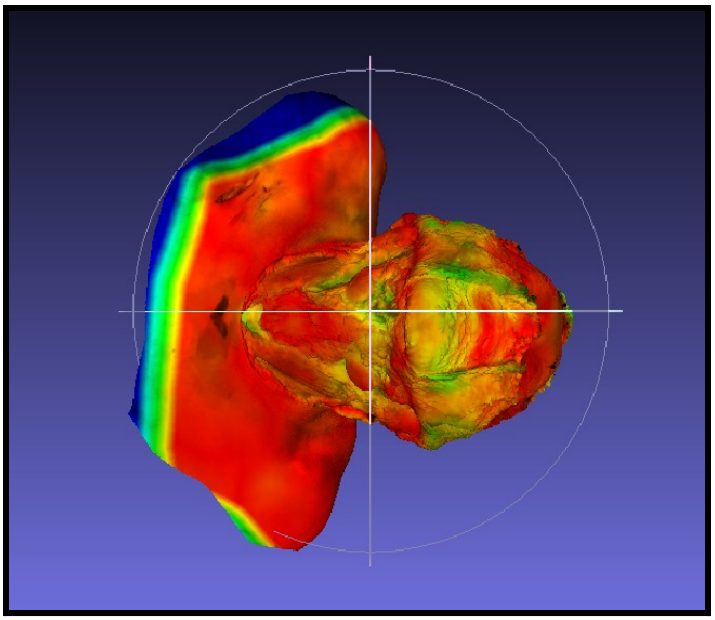
Rapport de stage

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 : **Gérard Subsol** – **Noura Faraj**
Tuteurs universitaire : **Pascal Giorgi**

Pour l'obtention du Master IMAGINA

2020-2021

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’environs 5 mois et demis, Je tiens donc à témoigner toute ma reconnaissance à l’ensemble de l’équipe ICAR ainsi qu’a tout le personnel administratif et technique de cette institution pour l’assistance et l’aide qu’ils m’ont offerts tout au long de ce stage.

Table des matières

Contexte du stage

Depuis quelques années, l'évolution des nouvelles technologies à permis la création d'enseignements 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 premiers gestes techniques chirurgicaux sans travailler sur un vrai patient, notamment via le projet VESALE-3D (Virtual Environment for Surgical and Anatomic Learning Enhancement). L'objectif de ce projet est tout d'abord de répertorier dans une base de données 3D toutes les régions anatomiques essentielles à l'apprentissage des techniques chirurgicales. Ensuite, la visualisation en trois dimensions des étapes successives de la dissection sera permise par le développement d'une méthode virtuelle. Enfin, ce travail permettra de présenter un outil novateur et pédagogique dans l'enseignement de l'anatomie et la chirurgie médicale.

Ce projet à été confié au LIRMM (Laboratoire d'Informatique, de Robotique et Microélectronique dépendant de l'Université de Montpellier) et plus spécifiquement à 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 deux tuteurs d'entreprise, un tuteur universitaire et un médecin enseignant, respectivement :

- M. Subsol Gérard : chercheur au CNRS, il dirige l'équipe ICAR en travaillant sur divers projets notamment sur les traitements des images 2D et 3D.
- Mme. Faraj Noura : enseignante chercheuse spécialisée dans l'informatique graphique à l'université de Montpellier, elle travaille dans l'équipe ICAR.
- M. Giorgi Pascal : 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. Captier Guillaume : chirurgien plastique pédiatrique et professeur d'anatomie exerçant au CHU et à la faculté de médecine à Montpellier, il a réalisé les dissections utilisée lors du développement de ce projet.

Compte tenu de la pandémie du COVID-19, ce stage de cinq mois et demi s'est effectué dans sa majorité en télétravail. En effet, une organisation particulière a dû être mise en place afin de pouvoir communiquer à distance sur l'avancement du projet. C'est pourquoi, des comptes rendus réguliers ont été communiqués et des visioconférences hebdomadaires ont permis de discuter en direct du travail effectué et des tâches à faire.

Introduction

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

Dans l'optique d'obtenir cette visualisation réaliste, des séries de maillages 3D surfaciques texturés à très haute définition ont été acquises au laboratoire d'anatomie de la Faculté de Médecine de Montpellier à l'aide d'un scanner surfacique (Artec Spider) . On appelle maillage un modèle en trois dimensions composé de plusieurs faces généralement triangulaires. Un maillage peut être séparé en sous maillages. C'est donc un vrai cadavre qui a été disséqué et scanné en 3D. Ces maillages correspondent aux différentes étapes de la dissection du cou. Les modèles obtenus sont presque similaires à l'exception des nouvelles parties disséquées à chaque étape. C'est ici qu'intervient le sujet de ce stage qui est le suivant « Traitement de maillages 3D pour la simulation d'une dissection chirurgicale ». En effet, ces maillages nécessitent donc divers traitements géométriques. Suite a plusieurs observations, trois objectifs ont été déterminés :

Tout d'abord, on observe que la majorité de l'information d'une série de maillage correspondant à une dissection est redondante. Afin d'optimiser la mémoire de stockage, on cherche à créer une super structure qui stockerait le minimum d'informations nous permettant de reconstruire chaque étape de dissection. Ainsi il serait possible de jongler d'une étape de dissection à une autre avec le minimum de redondance.

Par la suite, dans le cas où la représentation en trois dimensions est la plus réaliste possible, des évolutions peuvent survenir. En effet, le corps continu à se dégrader ou décomposer (couleur et texture de la peau) mais également, il peut y avoir des modifications apportées par le médecin chargé de la dissection (incisions, déplacements, prélèvements, …). C'est pourquoi chaque changement devra être analysé dans l'optique de le représenter ou non selon son importance.

Pour finir, dans le cas où les deux précédents objectifs seraient parfaitement réalisés, il faudrait adapter l'application à une interface tactile dans le but d'utiliser une tablette pour manier plus facilement le logiciel (utilisation d'un stylet avec pression pour inciser).

Tout au long de ce rapport nous tenterons donc de répondre à la problématique « Comment implémenter une structure de "super maillages" permettant de reconstruire facilement les étapes de

la chirurgie tout en optimisant l'espace mémoire utilisé et en réduisant le bruit d'acquisition ? ». En effet, dans une première partie, nous détaillerons la partie de recherche qui a été nécessaire pour commencer le travail. Puis dans un second temps, nous détaillerons tout le processus de développement de l'application qui a été fait pour amener à une partie exposant les résultats obtenus ainsi que leurs interprétations.

1. Étude

Dans cette première partie, nous allons présenter le problème et les recherches qui ont été nécessaires à la conception de la méthode virtuelle qui sera utilisée tout au long de ce projet.

