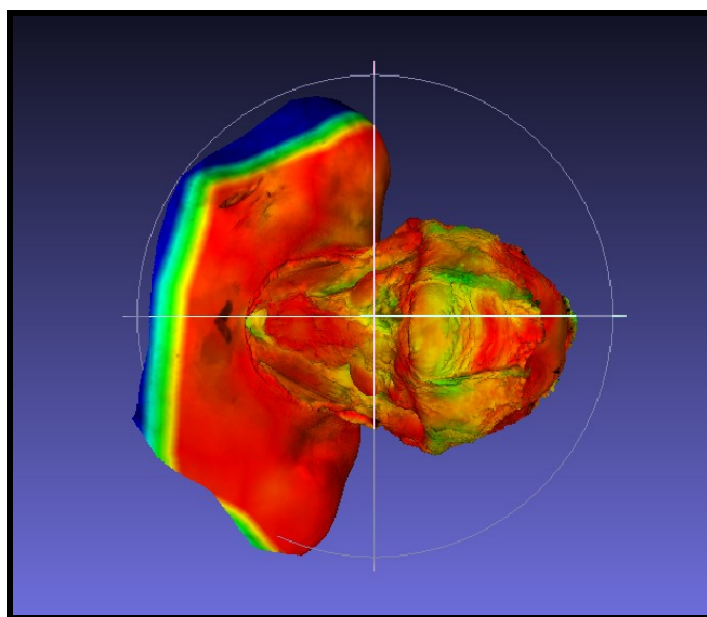


# Rapport de stage

*Du 10/02/20 au 25/07/20*

## « Traitement de maillages 3D pour la simulation de dissection chirurgicale »

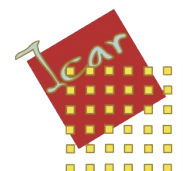


Rédigé par **Thibault ODORICO**

Tuteurs de stage : **Guillaume Captier - Noura Faraj - Gérard Subsol**  
Tuteur universitaire : **Pascal Giorgi**

***Pour l'obtention du Master IMAGINA***

*2020-2021*



# Résumé

L'enseignement de la médecine, depuis plusieurs années, tend à améliorer sa pédagogie en utilisant de nouveaux outils informatiques. Il existe déjà quelques outils qui permettent de visualiser le corps humain, permettant un apprentissage plus concret. Toutefois, l'amélioration du réalisme de ces représentations reste un point important dans ce contexte. C'est pourquoi, un projet novateur a été proposé visant à créer un outil pédagogique de simulation d'une dissection chirurgicale basée sur des données acquises lors d'une réelle chirurgie au laboratoire d'anatomie de Montpellier.

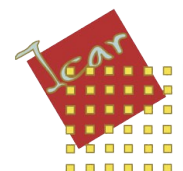
Le présent rapport permettra de détailler les premières expérimentations de ce projet, soit l'application de plusieurs traitements géométriques permettant de réaliser une dissection virtuelle couche par couche.

**Mots-clés : Outils, réalisme, dissection chirurgicale, traitements géométriques, dissection virtuelle.**

# Abstract

The teaching of medicine, for several years, aims to improve its pedagogy by using new computer tools. There are already some tools that allow to visualize the human body, allowing a more concrete learning. However, improving the realism of these representations remains an important point in this context. This is why an innovative project has been proposed to create an educational tool to simulate a surgical dissection based on data acquired during a real surgery at the Montpellier Anatomy Laboratory. This report will detail the first experiments of this project, the application of several geometric treatments to perform a virtual dissection layer by layer.

**Mots-clés : Tools, realism, surgical dissection, geometric treatments, virtual dissection.**



# Remerciements

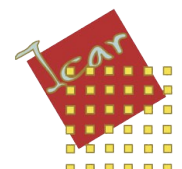
Tout d'abord j'adresse mes remerciements à Monsieur Gérard SUBSOL qui à proposé le sujet ainsi qu'à Madame Noura FARAJ. Ils ont tout deux acceptés de m'encadrer, fait profiter de leur savoir et de leurs grandes expériences lors de la réalisation de mon stage.

Je remercie également Monsieur Pascal GIORGI, pour son accompagnement, ses conseils et son soutien notamment dans mes démarches administratives.

Monsieur Guillaume CAPTIER à réalisé les dissections permettant l'acquisition surfacique nécessaire à la réalisation de mon stage, je le remercie donc pour son travail ainsi que pour son intérêt et sa disponibilité lors des entretiens réalisés avec mes encadrants de stages.

Ce travail sera examiné et évalué par mes professeurs, je leur adresse donc mes remerciements pour avoir accepté de faire partie du jury. J'exprime ma considération et mes vifs remerciements à tous mes enseignants du département des sciences de l'informatique pour les enseignements qu'ils m'ont prodigués tout au long de mes années de formation en Licence informatique et en Master IMAGINA à la faculté des sciences de Montpellier.

Ce stage a été réalisé au sein de l'équipe ICAR du LIRMM pendant une durée d'environ 5 mois et demi, Je tiens donc à témoigner toute ma reconnaissance à l'ensemble de l'équipe ICAR ainsi qu'à tout le personnel administratif et technique de cette institution pour l'assistance et l'aide qu'ils m'ont offertes tout au long de ce stage.

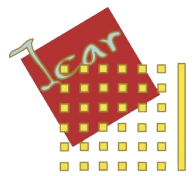


## Table des matières

I. Recherche et Conception.....	13
1) Analyse du problème.....	13
2) Recherches des algorithmes.....	16
3) Algorithme proposé.....	26
II. Développement et Expérimentations.....	29
1) Sélection et mise en place des technologies.....	29
2) Implémentation et expérimentations.....	30
III. Résultats.....	31
1) Résultats de la méthode.....	31
2) Test qualité.....	31

## Table des figures

Figure 1: maillages 3D obtenus lors d'une dissection du cou.....	8
Figure 2: simulation de découpage avec stylet tactile.....	9
Figure 3: série de trois maillages très similaires superposés.....	9
Figure 4: reconstruction des étapes de dissection à partir du Super-Maillage.....	12
Figure 5: maillages non superposables.....	12
Figure 6: séries de maillages.....	13
Figure 7: superposition de la série de maillages.....	13
Figure 8: étapes de construction du Super-Maillage.....	15
Figure 9: comparaison de superposition de maillages idéaux / réalistes.....	15
Figure 10: création du Super-Maillage dans le cas réaliste.....	16
Figure 11: Seuillage de points(modif éloigné en distant).....	19
Figure 12: découpage de maillage en sous-maillages.....	20
Figure 13: étapes de projections MLS.....	22
Figure 14: méthode de projection MLS.....	23
Figure 15: formation de l'option 3.....	24
Figure 16: découpage des transitions entre les maillages.....	24
Figure 17: Super-Maillage contenant parties et liaisons.....	25
Figure 18: étape de l'algorithme de création du Super-Maillage.....	28
Figure 19: visualisation de maillages formant la phrase « hello world » avec notre viewer.....	30
Figure 20: maillages parfaitement superposables.....	31



# Tables des Abréviations

**VESALE-3D** : Virtual Environment for Surgical and Anatomic Learning Enhancement

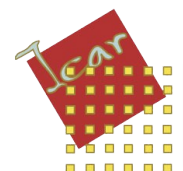
**LIRMM** : Laboratoire Informatique et de Robotique et Microélectronique de Montpellier

**CNRS** : Centre National de Recherche Scientifique

**MLS** : Moving Least Square

**APSS** : Algebraic Point Set Surface

**RAM** : Random Access Memory



# Contexte du stage

Depuis quelques années, l'évolution des nouvelles technologies a permis la création de méthodes d'enseignement numérique à la fois dynamiques et réalistes. C'est pourquoi, dès 2012, la Haute Autorité de Santé encourage la création de simulations informatiques permettant d'apprendre les gestes techniques sans intervenir sur un vrai patient [1]. C'est dans ce cadre que le projet VESALE-3D s'inscrit. Porté par Monsieur Guillaume Captier, Professeur d'anatomie à la faculté de Médecine et par le programme TAKE-OFF - soutien aux innovations pédagogiques du consortium MUSE, l'objectif de ce projet est tout d'abord de numériser lors d'une dissection, toutes les régions anatomiques essentielles à l'apprentissage d'une procédure chirurgicale puis, de traiter ces régions pour permettre à l'utilisateur de les visualiser et d'intervenir dessus en 3D. Ce travail permettra de présenter un outil novateur et pédagogique dans l'enseignement de l'anatomie et la chirurgie médicale.

Le développement informatique du projet est effectué en collaboration avec le LIRMM (Laboratoire d'Informatique, de Robotique et Microélectronique dépendant du CNRS et de l'Université de Montpellier) et plus spécifiquement avec l'équipe ICAR au sein de laquelle ce stage a été réalisé. Cette équipe est spécialisée dans l'interaction et le traitement des données visuelles telles que les images, les vidéos et les objets 3D.

Le présent travail s'inscrit dans le cadre d'un stage de fin d'étude permettant l'obtention du Master IMAGINA spécialisé dans le traitement d'images, la modélisation 3D, la création de moteurs de jeu ainsi que la création d'agents intelligents. Il est encadré par trois tuteurs d'entreprise, un tuteur universitaire et un médecin enseignant, respectivement :

- M. Guillaume Captier : chirurgien plastique pédiatrique et professeur d'anatomie exerçant au CHU et à la faculté de médecine à Montpellier, il est le responsable du projet VESALE-3D et réalisé les dissections utilisées lors du développement de ce projet.
- Mme. Noura Faraj : enseignante chercheuse spécialisée dans l'informatique graphique à l'université de Montpellier, elle travaille dans l'équipe ICAR.
- M. Pascal Giorgi : enseignant chercheur spécialisé dans l'algèbre linéaire, l'arithmétique, l'analyse d'algorithmes et le développement de leurs implémentations les plus optimales, il est à la tête de l'équipe ECO du LIRMM.
- M. Gérard Subsol : chercheur au CNRS, il dirige l'équipe ICAR en travaillant sur divers projets notamment sur les traitements des images 2D et 3D.

Compte tenu de la pandémie du COVID-19, ce stage de cinq mois et demi s'est effectué dans sa majorité en télétravail. C'est pourquoi, des comptes rendus réguliers ont été communiqués et des visioconférences hebdomadaires ont été organisées avec les 3 encadrants.

