

IMT Atlantique
Dépt. Informatique
Technopôle de Brest-Iroise - CS 83818
29238 Brest Cedex 3
Téléphone : +33 (0)2 29 00 13 04
Télécopie : +33 (0)2 29 00 10 12
URL : www.imt-atlantique.fr



Rendu Final PRONTO

Groupe 22

DIFFUSION PRIVÉE : UE PRONTO

Reconstruction du modèle 3D d'un objet à partir d'une vidéo

Auteurs :

VATON Samuel

HIGNARD NAUDEAU Pierre-Antoine

PELLETIER Anne-Lise

PINSARD Achile

Encadrante & Relectrice :

BERGANTIN Lucia

Date d'édition : 21 mai 2025

Version : 2.0



IMT Atlantique

Bretagne-Pays de la Loire
École Mines-Télécom

Client :



Sommaire

1. Introduction	5
2. Cahier des charges fonctionnel	5
2.1. Introduction.....	5
2.1.1. Terminologie.....	5
2.2. Présentation du système	6
2.2.1. Finalité, mission, objectifs	6
2.2.2. Liste des parties prenantes	6
2.2.3. Contexte d'utilisation du système	6
2.3. Expression fonctionnelle du besoin	7
2.3.1. Fonctions de service et de contrainte	7
2.4. Scénarios	7
2.4.1. Utilisation nominale	9
2.4.2. Cas limites et scénarios extrêmes	10
2.5. Validation du besoin.....	10
2.6. Conclusion du Cahier des Charges	11
3. Conception des composants et du système.....	11
3.1. COLMAP.....	11
3.1.1. Feature extraction.....	11
3.1.2. Feature matching	12
3.1.3. Reconstruction	12
3.2. Matériel informatique	12
3.2.1. Difficultés rencontrées.....	13
3.2.2. Solutions	13
3.3. Meshlab	13
3.4. Fusion 360.....	13
3.5. Modèle 3D de la chaise	13
3.6. Imprimante 3D.....	14
4. Développement.....	14
4.1. Code python transformation d'une vidéo en images	15
4.2. Code python pour traitement d'image et reconstruction.....	15
4.2.1. Traitement d'image	15
4.2.2. Reconstruction	16
4.3. Plateforme pour prise de photos	16
5. Intégration des composants et validation.....	17
5.1. Étapes de la reconstruction.....	17
5.1.1. Prise de photos ou vidéos	17
5.1.2. Reconstruction automatique avec COLMAP.....	18
5.1.3. Nettoyage du nuage de points avec Meshlab	19
5.1.4. Création d'une surface avec Meshlab	20
5.1.5. Transformation de la surface en volume avec Meshlab et Fusion 360.....	21
5.2. Comparaison rendu photo ou vidéo.....	23
5.3. Comparaison des résultats entre images traitées et non traitées	24

6. Clôture du projet	25
6.1. Création du nuage points avec COLMAP	25
6.2. Transformation en surface avec Meshlab.....	26
6.3. Transformation de la chaise en volume	29
6.4. Finalité du projet fils rouge.....	29
6.4.1. Comparaison avec Meshlab	29
6.4.2. Comparaison réelle	30
6.5. Livrables	31
6.6. Conclusion.....	31
7. Retour d'expérience sur la gestion de projet.....	32
8. Retours d'expérience sur le travail en équipe.....	36
8.1. Retour individuel - Pierre-Antoine HIGNARD.....	36
8.2. Retour individuel - Samuel VATON	36
8.3. Retour individuel - Anne-Lise PELLETIER	37
8.4. Retour individuel - Achile PINSARD.....	37
8.5. Retour commun	38
9. Conclusion et perspectives	38
Références	39

Liste des figures

1.	Diagramme du système dans son environnement	6
2.	Diagramme des fonctions principales de service et de contrainte	7
3.	Schéma des scénarios d'utilisation du produit et des interactions entre les acteurs, depuis la demande client jusqu'à la modélisation 3D.	9
4.	Flow Chart d'utilisation du logiciel	9
5.	Flow Chart de test des variables choisies	10
6.	Image d'un ensemble de points reconstruits par COLMAP à partir de photos d'une pomme	11
7.	Image de la chaise imprimée en 3D	14
8.	Vue d'ensemble du fond	17
9.	Point de vue de la caméra	17
10.	Caméras du point de vue de l'objet	18
11.	Caméras du point de vue extérieur	18
12.	Capture d'écran de la fenêtre de reconstruction automatique de COLMAP	19
13.	Problème de points "fantômes"	19
14.	Problème de points en bordure	19
15.	Problème de "jupe" avec la reconstruction de surface Poisson	20
16.	Autre problème pour des objets contenant des trous	20
17.	Reconstruction de la chaise par Ball Pivoting	21
18.	Page Close Holes	21
19.	Boutons de préparation : nettoyage, groupes de faces...	22
20.	Bouton Convert Mesh	22
21.	Paramètres de conversion choisis	22
22.	NUAGE de points de la colle avec les photos prises une par une	23
23.	NUAGE de points de la colle avec les photos tirées de la vidéo	23
24.	Surface reconstruite à partir du nuage correspondant aux photos prises une par une	23
25.	Surface reconstruite à partir du nuage correspondant aux photos tirées de la vidéo	23
26.	Image de la chaise avant traitement	24
27.	Image de la chaise après traitement	24
28.	Image de la colle avant traitement	25
29.	Image de la colle après traitement	25
30.	Premier nuage de points de la chaise	26
31.	NUAGE de points de la chaise sans points blancs	27
32.	Surface de la chaise reconstruite avec Meshlab	27
33.	Schéma de modélisation de l'assise de la chaise	28
34.	Schéma du problème de modélisation du pied arrière de la chaise	28
35.	Chaise exportée sous format .stl	29
36.	Comparaison de la chaise, modèle 3D de départ et de la surface reconstruite avec Meshlab	29
37.	chaise initiale	30
38.	Chaise ré-imprimé	30
39.	comparaison des deux impressions avant et après reconstruction	30
40.	Diagramme de Gantt version initial	33
41.	Diagramme de Gantt version mi-projet	34
42.	Diagramme de Gantt version finale	35

Liste des tableaux

1.	Terminologie utilisée	5
2.	Rôles des membres de l'équipe	6

3. Tableau des livrables	31
------------------------------------	----

1. Introduction

Notre projet PRONTO *Reconstruction du modèle 3D d'un objet à partir d'une vidéo* consiste en la recherche des paramètres, des méthodes, et de l'environnement optimal pour la reconstruction du modèle 3D de petits objets à l'aide du logiciel COLMAP [1]. Ce projet répond à une problématique forte : La nécessité de rendre l'outil COLMAP plus accessible et simple d'utilisation pour la modélisation, en occurrence à partir de vidéos.

En effet, cet outil étant puissant, il n'en est pas moins complexe d'utilisation, avec de nombreux paramètres et programmes. C'est pourquoi l'entreprise Renault nous a contacté afin de palier à ces problèmes, et rendre l'outil standardisé pour notre employeur. Les contraintes seront abordées plus tard dans la partie "cahier des charges fonctionnel", mais sa standardisation doit correspondre à une taille de sujet de l'ordre de la dizaine de centimètres. Bien que nous ayons essayé d'adapter le fond d'image [cf. 4.3], il reste nécessaire que notre solution soit portable à la plupart des environnements. Cela est désiré afin de conserver la force de COLMAP et de la Structure from Motion (ou SfM, cf. tableau 1), qui est de pouvoir reconstruire un modèle 3D à partir d'images indépendantes (sans forcément venir du même appareil...) trouvés sur internet.

Ce rendu comporte les ajustements qui ont été soulignés comme à ajuster dans le cahier des charges initial, comme : le plan, exploiter les figures ainsi qu'un développement plus important.

Puis, notre démarche de résolution sera expliquée pas à pas sur les parties 3 à 5.

Enfin, voici une planification brève du rapport : celui-ci présente dans un premier temps les objectifs du projet, la conception et le développement du système, ainsi que son intégration et validation. Il inclut aussi un retour d'expérience sur la gestion du projet et le travail en équipe, suivi d'une réflexion individuelle et d'une bibliographie.

2. Cahier des charges fonctionnel

2.1. Introduction

Le Cahier des Charges fonctionnel rassemble l'ensemble des besoins, attentes, exigences et contraintes générés par les parties prenantes de la solution de **Structure from Motion** dans le contexte de PRONTO de la formation FISE A1 de l'IMT Atlantique.

2.1.1. Terminologie

Le tableau de terminologie est :

Terme	Abréviation	Définition
Structure from Motion	SfM	Technique de reconstruction 3D basée sur des images 2D en estimant la position de la caméra et les points communs.
Multi-View Stereo	MVS	Technique de reconstruction 3D utilisant plusieurs vues d'une scène pour créer un modèle dense en analysant les correspondances entre les images.
COLMAP		Logiciel open-source de photogrammétrie permettant la reconstruction 3D à partir d'images via SfM et MVS.
Mesher		Catégorie d'algorithmes cherchant à reconstruire un nuage de points en surface
Computer Assisted Design	CAD	Catégorie de logiciel de design sur ordinateur, on l'utilisera notamment pour convertir la surface en volume et l'exporter pour impression.

TABLE 1 – Terminologie utilisée

2.2. Présentation du système

2.2.1. Finalité, mission, objectifs

Finalité :

Création du modèle 3D d'un objet à partir d'images.

Mission :

Le logiciel prend des images (ou une vidéo, et dans ce cas en extrait des images) puis utilise le logiciel COLMAP pour en déduire un nuage de points. De ce nuage on déduit une surface avec le logiciel Meshlab. Finalement, on déduit de cette surface un volume avec fusion 360. Nous obtenons alors un modèle 3D, en format .stl utilisable et prêt à impression.

Objectifs :

L'objectif principal est de comprendre les étapes nécessaires à la création d'une reconstruction 3D de bonne qualité. Puis, nous chercherons à réaliser ces étapes et à simplifier la prise en main des logiciels. L'objectif final est d'obtenir le meilleur modèle 3D possible en comprenant et optimisant les paramètres qui influent sur la qualité d'un modèle 3D.

2.2.2. Liste des parties prenantes

Les parties prenantes sont, avec leurs rôles prévus :

Membres	Rôles
Lucia BERGANTIN	Professeur Référent
Anne-lise PELLETIER	Responsable organisationnel
Achile PINSARD	Responsable 3D
Pierre-Antoine HIGNARD	Responsable Gitlab
Samuel VATON	Responsable photos

TABLE 2 – Rôles des membres de l'équipe

2.2.3. Contexte d'utilisation du système

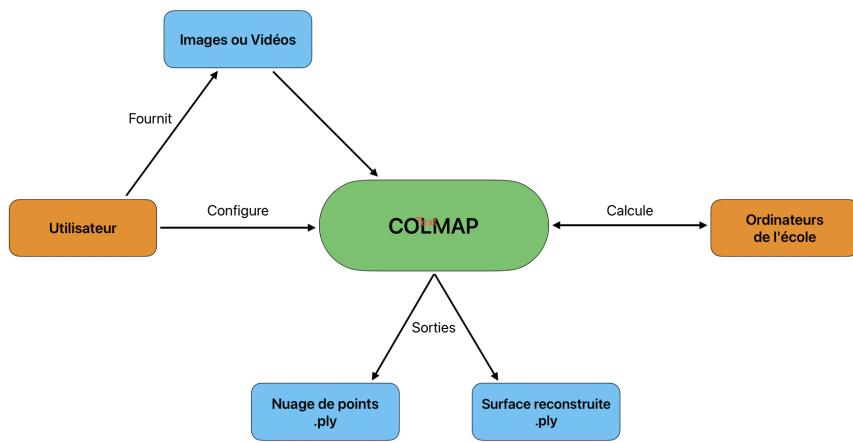


FIGURE 1 – Diagramme du système dans son environnement

La Figure 1 illustre le système COLMAP dans son environnement d'utilisation. Le schéma met en évidence les différentes interactions entre le logiciel, les utilisateurs et les ressources matérielles. Les flèches représentent les échanges de données et les étapes du processus : l'utilisateur configure COLMAP, et fournit les images sources. Le logiciel COLMAP prend avantage du CPU et de la carte graphique de l'ordinateur

pour effectuer les calculs nécessaires. Et finalement, COLMAP rend deux fichier .ply : le premier est le nuage de points obtenu, le second est la surface reconstruite à partir de ce nuage de points, mais il est souvent nécessaire de repartir du nuage de points pour avoir une reconstruction fiable.

