Nouvelles méthodes de synchronisation de nuages de points 3D pour l'insertion de données cachées

Vincent Itier

Université de Montpellier

Thèse

présentée au **L**aboratoire d'**I**nformatique de **R**obotique
et de **M**icroélectronique de **M**ontpellier pour
obtenir le diplôme de doctorat

Spécialité Doctorale : **Informatique**
École Doctorale : **Information, Structures, Systèmes**

Nouvelles méthodes de synchronisation de nuages de points 3D pour l'insertion de données cachées

---

par

Vincent ITIER

Soutenance prévue pour le 1 décembre 2015, devant le jury composé de :

Directeur de thèse :

M. William PUECH, Professeur des Universités ..... LIRMM, Université de Montpellier

Co-encadrant :

M. Gérard SUBSOL, Chargé de recherches ..... CNRS, LIRMM Montpellier

Rapporteurs :

M. Vincent CHARVILLAT, Professeur des Universités ..... IRIT, ENSEEIHT Toulouse

M. Guillaume LAVOUÉ, Maître de conférence HDR ..... LIRIS, INSA de Lyon

Examinateurs :

M. Stéphane BESSY, Maître de conférence HDR ..... LIRMM, Université de Montpellier

M. Jean-Marc CHASSERY, Directeur de recherches ..... GIPSA-Lab Grenoble

M. Gilles GESQUIÈRE, Professeur des Universités ..... LIRIS, Université de Lyon 2

Invité :

M. Jean-Pierre PEDEBOY, PDG, Ingénieur ..... STRATEGIES S.A. Rungis

Remerciements

Tout d'abord je remercie Vincent CHARVILLAT et Guillaume LAVOUÉ qui ont accepté de relire ce manuscrit et d'être rapporteurs de ma thèse. Leur lecture attentive et leurs remarques précises m'ont permis d'améliorer la version finale de ce manuscrit.

Je voudrais remercier tout particulièrement mon directeur de thèse William PUECH qui m'a encadré et dirigé pendant la durée de ma thèse. Son aide et ses conseils ont été des points essentiels pour comprendre et m'adapter aux enjeux de la recherche. Ses nombreuses relectures et corrections de mes travaux, en particuliers ceux de cette thèse, ont été très importantes pour leurs qualités finales. Travaillé sous sa direction a été très agréable, et pour cela merci.

Je remercie également, Gérard SUBSOL qui a co-encadré cette thèse et a su apporter un regard différent ainsi que des solutions claires sur certaines problématiques. Ses remarques ont été constructives pour l'avancement de mon travail ainsi que pendant la rédaction de mon manuscrit.

Je souhaite remercier les personnes qui m'ont permis d'avancer dans mon travail. Je remercie Adrian BORS pour l'accueil pendant un mois dans son laboratoire de l'Université de York ainsi que les collaborations sur la durée de ma thèse. Je remercie l'équipe ICAR pour toutes les aides et les conseils que j'ai obtenus auprès de tout le monde. Je remercie Gilles GESQUIÈRE pour m'avoir conseillé au début de mes travaux.

Plus généralement, je souhaite remercier l'ensemble de l'équipe ICAR, pour la bonne ambiance, le plaisir de venir au laboratoire et l'enthousiasme général. En particulier pendant les pauses café, et les fameux C&C.

Je souhaite aussi exprimer ma gratitude à la société STRATÉGIES, et plus précisément Jean-Pierre PEDEBOY, pour le financement de ma thèse au travers d'une convention Cifre entre l'entreprise et le LIRMM. Son intérêt pour les innovations et ses perspectives industrielles ont permis de cerner des besoins précis pour mes travaux de recherches. Finalement, je souhaite remercier mes amis et ma famille qui ont toujours été à mes côtés.

Je remercie particulièrement mes parents qui m'ont toujours encouragé à poursuivre des études qui me passionnent et qui m'ont tout apporté.

Table des matières

Chapitre 1

Introduction

1.1 Contexte de la thèse

Les travaux de cette thèse ont été réalisés dans le cadre d'un contrat CIFRE entre la société STRATEGIES [1] et l'équipe ICAR du LIRMM. La société développe une suite logicielle Romans CAD Software à destination des designers de l'industrie de la mode : chaussure, bagagerie et matériaux souples. La Fig. 1.1 présente un exemple d'utilisation d'un logiciel de la société.

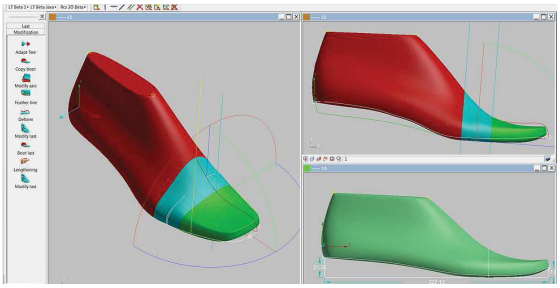


FIGURE 1.1: Illustration de la suite Romans CAD Software.

Les logiciels de CAD (Computer-Aided Design) permettent d'assister la création. Plus précisément, les outils de traitement numérique 3D développés par la société STRATEGIES permettent aux créateurs et aux designers de modéliser un produit final. Pour la modélisation de chaussures, les designers s'appuient sur une forme interne de chaussure, celle-ci peut être créée numériquement ou créée physiquement et scannée. Ces outils permettent également l'ajout de coutures, de textures, de couleurs ou d'éléments propres aux chaussures. Le modèle 3D peut ensuite être envoyé dans les entreprises de production de l'objet final. L'avantage de la modélisation numérique 3D est de permettre aux designers de visualiser le rendu en continu et d'avoir un contrôle direct de

1. www.cadwinfm.com

leur travail. Un autre avantage est de pouvoir concevoir des prototypes de façon rapide et simplifiée pouvant être imprimés à l'aide d'imprimantes 3D, ce qui permet un gain de temps et d'argent. Les maillages 3D réalisés au travers de ces logiciels, dans un format propriétaire, sont de grandes tailles et comportent de nombreux attributs comme la couleur ou la texture. Les travaux de cette thèse s'inscrivent dans la continuité du partenariat entre la société STRATEGIES et l'équipe ICAR du LIRMM, au cours duquel des recherches ont été menés comme l'aide au découpage automatique de pièces, la numérisation de formes 3D, l'analyse et le tatouage de maillages 3D.

1.2 Applications

Dans un cadre plus général, les maillages 3D représentent une part de plus en plus importante des médias numériques visuels, au travers des logiciels de CAD, de l'imagerie médicale, des applications pour le patrimoine culturel ou des jeux vidéos par exemple. Les objets 3D peuvent être représentés numériquement de plusieurs manières à l'aide de maillages, de surfaces implicites, de NURBS ou de voxels par exemple. Un maillage 3D est une approximation de la surface d'un objet 3D, définie à l'aide d'informations géométriques et topologiques. Ces maillages 3D sont devenus un standard de représentation 3D grâce à leur simplicité d'utilisation. D'autre part, les systèmes de scanners 3D sont de plus en plus répandus, et l'impression 3D devient aujourd'hui facilement accessible, ce qui renforce la présence des représentations 3D numériques sous forme de maillages.

De nombreux formats de maillages 3D existent. Ils consistent généralement à stocker l'information géométrique, la connectivité topologique et parfois des attributs additionnels comme des couleurs, des textures, etc. Les entreprises utilisant les maillages 3D, comme la société STRATEGIES, possèdent souvent un format propriétaire. Cependant, pour des applications, comme le partage ou l'impression 3D, les maillages propriétaires sont convertis dans des formats standards. Ces conversions peuvent s'accompagner d'une perte des données associées aux maillages comme des paramètres de création, dates, site web, etc. De plus, ces maillages 3D peuvent être copiés et redistribués sans perte de qualité par un pirate. Ces actions peuvent entraîner des dommages aux ayants droit et à la chaine de production.

Les travaux effectués dans les domaines des images ou des vidéos numériques montrent que l'insertion de données cachées est une solution intéressante à apporter à ces différents problèmes. L'insertion de données cachées 3D permet d'insérer de façon imperceptible un message dans un maillage 3D. Il existe plusieurs formes d'insertion de données cachées. Le tatouage robuste permet d'insérer un identifiant pour les droits

d'auteur, cet identifiant devrait être préservé même après des modifications. Le tatouage fragile permet de vérifier l'intégrité d'un maillage, il est conçu pour être altéré après certaines modifications. Il existe aussi l'insertion de données cachées haute capacité qui permet de cacher une grande quantité d'information dans un maillage 3D, comme des informations sur la création, du contenu sémantique ou des textures. L'avantage principal des méthodes d'insertion de données cachées est de conserver ces informations dans un seul maillage, même si son format est changé et cette méthode peut être vue comme une méthode de compression puisque la taille du maillage tatoué doit rester inchangée par rapport à la taille du maillage original.

Les applications possibles envisagées par la société STRATEGIES sont donc multiples. Un maillage 3D peut être une création collaborative, il faudrait donc que des informations sur la création puissent être liées avec le maillage. Le maillage 3D doit pouvoir être transmis de façon sécurisée entre un créateur et une entreprise de production. Une fois la transmission effectuée, il faudrait alors pourvoir clamer la propriété d'un maillage ou contrôler son intégrité. Un autre cas d'application est de pouvoir transmettre de façon secrète et sécurisée des méta-données comme un logo, une texture ou des lignes de style. Dans ce cas, un utilisateur non autorisé aura accès à la forme 3D mais ne pourra pas extraire les méta-données.

