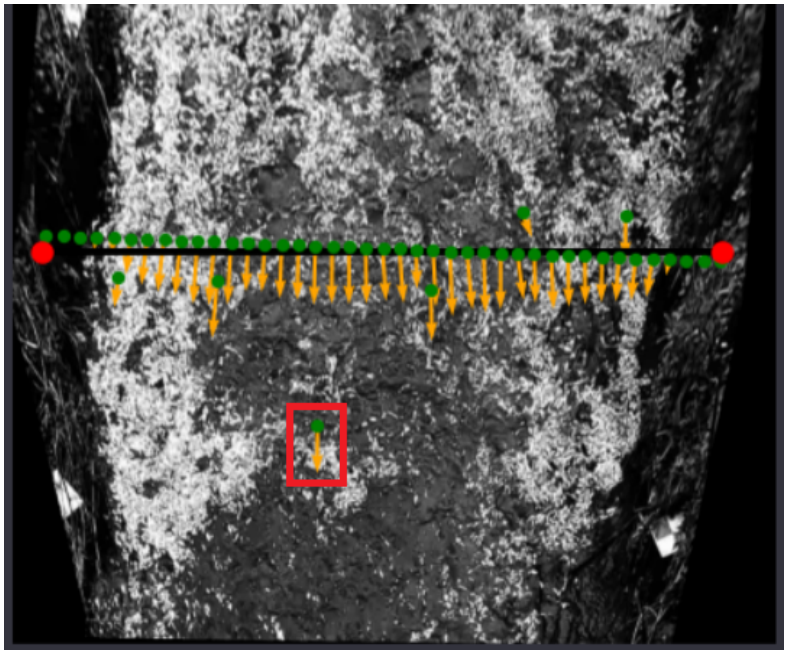


FIGURE 7.12 – Illustration du phénomène se produisant lorsque le nombre de sous-sections est trop élevé par rapport au nombre de vecteurs générés.

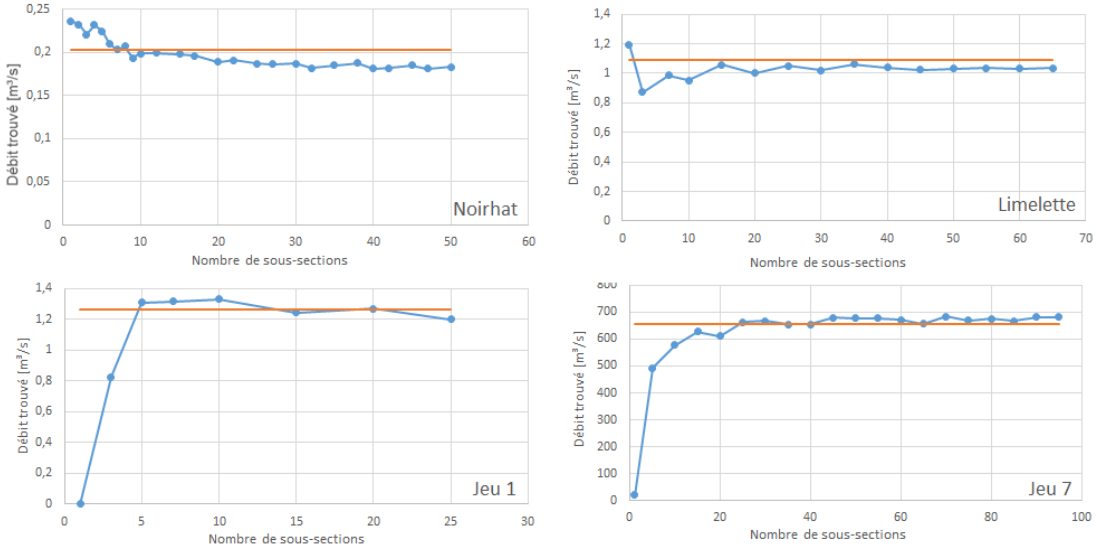


FIGURE 7.13 – Résultats obtenus en faisant varier le nombre de sous-sections.

Ces résultats, illustrés en figure 7.13, montrent, en effet, un résultat pauvre pour un faible nombre de divisions, mais qui s'approche rapidement de la valeur réelle. On observe aucune baisse de précision quand ce nombre continue d'augmenter.

Pour Noirhat, le débit trouvé semble se stabiliser en deçà de la valeur réelle. Nous supposons que la section fixée pour ce test n'était pas située à l'endroit idoine. Un plateau est toutefois observé comme sur les autres vidéos.

Sur base de l'analyse de ces quatre vidéos, nous pouvons estimer qu'il n'est pas nécessaire de chercher à maximiser le nombre de sous-sections en vue d'augmenter la précision.

De fait, le nombre minimum de sous-sections produit un résultat tout à fait honorable et qui, de plus, semble corrélé avec la taille de l'aire d'interrogation utilisée. Le plateau est alors atteint pour 5 sous-sections sur la vidéo du jeu 1 (aire d'interrogation utilisée : 50) et environ 25 pour la vidéo du jeu 7 (aire d'interrogation utilisée : 100). En ce sens, la valeur instaurée par défaut (20) semble pertinente, mais il faudra veiller à l'augmenter pour de grandes vidéos, où de grandes aires d'interrogation seront potentiellement utilisées.

7.2.3 Placement de la section de mesure

Dès nos premières mesures, nous nous sommes rendu compte que les endroits où l'on plaçait les points de la section de mesure dans l'application (Section 3.2.18) pouvaient faire varier considérablement le débit obtenu.

Effets attendus

Il est assez difficile de prévoir les effets d'une telle variation. On peut tout de même estimer que le meilleur résultat doit se situer à l'endroit où la bathymétrie a été mesurée, donc au milieu des 4 repères GCP dans la majorité des cas. De plus, les résultats obtenus loin de cette section devraient être les plus mauvais. Cette observation est marquée d'avantage si la vidéo a été orthorectifiée, d'autant plus quand le degré d'orthorectification est fort (c'est à dire que la vidéo a été prise avec un angle faible et que l'orthorectification déforme beaucoup l'image).

Résultats

Nous avons fait varier la hauteur de ces points par pas de 50 pixels sur les images orthorectifiées (Figure 7.14), encore une fois en gelant tous les autres paramètres. Les vidéos avec des berges assez parallèles ont été préférées à d'autres pour permettre de ne modifier que la hauteur de mesure.

FIGURE 7.14 – Variation de la hauteur introduite sur une vidéo de Noirhat. L'image a été orthorectifiée au préalable.

Remarque 34. *La hauteur est ici mesurée en pixels à partir du haut de l'image, l'origine du repère orthonormé pour les images étant situées en haut à gauche. Le repère ici décrit est donc bien différent de celui énoncé lors de l'orthorectification.*

Ces vidéos sont (Figure 7.15) :

- Jeu 7 non orthorectifié
- Noirhat orthorectifié
- Limelette orthorectifié
- Limelette pris au drone, orthorectifié très faiblement

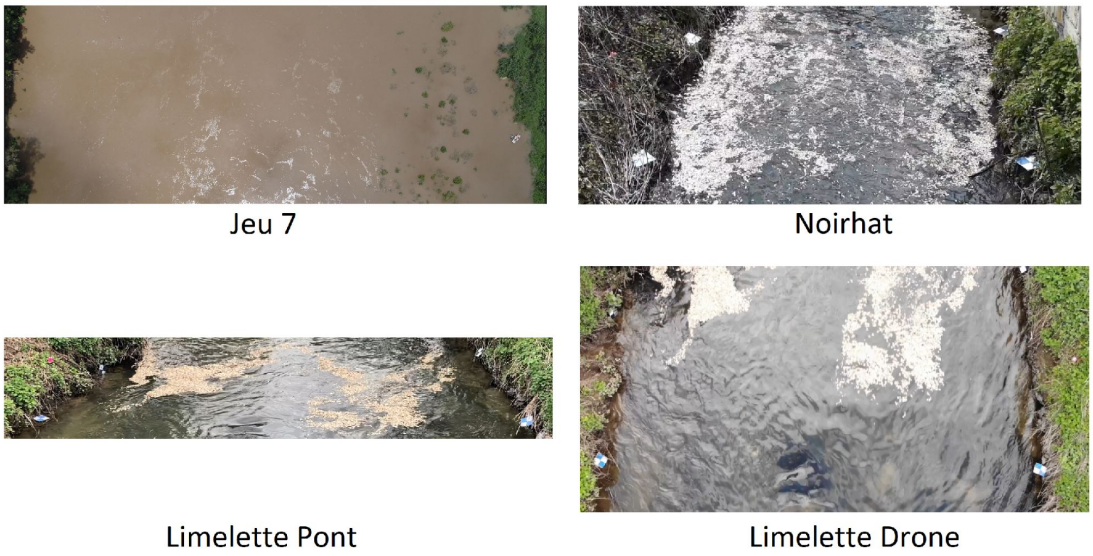


FIGURE 7.15 – Images tirées des vidéos utilisées (avant orthorectification).

Les résultats obtenus sont présentés sur la figure 7.16.

Pour la vidéo du jeu 7, on observe une large zone donnant de bons débits, la vidéo n'étant pas orthorectifiée et le courant très peu turbulent. Aux extrémités de l'image, les débits trouvés sont un peu plus aléatoires. Pour les vidéos de Noirhat et Limelette prises du pont, on observe clairement l'impact de l'orthorectification prononcée : le débit exact n'est atteint que près de la zone de la section de mesure et plus on s'en éloigne, moins le débit est précis.

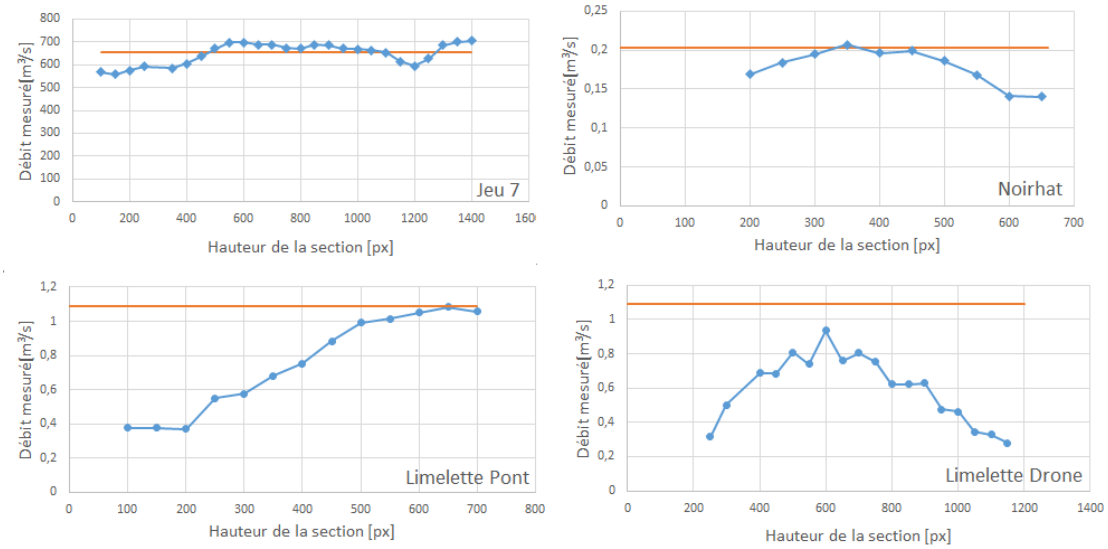


FIGURE 7.16 – Résultats de la variation de la hauteur introduite.

Le même phénomène se produit sur la vidéo de Limelette qui n'a été que peu déformée. En outre, le débit semble très variable quand la section d'intérêt est placée proche du bon endroit. Pour mieux observer cette singularité, nous avons conduit d'autres tests, en faisant varier la hauteur par 25 et 10 pixels sur la zone d'intérêt (Figure 7.17).

On observe que, bien que les débits se trouvent toujours dans le même intervalle, ils sont très variables d'une section à l'autre. On mesure une différence de 0.1181 $[m^3/s]$ (soit près de 11% du débit réel) entre deux sections situées à 10 pixels d'écart. Cette différence dans la zone où devrait se situer la bonne section est interpellante et suggère que selon le cours d'eau mesuré, la section peut jouer un grand rôle sur la précision de l'application.

Il faut donc être extrêmement précautionneux au moment du placement des points de la section. De plus, si de nouvelles mesures sont effectuées, nous suggérons de mettre en évidence la section de mesure afin de pallier ce problème.

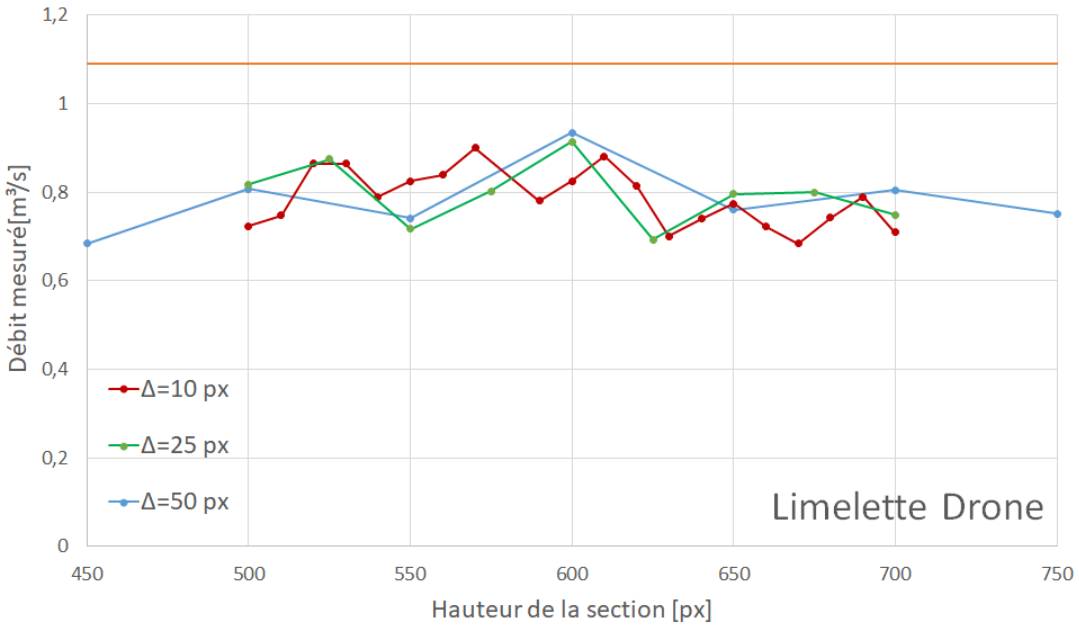


FIGURE 7.17 – Résultats de la variation de la hauteur introduite pour Limelette Drone.

7.2.4 Rapport signal/bruit - Signal to Noise Ratio - SNR

Lors de l'étape de l'introduction des paramètres utiles à l'analyse (Figure 3.42), on introduit une valeur de seuil en dessous duquel un vecteur est considéré comme faux et donc remplacé par un vecteur calculé via une moyenne locale des vecteurs environnants. Il apparaît donc que ce seuil doit être choisi avec soin : s'il est trop bas, des vecteurs incorrects seront incorporés dans la moyenne des images, alors même qu'ils pourraient la fausser de peu, voire de beaucoup si le nombre d'images pris en compte est faible. S'il est trop élevé, des vecteurs corrects seront remplacés, provoquant une perte d'informations. Surtout, si ce seuil est beaucoup trop élevé, une majorité des vecteurs sera remplacée. Étant donné qu'il reste peu de vecteurs considérés comme justes, l'application doit alors interpoler des vecteurs sur base de très peu d'informations donnant lieu à des déplacements erronés et totalement incohérents. C'est ce phénomène qui explique les résultats obtenus en premier lieu avec l'application et expliqués au point 4.3.

Influence du seuil SNR sur le champ de vecteurs

Pour évaluer l'influence du seuil SNR introduit dans l'application, nous avons dans un premier temps effectué l'analyse sur deux images consécutives issues de vidéos (Figure 7.18). Cela nous a permis de nous affranchir de la moyenne effectuée sur toutes les images et ainsi de mieux voir quels étaient les effets de la validation. De plus, la fonction `sig2noise` permet de générer un histogramme des SNR liés

à l'analyse PIV entre deux images. Ce graphe nous aidera à choisir des valeurs pertinentes de seuil. Les images utilisées dans un premier temps sont issues de la prise de mesure de Edouard De Grand Ry et Charlotte Everaert à Noirhat.

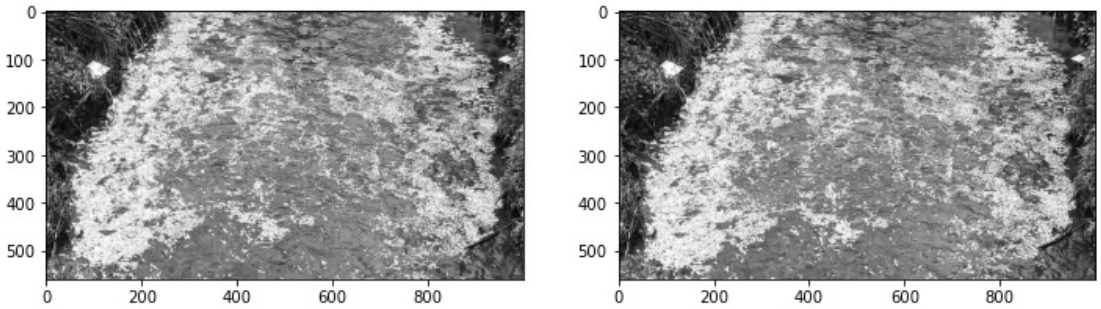


FIGURE 7.18 – Images utilisées pour évaluer l'influence du seuil SNR. Elles ont été extraites de la vidéo à 3 frames d'intervalles.