2.3. Expression fonctionnelle du besoin

2.3.1. Fonctions de service et de contrainte

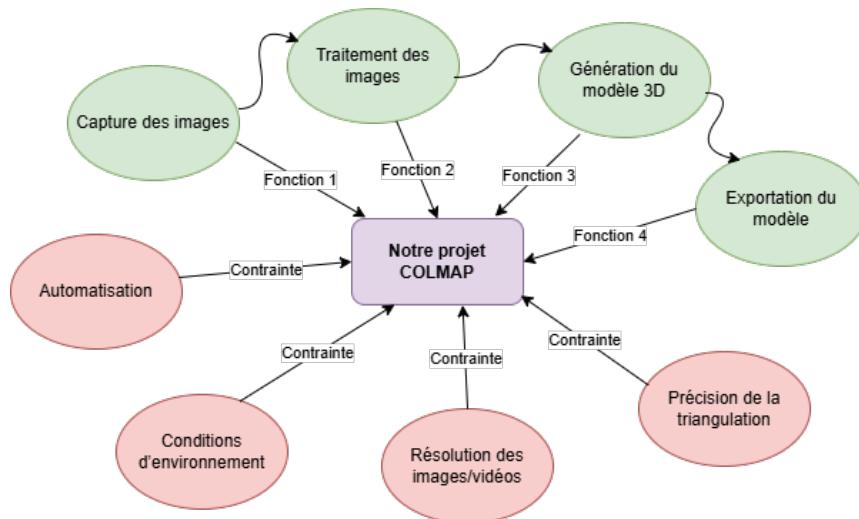


FIGURE 2 – Diagramme des fonctions principales de service et de contrainte

(a) En vert sont représentées les différentes fonction qui doivent être fait dans le projet finalisé.

(b) En rouge sont représentées les contraintes que nous rencontrons durant le projet.

Nous avons identifié **quatre fonctions principales** :

- la *capture des images*,
- le *traitement des images*,
- la *génération du modèle 3D*,
- l'*exportation du modèle*.

Ces étapes définissent la chaîne logique de notre projet, et sont schématisé sur la figure 2b.

En parallèle, plusieurs **contraintes techniques** ont été mises en évidence :

- l'*automatisation* du processus,
- les *conditions d'environnement* (lumière, stabilité, etc.),
- la *résolution des images et vidéos* utilisées,
- la *précision de la triangulation* pour obtenir un modèle fidèle.

Ce diagramme nous a aidé à visualiser clairement les interactions entre chaque fonction et les contraintes qui les accompagnent. Il a constitué un point essentiel pour planifier et anticiper les besoins techniques et les limites du projet COLMAP.

2.4. Scénarios

Le schéma ci-dessus présente les différents acteurs du système, leurs besoins, ainsi que les flux d'interaction entre eux. Il nous permet de décrire les scénarios nominaux d'utilisation du produit, tel qu'il a été conçu par notre équipe.

Le programme, **COLMAP optimisé pour des objets de petites tailles**, répond à trois attentes majeures du client :

- un *temps de modélisation réduit*,
- une *bonne qualité de modélisation*,
- et l'*usage d'un logiciel accessible*.

Le produit est destiné à des utilisateurs souhaitant réaliser une **modélisation 3D à partir d'images ou de vidéos**. Ces deux scénarios d'utilisation sont intégrés et fonctionnels :

1. **Scénario 1 (cas nominal)** : L'utilisateur fournit un ensemble d'images. Le système exécute la modélisation 3D via COLMAP, puis l'export au format STL est effectué pour impression 3D (cette dernière fonctionnalité reste hypothétique).
2. **Scénario 2 (cas nominal)** : L'utilisateur fournit une vidéo. Elle est convertie en images, puis traitée de la même manière que le scénario précédent.

Ces scénarios font intervenir plusieurs composants fournis par nos partenaires techniques : COLMAP, un ordinateur avec imprimante 3D, un convertisseur de vidéos en images et un outil d'export au format .stl.

Pour aller plus loin :

- Si les images sont de faible qualité ou floues, le modèle sera moins précis. Une alerte peut être intégrée pour recommander un nouveau jeu d'images.
- Si la vidéo est trop courte ou instable, le système peut refuser la modélisation et en demander une nouvelle.
- Si l'utilisateur ne suit pas les prérequis (format, nombre d'images, etc.), une documentation d'aide est prévue.

L'ensemble des fonctions prévues couvre à ce jour l'ensemble des cas d'usage identifiés, aussi bien en situation nominale qu'en cas d'anomalie. Ce travail s'inscrit dans un projet abouti et orienté vers l'utilisateur.

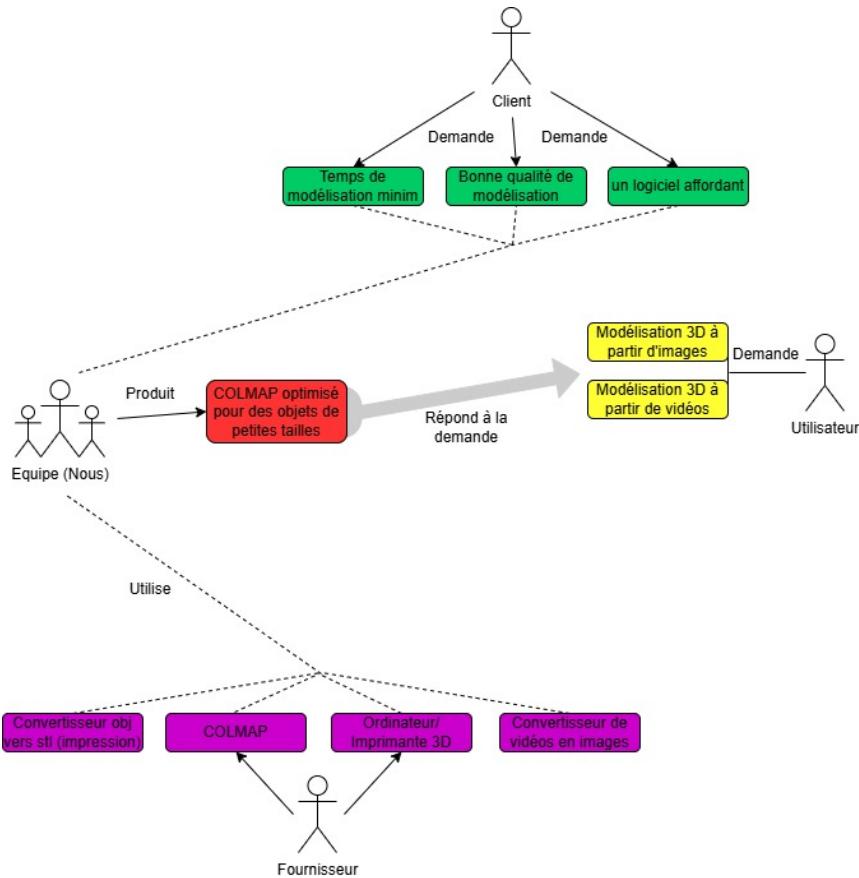


FIGURE 3 – Schéma des scénarios d'utilisation du produit et des interactions entre les acteurs, depuis la demande client jusqu'à la modélisation 3D.

Vert : représente les besoins exprimés par le client, comme la qualité, la rapidité et l'accessibilité du logiciel.

Jaune : indique les demandes spécifiques des utilisateurs, par exemple la modélisation 3D à partir d'images ou de vidéos.

Rouge : désigne le produit principal fourni par l'équipe, ici un COLMAP optimisé.

Violet : correspond aux outils techniques utilisés dans le processus (logiciels, matériel, convertisseurs, etc.).

2.4.1. Utilisation nominale

Le système de modélisation 3D avec COLMAP est utilisé pour transformer un ensemble d'images ou de vidéos en un modèle 3D exploitable, avec un temps de calcul inférieur à 30 minutes et une précision visuelle (on ne remarque pas visuellement trop d'erreurs). Le scénario d'utilisation classique est le suivant :



FIGURE 4 – Flow Chart d'utilisation du logiciel

- **Préparation de l'objet :** L'utilisateur prépare l'objet et l'arrière-plan pour que tout soit suffisamment éclairé et visible.
- **Acquisition des données :** L'utilisateur capture des images ou une vidéo sous différents angles d'un objet ou d'un environnement.
- **Importation des données dans COLMAP :** L'utilisateur charge les images dans le logiciel.
- **Traitement Structure-from-Motion (SfM) :** COLMAP reconstruit un nuage de points à partir des

images. Cette étape peut être très longue, mais pour des paramètres classiques : environ 60 images, qualité *high*, la reconstruction prends autour d'une heure sur un PC ayant une RTX A4500 et un CPU AMD EPYC 7543P.

- **Exportation** : Le modèle est exporté en format .ply

2.4.2. Cas limites et scénarios extrêmes

Certains scénarios impliquent des situations particulières qui doivent être anticipées :

- **Problèmes d'acquisition des images** :

- Flou, surexposition ou sous-exposition pouvant réduire la qualité du nuage de points.
- Mauvaise couverture de l'objet (angles insuffisants), entraînant des trous dans la reconstruction.

- **Problèmes de calcul** :

- Temps de calcul trop long (>2h) sur les ordinateurs à disposition, obligeant à réduire la résolution ou le nombre d'images.
- Erreur de reconstruction due à un nombre d'images insuffisant ou des conditions d'éclairage inadéquates.

- **Décalage des valeurs d'entrée** :

- Images mal alignées à cause d'une mauvaise calibration de la caméra.
- Variations importantes de luminosité entre les prises de vues perturbant le couplage des points.

- **Erreurs de l'utilisateur** :

- Chargement de fichiers non compatibles.
- Mauvaise configuration des paramètres de COLMAP (ex. : choix d'un algorithme inadéquat).

2.5. Validation du besoin

La validation du système se fera principalement à travers une approche comparative entre des modèles de référence et leurs reconstructions. Notre méthodologie de validation s'articule autour des points suivants :

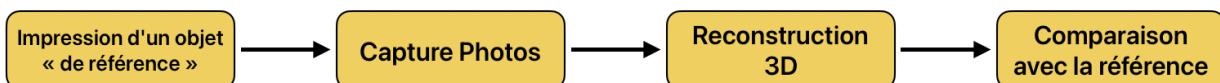


FIGURE 5 – Flow Chart de test des variables choisies

- **Protocole de test** :

- Utilisation de modèles 3D de référence issus de bibliothèques en ligne (nous avons notamment utilisé la chaise, que l'on peut considérer comme le fil rouge de ce projet)
- Impression 3D de ces modèles (en considérant l'erreur d'impression négligeable)
- Acquisition d'environ 60 photos par objet dans des conditions d'éclairage optimales (avec le fond 4.3). Ce nombre d'images a été choisi, car suite à nos expérimentations, l'intervalle 50-70 photos est un bon juste milieu entre qualité et temps de calcul.
- Reconstruction via COLMAP
- Comparaison visuelle, ou quantifiée (par la distance de Hausdorff par exemple) dans un logiciel de modélisation 3D

- **Critères d'évaluation** :

- Qualité visuelle globale du modèle reconstruit
- Fidélité géométrique par rapport au modèle de référence
- Respect des contraintes temporelles (2h maximum)

Cette phase de validation sera enrichie par l'étude de l'influence des différents paramètres de COLMAP sur la qualité de la reconstruction, en s'appuyant sur la documentation officielle [3] et les principes de la Structure-from-Motion [1].

2.6. Conclusion du Cahier des Charges

L'analyse des besoins a permis de définir un cadre clair pour notre projet de reconstruction 3D. Les prochaines étapes se concentreront sur :

- La réalisation d'un premier modèle avec les paramètres par défaut de COLMAP
- L'étude systématique de l'influence des variables de COLMAP sur la qualité de reconstruction
- L'optimisation des paramètres pour nos cas d'usage spécifiques

Le principal défi identifié reste la maîtrise approfondie de COLMAP et des principes de Structure-from-Motion. Bien que l'objectif de reconstruction en temps réel reste hypothétique à ce stade, nous nous concentrerons d'abord sur l'obtention de reconstructions précises et de qualité satisfaisante.

3. Conception des composants et du système

3.1. COLMAP

COLMAP est un logiciel open source qui permet à partir d'un ensemble de photos de reconstruire des objets en 3 dimensions sous la forme d'un nuage de points.

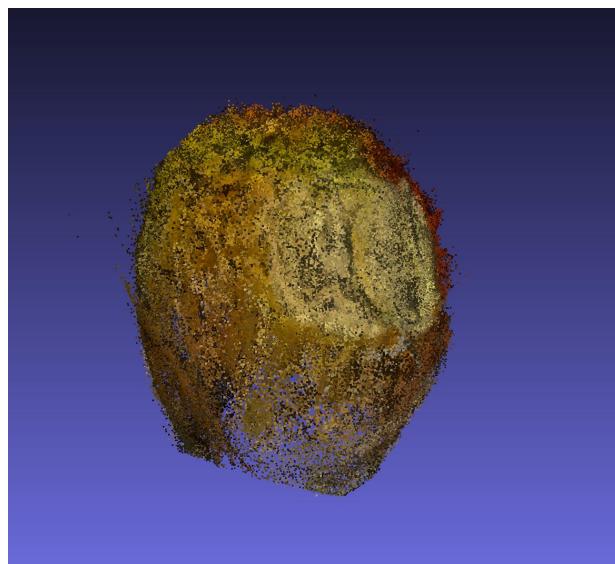


FIGURE 6 – Image d'un ensemble de points reconstruits par COLMAP à partir de photos d'une pomme

Pour ce faire le logiciel passe par plusieurs étapes. Afin de récréer un modèle il y a l'extraction de points clés sur les images. Ces points, dont les caractéristiques sont invariantes en fonction des images, sont liés entre eux puis triangulés pour former un nuage de points correspondant à l'objet photographié. Pour cela COLMAP utilise l'algorithme SIFT pour Scale Invariant Feature Transform. Nous allons détailler son fonctionnement.