1.3 Challenge et contributions

Après avoir défini les besoins et les attentes de la société STRATEGIES, nous proposons de lister les enjeux qu'ils soulèvent dans les domaines de recherche. En effet, l'insertion de données cachées 3D a un temps de retard sur la 2D, sachant que ce domaine de recherche est plus récent. Cependant, les contraintes liées au passage à la 3D créent de nouveaux défis. En effet, la synchronisation dans les maillages 3D, c'est-à-dire la définition d'un ordre de parcours des éléments du maillage, n'est pas triviale comme en 2D où les pixels sont attachés à une grille régulière. Notre première contribution a été d'analyser des méthodes de synchronisation de nuages de points dans l'espace. Un autre challenge est de pouvoir proposer des méthodes haute capacité, produisant peu de distorsions sur la surface tout en étant potentiellement indétectables. Finalement, le défi de la sécurité est encore peu abordé dans les travaux d'insertion de données cachées 3D. En effet, il faudrait proposer des méthodes dont l'accès au message caché est impossible à un utilisateur non autorisé. Pour répondre à ces problèmes, nous proposons de sécuriser à la fois l'étape d'insertion et l'étape de synchronisation. De plus nous proposons de rendre insoupçonnable un maillage marqué, c'est-à-dire qu'un attaquant ne puisse juger du fait qu'il soit marqué ou non.

1.4 Plan

Le reste du manuscrit se compose de deux parties, à savoir un état de l'art et une seconde partie consacrée à nos contributions :

Dans la première partie, nous présentons les maillages 3D dans le Chapitre 2. Dans le Chapitre 3, nous décrivons les principes de l'insertion de données dans des supports numériques visuels et ses propriétés. Ensuite, Chapitre 4 nous comparons les méthodes et solutions apportées par l'étude de la littérature de l'insertion de données cachées dans des maillages 3D.

Dans la seconde partie de cette thèse, nous présentons nos contributions. Dans un premier temps, dans le Chapitre 5 nous apportons une analyse des problèmes de synchronisation de nuages de points 3D. L'utilisation de l'analyse de la synchronisation est présentée dans le Chapitre 6 dans le but de proposer une insertion de données cachées haute capacité. Chapitre 7 nous proposons une autre méthode de synchronisation plus sécurisée, ainsi que des améliorations.

Finalement, nous concluons ce manuscrit dans le Chapitre 8, et après un bilan des travaux de recherches effectués, nous proposons quelques perspectives à étudier.

Première partie

État de l'art

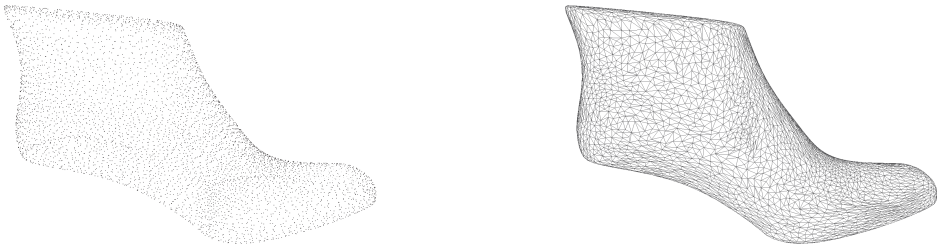
Chapitre 2

Maillages 3D

2.1 Introduction

Les représentations surfaciques 3D sont de plus en plus utilisées, que ce soit des objets créés numériquement ou des objets acquis. Les maillages 3D peuvent être produits par des designers ou des artistes, à l'aide de logiciels de CAD, ces maillages sont généralement de bonne qualité. Les représentations numériques d'objets 3D réels sont acquises à l'aide de scanners 2D ou 3D. Certains appareils permettent également d'ajouter une information de texture, de réflectance ou d'estimation de la couleur. Les systèmes d'acquisition peuvent être classés soit en actifs, soit en passifs. Ces systèmes peuvent utiliser un ou plusieurs capteurs (caméra, appareil photo). Les systèmes passifs utilisent seulement l'information de la scène, comme la différence des points de vue (stéréovision), ou le mouvement (shape from motion) par exemple. Les systèmes actifs contrôlent les conditions d'éclairement, projettent une lumière structurée ou un laser. Un état de l'art complet des méthodes d'acquisition a été proposé par Sansoni *et. al* [104]. Il existe un grand nombre de techniques de numérisation pour la reconstruction 3D qui peut être faite par le scanner ou *a posteriori*. Nous avons un aperçu des techniques et nous pouvons appréhender la diversité des problèmes de traitement auxquels la reconstruction 3D est confrontée. En effet, l'acquisition de nuages de points produit des erreurs comme un échantillonnage bruité, non uniforme ou avec des points aberrants. Du fait de certaines contraintes, comme ne pas déplacer une œuvre d'art, certaines zones peuvent être mal ou pas du tout acquises. Un nuage de points acquis est un échantillonnage de la forme de la surface continue de l'objet. Pour avoir une surface fermée une étape de maillage est nécessaire. Cette étape permet de donner une connectivité au maillage ainsi que sa topologie. Si le nombre d'échantillons est bien adapté, la surface continue peut être approximée de façon précise. L'étape de maillage est souvent une étape de

triangulation, qui peut être effectuée grâce aux algorithmes de triangulation de Delaunay [25] ou une reconstruction de poisson [62]. Un état de l'art récent des techniques de maillage de nuages de points 3D a été proposé par Berger *et. al* [7]. La Fig. 2.1 illustre la triangulation d'un nuage de points 3D acquis grâce un scanner à partir d'une forme de l'intérieur d'une chaussure. Les représentations d'objets 3D volumiques comme les voxels, les représentations implicites, ou représentations à l'aide de courbes de contrôle, ne sont pas considérées dans notre étude. Les principales représentations et le traitement bas niveau (réparation, remaillage, simplification) sont expliqués dans le livre de Botsch *et. al* [10]. La représentation d'un objet 3D par un maillage 3D permet de simplifier l'acquisition, le traitement, la manipulation et le stockage de ces objets.



(a) (b)

FIGURE 2.1: a) Nuage de 5002 points 3D, b) maillage triangulé associé.

Dans ce chapitre, Section 2.2 nous abordons les maillages 3D, comment les représenter et leurs propriétés. Nous expliquons quelles modifications sont susceptibles d'être appliquées à un maillage 3D. Dans la Section 2.3, nous présentons l'évaluation des modifications appliquées à un maillage et plus précisément les métriques utilisées dans nos travaux. Puis, dans la Section 2.4, nous expliquons le problème d'ordonnancement des primitives des maillages, au travers de différentes techniques.

2.2 Représentations des maillages 3D

Dans cette section, nous introduisons la représentation et le traitement des maillages 3D. Dans un premier temps, nous expliquons comment les maillages 3D sont définis. Puis, nous présentons les propriétés et définitions nécessaires à la compréhension et à la manipulation des maillages 3D. Enfin, nous listons une série de modifications auxquelles peut être soumis un maillage 3D.

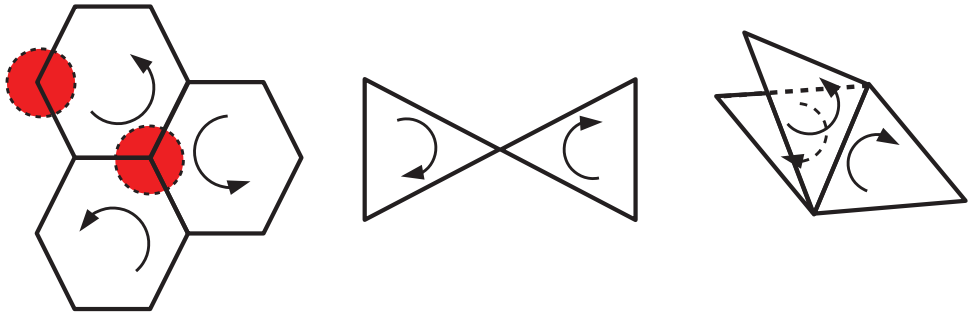
2.2.1 Définitions

Un maillage 3D est une approximation d'une surface continue, une bonne approximation est possible si la densité de sommets est adaptée [10]. Un maillage 3D, $M = (V, K)$, est donc défini par sa géométrie V et sa connectivité topologique K. Un nuage de points représente la géométrie d'un objet 3D. Ce nuage de points est l'ensemble de n points ou sommets, noté V tel que :

$$V = \{v_1, \ldots, v_n\},\ v_i \in \mathbb{R}^3,\ 1 \leq i \leq n. \tag{2.1}$$

Le terme topologie fait souvent référence à la connectivité du maillage, cependant certains auteurs pointent la différence entre la connectivité et la forme de la surface [101]. Nous distinguerons donc le terme connectivité topologique de celui de forme topologique, puisque la topologie est l'étude des propriétés invariantes aux homéomorphismes. Dans ce cadre, la réalisation topologique est un complexe de cellules, c'est la décomposition de l'espace en cellules. Une n-cellule dans un espace est définie comme un sous-ensemble homéomorphe à $E^n = \{x \in \mathbb{R}^n \mid |x| < 1\}$ [28]. Un maillage est donc la réalisation géométrique d'une représentation topologique qui est indépendante de l'espace dans lequel le maillage est plongé. On note le maillage $M = (V, F, E)$ et ses ensembles de primitives : l'ensemble de sommets V, l'ensemble de facettes F et l'ensemble des arêtes E du maillage. La plupart des algorithmes de traitement nécessitent des maillages triangulaires $f \in V \times V \times V$, puisqu'ils permettent à une représentation de s'appuyer sur des propriétés topologiques plus strictes. Plus généralement, une surface est définie comme une "variété de dimension deux (2-variété), compacte, connectée, orientable et avec éventuellement un bord, plongée dans $\mathbb{R}^3$" [93]. Cette définition de la surface suppose soit un maillage créé pour respecter ces règles, soit une étape de pré-traitement d'un nuage de points ou d'une soupe de polygones, pour obtenir une surface bien définie.