# Introduction

Depuis plusieurs décennies, la visualisation et l'interaction 3D sont devenus un moyen très pratique pour l'appréhension de l'anatomie humaine. Il existe trois types de visualisations principales dans le cadre de l'apprentissage de la médecine. Tout d'abord, la visualisation voxelique utilisée notamment dans l'imagerie médicale 3D (CT, IRM), introduit d'abord par le projet *Visible Human Project*, qui avait permis la première visualisation 3D complète d'un corps humain en 1998 [2]. Ensuite, la visualisation surfacique schématisée à plusieurs niveaux (os, tissus mous, ...), réalisée par les logiciels du type *Primal's 3D view*, *Anatomography* et *Zygote Body* [3]. Enfin, la visualisation de numérisation surfacique texturée en haute définition permet d'avoir une représentation réaliste. C'est cette visualisation qui sera utilisée dans ce projet pour simuler une dissection réaliste induisant une meilleure appréhension de l'anatomie humaine (affichage de structure anatomiques, texture de la peau, assemblage des muscles, etc).

Dans l'optique d'obtenir cette visualisation réaliste, des séries de modèles 3D surfaciques texturés à très haute définition ont été acquis au laboratoire d'anatomie de la Faculté de Médecine de Montpellier par numérisation surfacique (*Artec Spider*). Ces modèles 3D sont appelés maillages (modèles en trois dimensions composés de plusieurs faces généralement triangulaires). Dans notre application, ces maillages correspondent aux différentes étapes de la dissection du cou. Comme visible sur la figure 1, chaque étape de la dissection correspond à un maillage (il change entre chaque acquisition). Il s'agit donc d'un vrai corps qui a été disséqué et scanné en 3D à multiple reprises.

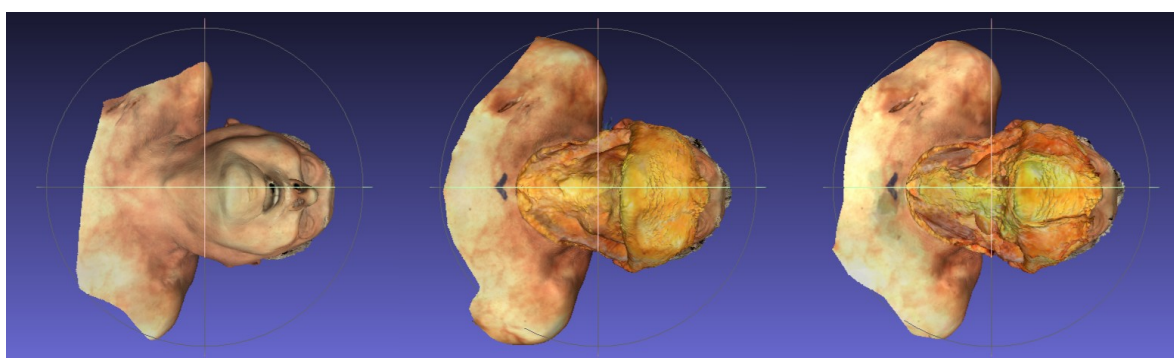


Figure 1: maillages 3D obtenus lors d'une dissection du cou

nbr de point et face pour la complexité du prb à rajouter

Pour faciliter la compréhension, nous appellerons Acquisition un maillage numérisé durant une dissection chirurgicale. On s'aperçoit que d'une étape à une autre, des changements surviennent. On cherche donc ici à séparer les parties changeantes et non changeantes pour pouvoir à la fin reconstruire n'importe quel maillage. Tous les maillages découpés issus des Acquisitions seront réunis dans un Super-Maillage et permettront de faire une Reconstruction d'une étape de

dissection. Les Acquisitions fournies par le laboratoire d'Anatomie doivent donc subir un ensemble de traitements en vue de les réunir dans un Super-Maillage, permettant une Reconstruction.

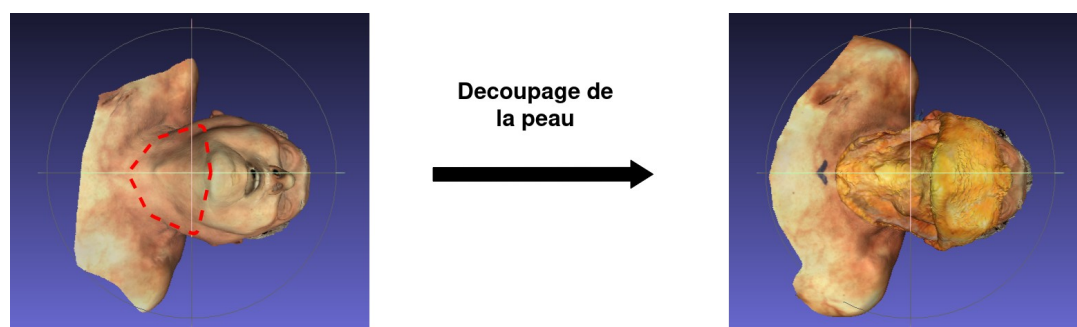


Figure 2: simulation de découpage avec stylet tactile

C'est ici qu'intervient le sujet de ce stage « *Traitement de maillages 3D pour la simulation d'une dissection chirurgicale* ». Concrètement, nous concevrons et développerons une méthode qui prendra en entrée un ensemble d'Acquisitions pour les transformer en Super-Maillage.

Comme visible sur la figure 2, des parties sont inchangées d'une étape à l'autre et constituent une redondance. Ceci est schématisé dans la figure 3 où l'on constate que les extrémités sont identiques. Ainsi, un découpage de ces zones redondantes sera fait afin de supprimer l'information répétée. Les Acquisitions étant à très haute résolution, cette méthode devra faire en sorte que la dissection réalisée soit non seulement réaliste visuellement mais également efficace. Pour cela il devra contenir un découpage intelligent des Acquisitions.

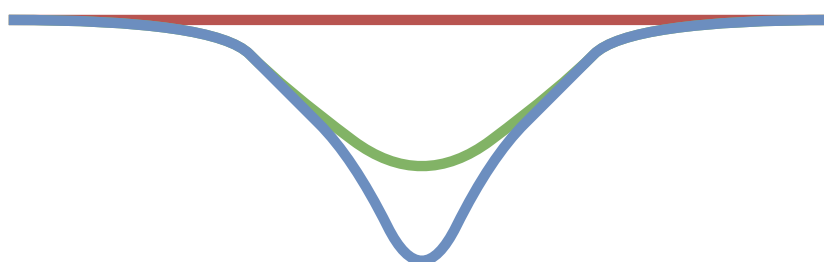


Figure 3: série de trois maillages très similaires superposés

Étant donnée qu'une grande partie de l'information des Acquisitions est redondante, le Super-Maillage sera composé de Maillages issus d'Acquisitions réduites. La réduction de données devra être telle, qu'elle permettra toutefois de reconstruire chaque Acquisitions à partir des Maillages composant le Super-Maillage. Nous aurons donc une perte minimale d'information et un gain important en mémoire.

La figure 4 illustre le Super-Maillage. Il est représenté par le cercle et contient des maillages découpés issus de M1, M2 ou M3. La Reconstruction de M1, M2 ou M3 est possible en assemblant les maillages découpés du Super-Maillage.



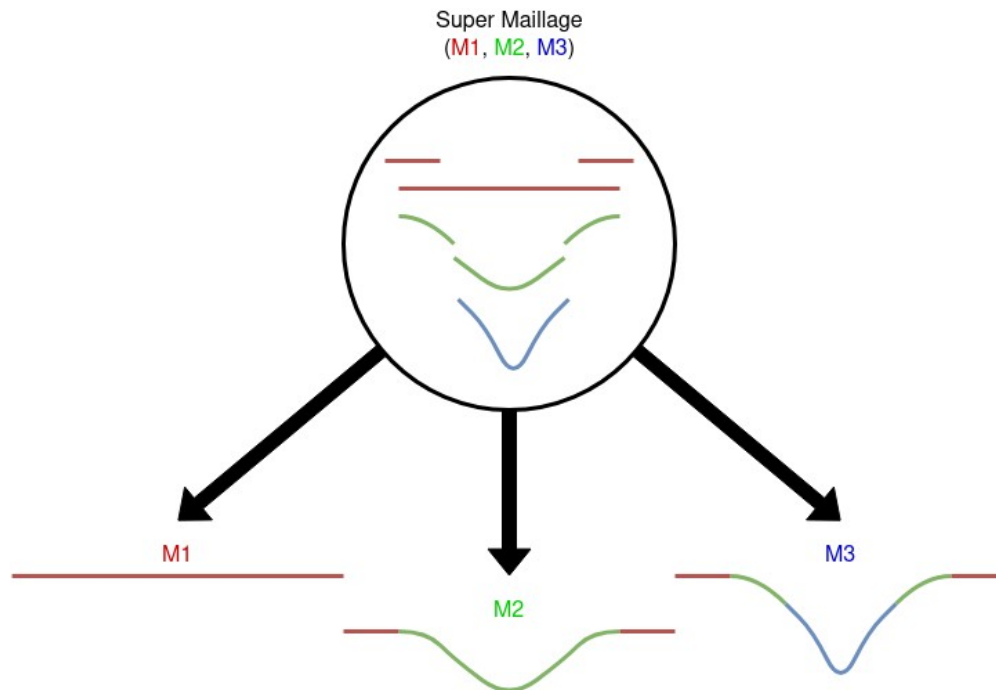


Figure 4: reconstruction des étapes de dissection à partir du Super-Maillage

Pour que la Reconstruction fonctionne, il faudra porter une attention toute particulière au fait que nos maillages ne sont pas parfaitement superposables entre eux. En effet, plusieurs phénomènes peuvent perturber l'acquisition (déplacement légers et involontaires du corps pendant la dissection, affaissement de certains tissus mous pendant la dissection qui dure plusieurs heures (cf. figure 1)). Ces biais d'acquisition compliquent la Reconstruction. Comme visible sur la figure suivante, les décalages ne permettent pas de reconstruire directement les Acquisitions. Il faudra créer des liaisons pour corriger ça.