3.1.1. Feature extraction

La **feature extraction** est l'étape qui consiste à extraire les points Clés des images. Pour une image donnée, l'algorithme SIFT va appliquer des effets de floue sur l'image et faire la différence des versions de floue pour repérer les points qui restent bien visible. Cela peut être les coins, les textures, les bords ou encore des motifs. Ces effets de floue se traduisent par un filtre gaussien à différentes intensités appliqué sur l'image. On prend alors les maximums ou minimums de ces différences par petites zones de l'image. Ce sont nos points clés.

Une fois trouvé, on calcul le gradient de notre point ce qui nous donne une direction et une intensité de variation lumineuse. Cette direction est très importante car elle est invariante par changements géométriques

donc l'orientation de notre point sera la même sur les autres images. Ces informations sont reliées au point dans un vecteur appelé descripteur qui agit comme l'empreinte de notre point. On reproduit cette opération sur toutes les images pour obtenir de nombreux points clés à comparer.

3.1.2. Feature matching

Une fois un certains nombre de points clés définis par images, l'étape cruciale est de faire correspondre les points trouvés de chaque images entre eux pour trouver des similarités dans les images. C'est le **feature matching**. Les images sont comparées 2 à 2 pour liés des points de chaque images qui décrivent la même partie de objet.

Pour chaque point clé d'une image, on cherche son plus proche voisin sur l'autre image en respectant un certain seuil pour éviter des matchings incorrectes. C'est à dire qu'on défini un seuil pour le rapport des distances entre le point de la première image et ses plus proches voisins. En effet, plus ce rapport sera proche de 1 et plus le choix du plus proche voisin sera ambiguë. Ainsi, si le seuil est dépassé on ne choisit pas ces points et on recommence avec un autre point de la première image. Cela peut donc arriver que des points ne correspondent pas mais ce n'est pas un très gros problème au vu du nombre de points clés généralement présents sur les images.

Une fois nos plus proches voisins trouvés pour tous les points ou presque, le matching est terminé et on possède une liste de correspondance de points.

Il existe plusieurs feature matching, par exemple on peut faire une comparaison de toutes les images entre elles c'est le **exhaustive matching**. On peut aussi suivre l'ordre donné des images ce qui aide si on les images sont tirées d'une vidéo. C'est le **sequential matching**.

3.1.3. Reconstruction

Pour finir la reconstruction il y a encore de nombreuses étapes qui demandent une grande capacité de calculs.

Nous allons procéder en plusieurs temps : tout d'abord, il faut retrouver la position des caméras des 2 premières images en utilisant la liste de correspondance des points clés entre ces deux images. Le calcul se fait avec une équation matricielle que nous ne pensons pas nécessaire de détailler ici contenant une matrice inconnue et les coordonnées des points clés associés de chaque images sous forme de vecteur. Cela va nous permettre de retrouver la position de la deuxième caméra à partir d'une rotation et d'une translation de la première.

Une fois l'obtention de 2 caméras, COLMAP utilise l'algorithme d'optimisation PnP (Perspective-n-Point) pour déduire les positions des caméras de chaque images ajoutées au fur et à mesure.

Enfin vient l'étape de triangulation dans COLMAP qui va permettre de créer un nuage de points à partir de toutes les étapes précédentes. Pour ce faire, pour chaque point clés présents dans un certains nombre d'images, il va tracer des trayons de chaque caméra vers ces points pour reproduire leur positionnement dans l'espace en croisant les rayons. Il choisit ensuite la meilleure position 3D pour minimiser l'erreur de projection.

On obtient ainsi un nuage de point représentant l'objet pris en photo au départ.

3.2. Matériel informatique

Pour le fonctionnement du projet, nous avons principalement besoin d'ordinateurs avec une puissance de calcul suffisante pour pouvoir effectuer les reconstructions avec COLMAP.

3.2.1. Difficultés rencontrées

Nous observons systématiquement des erreurs de lancement et des crashes de la part de COLMAP quand nous essayons de lancer une reconstruction automatique. Cela signifie que nos ordinateurs n'ont pas les ressources nécessaires pour faire des rendus avec COLMAP et le fait de baisser la qualité ou de ne pas utiliser de GPU (ce qui est possible) produit des résultats inexploitables. Pour pouvoir effectuer des rendus de la meilleure qualité possible, il nous faut de bonnes cartes graphiques et des processeurs robustes afin de pouvoir les compléter.

Nous avons alors décidé avec l'aide de la DISI d'utiliser des PC mis à notre disposition dans une salle d'un des bâtiments de l'établissement. Cependant, les PC fonctionnaient sur Linux et nous n'avions pas les droits suffisants pour télécharger COLMAP et donc pouvoir lancer de reconstruction.

Nous avons alors choisi d'opter pour des outils de virtualisation fournis par l'école afin de pouvoir faire les calculs. Pourtant, les logiciel refusaient de s'installer malgré tous nos efforts, rendant cette solution inenvisageable. Nous nous sommes donc retrouvé bloqués pendant un certains temps sans pouvoir réellement avancer le projet.

3.2.2. Solutions

Les solutions mises en œuvre pour pouvoir avancer ont été l'utilisation du compte Shadow de Pierre-Antoine, un outil en ligne qui fournit des ordinateurs puissants à distance moyennant une bonne connexion internet, et l'obtention d'un PC performant du fablab.

Ainsi nous avons pu nous organiser afin d'effectuer des reconstructions d'objets en 3D.

3.3. Meshlab

Meshlab est un logiciel open source pour le traitement et l'édition de maillages triangulaires 3D. Ce sont des modèles numériques de surface qui représente numériquement des objets en 3D. Le modèle se compose de sommets, d'arêtes et de polygones.

Les sommets sont utilisés comme coordonnées et les arêtes du modèle relient respectivement deux sommets voisins. Les faces principalement utilisées ici sont des triangles, qui englobent les arêtes et forment ainsi la surface de l'objet.

Meshlab fournit un ensemble d'outils pour l'édition, le nettoyage, la réparation, l'inspection, le rendu, la texturation et la conversion des maillages. Il offre des fonctionnalités pour le traitement des données brutes produites par les outils de numérisation 3D et pour la préparation des modèles pour l'impression 3D.

Il permet d'obtenir une surface à partir d'un nuage de points fourni par exemple par COLMAP. Cette surface peut alors être nettoyée, ce qui correspond à enlever les imperfections provoquées par des erreurs de reconstruction dans le nuage de points. Elle est ensuite utilisée comme fondement de l'impression 3D.

3.4. Fusion 360

Fusion 360 est un logiciel de type CAD pour Computer Asisted Design. Il permet de transformer une surface en volume prêt à l'impression 3D.

3.5. Modèle 3D de la chaise

Pour représenter notre projet nous avons décidé de partir d'un objet comme référence. Un modèle 3D de chaise trouvé sur internet pour nous exercer.

L'objectif était le suivant :

- imprimer la chaise en 3D à l'aide d'une imprimante 3D,
- prendre des photos de la chaise,
- reconstruire un nuage de point de la chaise sur COLMAP,

- transformer la chaise en surface avec Meshlab,
- transformer la chaise en volume avec le logiciel fusion 360,
- comparer le modèle de départ et le modèle recréé sur Meshlab.

Tout cela pour nous permettre de mettre en oeuvre le processus de reconstruction dans son intégralité et de la faire sur un objet relativement difficile à modéliser dans un premier temps à cause de son nombre important d'ouvertures.

3.6. Imprimante 3D

Nous avons utilisé une imprimante 3D présente au fablab pour imprimer le modèle de la chaise. Pour cela nous avons bénéficié d'une formation sur les imprimantes 3D pour prendre en mains l'impression. Le modèle des imprimantes sont une Ultimaker 2+(pour la chaise initiale) et une créality K1 (pour la chaise imprimé après modélisation COLMAP).

Pour l'impression du modèle de la chaise nous avons utilisé du PLA comme matériau. De plus pour diminuer le temps d'impression qui pouvait être très conséquent nous avons décidé de rendre la chaise le plus petit possible mais sans la rendre inutilisable. Elle mesure environ 5 cm de hauteur

Enfin lors de l'impression, nous avons séparé le modèle en deux au niveau de l'assise afin de réduire encore le temps d'impression.

Cela nous a laissé avec un temps d'impression d'environ deux heures ce qui nous semblait acceptable.

Une fois terminé et recollé, la chaise ressemblait à cela :



FIGURE 7 – Image de la chaise imprimée en 3D

4. Développement

Nous avons décidé de créer plusieurs programme python pour rendre la création des modèles automatique. Pour ce faire nous avons donc plusieurs programme à notre disposition.

D'un côté, un programme transforme les photos en vidéos. De l'autre, un programme traite les images puis lance une reconstruction. Nous avons séparé ces programmes car ils ont des utilisations différentes et nous n'avons pas besoin qu'ils fonctionnent ensemble.

4.1. Code python transformation d'une vidéo en images

Pour le premier code, nous avons utilisé la librairie OpenCV [6] de python qui permet de traiter et manipuler des images ou des vidéos. Le but du programme est d'extraire des images à partir du vidéo. En effet, le logiciel COLMAP ne prend que des images en entrée mais il nous semblait tout de même important de pouvoir créer des modèles 3D à partir de vidéos des objets.

Ce processus fonctionne car les vidéos sont constitué de frames, des images qui mises bout à bout permettent de reconstituer le réel. Notre programme consiste à extraire ces frames des vidéos. Elles sont choisis en divisant le nombre de frames de la vidéo par le nombre d'images voulues.

Plus précisément, nous ouvrons la vidéo avec la fonction `Video_Capture()` qui prend en entrée le chemin de la vidéo et permet de lire la vidéo. Ensuite on défini un intervalle de frames correspondant au rapport entre le nombre total de frames et le nombre d'images souhaitées. A chaque intervalle de frames, on enregistre l'image dans un dossier créé avec `imwrite()` qui prend en entrée l'image en question et le chemin vers le dossier. A la fin de ce programme, nous avons le nombre d'images souhaitées tirées de notre vidéo.

4.2. Code python pour traitement d'image et reconstruction

4.2.1. Traitement d'image

Au cours de nos différents tests et recherches nous nous sommes rendus compte de l'importance du traitement des images à envoyer dans COLMAP. En effet, il semble que la qualité des images de départ à plus d'importance que les paramètres de COLMAP pour optimiser le résultat. C'est pourquoi nous avons choisi d'effectuer un traitement d'images automatique avec la librairie OpenCV de python.

Pour ce faire, cette partie du programme utilise 2 fonctions.

□ La première, `process_images()` prend le chemin vers une image à traiter et le chemin souhaité pour enregistrer l'image traité. Le but de cette fonction est d'effectuer le traitement d'une image.

Plus précisément, la fonction permet de :

1) Redimensionner l'image avec `resize()` qui prend en entrée l'image et le couple de dimensionnement ici (1600, 1200) . Le couple est proche de la résolution des images que nous avons capturées ce qui permet d'équilibrer la dimension de toutes les images pour éviter des changements trop importants.

2) Améliorer le contraste grâce à une méthode appelée CLAHE. Cette technique permet de rendre les détails plus visibles, surtout dans les zones sombres ou peu nettes.

Concrètement, on change le mode de couleur de l'image avec `cvtColor()` qui passe de RGB en mode LAB, c'est à dire diviser l'image en sa luminosité L, en les composantes de ces couleurs A et B. D'ici, on crée un objet CLAHE avec `createCLAHE()` qui prend en entrée deux paramètre : `clipLimit` qui limite la valeur du contraste, et la taille de carrés choisis pour diviser l'images et ajuster localement la répartition des zones claires et sombres.

On applique alors l'objet à notre L qui nous donne L_{modif} puis on le réinsert dans notre image avec `merge()` qui prend en entrée les paramètres du mode LAB (L_{modif} , A et B) et permet de reconstituer notre image avec un nouveau contraste.

Grâce à ce traitement, l'image devient plus équilibrée : les détails ressortent mieux, sans que les couleurs soient déformées.

3) Améliorer la netteté avec un floue gaussien. On applique la fonction `gaussianBlur()` qui prend notre image en entrée ainsi que la taille de la zone de floue à appliquer autour de chaque pixel. Le but est de créer une version "adoucie" de l'image qui servira à isoler les contours.

Ensuite on applique la fonction `addWeighted()` qui va combiner l'image floutée avec l'image de départ

avec des poids différents. On soustrait l'image floutée à la principale pour faire ressortir les contours et donc avoir une image plus nette.

4) Supprimer le bruit (pixels indésirables) de l'image grâce à la fonction `fastNlMeansDenoisingColored()`, qui réduit le bruit tout en conservant les détails importants. Cette méthode compare chaque petite zone de l'image à d'autres zones similaires dans un voisinage plus large. Si elle trouve des motifs proches, elle les utilise pour recalculer la valeur du pixel, ce qui permet d'adoucir les irrégularités dues au bruit sans flouter les contours.

Le résultat est une image visuellement plus propre et plus agréable, notamment dans les zones uniformes comme les fonds ou les surfaces lisses.

Notre image est maintenant traitée, il nous faut recommencer avec chaque images du dossier de départ.

□ La deuxième fonction, `process_folder()` prend en entrée le dossier contenant les images à traiter et le chemin vers un potentiel dossier des images traités.

Cette fonction parcourt le dossier des images et applique la fonction `process_images()` pour traiter chaque images. Elle crée le dossier de sortie au chemin indiqué et y place toutes les photos traités.