Une surface connectée de dimension 2 est 2-variété, si en tout point x, le voisinage local peut être déformé continuellement en un disque [45]. Plus simplement, nous considérons le cas où une arête est partagée par exactement deux facettes et où la surface ne s'intersecte pas elle même. La Fig. 2.2.a présente un maillage 2-variété, et deux exemples de maillages non-variétés Fig. 2.2.b et Fig. 2.2.c. Les cercles rouges de la Fig. 2.2.a représentent un disque ouvert sur la bordure et un disque fermé au centre. Dans ces exemples, la surface est bornée et possède une bordure. Une surface sans bord et bornée est dite fermée. Par exemple, un plan est une 2-variété sans bord dans $\mathbb{R}^d$ mais n'est pas borné. Alors qu'une sphère est une 2-variété fermée dans $\mathbb{R}^d$. Une surface 2-variété est orientable si toutes ses facettes adjacentes deux à deux ont une orientation compatible. L'orientation d'une facette est un ordre cyclique des sommets qui la



(a) (b) (c)

FIGURE 2.2: Exemple de configuration a) 2-variété, b) non-variété orientable, c) non-variété non orientable.

composent. Pour que deux facettes adjacentes soient compatibles, il faut que l'arête en commun soit utilisée une fois dans un sens, une fois dans l'autre. Le maillage de la Fig. 2.2.a est orientable, par contre le maillage de la Fig. 2.2.c n'est pas orientable puisqu'une arête est partagée par trois polygones. Il existe cependant des surfaces 2-variétés non orientables comme le ruban de Möbius et la bouteille de Klein.

La caractéristique d'Euler décrit un invariant topologique pour un complexe cellulaire fini M. Les complexes finis ayant la même caractéristique d'Euler sont équivalents :

$$\chi(M) = \sum_{i \geq 0} (-1)^i n(i), \tag{2.2}$$

où $n(i)$ représente le nombre de i-cellules. Euler a établi cette formule pour montrer que pour une sphère chaque polyèdre l'approximant donne une valeur de : $n(0)-n(1)+n(2) = 2$, ce qui correspond pour un maillage $M = (V, F, E)$ à : $|V| - |E| + |F| = 2$. Tous les maillages homéomorphes à une sphère ont donc une caractéristique d'Euler de 2. La caractéristique d'Euler dépend du nombre de poignées d'un maillage, appelé genre du maillage. Par exemple, le maillage M d'un tore a une poignée et sa valeur $\chi(M) = 0$.

L'association d'un graphe $G = (V, E)$ à un maillage $M = (V, E, F)$, est évidente et permet d'utiliser les propriétés des graphes pour obtenir des informations sur la topologie d'un maillage.

Définition 2.1. Un complexe de dimension 1 est un graphe.

Par exemple, si le graphe associé d'un maillage complexe est connexe, le maillage l'est également.

Le degré d'un polygone est le nombre d'arêtes qui le composent. La valence d'un sommet est définie comme le nombre d'arêtes incidentes à ce sommet. Le voisinage d'un

sommet v_i est défini par sa connectivité. Le voisinage à une distance d'une arête (1-ring neighborhood), est l'ensemble des sommets v_j tel que $\exists\ e_{v_i,v_j} \in E$. Le voisinage est utile pour le calcul de distances géodésiques ou de courbures locales. La distance géodésique sur une surface M est la plus courte distance entre deux points v_i, $v_j \in M^2$ sur cette surface. Surazhsky *et. al* [108] proposent des méthodes pour le calcul de la distance géodésique sur des maillages de façon exacte ou approximée. La courbure locale en un point d'un maillage est approximée à l'aide de la normale en ce point et de son voisinage local par Taubin [110]. La normale N_i d'une facette f_i peut être obtenue en calculant le produit vectoriel de deux vecteurs non-colinéaires du plan dans lequel est plongée la facette f_i. Généralement, la normale N d'un sommet v est la moyenne pondérée des normales N_i des n facettes auxquelles il appartient :

$$N = \frac{\sum_{i=1}^{n} N_i}{|\sum_{i=1}^{n} N_i|}. \tag{2.3}$$

Shuangshuang *et. al* [106] comparent différentes méthodes de calcul de la normale d'un sommet. Les maillages réguliers sont souvent générés ou construits sur un échantillonnage de points obtenus à intervalle régulier. Les maillages réguliers sont de valence six. Les maillages semi-réguliers sont obtenus par subdivision uniforme de maillages irréguliers. Les maillages irréguliers sont des maillages avec un échantillonnage irrégulier et une valence quelconque. La saillance est définie comme les zones d'un maillage qui vont attirer l'attention d'un observateur humain. Lee *et. al* [74], définissent les zones de saillances d'un maillage 3D par leur courbure.

Maillages triangulaires

Les maillages 3D sont souvent représentés sous forme de facettes triangulaires liées par leurs arêtes. Formellement, un maillage M triangulaire est un complexe simplicial [28], qui est noté :

$$M = (V, S), \tag{2.4}$$

où S est un ensemble fini et non vide de sous-ensemble de V, et désigne la connectivité. Un q-simplex est un ensemble $s \in S$ tel que $q = |s| - 1$. Et on appelle q la dimension du q-simplex $s \in S$. On appelle facette un élément $t \subset s$, et plus particulièrement un 0-simplex est un sommet, et un 1-simplex une arête, comme illustré Fig. 2.3. Finalement, un maillage M est un complexe simplicial de dimension 2, composé de simplexes :

— un ensemble de sommets, les 0-simplexes de S :

$$S_0 = \{\{v_0\}; \{v_1\}; \ldots\},$$

(a) (b) (c) (d)

FIGURE 2.3: a) 0-simplex : sommet, b) 1-simplex : arête, c) 2-simplex : triangle, d) 3-simplex : tétrahèdre.

— un ensemble d'arêtes, les 1-simplexes de S :

$$S_1 = \{\{v_0, v_1\}; \{v_0, v_2\}; \ldots\},$$

— un ensemble de facettes triangulaires, les 2-simplexes de S :

$$S_2 = \{\{v_0, v_1, v_2\}; \{v_0, v_1, v_3\}; \ldots\},$$

Un sommet est un point unique, qui est associé avec un point unique de l'espace de représentation. C'est cette association qui permet la réalisation géométrique d'une forme topologique.

Maillages quelconques

Les maillages de polygones sont une généralisation des maillages triangulaires. Chaque facette $f \in F$ est un ensemble de $n > 2$ sommets $f = \{v_0, \ldots, v_n\}$, tel que $v_i \in V$. Les sommets d'une facette sont reliés successivement entre eux par des arêtes, notées $e \in E$, $e = \{v_i, v_j\}$ tel que $i \neq j$. Les facettes peuvent avoir un degré différent dans le même maillage, ou avoir le même degré comme dans les maillages quadrangulaires auquel cas les cellules sont des hypercubes. Il est aussi important de savoir manipuler de objets tels que des maillages générés automatiquement, ou tels que des nuages de points acquis par scanner. En effet, les étapes de traitement ou de maillage de ces objets font face à différents problèmes, tels que le bruit géométrique, les trous, et la non-variété. D'autre part, dans un contexte où les avancées sont rapides, il peut être important de posséder l'original d'un nuage de points ou d'une surface mal maillée.

Stockage des maillages

Nous nous intéressons ici au moyen de décrire un maillage et de le conserver. Weiler [129] a comparé plusieurs méthodes de description de maillages, chaque structure de

données est appropriée pour un traitement spécifique. Par exemple, la structure "Winged Edge" [4] permet de stocker les relations d'adjacence des arêtes. Généralement, les formats de fichiers se composent d'une liste de sommets décrits par leur localisation dans $\mathbb{R}^3$ et d'une liste de facettes quelconques. Il est alors possible de conserver des nuages de points et des soupes de polygones par exemple. Cependant, certains formats de compression nécessitent des maillages triangulaires variétés. Par exemple, la méthode "Edgebreaker" [103] utilise le fait que la surface soit orientable pour définir un ordre de parcours des facettes. Notons que transformer un maillage de polygones en maillage triangulaire est facilement réalisable. Il est également possible d'ajouter des paramètres de couleur pour chaque sommet, la couleur des facettes est interpolée en fonction des sommets qui la composent. Si l'échantillonnage est assez dense, le rendu visuel peut être suffisant. Néanmoins, une texture peut être appliquée pour avoir un meilleur rendu. Il s'agit d'une image et d'un ensemble de coordonnées de texture permettant de plaquer l'image sur le maillage. Il est également possible d'affecter des grandeurs physiques à chaque sommet pour faire de la simulation ou de la visualisation de phénomènes. Dans nos travaux, nous ne considérons pas ces paramètres d'apparence ou de comportement, puisque dans le contexte de l'insertion de données cachées, la forme est le support de l'information. D'autre part, les paramètres additionnels des maillages sont changés ou supprimés de façon simple. Cependant, il est à noter, que le contexte de visualisation, l'éclairement de la scène et les paramètres d'apparence affectent fortement notre perception sur un écran 2D ou 3D.

