

# Rapport de stage

Complétion de modèles  
numériques 3D en immersion  
virtuelle

JOMAIN Benjamin

Encadré par :

JAILLET Fabrice

FARRUGIA Jean-Philippe

Octobre 2019

## Remerciements

Avant de présenter mon rapport, je souhaite exprimer mes remerciements à toutes les personnes qui ont pu contribuer au bon déroulement de ce stage.

Je remercie tout d'abord les membres de l'administration de l'INSA qui ont facilité la mise en place du stage que ce soit pour la validation de celui-ci, pour l'édition de la convention de stage et bien d'autres choses.

Je remercie également Maud MARCHAL, professeur à l'INSA pour m'avoir donné le contact me permettant d'obtenir ce stage.

Je remercie aussi les membres de l'IUT de Bourg en Bresse pour leur accueil et leur bienveillance à mon égard.

Enfin je remercie plus particulièrement mes deux encadrants de stage Fabrice JAILLET et Jean-Philippe FARRUGIA qui m'ont donné ma chance. Ils ont aussi été présent pour moi tout au long du stage.

# Sommaire

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>L'entreprise</b>	<b>5</b>
2.1	Prise de contact . . . . .	5
2.2	LIRIS . . . . .	5
<b>3</b>	<b>Objectif du Projet</b>	<b>6</b>
3.1	Contexte . . . . .	6
3.2	Projet existant . . . . .	6
3.3	Reprise de projet . . . . .	6
<b>4</b>	<b>Architecture du Projet</b>	<b>8</b>
4.1	Environnement matériel . . . . .	8
4.2	Environnement logiciel . . . . .	8
<b>5</b>	<b>L'application</b>	<b>10</b>
5.1	Calculs sur les maillages . . . . .	10
5.2	Mise à jour . . . . .	12
5.3	Fonctionnement en 4 modes . . . . .	13
5.4	Ajouts globaux . . . . .	15
<b>6</b>	<b>Analyse des résultats</b>	<b>19</b>
6.1	Données temporelles . . . . .	19
6.2	Données du Maillage . . . . .	19
<b>7</b>	<b>Conclusion</b>	<b>20</b>
7.1	Difficultés rencontrées . . . . .	20
7.2	Points d'amélioration . . . . .	20
7.3	Apports du stage . . . . .	20

# 1 Introduction

L'utilisation de modèles 3D numériques est fréquente dans les domaines de la simulation, des jeux vidéos ou encore des effets spéciaux. L'acquisition des données permettant de recréer un objet en 3D peut parfois s'avérer difficile et produire des résultats incomplets avec des objets plein de trous.

La réalité virtuelle permet de simuler un environnement en 3 dimensions dans lequel un utilisateur peut interagir. Il ne faut pas la confondre avec la réalité augmentée qui elle consiste à ajouter des éléments virtuels dans un monde réel tandis que la réalité virtuelle est totalement dans un monde virtuel. Cette technologie se développe de plus en plus surtout pour l'apprentissage puisqu'elle permet de simuler une situation précise et de pouvoir interagir dans cette situation.

Ce stage a pour objectif de manipuler un maillage 3D dont il manque des parties. Par le biais de plusieurs manipulations de l'utilisateur associées à différents calculs, le but est de reconstituer intégralement l'objet. Il s'agit d'une reprise de projet, cela implique donc une relecture et remise à niveau du code. Le deuxième objectif est de pouvoir mesurer l'apport de la réalité virtuelle dans cette expérience par rapport au classique clavier-souris et donc l'apport de la 3D face à la 2D.

## 2 L'entreprise

### 2.1 Prise de contact

Étant élève à l'INSA en quatrième année, j'étais à la recherche d'un stage d'une durée de 3 mois pendant mes vacances d'été. Je désirais que ce stage soit dans le domaine de l'informatique graphique notamment avec l'utilisation de l'outil Unity 3D que j'avais eu l'occasion d'utiliser lors d'un projet d'école sur l'archéologie en réalité virtuelle.

Grâce à une professeure de l'INSA, j'ai pu avoir contact avec l'IUT de Bourg en Bresse. Il m'a été proposé ce projet mêlant manipulation de maillage 3D et réalité virtuelle, co-encadré par Fabrice JAILLET et Jean-Philippe FARRUGIA. Ce stage est également en collaboration avec l'Institut Image à Chalon-sur-Saône.

### 2.2 LIRIS

Le LIRIS ou Laboratoire d'InfoRmatique en Image et Systèmes d'information est une unité mixte de recherche portée par par le CNRS, l'INSA de Lyon, l'Université Claude Bernard Lyon 1, l'Université Lumière Lyon 2 et l'École Centrale de Lyon. Il est composé de 330 membres qui sont répartis selon 14 équipes de recherche et 6 pôles de compétences avec des activités reconnues au niveau international.

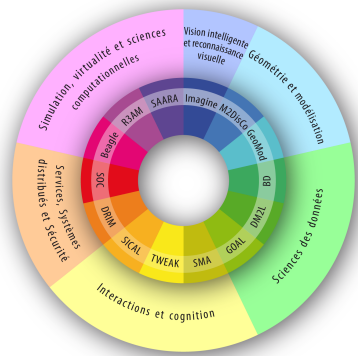


Figure 1: Pôles et équipes du Liris

Mes deux encadrants font également partie de Liris. Fabrice Jaillet étant enseignant chercheur dans l'équipe SAARA tandis que Jean-Philippe Farrugia est enseignant chercheur dans l'équipe R3AM. Tous deux font donc partie du pôle "Simulation, virtualité et sciences computationnelles".