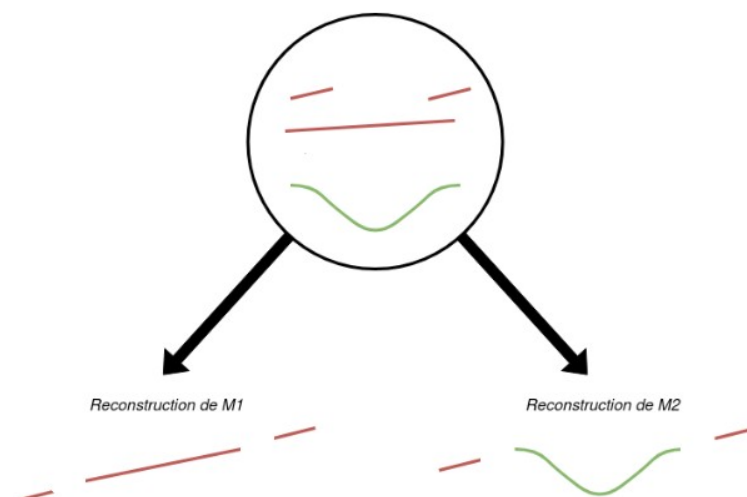


Figure 5: maillages non superposables

Tout au long de ce rapport nous tenterons donc de répondre à la problématique suivante « Comment créer un « Super-Maillage » à partir d'un ensemble de maillages 3D afin de permettre une reconstruction réaliste et efficace des étapes d'une dissection ? ».

Dans une première partie, nous détaillerons les étapes de conception qui ont été nécessaires pour proposer une méthode de création d'un Super-Maillage. Puis dans un second temps, nous détaillerons tout le développement de l'application qui a été fait pour amener à une dernière partie exposant les résultats obtenus ainsi que leurs interprétations.

## I. Recherche et Conception

### 1) Analyse du problème

Nous allons d'abord partir du cas simple (sans « bruit » ni incertitude) pour amener à la complexité du problème. Soit 3 Maillages, M1, M2, M3 représentés ci-dessous de profil (cf figure 3) :



Figure 6: série de maillages

En superposant les Maillages on peut se rendre compte que des parties sont partagées, il y a donc de la redondance dans les données.

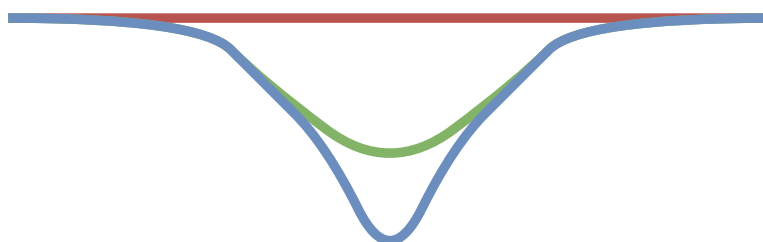


Figure 7: superposition de la série de maillages

L'idée ici est de parcourir la série de maillages à l'aide d'un algorithme. En s'appuyant sur la figure ci-dessous, par lecture de gauche à droite, il faut :

- Détecter les limites entre les maillages présentant une redondance comme représenté sur la ligne 1.
- Découper les deux maillages, supprimer la redondance de l'information en ne gardant qu'un unique sous-maillage commun pour toutes les Acquisitions, schématisé à la ligne 2.
- Greffer les parties restantes entre elles afin de s'assurer d'une reconstruction cohérente (surface continue sans trous), représenté à la dernière ligne.
- Stocker les parties utiles dans un Super-Maillage en prenant soin de noter les associations pour pouvoir par la suite reconstruire notre série de maillages au complet (le cercle schématise le Super-Maillage).

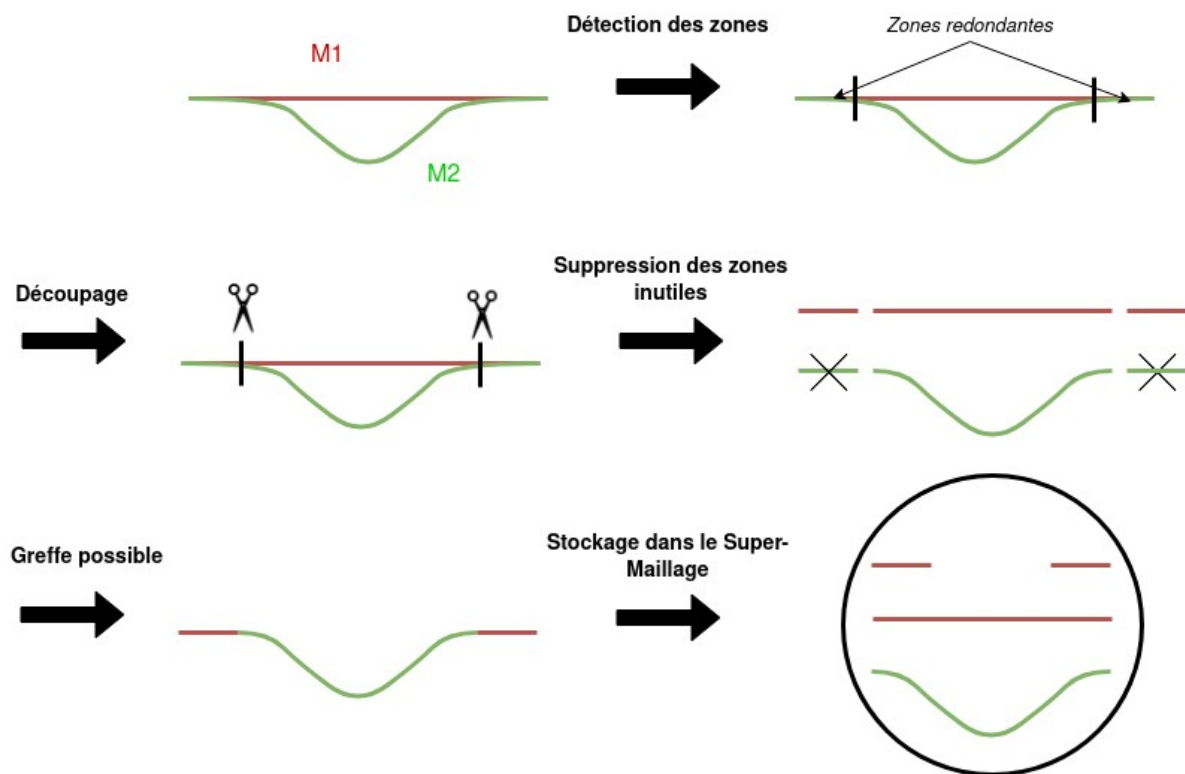


Figure 8: étapes de construction du Super-Maillage

Nous avons vu précédemment un cas théorique, cependant, les maillages réels ne sont pas parfaitement superposables. Ainsi, on déterminera un seuil arbitraire d'une distance à partir de laquelle les parties d'un maillage seront considérées comme identiques.

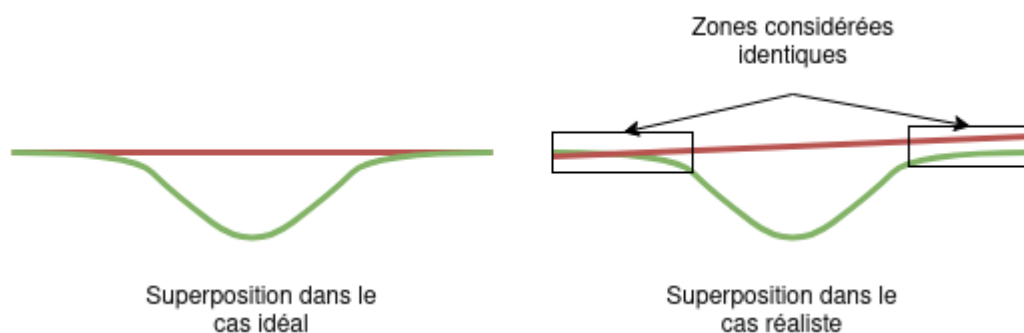


Figure 9: comparaison de superposition de maillages idéaux / réalistes

L'étape dite de greffe dans la figure 8 sera donc complexifiée. Comme dessiné sur la figure ci-dessus, les maillages ne sont pas parfaitement superposables. Lors de la découpe, représenté ci-dessous, la greffe ne se fait automatiquement comme pour un cas idéal. Il faut donc créer une liaison adaptée pour permettre des reconstructions entre maillages. In fine, notre Super-Maillage contiendra tant les maillages découpés ainsi que les zones de liaisons (représentées en orange sur la figure).

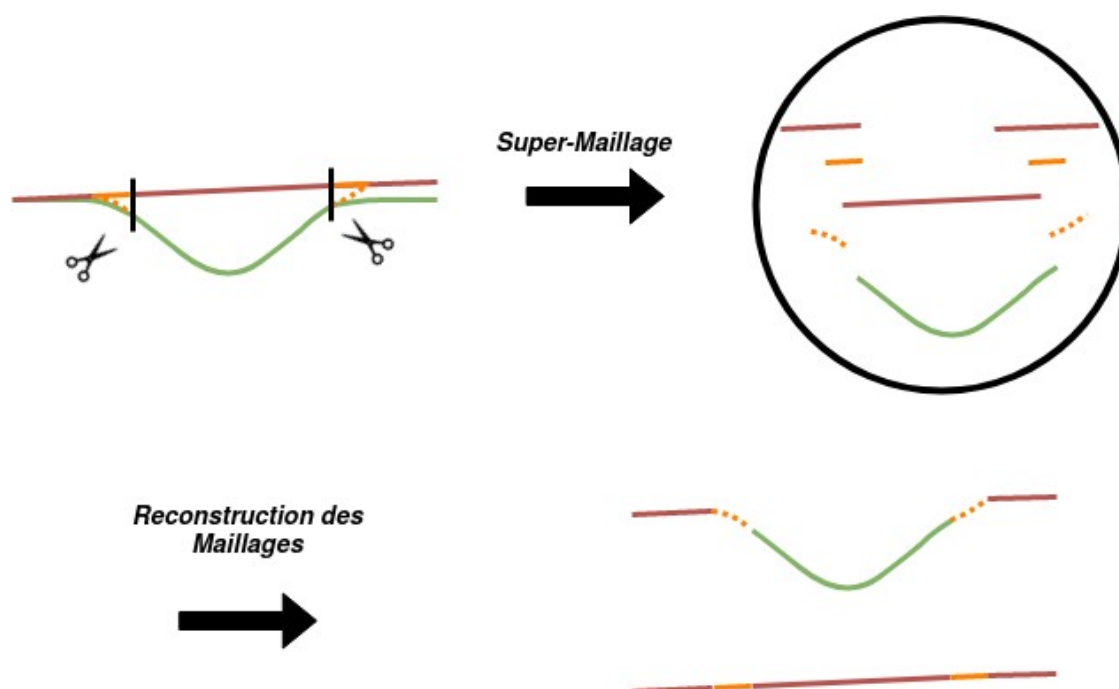


Figure 10: création du Super-Maillage dans le cas réaliste

Pour pouvoir réaliser ce projet, nous avons recherché les algorithmes de traitements adaptés existants afin de déterminer lesquels nous utiliserons.