L'histogramme lié à ce couple d'images est représenté sur la figure 7.19.

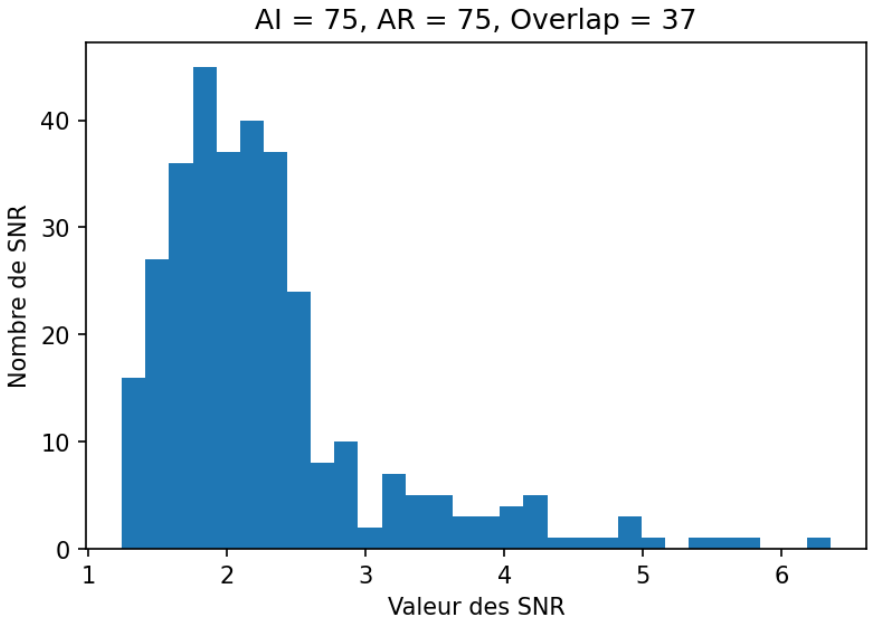


FIGURE 7.19 – Histogramme des SNR lié au couple d'images issu de la vidéo de Noirhat

Grâce à cet histogramme, il est possible d'évaluer la proportion de vecteurs qui seront remplacés en fonction de la valeur du seuil que l'on introduira. Nous avons donc fait varier le seuil de remplacement entre 1 (aucun vecteur remplacé) à 2, pour voir les effets de ces remplacements et quel est la valeur idéale.

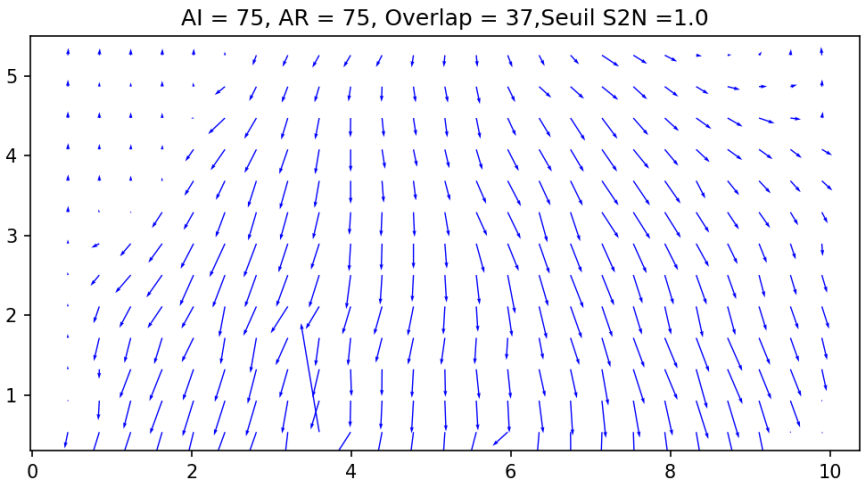


FIGURE 7.20 – Champ de vecteur lié aux images montré précédemment (Seuil fixé à 1). Le background n'a pas été illustré afin de clarifier la lecture.

Quand le seuil est fixé à 1, le champ de vecteurs est déjà assez cohérent, à l'exception de quelques vecteurs manifestement erronés (Figure 7.20) .

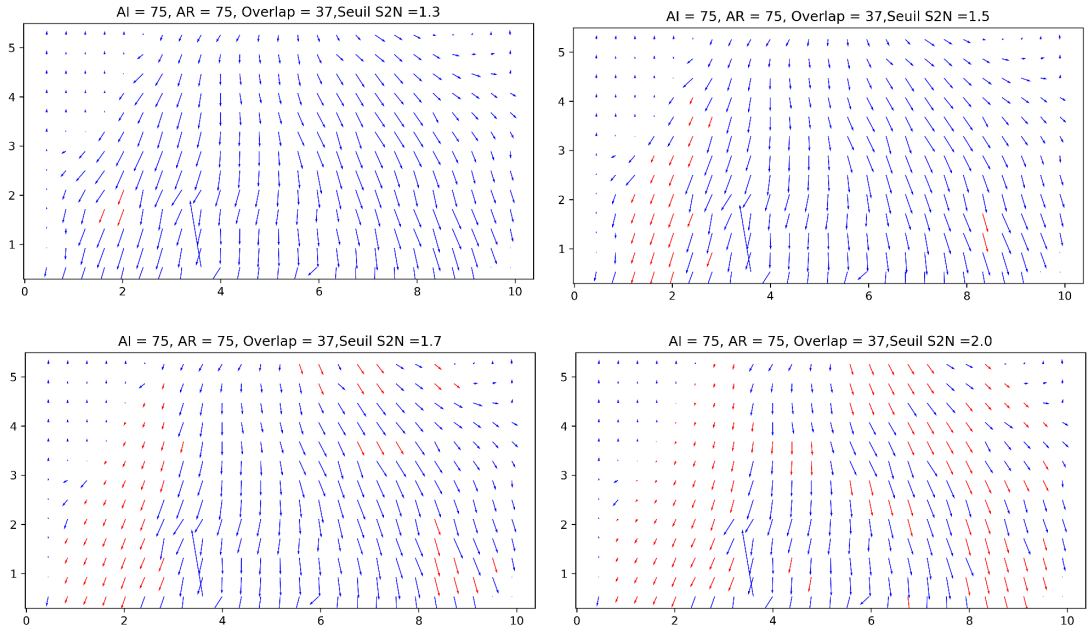


FIGURE 7.21 – Champs de vecteurs obtenus en faisant varier le seuil de remplacement de 1.3 à 2. Les vecteurs rouges sont les vecteurs qui ont été remplacés.

En faisant varier le seuil en dessous duquel les vecteurs sont remplacés, on peut observer que les remplacements s'opèrent par zones (Figure 7.21). Il y a donc des espaces dans lesquels les vecteurs sont moins fiables.

Ces zones s'étendent au fur et à mesure que le seuil est augmenté. Au coeur de ces zones, les vecteurs sont très interpolés et l'on peut voir les pertes d'informations liées à ces remplacements de masse dans les zones rouges : les vecteurs sont très alignés. Il faut donc trouver un seuil permettant de limiter l'apparition de ces zones. Après des tests effectués sur des vidéos aux caractéristiques différentes, il nous a semblé que le seuil idéal entre deux images était de 1.5. Ce seuil permet de remplacer les vecteurs efficacement dans les zones moins précises (dénuées de traceurs par exemple) tout en conservant l'allure générale de l'écoulement. Des exemples sont illustrés à la figure 7.22.

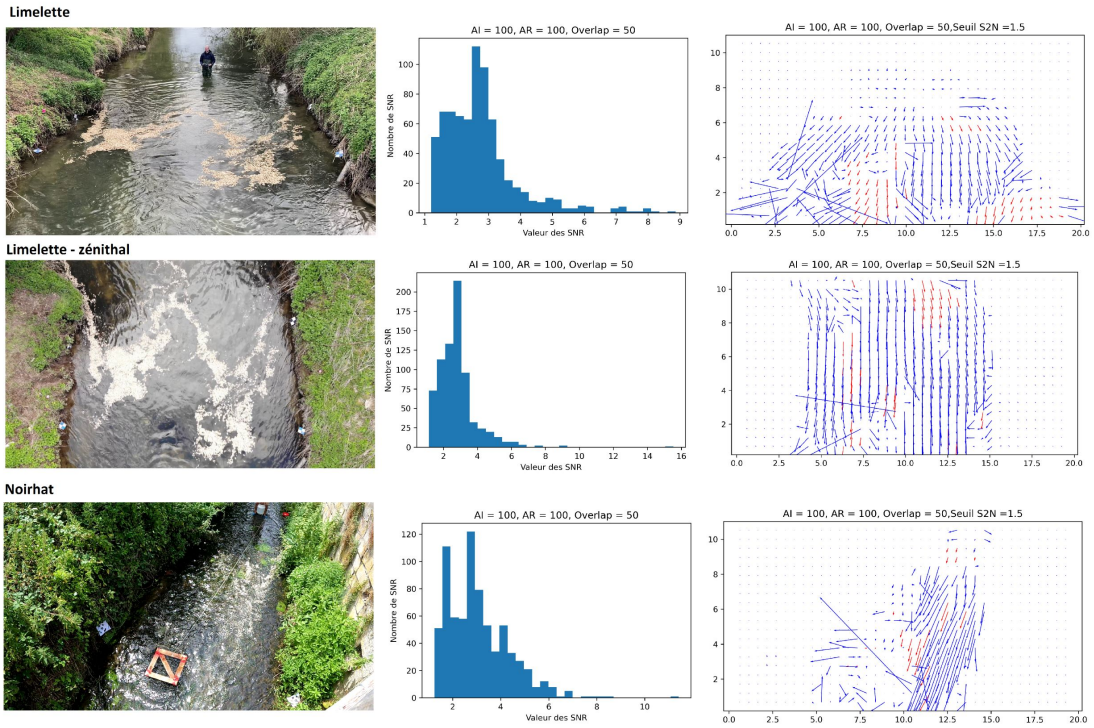


FIGURE 7.22 – Images, histogrammes et champs de vecteurs obtenus en fixant le seuil SNR à 1.5

Remarque 35. *On peut apercevoir que des vecteurs a priori incohérents sont épargnés par le remplacement. Souvent, ils se situent hors de la zone réellement étudiée (trop en amont ou aval) et n'affecteront donc pas la mesure du débit final. De plus, ces champs concernent l'écoulement entre 2 images seulement, sur des temps très courts donc. Il est possible que sur ce laps de temps, un remous ait eu lieu à contre-courant et que la zone associée conserve un SNR assez haut pour ne pas être remplacé. Ces singularités sont lissées grâce à la moyenne temporelle et n'affectent donc également que très peu le débit final.*

Toutefois, il faut garder à l'esprit que ces conclusions ont été effectuées sur base de l'analyse entre deux images, alors que l'application effectue cette opération plusieurs fois voire dizaines de fois selon le temps et le step choisis. De plus, beaucoup de traceurs sont présents sur les images de la figure 7.22, ce qui élève la valeur de beaucoup de SNR. Il s'avère que le seuil nécessaire à l'obtention d'un champ de vecteur correct dans le cas d'une vidéo entière est beaucoup plus bas sous peine d'en détériorer fortement l'allure (Figure 7.23).

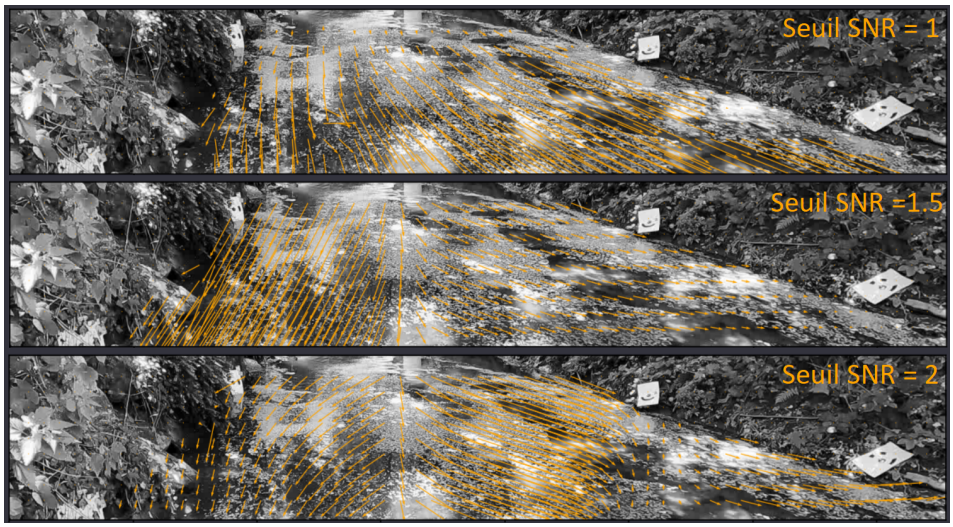


FIGURE 7.23 – Champs de vecteurs obtenus sur une vidéo de Suzeryl avec le seuil SNR varié de 1 à 2.

Dès lors, une bonne pratique serait de performer une analyse PIV en fixant le seuil à 1 pour évaluer la pertinence de remplacer les vecteurs dans certaines zones et si besoin d'augmenter peu à peu ce seuil pour combler les carences d'un champ de vecteurs donnés (exemple à la figure 7.24). Selon notre expérience, ce seuil ne dépasse pas 1.1 pour les vidéos à notre disposition .

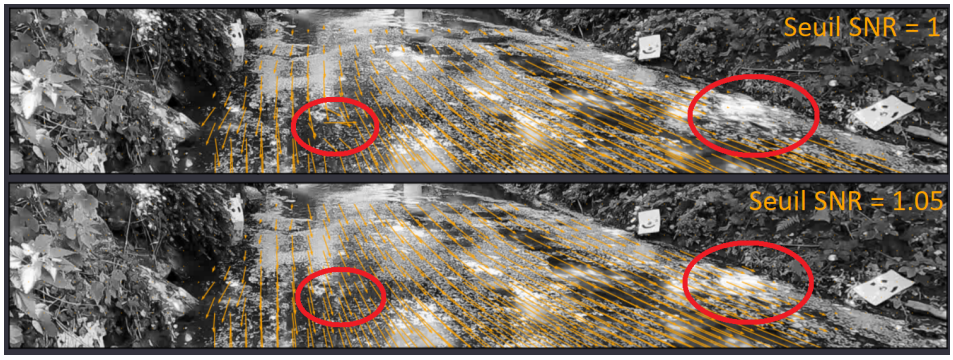


FIGURE 7.24 – Carences comblées via l'application d'un seuil SNR adéquat.

Chapitre 8

Mise en Open Source

8.1 L'Open Source : Pourquoi ?

Lorsque nous avons dressé la liste des objectifs de notre TFE, nous avions l'intention de mettre le code informatique de l'application en Open Source. Mais pourquoi mettre ce code en Open Source et pourquoi cela semble-t-il utile pour la suite et le futur de ce travail de fin d'études ?

À cela nous pouvons énoncer plusieurs réponses.
La première raison de mettre ce code en Open Source est liée à la transmission des informations. En effet, l'application que nous utilisons et tentons d'améliorer est le fruit de plusieurs fonctions de longueur variables mais pouvant atteindre pour certaines quelques centaines voir quelques milliers de lignes. Nous nous trouvons dès lors devant une certaine quantité d'informations à transférer d'un disque dur à un autre entre différents étudiants qui parfois ne se trouvent plus sur le site universitaire. La transmission se fait donc via des logiciels tels que ***Transvol*** ou ***WeTransfer*** et est donc parfois source de pertes et/ou d'oublis.
C'est d'ailleurs ce qui s'est passé en partie pour nous lors du partage des fichiers (voir section 4.1).

Une seconde raison que l'on peut énoncer est le fait que ce projet a été et pourra encore être étudié par binôme. Il s'avère qu'il est à la fois compliqué de travailler à deux sur la même chose à distance et également une perte de temps si les deux parties doivent sans cesse se réunir pour travailler ensemble ou transférer les changements entre eux. Ce qui est vrai pour un binôme prend encore plus d'ampleur quand il s'agit du travail de toute équipe de plus de deux personnes.

La dernière raison qui nous a poussé à nous tourner vers l'Open Source est la

constante amélioration du langage ***Python***. En effet, certaines nouvelles versions des packages ne correspondent plus entre elles et rendent la compilation du code impossible (Section 4.2). Il est donc nécessaire de fournir un environnement fixe sur lequel l'application fonctionne correctement.

Tous ces problèmes rencontrés peuvent être résolus par l'utilisation de l'Open Source !