## 3 Objectif du Projet

### 3.1 Contexte

La problématique de ce projet s'est beaucoup transformée au cours du temps. A l'origine, un stage a été réalisé en 2011 par Zhifan Jiang intitulé "Reconstruction et complétion de maillages sous contraintes" [7]. Au départ, le but était de récupérer des modèles géométriques d'organes provenant de données issues d'appareils d'imagerie médicale. Le problème de ces données est qu'elles ne représentaient qu'une partie de l'organe. Cela peut s'expliquer par deux raisons. La première est qu'il manquait les bords, en effet, pour l'IRM par exemple, on limite la zone pour des raisons de temps d'acquisition et de coût. La seconde est que l'organe était composé de trous du fait que le patient puisse un peu bouger lors de la récupération de données. Il fallait donc trouver un moyen de reconstruire l'organe dans sa totalité à partir de ces données.

Ensuite, la problématique a été reprise en 2014 par Armand Le Gouguec et renommée "Complétion de maillage 3D par analyse sémantique" [1]. Ce projet a permis de trouver une méthode plus souple pour la complétion de trous de maillage et d'élargir les domaines d'application de la problématique avec la rétro-conception qui s'appliquerait dans les domaines du patrimoine architectural ou de l'ingénierie mécanique. Cette étude a permis de mettre en place une méthode basée sur l'utilisation de Ridges que je présenterais un peu plus tard dans ce rapport. A l'issue de ce stage l'utilisation d'une interface a été le point de démarrage du suivant.

En 2019, ce stage a été poursuivi par Julien Brissonnet avec la problématique "Complétion de modèles numériques 3D en immersion virtuelle" [3]. Les objectifs de ce stage étaient double. Dans un premier temps, la création d'un outil logiciel basé sur les précédentes recherches et permettant ainsi d'évaluer leur pertinence. Dans un second temps, l'évaluation de l'utilisation de la réalité virtuelle et donc une interface 3D par rapport à une interface 2D traditionnelle.

### 3.2 Projet existant

Le projet avant mon stage était donc celui de Julien Brissonnet. Dans ce projet les calculs ont été mis en place et une interface utilisateur a été créée. L'utilisateur pouvait utiliser l'application soit au clavier souris, soit avec le bras haptique Phantom Omni associé à un casque de réalité virtuelle HTC Vive. L'intérêt de ce bras haptique était de pouvoir avoir un retour d'effort sur les différentes manipulations effectuées.

A l'issue de ce stage, certains points d'améliorations ont été mis en avant:

- Certaines opérations de calculs peuvent prendre du temps.
- Certains calculs pourraient être réduits à des zones spécifiques du maillage.
- Les calculs et leur bon fonctionnement sont assez sensibles à la qualité du maillage original.

### 3.3 Reprise de projet

L'objectif est toujours le même, développer un outil de manipulation de maillage 3D et déterminer l'apport de la réalité virtuelle dans cette application.

La reprise se traduit d'abord par une phase de remise à niveau du projet que ce soit avec Unity ou avec les différents assets et packages utilisés.



Figure 2: Bras haptique

Ensuite, les outils de manipulation ont été changés. Pendant mon stage, je n'ai pas eu accès au bras haptique. Il a donc fallu s'adapter et utiliser l'application avec les deux manettes de l'HTC Vive. Cela fait néanmoins perdre la dimension Haptique de l'application.

Une restructuration de l'application ainsi que l'ajout de fonctionnalités. Cela permet de simplifier son utilisation mais aussi d'apporter de nouvelles possibilités d'interactions.

Enfin une partie de tests permettant de comparer l'utilisation en mode clavier-souris avec l'utilisation en mode HTC Vive.

## 4 Architecture du Projet

L'architecture du projet se basera exclusivement sur la version actuelle de l'application et non pas sur ce qu'elle a pu être au début du stage lors de la reprise du projet

### 4.1 Environnement matériel

L'application se base sur deux modes : un mode en 2D et un mode en 3D.

Le mode 2D utilise simplement l'écran de l'ordinateur ainsi que le clavier et la souris.

Du fait que je n'ai pas eu accès au bras haptique, l'application utilise un HTC Vive pour le mode en 3D. Il est composé de trois éléments importants: un casque, une paire de contrôleurs et une paire d'émetteurs infrarouges. Le dispositif utilise la technologie de "positional tracking" pour transcrire les mouvements de la tête et des mains dans l'environnement virtuel, ce qui permet une bonne immersion visuelle.



Figure 3: HTC Vive

Le seul inconvénient est que les manettes ne permettent pas de produire un retour haptique pour l'utilisateur comme peut le faire le bras haptique.

### 4.2 Environnement logiciel

#### 4.2.1 Unity 3D

Unity 3D est un moteur de jeu permettant de créer simplement des objets et des scènes en 3D. Il permet également de définir l'environnement dans lequel l'utilisateur va pouvoir interagir. Il



est également compatible avec plusieurs casques de réalité virtuelle dont le HTC Vive. Il lui est associé Visual Studio qui est un IDE et permet d'éditer de nouveaux scripts en C#.

Un projet Unity est composé d'éléments regroupés sous le nom d'Assets. Cela peut être des scripts, images, sons, modèles 3D. On peut également utiliser différents packages que l'on peut importer depuis l'Asset Store de Unity et qui correspondent à un ensemble d'assets développés par la communauté de Unity et qui peuvent être réutilisés dans des projets.