4.2.2. Reconstruction

Enfin pour achever la reconstruction d'un modèle une fois les photos traités, nous utilisons le logiciel COLMAP à l'aide de sa librairie python associée `pycolmap` [7]. Pour ce faire le programme effectue le processus de reconstruction tel que décrit dans la section COLMAP avec différentes fonctions.

Le programme utilise le dossier contenant les images de l'élément à reconstruire et effectue les différentes étapes de la reconstruction. Le **feature extraction**, le **feature matching** et enfin la reconstruction.

4.3. Plateforme pour prise de photos

Dans notre démarche visant à essayer le plus d'approches possibles, et en nous inspirant des installations utilisées par des personnes plus expérimentées , nous avons essayé de faire un fond blanc à l'image afin de limiter le nombre de features détectées en dehors de l'objet recherché, sans nous attarder avec des CCtags (qui ne fonctionnent qu'avec Meshroom, mais qui auraient permis de mieux ancrer le modèle sur un plateau tournant et de fixer la taille réelle du modèle...). Cela nous permettrait ainsi de pouvoir déjà limiter les calculs inutiles de modélisation de l'arrière plan, mais surtout de pouvoir **retourner l'objet en question à l'envers** sans qu'il y ait des conflits entre la position relative de l'objet par rapport au fond.

La plateforme est de construction assez simple, avec deux calendriers en carton attachés l'un à l'autre en angle droit *fig. 8*. Puis nous avons recouvert les surfaces visibles de plusieurs couches de papier, et en arrondissant les angles afin de rendre le fond le plus uniforme possible *fig. 9*.



FIGURE 8 – Vue d'ensemble du fond



FIGURE 9 – Point de vue de la caméra

5. Intégration des composants et validation

Dans cette partie nous allons évoquer les différentes étapes à suivre pour la reconstruction d'un modèle complet et les choix que nous avons fait concernant l'utilisation des logiciels impliqués.

5.1. Étapes de la reconstruction

Les étapes de reconstruction d'un modèle 3D à partir de photos sont les suivantes :

5.1.1. Prise de photos ou vidéos

La prise de photos, bien qu'elle puisse paraître assez simple, est la partie la plus difficile à prendre en main. En effet, les parties suivantes ne requièrent que des automatismes, alors que celle-ci parvient encore parfois à nous échapper. Cependant, nous avons fait une liste de points qui sont importants à suivre :

- Éviter à tout prix la lumière du soleil en intérieur, car elle ne sera donc pas uniforme (venant de la fenêtre) et très puissante. Dehors cela ne pose pas trop de problèmes.
- Garder les métadonnées de la photo, de nombreux paramètres comme l'ouverture sont essentiels à la reconstruction. En l'absence de, COLMAP utilisera des valeurs par défaut, ce qui posera de nombreux problèmes. Aussi, ne jamais recadrer l'image : si des parties sont dérangeantes, il faut absolument utiliser la fonction de **mask**.
- Le "coup de main" prend plusieurs essais à trouver, mais il est essentiel de tourner autour de l'objet ; avoir différents angles, non seulement autour de l'objet, mais également du point de vue de la caméra ; et d'alterner entre un point de vue proche et éloigné. L'objectif n'est pas forcément de prendre une grande quantité d'images (<100), mais d'en prendre à partir du plus d'endroits possibles : c'est la profondeur qui fait fonctionner l'algorithme.

Une prise correcte donne les emplacements de caméra (en rouge, pour un objet face à un mur) visibles sur les images 10 et 11.

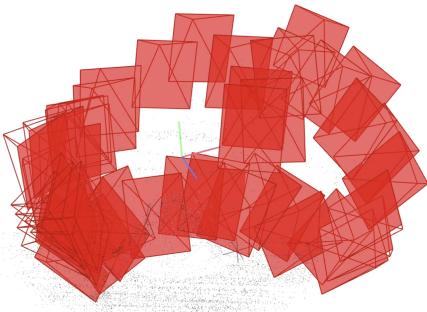


FIGURE 10 – Caméras du point de vue de l'objet

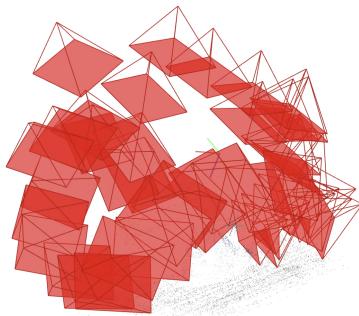


FIGURE 11 – Caméras du point de vue extérieur

Notre protocole a été de prendre avec un de nos téléphones, une soixantaine de photos en suivant les précédentes recommandations. Puis, on importe ces images sur une solution de Drive présente à la fois sur les téléphones et l'ordinateur de calcul, ce qui permet de faire le transfert facilement. Enfin, dans le cas de figure où nous avions choisi de prendre une vidéo, nous utilisons le programme Python `video_to_images()` présenté précédemment pour en extraire les images voulues.

5.1.2. Reconstruction automatique avec COLMAP

Sur COLMAP, nous utilisons l'onglet reconstruction automatique qui nous fournit un nuage de points grâce aux méthodes décrites précédemment.

Une précision est à apporter sur le choix d'utiliser COLMAP en mode reconstruction automatique. En effet, il est possible de modifier de nombreux paramètres sur COLMAP pour la reconstruction que ce soit le type de feature extraction, de feature matching, ou de nombreux paramètres permettant de modifier la qualité du rendu en dépendance d'une certaine puissance de calcul.

Premièrement nous n'avons pas trouvé d'informations concernant l'exploitation de ces paramètres par d'autres personnes. De plus la modification de certains paramètres ne nous a pas semblé avoir une réelle incidence sur les résultats.

Nous avons préféré nous occuper de l'amélioration de notre prise d'images ou bien du traitement en amont de celles-ci.

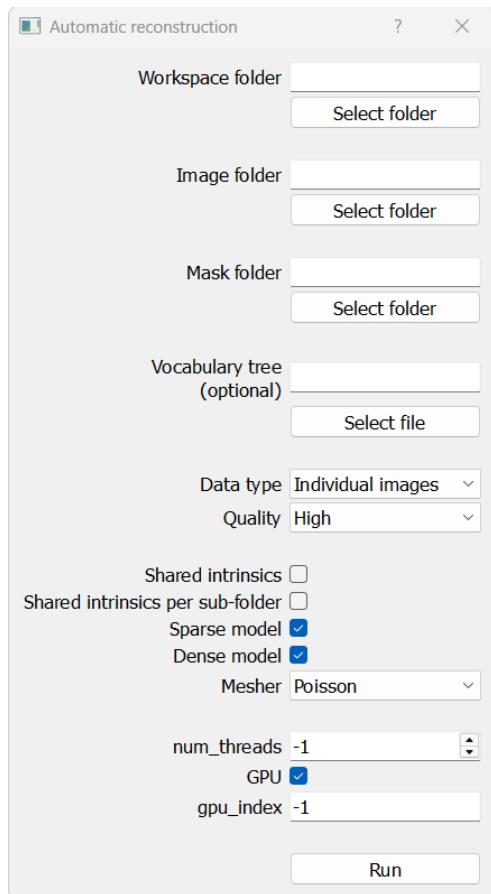


FIGURE 12 – Capture d'écran de la fenêtre de reconstruction automatique de COLMAP

Nous avons cependant choisi la qualité maximale de rendu possible dans la fenêtre de reconstruction automatique et l'utilisation de GPU qui nous permet aussi d'obtenir de meilleurs résultats.

5.1.3. Nettoyage du nuage de points avec Meshlab

Le nuage de points créé par COLMAP est rarement parfait, nécessitant toujours de faire des ajustements avant de pouvoir passer à l'étape suivante de la création de surface.

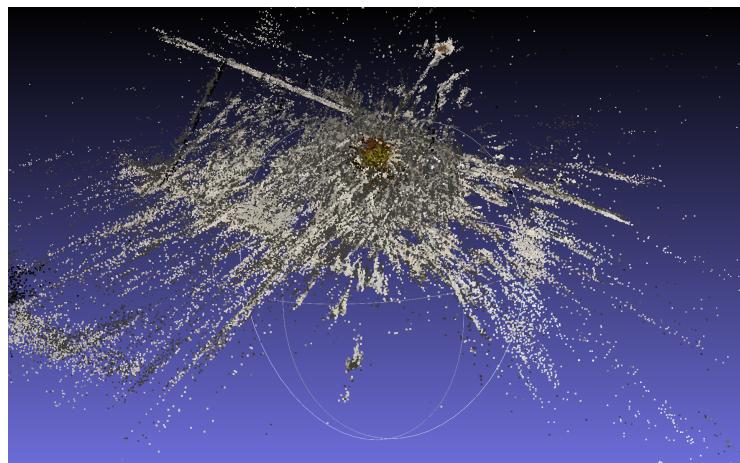


FIGURE 13 – Problème de points "fantômes"



FIGURE 14 – Problème de points en bordure

Problème 1 : Points "fantômes" éloignés (e.g. fig. 13)

Meshlab offre une solution simple pour cela : **Select vertices** sélectionne tous les points se situant dans la zone indiquée. Il suffit ensuite de les supprimer avec **Delete Selected Vertices**.

Problème 2 : Surface colorée du modèle (e.g. fig. 14)

Meshlab permet une multitudes d'autres moyens de sélectionner des points de manière plus avancée, comme le **Select faces by color** qui est très bien adapté à ce type d'erreur. Il suffit de garder les paramètres à 0.2 de proximité de la couleur d'un point, pour bien sélectionner les points non voulus et les supprimer par la suite.

Nous n'avons pas complètement compris ce qui mène à obtenir ces points ajoutés, mais il s'agit possiblement de COLMAP qui voit constamment un point blanc (qui vient du fond) à proximité des features qu'il reconnaît, donc pense qu'il y a quelque chose de blanc autour du modèle.

5.1.4. Création d'une surface avec Meshlab

Afin de recréer des surfaces, il existe une multitudes d'algorithmes, dont 3 différents sont disponibles sur Meshlab. Parmi eux, la reconstruction de Poisson "filtrée" (**Screened Poisson reconstruction**) est celle que nous avons le plus utilisée. Les principaux avantages que cette solution apporte sont :

- Facilité de choix entre la précision et l'adoucissement des erreurs, le poids de l'interpolation, ou "*interpolation weight*". C'est de cela que vient le "screened" du nom de cet algorithme, qui est un ajout permettant d'adoucir les imperfections, au coût d'un rendu plus "flou".
- Forte capacité à boucher les trous, à obtenir des modèles *watertight* (essentiel pour le 5.1.5).

Cependant, le dernier point est à nuancer, car cet algorithme a tendance à remplir des trous là où ce n'est absolument pas nécessaire. Cela débouche souvent en la création d'une "jupe", notamment dans le cas de la chaise, où beaucoup de détails du bas de la chaise ont été négligés.

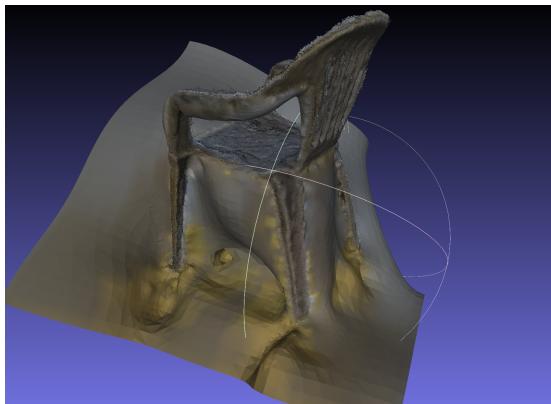


FIGURE 15 – Problème de "jupe" avec la reconstruction de surface Poisson

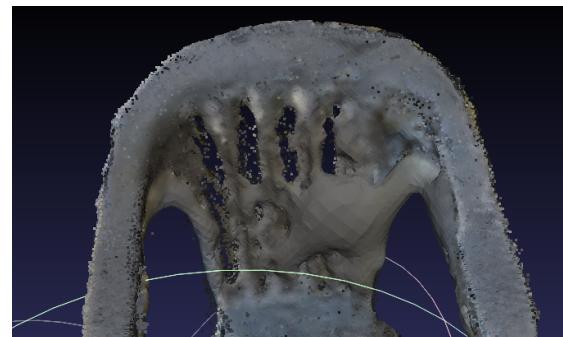


FIGURE 16 – Autre problème pour des objets contenant des trous

C'est pour cela que nous avons également expérimenté avec d'autres algorithmes. Cependant, nous n'avons pas réussi à utiliser l'algorithme VCG, et celui de **Ball Pivoting** était loin d'être idéal : comme on peut le voir dans la figure 17, il reste beaucoup de trous, la surface est assez rugueuse car constituée uniquement de triangles, et, peut-être moins important en fonction de l'application, mais les couleurs ont disparues.