A - Présentation du problème

Afin de percevoir la réalité du problème, nous travaillerons dans un premier temps sur des maillages 3D parfaitement alignés tels que ci-dessous.

Photo 3D

Cependant pour avoir une bonne appréhension du problème et des objectifs nous allons partir d'un cas schématique simple en 2D. Soit 3 maillages, M1, M2, M3 représentés ci-dessous :

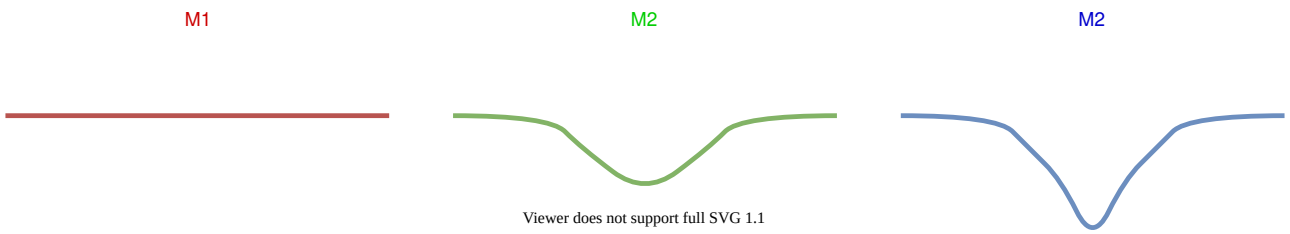


Figure 1: Séries de maillages

En superposant les maillages on peux se rendre compte que des parties sont partagées par la série de maillages, il y a donc de la redondance dans l'information ce qui n'est pas nécessaire.

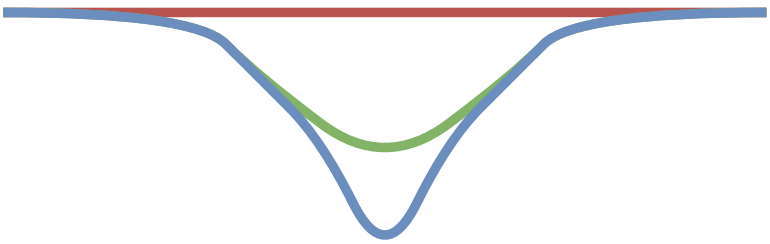


Figure 2: Superposition de la série de maillages

L'idée ici est de parcourir la série de maillages à l'aide d'un algorithme qui fera une comparaison deux à deux des maillages successifs de la manière suivante :

- Détecter les limites entres zones similaires et zones différentes entre 2 maillage ce qui peut être accompli en calculant les distances points à points.
- Découper les deux maillages en sous parties selon les différentes zones, cela nous permettra de supprimer la redondance de l'information.
- Greffer les parties non communes du second maillage sur les parties communes du premier maillage, cela nous permettra d'avoir une reconstruction sans perte de notre second maillage.
- 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, mais cette fois-ci avec un gain de mémoire conséquent.

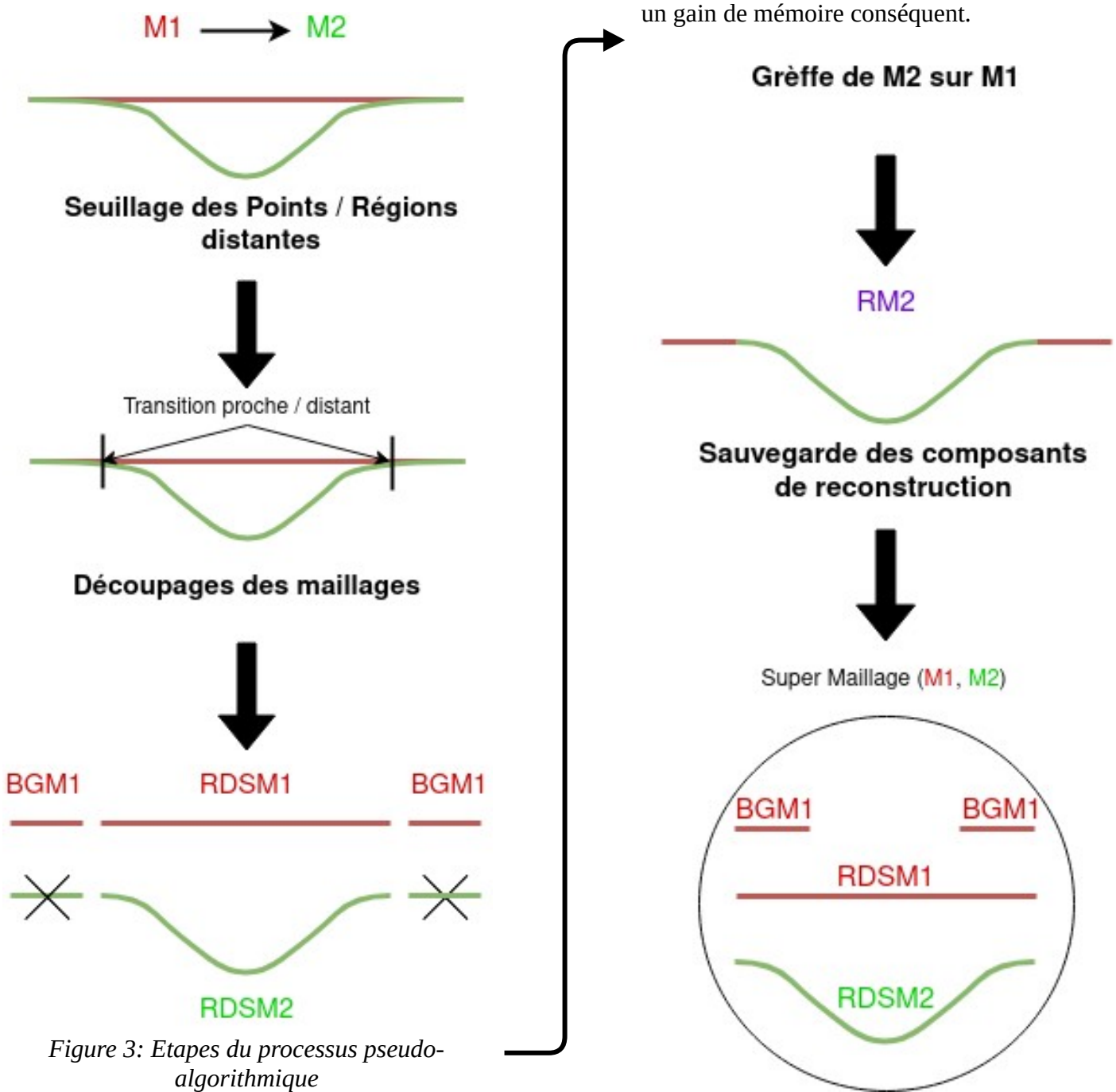


Figure 3: Etapes du processus pseudo-algorithmique

Notre Super Maillage final nous permettra de reconstruire tout les maillages, sans avoir à stocker les parties redondantes, comme indiqué dans le schéma ci-dessous.