Parmi ces packages, SteamVR et VRTK permettent de gérer plus facilement l'utilisation du casque de réalité virtuelle. Ces packages seront présentés plus en détail dans la partie "Mise à jour" de la section "L'application".

Enfin Unity3D possède un système de GameObject qui sont tous les objets présents dans une scène. Ce sont des sortes de conteneur dans une scène Unity. On leur associe différents composants tels que des meshes, des lumières, des scripts, des matériaux, ... Ils peuvent ainsi représenter un personnage jouable, un élément de décor, une caméra, une lumière , ...

#### **4.2.2 CGAL**

Les calculs effectués lors de l'utilisation de l'application sont complexes. Il est donc difficile de les réaliser avec Unity de façon rapide et optimisé. Ainsi, tous ces calculs seront réalisés par CGAL.

CGAL, pour Computational Geometry Algorithm Library, est une librairie écrite en C++ qui a été créée par la collaboration de plusieurs universités, entreprises et instituts de recherche. Elle met à disposition différents algorithmes et structures de données efficaces. Elle permet notamment de représenter des maillages 3D et d'effectuer des calculs dessus.

## 5 L'application

L'application étant une combinaison de deux stages, cette partie présentera à la fois ce qui a été produit lors du précédent stage et ce qui a été ajouté ou modifié lors de la reprise.

### 5.1 Calculs sur les maillages

Les recherches dans les précédents stage ont permis de trouver des calculs permettant d'obtenir un maillage "complet" à partir d'un maillage troué. Ces calculs ont été mis en place dans le projet par Julien. Cette partie du rapport permet de voir quels sont ces calculs et ainsi de mieux comprendre les étapes que devra réaliser l'utilisateur dans le projet.

#### 5.1.1 Calcul des ridges

Les ridges ou lignes de courbures extrémales sont des lignes qui nous permettent de représenter le contour d'un objet. La courbure d'un objet est une valeur définissant le fait que l'objet soit plus ou moins courbé. Dans un espace 3D, on calcule la courbure en utilisant un plan qui passe par la normale à la surface de l'objet au niveau du point que l'on est en train d'étudier. L'intersection de ce plan et de la surface de l'objet forme une courbe dont on étudie la courbure. Le problème est qu'il y a une infinité de plans possibles puisque l'on peut effectuer une rotation du plan selon le vecteur normal. On calcul alors la courbure pour chacun de ces plans et on garde la valeur minimale et la valeur maximale, cela nous donne la courbure principale minimale et la courbure principale maximale.

On associe des directions à ces courbures principales. Ces directions sont représentées par l'intersection entre les plans des courbures principales et le plan tangent à la surface au niveau du point que l'on étudie.

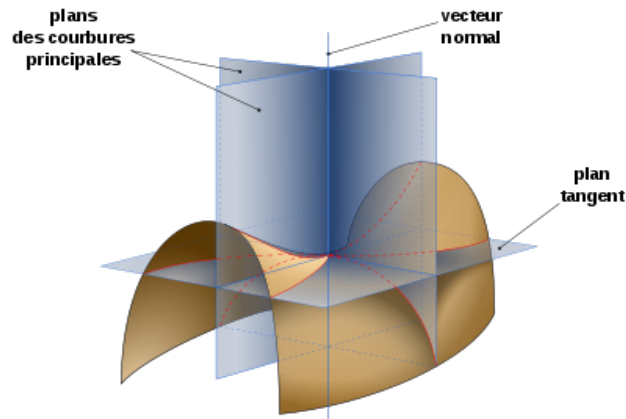


Figure 4: Plans définissant les courbures principales d'une surface (Wikipédia)

Il existe des points pour lesquels les courbures maximale et minimale sont égales, ce sont des ombilics. Les lignes utilisées pour l'application sont celles qui appartiennent à la surface de l'objet et sont tangentes aux directions principales. Il en existe une infinité et on peut remarquer

que chaque point est traversée par deux lignes de courbure (une maximale et une minimale) sauf les ombilics.

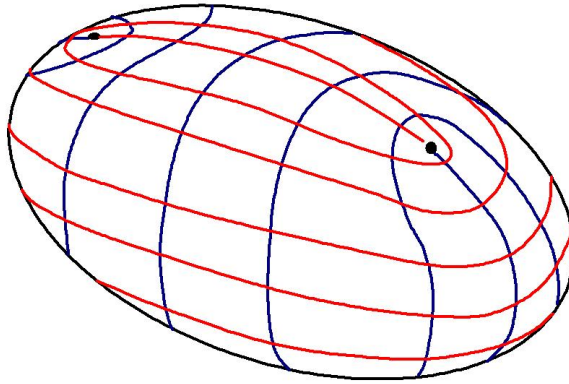


Figure 5: Lignes de courbures et Ombilics. En bleu les lignes de courbure maximales et en rouge les lignes de courbure minimales (Wikipedia)

### 5.1.2 Complétion de trous

La complétion consiste à "compléter" le maillage pour qu'il n'ait plus de trou. Le but est de localiser le trou et de le reboucher. L'algorithme se fait en trois étapes. Tout d'abord, la zone du trou est triangulée de façon grossière. Le but est de trianguler le trou sans ajouter de nouveaux sommets et à minimiser la surface de ce que l'on appelle le "patch" (trou rebouché). Ensuite le maillage du patch est raffiné. Le but est de faire en sorte que le maillage du patch soit plus uniforme avec une densité proche de celle du reste de l'objet. Enfin, le patch est lissé en prenant en compte les bords du trou pour faire en sorte qu'il y ait une continuité des tangentes au sommet du patch.