## 2) Recherches des algorithmes

→ Méthode de détection :

Pour faire une détection de la redondance entre deux maillages, on peut utiliser des algorithmes de seuillage de distance de points. Le seuillage consiste à sélectionner les points qui nous intéressent grâce à un seuil fixé arbitrairement. Ce seuil représente la distance euclidienne maximale entre deux points pour que ces points soient considérés comme proches. Prenons la figure ci-dessous, on compare la distance entre les points rouges et les verts. Grâce au seuil établi, on sépare les points en deux ensembles : un ensemble de points proches et un ensemble de points distants du premier maillage par rapport au deuxième maillage. La transition entre l'ensemble proche et l'ensemble distant constitue la zone de découpage. On compare donc dans un premier temps les points du maillage 2 par rapport à ceux du maillage 1, puis vice-versa.

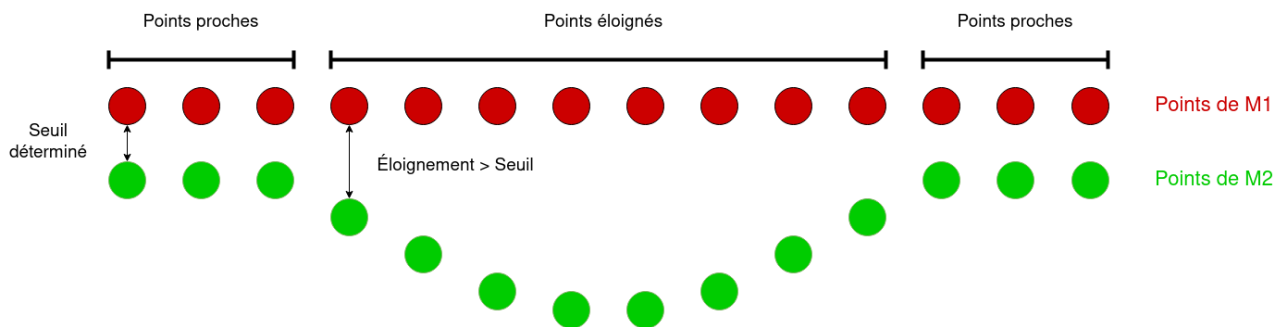


Figure 11: Seuillage de points (modif éloigné en distant)

Il existe plusieurs moyens d'effectuer un seuillage de distance de points. La manière la plus intuitive est d'utiliser un algorithme glouton. Cet algorithme compare la distance entre chaque point du maillage M1 par rapport à tout les points du maillage M2. Pour chaque point de M1, si la distance point à point par rapport à M2 est supérieure au seuil alors ce point est stocké dans une liste de points distants. Dans le cas contraire, il est stocké dans la liste de points proches. Toutefois, cette méthode nécessite beaucoup de temps de calcul ce qui n'est pas favorable à notre jeu de données composé de plusieurs millions de points. Nous allons donc utiliser un *kd-tree* pour gagner en rapidité. Un *kd-tree* est une structure de données de partition de l'espace permettant de stocker des points et de faire des recherches de plus proche voisin [4].

### complexité et partitionnement

Ceci permettra d'obtenir les points les plus proches dans un temps logarithmique, fonction du nombre de points.

→ Méthode de découpage :

Chaque maillage représentés est composés de triangles (cf figure 11). Ces triangles sont composés de sommets connectés par les arêtes. Le découpage a lieu au niveau de ces points et arêtes. Toutefois, lors du découpage, il est nécessaire de dupliquer ces points et arêtes afin de séparer le maillage en deux. Cette séparation permettra une reconstruction éventuelle.

Dans la figure ci-dessous, par lecture de gauche à droite, on détermine les points limites (noirs). Un point dit limite, ici, est un point proche ayant au moins un voisin considéré distant. C'est au niveau de ces points noirs que la découpe va avoir lieu. Ce découpage est une duplication de ces points et des arêtes les reliant, comme dessiné à droite de la figure. Cette duplication permettra de séparer totalement le maillage en deux sous-maillages.

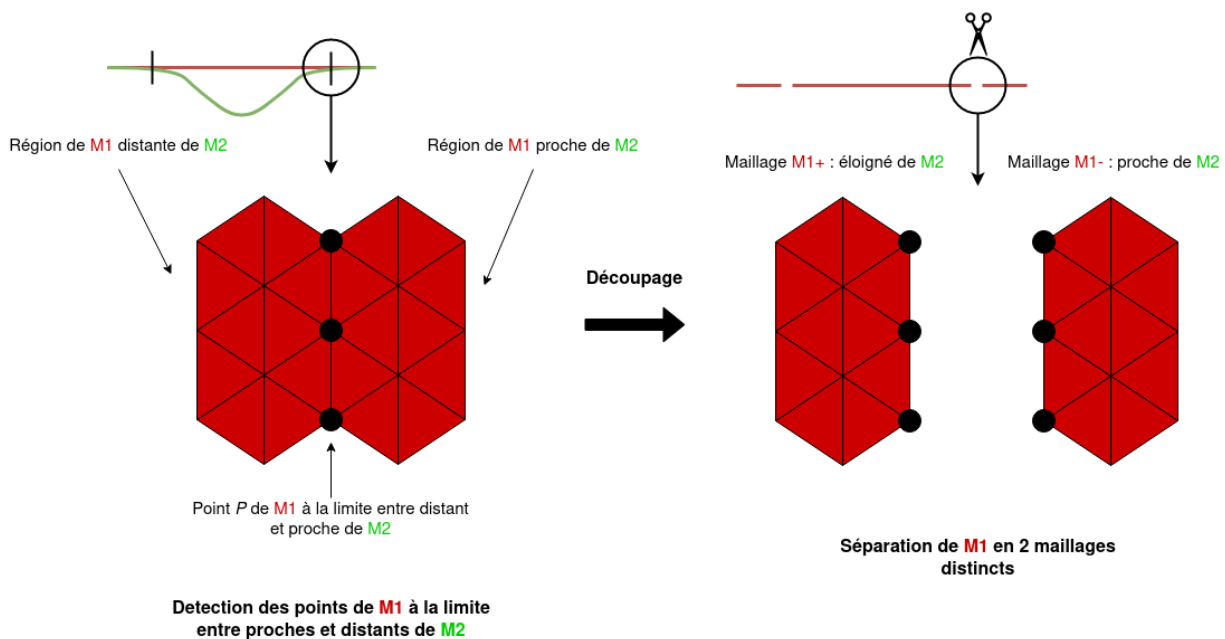


Figure 12: découpage de maillage en sous-maillages

→ Méthode de greffe :

Le découpage étant fait, il faut maintenant greffer les zones d'intérêts. Cette étape est la plus importante de notre travail car elle doit permettre de résoudre les problèmes liés aux décalages entre les deux maillages. Comme abordé précédemment, des liaisons doivent être créées pour permettre une reconstruction cohérente entre sous-maillages.

La méthode utilisée ici est une projection MLS ou « Moving Least Square ». Aussi appelée « méthode des moindres carrés mobiles », elle est utilisée pour créer une surface en trois dimensions à partir d'un nuage de point [5]. L'approximation MLS repose sur trois composants : la fonction de base, la fonction poids associée à chaque nœud et un ensemble de coefficients qui dépendent de la position [6]. Cette méthode consiste en la projection de points sur un ensemble d'autres points qui possèdent des normales. Tout d'abord, on sélectionne les  $k$  plus proches points du point à projeter. Le point est ensuite projeté grâce à une formule mathématique tenant compte de la position, l'orientation des points proches et du nombre d'itérations  $i$ .

Il existe cependant un cas particulier de projection MLS qui permet d'affiner la qualité de la projection d'un point sur un ensemble d'autres points, nommé APSS où « Algebraic Point Set Surface » [7].

Le schéma ci-après illustre le principe théorique de la méthode MLS. Les points rouges représentent les points proches sélectionnés. Le point noir est celui que l'on cherche à projeter sur l'ensemble de points bleus.