2.2.2 Modifications des maillages

Le terme modification désigne les changements volontaires ou involontaires, de façon malicieuse (attaque) ou non. Ces modifications peuvent avoir pour but la compression (simplification, quantification), la visualisation (lissage), la réparation (remplissage de trou), ou encore être des dégradations volontaires (ajout de bruit, découpage,...). Les principales modifications sont classées en fonction de leur type [101, 123] :

— **Transformations affines** : elles comprennent, la translation, la rotation, le changement d'échelle uniforme et leurs combinaisons. Ces transformations sont des opérations basiques de manipulation. Au contraire, d'autres transformations affines comme le changement d'échelle non-uniforme, sont considérées comme des attaques intentionnelles.

— **Ajout de bruit et lissage** : l'ajout d'un bruit sur la géométrie du maillage est souvent utilisé pour enlever, ou faire perdre la synchronisation d'une marque insérée dans un objet. Pour une attaque sans connaissance de la méthode d'insertion, on utilise souvent un bruit gaussien. Il existe également des méthodes de lissage

qui peuvent être utilisées pour améliorer la qualité visuelle d'un maillage. Ces méthodes, dont la plus connue est le lissage Laplacien [119], affectent la géométrie du maillage.

— **Attaque sur la connectivité** : les attaques sur la connectivité ne modifient pas la géométrie. Les attaques peuvent changer les relations d'adjacences entre les primitives du maillage, soit par remaillage complet, soit par retournement d'arêtes. Une attaque sur la connectivité, dite sans distorsion consiste simplement à réorganiser l'ordre des primitives dans le format de représentation.

— **Ré-échantillonnage** : le ré-échantillonnage complet crée un nouveau maillage respectant la forme topologique du premier mais en modifiant la connectivité. La subdivision du maillage est également considérée comme une attaque de ré-échantillonnage. De même que l'opération contraire, qui consiste à simplifier le maillage comme avec la méthode de contraction d'arête [42].

— **Attaques topologiques** : les attaques sur la forme d'un maillage peuvent être malicieuses comme les attaques de découpage. Le découpage est une attaque qui consiste à conserver seulement une partie du maillage. En général, la surface est refermée à la suite d'un découpage, cette opération est considérée comme de la réparation comme dans le cas du bouchage de trous par exemple.

— **Compression** : la compression avec pertes, la quantification des coordonnées des sommets par exemple est un challenge puisque beaucoup de méthodes de tatouage utilisent la position des sommets comme support du message secret. De plus, la compression n'est pas une attaque malicieuse *a priori*.

Cette liste n'est pas exhaustive mais représente la variété des modifications auxquelles peut être confronté un maillage 3D.

2.3 Évaluation des modifications

Lors d'un traitement effectué sur le maillage d'un objet 3D, il est nécessaire de pouvoir estimer les distorsions entre le maillage original et le maillage modifié. Les métriques d'évaluation de qualité avec références permettent de quantifier la qualité d'un maillage par rapport à un autre de façon objective. Généralement, ces métriques peuvent être classées en deux catégories, celles qui sont corrélées avec le système visuel humain (SVH) et celles qui ne le sont pas. Ces dernières sont généralement toujours utilisées puisque bien intégrées [19], comme la distance de Hausdorff, la racine carrée de l'erreur quadratique moyenne RMSE (root mean square error), le PSNR (pick signal-to-noise ratio) ou encore le Laplacien géométrique proposé par Karni et Gotsman [60]. Les métriques corrélées avec le SVH, sont plus récentes et reposent sur des évaluations perceptuelles. Comme la

métrique 3DWPM (3D Watermarking Perception Metric), proposé par Corsini *et. al* [21] dans le but d'évaluer leur méthode de tatouage. Une des métriques les mieux corrélées avec le SVH est le MSDM2 (Mesh Structural Distortion Measure 2) proposé par Lavoue [71] est, également, une métrique de référence grâce aux outils développés [73]. Plus récemment, Torkhani *et. al* [113] ont proposé une méthode prenant en compte l'amplitude et la direction des tenseurs de courbures. Nous présentons plus en détail quatre métriques objectives que nous avons utilisé pour évaluer la distances entre deux surfaces dans nos travaux de recherches.

Distance de Hausdorff

La distance de Hausdorff permet de comparer deux surfaces 3D. Cigoni *et. al* [19] ont introduit une première méthode pour évaluer la qualité d'un maillage simplifié. Aspert *et. al* [2] ont proposé une méthode plus efficace en temps. Ces méthodes permettent une bonne approximation de la distance par l'échantillonnage de la géométrie des modèles à comparer. Dans un premier temps, il faut définir la distance entre un point p, appartenant à une surface S, et un point p' appartenant à une autre surface S' :

$$d(p, \mathcal{S}') = \min_{p' \in \mathcal{S}'} \left\| p - p' \right\|_2 , \tag{2.5}$$

où $\|p - p'\|_2$ est la distance euclidienne dans $\mathcal{R}^3$. La distance unilatérale entre deux surfaces est alors :

$$d(\mathcal{S}, \mathcal{S}') = \max_{p \in \mathcal{S}} d(p, \mathcal{S}'). \tag{2.6}$$

Cette distance n'est pas symétrique, en général, $d(S, S') \neq d(S', S)$. La distance de Hausdorff est définie comme :

$$d_s(\mathcal{S}, \mathcal{S}') = \max(d(\mathcal{S}, \mathcal{S}'), d(\mathcal{S}', \mathcal{S})), \tag{2.7}$$

ce qui permet d'avoir une estimation plus précise.

Métrique RMSE

La métrique RMSE peut être, selon les auteurs, une distance surface à surface [2]. Dans ce cas, elle est définie à l'aide de la distance point-surface (Equation 2.5) comme :

$$RMSE(\mathcal{S}, \mathcal{S}') = \sqrt{\frac{1}{|\mathcal{S}|} \int \int_{p \in \mathcal{S}} d(p, \mathcal{S}')^2 \mathrm{d}\mathcal{S}}. \tag{2.8}$$

Cette fonction n'étant pas symétrique, on définit généralement une valeur symétrique MRMSE (maximum RMSE) :

$$MRMSE(\mathcal{S}, \mathcal{S}') = \max(RMSE(\mathcal{S}, \mathcal{S}'), RMSE(\mathcal{S}', \mathcal{S})). \tag{2.9}$$

D'autres auteurs [20, 15] proposent une version approximée de la métrique RMSE calculée sur la position des sommets ou sur la valeur des normales des sommets entre deux maillages M et M' ayant la même connectivité :

$$RMSE_v(M, M') = \sqrt{\frac{1}{|V|}\sum_{1}^{|V|} \|v_i - v_i'\|_2^2}, \tag{2.10}$$

où $v_i \in M$ et $v_i' \in M'$ respectivement :

$$RMSE_n(M, M') = \sqrt{\frac{1}{|V|}\sum_{1}^{|V|} \langle n_i, n_i' \rangle^2}, \tag{2.11}$$

où $n_i \in M$, $n_i' \in M'$ et $\langle \cdot, \cdot \rangle$ représente le produit scalaire.

Métrique PSNR

Le PSNR étant souvent utilisé comme métrique de référence en 2D, certains auteurs l'utilisent en 3D. Il existe deux versions définies par Chao *et. al* [15], une quantifiant la distorsion sur la position des sommets, l'autre la distorsion sur les normales des sommets. Le PSNR entre deux maillages M et M' est alors défini respectivement, comme :

$$PSNR_v(M, M') = 20 \log_{10} \frac{D_{max}}{RMSE_v(M, M')}, \tag{2.12}$$

$$PSNR_n(M, M') = 20 \log_{10} \frac{D_{max}}{RMSE_n(M, M')}, \tag{2.13}$$

où D_{max} est la longueur de la diagonale de la boîte englobante du maillage de référence. Cette métrique n'échantillonne pas toute la surface, mais donne une bonne approximation quand elle est utilisée par exemple en stéganographie [15] puisque les déplacements sont très faibles et que l'appariement des points est immédiat.

Métrique MSDM

Les distances précédentes ont comme principal défaut de ne mesurer qu'une distorsion géométrique sans prendre en compte les aspects visuels. En effet, la visibilité

des distorsions sur des maillages dépend de la structure locale du maillage. Clairement, ajouter un bruit dans une zone texturée du maillage (comme des cheveux) ne va pas beaucoup affecter la perception pour un observateur. Au contraire, ajouter un bruit dans un plan conduit à une mauvaise appréciation du maillage 3D. Plus généralement, les modifications du maillage qui ont pour but de ne pas laisser de traces doivent conserver localement les structures, comme les zones lisses. Pour répondre à ce problème, la métrique MSDM (Mesh Structural Distortion Measure), proposée par Lavoué *et. al* [70], transpose la métrique 2D classique SSIM [128], aux maillages 3D. Les auteurs définissent une métrique locale $LMSDM$ entre deux fenêtres locales a et b appartenant respectivement à M et à M' :

$$LMSDM(a,b) = \left(0.4 \times L(a,b)^3 + 0.4 \times C(a,b)^3 + 0.2 \times S(a,b)^3\right)^{\frac{1}{3}}, \tag{2.14}$$

où L, C, S représentent les fonctions de comparaison de courbures, de contrastes et de structures [70]. Finalement, la métrique MSDM est calculée, comme la somme de Minkovski des n_w distances locales :

$$MSDM(M,M') = \left(\frac{1}{n_w}\sum_{j=1}^{n_w} LMSDM(a_j,b_j)^3\right)^{\frac{1}{3}}. \tag{2.15}$$

Une extension multi-échelle, appelée MSDM2, a été proposée par Lavoué *et. al* [71]. Elle permet l'appariement des points de maillages ne partageant pas la même connectivité.