### 5.1.3 Déformation

La déformation s'appuie sur les deux étapes précédentes. En effet le but est de déformer le patch en s'appuyant sur des lignes de contraintes définies grâce aux ridges. Le but est de projeter les lignes de contraintes sur le patch et donc d'associer chaque point de la projection à chaque point des lignes. Ainsi on définit des contraintes de position à certains points du patch. Pour résoudre le système de déformation, il est nécessaire d'avoir:

- un maillage triangulaire surfacique
- un ensemble de sommets du maillage constituant la zone à déformer (ROI region-of-interest)
- un sous-ensemble de la ROI représentant les sommets que l'utilisateur souhaite modifier.
- un ensemble de positions cibles associées aux sommets de l'ensemble précédent.

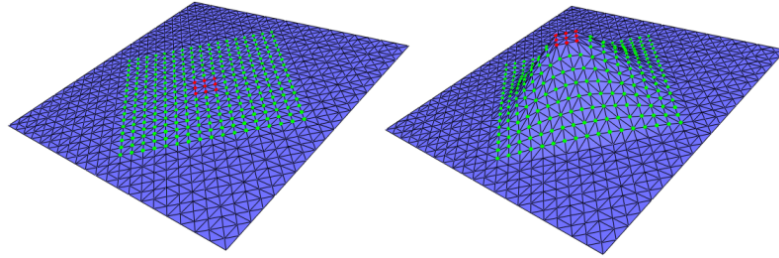


Figure 6: Déformation d'un maillage surfacique triangulaire: en vert la ROI, en rouge les sommets de contrôle

#### 5.1.4 Utilisation de CGAL

Du fait que les recherches des précédents stages aient été effectuées un certain temps avant celui-ci, les calculs ont déjà pu être implémentés par CGAL. Ainsi, on retrouve les calculs de ridges dans "Approximation of Ridges and Umbilics on Triangulated Surface Meshes" [4], la complétion dans le package "Polygon Mesh Processing" [5] et enfin la déformation dans "Triangulated Surface Mesh Deformation" [6].

CGAL étant codé en C++, une DLL a été créée permettant de faire le lien entre Unity et CGAL. Ainsi lors de l'appel d'un calcul, les données importantes du maillage ou des ridges sont transférées grâce à des tableaux de double contenant la position des vertices (points du maillage), les index des vertices pour les triangles, ...

## 5.2 Mise à jour

Au début du stage le projet a dû être remis à jour pour ce qui est des logiciels mais aussi pour le matériel.

### 5.2.1 Bras Haptique et HTC Vive

L'intégralité de l'application fonctionnait grâce à un bras haptique. N'ayant pas accès à ce bras lors de mon stage, l'intégralité de l'application et donc des scripts ont été adaptés à l'utilisation des manettes de l'HTC Vive. Le seul inconvénient avec ce changement est que l'on perd la dimension haptique pour l'utilisateur lors de la manipulation.

### 5.2.2 Unity 3D

Unity3D est en constante évolution. Ses mises à jour sont nombreuses, ainsi il faut mettre l'intégralité du projet à jour. Cela se traduit généralement par le remplacement de certaines fonctions qui sont devenues obsolètes.

Le passage à Unity 2019.1.3 a aussi permis d'intégrer le package "XR\_input Legacy". Il fournit des outils sur tous les périphériques d'entrée supportés par Unity pour la réalité virtuelle.

### 5.2.3 SteamVR et VRTK

L'application fonctionnait en utilisant deux packages particuliers : SteamVR et VRTK (Virtual Reality Tool Kit). Ils permettent la gestion du tracking des éléments de réalité virtuelle et fournissent également des éléments permettant d'interagir avec dans Unity3D.

Le problème est que SteamVR passait à la version 2.0 qui est incompatible avec VRTK. Un choix a dû être fait entre les deux. Finalement c'est VRTK qui a été conservé puisqu'il fournit des éléments de tracking mais aussi des éléments permettant de faciliter l'utilisation sous Unity3D.

Il a même été décidé d'utiliser VRTK 4.0 dans sa version bêta puisque dans cette version, il n'y a pas besoin de configurer un "SDK Manager" comme il a fallu le faire dans la version précédente, ce "SDK Manager" étant SteamVR. L'autre point positif est que cette nouvelle version de VRTK prend en compte UnityXR qui a été ajouté depuis Unity 2019. L'inconvénient de cette bêta est que les éléments déjà codés permettant d'interagir dans Unity en réalité virtuelle n'ont pas tous été développés et sont à recoder ( exemple : l'utilisation de rayon pour interagir avec un menu).

## 5.3 Fonctionnement en 4 modes

Lors du début du stage, l'application fonctionnait sans "modes" et s'en retrouvait assez complexe et le menu était rempli de boutons. Le fonctionnement a été revu pour se construire suivant 4 modes permettant à l'utilisateur de repérer les étapes clés de l'application. Ces modes sont aussi associés à des couleurs de lignes pour une meilleure compréhension.

### 5.3.1 Mode vue de l'objet

Ce mode permet d'avoir une vue complète de l'objet sans les éléments de construction qui l'accompagne (lignes et plans). Un mode "couleur" a été ajouté permettant de voir comment est construit le maillage en colorant les sommets de couleurs différentes.