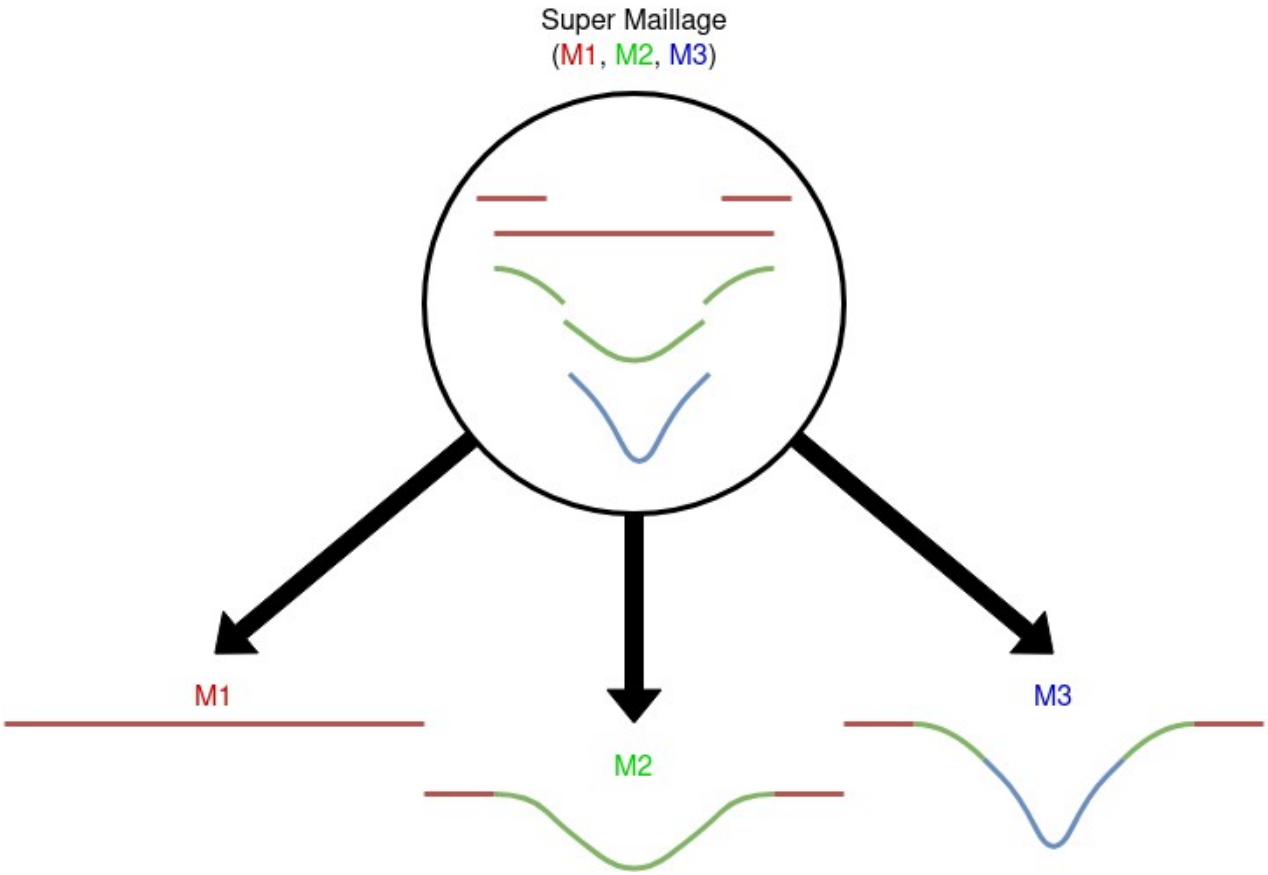


Figure 4: Reconstruction des maillages à partir des parties communes

Au premier abord, le problème pourrait paraître simple. Cependant la tâche se complique avec le jeu de données fournis. En effet, la série de maillages chirurgicaux acquise en 3D est à très haute résolution, elle est donc non seulement très gourmande en mémoire mais également longue à traiter géométriquement. Il faut ainsi utiliser des traitements et outils performants pour pouvoir offrir un algorithme utilisable. De plus, les acquisitions ne pouvant être parfaites, les maillages possèdent de légers décalages sur des parties que l'on souhaiteraient être à la base identiques entre 2 maillages.