8.2 Qu'est ce que l'Open Source ?

8.2.1 Un brin d'histoire...

Au début des années 80, un mouvement prend forme sous l'impulsion du programmeur américain : Richard Matthew Stallman. Ce mouvement grandissant est celui que l'on connaît sous le nom de mouvement du logiciel libre. À ce moment là, ce sont les logiciels prioritaires qui sont majoritaires c'est-à-dire des logiciels où il est impossible à quiconque de modifier ou de copier, du moins de manière légale, le code source. Les utilisateurs paient pour pouvoir utiliser le logiciel conformément à l'usage prévu par le propriétaire. Stallman soulève alors le besoin d'octroyer des libertés aux utilisateurs. Ces libertés énoncées par le programmeur étaient la liberté : d'accéder au code source, de le modifier et de le redistribuer, pour le rendre disponible à d'autres personnes, qui elles-même peuvent à leur tour procéder de la sorte.

Dans ces conditions, le mouvement commence à grandir et les logiciels libres se mettent à apparaître de plus en plus. Cependant, par abus de langage et de traduction, les logiciels libres sont apparentés à des logiciels gratuits. En effet, en anglais logiciel libre se traduit par "*free software*", or le mot "*free*" possède plusieurs significations dans la langue de Shakespeare. D'un côté celui-ci signifiera "*libre*" mais de l'autre il se traduira par "*gratuit*". Cette double traduction apporte son lot de confusions car il est vrai que les logiciels sont bien libres mais leur gratuité n'est en rien une condition suffisante pour être un logiciel libre.
Dès lors, et pour pallier ce soucis d'interprétation, la scientifique Christine Peterson proposa pour la première fois le mot ***Open Source*** à un groupe de travail dont l'objectif était de démontrer au monde entier qu'un logiciel était plus puissant s'il était collaboratif et partagé.

Ce n'est finalement qu'en 1997, qu'un certain Eric Raymond, membre du groupe de travail précédemment cité, publia un livre ("*La Cathédrale et le Bazar*") et répandit le mot ***Open Source*** à travers le monde.
La création de *Open Source Initiative* au début de l'année 1998, officialisa définitivement l'expression.

Bien que vus d'un mauvais oeil au début de leur ascension, les logiciels libres ont su effacer leurs débuts compliqués et sont maintenant utilisés et répandus dans le monde entier. Ils constituent une énorme part des logiciels actuels. [17, 30]

8.2.2 Caractéristiques de l'Open Source

Le terme ***Open Source*** peut être traduit littéralement par "*code source ouvert*" ou "*source en accès libre*". Ces traductions montrent déjà la volonté de posséder un libre accès au code source d'un logiciel. Par extrapolation, on peut définir l'***Open Source*** comme étant un logiciel dont le code source est accessible, modifiable et intégrable par d'autres développeurs que le propriétaire.
On peut donc établir les trois caractéristiques phares qui définissent l'***Open Source*** :

- Libre accès au code source du logiciel.
- Libre modification du code source.
- Libre redistribution du code source.

Néanmoins, afin de donner une définition plus stricte, *l'Open Source Initiative* a listé les dix caractéristiques qui représentent au mieux l'***Open Source***. Ces caractéristiques sont les suivantes[32, 69] :

- **Redistribution libre**
  La licence n'empêche aucune partie de vendre ou de céder le logiciel [...].
- **Code source**
  Le programme doit inclure le code source et doit permettre la distribution tant du code source que de sa forme compilée. [...] Le code source délibérément obfusqué [1] n'est pas autorisé [...].
- **Travaux dérivés**
  La licence doit permettre les modifications et les travaux dérivés, ainsi que leur distribution dans les mêmes conditions que la licence du logiciel original.
- **Intégrité du code source de l'auteur**
  La licence doit explicitement autoriser la distribution de logiciels construits à partir de code source modifié. La licence peut exiger que les oeuvres dérivées portent un nom ou un numéro de version différent de celui du logiciel d'origine.
- **Pas de discrimination contre les personnes ou les groupes**
  La licence ne doit pas discriminer une personne ou un groupe de personnes.

1. (Informatique) Rendre (en particulier un programme) illisible pour éviter qu'il soit exploité de façon non autorisée, par exemple en transformant les noms des variables pour qu'ils deviennent incompréhensibles. [70]

- **Pas de discrimination dans les domaines de compétences**
  La licence ne doit empêcher personne d'utiliser le programme dans un domaine d'activité spécifique. Par exemple, elle ne doit pas empêcher le programme d'être utilisé dans une entreprise ou d'être utilisé pour la recherche génétique.
- **Distribution de licence**
  Les droits attachés au programme doivent s'appliquer à tous ceux à qui le programme est redistribué, sans nécessité de conclure une licence additionnelle avec ces parties.
- **La licence ne doit pas être spécifique à un produit**
  Les droits attachés au programme ne doivent pas dépendre de son appartenance à une distribution de logiciel particulière. Si le programme est extrait de cette distribution et utilisé ou distribué selon les termes de la licence du programme, toutes les parties auxquelles le programme est redistribué devraient bénéficier des mêmes droits que ceux concédés avec la distribution de logiciel d'origine.
- **La licence ne doit pas restreindre d'autres logiciels**
  Les droits attachés au programme ne doivent pas dépendre de son appartenance à une distribution de logiciel particulière. Si le programme est extrait de cette distribution et utilisé ou distribué selon les termes de la licence du programme, toutes les parties auxquelles le programme est redistribué devraient bénéficier des mêmes droits que ceux concédés avec la distribution de logiciel d'origine.
- **La licence doit être neutre d'un point de vue technologique**
  Aucune disposition de la licence ne peut être basée sur une technologie ou un style d'interface individuel.

On peut remarquer que le mot licence a été introduit lors du listing des caractéristiques. Ce terme permet de différencier les logiciels ***Open Source*** et ceux simplement libres. En effet, même si les critères principaux sont identiques, la licence qui restreint la modification ou la redistribution dans le premier cas est totalement absente dans le second.
Cette licence va également permettre de différencier certaines classes de logiciels ***Open Source*** et c'est ce qui sera présenté dans la section suivante 8.3. [9, 17, 30, 59, 63].

8.3 Types de licences

La licence est en quelque sorte le contrat qui accompagne chaque logiciel et qui en restreint ou non la modification et/ou la redistribution. De cette façon, on verra certaines licences être totalement permissives, c'est-à-dire qu'elles laissent aux utilisateurs ultérieurs le droit de faire ce qu'ils souhaitent, y compris faire du logiciel un logiciel propriétaire après modifications, et d'autres imposer une redistribution sous la licence du logiciel initial ou autre. Cette obligation est ce que l'on appelle le ***Copyleft***.

Parmi les licences copyleft, on peut distinguer deux grandes familles :

- **Les copylefts faibles**
  Dans ce cas-ci, les utilisateurs pourront disposer du code, le modifier et finalement le redistribuer sous la licence initiale ou une autre. Il faut néanmoins faire attention à la compatibilité entre licences. En effet, il est impossible de publier une modification d'un logiciel, initialement sous licence copyleft, sous une licence entièrement permissive. Un tableau reprenant les compatibilités entre certaines licences est repris en annexe .5.
- **Les copylefts forts**
  Les fonctionnalités sont les mêmes que pour les copylefts faible hormis le fait que la redistribution doit s'exécuter uniquement dans la licence initiale. Il est alors impossible de rendre le logiciel propriétaire ce qui est possible pour les licences permissives ainsi que pour certaines copylefts faibles.[59]

On peut donc déceler trois grandes familles de licence. La première regroupe les licences permissives, parmi lesquelles on peut énoncer la licence ***MIT***. Les licences copyleft, grandement représentées par la famille des licences ***GPL***, représente la deuxième grande famille. Et finalement, entre les deux, on trouvera la famille des licences copyleft (faibles) mais qui possèdent certaines propriétés des licences permissives. C'est le cas de la licence ***LGPL***.

8.4 Choix de la licence appliquée

Le choix de la licence n'est pas une tâche aisée. En effet, le choix effectué aujourd'hui peut impacter la redistribution du code source dans plusieurs années. C'est pourquoi, il convient de faire les bons choix et de se renseigner sur les portes que l'on pourrait fermer.

Connaissant l'état actuel des choses et les potentielles futures utilisations de l'application au sein de milieux universitaires, il semble que le choix de la licence n'ait pas de conséquences directes. Cependant, si l'on imagine plusieurs scénarios

pour les prochaines années, il s'avère que les licences copylefts (forts) pourraient être moins applicables et ce pour deux raisons.

La première étant que, de par la définition même de leur licence, la redistribution doit obligatoirement se faire sous la même licence. Un retour en arrière en termes de permissivité est donc à exclure car on se trouverait devant un soucis de compatibilité comme le montre l'annexe .5.

La seconde est que, si l'université venait à vouloir établir un partenariat financier avec une entreprise quelconque, cette dernière pourrait être rebutée par la licence. De fait, l'utilité pour une entreprise étant d'obtenir des bénéfices, elle désirera faire de la base du logiciel un logiciel propriétaire, ce qui est totalement prohibé avec ce genre de licence.

Bien que les licences permissives telle que la ***MIT*** s'opposent aux éléments énoncés ci-dessus, elle n'est pas pour autant la solution la plus adaptée. En effet, sous de telles licences, quiconque pourrait se procurer le code source et en faire ce qu'il désire. L'UCLouvain perdrait alors le bénéfice de l'élaboration de l'application en son sein.

Afin de pouvoir envisager plus de compatibilité ou de potentiels partenariats tout en gardant un minimum la "main mise" sur le code source, la licence ***LGPL*** semble un bon compromis.
Cette licence, contrairement à ces dérivées plus strictes ***(A)GPL***, octroie le droit à un utilisateur du code source de rendre le logiciel propriétaire, dans le sens où celui-ci pourrait utiliser le code pour ériger un site payant par exemple. Cependant, cet utilisateur se devra de toujours laisser libre accès au code source ainsi qu'aux modifications éventuelles qui seront elles-mêmes publiées sous cette même licence.

Même si cette dernière licence paraît intéressante et appropriée à l'application à laquelle nous avons contribué. Il convient avant tout de même de vérifier que celle-ci est bien compatible avec les licences des packages *Python* utilisés. La liste des licences se trouve en annexe .6.

Le choix de la licence ***LGPL*** semblait en accord avec la plupart de packages utilisés. Seulement, le package ***openpiv*** se trouve sous licence ***GPL3***, qui est fortement copyleft et impose que la distribution d'un code utilisant ce package se fasse sous la même licence.
Nous avons donc été contraints de choisir la licence ***GPL3*** pour l'application et ce malgré les raisons énoncées plus haut.

Enfin et afin d'officialiser la mise en application de la licence, nous avons dû demander l'accord des contributeurs précédents. Une fois obtenu, nous avons pu

ajouter dans le ***Repository*** de notre logiciel le texte de la licence (COPYING.txt), un fichier reprenant les différents auteurs (AUTHORS.txt), un fichier reprenant une brève description de tout (LICENSE.txt) ainsi que deux phrases de rappel sur les auteurs et la licence au début de chaque fichier ***Python***. [62]

Chapitre 9

Pistes d'améliorations

Au cours de la prise en main de l'application et des tests effectués, nous avons constaté plusieurs problèmes ou chemins d'analyse qui demandent encore des améliorations dans un futur plus ou moins proche.
En effet, cette application, bien que sur pied depuis maintenant trois ans, n'est pas parfaite et nécessite encore des modifications et des attentions particulières concernant certains points de son utilisation.

Nous avons donc rassemblé ici les pistes d'améliorations qui nous semblent les plus intéressantes à approfondir que ce soit dans l'utilisation à proprement parler de l'application ou de la prise de mesures :

- Parmi les améliorations à apporter, la plus nécessaire est sans doute celle liée aux problèmes décrits à la section 4. En effet, cette section reprend de nombreux problèmes que nous avons découverts lors de nos multiples utilisations. Bien que la plupart de ces problèmes ne constituent pas un frein à l'application, certains rendent l'utilisation de cette dernière moins pratique et moins ergonomique. L'objectif d'une telle amélioration serait donc de rendre l'application plus intuitive et plus robuste dans le futur.
  Afin que ceci soit fait de la meilleure manière, nous pensons qu'il serait préférable que des étudiants plus spécialisés dans l'informatique ou même des professionnels s'attellent à cette tâche.
- Dans la mesure où de prochaines études pourraient approfondir l'analyse de la sensibilité de l'application, il nous apparaît clairement que de tels travaux devraient commencer par l'automatisation de l'application. Nous entendons par là un dispositif capable de réaliser plusieurs mesures consécutives avec des paramètres prédéfinis, permettant de fixer et de faire varier certaines valeurs selon une trame spécifiée à l'avance, et de stocker les résultats

obtenus (débits finaux mais également les développements intermédiaires), le tout sans intervention humaine. Une telle fonctionnalité décuplerait le potentiel d'analyse de la sensibilité et contribuerait grandement à comprendre l'application.

- Même si nous avons procédé à plusieurs tests de sensibilité sur quelques paramètres, nous n'avons pu tester tous les filtres sur toutes les vidéos que nous possédons. De plus, certains filtres semblent ne pas fonctionner. Il serait donc intéressant d'établir de nouvelles séries de tests afin de comprendre mieux leur utilisation pour les uns et de corriger les problèmes pour les autres.
- Un des objectifs de notre travail de fin d'études était de mettre en place un kit de mesure dont le but principal était de performer une orthorectification sur tous les cours d'eau, même ceux dont les berges sont inaccessibles et où il est dangereux de s'aventurer dans l'eau. Une première ébauche de ce kit a été construite et testée via plusieurs séries d'expériences. Bien qu'il ait montré des résultats prometteurs, il est évident que certaines améliorations peuvent lui être apportées afin de le rendre plus performant et ergonomique (différentes tailles, pliable,...) (Section 6.1 et 7.1.3).
- Lors de notre première rencontre avec Firas Ghali, qui était en stage sur le site de Louvain-la-Neuve, il a tout de suite demandé pourquoi nous n'utilisions pas la médiane au lieu de la moyenne lors de la dernière étape du calcul du débit. Pour rappel l'analyse PIV est effectuée autant de fois qu'il y a de paires d'images, générant autant de vecteurs sur un même noeud. L'application mesure donc actuellement la moyenne entre ces vecteurs. Il serait effectivement intéressant de comparer les deux approches et donc d'implémenter cette possibilité de calcul.
  Il semblerait d'ailleurs que Firas ait déjà implémenté une fonction qui permette de mesurer le débit à l'aide de la médiane et qui montre des premiers résultats encourageants.
  D'autre part, dans leur mémoire, Bia T. et Somers H.[23] évoquent la moyenne des matrices de corrélation à la place de celle de vecteurs vitesse et cette alternative semblait montrer de bons résultats.
  Il serait judicieux d'implémenter ces différentes approches et de les tester via l'application.
- Cette année, nous avons eu le privilège de pouvoir échanger avec des chercheurs à l'étranger et obtenir des vidéos de leurs rivières locales.
  Effectuer des tests sur de nouvelles rivières représente une occasion unique de valider ou non certaines fonctionnalités de l'application avec des cours d'eau aux caractéristiques différentes de ceux de nos contrées. De plus, ces vidéos seraient alors disponibles pour des analyses ultérieures.
  Garder de tels contacts ou en établir de nouveaux serait une opportunité alléchante pour le futur de l'application.

Conclusion

Il est maintenant admis que le changement climatique favorise les évènements naturels extrêmes, impactant directement les débits des cours d'eau du monde entier en augmentant la fréquence et l'intensité des périodes de crues et d'étiages. Alors que c'est dans ces moments cruciaux que la gestion des débits est la plus importante, les appareils de mesure utilisés actuellement ne sont pas adaptés : non-calibrés aux valeurs extrêmes ou dangereux pour les utilisateurs et le matériel.

C'est en réponse à ces problématiques que Simon Renard, et plus tard Charlotte Everaert et Edouard de Grand Ry ont développé une application capable d'estimer un débit sur base de vidéos. C'est leur travail et leurs analyses que nous avons repris tout au long de cette année.

Tout d'abord, nous avons récupéré le code source de l'application, et tenté de d'effectuer nos premières mesures. Nous avons rapidement fait face à des difficultés pour obtenir, lancer ou faire fonctionner le programme. Notre première tâche fut donc non seulement de résoudre ces problèmes mais aussi de faire en sorte qu'ils ne surviennent pas pour les futurs utilisateurs et développeurs. Nous avons notamment rétabli l'accès du code aux différentes librairies et l'avons introduit sur la forge logicielle de l'UCLouvain.

Toujours dans le but de faciliter cette prise en main, nous avons créé un guide utilisateur complet décrivant chaque écran ainsi que toutes les fonctionnalités fournies. Pour les prochains développeurs, tous les problèmes que nous avons rencontrés ont été consignés et classés, également sur la forge. Le processus de mise en Open Source, évoqué comme une piste d'amélioration par nos prédécesseurs, a été effectué par la mise en place d'une licence. Une fois l'application prête à être distribuée, il ne restera qu'à la mettre sur une plateforme de partage de code adaptée.