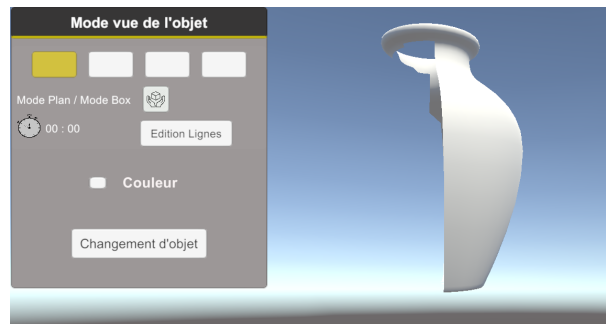


Figure 7: Mode vue de l'objet

### 5.3.2 Mode préparation des symétries

Ce mode permet de construire les lignes de symétrie. Il est basé sur la construction de lignes rouges. La première action est la construction des ridges (voir Calculs sur les maillages). Pour

cela un menu est disponible pour gérer les différents paramètres possibles pour le calcul. Ensuite, l'utilisateur peut effectuer une complétion pour créer le patch dont on a besoin (voir Calculs sur les maillages). Ensuite un plan de symétrie doit être créé. L'application est centré pour l'instant sur la complétion d'objets possédant une symétrie. Une fois ce plan créé et bien positionné, les lignes de symétrie sont construites à partir des ridges. Ces lignes qui sont représentées en bleu peuvent ensuite être ajustées en temps réel en bougeant le plan. La confirmation du placement de ces lignes conclut l'utilisation de ce mode.

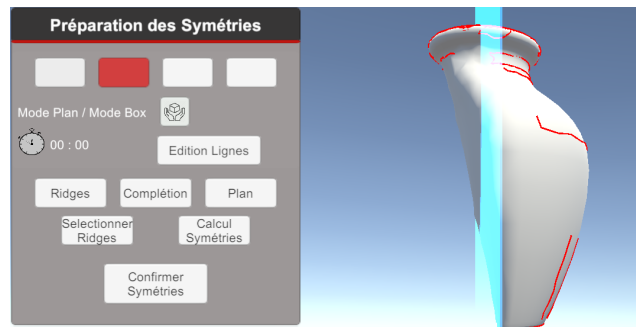


Figure 8: Mode préparation des symétries

### 5.3.3 Mode préparation de la déformation

Ce mode permet comme son nom l'indique de préparer l'objet à sa déformation. Il se base sur les lignes bleues créées précédemment. Le but est de relier les points du patch aux points des lignes bleues pour indiquer comment se fera la projection. Ce mode consiste principalement à orienter un plan de projection pour indiquer dans quelle direction les points des lignes bleues doivent se projeter sur le patch. L'avantage est que l'on peut voir cette projection en temps réel pour mieux l'ajuster. Ce mode se termine par la déformation du patch.

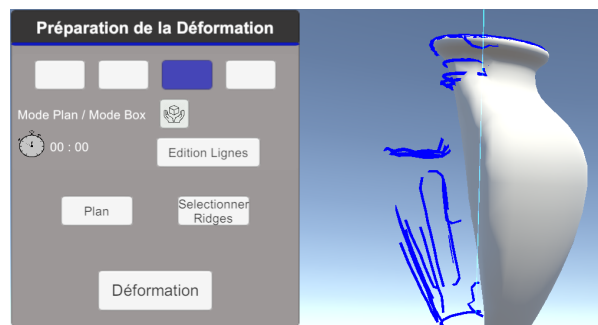


Figure 9: Mode préparation de la déformation

### 5.3.4 Mode finalisation

Ce mode permet d'effectuer les derniers changements sur l'objet. Un lissage global peut lui être appliqué ou des déformation via l'édition de lignes (voir Ajouts globaux / Édition de lignes) Enfin c'est ici que l'objet peut être enregistré.

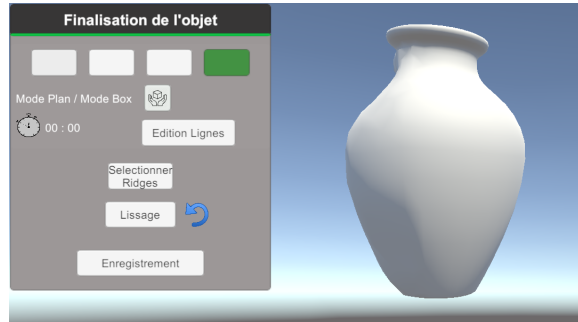


Figure 10: Mode finalisation

### 5.3.5 Avantage du passage à 4 modes

Les différentes étapes des 4 modes peuvent être retrouvées dans le Manuel Utilisateur [2] avec un détail sur chacune d'entre elle en fonction de l'environnement utilisé ( 2D ou 3D).

L'avantage de ces 4 modes est que l'utilisateur a moins de boutons en un seul menu. En effet ils sont partagés entre le 4. De plus le code couleur des lignes permet de mieux comprendre leur utilisation. Enfin l'avantage est que l'on peut passer d'un mode à l'autre à n'importe quel moment de l'expérience. Si une étape à mal été réalisée ou si on s'aperçoit que le résultat ne nous convient pas, il est facile de revenir en arrière et de modifier ce que l'on veut.

## 5.4 Ajouts globaux

De façon plus générale par rapport aux 4 modes, des ajouts ont été apportés sur toute l'application pendant mon stage permettant plus d'efficacité ou simplement de nouvelles fonctionnalités.