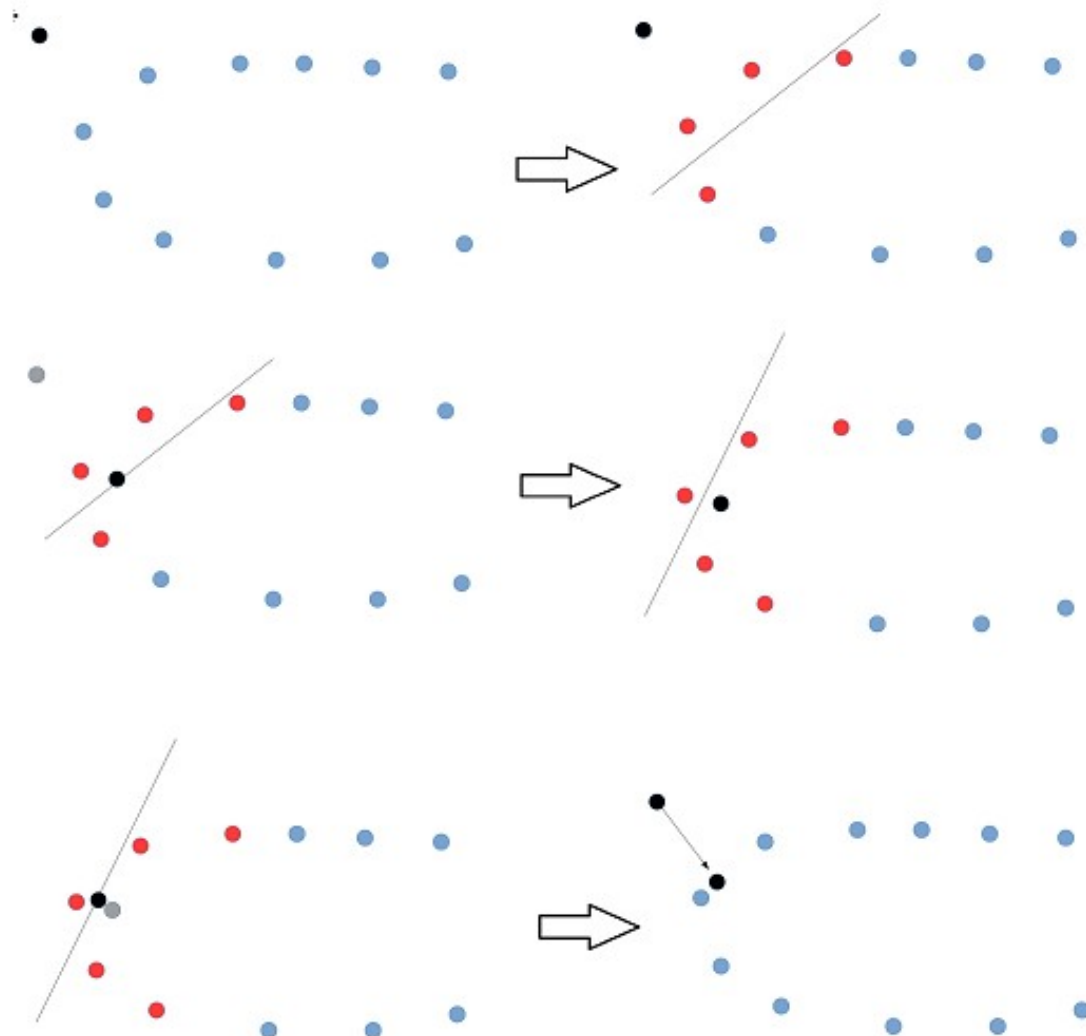


Figure 13: étapes de projections MLS

Le schéma ci-après permet de comprendre l'application de cette méthode de projection sur notre projet. Il se lit de haut en bas. En effet, après le découpage et la suppression des parties communes (premier et deuxième dessin), il faut raccorder les parties souhaitées par le biais de liaisons. Ces liaisons sont réalisées par des projections. A partir de l'étape de greffe, deux options s'offrent à nous afin de réaliser le raccord. Ces deux options sont basées sur la conservation de parties découpées dans l'étape de découpage. L'option 2 (à gauche) permet la projection d'une partie existante sur une partie conservée (pointillés rouges). L'option 1 (à droite), quant à elle, fonctionne de manière contraire à savoir une projection de la partie conservée lors du découpage sur la partie existante (trait continu vert).

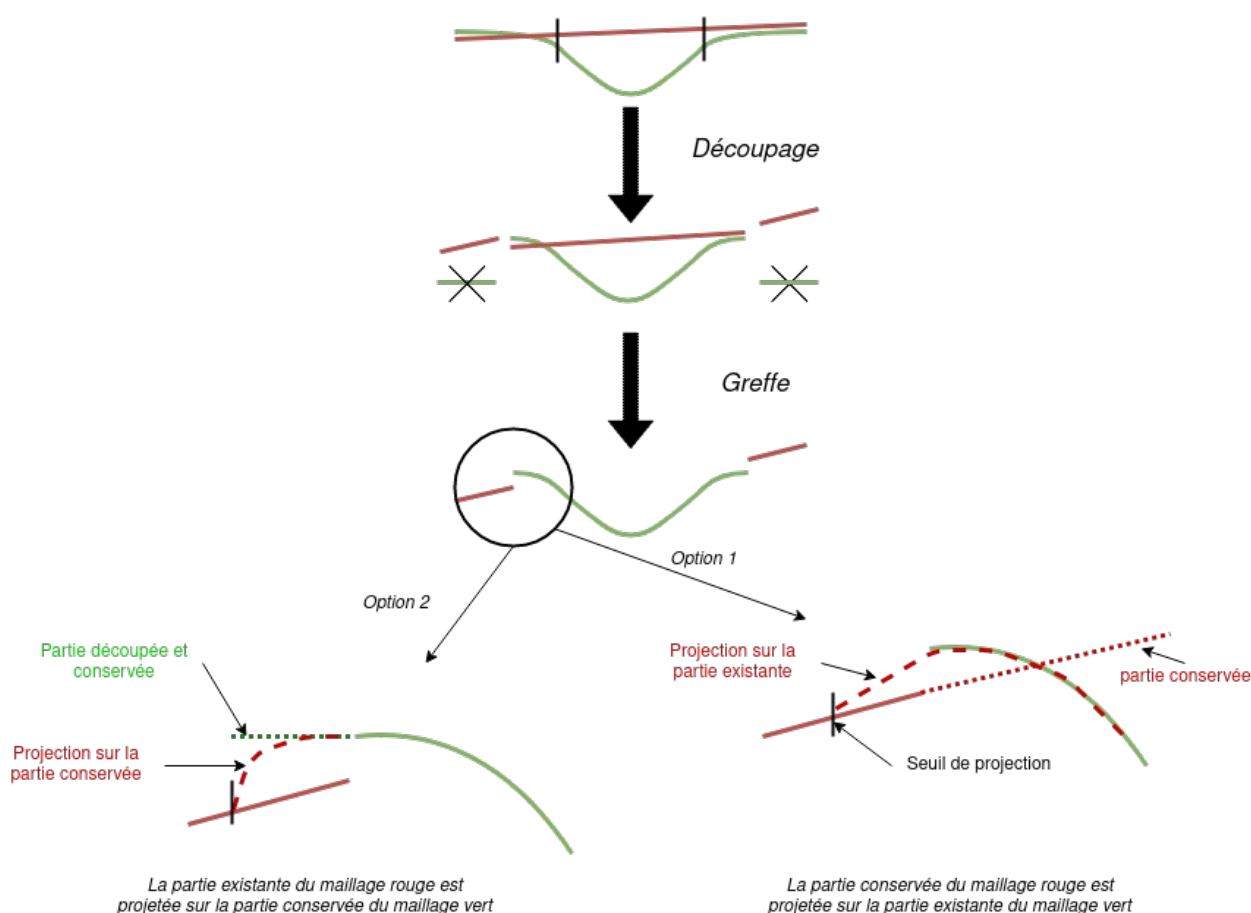


Figure 14: méthode de projection MLS

Cependant, l'option 1 et 2 sont combinables (option 3) permettant d'améliorer la création d'une jonction, de manière plus malléable et surtout adaptable à n'importe quel maillage.



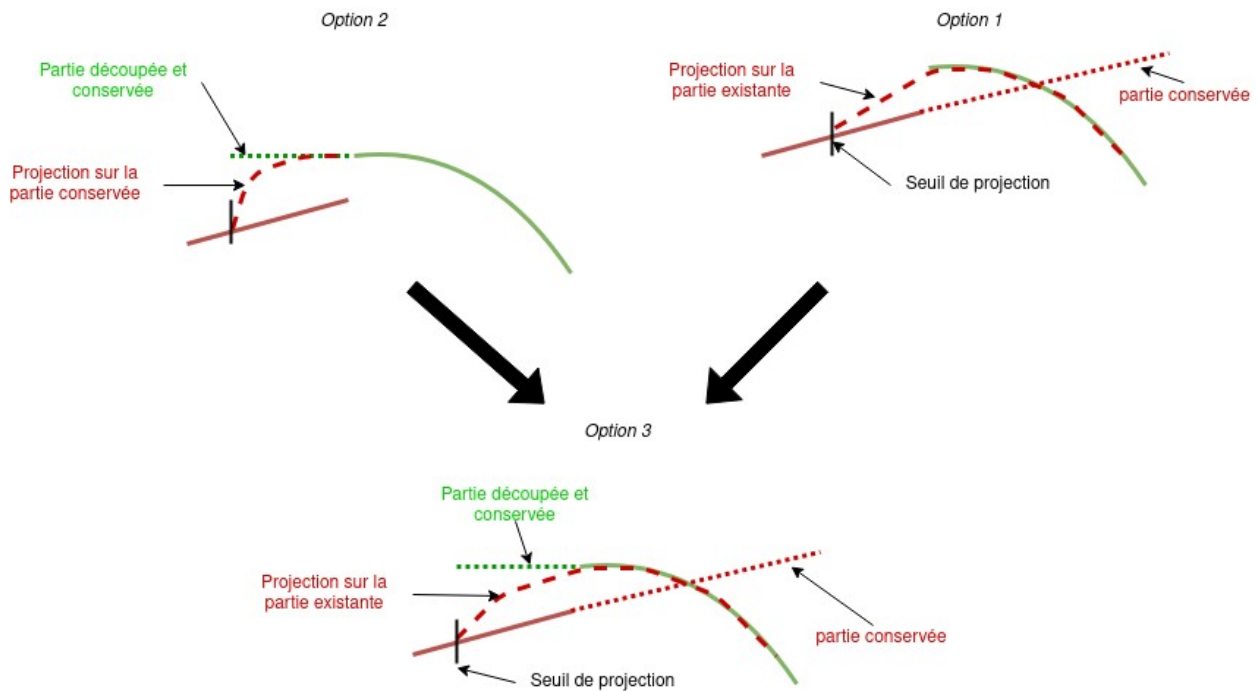


Figure 15: formation de l'option 3

Il est nécessaire de bien comprendre que dans le cas où des décalages sont notés, les parties découpées existantes entre un maillage et un autre ne permettent pas une liaison. C'est pourquoi, une liaison adaptable permettra de relier chaque maillage selon sa topologie (pointillé orange de la figure 14). La création de cette liaison adaptable résulte tant du maillage vert que du maillage rouge. Sur le schéma ci-dessus, on voit bien la projection du maillage vert et celle du maillage rouge, représentés en pointillés. Pour pouvoir utiliser cette liaison pour n'importe quel maillage, il faudra la découper au niveau des jonctions. Le schéma ci-après montre les parties à découper après la création de la liaison adaptable par projection MLS.