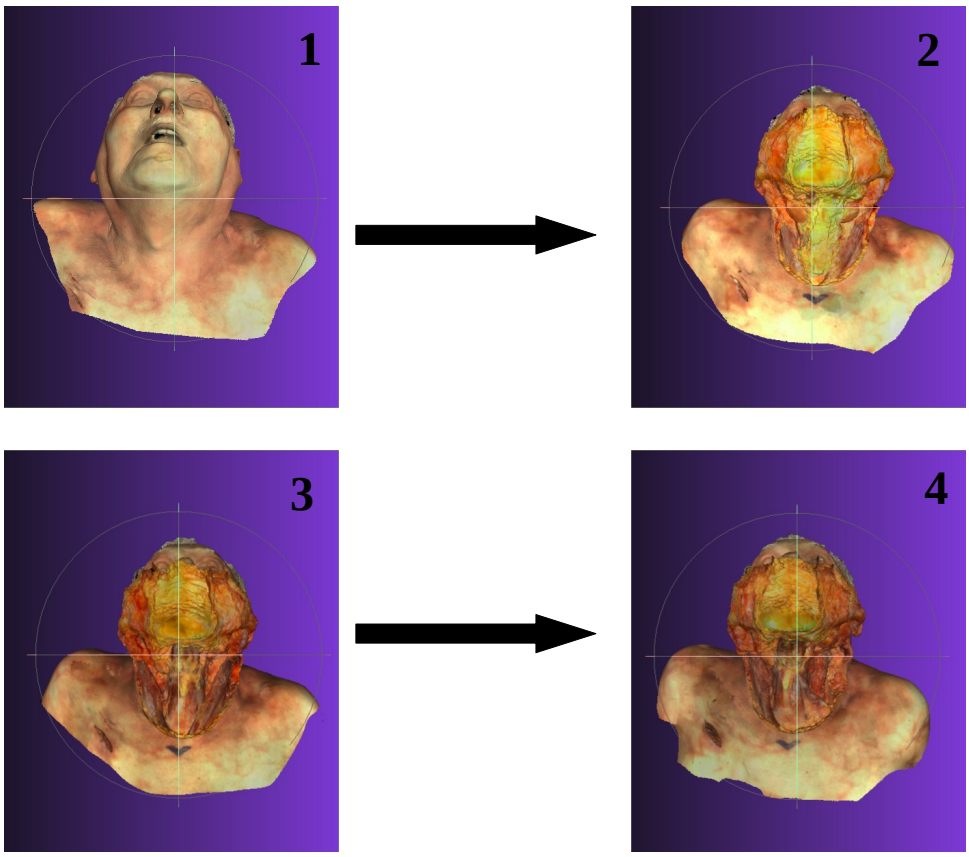


Figure 5: Plans de dissections chirurgicales

L'image ci-dessus représente une des séries de maillages chirurgicaux fournies pour la réalisation du projet. Comme on peut le voir, entre chaque étapes, des chairs ont été prélevées modifiant une petite partie des maillages acquis. C'est ici que notre problématique de redondance intervient de manière concrète. En effet, comme expliqué schématiquement auparavant, on cherche à supprimer l'information redondante des parties inchangées entre chaque acquisition. La figure 5 permet de mieux percevoir les parties communes à chaque acquisition, et celles qui changent.

Il est à noter que contrairement au cas théorique présenter plus haut, les maillages réels ne sont pas parfaitement alignés ou superposés. En effet, comme énoncé dans l'introduction, des décalages surviennent au cours des diverses étapes de la dissection. Ainsi, on déterminera un seuil arbitraire d'une distance à partir de laquelle les parties d'un maillages seront considérées comme identiques, et les décalages seront considérés comme négligeables. C'est pourquoi une attention particulière devra être donnée dans les étapes de découpages et greffes de maillages pour obtenir un résultat visuel convenable.

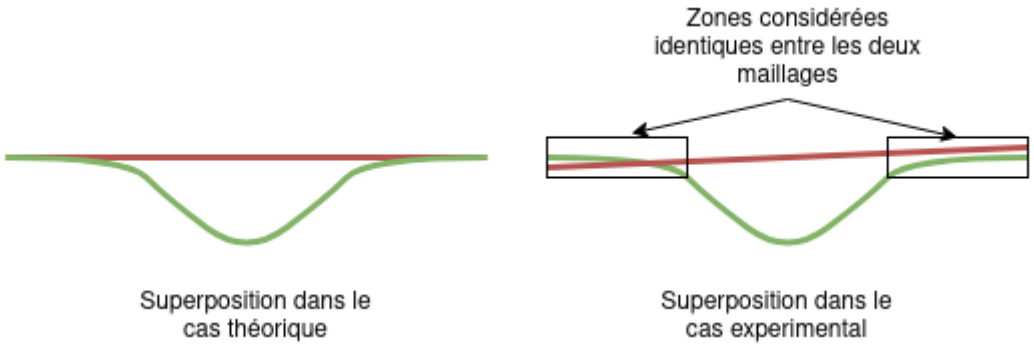


Figure 6: Comparaison de maillages théoriques/expérimentaux

Pour pouvoir effectuer toutes les tâches listées nous allons rechercher toutes les méthodes adaptées existantes afin de déterminer laquelle nous utiliserons.

B - Méthodes envisagées

a) Méthode de détection des zones d'intérêts:

Pour faire une détection de zones d'intérêts entre deux maillages, on peut utiliser des algorithmes de seuillage de points. Le seuillage consiste à filtrer les points qui nous intéressent grâce à un seuil fixé arbitrairement. Ce seuil représente la distance maximale entre deux points pour que ces points soient considérés comme proche. On obtiendra ainsi un ensemble de points proches et un ensemble de points éloignés du premier maillage par rapport au deuxième maillage. On compare donc dans un premier temps les points du maillage 2 par rapport à ceux du maillage 1, puis vice-versa.

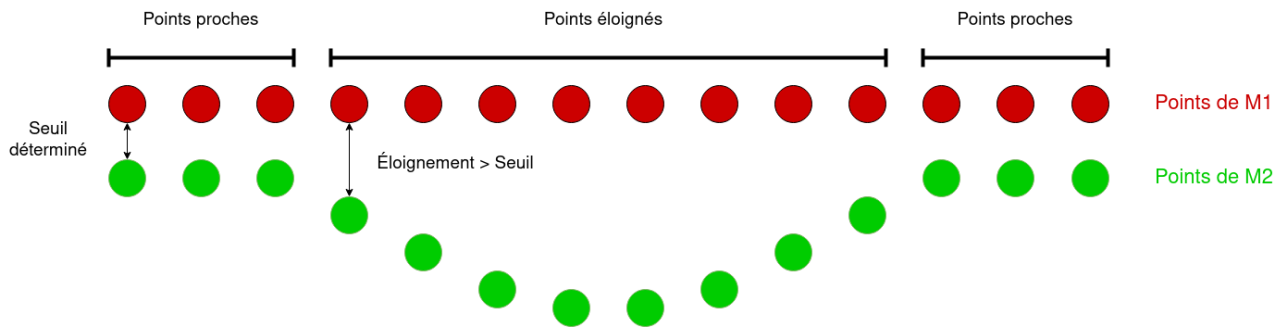


Figure 7: Seuillage de points

Il existe plusieurs moyens d'effectuer un seuillage de points. La manière la plus intuitive est d'utiliser un algorithme glouton. Cet algorithme compare la distance entre chaque point M1 par rapport à tout les points 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 donnés composé de plusieurs millions de points. Ainsi, on favorisera l'utilisation d'une autre méthode, plus rapide, qui permet automatiquement de déterminer quel point est le plus proche. Pour cela, on indexe tout les points du maillage 2 (dans le cas du maillage 2 par rapport au 1) dans un kd-tree qui va trier chaque point en fonction de leurs coordonnées. Ceci permettra d'obtenir le point le plus proche dans un temps logarithmique, fonction du nombre de points.

b) Méthode de découpage des zones d'intérêts:

Après avoir détecté les zones d'intérêt, l'étape suivante et le découpage. Cette étape doit permettre de découper les maillages en tenant compte du fait que ces maillages doivent pouvoir être, in fine, réassemblés entre eux.

Soit la photo ci-après. Le maillage 3D représenté correspond au maillage schématisé en 2D précédemment.