### 5.4.1 Calculs en temps réel

Le fait d'avoir des calculs complexes implique des temps de calculs. Le problème est que l'on veut s'approcher le plus possible de calculs en temps réels. En effet, si les calculs ne sont pas en temps réel, l'application aura un "freeze", cela implique que l'écran se bloque, comme mis sur pause et qu'aucune interaction n'est possible tant que le calcul n'est pas effectué. Cela est un peu gênant pour l'utilisation clavier-souris. Ça l'est beaucoup plus pour l'utilisation en réalité virtuelle. En effet, un "freeze" en réalité virtuelle peut provoquer des gênes importantes pour l'utilisateur puisque sa vue ne sera pas en accord avec ses mouvements. Une sécurité est tout de même présente qui renvoie l'utilisateur dans un menu VR pendant le "freeze". Pour pallier ceci, des Threads ont été mis en place pour chaque calcul. Ainsi le thread principal se concentre sur l'environnement virtuel et évite le freeze pendant que le second thread s'occupe des calculs.

L'utilisateur peut ainsi continuer de manipuler l'objet ou les éléments de construction pendant que le calcul s'effectue.

#### 5.4.2 La Box

Toujours dans une optique d'amélioration de l'efficacité des calculs, une "Box" est utilisée. Cette Box est en fait un cube transparent. L'objectif est d'englober la zone de l'objet où l'on veut effectuer le calcul. Cette zone est ensuite détachée du mesh et envoyée à la DLL pour effectuer les calculs. Une fois les résultats renvoyés, le "nouveau sous-mesh" est rattaché au mesh global. Cela permet d'étudier des objets plus gros ou même composé de plusieurs trous et aide la machine à trouver le trou pour la complétion par exemple. Cet outil est donc utilisable dans n'importe quel mode et permet des calculs plus localisés.

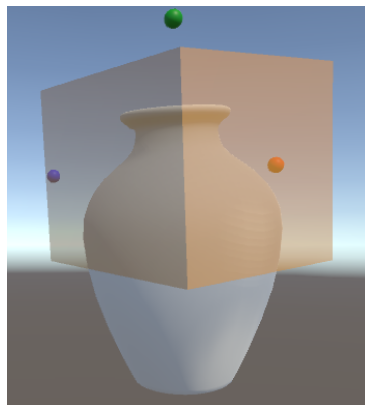


Figure 11: Sélection d'une partie de l'objet à l'aide de la Box

Au niveau du code, cela se déroule en trois étapes. Tout d'abord, on sépare le Mesh en deux sous-meshs: celui à l'extérieur de la Box et celui à l'intérieur. Pour cela on compare chaque points composant le Mesh global avec les différents plans de la Box. Au niveau de la limite de découpe, les points sont dupliqués pour être présents à la fois dans le Mesh interne et dans le Mesh externe. La deuxième étape consiste à ré-indexer le sous-mesh de l'intérieur de la Box pour ensuite le passer en calcul à la DLL. Enfin, lorsque le calcul est terminé, on "recoud" le Mesh sur l'autre sous-mesh. Pour cela on associe les points qui avaient été dupliqués auparavant en comparant leurs positions (les calculs ne modifient pas les points aux bords) ce qui permet de reconstruire le Mesh global.

#### 5.4.3 Édition de lignes

L'édition de lignes permet d'ajouter ou de modifier des lignes. Elle se divise en un mode Dessin pour la création et un mode Édition pour la modification. Elle est utilisable dans les 3 modes de l'application et s'adapte en fonction.

- **Le mode dessin** permet de créer soit des lignes rouges, soit des lignes bleues, soit des lignes noires (mode 2,3 et 4 respectivement). Les lignes rouges seront de nouvelles lignes auxquelles on pourra appliquer des symétries en plus des ridges. Les lignes bleues



seront des lignes grâce auxquelles on pourra effectuer la déformation en plus des lignes de symétries. Enfin les lignes noires du mode Finalisation permettent une dernière sorte de déformation qui sera présentée un peu plus bas.

- **Le mode édition** permet de manipuler les lignes. Lorsqu'une ligne est sélectionnée, ses points de construction apparaissent et peuvent être déplacés. Lors du déplacement d'un point la ligne est calculée en temps réel pour une meilleure visualisation du résultat. En mode Finalisation, l'édition de ligne peut être accompagnée d'une déformation.

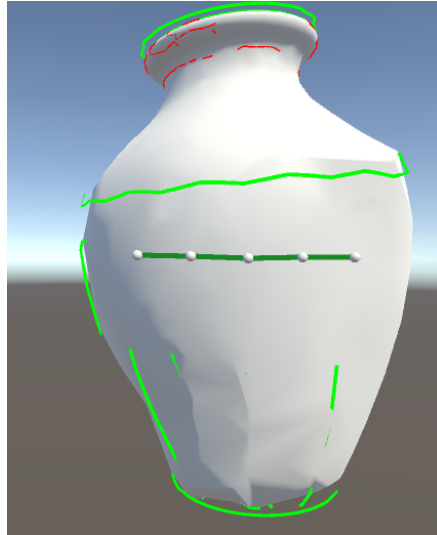


Figure 12: Apparition des points de contrôle d'une ligne en mode édition

- **L'édition avec déformation** du mode Finalisation s'effectue lorsqu'un point d'une ligne est déplacé. Une déformation de l'objet est alors appliquée le long de cette ligne. Cela permet d'effectuer des dernières retouches à la fin de l'expérience.