En parallèle des actions menées sur le code, nous avons préparé des mesures sur des cours d'eau proches de l'université, ce qui nous a amenés à concevoir un protocole de mesure que nous avons eu l'occasion d'envoyer à des chercheurs étrangers. Au vu de la qualité des données qui nous ont été renvoyées par Rotchild Louis, le protocole a été bien compris et peut servir de base à de prochaines liaisons.

De notre côté, nos excursions sur le terrain nous ont permis, en plus de récolter des vidéos aux caractéristiques différentes, de tester un prototype de kit facilitant les mesures. Il s'avère que moyennant une utilisation consciencieuse et précise, il peut tout à fait supplanter l'étape de placement des repères utiles à l'orthorectification.

Enfin, l'analyse de la sensibilité de l'application nous a permis de mieux la comprendre, et d'évaluer des paramètres optimaux selon les caractéristiques des vidéos étudiées. Cela nous a également permis d'ajouter des conseils au guide utilisateur.

Ce travail de fin d'études nous a donc permis, plus que d'améliorer réellement la précision de l'application, de développer sa transmission, tant vers de futurs utilisateurs que vers les développeurs qui viendront après nous. Des pistes d'amélioration ont d'ailleurs été listées dans ce mémoire à l'intention de ces derniers.

Et qui sait, peut-être un jour pourrons nous mesurer les débits des cours d'eau de nos régions en utilisant l'application à laquelle nous avons consacré ce travail de fin d'études ?

Bibliographie

[1] Anaconda.org. Managing environments. https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html. Consulté le 23 juin 2022.

[2] Anonyme. 4.limnimétrie. http://medhycos.mpl.ird.fr/en/data/hyd/Drobot/4C.htm#:~:text=Les%20principaux%20types%20de%20limnigraphes,%C3%A9lectroniques%20%C3%A0%20capteur%20pi%C3%A9zo%2D%C3%A9lectrique. Consulté le 24 juillet 2022.

[3] Anonyme. Chapitre 7 la mesure hydrologique. http://voies-hydrauliques.wallonie.be/opencms/opencms/fr/hydro/Gestion/Mesures/Hydrometrie/index.html#Jaugeage. Consulté le 24 juillet 2022.

[4] ArcGis. Notions de base de l'orthorectification d'un jeu de données raster. https://desktop.arcgis.com/fr/arcmap/10.6/manage-data/raster-and-images/fundamentals-of-orthorectifying-a-raster-dataset.htm. Consulté le 16 août 2022.

[5] Wieneke Bernhard. *PIV Uncertainty Quantification and Beyond*. PhD thesis, Delft University of Technology, 2017.

[6] R. Bhagat. How to measure flow of water in open channels ? | water engineering. https://www.engineeringenotes.com/water-engineering-2/open-channels/how-to-measure-flow-of-water-in-open-channels-water-engineering/44684. Consulté le 25 juillet 2022.

[7] Ambroise Carton. Inondations de juillet 2021 en belgique : le résumé des événements heure par heure (ligne du temps). *RTBF*, 2022. Consulté le 16 août 2022.

[8] climat et territoires de demain Cerema. Mesures de débit. http://wikhydro.developpement-durable.gouv.fr/index.php/Mesures_de_d%C3%A9bit#:~:text=Le%20courantom%C3%A8tre%20%C3%A9lectromagn%C3%A9tique%20permet%20%C3%A9galement,%C3%A0%20la%20vitesse%20d'%C3%A9coulement. Consulté le 25 juillet 2022.

[9] Cyrille Chausson. Open source. https://www.lemagit.fr/definition/Open-Source. Consulté le 22 juin 2022.

[10] Sahil Dadia. Pyimagesearch/imutils. https://github.com/PyImageSearch/imutils/blob/master/LICENSE.txt. Consulté le 11 août 2022.

[11] Charlotte de Gret Ry, Edouard et Everaert. Mesures de débit en rivière à partir de techniques d'imagerie : développement d'une application. Master's thesis, UCLouvain, Louvain-la-Neuve, Belgium, 2021.

[12] DTG. Jaugeage par dilution de traceur à injection globale - exemple du nacl et de la rhodamine. http://www.meteo.fr/cic/meetings/2012/journees_hydrometrie/pres/Jaugeage_par_dilution_de_traceur.pdf. Consulté le 24 juillet 2022.

[13] Royaume du Maroc. Note technique no6/197 - moulinets hydrométriques. Technical report, Ministère des travaux publics et des communications - Direction de l'hydraulique - Division des ressources en eau, Décembre 1974.

[14] EauFrance. Les stations hydrométriques. https://www.artois-picardie.eaufrance.fr/qualite-et-quantite-des-eaux/situation-hydrologique-hydrogeologique-et-pluviometrique/article/les-stations-hydrometriques. Consulté le 24 juillet 2022.

[15] L'hydrométrie en Wallonie. Réseaux de mesure. https://hydrometrie.wallonie.be/home/en-savoir-plus/reseaux-de-mesure.html. Consulté le 24 juillet 2022.

[16] L'hydrométrie en Wallonie. Sècheresse extrême et précision des instruments de mesure. https://hydrometrie.wallonie.be/news/secheresse-extreme-et-precision-des-instruments-de-mesure. Consulté le 12 août 2022.

[17] Astuces et Aide Informatique. Qu'est-ce que l'open source ? https://www.astuces-aide-informatique.info/7502/open-source-definition#definition_de_lopen_source. Consulté le 22 juin 2022.

[18] Fujita et al. Large-scale particle image velocimetry for flow analysis in hydraulic engineering applications. *Journal of Hydraulic Research*, 1997.

[19] Raffel M. et al. *Particle Image Velocimetry - A Practical Guide*, page 669. Springer International Publishing AG, 2018.

[20] Fabrice Mourau et Bruno Arfib. Jaugeage du débit d'un cours d'eau par dilution de chlorure de sodium. https://www.pedagogie.ac-nice.fr/svt/?p=1744. Consulté le 24 juillet 2022.

[21] Willemart Emmanuel et Lebon Loïc. Mesures de débits en rivière à partir d'un drone : validation de méthodes basées sur la LSPIV à partir de mesures sur le terrain. Master's thesis, UCLouvain, Louvain-la-Neuve, Belgium, 2021.

[22] Breels Nicolas et Marneffe Stephan. Mesure de débit en rivière à partir de techniques d'imagerie combinant LSPIV et photogrammétrie. Master's thesis, UCLouvain, Louvain-la-Neuve, Belgium, 2021.

[23] Bia Tristan et Somers Harold. Mesure du champ de vitesses superficielles en rivière à partir d'un drone. Master's thesis, UCLouvain, Louvain-la-Neuve, Belgium, 2018.

[24] Shauna Ní Fhlaithearta and Andrea Spitzner. Improving the accuracy of adcp discharge measurements. https://www.hydro-international.com/content/article/improving-the-accuracy-of-adcp-discharge-measurements#:~:text=One%20standard%20method%20to%20measure,both%20as%20transmitters%20and%20receivers. Consulté le 25 juillet 2022.

[25] Sanja Fidler. Matching planar objects in new viewpoints... and much more – via homography. http://www.cs.toronto.edu/~fidler/slides/2017/CSC420/lecture9.pdf. Consulté le 16 août 2022.

[26] OTT France. Ott mf pro. https://www.ott.com/fr-fr/produits/le-debit-deau-70/ott-mf-pro-177/. Consulté le 25 juillet 2022.

[27] Cumps Freddy. Hydrométrie - mesure de débits solides, mesures de débits liquides. FHR, 2008.

[28] R. Gurka T. Hadad. Effects of particle size, concentration and surface coating on turbulent flow properties obtained using piv/ptv. *Elsevier*, novembre 2012.

[29] Olivier Hagolle. L'ortho-rectification, comment ça marche ? https://labo.obs-mip.fr/multitemp/lortho-rectification-comment-ca-marche/. Consulté le 16 août 2022.

[30] Red Hat. L'open source, qu'est-ce que c'est ? https://www.redhat.com/fr/topics/open-source/what-is-open-source#histoire-de-lopen%C2%A0source. Consulté le 22 juin 2022.

[31] Red Hat. Un environnement de développement intégré, qu'est-ce que c'est ? https://www.redhat.com/fr/topics/middleware/what-is-ide#:~:text=IDE. Consulté le 12 août 2022.

[32] Open Source Initiative. The open source definition. https://opensource.org/osd. Consulté le 9 août 2022.

[33] Benjamin Jean. *Option libre : Du bon usage des licences libres*, page 316. 2012.

[34] avec Carl Defoy Jean-François Noulet. Sécheresse en région wallonne : la cellule de crise a fait le point "sur les bases d'une année historique". *RTBF*, 2022.

[35] Mr. Kerboub. Chapitre 4 : Hydrométrie mesure des paramètres de l'écoulement. https://iast.univ-setif.dz/documents/Cours/CHAP4Hydrom%C3%A9trieL2GAT21.pdf. Consulté le 24 juillet 2022.

[36] Kivy. Kivy-home. https://kivy.org/#home. Consulté le 11 août 2022.

[37] Larousse. Définition stabiliser. https://www.larousse.fr/dictionnaires/francais/stabiliser/74405. Consulté le 16 août 2022.

[38] Satya Mallick. Object tracking using opencv (c++/python). https://learnopencv.com/object-tracking-using-opencv-cpp-python/. Consulté le 16 août 2022.

[39] Matplotlib. License. https://matplotlib.org/stable/users/project/license.html. Consulté le 11 août 2022.

[40] Numpy. Numpy-home. https://numpy.org/. Consulté le 11 août 2022.

[41] U.S. Department of the Interior U.S. Geological Survey. *Measuring Discharge with Acoustic Doppler Current Profilers from a Moving Boat*, page 117. Decembre 2013.

[42] OpenCV. License. https://opencv.org/license/. Consulté le 11 août 2022.

[43] OpenPiv. Github of openpiv. https://github.com/OpenPIV/openpiv-python/blob/master/openpiv/pyprocess.py. Consulté le 13 juillet 2022.

[44] OpenPIV. Openpiv tutorial. https://openpiv.readthedocs.io/en/latest/src/tutorial1.html. Consulté le 12 juin 2022.

[45] Phtyon. tifffile 2019.7.26.2. https://pypi.org/project/tifffile/2019.7.26.2/. Consulté le 11 août 2022.

[46] Pillow. License. https://pillow.readthedocs.io/en/stable/about.html#license. Consulté le 11 août 2022.

[47] Python. functools 0.5. https://pypi.org/project/functools/. Consulté le 11 août 2022.

[48] Python. glob.py. https://pypi.org/project/glob2/0.6/. Consulté le 11 août 2022.

[49] Python. Openpiv 0.23.4. https://pypi.org/project/OpenPIV/0.23.4/. Consulté le 11 août 2022.

[50] Python. pathlib 1.0.1. https://pypi.org/project/pathlib/. Consulté le 11 août 2022.

[51] Python. python-math 0.0.1. https://pypi.org/project/python-math/. Consulté le 11 août 2022.

[52] Python. python/cpython. https://github.com/python/cpython/blob/3.10/Lib/re.py. Consulté le 11 août 2022.

[53] Python. threading2 0.3.1. https://pypi.org/project/threading2/. Consulté le 11 août 2022.

[54] Python.org. Environnements virtuels et paquets. https://docs.python.org/fr/3/tutorial/venv.html. Consulté le 30 mai 2022.

[55] Meynart R. Convective flow field measurement by speckle velocimetry. février 1982.

[56] Des rivières et des hommes. Les instruments de mesure - comment mesurer les flux d'eau et de sédiments. https://vod.canal-u.tv/vod/media/canalu/documents/unittv/les.instruments.de.mesure_18738/sem1_seq4_instruments_dia_to_pdf.pdf?fbclid=IwAR2tXU-ROFiabGLFiw8ZWzCtXD4OaRapLrT6HOtzE0u1aoZaAkCtGs7SCcU. Consulté le 24 juillet 2022.

[57] Adrian R.J. Twenty years of particle image velocimetry. août 2005.

[58] Adrian Ronald R.J. Scattering particle characteristics and their effect on pulsed laser measurements of fluid flow : Speckle velocimetry vs particle image velocimetry. juin 1984.

[59] Maëlys De Santis. Qu'est-ce qu'une licence open source ? définition et types de licences. https://www.appvizer.fr/magazine/services-informatiques/gestion-licences/licence-open-source. Consulté le 22 juin 2022.

[60] Scipy. Scipy 1.9.0 released ! https://scipy.org/. Consulté le 11 août 2022.

[61] Renard Simon. Development of an application for river discharge measurement. Master's thesis, UCLouvain, Louvain-la-Neuve, Belgium, 2020.

[62] ADAM Sébastien. I want to distribute my software developments. how to define a licensing strategy ? Louvain-la-Neuve, Belgique, 2018.

[63] Futura Tech. Open source : qu'est-ce que c'est ? https://www.futura-sciences.com/tech/definitions/informatique-open-source-18154/. Consulté le 22 juin 2022.

[64] Wikipédia. 7.3.1 la mesure des hauteurs d'eau. https://echo2.epfl.ch/e-drologie/chapitres/chapitre7/chapitre7.html#:~:text=Le%20limnigraphe%20%C3%A0%20pression%20ou,immerg%C3%A9%20reli%C3%A9%20%C3%A0%20la%20bonbonne. Consulté le 24 juillet 2022.

[65] Wikipédia. Acoustic doppler current profiler. https://en.wikipedia.org/wiki/Acoustic_Doppler_current_profiler. Consulté le 25 juillet 2022.

[66] Wikipédia. Courantomètre. https://fr.wikipedia.org/wiki/Courantom%C3%A8tre. Consulté le 25 juillet 2022.

[67] Wikipédia. Hydrométrie. https://fr.wikipedia.org/wiki/Hydrométrie. Consulté le 24 juillet 2022.

[68] Wikipédia. Limnimètre. https://fr.wikipedia.org/wiki/Limnim%C3%A8tre. Consulté le 24 juillet 2022.

[69] Wikipédia. Open source initiative. https://fr.wikipedia.org/wiki/Open_Source_Initiative. Consulté le 9 août 2022.

[70] Wiktionnaire. Définitions de obfusquer. https://fr.wiktionary.org/wiki/obfusquer#:~:text=Verbe,-obfusquer. Consulté le 11 août 2022.

[71] John J. Charonko2 Zhenyu Xue1 and Pavlos P. Vlachos. Signal-to-noise ratio, error and uncertainty of piv measurement. 2014.

Annexes

.1 RiverApp - Instructions première utilisation

RiverApp : Instructions pour une première utilisation

DUFROMONT Célestin LAURENT Pierre

Ce document reprend des instructions à suivre lors d'une première utilisation de RiverApp afin de vérifier que l'installation de tous les fichiers, packages et environnements a eu lieu sans souci. Pour ce faire, nous allons performer une mesure basique du début à la fin. Les vidéos ainsi que les données nécessaires sont reprises dans le reste du dossier.

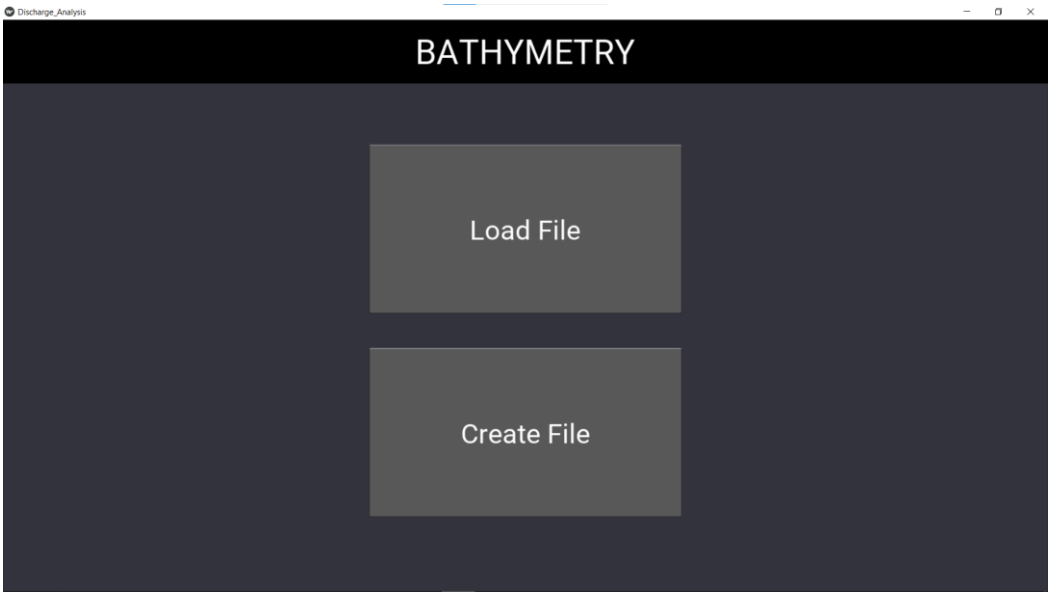
Attention, l'application n'étant pas encore très robuste, nous vous conseillons de prendre le temps de réaliser les étapes très minutieusement, sous peine de devoir recommencer le processus.

1. Lancer l'application

La première étape est bien sûr de lancer l'application. Pour cela, il faut exécuter le fichier PY nommé *riverDischargeApplication.py* avec un interpréteur python.