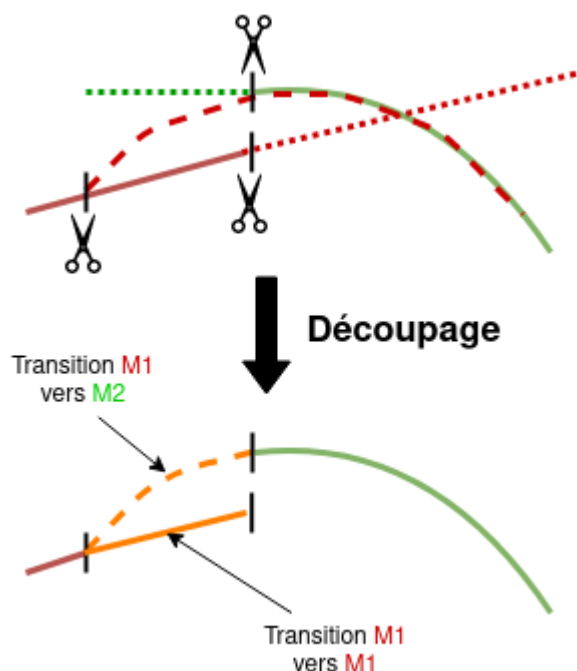
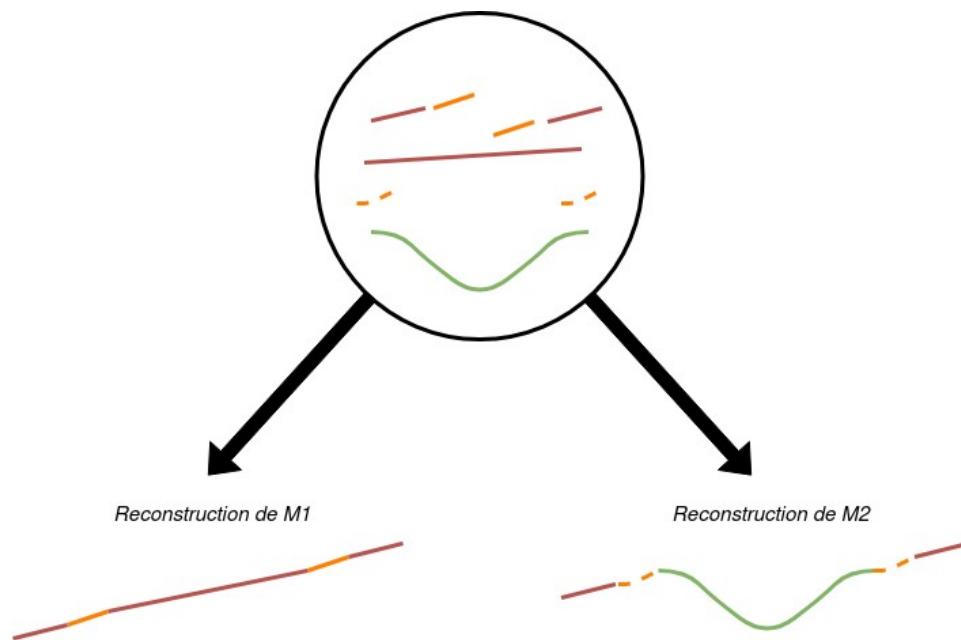


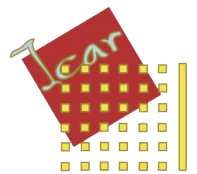
Figure 16: découpage des transitions entre les maillages

Ainsi on obtient dans le Super-Maillage, les parties découpées mais aussi les liaisons adaptables en orange permettant de reformer à souhait n'importe quel maillage.



*Figure 17: Super-Maillage contenant parties et liaisons*

Pour conclure cette partie, nous avons détaillé les méthodes qui seront utilisées pour détecter, découper, greffer et stocker nos maillages.



### 3) Algorithme proposé

Grâce aux méthodes détaillées précédemment, nous avons tout les traitements requis pour élaborer un algorithme de création de Super-Maillage.

(en cours)

Données : (M1, M2, S), respectivement maillage 1, maillage 2 et un seuil

Résultats : (Ri, SM), respectivement reconstruction du maillage i, super-maillage

Algorithme : decoupe\_seuil ( M1, M2, S )  $\rightarrow$  (M1-, M1+)

    seuillage\_points\_proche (M1, M2, S)  $\rightarrow$  Z1 : zones de M1 proches de M2

    points\_limites (Z1)  $\rightarrow$  L1 : points limites entre éloignés et proches de M2

    decoupe (M1, L1)  $\rightarrow$  (M1-, M1+) : divise M1 en deux maillages

Algorithme : projection (M1, M2)  $\rightarrow$  (M1~)

Algorithme : decoupe\_sans\_redondance (M1, M2, S, Epsilon)  $\rightarrow$  (R1, R2)

    decoupe\_seuil (M1, M2, S)  $\rightarrow$  (M1-, M1+)

    decoupe\_seuil (M2, M1, S)  $\rightarrow$  (M2-, M2+)

    decoupe\_seuil(M1-, M2-, S - Epsilon)  $\rightarrow$  (M1--, M1-+)

    decoupe\_seuil(M2-, M1-, S - Epsilon)  $\rightarrow$  (M2--, M2-+)

    projection (M1-+, M2-+)  $\rightarrow$  M1-~

    R1 = (M1--, M1-+, M1+)

    R2 = (M1--, M1-~, M2+)

Algorithme : creation\_super\_maillage (S, Epsilon, M1, M2, ... Mn)  $\rightarrow$  SM

    M\_courant = M1

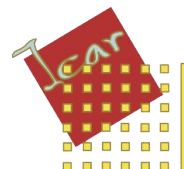
    SM = {}

    pour tout Maillages M\_suivant dans (M2, ... Mn) faire

        (Ri, Ri+1) = decoupe\_sans\_redondance(M\_courant, M\_suivant, S, Epsilon)



**LIRMM**



$M_{\text{courant}} = R_{i+1}$

$SM += R_i$

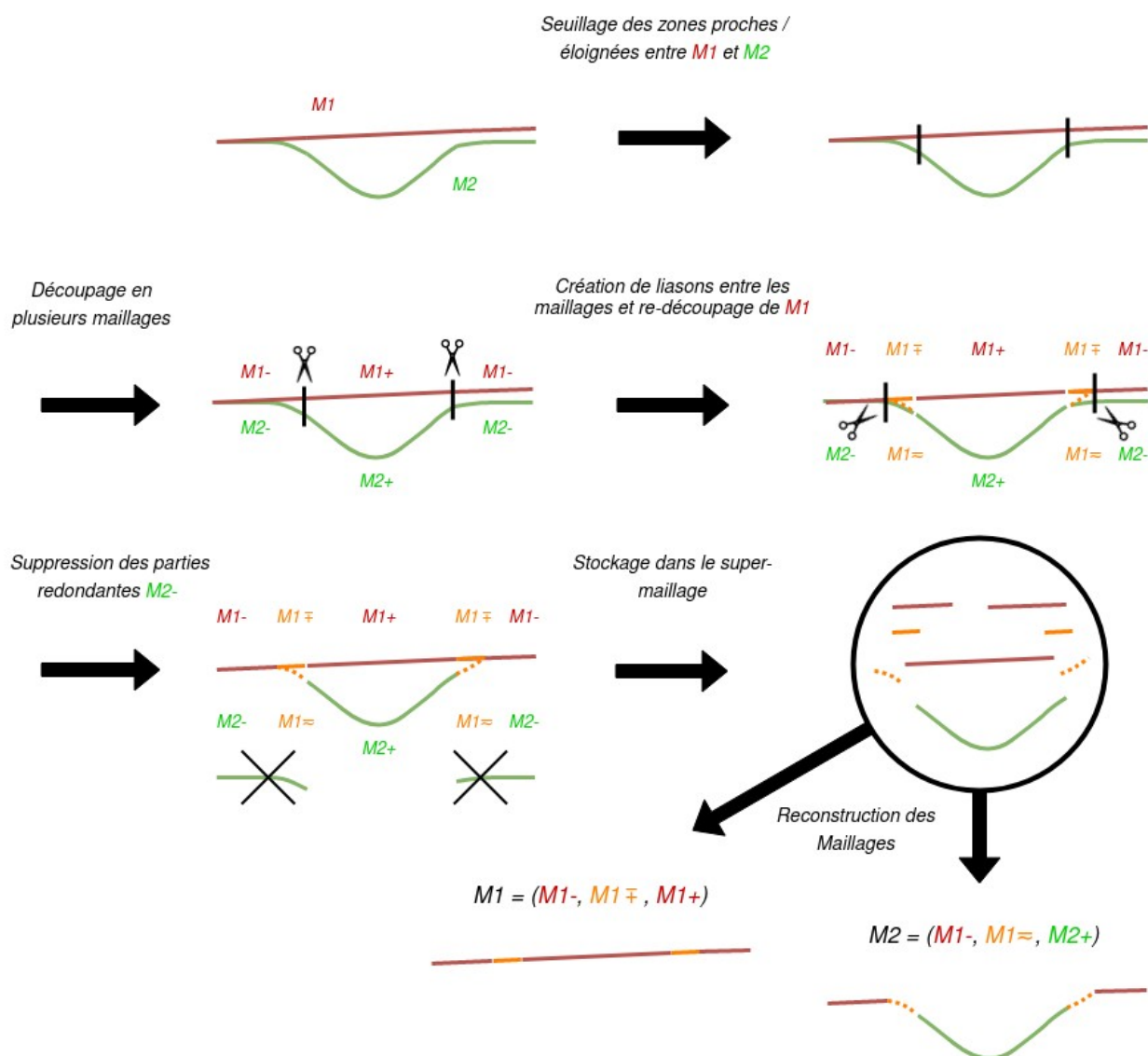


Figure 18: étape de l'algorithme de création du Super-Maillage

## II. Développement et Expérimentations

### 1) Sélection et mise en place des technologies

Après analyse du problème et de ses contraintes, il en découle plusieurs fonctionnalités nécessaires à la réalisation de notre projet :

- Les Acquisitions étant volumineuses, leurs **stockages** en mémoire (RAM) doivent être **efficaces** pour qu'elles puissent être traitées. De plus, les traitements appliqués devront être d'une **complexité** au maximum **linéarithmique** pour avoir un algorithme qui se termine vite et ainsi un rythme de développement rapide. **Détaillé avec des calcul complexité ?**
- Les Acquisitions étant enregistrées au format OBJ avec une texture au format JPEG, Il nous faut un mécanisme d'**importation** / **exportation** de maillage (cela permet de convertir ce qui est stocké dans un fichier en une structure qui sera utilisée pour les traitements).
- L'utilisateur doit pouvoir visualiser et interagir avec les maillages en 3D, il nous faudra donc avoir des fonctionnalités qui permettent **l'affichage** et la **manipulation** d'objets **3D**.
- Le problème étant complexe, on préférera réaliser nos expérimentations sur des **maillages simples** et peu volumineux dans un premier temps. Cela devrait faciliter le développement du projet.