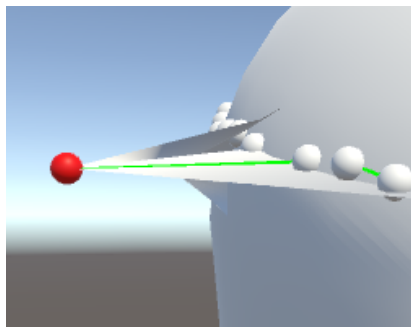


Figure 13: Déformation de l'objet lors du déplacement d'un point

#### **5.4.4 Changement d'objet**

Au début du stage, un seul objet était utilisé. Un menu de changement d'objet a donc été mis en place. La particularité de ce menu est qu'il est codé de façon à ce qu'il se remplit automatiquement en fonction des données qu'on lui fournit. Autrement dit, un script lui est associé; et dans ce script il y a une liste de données qui peuvent être remplies facilement par un développeur. On peut donc facilement augmenter la taille de la liste pour y ajouter les données d'un nouvel objet. Ces données sont son nom, son maillage troué, son maillage sans trou s'il existe et sa taille. Une fois l'expérience finie sur un objet, on peut facilement passer à un autre et recommencer.

#### **5.4.5 Aides pour l'utilisateur**

L'expérience pouvant être compliquée à appréhender différentes aides ont été mises en place pour l'utilisateur à l'intérieur même de l'application.

Un menu d'instructions qui s'adapte au mode dans lequel se trouve l'utilisateur pour lui indiquer l'ordre dans lequel il doit réaliser les étapes de l'expérience.

Un menu d'aide permettant d'expliquer à l'utilisateur les principes et objectifs de chaque étape mais aussi lui permettant d'apprendre les contrôles et méthodes de manipulation qu'il soit en utilisation clavier-souris ou en utilisation HTC Vive. Ce menu s'adapte également au mode dans lequel se trouve l'utilisateur.

Enfin un Manuel utilisateur est disponible et reprend les différentes explications du menu d'aide.

## 6 Analyse des résultats

Le deuxième objectif du projet étant de déterminer l'apport de l'utilisation de la réalité virtuelle, des éléments ont été mis en place pour récolter des données. A l'origine, le stage devait se finir par une phase de tests sur différents groupes. Par manque de temps, ces tests n'ont pas été effectués mais tous les éléments sont prêts pour les faire.

### 6.1 Données temporelles

Un chronomètre a été mis en place. Il doit être activé manuellement par l'appui d'un bouton au début de l'expérience. Ce chronomètre mesure le temps total mais aussi le temps passé sur chacun des 4 modes. Cela permet de voir quel mode est le plus sollicité lors de l'expérience mais aussi de comparer les temps entre les environnements 2D et 3D.

### 6.2 Données du Maillage

Dans le dernier mode, lorsque l'objet convient à l'utilisateur, il peut être enregistré. Il est alors sauvegardé dans un fichier ".off" qui peut être lu par n'importe quel lecteur d'objet 3D comme par exemple MeshLab.

Si l'objet possède un élément de comparaison (voir Ajouts Globaux / Changement d'objet) alors il est comparé à celui-ci et une distance de Hausdorff est calculée. Cette distance consiste à calculer la distance minimale entre chaque point d'un Mesh aux points d'un autre. Une fois le minimum trouvé pour chaque point, la distance de Hausdorff correspond au maximum parmi ces minimums. La distance en elle-même n'apporte pas beaucoup mais le calcul des minimums permet ensuite d'appliquer une coloration à chaque point en fonction de la distance et de pouvoir ainsi distinguer les endroits où il y a une différence importante. Cette coloration est appliquée à la fois sur Unity pour que l'utilisateur ait un retour sur ce qu'il a fait mais aussi dans l'enregistrement du fichier ".off" pour pouvoir l'ouvrir en couleur.

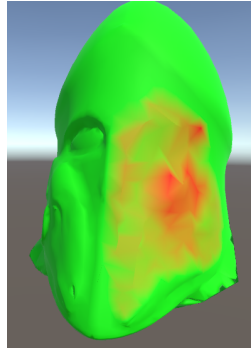


Figure 14: Coloration de l'objet final en utilisant la distance de Hausdorff

## 7 Conclusion

Le projet est fonctionnel et est prêt pour des phases de tests et de récolte de données. Il est envisagé d'écrire un papier dessus et même de le présenter à la conférence jFIG2019.

### 7.1 Difficultés rencontrées

La reprise de projet a été d'abord assez compliquée pour moi. En effet les calculs n'étaient pas forcément évidents à comprendre et l'utilisation de DLLs m'a posé pas mal de problèmes.

La remise à niveau du code a pris pas mal de temps. Le fait d'enlever SteamVR impliquait de devoir recoder pas mal d'éléments qui aident à la manipulation en VR.

Le temps m'a manqué pour terminer le projet et pouvoir passer aux phases de tests. Je pense qu'une ou deux semaines supplémentaires m'auraient permis de terminer parfaitement ce projet.