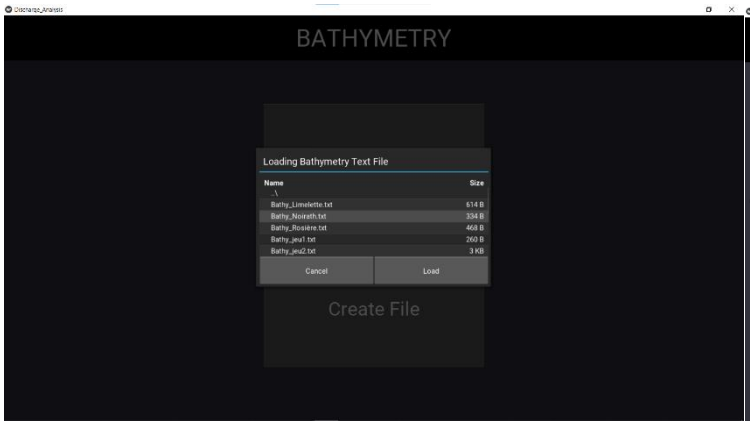
Attention, il ne faut pas confondre avec un autre fichier au nom identique, mais doté d'une extension.*kv* .

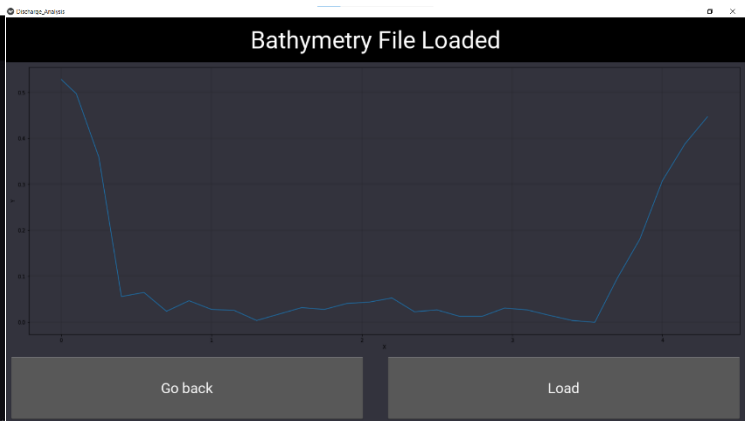
Une fois que cela est fait, la fenêtre pop-up suivante devrait apparaitre :



2. Choix de la bathymétrie

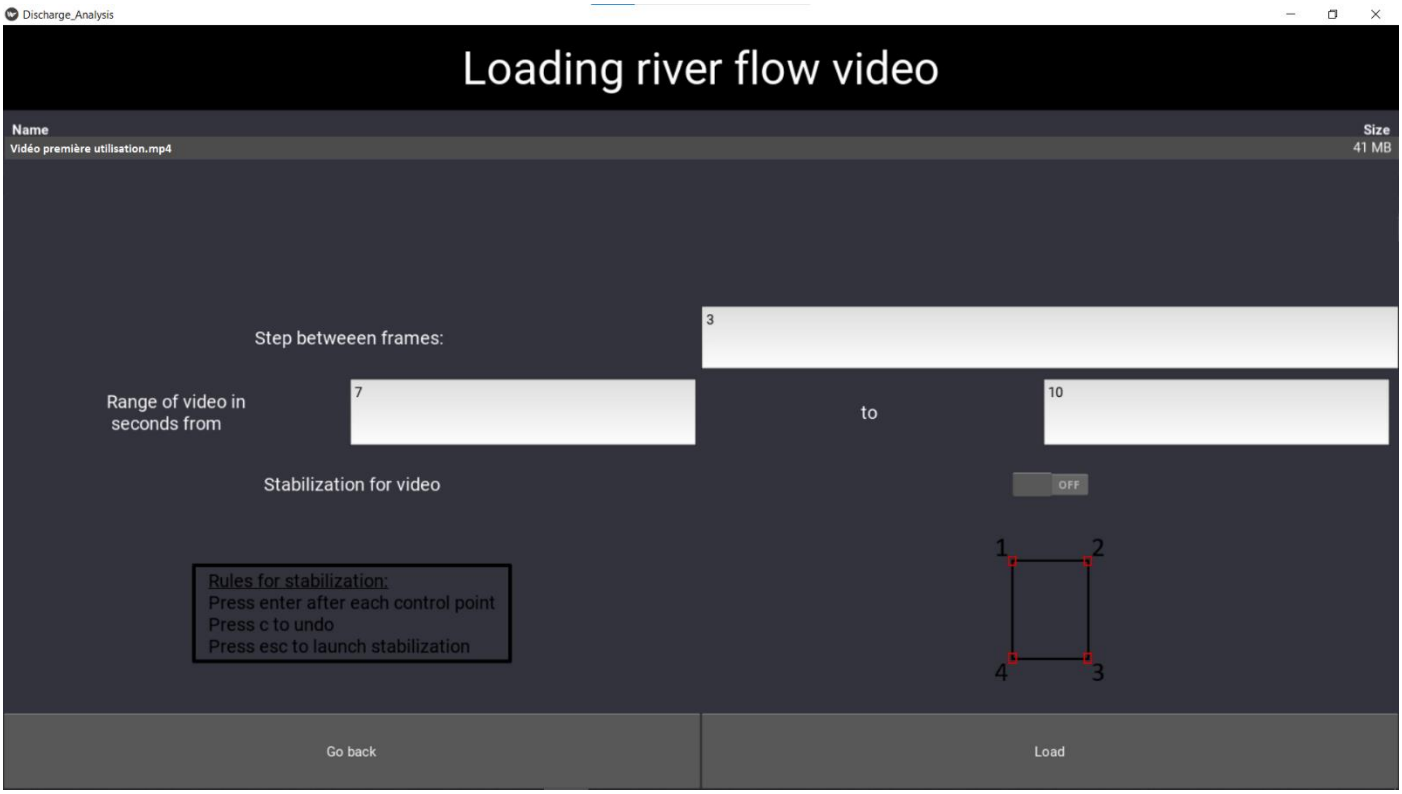
Nous allons faire la mesure sur une bathymétrie connue, il faut donc cliquer sur *Load File* et naviguer jusqu'à choisir la bathymétrie que nous souhaitons utiliser, ici : *Bathy_Noirath.txt*. Ce fichier se trouve dans le dossier *RiverApp – Première utilisation.* Une fois sélectionné, validez avec *Load*. La section devrait apparaitre. Une nouvelle fois, validez avec *Load*.





3. Choix de vidéo et paramètres liés

Il faut maintenant sélectionner la vidéo sur laquelle la mesure va être faite. Naviguez donc jusqu'à celle-ci et cliquez dessus. La durée de la vidéo, soit 19.91 [s] devrait être indiquée dans la case correspondante. Nous allons effectuer la mesure de la 7^{e} à la 10^{e} seconde en prenant une image sur 3. Indiquez donc ces chiffres dans les cases dédiées, comme sur l'image suivante :



Cette vidéo ne nécessite pas de stabilisation donc veillez à ce que le bouton *Stabilization for video* soit bien sur *Off.* Ensuite, validez avec *Load*.

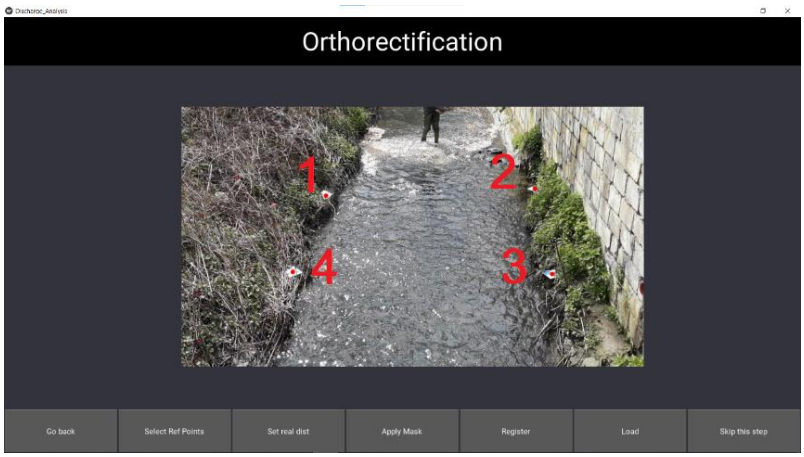
4. Calibration

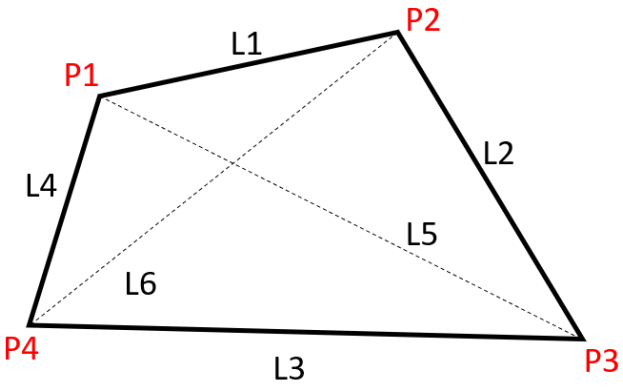
Nous n'effectuerons pas de calibration dans cet exemple. Vous pouvez donc passer cette étape en utilisant sur le bouton *Skip this step* situé en bas de l'écran.

5. Orthorectification

1) Placement des points de référence

Cliquez sur *Select ref points* puis cliquez consécutivement sur chacune des balises visibles sur l'image **dans l'ordre spécifié sur l'exemple ci-dessous** c'est-à-dire en commençant par le point le plus en haut à gauche et en continuant dans le sens horlogique.

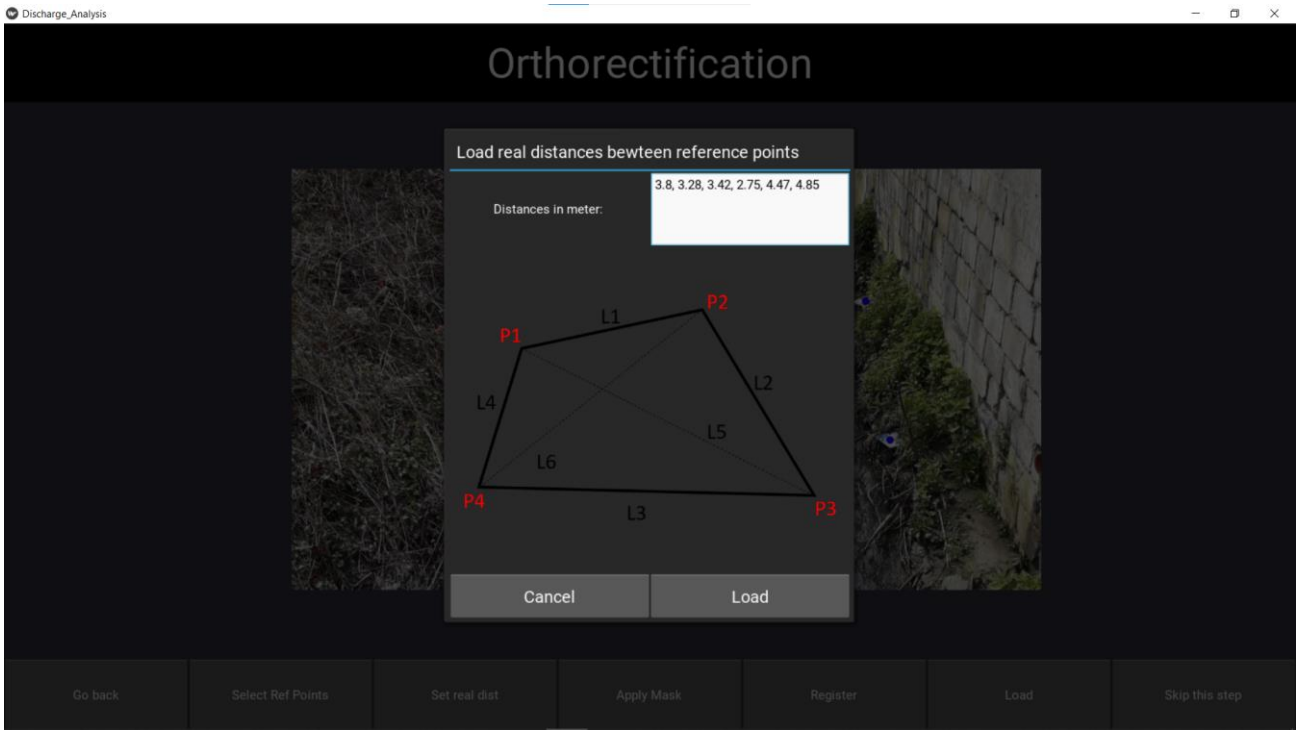




Une fois que cela est fait, appuyez sur *Register*, les points rouges devraient apparaitre bleus.

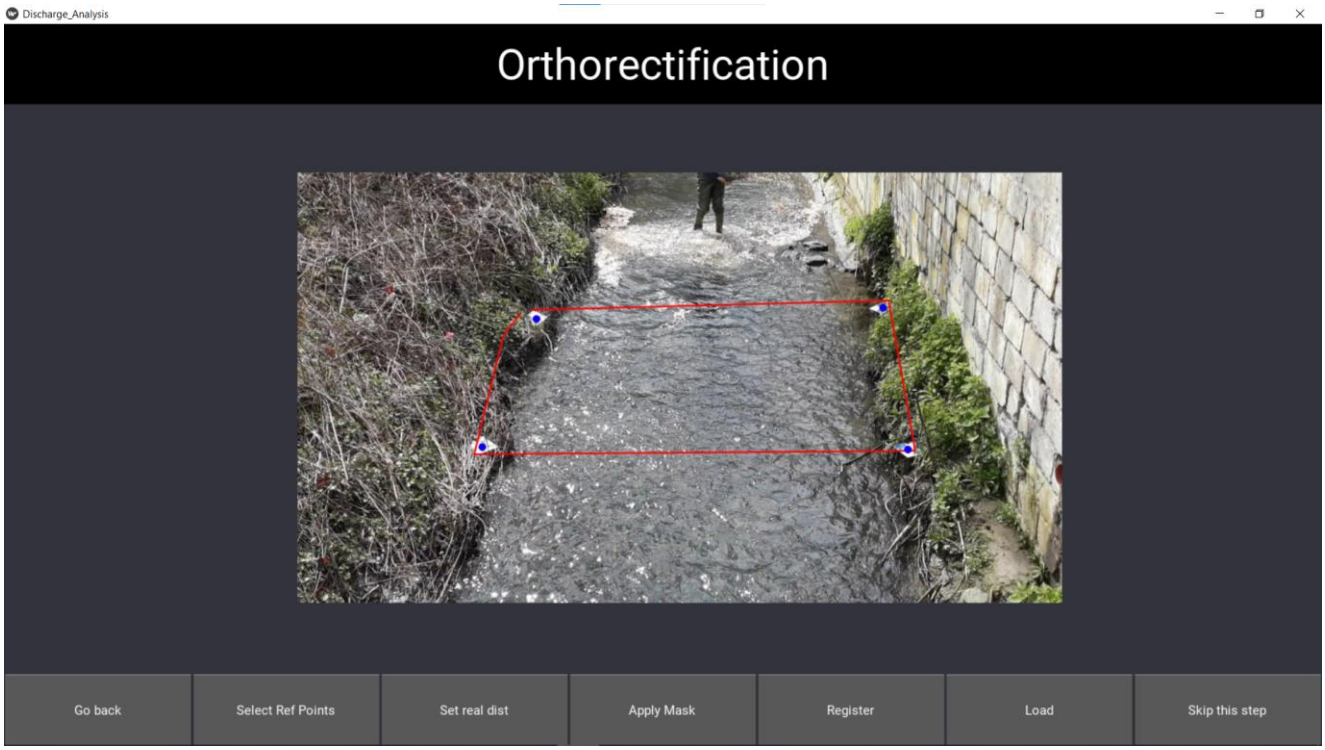
2) Introduction des distances

Cliquez maintenant sur *Set real dist* et copiez-collez dans le pop-up qui s'ouvre les distances données dans le fichier texte. Validez avec *Load.*



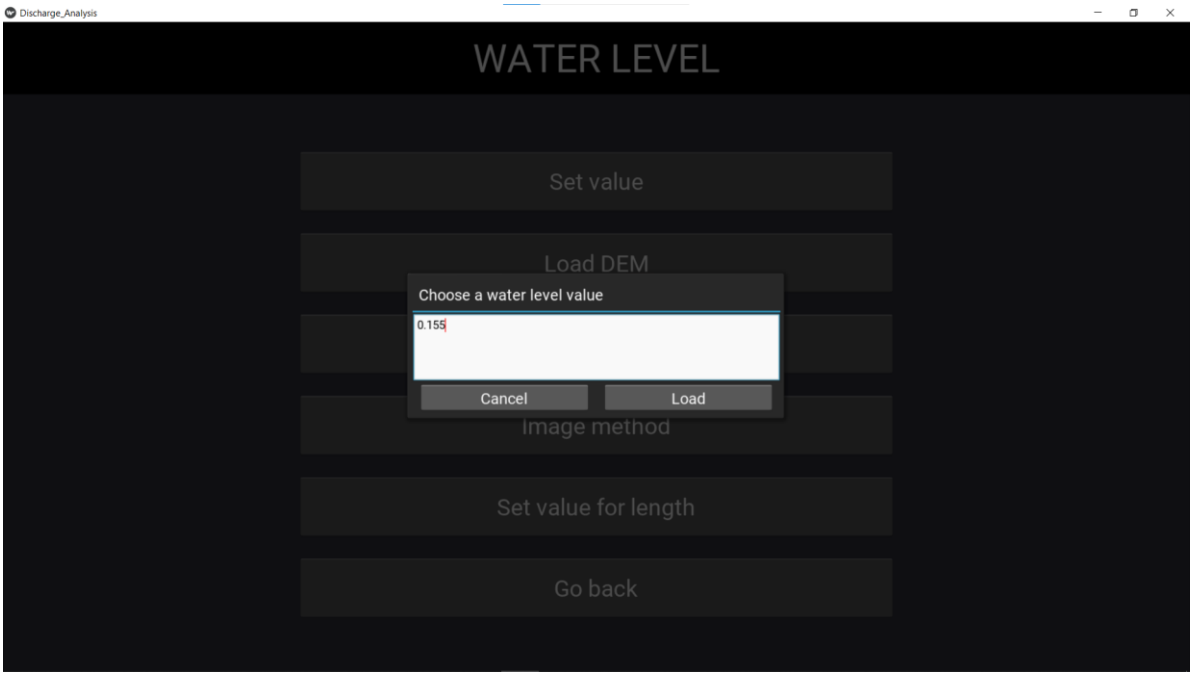
3) Application du masque d'orthorectification

Cliquez sur *Apply Mask*. Ensuite, entourez grossièrement la zone entourée par les 4 points bleus comme dans l'image ci-dessous en cliquant avec votre souris. Appuyez sur *Register*, et les points devraient devenir bleus. Appuyez ensuite sur *Load* pour passer à l'étape suivante.