Pour le stockage en mémoire (RAM) et les traitements géométriques, notre choix s'est porté sur la bibliothèque **CGAL**. Elle dispose d'un grand nombre de traitements géométriques très génériques qui peuvent être applicables à leurs implémentations d'une structure de Maillage en demi-arêtes très efficace pour l'application de traitements **justifié ?** nommé Surface\_mesh.

Pour l'importation et l'exportation de maillages texturés notre choix s'est porté sur la bibliothèque **ASSIMP**. Elle est capable d'importer / exporter des objets géométriques depuis un grand nombre de formats de fichiers (OBJ, PLY, OFF, ...).

Pour ce qui est de l'affichage, nous avons choisis d'utiliser la bibliothèque **Qt**. Elle a l'avantage d'être utilisable sur beaucoup de systèmes d'exploitations et de supporter un grand nombre de fonctionnalités (fenêtres imbriquées, entrées tactiles, ...).

Nous avons également choisis la bibliothèque **LibQGLViewer**. Cette dernière permettant de faciliter la visualisation et la manipulation d'objets 3D à partir de Qt et d'OpenGL grâce à un système de scène et caméra 3D déplaçable par l'utilisateur.

Enfin, nous utiliserons le logiciel **Blender** du fait de sa simplicité d'utilisation pour générer des maillages simples qui nous permettront de tester nos traitements géométriques.

Pour assurer un suivi et un versionnage du code, un répertoire **Github** à été créé. Il est accessible à tous à l'adresse suivante <https://github.com/todorico/surgery-viewer>. En cas de problèmes lors du développement ou de perte de données, il sera ainsi possible de revenir à une version du code sauvegardée sur ce répertoire.

Dans l'optique de pouvoir compiler le code facilement et sur le plus de plateformes possible le projet utilisera le système de construction logicielle **CMake**. En suivant les indications détaillées dans le README du répertoire, il est possible de compiler facilement les sources sur n'importe quelle plateforme la reprise et la modification du code pour un autre développeur sera ainsi facilitée.

Une fois les technologies mises en places, un viewer (programme de visualisation de maillages) basique à été développé. Il permet de visualiser un nombre générique de maillages à partir de plusieurs fichiers, de contrôler la caméra à l'aide de la souris, d'activer ou non les éléments à afficher des maillages (points, arêtes, faces). Durant le projet, ce programme a servi de base pour vérifier de manière visuelle que nos traitements géométriques fonctionnaient.

La figure ci-dessous propose un exemple de visualisation de maillages avec le viewer qui sera utilisé.



*Figure 19: visualisation de maillages formant la phrase « hello world »*

## 2) Implémentation et expérimentations

Pour pouvoir expérimenter sans avoir à s'encombrer de la complexité des Acquisitions, nous avons généré une série de maillages simples et parfaitement superposables à l'aide de blender.

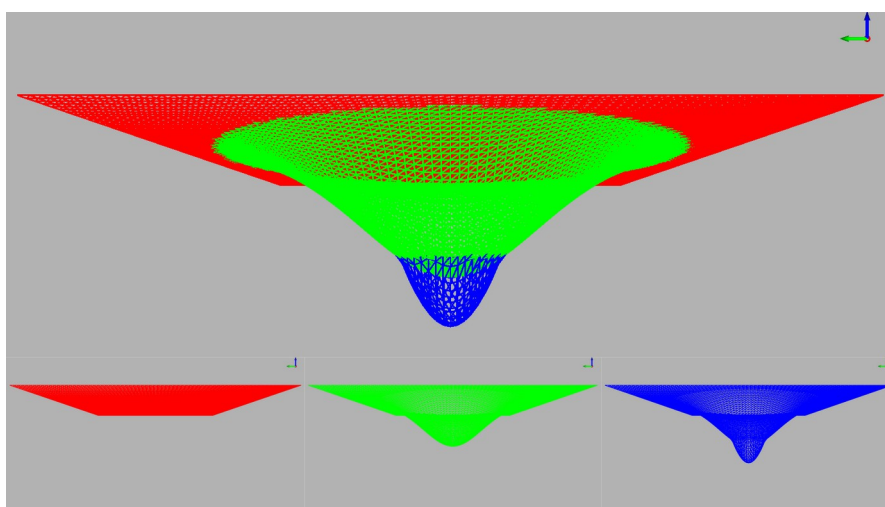


Figure 20: maillages parfaitement superposables

La partie basse de la figure ci-dessus est la visualisation 3D des maillages **M1**, **M2**, **M3** qui sont schématisés dans la figures 6 . La partie haute est une superposition de ces 3 maillages (figure 7). Il est a noter que comme ces maillages s’emboîtent parfaitement, on ne distingue pas avec les couleurs qu’il y a des parties redondantes. Elle sont cependant bien présentes.

Nous voulons également générer des maillages plus représentatifs de la réalité mais avec un volume moins important que les Acquisitions de base. C’est pourquoi nous traiterons également des découpes d’Acquisitions pour faciliter nos expérimentations. Elles sont présentées dans les figures ci-dessous.

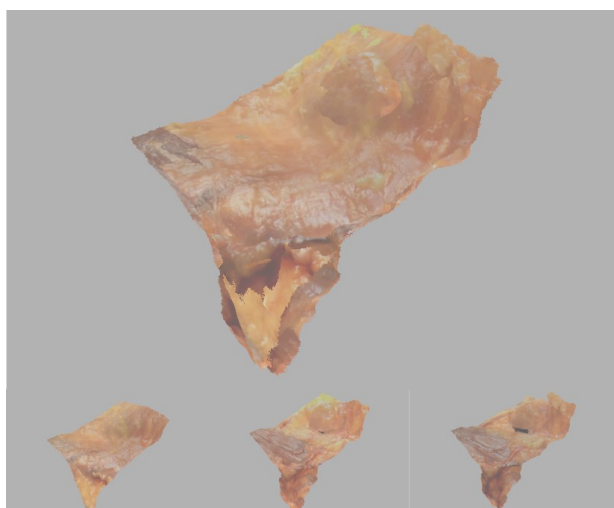


Figure 22: découpage d'acquisitions successives (en bas) avec leurs superposition (en haut)

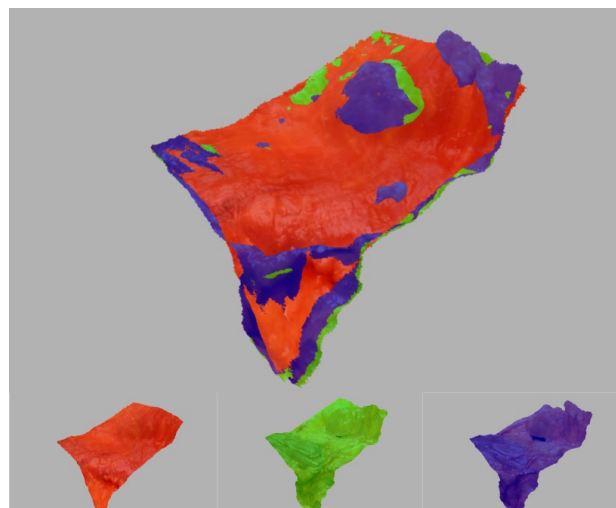


Figure 21: découpage d'acquisitions avec le code couleur



Par la suite nous appellerons **A1**, **A2**, **A3**, les découpages d'acquisitions des figures ci-dessus. On remarque avec leurs superposition que la tâche de reconstruction est bien plus ardu ici que dans le cas idéal.

Nous allons maintenant présenter les résultats des différents traitements nécessaires à notre algorithmes sur nos maillages **M1**, **M2** et nos Acquisitions **A1**, **A2**.

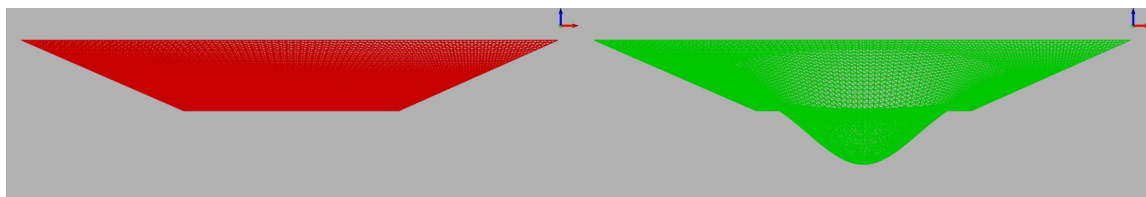


Figure 23: maillages **M1** (à gauche) et **M2** (à droite)

→ Détection de régions proches entre **M1**, **M2**

Les points colorés en violet sur les figures ci-dessous sont les points du deuxième maillage / acquisition qui sont considérés comme proches au premier maillage / acquisition. Les autres points sont les points distants.