2.4 Ordonnancement 3D

Les maillages 3D ne possèdent pas de structures permettant un ordonnancement trivial des primitives. Une étape d'ordonnancement des éléments que l'on souhaite parcourir est nécessaire. Les formats de fichiers de maillages 3D permettent généralement de stocker la géométrie et la topologie d'un maillage. Puisque les maillages 3D sont non structurés, la position de ses éléments dans un fichier n'affecte pas le rendu. Il est alors nécessaire d'avoir un ordre de parcours des éléments d'un maillage ne dépendant pas du format de stockage. La "traversée" d'un maillage est utilisée dans différents traitements, comme la compression, la visualisation, ou l'insertion de données cachées. Selon l'utilisation, les auteurs choisissent de définir cet ordre à partir de la topologie du maillage ou de la géométrie. L'ordonnancement 3D est un challenge puisque contrairement au traitement d'images, ou aux représentations par voxels, le maillage n'est pas plongé dans une grille 2D ou 3D régulière.

2.4.1 Ordonnancement par traversée

L'ordonnancement en fonction de la topologie a été développé, dans un premier temps, pour des systèmes de compression de maillage triangulaire. Les premières méthodes définissent un ordre de parcours des triangles, comme la méthode "Edge-breaker" proposée par Rossignac [103]. Cet algorithme effectue la même traversée du maillage en passant de triangle en triangle, par leur lien d'adjacence. Ces relations d'adjacence sont notées par des symboles, ce qui permet un codage entropique. L'utilisation de maillage triangulaire manifold et orientable permet de parcourir les facettes. Ce parcours a été proposé initialement par Ohbuchi *et. al* [90] et est appelé TSPS (Triangle Strip Peeling Sequence). Ce genre d'ordonnancement a été utilisé en tatouage par Mao *et. al* [83], par Cayre *et. al* [12] et Bajaj *et. al* [3] qui déroulent les facettes en spirale. Cette méthode est très sensible aux modifications de la connectivité du maillage comme le basculement d'arêtes pour des maillages triangulaires. Pour traverser les facettes d'un maillage à partir d'une facette donnée, Lin *et. al* [77] utilisent un parcours en largeur, l'ordre de sélection des facettes et des sommets est donné par une clé secrète. Huang and Tsai [48], proposent une autre traversée basée sur le parcours en largeur mais en utilisant une analyse en composantes principales pour sélectionner la première facette et l'ordre de parcours.

2.4.2 Ordonnancement de patchs

Luo and Bors [82] ont proposé une méthode de tatouage basée sur le partitionnement du maillage en régions à distance géodésique égale. À partir d'un sommet donné, les distances géodésiques avec les autres sommets sont calculées. Chaque bande de sommets est utilisée comme support pour insérer un bit. L'ordre entre les patchs est immédiatement donné par la distance au sommet d'entrée. La méthode de tatouage de Wang *et al.* [125] génère des patchs cylindriques sur le maillage qui sont ordonnés par leur localisation dans l'espace.

2.4.3 Ordonnancement basé sur les graphes

Certains auteurs considèrent les maillages comme des graphes pour l'étape d'ordonnancement. En effet, à un maillage complexe $M = (V, F, E)$ est associé un graphe $G = (V, E)$ (Définition.2.1). Le maillage étant une réalisation géométrique de cette connectivité topologique, le graphe pondéré $G = (V, E, \omega)$ est son graphe associé, où $\omega : E \to \mathbb{R}^+$. Plus généralement, si l'on considère un graphe complet $G' = (V, E')$ sur les sommets du maillage, alors $\forall i, j,\ i \neq j,\ e_{v_i,v_j} \in E'$ et en particulier $E \subseteq E'$. Le

(a) (b)

FIGURE 2.4: a) Nuage de 3006 points 3D, b) ACPME construit sur le nuage de points.

parcours de graphes est un problème complexe, nous présentons deux types de parcours, les arbres couvrants et les chemins Hamiltoniens.

2.4.3.1 Arbres couvrants

Un arbre couvrant sur un graphe est un sous graphe connecté et acyclique inclus dans ce graphe qui connecte tous ses sommets. Il est possible de définir un ordre en partant d'un sommet et en faisant un parcours de l'arbre en largeur ou en profondeur par exemple. Les arbres couvrants sont utilisés en compression de nuages de points, puisqu'ils permettent de définir une structure sur les sommets sans avoir à effectuer une étape de maillage, puis une étape de compression qui au final sera plus coûteuse en temps. Gumhold *et. al* [40] et Merry *et. al* [86] ont proposé des méthodes similaires de construction itérative d'un arbre couvrant sur un nuage de points. L'arête à ajouter au sous-arbre est choisie pour produire le plus petit résidu de prédiction pour la méthode de compression. Les ACPMs (Arbres Couvrants de Poids Minimums), sont un problème bien connu dans la théorie des graphes, notamment au travers des algorithmes de Kruskal [66] et de Prim [98]. La Fig. 2.4.a présente un nuage de points de l'objet 3D "horse", et l'ACPM unique construit dessus, Fig. 2.4.b. Amat *et. al* [1] ont proposé une méthode de tatouage fragile. L'ordonnancement se fait par le calcul de l'ACPM d'un nuage de points puis grâce à un point d'entrée et un schéma de balayage, ils définissent un ordre de parcours de l'arbre. Tournier *et. al* [114] ont proposé de construire un ACPME (Arbres Couvrants de Poids Minimums en distance Euclidienne) robuste dans un nuage de points, dans le but de définir un ordre robuste au déplacement des sommets.

2.4.3.2 Chemin hamiltonien

Les chemins hamiltonien sont des chemins, qui passent une et une seule fois par tout point du graphe [43]. Un graphe hamiltonien est un graphe comportant un chemin

(a) (b)

FIGURE 2.5: a) Nuage de 3006 points 3D, b) Chemin Hamiltonien construit sur le nuage de points.

hamiltonien. Il n'existe pas de condition nécessaire et suffisante pour trouver un chemin hamiltonien, mais il existe de nombreuses conditions suffisantes données par exemple par le théorème de Dirac (1952) qui dit qu'un graphe simple avec $n \geq 3$ sommets est hamiltonien si le degré de chaque sommet est supérieur ou égal à $n/2$. Plus généralement, le théorème de Koenig-Redei dit qu'un graphe complet est hamiltonien.

En considérant, le graphe complet construit sur l'ensemble des sommets du maillage illustrés Fig. 2.5.a, la Fig. 2.5.b présente le chemin hamiltonien construit sur ce graphe, en commençant par un point donné. Trouver un chemin hamiltonien minimal dans un graphe complet est également un problème complexe qui a été prouvé NP-complet, c'est le problème du voyageur de commerce. Cependant trouver un chemin Hamiltonien quelconque dans un graphe complet est un problème simple, il permet d'obtenir un ordre de façon efficace dans un nuage de points en fonction de l'heuristique de choix du sommet à connecter. En compression, Gurung *et. al* [41] ont proposé une méthode d'ordonnancement des triangles en suivant un chemin hamiltonien à la façon de "Edgebreaker". Zhang *et. al* [139] ont amélioré cette approche avec un meilleur taux de compression. Les chemins hamiltoniens dépendent du coût donné à chaque arête. Pour un graphe dans l'espace $\mathbb{R}^3$, la distance euclidienne est souvent l'option retenue. Il est alors possible de connecter un sommet à un autre non visité tel que le coût soit minimal. Notons que le choix du coût minimal est arbitraire, il pourrait être maximal.

2.5 Conclusion

Dans cette section nous avons défini les représentations 3D que nous allons étudier, les maillages de polygones et les nuages de points. Nous avons listé leurs propriétés ainsi que les manipulations usuelles auxquelles elles sont soumises. Nous avons présenté des méthodes d'évaluation des distorsions d'un maillage 3D modifié par rapport à un maillage de référence. Ces métriques sont utilisées pour évaluer et valider nos résultats

expérimentaux. Finalement, un état de l'art des techniques d'ordonnancement a été proposé afin de comparer les différentes façons de définir un ordre sur les primitives d'un maillage, et celles qui permettent de définir un ordre sur des représentations non-structurées comme les nuages de points. Ce dernier point nous intéresse dans nos travaux présentés dans les contributions, notamment les traversées à l'aide de chemins hamiltoniens construits sur le graphe complet des sommets.

Chapitre 3

Insertion de données cachées dans un support numérique visuel

3.1 Introduction

L'insertion de données cachées (IDC) permet d'insérer dans un support numérique de l'information additionnelle de façon imperceptible tout en respectant le format initial. Cette information, peut être par exemple un message secret, des méta-données, un identifiant ou une marque. L'IDC consiste donc à modifier un support numérique pour ajouter de l'information. En effet, un média dans lequel est insérée de l'information doit pouvoir être visualisé ou manipulé à l'aide de logiciels standards dans un format de fichier standard. La taille du fichier de stockage doit être préservée pour éviter un surcoût de mémoire ou de bande passante. De plus, dans certains scénarios d'attaque, l'augmentation de la taille d'un fichier en fonction du message peu être suspecte. Certaines méthodes utilisent les parties commentaires pour insérer le message, ce qui ne permet pas de marquer le contenu visuel en lui même. Dans ce cas un changement de format peut supprimer le message caché. Dans ce chapitre nous nous focalisons donc sur les méthodes qui modifient le contenu visuel de façon imperceptible pour insérer des données cachées.