6. Introduction du niveau d'eau

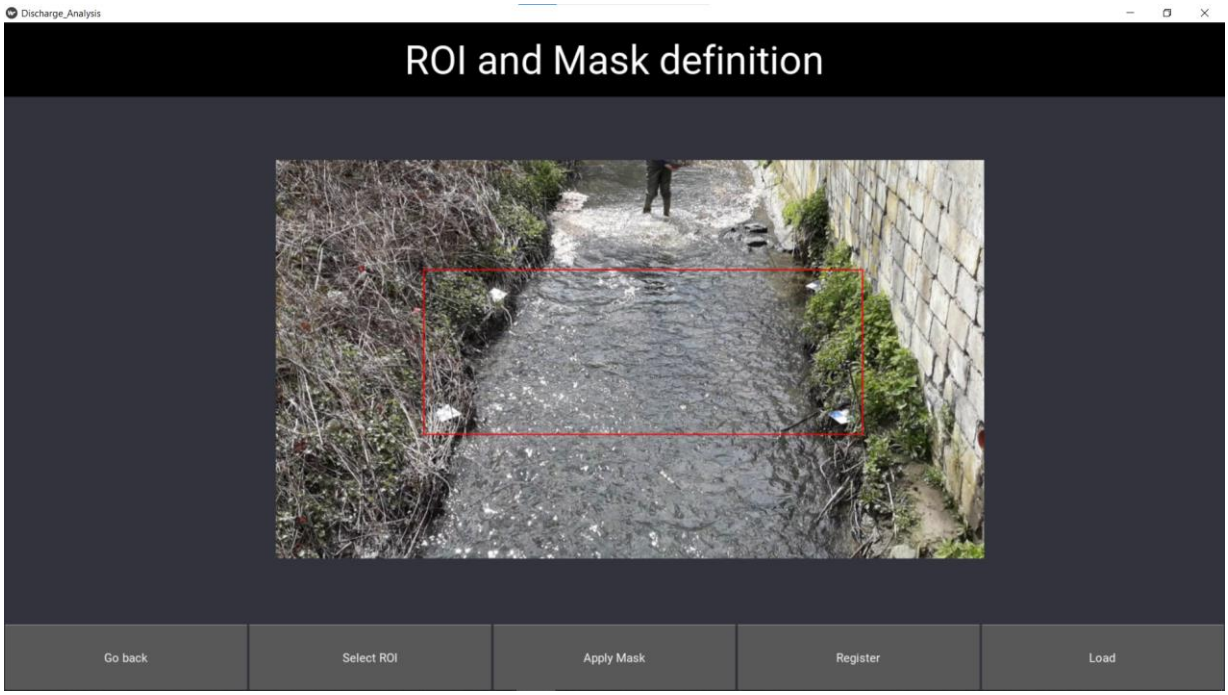
S'ouvre maintenant une fenêtre proposant les différentes options servant à déterminer la profondeur d'eau. Nous allons l'introduire nous-mêmes. Il faut donc appuyer sur la première option : *Set value* et introduire la valeur de **0.155** (avec un **point** et non une virgule). Validez avec *Load.*



7. Zone d'intérêt et masques

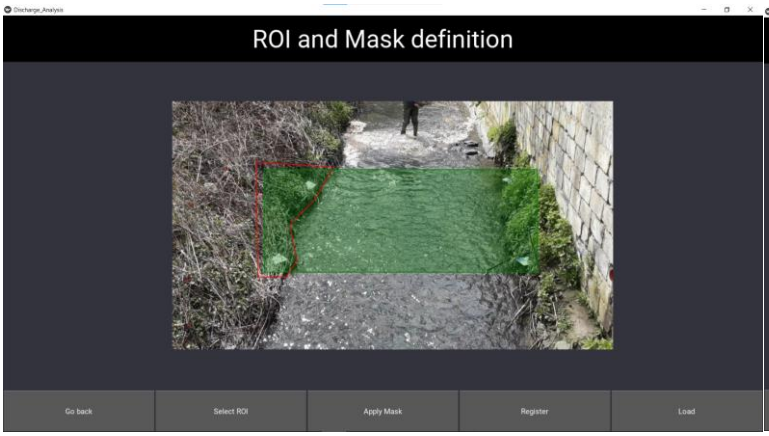
1) Zone d'intérêt (Region Of Interest)

Cliquez sur *Select ROI* et délimitez un rectangle en cliquant à 2 coins opposés de la zone. Cliquez ensuite sur *Register*, la zone précédemment sélectionnée devrait s'afficher en vert translucide.



2) Masques

Appuyez sur *Apply mask* et délimitez un polygone autour de la zone de la berge comprise dans la zone verte comme dans l'image ci-dessous. Validez avec *Register*, la zone devrait devenir rouge translucide. Appuyez à nouveau sur *Apply mask* et faites de même pour l'autre berge. Validez enfin avec *Load*

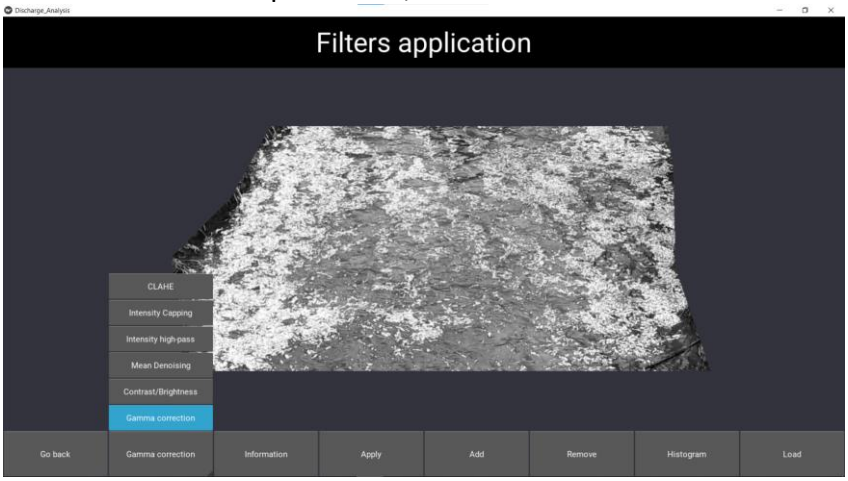


8. Application des filtres

1) Choix du filtre

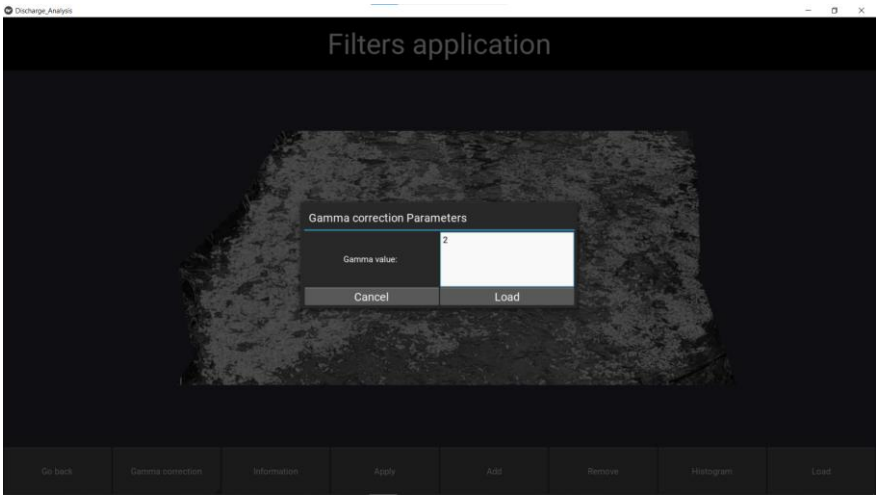
Vous arrivez maintenant sur une fenêtre qui vous montre la portion de rivière en noir et blanc sans les berges.

Appuyez sur *Select filter* et dans le menu qui s'ouvre, sélectionnez *Gamma correction.*



2) Application du filtre

Cliquez sur *Apply mask*. Dans le pop-up qui s'ouvre, changez la valeur par défaut et introduisez *2.* Validez avec *Load.* L'image devrait maintenant s'être assombrie légèrement.



3) Addition du filtre

Cliquez maintenant sur *Add* puis sur *OK* dans le pop-up qui s'ouvre. Appuyez ensuite sur *Load* pour passer à l'étape suivante.

Veillez à n'appuyer qu'une seule fois sur *Add filter.*

9. Analyse PIV

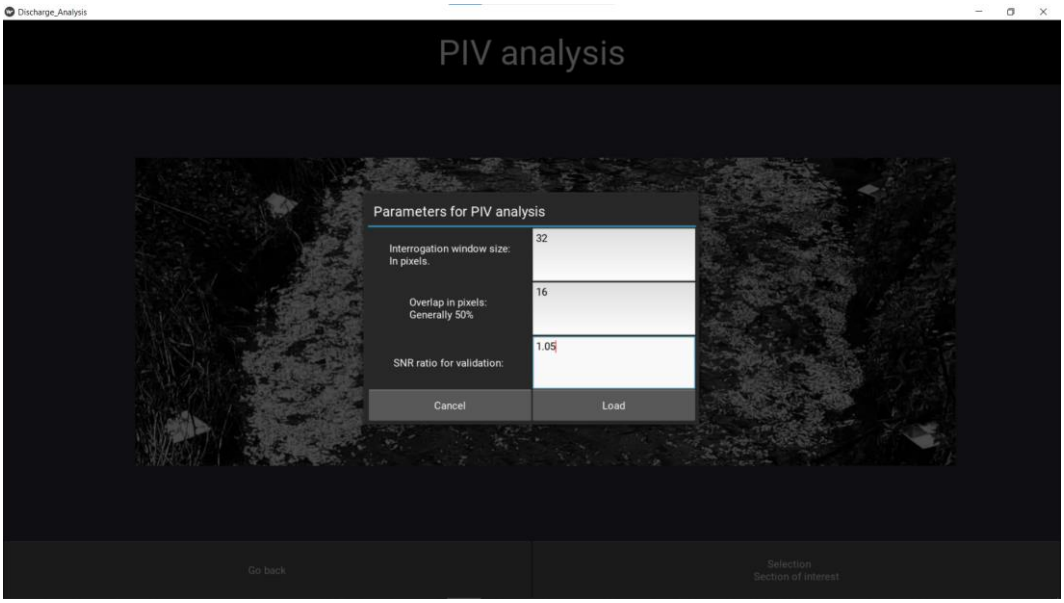
À cette étape, sélectionnez l'option *Single pass.* Ensuite, introduisez les paramètres suivants :

Interrogation windows size in pixels: 32

Overlap, generally 50% :16

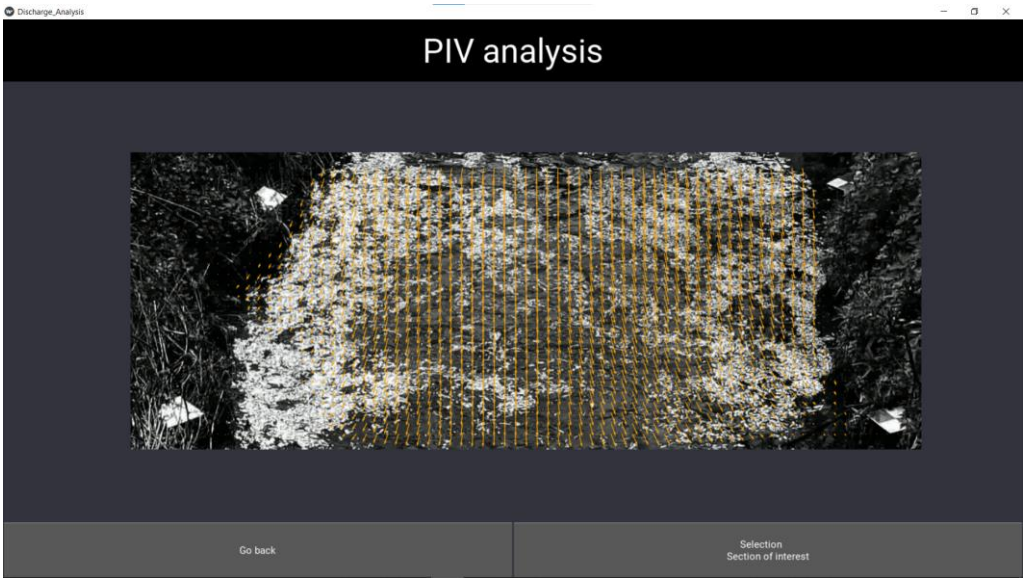
SNR ratio for validation: **1.05**

, comme indiqués sur l'image ci-dessous, et validez avec *Load.*



Un écran de chargement apparait. Avec ces paramètres, l'analyse PIV peut prendre quelques minutes.

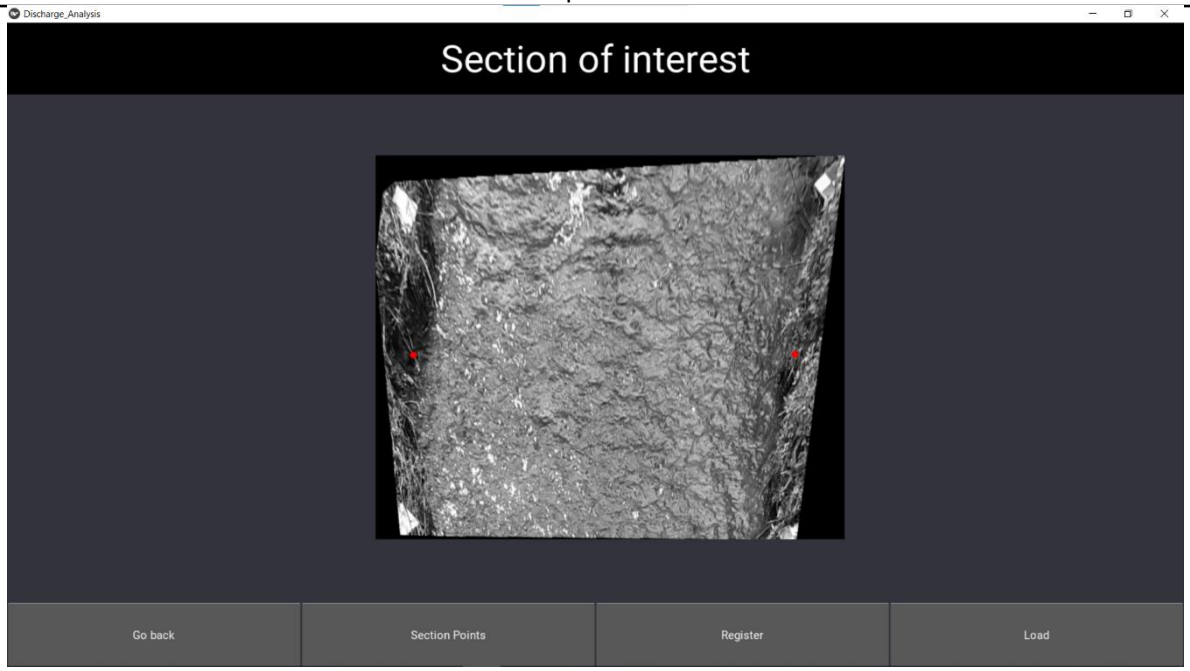
Si vous cliquez malencontreusement quelque part pendant ce chargement, il se peut que la barre de progression disparaisse. L'analyse PIV est malgré cela en train de s'effectuer, attendez alors simplement que l'écran suivant apparaisse.