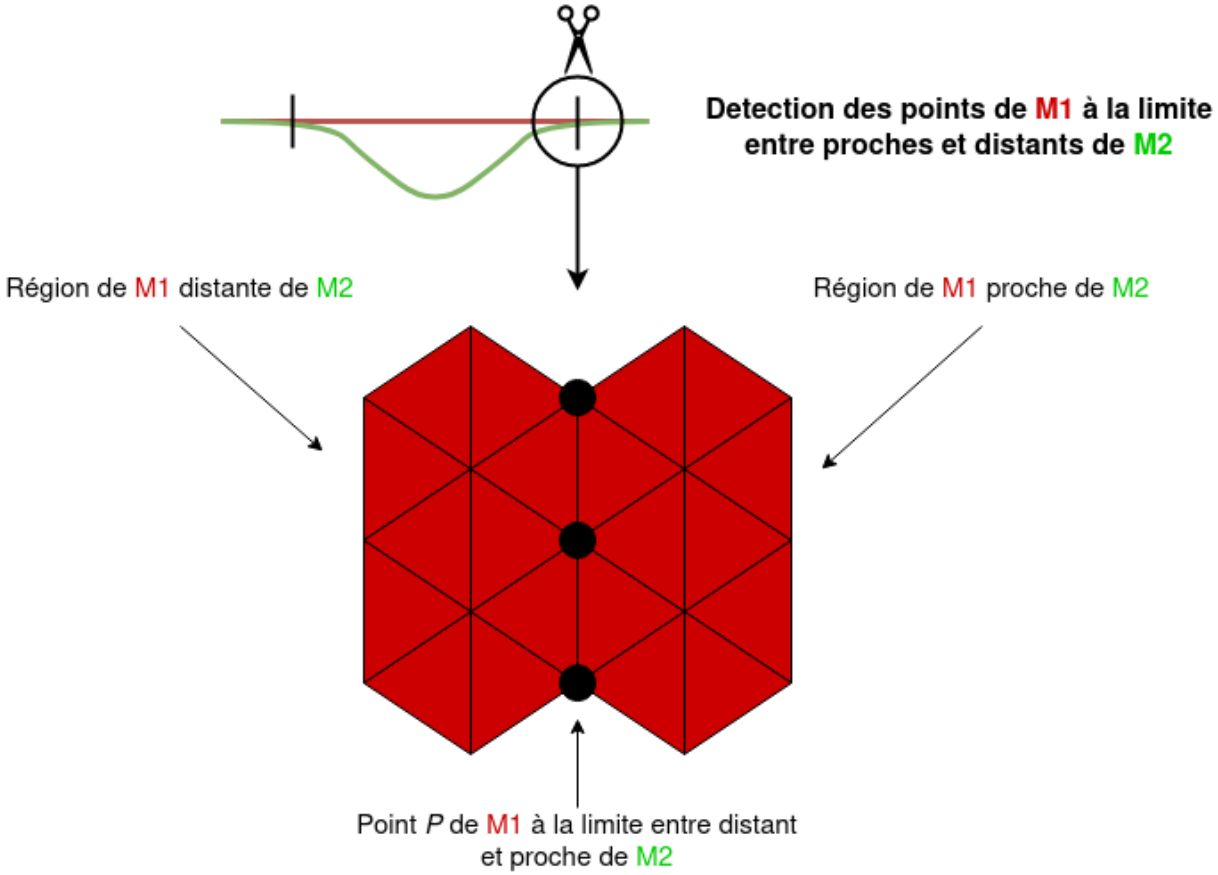


Figure 8: Détection des points à la limite entre proche et distant

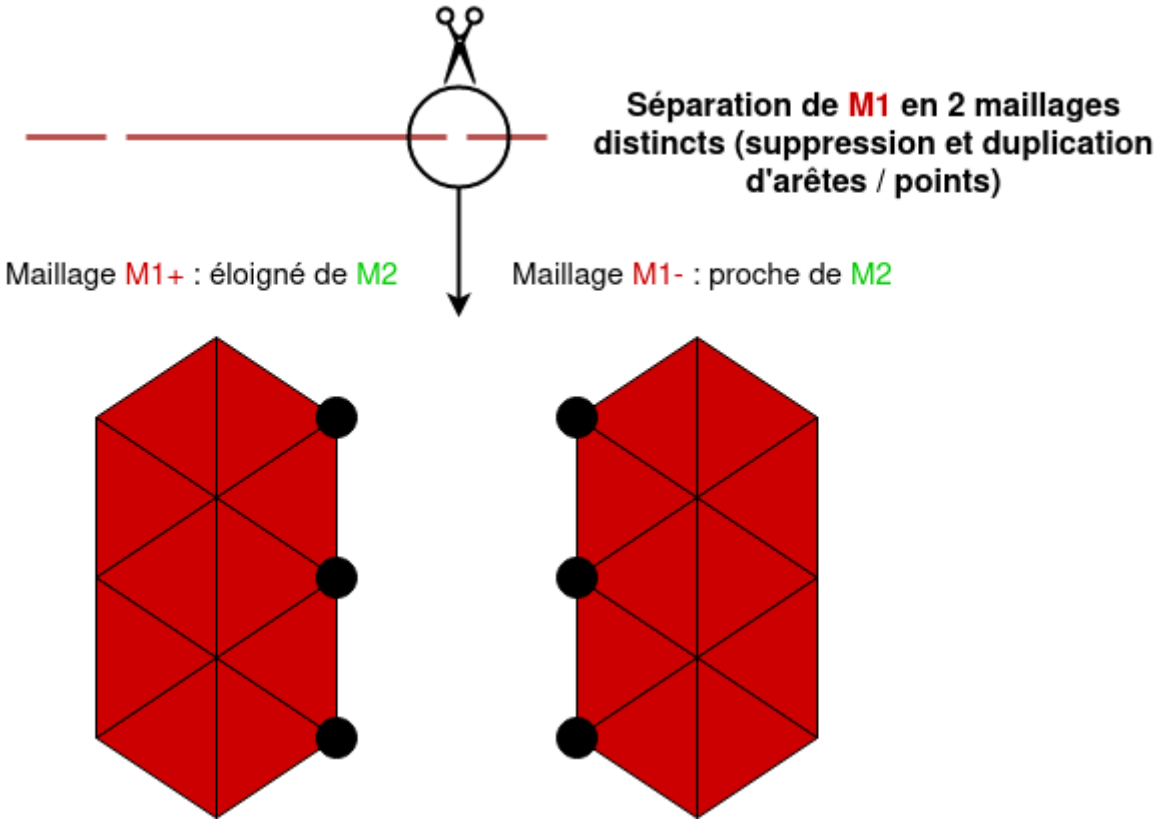


Figure 9: Découpe en dupliquant / supprimant des arêtes

c) Méthode de greffe des zones d'intérêts:

Cette étape est la plus importante de notre travail car cette méthode va permettre de résoudre les problèmes liés aux décalages entre les deux maillages (cf figure 6). Lors de l'étape de découpage, les parties décalées avaient été traitées comme identiques. Toutefois, pour la greffe, on ne peut négliger ces différences. En effet, les parties découpées n'étant pas parfaitement superposables (dû à la réalité des maillages acquis), il faut trouver des méthodes permettant de connecter les différentes structures entre elles en tenant compte de ce défaut de superposition. C'est pour cela qu'il faut créer des zones de jonctions malléables et adaptables permettant de pouvoir associer n'importe quel maillage malgré des décalages éventuels.