### 7.2 Points d'amélioration

Même si de nouvelles fonctionnalités ont été ajoutées pour apporter une plus "vaste" utilisation et une meilleure compréhension de l'application, des améliorations peuvent être envisagées :

- L'utilisation de l'haptique dans l'application a été perdue. Il serait intéressant d'essayer de récupérer cette fonctionnalité. Cela pourrait par exemple s'effectuer grâce à des vibrations sur les manettes de l'HTC Vive ou par la réutilisation du bras haptique.
- L'application est centré sur l'utilisation de symétries planaires. Même si l'édition de nouvelles lignes permet de pallier un peu cela, il serait intéressant de pouvoir effectuer d'autre types de symétries ou même de pouvoir extraire un motif qui pourrait se répéter à un autre endroit de l'objet pour pouvoir le placer à notre convenance.
- Les données récoltées pour les tests sont peut être peu nombreuses. Il faudrait voir quel autre type de données peuvent être récoltées à partir de l'application.

### 7.3 Apports du stage

Ce stage m'a beaucoup apporté sur de nombreux plans:

Tout d'abord il m'a permis d'acquérir de nouvelles connaissances sur Unity3D combiné à la réalité virtuelle. En effet ces connaissances se limitaient à un projet de 4ème année à l'INSA sur de l'archéologie virtuelle.

J'ai ensuite pu découvrir le travail d'un ingénieur dans un environnement de recherche. Un environnement de rigueur et d'innovation qui me plaît beaucoup. Cela m'a permis de développer mon autonomie dans le travail mais aussi mon sens de l'initiative en proposant de nouvelles fonctionnalités.

Pour ce qui est de mon futur, maintenant que j'ai découvert le monde de la recherche, je souhaiterais faire mon prochain stage dans une entreprise pour avoir une comparaison de ces deux univers. Ce qui est sûr c'est que le domaine de l'informatique graphique me plaît beaucoup et que c'est là que je veux me diriger. Je garde la recherche en tête en sachant que cela pourrait beaucoup m'intéresser de travailler en tant qu'enseignant chercheur.

## References

- [1] Ruding Lou Armand Le Gouguec Fabrice Jaillet. *Complétion de maillages 3D par analyse sémantique*. [Rapport de projet de fin d'études]. Arts et Métiers ParisTech Cluny, 2014.
- [2] Jean-Philippe Farrugia Jomain Benjamin Fabrice Jaillet. *Manuel Utilisateur : Complétion de maillages numériques 3D en immersion virtuelle*. [Manuel utilisateur]. IUT Lyon 1 Bourg-en-Bresse, 2019.
- [3] Jean-Rémy Chardonnet Julien Brissonnet Fabrice Jaillet. *Complétion de maillages numériques 3D en immersion virtuelle*. [Rapport de stage de fin d'études]. Arts et Métiers ParisTech Cluny, 2019.
- [4] Marc Pouget and Frédéric Cazals. *Approximation of Ridges and Umbilics on Triangulated Surface Meshes*. [CGAL User and Reference Manual]. CGAL Editorial Board, 4.10 edition, 2017.
- [5] Ilker O. Yaz Sébastien Lorient Jane Tournois. *Polygon Mesh Processing*. [CGAL User and Reference Manual]. CGAL Editorial Board, 4.10 edition, 2017.
- [6] Yin Xu Sébastien Lorient Olga Sorkine-Hornung and Ilker O Yaz. *Triangulated Surface Mesh Deformation*. [CGAL User and Reference Manual]. CGAL Editorial Board, 4.10 edition, 2017.
- [7] Florence Zara Zhifan Jiang Fabrice Jaillet. *Reconstruction et complétion de maillages sous contraintes*. [Rapport de recherche]. LIRIS UMR CNRS 5205, 2011.

## Résumé

Ce stage s'inscrit dans la continuité de plusieurs stages de recherche sur la complétion de modèles 3D. Son objectif est double. Tout d'abord le développement d'une application permettant à un utilisateur de réaliser toutes les étapes lui permettant de réaliser la complétion d'un objet en 3D. Ensuite, le but est de pouvoir comparer l'utilisation de l'application dans un environnement 2D en utilisant la souris et le clavier avec l'utilisation dans un environnement 3d en réalité virtuelle en utilisant un HTC vive.

Ce stage étant une reprise de projet, il implique tout d'abord une phase de lecture et de compréhension du code suivie d'une phase de remise à niveau des scripts en fonction des nouvelles versions des logiciels. La suite est composée d'améliorations de l'application que ce soit par l'ajout de nouvelles fonctionnalités ou par la modification de certaines déjà existantes. La restructuration du code selon 4 modes d'utilisations a par exemple été l'une des modifications.

A travers ce stage, j'ai pu consolider mes connaissances sur Unity3D et sur le domaine de la réalité virtuelle mais aussi découvrir la manipulation de maillages 3D. Cela m'a également permis une immersion dans le monde de la recherche. J'ai pu y développer mon autonomie et mon initiative. De plus cela a pu me donner des pistes pour mon futur métier.

## Abstract

This internship is a continuation of several research internships on the completion of 3D models. Its objective is twofold. First of all, the development of an application allowing a user to perform all the steps allowing him to complete an object in 3D. Then, the goal is to be able to compare the use of the application in a 2D environment using the mouse and keyboard with the use in a 3d environment in virtual reality using a live HTC.

This internship being a project resumption, it implies first of all a phase of reading and understanding the code followed by a phase of updating the scripts according to the new versions of the software. The suite consists of improvements to the application, either by adding new functionalities or by modifying existing ones. The restructuring of the code according to 4 modes of use was one of the changes, for example.

Through this internship, I was able to consolidate my knowledge of Unity3D and the field of virtual reality but also discover the manipulation of 3D meshes. It also allowed me to immerse myself in the world of research. I was able to develop my autonomy and initiative. Moreover, it could give me some ideas for my future job.