Passez à la suite via le bouton *Selection section of interest*.

10. Section d'intérêt

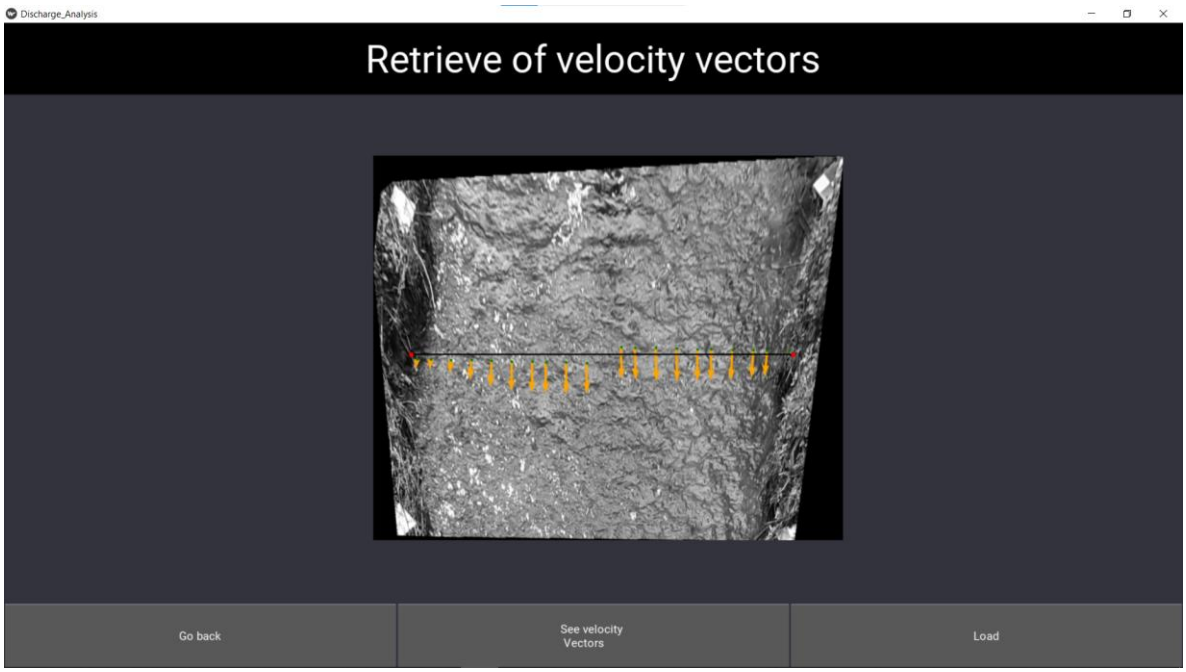
Vous obtenez maintenant une vue de la rivière orthorectifiée. Sélectionnez *Section Points* puis cliquez une fois sur chaque berge comme indiqué sur l'image.



Appuyez ensuite sur *Register* et les points devraient apparaitre en bleu. Passez à la suite avec *Load.*

11. Nombre de sous-sections

L'application va maintenant vous demander le nombre de sous-sections à considérer. Indiquez le chiffre 20 et validez. Appuyez ensuite sur *See velocity vectors* pour obtenir le profil de vitesse sur la section que vous avez choisie. Appuyez ensuite sur *Load* pour passer à l'étape suivante.



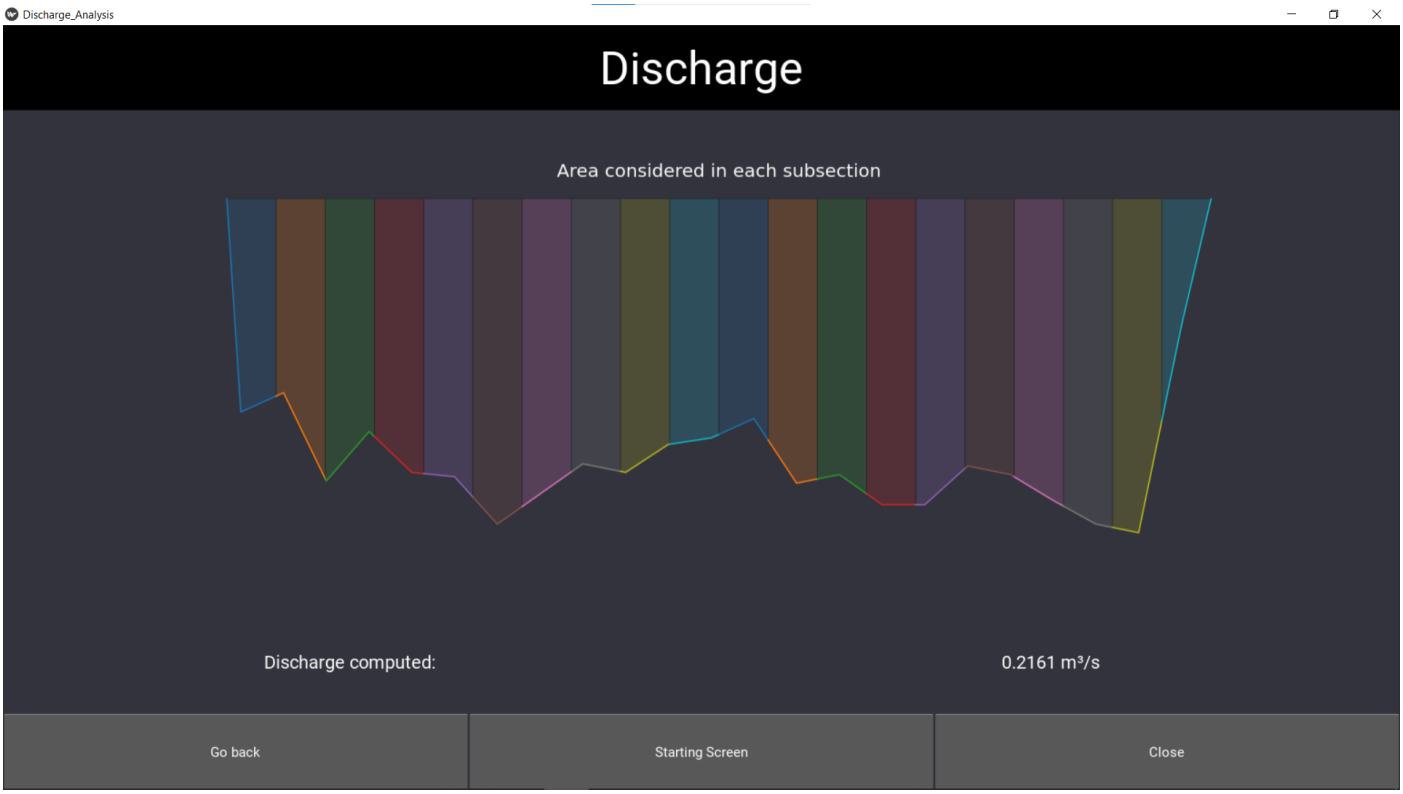
12. Coefficient de vitesse

Dans la fenêtre qui s'ouvre, indiquez le chiffre **0.85.**

13. Résultat

La fenêtre qui s'ouvre maintenant est la dernière. Elle représente les sous-sections qui ont été utilisées pour le calcul final.

Vous avez maintenant obtenu votre premier résultat. Félicitations ! Celui-ci devrait se situer autour de 0.203 [m³/s]. Cependant, il dépend de beaucoup de facteurs qui sont encore à l'étude donc ne vous étonnez pas si ce premier résultat est assez éloigné de cette valeur.



N'hésitez pas à nous contacter si vous avez eu le moindre problème avec ce tutoriel ou l'application, ou si les résultats que vous obtenez vous semblent incohérents.

.2 Données première utilisation

```
=========================================================

Step between frames: 3
Range of videos in seconds from   7  to   10

=========================================================

Distances in meters:

3.8, 3.28, 3.42, 2.75, 4.47, 4.85

(L'idéal est de copier-coller la ligne)

=========================================================

Choose a water level value:

0.155

=========================================================

Gamma value: 2

=========================================================

Interrogation windows size in pixels: 32

Overlap, generally 50% :16

SNR ratio for validation : 1.05

=========================================================

Number of subsections: 20

=========================================================

Coefficient from surface velocity to mean velocity: 0.85

=========================================================
```

FIGURE 1 – Données utilisées pour le kit de première utilisation

.3 Protocole de prises de mesures

RiverApp : Protocole et prise d'images

DUFROMONT Célestin LAURENT Pierre

1. Matériel nécessaire

- 4 GCPs
- Une bombe de couleur/peinture
- Des clous/sardines
- Un marteau
- De la corde (en suffisance)
- Un mètre gradué ou au minimum une latte de 1m de long
- Du ruban adhésif
- Un damier
- Une caméra (plusieurs en fonction des tests)
- Un trépied (par caméra serait le mieux)
- Un long bout de bois/tige métallique/selfie stick
- Des traceurs en suffisance (sciure de bois/copeaux de bois)
- Des claquettes et un maillot ou des bottes et un ciré si vous ne voulez pas être mouillés 😄
- Une éponge
- Un chronomètre
- Kit de mesure (si à disposition)

2. Général

Afin de trier correctement les prises de données ainsi que de pouvoir faire ressortir facilement le paramètre à tester, il convient d'abord de remplir les informations générales de la prise de donnée (et le tableau en dernière page):

Personnes : ……

Lieu : ……

Cours d'eau : …… (Type de lit :……)

Date : ……./……./…… Heure :……..

Prise de vue : ……

Luminosité : ……

Météo : ……

Dans le champ « Luminosité », il est entendu par là que l'on note si la rivière est ombragée ou dégagée, s'il y a des reflets apparent sur la vidéo, etc.

3. Mesure de la bathymétrie (si inconnue)

La connaissance de la bathymétrie est essentielle pour le fonctionnement de l'application et donc pour l'obtention du débit. Pour la mesure de cette bathymétrie, il faudra mesurer à plusieurs intervalles la profondeur du lit du cours d'eau. Pour ce faire, il est intéressant de fixer les intervalles afin de rester précis au maximum. On peut donc imaginer tendre une corde par-dessus le cours d'eau, corde le long de laquelle seraient disposés plusieurs nœuds représentants les différents intervalles. Ensuite, la personne chargée de faire les mesures se munira du mètre gradué en métal ou de la latte afin de mesurer la profondeur au droit de chaque nœud. La photo suivante montre un exemple de cette mesure.

La personne peut également mesurer la profondeur à l'aide d'un bout de bois mais le temps de mesure se verrait significativement augmenté.
Pour ensuite encoder la bathymétrie, il faut créer un fichier .txt et encoder l'une en dessous de l'autre les coordonnées, elles-mêmes séparées par une virgule. Ex : abscisse, ordonnée. Pour les nombres décimaux, c'est le point qui fera office de virgule. Le point le plus à gauche sera le point de départ des abscisses alors que le point le plus profond sera celui des ordonnées.
Même si la bathymétrie est connue, il faut également connaître la profondeur. Il faudra donc déterminer la profondeur au point le plus bas. Celle-ci peut être obtenue via Aqualim si nécessaire.

On peut noter le nom du fichier contenant la bathymétrie ainsi que la profondeur d'eau ci-dessous :

Nom du fichier contenant la coupe de la section : ……

Si possible, hauteur d'eau au point le plus bas : …….

4. Calibration

On peut aussi choisir de calibrer la caméra que l'on utilise. Cette étape n'est pas obligatoire et peut se faire aussi bien avant que après la prise de mesure. Il existe plusieurs techniques décrites dans le guide utilisateur. La plus facile consiste à filmer pendant quelques secondes et en bougeant un peu un damier.

5. Choix de la prise de vue

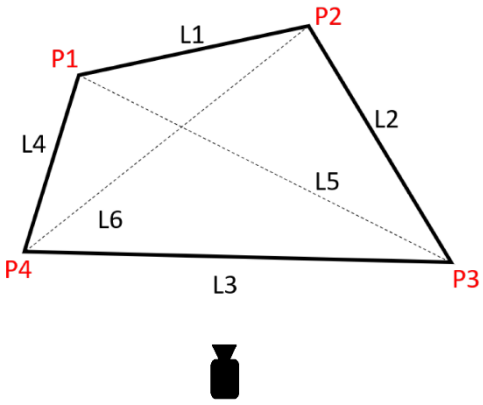
Une fois la rivière et le spot choisi, il faut délimiter la prise de vue de la caméra. Il faut donc zoomer/orienter l'objectif comme on le souhaite ainsi que se déplacer/s'éloigner comme on le désire afin de fixer l'endroit et la position de la caméra. De cette manière, on pourra plus rapidement savoir si les éléments devant être présents dans le champ de vision y résident bien.

6. Avant la prise d'images

Dans le cas où l'angle d'incidence de la caméra par rapport à la surface de l'eau ne serait pas égal à 90°, l'application devra effectuer au préalable une orthorectification afin d'aplatir la vidéo. Pour effectuer cette orthorectification, l'application aura besoin de 4 points fixes. Le but de ces quatre points est de connaître des dimensions précises. La plupart du temps, nous utiliserons des GCPs (Ground Control Points). Ces quatre points de références doivent être placés sur le même plan que celui formé par le plan d'eau et être répartis aux quatre extrémités de la surface étudiée. Ces points peuvent également être des éléments naturels, tant qu'ils sont parfaitement reconnaissables et que leurs positions, tout en respectant les conditions, sont connues.

Afin de fixer un GCP, il faudra se munir du marteau et des clous/sardines pour l'enfoncer au mieux dans la berge. Si la pose du GCP se fait sur une matière plus rocailleuse, le ruban adhésif sera certainement plus opportun. Si, par contre, c'est un élément naturel qui constitue un point fixe, il sera intéressant de le marquer à l'aide de la peinture ou de la bombe de couleur.

Enfin, et une fois les quatre points fixes sont bien identifiés, les distances « L » ci-dessous devront être mesurées à l'aide de la corde et du mètre gradué. Il convient alors d'être suffisamment précis afin que l'orthorectification soit correctement effectuée. Attention, l'ordre des longueurs aura toute son importance lors de l'analyse. Il convient donc de respecter l'ordre établi sur le schéma si dessous.
On pourrait également imaginer renseigner les coordonnées GPS des points de repère et ainsi calculer les distances depuis un ordinateur.



L1	……[m]
L2	……[m]
L3	……[m]
L4	……[m]
L5	……[m]
L6	……[m]

7. Ensemencement

Pour déterminer le débit, l'application doit pouvoir détecter la vitesse de surface du cours d'eau étudié. Dans certains cas, les remous créés par l'environnement seront suffisant. Dans d'autres cas, un ensemencement de la rivière sera nécessaire. Pour ensemencer la rivière, il faudra se munir de traceurs (copeaux/sciure de bois) et se placer en amont de la région d'intérêt. Pour un ensemencement correct de la rivière, il conviendra de jeter, en nombre suffisant, les traceurs sur toute la largeur de la rivière.

8. La vidéo

Il suffit de prendre la vidéo que l'on souhaite. Une certaine hauteur de point de vue sera néanmoins demander pour l'exploitation de la vidéo. Il faut également que la vidéo soit exportable en .mp4 ou .mov pour pouvoir la lire avec l'application. Il est évident que plus la qualité de la vidéo est haute plus cela est intéressant. Cependant, une trop grande résolution ralentit les résultats. A contrario, une résolution trop faible rendra l'utilisation de l'application impossible.

Afin de garder en mémoire l'historique des vidéos faites, il convient de tenir à jour un Excel reprenant toutes les prises effectuées. Si vous ne disposez pas d'un ordinateur, le tableau (situé à la dernière page) sera déjà suffisant.

Une bonne manière de reconnaître facilement le test effectué, est d'annoncer au début de chaque prise les conditions réunies au cours de la vidéo.

9. A volonté

Les étapes 7 et 8 peuvent être recommencées autant de fois que vous le désirez. Seuls la mémoire des caméras et/ou les traceurs pourraient vous empêcher de continuer. Pour que la première n'arrive pas ainsi que pour vérifier le bon angle de prise de vue, il est intéressant de prendre un ordinateur portable afin de transférer les vidéos et/ou d'en vérifier la qualité.

10. Le débit

Pour pouvoir valider l'application, il sera intéressant de comparer le débit calculé au débit réel. Si l'endroit de mesure se trouve au niveau d'une station Aqualim (en Wallonie), l'information pourra être trouvée sur le site suivant : http://aqualim.environnement.wallonie.be/login.do?time=1659447935637

Si ce n'est pas le cas, il faudra à l'aide de l'éponge, attachée à une corde, et le chronomètre établir le temps moyen (autour de 20 prises) que l'éponge met pour parcourir la distance de la corde. Cette vitesse de surface, combinée à la section du cours d'eau, donnera le débit. Bien que pratique pour évaluer la vitesse de surface, cette technique reste approximative et il est préférable d'obtenir le débit ou la vitesse de surface directement via des sites internet officiels tels que Aqualim ou par le biais d'instrument de mesures dédié à cet effet.