La méthode utilisée ici est une projection MLS ou « Moving Least Square ». 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* :

$$\begin{cases} c = \sum_{p_i \in NN(x)} w_i p_i / \sum_{p_i \in NN(x)} w_i \\ Cov = \sum_{p_i \in NN(x)} w_i (p_i - c).(p_i - c)^T \in \mathbb{R}^{3x3} \qquad n = eig_3(Cov) \end{cases}$$

Figure 10: Formule de projection des points

$$x_{k+1} = c - \frac{(x_k - c)^T.n}{||n||} n$$

Figure 11: Formule de projection sur le plan

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 ». Ce cas particulier sera celui que nous utiliserons grâce à la formule suivante :

$$\begin{cases} w_j \leftarrow \frac{w_j}{\sum_i w_i} \qquad \text{(normalisation des poids)} \\ u_4 = \frac{1}{2} \frac{(\sum_i w_i p_i^T.n_i) - (\sum_i w_i p_i)^T.(\sum_j w_j n_j)}{(\sum_i w_i p_i^T.p_i) - (\sum_i w_i p_i)^T.(\sum_j w_j p_j)} \\ \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \sum_i w_i (n_i - 2u_4 p_i) \\ u_0 = -\sum_i w_i ([u_1 u_2 u_3] + u_4 p_i)^T.p_i \end{cases}$$

Figure 12: Formule de projection APSS (cours Mme Noura Faraj)

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.

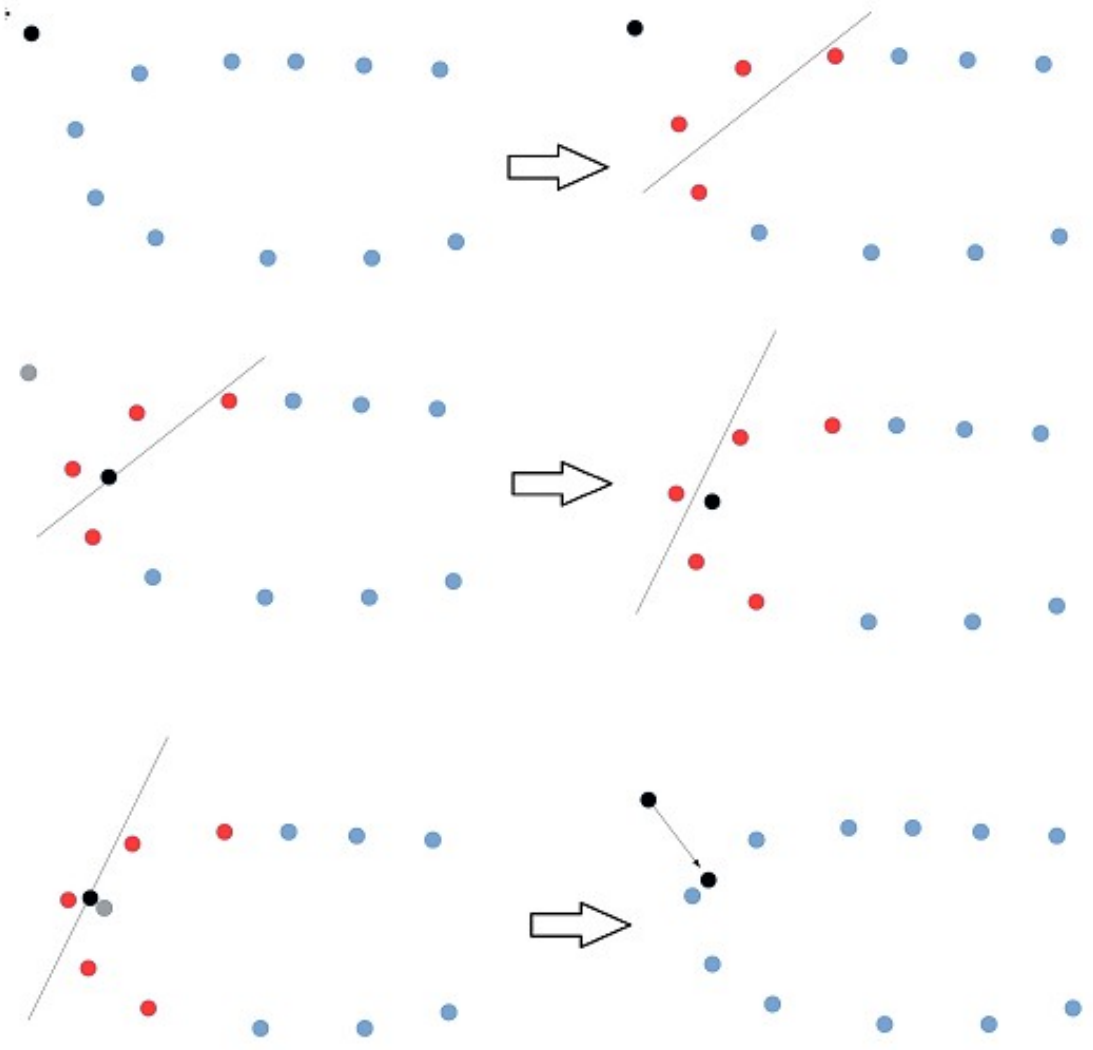


Figure 13: Étapes de projections MLS

Le schéma ci-après permet de comprendre l'application de cette méthode de projection sur notre projet. En effet, après le découpage des parties communes (cf figure 3), il faut raccorder les parties souhaitées par le biais de projections adaptables à n'importe quel maillages. 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'une des options permet la projection d'une partie existante sur une partie conservée (pointillés rouges), il s'agit de l'option 2. L'option 1, 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).