Dans la Section 3.2 nous définissons les principes et propriétés sur lesquelles reposent l'IDC. Ensuite, nous abordons différentes méthodes d'IDC : la stéganographie dans la Section 3.3, l'insertion haute capacité dans la Section 3.4 et les méthodes de protection de droits d'auteur dans la Section 3.5. Nous introduisons, dans la Section 3.6, comment évaluer la sécurité de ces méthodes d'IDC.

3.2 Principes et propriétés

L'insertion de données cachées consiste à dissimuler, de façon imperceptible, des données dans un support hôte. Le livre de Cox *et. al* [23] pose clairement le contexte de l'IDC, les méthodes comportent deux modules principaux, l'insertion et l'extraction du message. Une étape de synchronisation est nécessaire pour définir le même ordre entre les étapes d'insertion et d'extraction. Dans la Fig. 3.1, une image est le support d'un message secret, le module d'insertion, permet de produire une image marquée, proche de l'originale l'image support.

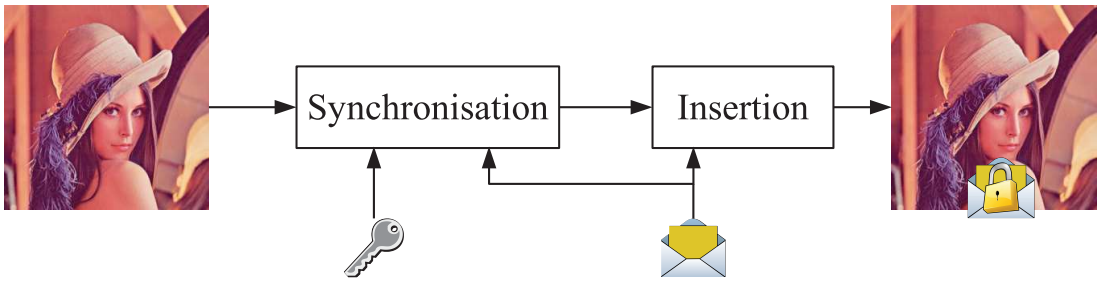


FIGURE 3.1: Schéma d'une méthode d'insertion de données cachées, l'image est marquée avec un message secret à l'aide d'une clé secrète.

Dans un premier temps, une clé secrète est utilisée pour sécuriser l'étape de synchronisation et définir un ordre sur les pixels ou des zones choisies pour l'insertion. Une fois les pixels support définis, l'étape d'insertion permet de cacher le message dans l'image support. Par exemple, la méthode la plus classique consiste à modifier les bits de poids faibles codant les pixels. Notons que généralement le message est chiffré en utilisant la clé secrète. Le module d'extraction est illustré dans la Fig. 3.2, et permet d'extraire le message caché grâce à la clé qui a été utilisée à l'insertion.

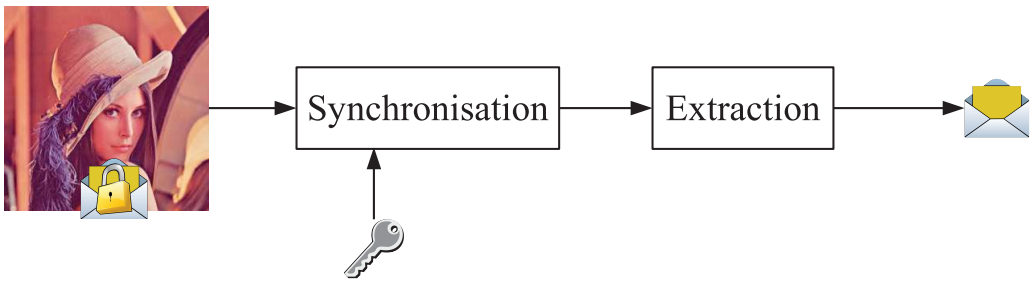


FIGURE 3.2: Schéma d'une méthode d'extraction de données cachées, l'information cachée est retrouvée suivant l'ordre donné par la synchronisation et la clé secrète.

Le message peut être reconstruit en deux étapes qui consistent à retrouver les zones d'insertions, puis à extraire les bits dans l'ordre. Certaines méthodes utilisent un vote

majoritaire pour reconstruire le message est ne nécessitent alors pas d'étape de synchronisation au décodage [127]. Cependant, ce type de méthodes souffre d'un manque de sécurité comme l'ont démontré He et Zhang [44]. Les méthodes d'IDC doivent également reposer sur le principe de Kerckhoffs [63], qui dit que le secret d'une méthode ne peut pas dépendre du secret de l'algorithme mais doit reposer uniquement sur le secret d'une clé. Ainsi, dans le contexte de l'insertion de données cachées, la méthode est supposée connue de tous, l'extraction des données dépend uniquement de la connaissance de paramètres secrets utilisés comme clé. Cette idée s'oppose au principe de la "sécurité par l'obscurité" qui suppose que l'algorithme et son implémentation restent secrets.

Les différentes classes d'IDC sont présentées dans la Fig. 3.3. Elles sont organisées de la plus générale, aux plus spécifiques, nous les décrivons par la suite. Chacune est définie en fonction d'un scénario d'utilisation de l'IDC.

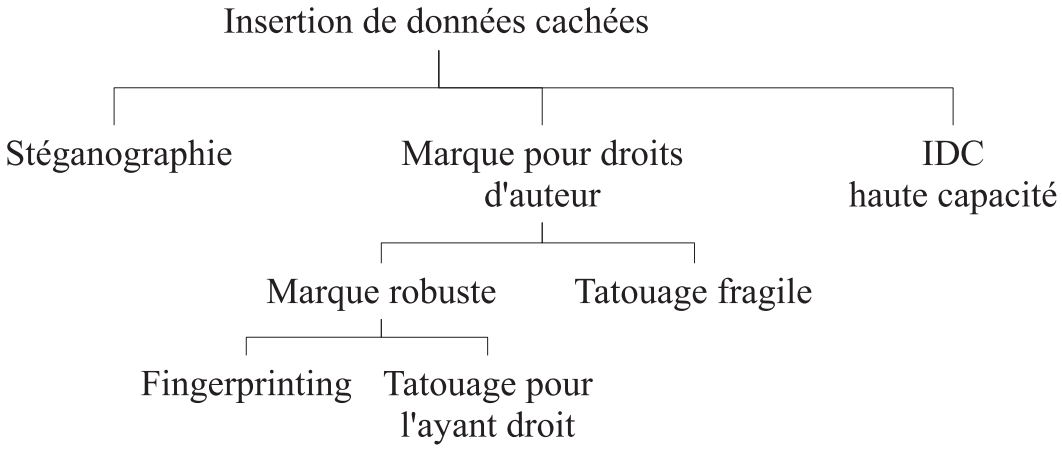


FIGURE 3.3: Classification des méthodes d'insertion de données cachées, basée sur les travaux de [95, 17].

Chaque méthode requière différentes propriétés pour la méthode d'insertion, certaines sont robustes, d'autres fragiles et avec un plus ou moins grande capacité. Ces caractéristiques représentent les principales qualités d'une méthode d'IDC. Ces méthodes doivent avoir une clé secrète afin de respecter le principe de Kerckhoffs. Il existe des propriétés inhérentes aux systèmes d'IDC, l'imperceptibilité, la robustesse, la capacité et la sécurité. Nous détaillons ces différentes propriétés par la suite, mais il est à noter que ces propriétés sont liées.

En effet, les méthodes d'IDC font face à un compromis entre la robustesse, la capacité, la sécurité et l'imperceptibilité. En général, améliorer une de ses propriétés fait décroitre les autres. Il faut donc faire un compromis, comme illustré Fig. 3.4, qui est dicté par le scénario considéré, Fig. 3.3. Dans la majorité des cas, la marque doit être imperceptible pour la manipulation, le compromis se fait alors entre la capacité, la robustesse, la sécurité et une imperceptibilité statistique définie Section 3.3.

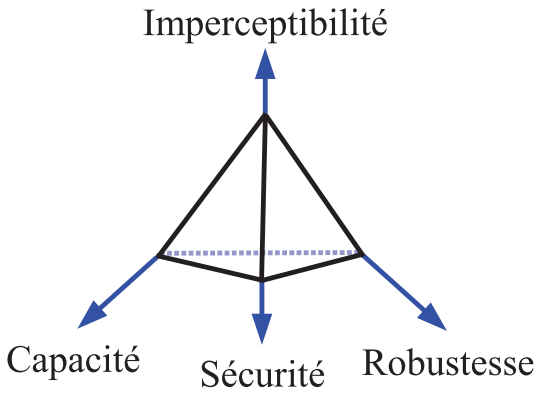


FIGURE 3.4: Schéma des compromis.

Souvent le compromis se résume à capacité/robustesse ou robustesse/sécurité comme avec les techniques d'insertion par étalement de spectre [11]. Plus généralement, le compromis dépends de la technique utilisée et du scénario d'utilisation. Dans la suite de ce chapitre, nous présentons les propriétés de l'IDC et les cas d'utilisation.

Insertion et extraction

Il existe plusieurs algorithmes d'insertion. Généralement, les auteurs séparent les méthodes dans le domaine spatial : substitution, étalement de spectre, statistique, des méthodes dans les domaines transformés (ondelettes, DCT, DFT,...). Le Tableau 3.1 présente les principales différences entre les deux domaines d'insertion.