La section peut également être calculée à partir de la bathymétrie, à l'aide d'une fonction informatique.

```python
# -*- coding: utf-8 -*-
"""
Created on Mon Jul 25 12:41:30 2022

@author: coper
"""

import numpy as np

Coord = np.zeros((25,2))
aire = 0

with open("NomDeFichier.txt", "r") as fichier:
    sp = fichier.read().split('\n')
    
for i in range (25):
    data = sp[i].split(',')
    Coord[i,0] = float(data[0])
    Coord[i,1] = abs(float(data[1]) - PROFONDEUR DE L EAU)
    
for j in range (len(Coord)-1):
    aire += ((Coord[j,1] + Coord[j+1,1]) * (Coord[j+1,0] - Coord[j,0]))/2

print (aire)
```

Une fois obtenu, le débit peut être noté ci-contre :……[m³/s]

Nom du fichier	Heure de prise	Stabilisée ?	Faite pour ?

.4 Récapitulatif des tests effectués sur le terrain

Nom de la vidéo	Utilisée pour	Temps traceurs	Stable	Rem	Nom du fichier MP4	Heure de prise	Durée de prise
1 - Noirath - 12.7.22	Test	-	OUI		GX018563	12h36'14''	20''
2 - Noirath - 12.7.22	Test	25 -> 30	OUI		GX018571	13h23'36''	1'21''
3 - Noirath - 12.7.22	Stabilisation	54 -> 64	OUI	Même moment que 4	GX018572	13h25'22''	1'12''
4 - Noirath - 12.7.22	Stabilisation	27 -> 34	NON	Même moment que 3	GX018111	13h25'18''	41''
5 - Noirath - 12.7.22	Stabilisation	22 -> 40	OUI	Même moment que 6	GX018573	13h26'28''	48''
6 - Noirath - 12.7.22	Stabilisation	14 -> 24	NON	Même moment que 5	GX018112	13h26'26''	38''
7 - Noirath - 12.7.22	Stabilisation	10->15	OUI	Même moment que 8	GX018574	13h27'18''	33''
8 - Noirath - 12.7.22	Stabilisation	8->14	NON	Même moment que 7	GX018113	13h27'16''	28''
9 - Noirath - 12.7.22	Beaucoup de copeaux d'un coup	10->14	OUI	Même moment que 10	GX018575	13h27'46''	21''
10 - Noirath - 12.7.22	Beaucoup de copeaux d'un coup	8->11	NON	Même moment que 9	GX018114	13h27'44''	16''
11 - Noirath - 12.7.22	Très peu de copeaux	10->13	NON	Même moment que 12	GX018115	13h28'26''	20''
12 - Noirath - 12.7.22	Très peu de copeaux	9->13	OUI	Même moment que 11	GX018576	13h28'26''	19''
13 - Noirath - 12.7.22	Beaucoup de copeaux d'un coup	7->11	OUI	Même moment que 14	GX018577	13h28'52''	18''
14 - Noirath - 12.7.22	Beaucoup de copeaux d'un coup	5->9	NON	Même moment que 13	GX018116	13h28'52''	15''
15 - Noirath - 12.7.22	Copeaux mis plus en amont	44 -> 60	OUI	Même moment que 16	GX018578	13h30'54''	1'13''
16 - Noirath - 12.7.22	Copeaux mis plus en amont	37->53	NON	Même moment que 15	GX018117	13h30'52''	1'03''
17 - Noirath - 12.7.22	Kit en amont	35 ->45	OUI		GX018581	13h48'32''	54''
18 - Noirath - 12.7.22	Kit en aval	36 ->45	OUI		GX018582	13h49'56''	50''
19 - Noirath - 12.7.22	Kit en aval - Grand angle	12-> 20	OUI		GX018584	13h51'18''	31''
20 - Noirath - 12.7.22	Kit en amont - Grand angle	28 -> 33	OUI		GX018585	13h52'26''	42''
21 - Noirath - 12.7.22	Kit très en amont - Grand angle	6->11	OUI		GX018586	13h53'58''	42''
22 - Noirath - 12.7.22	Qualité -2704x1520	40 -> 49	OUI	3e GCP non-visible	GH018588	14h01'16''	57''
23 - Noirath - 12.7.22	Qualité -3840x2160	12 -> 19	OUI	3e GCP non-visible	GH018590	14h02'24''	31''
24 - Noirath - 12.7.22	Qualité -1920x1080	14 -> 22	OUI	3e GCP non-visible	GX018591	14h03'12''	34''

FIGURE 2 – Récapitulatif des tests effectués sur le site de Noirhat

Nom de la vidéo	Utilisée pour	Temps traceurs	Stable	Rem	Nom du fichier MP4	Heure de prise	Durée de prise
GX018137	Stab	28->38	NON		GX018137	13:18:46	47
GX018138	Stab	27->37	NON		GX018138	13:20:36	47
GX018140	Stab	30->38	NON		GX018140	13:24:28	45
GX018144	2 angles	33->43	OUI		GX018144	13:35:40	53
GX018145	Bcp de copeaux	33->43	OUI		GX018145	13:37:02	44
GX018146	Peu de copeaux	28->38	OUI		GX018146	13:38:16	49
GX018147	Kit en amont	22->32	OUI		GX018147	13:47:30	51
GX018148	Kit en amont	28->38	OUI		GX018148	13:48:48	58
GX018150	Kit en amont (en oblique)	26->36	OUI		GX018150	13:58:52	45
GX018151	Kit en aval	24->34	OUI		GX018151	14:01:32	53
GX018152	Kit en aval	40->50	OUI		GX018152	14:08:56	1'09
GX018153	Kit non tenu	29->39	OUI		GX018153	14:10:42	51
GX018154	Kit non tenu	18->28	OUI		GX018154	14:12:32	33
GX018155	Test	22->32	OUI		GX018155	14:19:16	58
GX018156	Focalisation	25->35	OUI	Sur ombre	GX018156	14:20:48	53
GH018157	Qualité	24->34	OUI	2704x1520	GH018157	14:24:18	42
GX018158	Qualité	24->34	OUI	1920x1080	GX018158	14:26:08	46
GX018160	Qualité	26->36	OUI	1920x1080	GX018160	14:30:08	56
GX018161	Énormément de copeaux	30->1'07	OUI	Reste de copeaux	GX018161	14:33:00	1'25
GX018610	Test	57->1'05	OUI		GX018610	13:10:00	1'21
GX018611	Test	30->45	OUI		GX018611	13:11:00	53
GX018612	Test	20->26	OUI		GX018612	13:12:00	42
GX018139	Stab	24->32	NON		GX018139	13:22:38	42
GX018613	Stab	30->40	OUI		GX018613	13:18:46	50
GX018614	Stab	27->37	OUI	Très sombre	GX018614	13:20:36	47
GX018615	Stab	24->32	OUI	Très sombre	GX018615	13:24:28	42
GX018617	2 angles	33->43	OUI		GX018617	13:35:40	52
GX018619	Peu de copeaux	33->43	OUI		GX018619	13:38:16	1'02
GX018621	Kit en amont	25->35	OUI		GX018621	13:48:48	44
GX018622	Kit en amont (en oblique)	28->38	OUI		GX018622	13:58:52	1'01
GX018623	Kit en amont (en oblique)	32->42	OUI		GX018623	13:56:30	1'01
GX018624	Kit en aval	33->43	OUI		GX018624	14:01:32	52
GX018625	Kit en aval	33->43	OUI		GX018625	14:08:56	1'10
GX018629	Focalisation	25->35	OUI	Sur soleil	GX018629	14:20:48	55
GX018630	Qualité	22->32	OUI	1920x1080	GX018630	14:24:18	43
GX018632	Qualité	24->35	OUI	1920x1080	GX018632	14:26:08	47
GX018633	Qualité	26->36	OUI	3840x2160	GX018633	14:28:00	56
GX018634	Énormément de copeaux	26->58	OUI	Reste de copeaux	GX018634	14:33:00	1'10
GX018616	Stab	30->38	OUI	Très sombre	GX018616	13:22:38	45
GX018149	Kit en amont (en oblique)	32->42	OUI		GX018149	13:56:30	59
GX018159	Qualité	25->35	OUI	3840x2160	GX018159	14:28:00	49
GX018618	Bcp de copeaux	33->43	OUI		GX018618	13:37:02	57
GX018620	Kit en amont	30->40	OUI		GX018620	13:47:30	51
GX018626	Kit non tenu	20->30	OUI		GX018626	14:10:42	53
GX018627	Kit non tenu	36->36	OUI		GX018627	14:12:32	46
GX018628	Test	22->32	OUI		GX018628	13:13:00	57

FIGURE 3 – Récapitulatif des tests effectués sur le site de Suzeryl

Nom de la vidéo	Utilisée pour	Temps traceurs	Stable	Rem	Nom du fichier MP4	Heure de prise	Durée de prise
1 - Lissoir - 10.7.22	Prise axiale sans traceurs	-	NON		AxeSansTrac	18h10'45''	14''
2 - Lissoir - 10.7.22	Prise axiale avec traceurs	31 -> 39	NON	Sciure de bois	AxeAvecTrac	18h11'42''	42''
3 - Lissoir - 12.7.22	Prise oblique sans traceurs	-	NON		CoteSansTrac	18h51'14''	14''
4 - Lissoir - 12.7.22	Prise oblique avec traceurs	9 -> 25	NON	Sciure de bois	CoteAvecTrac1	18h51'55''	33''
5 - Lissoir - 12.7.22	Moins de traceurs	11 -> 21	NON	Sciure de bois (moins)	CoteAvecTrac2	18h53'02''	30''
6 - Lissoir - 12.7.22	Type de traceurs	8 -> 15	NON	Feuilles	CoteAvecTrac3	18h53'54'	20''
7 - Lissoir - 12.7.22	Type de traceurs	6 -> 18	NON	Mie de pain	CoteAvecTrac4	18h55'08'	26''
8 - Lissoir - 13.7.22	Luminosité sans traceurs	-	NON	En soirée	CoteSansTracSoir	21h37'39''	15''
9 - Lissoir - 13.7.22	Luminosité avec traceurs	8 -> 25	NON	Mie de pain, en soirée	CoteAvecTracSoir	21h38'27''	33''
10 - Lissoir - 10.7.22	Test d'un kit	9 -> 20	NON	Mie de pain	TestKit	-	30''

FIGURE 4 – Récapitulatif des tests effectués sur le site de Lissoir

Nom de la vidéo	Utilisée pour	Temps traceurs	Stable	Rem	Nom du fichier MP4	Heure de prise	Durée de prise
1 - Limelette - 21.7.22	Qualité -1920x1080 24FPS	16 -> 25	OUI	Peu de copeaux de bois	GX019085	12h38'	36''
2 - Limelette - 21.7.22	Qualité -1920x1080 24FPS	13 -> 30	OUI	Plus de copeaux de bois	GX019086	12h39	37''
3 - Limelette - 21.7.22	Qualité -1920x1080 24FPS	15 -> 30	OUI	Encore plus de copeaux de bois	GX019087	12h40'	34''
4 - Limelette - 21.7.22	Qualité -3840x2160 24FPS	18 -> 32	OUI		GX019088	12h43'	36''
5 - Limelette - 21.7.22	Qualité -1920x1080 30FPS	16 -> 30	OUI		GX019090	12h45	39''
6 - Limelette - 21.7.22	Kit en amont	11 -> 25	NON	Même moment que 7	GX018417	12h59'	30''
7 - Limelette - 21.7.22	Kit en amont	13 -> 25	OUI	Même moment que 6	GX019091	12h59'	35''
8 - Limelette - 21.7.22	Kit centré	10 -> 22	NON	Même moment que 9	GX018418	13h01'	42''
9 - Limelette - 21.7.22	Kit centré	13 -> 24	OUI	Même moment que 8	GX019092	13h02'	50''
10 - Limelette - 21.7.22	Kit aval	10 -> 22	NON	Même moment que 11	GX018419	13h04'	34''
11 - Limelette - 21.7.22	Kit aval	12 -> 24	OUI	Même moment que 10	GX019093	13h04'	40''
12 - Limelette - 21.7.22	Kit libre	20 -> 31	NON	Même moment que 13	GX018420	13h06'	39''
13 - Limelette - 21.7.22	Kit libre	25 -> 35	OUI	Même moment que 12	GX019094	13h07'	45''
14 - Limelette - 21.7.22	Sans traceurs	-	OUI		GX019095	13h14'	15''
15 - Limelette - 21.7.22	Angle avec la surface: 20°	18 -> 30	NON		GX018448	14h14'	55''
16 - Limelette - 21.7.22	Angle avec la surface: 15°°	61 -> 71	NON		GX018449	14h53'	1'30''
17 - Limelette - 21.7.22	Angle avec la surface: 12,5°	34 -> 46	NON	3ème GCP non visible	GX018450	14h56'	1'10
18 - Limelette - 21.7.22	Angle avec la surface: 10°	63 -> 78	NON		GX018451	14h58'	1'46''
19 - Limelette - 21.7.22	Enormément de copeaux de bois	17-> 45	OUI		GX019098	15h04'	54''

FIGURE 5 – Récapitulatif des tests effectués sur le site de Limelette

.5 Tableau de compatibilité des licences Open Source

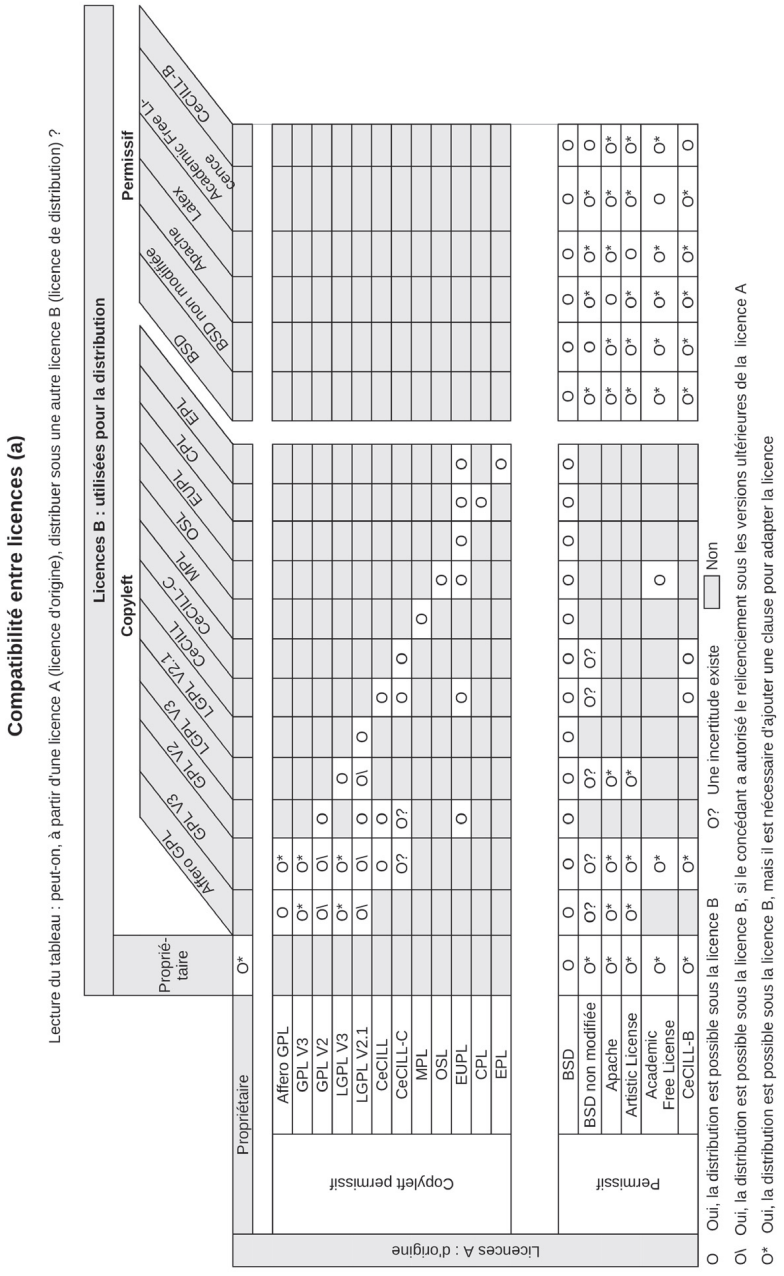


FIGURE 6 – Tableau de compatibilité des licences Open Source [33]

.6 Liste des licences des packages importés

Package	Version (le cas échéant)	Licence
Kivy [36]	1.11.1	MIT
Numpy [40]	1.18.5	BSD
Opencv [42]	4.0.1	BSD
Glob [48]	-	BSD
Imutils [10]	0.5.3	MIT
Re [52]	-	CNRI Python License
Math [51]	-	MIT
Threading [53]	-	MIT
Pathlib [50]	1.0.1	MIT
Functools [47]	-	BSD
Scipy [60]	1.5.0	BSD
Tifffile [45]	2019.7.26.2	BSD
Matplotlib [39]	3.2.2	BSD
Pillow [46]	9.0.1	HPND
Openpiv [49]	0.23.4	GPL 3

TABLE 1 – Liste des licences des packages importés

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**