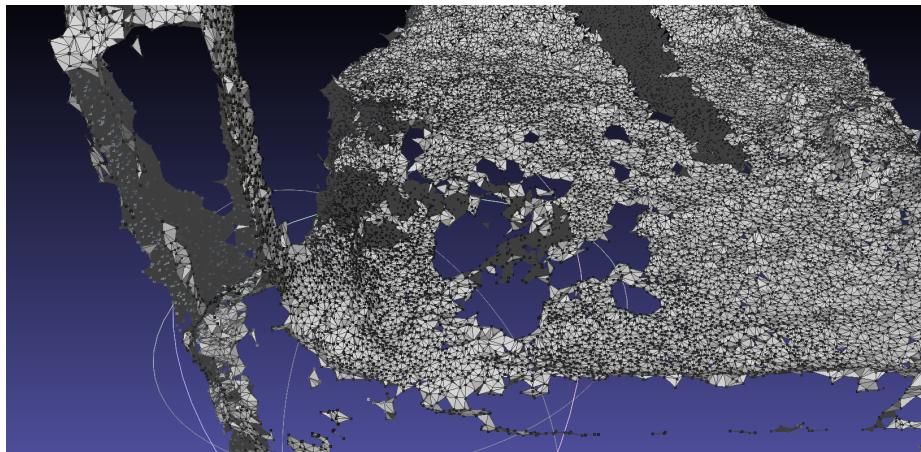


FIGURE 17 – Reconstruction de la chaise par Ball Pivoting

C'est ainsi que, malgré les quelques problèmes que comporte **Screened Poisson**, nous n'avons utilisé que cet algorithme dans nos autres expérimentations.

5.1.5. Transformation de la surface en volume avec Meshlab et Fusion 360

Adapter la surface dans Meshlab

Pour convertir une surface en volume, il est nécessaire de commencer par une surface fermée, ou *watertight*.

Cela est souvent le cas sur la plupart de la surface avec le mesher Poisson, mais il reste toujours le problème de "jupe" de la figure 15. Afin de résoudre ce genre de problèmes, Meshlab permet de sélectionner des surfaces et de les supprimer. Il reste donc à combler le dernier trou. Meshlab permet de faire cela à travers de nombreuses techniques, mais la plus simple à essayer en premier est le bien nommé **Close holes** fig. 18, qui peut à la fois remplir plein de petits trous, ou peut être aidé en sélectionnant la bordure du trou qu'on cherche à fermer, et en cochant l'option "*Close holes with selected faces*".

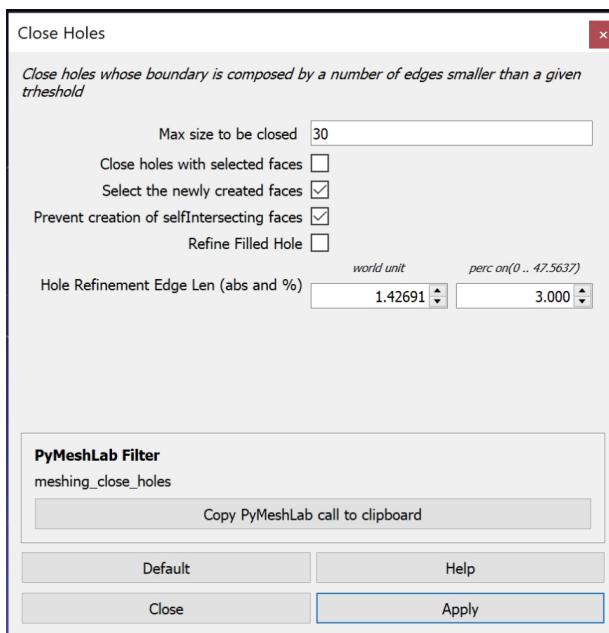


FIGURE 18 – Page Close Holes

Sinon, il est possible de refaire appel au mesher *Poisson*, car il est également applicable aux surfaces. Cependant, il est toujours possible de faire face au même problème qu'à l'origine, il est donc parfois nécessaire d'ajouter manuellement une partie de la surface et ensuite réessayer à partir de là. Mais, dans ce cas de figure, il faut envisager de refaire le modèle depuis 0, ou d'ajouter d'autres images, car le rendu ressemblera plus à la modélisation à la main qu'à la véritable reconstruction 3D par SfM.

Toutefois, il est nécessaire de se rappeler que le fait de remplir les trous se base sur peu ou pas du tout d'information, et sera grossier, avec une approximation des couleurs (voire pas du tout).

La procédure, une fois la surface fermée obtenue, devient très simple, il suffit de rendre les normales de la surface fermée dans le bon sens **Re-Orient all faces coherently** (bien que ça ne soit souvent pas nécessaire). Ensuite, diminuer le nombre de faces à environ 10.000, car le calcul de surface commence à être trop compliqué (et non nécessaire, car la précision de l'imprimante n'est pas à ce niveau), et effectuer des nettoyages comme **Repair non-manifold edges**, car Poisson a tendance à faire quelques surfaces non conformes, qui s'auto-intersectent....

Convertir en volume

Une fois cela fait, il faut exporter la surface en **.obj**, et l'importer dans un logiciel de CAD (cf. tableau 1) comme Fusion 360.

Ensuite, il faut faire une dernière étape de préparation du modèle : **Generate Face Groups**, qui d'après nos sources [8], est sensé aider Fusion à effectuer ses calculs par la suite.

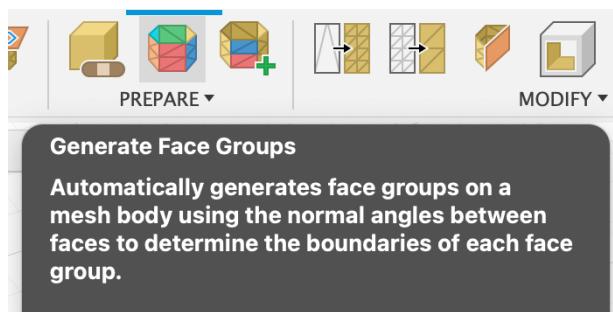


FIGURE 19 – Boutons de préparation : nettoyage, groupes de faces...

Finalement, après tout ce processus, il est possible de faire la conversion en volume avec la fonction **Convert Mesh**, avec les paramètres "*Base Feature*" et "*Faceted*" :

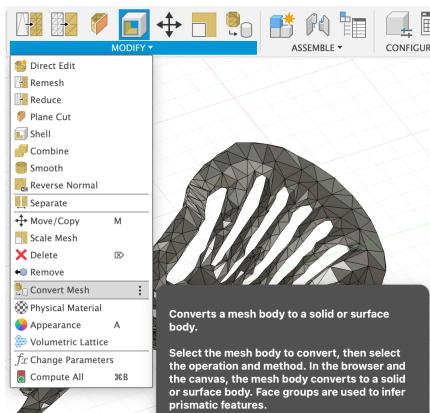


FIGURE 20 – Bouton Convert Mesh

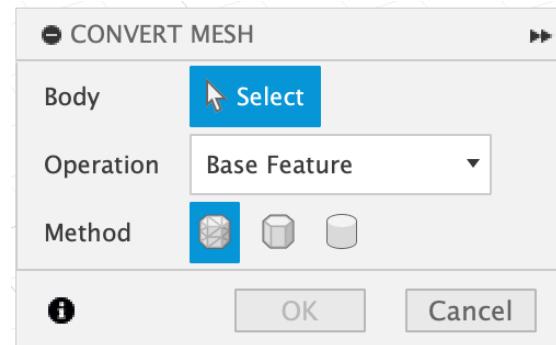


FIGURE 21 – Paramètres de conversion choisis

Nous avons enfin un volume, il ne reste plus qu'à l'exporter en **.stl**. Les options d'exportation choisies sont "*3D Print*", puis *Format = "STL (ASCII)"* et *Unit Type = "Centimeter"*.

5.2. Comparaison rendu photo ou vidéo

Pour tester la pertinence du découpage de vidéo nous avons tenté de réaliser un modèle (surtout un nuage de points) avec COLMAP à partir d'un côté d'un ensemble de photos prises une par une et de l'autre par des photos extraites d'une vidéo.

Nous obtenons le nuage de points suivant :



FIGURE 22 – Nuage de points de la colle avec les photos prises une par une

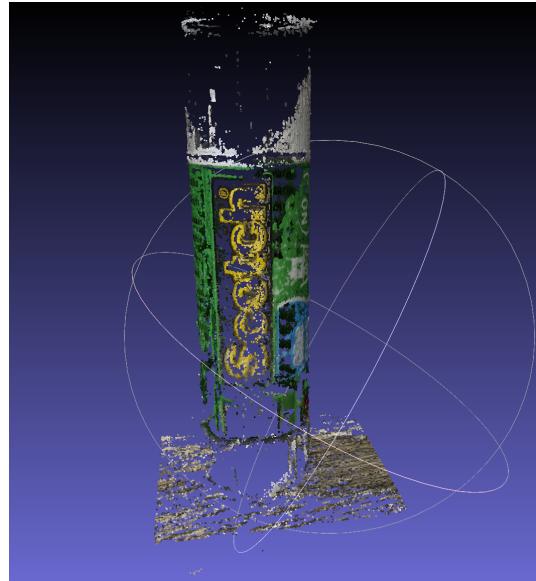


FIGURE 23 – Nuage de points de la colle avec les photos tirées de la vidéo

et pour la surface reconstruite :



FIGURE 24 – Surface reconstruite à partir du nuage correspondant aux photos prises une par une

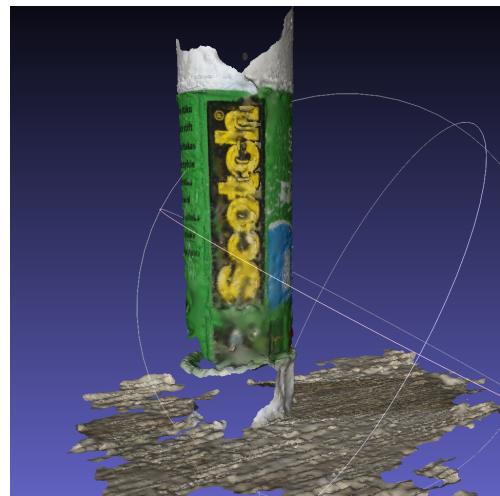


FIGURE 25 – Surface reconstruite à partir du nuage correspondant aux photos tirées de la vidéo

Nous remarquons que les résultats obtenus pour le reconstruction faites avec les images tirées de la

vidéo sont moins propres que ceux de la reconstruction à partir des images prises une par une.

Notre première hypothèse était que comme le programme ne choisissait que des images à intervalles régulières, il était très probable de retrouver parmi celles sélectionnées des images floues. Ainsi, nous avons créé une seconde itération de ce programme, `video_to_images_quality.py`, utilisant le gradient du Laplacien pour déterminer l'image la moins floue sur chacune des n sections de la vidéo. Le résultat n'a cependant pas été concluant, les images individuelle donnent toujours un meilleur rendu.

Cela est probablement dû au simple fait que le processus de création de la vidéo implique du mouvement, qui ensuite crée un flou sur la quasi-totalité des images qu'elle contient.

Nous décidons donc de prendre uniquement des photos pour faire nos rendus d'objets. Cela n'est pas nécessairement plus complexe car dans notre cas, le traitement par COLMAP ne nécessite pas un très grand nombre d'images (au maximum une centaine).

5.3. Comparaison des résultats entre images traitées et non traitées

Nous objectif dans cette partie était d'utiliser les programmes python pour le traitement d'image et la reconstruction afin de comparer le résultat des rendus avec des images traités et des images brut. Cependant, par manque de temps nous n'avons pas pu effectuer ces test jusqu'à la reconstruction avec COLMAP.

Nous allons néanmoins montrer les résultats du traitement des images à travers 2 objets.

Le premier est la chaise où nous avons appliqué la partie du programme python associé au traitement et cela nous donne le résultat suivant :



FIGURE 26 – Image de la chaise avant traitement



FIGURE 27 – Image de la chaise après traitement

Nous observons qu'il y a plus de détail autour de la chaise traitée. La table sur laquelle elle est posée semble mieux éclairée et permet de mettre la chaise en valeur. Les ombres ressortent plus et la luminosité est moins homogène ce qui rend l'image la chaise plus distincte. Cependant, l'assise de la chaise est plus floue que sur la chaise non traité.

Nous l'avons aussi appliqué à une colle :



FIGURE 28 – Image de la colle avant traitement



FIGURE 29 – Image de la colle après traitement

On remarque directement que l'image traité est plus aplatie ce qui est normal car elle est redimensionnée. Cependant, cela ne semble pas pertinent dans ce cas, il faudrait alors changer la dimension pour une meilleure reconstruction. Nous remarquons aussi la différence entre les bordures de la colle traité et de la colle non traité. Il y a comme un floue qui accentue la délimitation de la colle traité.

Ces deux exemples nous font comprendre l'importance des paramètres à utiliser pour le traitement des images et l'impact du programme sur la qualité de celles-ci et donc sur la qualité du résultat final. Nous avons manqué de temps pour exploiter pleinement cette possibilité afin de pouvoir optimiser les paramètres de traitement par rapport aux images choisies. Nous n'avons pas pu entamer de reconstruction avec les 2 types d'images pour pouvoir comparer les modèles 3D et justifier de la pertinence du traitement d'images pour la reconstruction en 3D.

6. Clôture du projet

Nous allons montrer nos tests et les résultats obtenus sur la chaise.

6.1. Création du nuage points avec COLMAP

Tout d'abord, nous commençons par prendre notre chaise en photo c'est un processus que nous avons pu essayer plusieurs fois, mais le cas présent a été fait avec le fond (cf. 4.3) présenté précédemment.

Nous entrons le dossier des photos dans COLMAP puis lançons la reconstruction automatique avec GPU en mettant la qualité High.

Nous obtenons le nuage de points suivant :



FIGURE 30 – Premier nuage de points de la chaise

Nous remarquons qu'il y a de nombreux points blanc autour de la chaise. Cela correspond à des points de la plateforme sur laquelle nous avons pris les photos. Mais le modèle dans sa globalité est très propre et nécessite peu de traitement. De plus nous pouvons obtenir un meilleur nuage grâce à Meshlab.