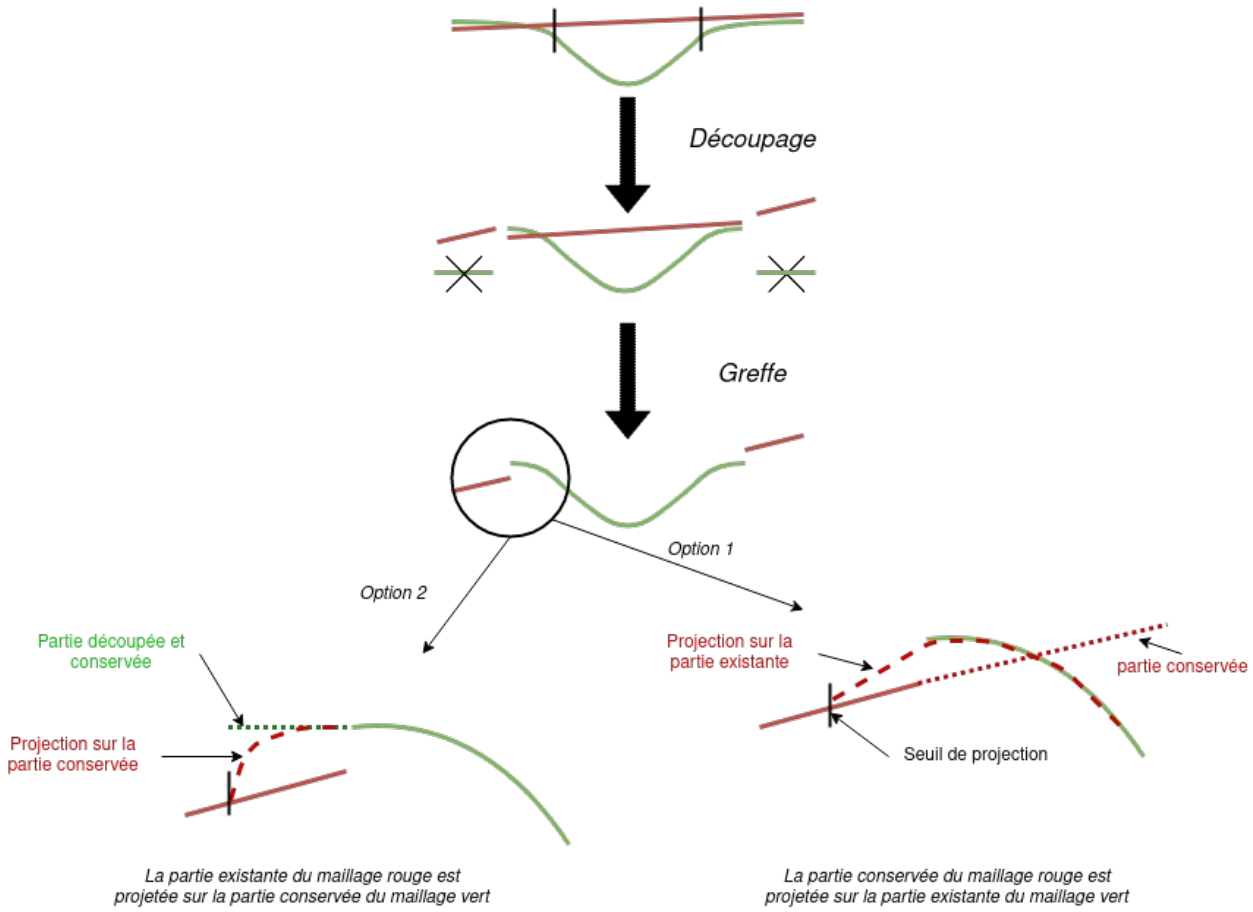


Figure 14: Méthode de projection MLS

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.

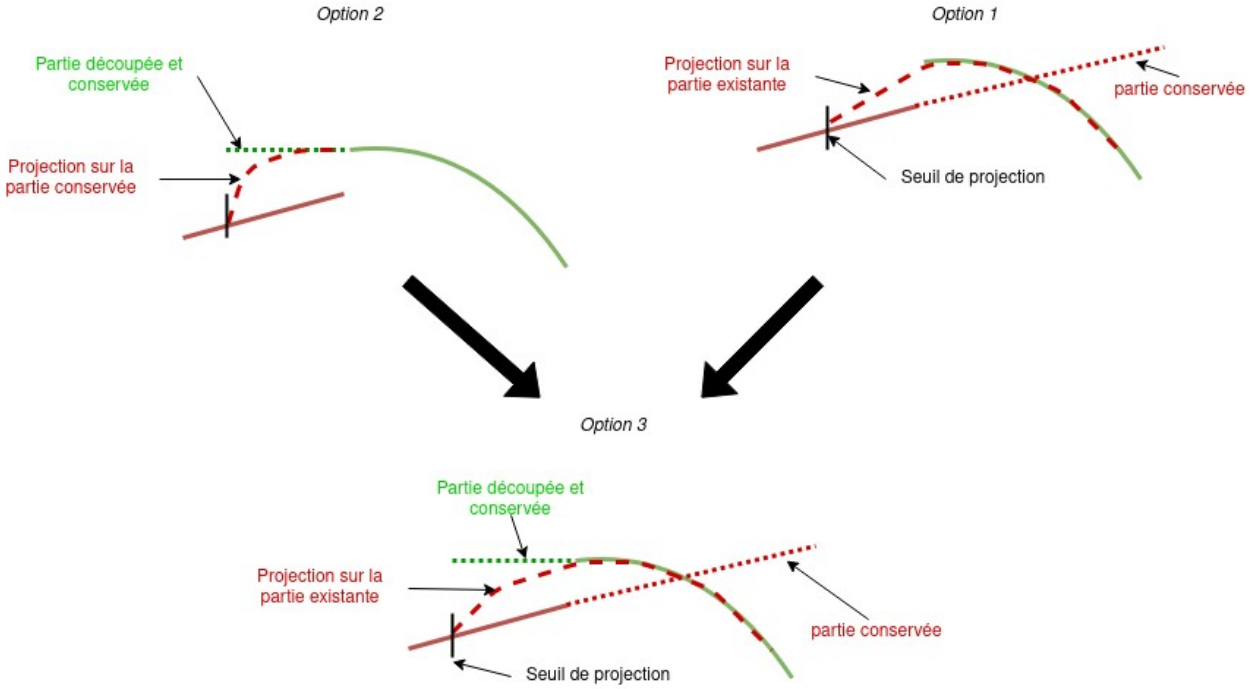


Figure 15: Formation de l'option 3

Il est nécessaire de bien comprendre que dans le cas où des décalages sont notés, les jonctions 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. Pour pouvoir utiliser cette liaison orange pour n'importe quel maillage, il faudra la découper au niveau des jonctions. Ce schéma montre les parties à découper après la création de la liaison adaptable par projection MLS.

Puissance de la projection reglable avec nb iterations, projections.