Facteurs	Domaine spatial	Domaines transformés
Coût de calcul	Faible	Important
Robustesse	Faible	Plus robuste
Qualité perceptuelle	Contrôlable	Peu de contrôle
Complexité	Faible	Haute
Temps de calcul	Faible	Plus important
Capacité	Haute	Moindre

TABLEAU 3.1: Comparaison entre une insertion dans le domaine spatial et dans un domaine transformé.

Nous constatons que les méthodes dans le domaine spatial correspondent plutôt à des scénarios d'insertion haute capacité, tandis que les méthodes dans les domaines transformés sont adaptées aux scénarios de tatouage robuste. La sécurité n'est pas prise en compte ici puisqu'il existe des méthodes sécurisés dans les deux cas. Les méthodes d'insertion pour l'IDC utilisent différentes stratégies classées en plusieurs catégories de techniques. L'insertion peut se faire par :

— **Injection** : le message est inséré directement dans le média, ce qui provoque une augmentation de la taille du support. Ce comportement est une faille de sécurité par rapport à un potentiel attaquant.

— **Substitution** : le message est inséré de façon à remplacer l'information redondante du support ou à substituer une partie de l'information qui altère le moins le support. Cette technique est la plus utilisée.

— **Distorsion** : l'extraction se fait en analysant cette différence entre les objets supports et les objets marqués.

Souvent, les méthodes par distorsion nécessitent l'objet support (méthode non aveugle) ce qui est rarement envisageable dans des cas pratiques. Les techniques par substitution sont les plus utilisées. Ce sont celles que nous retiendrons pour nos recherches. Nous présentons deux grandes classes de méthodes d'insertion. Les méthodes d'étalement de spectre (Spread Spectrum), popularisées par Cox *et. al* [24], insèrent un message à partir d'une combinaison linéaire du signal hôte avec un signal de bruit modulé par le signal à insérer. Une approche de la théorie de l'information consiste à voir l'IDC comme une communication avec information adjacente. Théoriquement, ce point de vue est décrit par les codes "Dirty Paper", dont une implémentation pratique a été apportée par Chen et Wornell [16]. Leur méthode QIM (Quantization Index Modulation) est une procédure qui définie des fonctions de quantification, dont on peut voir un exemple Fig. 3.5.

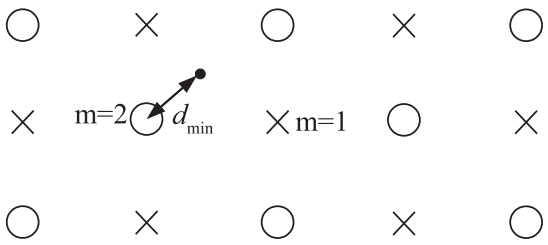


FIGURE 3.5: Illustration de la méthode QIM proposée par Chen et Wornell [16].

Un message $m \in \{1, 2\}$, nécessite deux fonctions de quantification, leurs ensembles sont représentés par les points $\times$, si $m = 1$ et les points $\bigcirc$ si $m = 2$. Le signal hôte $\bullet$ est quantifié avec le quantificateur le plus proche selon le message. La distance d_{min} mesure la robustesse aux perturbations. Le signal tatoué est alors transmis par un canal de communication. Le receveur doit estimer le message en utilisant les quantificateurs de l'encodeur. Le quantificateur utilisé à l'encodage est déduit comme étant celui qui produit le moins de distorsion. Le message décodé est celui correspondant au quantificateur prédit. Un des systèmes de tatouage dérivés du schéma QIM, est le SCS (Scalar Costa Scheme) proposé par Eggers *et. al* [29].

Méthode aveugle

Une méthode est dite “aveugle” si l'extraction du message ne nécessite pas d'information sur le maillage original. Les méthodes non-aveugles sont plus robustes, néanmoins la plupart des applications ne permettent pas l'accès à l'objet original à cause des problèmes de sécurité. Par exemple, les techniques d'IDC par distorsion sont rarement aveugles, puisqu'elles nécessitent d'avoir le média support pour calculer la différence du média marqué avec celui-ci. Cependant, ces méthodes sont plus simples puisque la synchronisation se fait par recalage. De plus, elles peuvent être utilisées avec une base de données contenant les objets supports originaux. Dharwadkar *et. al* [27] proposent une méthode de tatouage d'images non aveugle et robuste pour des images couleurs. Dans le cas d'images en niveaux de gris, Yamasaki *et. al* [131] proposent une méthode non aveugle qui ne nécessite pas l'image originale mais juste des descripteurs SIFT (Scale-Invariant Feature Transform) localisés. Ce type de méthode peut être qualifié de semi-aveugle. Dans le cadre du tatouage de maillages 3D, Garg *et. al* [37] détectent les modifications insérées sur la norme des sommet, par différence avec les valeurs du maillage original. Bien que des méthodes non aveugles sont utilisées pour des applications très spécifiques, nous considérons par la suite uniquement les méthodes aveugles qui possèdent plus d'avantages pour nos travaux.

Robustesse

La robustesse qualifie le degré de résistance d'un message caché à une modification du support. Cette modification peut être maligne et viser à détruire le message (suppression ou désynchronisation du message, ...) ou une manipulation standard (changement de format, compression, erreur de communication, ...). Les modifications destructrices du message caché sont appelées attaques et sont généralement menées par des pirates soupçonnant ou démasquant le fait que le support contient un message. Pour les maillages 3D, les modifications possibles sont listées Section 2.2.2, et correspondent souvent à des manipulations qui n'ont pas pour but d'attaquer l'objet. Une méthode est dite robuste à une attaque quand le message peut être extrait du média marqué après modifications avec une certaine précision. La qualité du message extrait est mesurée en terme de taux d'erreurs binaires, noté BER (Bit Error Rate). Le BER est le taux d'erreurs binaires, il est calculé comme le rapport entre le nombre d'erreurs NE binaires et le nombre de bits du message m :

$$BER = \frac{NE}{|m|}, \tag{3.1}$$

où NE est le nombre d'erreurs binaires entre un message original m et m' un message possiblement modifié, calculé comme :

$$NE = |m| - |m'| + \sum_{i=0}^{|m|-1} \begin{cases} 1 \ si & m_i \neq m'_i \\ 0 \ sinon & \end{cases}. \tag{3.2}$$

En 2D les méthodes robustes atteignent de bonnes performances comme celle proposée par Lin *et. al* [78] qui est une méthode de tatouage d'image, robuste à la rotation, à la dilatation ou à la translation. Ces attaques constituent une bonne diversité des transformations affines possibles. Au contraire, un des principaux défis pour les maillages 3D réside dans le fait qu'il est plus facile, dans le cadre d'une attaque volontaire, d'utiliser des méthodes classiques de manipulation pour perdre la marque, que d'avoir à attaquer la sécurité (Section 3.6).

Capacité

La capacité indique la taille limite de la charge utile (payload) binaire pouvant être insérée par éléments du média. Elle est par exemple exprimée en nombre de bits par pixel (bpp) pour une image et en nombre de bits par sommet (bps) pour un maillage 3D. Le Tableau 3.2, présente les capacités généralement produites par les différentes méthodes d'IDC. Les capacités sont données en nombre de bits par maillage, en moyenne de la littérature ou en fonction de paramètres. La robustesse est exprimée de façon relative avec des valeurs de "- -" pour la moins robuste à "+++" pour la plus robuste.

Tableau 3.2: Capacité en fonction de la méthode d'IDC

Méthode	Capacité	Robustesse
0-bits	1 bit	+++
Tatouage	identifiant : 64, 128 bits	++
Fingerprinting	borne minimale par le nombre d'utilisateurs	+
Tatouage fragile	max	-
Stéganographie	borne maximale pour rester indétectable	- -
Haute capacité	max	- -

Nous constatons dans le tableau 3.2 que le compromis entre la capacité et la robustesse est bien présent. Les auteurs cherchent à obtenir le meilleur compromis entre la capacité et la robustesse tout en maintenant un haut niveau d'imperceptibilité. L'idée du tatouage 0-bit est qu'en réduisant la capacité au minimum (objet marqué ou non), la robustesse atteigne alors son maximum. Furon [33] décrit les applications et le cadre d'un tel système. Au contraire, certaines méthodes de vérification d'intégrité comme le tatouage fragile cherchent à détecter tout changement possible, la marque doit donc être

affectée par les manipulations du média que le système doit détecter. Par exemple, la méthode de Kundur et Hatzinakos [67] permet de détecter les zones modifiées d'une image. Le fingerprinting consiste à insérer l'identifiant des utilisateurs pour trouver les pirates à l'aide de codes de traçage de traitre (définis Section 3.5). La capacité du système dépend du nombre d'utilisateurs, du nombre de collusions considérées et de l'acceptation de faux négatifs, ces bornes sont souvent théoriques. Cependant, Furon et Desoubeaux [35], comparent des décodeurs récents dans un cadre opérationnel et montrent que la longueur théorique d'un code permettant d'éviter d'accuser un innocent est trop grande par rapport au besoin réel. Les méthodes précédentes requièrent une capacité faible pour espérer une bonne robustesse. Au contraire, la stéganographie et l'IDC haute capacité augmentent le plus possible la capacité en considérant un minimum de robustesse.