6.2. Transformation en surface avec Meshlab

Pour continuer, nous chargeons le nuage de points dans le logiciel Meshlab. Pour cela il faut trouver un fichier **.ply** qui correspond au nuage.

Pour effectuer la transformation en surface nous traitons d'abord le nuage de point avec le filtre **select faces by color** qui nous permet de retirer la partie blanche autour de la chaise.
Ce qui nous donne le nuage suivant :



FIGURE 31 – Nuage de points de la chaise sans points blancs

Il faut généralement retirer des points inutiles autour des modèles pour que la reconstruction fonctionne bien. Pour ce modèle, il y en avait très peu, c'est pourquoi nous avons choisi de le garder malgré le trou observé dans l'assise.

Maintenant utilisons Meshlab pour effectuer la reconstruction en surface comme décrite dans la section 5.1.4

Cela nous donne la surface suivante :



FIGURE 32 – Surface de la chaise reconstruite avec Meshlab

Bien qu'il y ait un trou conséquent dans l'assise de la chaise, nous choisissons de poursuivre les étapes de reconstruction avec ce modèle car il est le plus prometteur jusqu'ici.

Nous avons quelques précisions à apporter sur la modélisation globale de la chaise.

La chaise étant notre objet de modélisation "fil rouge", nous avons constaté dès nos premières modélisations que, peu importe la qualité de la modélisation globale, certains éléments du modèle étaient mal restitués, ce qui nous permet de comparer nos différents processus et d'évaluer nos avancées. En effet, la droite de l'assise ainsi que le pied arrière gauche (référentiel d'une personne assise dessus) restent toujours très mal modélisés (figure 33 34).

Après plusieurs tentatives nous avons déduit que le problème était matériel, et sommes tombé sur deux conclusions :

1. Comme la chaise a été imprimée en deux parties et collé au niveau de l'assise, des points de colle doivent faire réfléchir la lumière (les reflets sont un des plus gros points faibles de COLMAP). Nous ne pouvons pas retirer la colle pour valider notre hypothèse car cela endommagerait la chaise.
Notre hypothèse est schématisée ci-dessous :

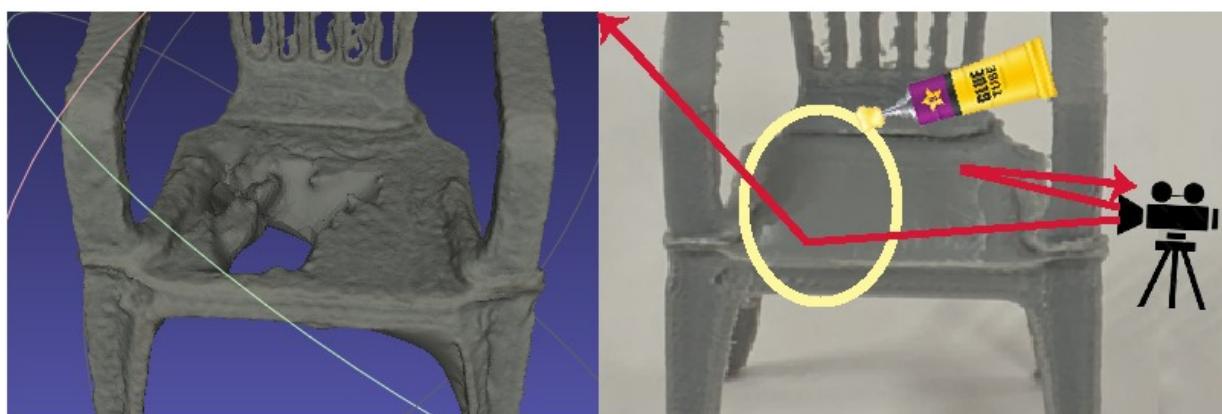


FIGURE 33 – Schéma de modélisation de l'assise de la chaise

2. La mauvaise modélisation du pied est causée par des imperfections aux supports de l'impression qui n'ont pas assez été poncé à cause de la fragilité de la chaise. Néanmoins nous avons pu poncer la zone problématique, ce qui donne le rendu visible sur la figure suivante :



FIGURE 34 – Schéma du problème de modélisation du pied arrière de la chaise

6.3. Transformation de la chaise en volume

Nous arrivons à la dernière étape du projet qui consiste à transformer la surface fermée obtenue avec Meshlab en volume imprimable.

Pour cela nous chargeons le fichier dans fusion 360 et lançons la conversion de la surface en Volume.

Nous obtenons la chaise suivante en format .stl prête à l'impression :



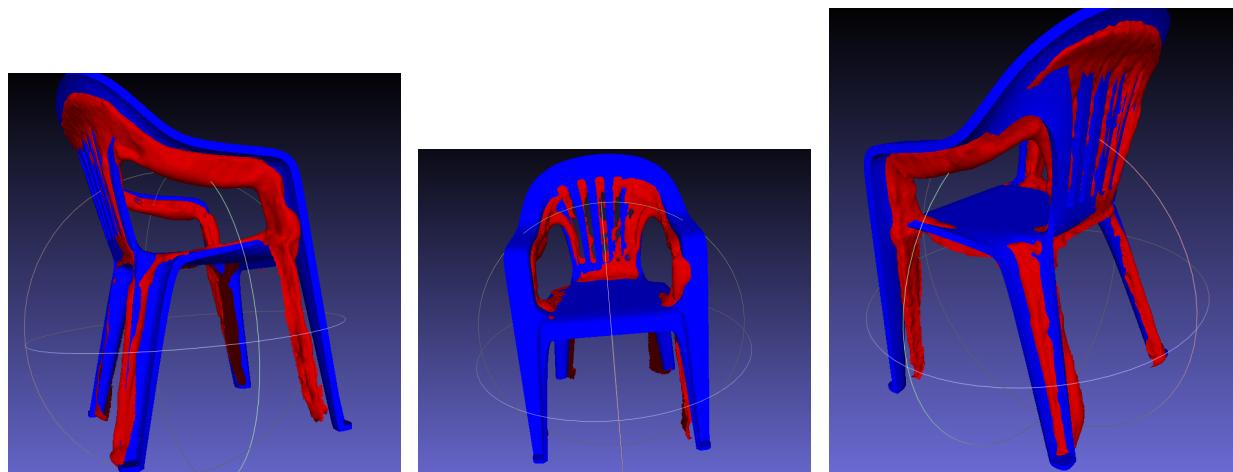
FIGURE 35 – Chaise exportée sous format .stl

6.4. Finalité du projet fils rouge

6.4.1. Comparaison avec Meshlab

Pour la comparaison des modèles, nous avons utiliser Meshlab pour superposer les deux modèles de chaise : le modèle de départ et le modèle de surface reconstruit de la chaise.

Nos obtenons les superpositions suivantes :



Comparaison des chaises côté droit Comparaison des chaises de face

Image 3

FIGURE 36 – Comparaison de la chaise, modèle 3D de départ et de la surface reconstruite avec Meshlab

En bleu le modèle 3D de départ de la chaise pris sur internet en .stl et en rouge, la surface reconstruite

de la chaise par Meshlab en .ply.

Nous avons donc superposé nos modèles dans Meshlab, l'objectif était ici d'établir une comparaison quantitative de la différence entre le modèle de départ et le modèle reconstruit. Cependant, nous n'avons pas pu finaliser cette étape et avons seulement ici des observations qualitatives.

Tout d'abord, le positionnement des objets dans Meshlab se fait à la mains de manière fastidieuse car on ne peut pas déplacer les objets manuellement. Il faut effectuer des rotations, des translations et des mises à l'échelle ce qui rend toute comparaison quantitative peu précise.

Nous pouvons tout de même voir que les modèles sont plutôt similaire au niveau du dos et que les pieds arrières s'alignent bien.

Le modèle reconstruit semble un peu plus petit mais cela peut être du à une mise à l'échelle pas suffisante. Ainsi, notre résultat est plutôt concluant même si nous n'avons pas de moyen de le quantifier, l'observation ici faite est que les modèles sont plutôt ressemblants et comparables.

6.4.2. Comparaison réelle



FIGURE 37 – chaise initiale



FIGURE 38 – Chaise ré-imprimé

FIGURE 39 – comparaison des deux impressions avant et après reconstruction

Nous pouvons voir sur la figure 39 que l'impression de la reconstruction est plutôt bonne, avec de bon détaille, malgré des problèmes de proportion (voir les deux accoudoirs qui ne sont pas symétriques) ou encore les éléments que nous avions prévu comme l'assise qui reste troué, ainsi que le dessous qui est mal reconstitué car bombé (non visible sur la figure même).

6.5. Livrables

Identifiant	Livrables	Description	Dates	Moyen de livraison
L1	Document de cadrage	Mise en place du cahier des charges, de l'organisation de l'équipe et définition des risques	27/02	Moodle + mail
L2	Premier tests du modèle avec la chaise	Visualisation du projet avec la chaise imprimée en 3D	11/03	Réunion avec l'encadrant
L3	Tests avec d'autres modèles	Tester avec d'autres modèles COLMAP en mode par défaut	25/03	réunion avec l'encadrant
L4	Problèmes rencontrés	Liste des problèmes rencontrés au L4 et fil à suivre	08/04	réunion avec l'encadrant
L5	Tests avec tous les paramètres	Tests du modèle choisi en configurant intégralement les paramètres	22/04	réunion avec l'encadrant
L6	Synthèse	Synthèse du L5 et tests d'améliorations	29/04	réunion avec l'encadrant
L7	Bonus	Optimisation des paramètres et du traitement d'image	13/05	réunion
L8	Rendu Final	Synthèse du Projet + prototype final	29/05	Moodle + mail
L9	Livrable envisageable	Reconstruction en temps réel		démonstration en réunion

TABLE 3 – Tableau des livrables

6.6. Conclusion

Au cours du projet nous avons développé de nombreux processus de test ainsi que des outils, cela nous a permis d'identifier le facteur le plus important pour avoir la reconstruction la plus aboutie : de bonnes images.

En effet, avoir une bonne base de départ avant de lancer une modélisation COLMAP permet de réduire le nettoyage manuel, qui est le plus grand obstacle dans la reconstruction automatique ou à grande échelle pour Renault. Nous avons donc mis en évidence une méthode type pour améliorer au mieux la qualité du modèle.

Dans un premier temps notre plateforme de shooting photo à permis d'obtenir des rendus d'images qui ne demande presque pas de retouche poste modélisation. Notre employeur ne souhaite pas avoir besoin de cette plateforme, un fond uniforme et une lumière homogène sur lequel nous pouvons tourner le modèle (et non tourner autour) donnera le même rendu.

Dans un second temps le programme de traitement d'images permet d'améliorer le rendu des images, bien que nos paramètres actuels ne sont pas mélioratifs, avec quelques ajustements les paramètres pourront être normalisé et mis en automatique. Ajouté à cela, avec un bon GPU nous pouvons effectuer des rendus en qualité "high", ce qui est tout à fait possible pour Renault, et si les applications sont pour un système embarqué (les ressources sont limitées), une qualité inférieure peut amplement suffire.

Côté automatisation, notre projet est proche de sa version finale. Le programme qui permet l'automatisation du traitement de images et la reconstruction avec COLMAP est fonctionnel mais nous n'avons pas pu le tester pour une reconstruction. Il nous permet, après avoir pris les images ou vidéos (qui sont automatiquement converti en images) de lancer COLMAP via python. Après la reconstruction, la seule

action humaine (dans les modèles les plus abouti) est de supprimer les effets de bord, ce qui peut difficilement être automatisé, mais ne prends que quelques minutes. Suite à cela, la mise en surface peut être automatisé dans le futur mais le choix de la méthode reste importante. Comme cela étant pour une mise en place a grande échelle, nous pourrions imaginer une IA qui en fonction de ce qu'elle a déjà vu, choisi la méthode de reconstruction la plus adéquate en fonction des caractéristique du modèle.

Néanmoins voici les limites qui persistent :

- La texture : Les imperfections du modèle à reconstituer peuvent brouiller COLMAP et aller jusqu'à l'empêcher de mettre des points sur la surface (fig. 34).
- La matière : Les matériaux réfléchissant, ou encore absorbant peuvent également laisser des trous dans la reconstruction.
- Passer du modèle en surface vers un volume pour imprimer est encore en recherche et reste actuellement pas abouti et laborieux.

Pour palier à cela il est préférable de faire les modifications directement sur le modèle d'origine si possible, ou d'ajouter/modifier les éléments les éléments problématiques après la modélisation.

Quant à l'impressionnabilité du modèle, du temps supplémentaire pourrait donner un projet totalement fonctionnement et semi-automatique car une simple présence humaine sur 5% du processus est nécessaire.

7. Retour d'expérience sur la gestion de projet

Concernant le retour sur la gestion du projet, nous remarquons que nous n'avons pas beaucoup utilisé Gitlab. Cela ne nous a pas semblé pertinent dans la mesure où nous rendions les livrables par mail et le projet ne nécessitait pas un grand nombre de programmes informatiques. L'important était surtout la prise et le stockage des photos pour ceux qui s'occupaient de faire la reconstruction soit avec le PC de Pierre-Antoine, soit avec le PC à disposition au fablab. Nous nous sommes donc tournés vers d'autres outils comme Google Drive et WhatsApp pour la partie communication.