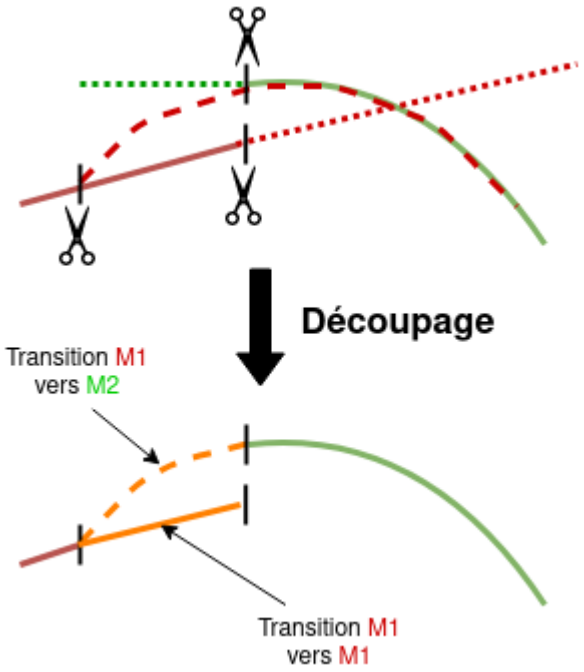


Figure 16: Découpage des transitions entre les maillages

Ainsi on obtient dans le super maillage, les parties découpées mais aussi les liaisons adaptables permettant de reformer à souhait n'importe quel maillage. A noter que les parties de transition projetées (pointillés oranges) seront des nouvelles informations rajoutées aux maillages initiaux.

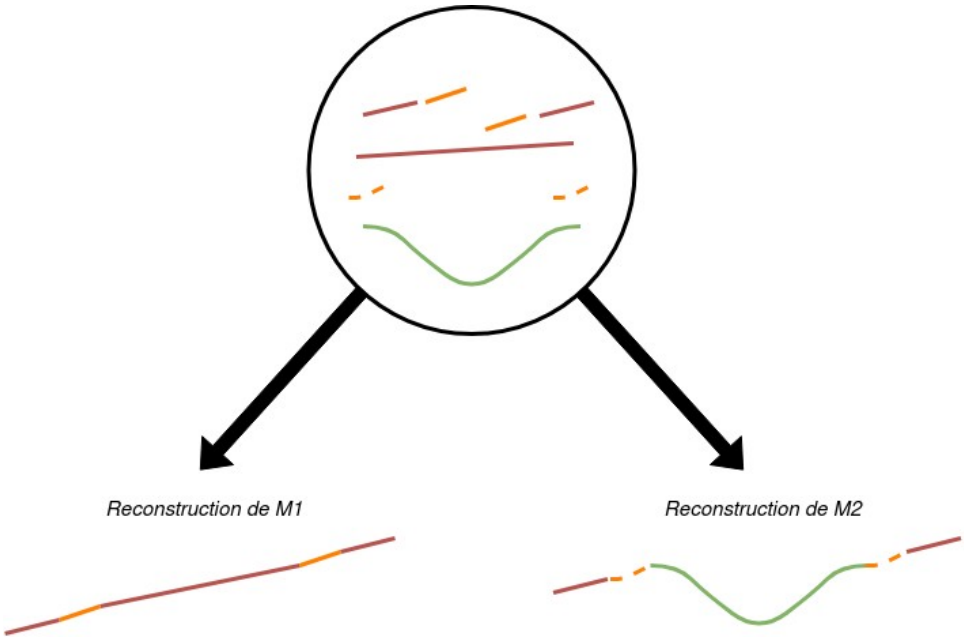


Figure 17: Super maillage contenant parties et liaisons

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

C - Algorithme proposé

1. Sélection des points distants entre M1 et M2 selon un seuil paramétrable

   → Points Distants de M2 appartenant à M1 : **PDM1**

   → Points Distants de M1 appartenant à M2 : **PDM2**

2. Utilisation des points distants comme graines d'un algorithme de « Region Growing » selon un paramètre de propagations des régions

   → Régions Distantes de M2 appartenant à M1 : **RDM1**

   → Régions Distantes de M1 appartenant à M2 : **RDM2**

3. Sélection des régions intéressantes selon un critère de taille (les régions trop petites seront ignorées)

→ Régions Distantes de M2 puis seuillés appartenant à M1 : **RDSM1**

→ Régions Distantes de M1 puis seuillés appartenant à M2 : **RDSM2**

4. Création d'un découpage de M1 à l'aide de **RDSM1** selon un paramètre de zone de transition à conserver (Cela permettra de greffer les régions distantes de M2 sur une base appartenant à M1)

→ Base de Greffe appartenant à M1 : **BGM1**

5. Projection des zones de transition de **BGM1** sur les régions de **RDSM2** correspondantes avec une structure MLS (Cela permettra d'avoir une transition fluide entre les deux)

→ Base de Greffe appartenant à M1 Projeté sur les régions de **RDSM2** : **BGPM1**

→ (Optionnel) Projeté **RDSM2** sur **BGPM1** (meilleur résultats visuel ?)

6. Fusion de **BGPM1** avec **RDSM2** à l'aide d'un lissage laplacien (meilleur résultats visuel ?)

→ Reconstruction de M2 à partir des morceaux similaires de M1 : **RM2**

7. Indexation des composants nécessaire à la reconstruction de M1 et M2 en récoltant **RDSM1**, **RDSM2** ainsi que les éléments produits par la différence de **RM2** avec **RDSM2** et

→ Composants de reconstruction de **M1** et **M2** : **CRM1M2**

D - Sélection des outils

2. Développement et Expérimentations

A - Mise en place des technologies

(Utilisation de Cmake pour pouvoir compiler en multi-plateforme)

(Utilisation d'un répertoire Github avec travis pour partager et tester les build en multi-plateforme)

(Mise en place d'un outils de format de code source clang-format, cmake-format, ...)

(Optimisation de la durée de compilation)

→ Accélère le temps de développement, comparaison avant après

(Création d'un programme test (affiche un modèle 3D))

B- Implémentation

(Création d'un jeux de données simple)

→ Accélère le temps de développement, comparaison avant après

(Implémentation de l'algorithme de Matching avec le visualiseur CGAL)

→ Limites du visualiseur observés + ajout de fairing pour améliorer le rendu

→ Préciser le nouveaux protocole

(Implémentation d'un visualiseur avec Qt5 + LibQGLViewer)

3. Résultats

Algorithme final de Matching

(Comparaison des différentes versions)

(Afficher les distances de hausdorf)

(Cas limites)

(Points d'améliorations)

Présentation du visualiseur

(Fonctionnalités)

(Utilisation)

(Application sur de vrais jeux de données)

Conclusion

Références bibliographique