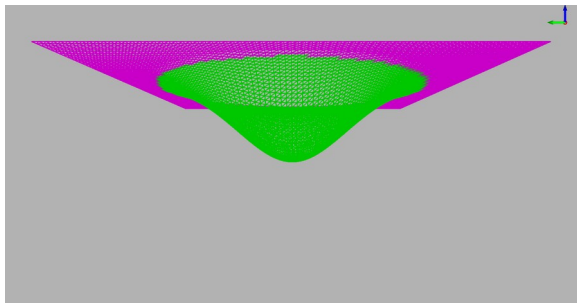


Figure 24: coloration des points de **M2** proches de **M1** avec un seuil de 0.01

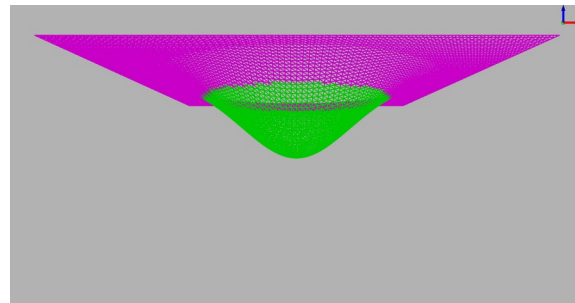
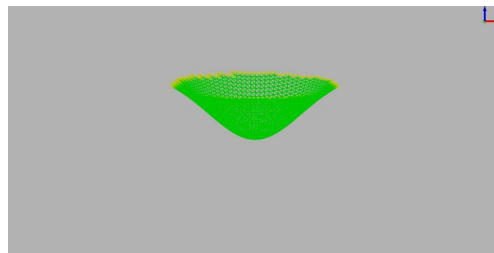
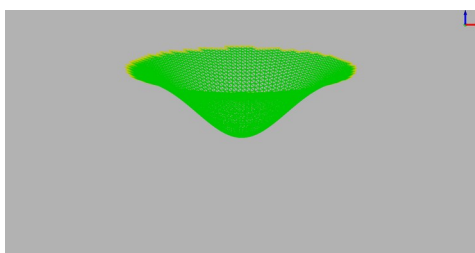


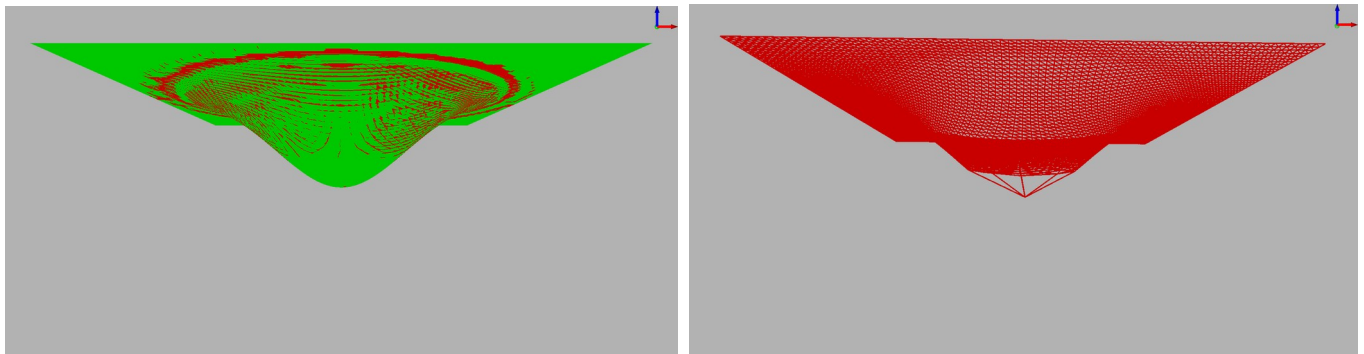
Figure 25: coloration des points de **M2** proches de **M1** avec un seuil de 0.1

→ Détection des limites entre régions proches et distantes puis découpage M2+

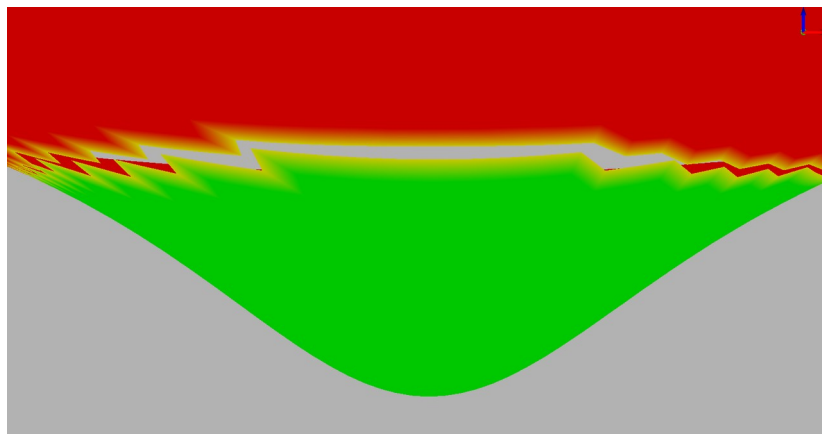
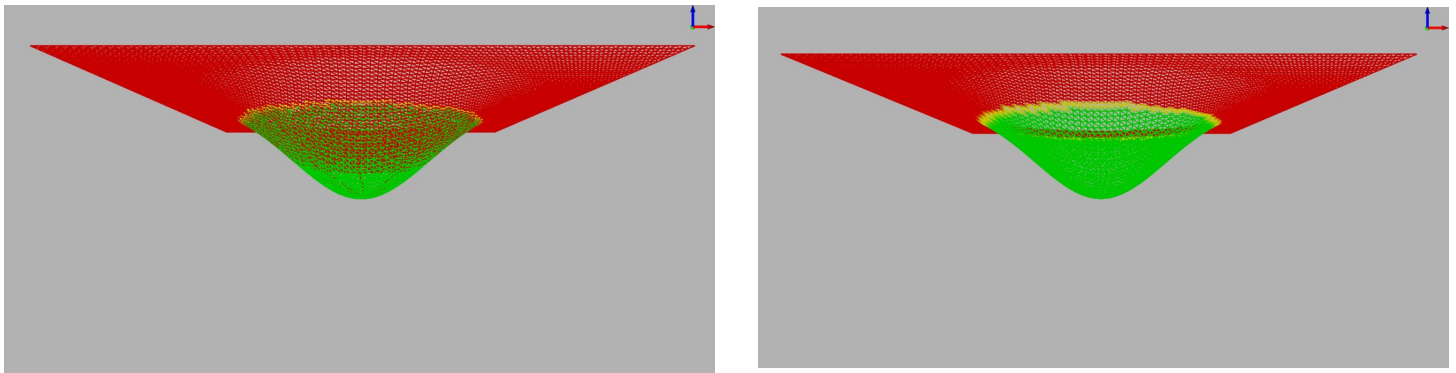


→ Création de liaisons :

→ projection de M1 sur M2 (peu importe le seuil)



→ découpage de la liaison en fonction de M2+ pour créer M1-~



la géométrie de M1-~ doit être adapter pour se rabouter à M2+

→ Détection de régions proches entre **A1**, **A2**

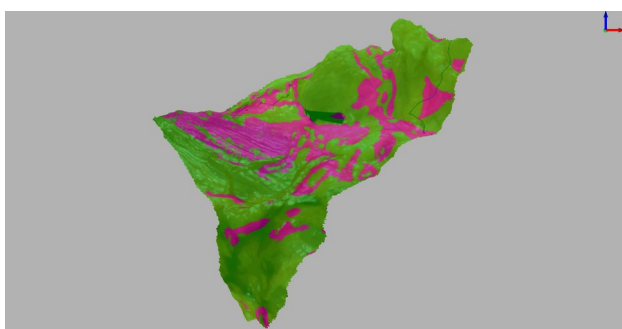


Figure 27: coloration des points de **A2** proches de **A1** avec un seuil de 0.5

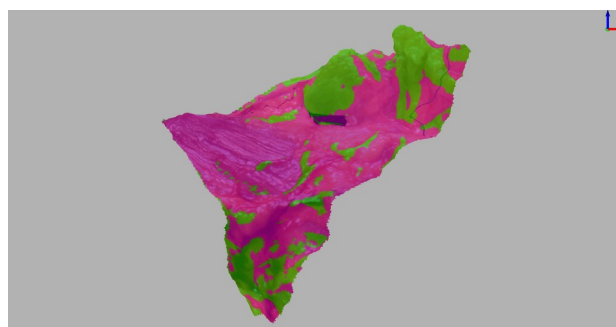


Figure 26: coloration des points de **A2** proches de **A1** avec un seuil de 1

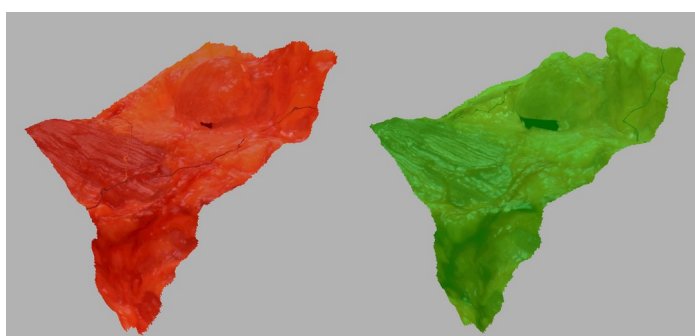
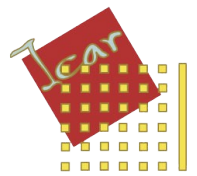


Figure 28: acquisition **A1** (à gauche) et **A2** (à droite)



### III. Résultats

#### 1) Résultats de la méthode

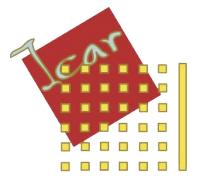
Présentation du produit final ( cas parfait et réels)

utilisation de plusieurs seuil et comparaison des résultats en fonction du seuil et des Acquisitions

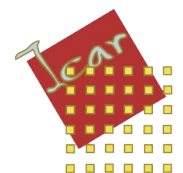
#### 2) Test qualité

comparaison reconstruction avec acquisitions de départs → distances de hausdorf

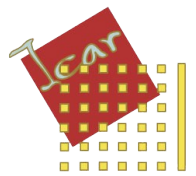
limites de cette méthode



# Conclusion



résumé du travail, résultats obtenus, amélioration et perspective  
apports personnels du stage



# Références bibliographiques

- [1] *Dissection virtuelle 3D dynamique pour l'enseignement de l'anatomie chirurgicale*
- [2] *Marching Through the Visible Man*, Bill Lorensen (GE Imaging & Visualization Laboratory).
- [3] [https://fr.wikipedia.org/wiki/Zygote\\_body](https://fr.wikipedia.org/wiki/Zygote_body)
- [4] *Multidimensional binary search trees used for associative searching*, *Communications of the ACM*, vol. 18, J. L. Bentley,
- [5] *Le pouvoir d'approximation des moindres carrés mobiles*, *Mathématiques du calcul*, volume 67, David Levin
- [6] *Méthode numériques avancées sous matlab 2*, volume 7, Bouchaïb Radi, Abdelkhalak el Hami
- [7] *Algebraic Point Set Surface*, Gael Guennebaud, Markus Gross