Voici maintenant les diagrammes de Gantt du début, milieu et fin de projet PRONTO. Nous permettant de planifier visuellement les tâches du projet sur une échelle temporelle.

Au fil de l'évolution du projet, nous avons ajusté notre diagramme de Gantt initial de manière significative. Dans le Gantt initial, toutes les tâches étaient réparties de manière assez simple, avec peu de marge pour les imprévus. Les dates de rendu des livrables était rapproché, et les phases de test étaient concentrées sur une période relativement courte.

À mi-parcours, nous avons constaté que certaines tâches, notamment les tests, prenaient plus de temps que prévu. Nous avons donc allongé leur durée sur le planning, ce qui s'est révélé être un choix judicieux. Aussi, certaines tâches qui n'avait plus de sens suite à l'avancée du projet ont été abandonné (notamment les tests 3.3.3 et 3.3.4), afin de nous concentrer sur l'essentiel.

Le Gantt final montre une planification bien mieux maîtrisée. La plupart des tâches sont finalisées à 100 %.

Certain livrables (peut être trop ambitieux) ont été supprimé. Ces ajustements ont permis une meilleure répartition de la charge de travail et une progression plus fluide.

Pour la suite, nous ferons plus attention à anticiper les durées des tests dès le début des futurs projets.

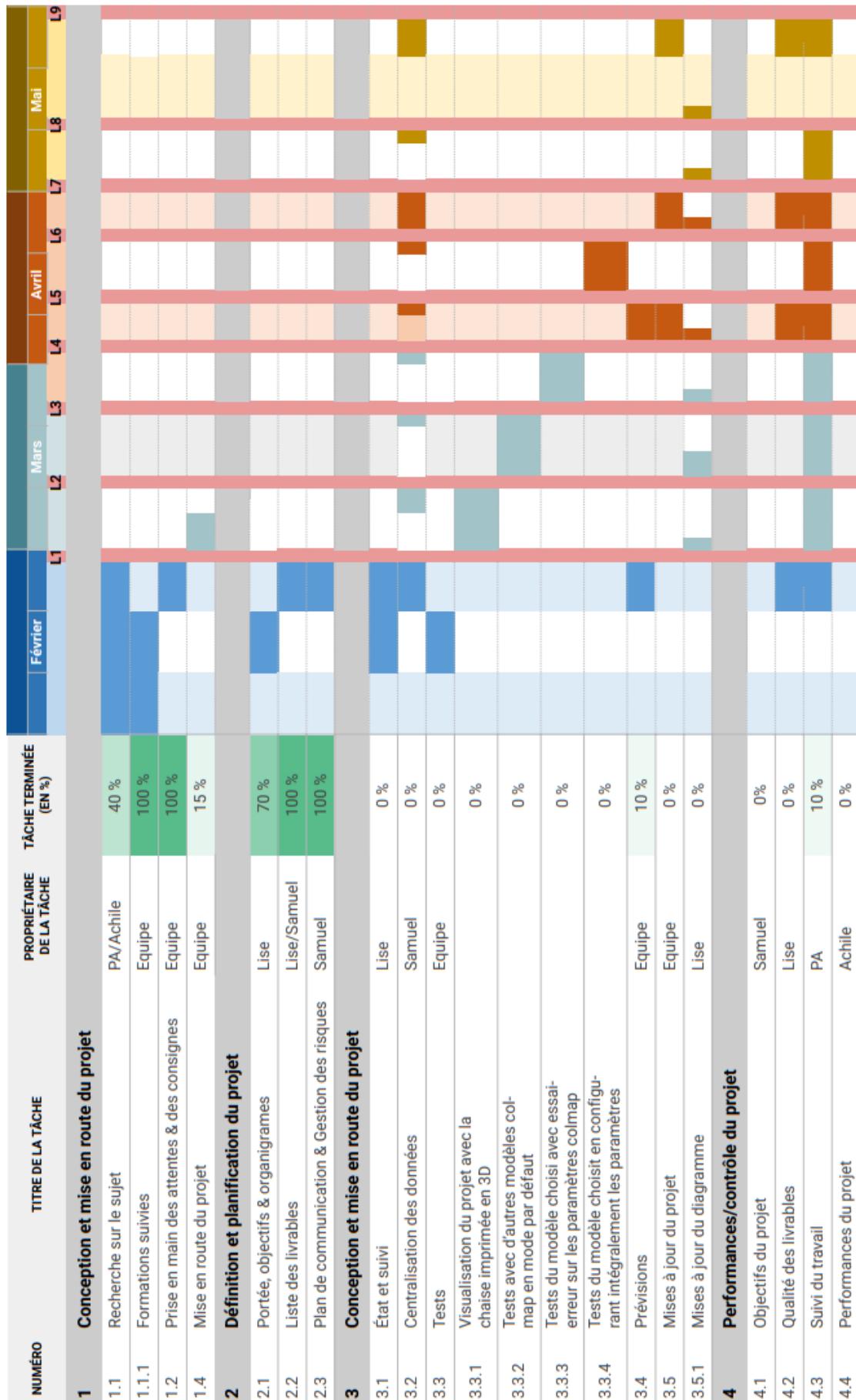


FIGURE 40 – Diagramme de Gantt version initial

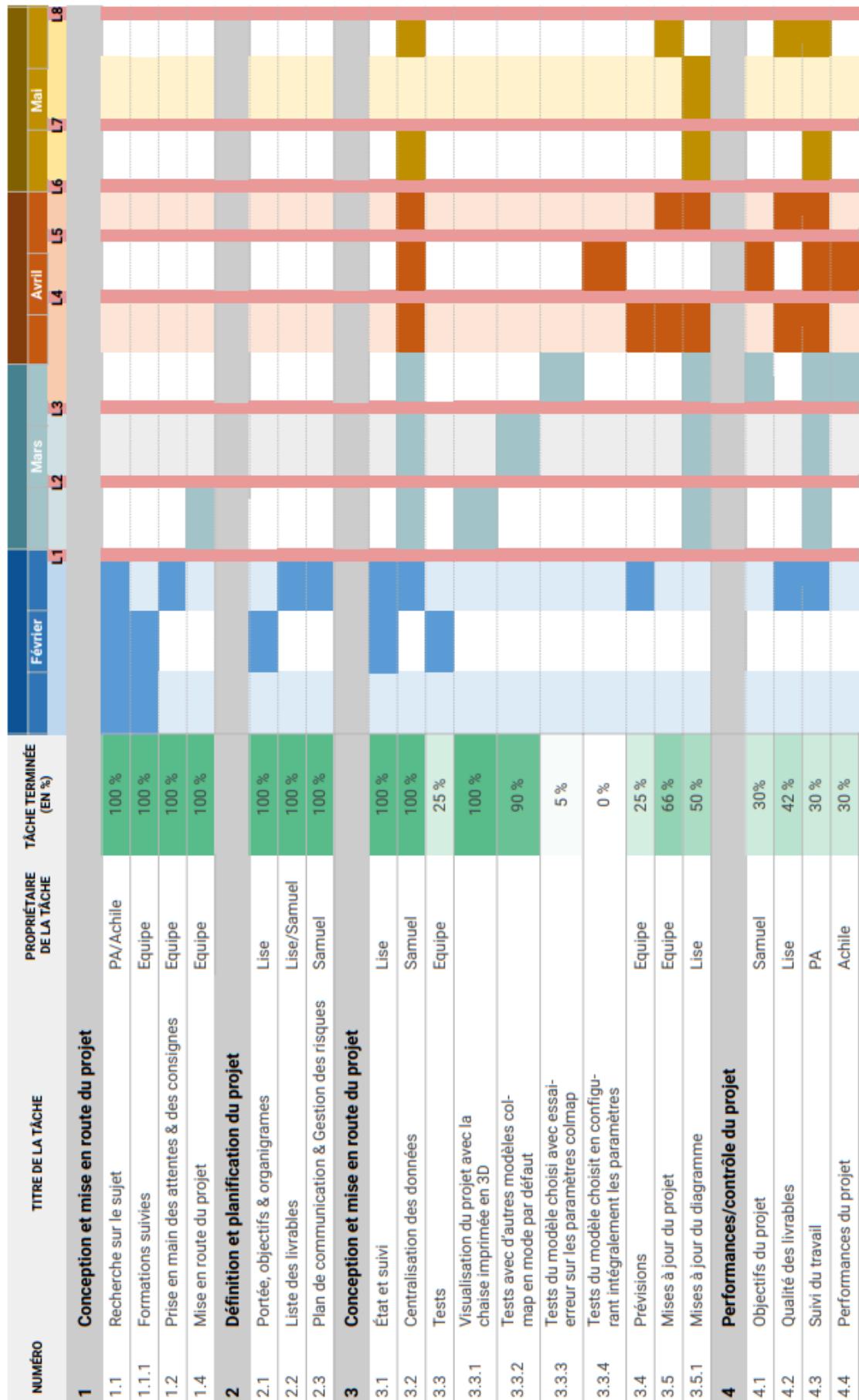


FIGURE 41 – Diagramme de Gantt version mi-projet

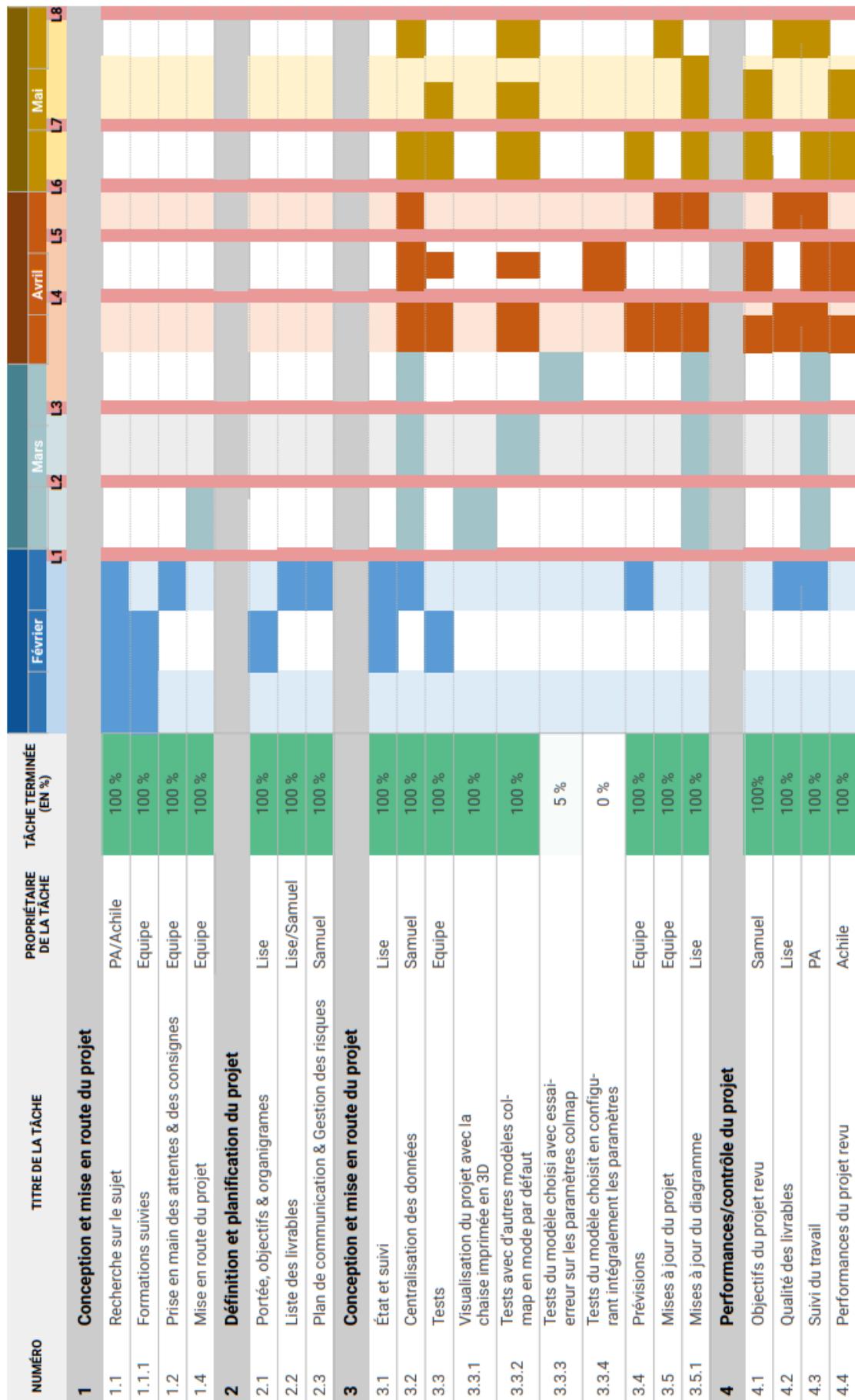


FIGURE 42 – Diagramme de Gantt version finale

8. Retours d'expérience sur le travail en équipe

8.1. Retour individuel - Pierre-Antoine HIGNARD

Décrivez une situation où vous avez pris le lead pendant ce projet.

La situation où j'ai le plus été directif lors de ce projet était dans la première partie de celui ci. Le problème étant que nous n'avions pas accès à un ordinateur de l'école assez puissant pour être utilisable dans notre contexte. Ainsi, l'ordinateur (personnel) Shadow auquel j'avais accès était le seul nous permettant d'effectuer des tests. J'ai alors pris une certaine longueur d'avance au début, car il était plus facile pour moi d'expérimenter sur COLMAP.

C'est dans ce cadre que j'ai du prendre les devants pour essayer de répartir l'acquisition de connaissances, et j'ai donc essayé de motiver les autres à faire les activités annexes à l'utilisation du logiciel COLMAP en lui même, comme la prise de photos, le nettoyage du rendu, la recréation de surface, etc... Cependant, cette tâche s'est avérée difficile, car tout le monde (et à raison) cherchait des alternatives à cette situation peu agréable. Samuel, par exemple a passé plusieurs cours du mardi matin à essayer d'installer COLMAP sur les PC Linux en B03, mais malheureusement, sans résultat. Le problème de temps allait de pair avec celui de motivation, car il était difficile pour les autres d'explorer pleinement (et indépendamment).

Cependant, cette période n'a pas été sans résultats, car nous avons tout de même pu avoir nos premières reconstructions, et premières prises en main du projet. C'est notamment dans cette période que nous avons pu faire le test entre la prise d'image par photos ou vidéo, où Samuel s'est occupé de gérer le code d'extraction d'images, pendant que Achile et moi étions entrain de prendre des photos/vidéos.

Une fois qu'Achille a pu obtenir l'accès au PC du Fablab, les autres ont gagné en indépendance. Pourtant, mon expérience passée d'aider un peu partout sans me focaliser nulle part m'a permis, par la suite de pouvoir continuer à avoir une vision globale sur le projet. Cela m'a alors permis (en ayant désormais une posture moins directive) de pouvoir participer à la plupart des résolutions de problème dans la suite.

Cette posture de "lead" a été intéressante, mais comporte toutefois des éléments qui correspondent un peu moins à ce que je cherche. En effet, le fait de "toucher à tout" implique de ne se focaliser réellement nulle part, et je ne saurais évoquer une partie que j'ai véritablement pu explorer en profondeur autant que je l'aurais voulu.

8.2. Retour individuel - Samuel VATON

Racontez une situation dans ce projet où vous étiez confrontée à un problème qui avait plusieurs solutions possibles.

Dans l'objectif de traiter les images à envoyer dans COLMAP pour observer l'impact du traitement sur la qualité du nuage de points, nous avons décidé d'utiliser un programme python pour d'une part traiter les images et de l'autre effectuer la reconstruction dans COLMAP. C'est cette dernière étape qui est l'objet de mon attention car je devais m'en charger.

En effet, il y a plusieurs moyen d'arriver à lancer une reconstruction. Il existe une librairie python appelé pycolmap qui a été conçu pour cette utilisation précise. Elle permet d'effectuer toutes les étapes de reconstruction avec un fort contrôle sur les paramètres mais aussi la possibilité de prendre des paramètres par défaut, ce qui revient à choisir le mode reconstruction automatique. Cependant, Pierre-Antoine ayant déjà testé d'utiliser la librairie, nous avions peur de ne pas pouvoir activer l'utilisation du GPU pour le rendu du nuage de points, ce qui aurait rendu la méthode inutile.

Il existait alors une autre méthode se basant sur le fonctionnement de COLMAP et le format des fichier permettant de lancer un projet. En effet, notre utilisation de COLMAP nous a fait comprendre que les projets du logiciel étaient sous forme de .ini avec par exemple un dossier spécifique **workplace folder** puis un fichier **projet.ini** que l'on pouvait charger dans le logiciel.

Le .ini était en réalité un fichier texte que l'on pouvait écrire directement. Ainsi, en créant un programme python remplissant tous les paramètres nécessaires dans un nouveau fichier .ini, nous pouvions lancer un projet dans COLMAP. Le fichier contenait de nombreux paramètres comme le chemin vers le dossier d'images à utiliser ainsi que des paramètres de reconstruction. L'avantage de cette méthode était quelle permettait d'utiliser toutes les ressources de COLMAP sans restrictions associées à la bibliothèque pycolmap.

J'avais donc plusieurs possibilités à ma disposition pour effectuer la reconstruction etachever la création du programme python de traitement d'images puis reconstruction.

J'ai finalement choisi d'utiliser la méthode avec pycolmap car elle a été créée dans cet objectif : faciliter le travail d'automatisation d'utilisation de COLMAP. De plus il ne semble finalement pas y avoir de restriction sur l'utilisation de GPU par COLMAP avec la librairie pycolmap. Pour finir, bien que les solutions soient différentes, je suppose que les 2 méthodes auraient donné un résultat similaire dans l'application de COLMAP même si la deuxième aurait nécessité un effort sur l'ouverture du fichier .ini directement avec COLMAP, difficulté sur laquelle je ne me suis pas penché.

8.3. Retour individuel - Anne-Lise PELLETIER

Dans ce projet, quand avez-vous pris un risque, fait une erreur ou échoué ?

Lors du projet de groupe, notre objectif initial était clair : reproduire le plus fidèlement possible une chaise existante. C'était un objectif assez simple en apparence, mais qui nous demandait en réalité beaucoup de rigueur.

Au début, tout semblait bien se passer. J'étais assez investie dans la partie nettoyage et correction du modèle. Petit à petit, j'ai commencé à corriger certaines irrégularités sur MeshLab comme lisser les bords un peu tordus, combler les trous, corriger des asymétries dans la structure. Sur le moment, toutes ces modifications me semblaient logiques et "nécessaires" pour améliorer la qualité de notre rendu final.

Mais avec du recul, je me suis rendu compte que j'étais allée trop loin. À force de retouches, notre modèle devenait presque une "nouvelle chaise". Il était plus propre, plus symétrique, peut-être plus esthétique aussi, mais il ne ressemblait plus vraiment à la chaise qu'on a modélisée à partir des photos. On s'éloignait des données issues de la simulation, qui étaient censées être notre base de travail.

Ce que je percevais comme des défauts à corriger étaient en réalité des caractéristiques importantes de l'objet réel. Ces irrégularités faisaient justement partie de ce qu'on devait analyser. En voulant "bien faire", j'ai donc pris un risque sans m'en rendre compte : celui de perdre le sens de notre projet.

C'est lors d'une séance d'échange avec le groupe que j'ai pu vraiment me rendre compte de cette dérive. On a discuté de l'écart entre notre modèle et la réalité, et c'est là que j'ai compris que mes modifications, même si elles partaient d'une bonne intention, nuisaient à l'authenticité de notre travail. À partir de ce moment-là, j'ai décidé de repartir sur une version plus brute du modèle, avec ses défauts et ses irrégularités, mais plus fidèle à la réalité.

Cette erreur m'a appris plusieurs choses. D'abord, que le dialogue avec les autres membres du groupe est essentiel pour prendre du recul et éviter de s'enfermer dans une logique personnelle. Aussi, cette expérience m'a permis de comprendre que dans un projet technique, il faut toujours garder une vision globale pour rester cohérent avec nos objectifs.

8.4. Retour individuel - Achile PINSARD

Qu'avez-vous fait quand vous deviez motiver un groupe ou une personne dans ce projet ?

Au cours de ce projet, la motivation du groupe a connu plusieurs variations, principalement dues à des contraintes techniques et des retards qui ont parfois ralenti notre dynamique. Dès les premières séances, nous avons été confrontés à un problème majeur : l'impossibilité d'utiliser COLMAP sur nos ordinateurs personnels et les ordinateurs mis à disposition par l'école. Après de multiples tentatives et recherche de solutions, une forme de lassitude s'est installée dans le groupe, moi compris.

J'ai alors proposé une nouvelle organisation qui nous permettait de continuer à progresser malgré ce blocage : concentrer l'utilisation de COLMAP sur l'unique ordinateur fonctionnel, tandis que le reste du groupe se répartissait sur d'autres tâches en parallèle (prise de photos, nettoyage de modèle, documentation). Ce fonctionnement nous a permis de redonner un rythme au projet et d'éviter l'inaction. Pour renforcer cette dynamique, j'ai également proposé l'idée d'un objet « fil rouge » qui guiderait nos essais et permettrait de visualiser concrètement nos avancées. Ce repère a contribué à remobiliser le groupe.

Dans la foulée, et en tant que responsable Fablab, j'ai également pris l'initiative de chercher une solution au fablab afin de savoir si un ordinateur pouvait être mis à disposition. Nous avons ainsi obtenu un poste adapté, ce qui a été une avancée décisive pour la suite du projet.

Par ailleurs, face à des retards répétés d'un membre du groupe, j'ai tenté de comprendre la situation sans jugement. En discutant avec la personne, j'ai pu identifier des freins. Cela m'a permis de lui proposer des tâches plus ciblées, et s'adapter au problème, afin de le réintégrer dans une dynamique de groupe plus fluide.

Enfin, après le retour du premier rendu, plutôt négatif, le groupe a subi un nouveau coup de mou. J'ai essayé de recentrer les échanges sur les prochaines étapes, en insistant sur le fait que nous étions désormais libérés du cadrage, et que nous pouvions pleinement nous investir dans la partie technique et créative du projet, bien plus motivante. Cela a permis de relancer l'énergie collective.

Au final, je pense que ma manière de motiver est surtout passée par des réorganisations concrètes, des propositions structurantes, et une écoute active lorsque des décrochages apparaissaient. Je n'ai pas imposé une direction, mais j'ai essayé de maintenir une dynamique positive et collective à chaque étape clé du projet.

8.5. Retour commun

Dès le début du projet, nous nous sommes bien entendu, la communication était bonne et l'ambiance de travail également. Notre première difficulté fut un manque de communication avec notre référente, ce qui nous a directement pénalisé sur le premier rendu (rendu de cadrage), c'est pourquoi ce rendu nous a été bénéfique et a permis de nous ajuster. Malgré les problèmes de matériels, en l'occurrence d'ordinateurs pour faire tourner COLMAP, nous nous sommes bien ajusté sur seulement l'ordinateur virtuel que Pierre-Antoine a gentiment mis à notre disposition. Notre point fort est de très bien nous adapter à l'environnement et aux différentes difficultés. Pour les points à améliorer, il faudrait plus de rigueur et moins de tolérance sur les retards, car cela a pour conséquence de casser les dynamiques de travail (certains travaillent alors qu'il manque quelqu'un, il faut mettre la personne à jour).

9. Conclusion et perspectives

Le projet PRONTO nous a apporté beaucoup de choses, nous avons pu observer le travail en équipe dans un projet complexe sur une longue durée. Il nous a fallu un temps d'adaptation mais nous avons su rebondir sur nos difficultés pour progresser ensemble. Certes, nous avons manqué de temps pour finaliser toutes nos réalisations, mais nous avons appris de nombreuses choses sur la vision par ordinateur et la reconstruction de modèles en 3D. De plus nous avons réussi à réimprimer une chaise à partie du modèle de départ ce qui constitue une forte réussite. Ainsi, nous repartons satisfait de ce projet que nous avons mené et espérons que ce rendu reflète nos efforts et le travail que nous avons fourni tout au long de ce dernier.

Références

- [1] Schönberger, J. L., & Frahm, J. M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4104-4113). IEEE.
- [2] First Principles of Computer Vision. (2021, May 2). *Optical Flow | Structure from Motion | Object tracking* [Video playlist]. YouTube. <https://www.youtube.com/playlist?list=PL2zRqk16wsdoYzrWStffqBAoUY8XdvatV>
- [3] Schönberger, J. L. (2023). *COLMAP - Structure-from-Motion and Multi-View Stereo*. Documentation officielle. <https://colmap.github.io/>
- [4] Mines Paris PSL. *TP Meshlab*. https://people.minesparis.psl.eu/olivier.stab/TP_scilab_MG91/TP_MeshLab/TP_MeshLab_1.html
- [5] MPR (Youtube). *Meshroom : Initial pipeline, CCTags, using a turntable and known camera positions*. <https://www.youtube.com/watch?v=XUKu1apUuVE>
- [6] Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*. OpenCV Team. <https://opencv.org>
- [7] Schönberger, J. L. (2023). *pycolmap : Python bindings for COLMAP*. Documentation officielle. <https://colmap.github.io/pycolmap/pycolmap.html>
- [8] Autodesk Support, *How to convert a mesh to a solid or surface body in Fusion* <https://www.autodesk.com/support/technical/article/caas/sfdcarticles/sfdcarticles/How-to-Convert-a-Mesh-to-a-BRep-in-Fusion-360.html>

OUR WORLDWIDE PARTNERS UNIVERSITIES - DOUBLE DEGREE AGREEMENTS



3 CAMPUS



IMT Atlantique Bretagne-Pays de la Loire – <http://www.imt-atlantique.fr/>

Campus de Brest

Technopôle Brest-Iroise
CS 83818
29238 Brest Cedex 3
France
T +33 (0)2 29 00 11 11
F +33 (0)2 29 00 10 00

Campus de Nantes

4, rue Alfred Kastler
CS 20722
44307 Nantes Cedex 3
France
T +33 (0)2 51 85 81 00
F +33 (0)2 99 12 70 08

Campus de Rennes

2, rue de la Châtaigneraie
CS 17607
35576 Cesson Sévigné Cedex
France
T +33 (0)2 99 12 70 00
F +33 (0)2 51 85 81 99



IMT Atlantique

Bretagne-Pays de la Loire
École Mines-Télécom