Imperceptibilité

L'imperceptibilité est la propriété qui nécessite d'être maintenue le plus possible à un haut niveau. Cox *et. al* [23] souligne l'ambiguïté du terme "cachée", il peut référer soit au fait de rendre l'information insérée imperceptible (IDC) ou à conserver secrète sa présence (stéganographie). Par exemple, en stéganographie, l'objet marqué ne doit pas être suspect. Concernant, l'IDC il est important de préserver l'imperceptibilité pour l'utilisateur et de ne pas affecter le traitement numérique des objets. L'imperceptibilité est étudiée et validée à l'aide de métriques visuelles qui quantifient la qualité d'un média : vidéo, image ou maillage 3D (voir Section 2.3). Il existe des métriques absolues mais dans les scénarios d'IDC les métriques relatives peuvent comparer la qualité du média marqué par rapport au média support. L'imperceptibilité visuelle dépend de nombreux paramètres comme les conditions de visualisation. L'imperceptibilité ne se limite pas à l'impression de l'utilisateur, il faut également que les distorsions ne perturbent pas la manipulation ou le traitement numérique d'un média. Notamment, même si la méthode d'insertion s'est focalisée sur les zones visuellement peu sensibles, pour l'impression 3D, les distorsions peuvent être inacceptables. Dans le cas de la stéganographie, l'imperceptibilité est analysée d'un point de vue statistique. Nous parlons alors de stéganalyse, comme détaillé Section 3.3.

Réversibilité

Les techniques réversibles permettent de retrouver intégralement l'objet original après extraction du message. Les méthodes de tatouages réversibles sont par exemple utilisées en imagerie médicale, pour des applications militaires ou en CAO. En effet, ces

types de données sont en général acquis avec une très bonne résolution et requièrent une grande précision dans leur manipulation. L'IDC, même avec de faibles distorsions dues à l'insertion, peut avoir un effet néfaste sur le traitement, l'interprétation ou l'impression d'objets 3D par exemple. Tian [111] a défini une technique d'IDC fragile dans des images en niveaux de gris par expansion de différence. Elle permet une complète réversibilité et une grande capacité en exploitant les différences entre pixels voisins. Ni *et. al* [89] insèrent un message en décalant l'histogramme d'une image en niveaux de gris codés sur 8 bits, tout en maintenant une bonne qualité de l'image. La capacité est de maximum 0.3 bit par pixel. Le tatouage réversible est une propriété des méthodes présentées dans la Fig. 3.3. Il possède ses propres caractéristiques, bien que pouvant avoir différents scénarios d'utilisation. En effet, les dégradations produites par l'insertion de données sont corrigées au décodage, et généralement ces méthodes sont difficiles à sécuriser. Cependant, une des caractéristiques les plus intéressantes des techniques réversibles est qu'il est possible d'appliquer plusieurs itérations d'une méthode pour augmenter la capacité, tout en conservant des distorsions acceptables. En effet, les itérations étant réversibles, il est possible de revenir au média support.

Attaques sur la robustesse et la sécurité

Les attaques sont des opérations ou des manipulations d'un média marqué qui peuvent affecter la robustesse ou la sécurité de la marque insérée. Ces attaques sont soit involontaires, soit malicieuses c'est-à-dire que l'attaquant soupçonne que le média est marqué. Elles sont classées en quatre catégories par Voloshynovskiy *et. al* [120] :

— les attaques de suppression qui ont pour but de supprimer la marque.

— les attaques géométriques qui n'ont pas pour but de supprimer la marque elle même, mais de faire perdre la synchronisation des données cachées.

— les attaques cryptographiques qui ont pour but d'extraire le message ou les paramètres secrets utilisés pour l'insertion.

— les attaques de protocole qui ont pour but d'attaquer le concept de tatouage en lui même.

Les deux premières catégories correspondent aux attaques sur la robustesse et les deux suivantes concernent la sécurité. Les attaques sur le fichier ou le format dans lequel est stocké le maillage sont des attaques de suppressions. En effet, certaines méthodes de stéganographie 3D sont sensibles au réordonnancement des primitives d'un maillage ou à un changement de format de stockage. Comme présenté Section 2.2.2, la manipulation d'un maillage est considérée comme une attaque qui peut être classée dans une des catégories décrites précédemment. Du fait de l'expansion récente du champ de recherche sur l'IDC dans les maillages 3D, la plupart des méthodes d'IDC cherchent à résister aux

attaques géométriques contre la synchronisation. En effet, elles sont plus simples à mettre en place que les attaques consistant à essayer de supprimer la marque ou à l'extraire [12]. Les attaques de protocoles mettent en défaut certains scénarios considérées par l'IDC en général. Par exemple, Kutter *et. al* [68] montrent qu'il est possible de tatouer une image avec la marque d'une image déjà marquée, ce qui pose des problèmes lorsque la marque est utilisée pour de la protection du droit d'auteur. Finalement, le changement de représentation d'un média n'est que partiellement étudié comme le passage d'une représentation surfacique à une représentation volumique par exemple. Selon le scénario de l'IDC, il faut considérer un type d'attaques adapté auquel la méthode doit faire face.

3.3 Communication secrète

3.3.1 Stéganographie

La stéganographie [1] est l'art de cacher un message secret dans un média hôte de façon à ce que ce média paraisse inchangé. La stéganographie est "définie comme un procédé de communication indétectable d'un message, dans un objet de couverture" [31]. Ce qui s'oppose à l'IDC, pour laquelle le support a de la valeur en lui même.

Dans un système de communication, les méthodes à clés peuvent être à clés secrètes ou à clés privées/publiques. Les méthodes à clés secrètes requièrent un canal sécurisé pour l'échange de la clé, si celle-ci doit être transmise. Les méthodes à clés privées/publiques supposent que chaque personne a une clé privée qu'il conserve et une clé publique qu'il peut partager. Par exemple, Bob pour envoyer un message à Alice va utiliser la clé publique d'Alice et Alice sa clé privée pour extraire ou décoder le message. Ces méthodes sont généralement plus complexes et longues, néanmoins la clé d'un système à clés secrètes peut être échangée grâce à une méthode de type clés privées/publiques sur un canal non sûr. La Fig. 3.6 présente le schéma des méthodes de stéganographie sous forme d'un système de communication.

Pour illustrer ce problème, Simmons [107] pose le problème des prisonniers, dans lequel Alice et Bob sont deux prisonniers qui souhaitent échanger des messages. Évidemment les échanges sont contrôlés par une gardienne, Eve. Si le message est chiffré ou semble suspect il n'est pas remis au destinataire. Alice et Bob créent ainsi un canal caché, en dissimulant le message dans un support anodin. Les conversations sont alors autorisées à circuler sur le canal de communication. Ce canal peut être actif, les conversations sont altérées systématiquement, ou passif dans le cas contraire. Les méthodes

1. Du grec steganos dissimulé et graphy écriture

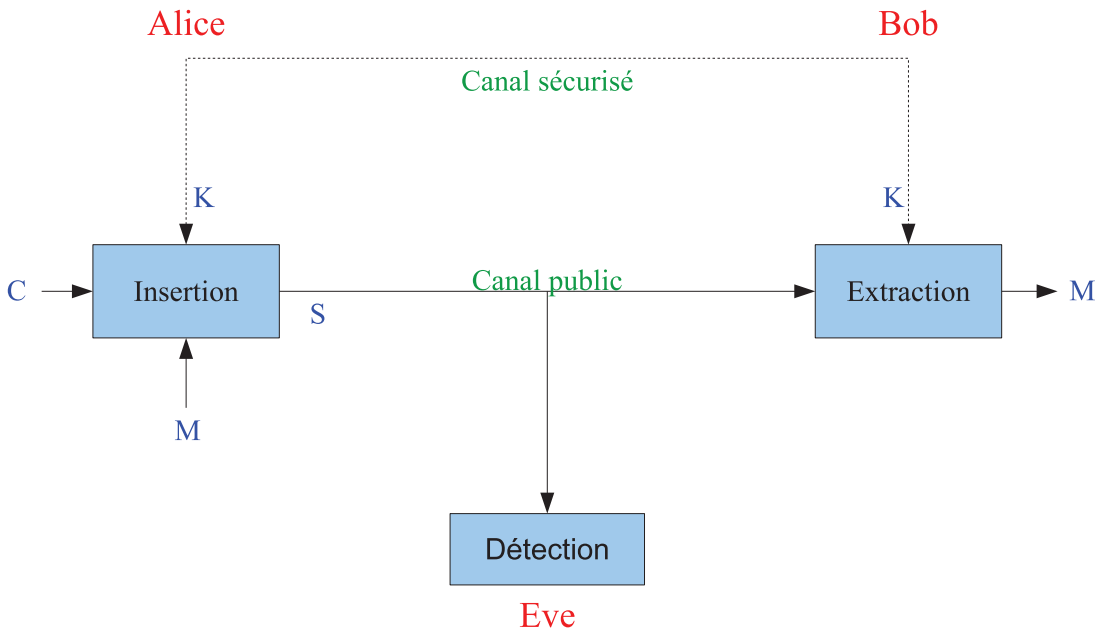


FIGURE 3.6: Schéma d'une méthode de stéganographie, le média support C (cover), le stego-média S, le message secret M, et la clé secrète K (key).

de stéganographie doivent être imperceptibles aussi bien visuellement que statistiquement, plutôt que robustes. Les techniques d'insertion présentées précédemment peuvent être utilisées pour la stéganographie. Il existe cependant d'autres méthodes d'insertion spécifique, les insertions par :

— **Sélection**, Le message est inséré dans un média support sélectionné depuis une base, de façon à minimiser le taux de détectabilité.

— **Génération**, un média marqué est construit autour du message.

Le problème principal des méthodes par génération est la difficulté de créer un média support réaliste. Les méthodes par sélection sont coûteuses en temps de recherche, cependant elles offrent une bonne résistance aux attaques. Par exemple, les méthodes par substitution des LSB (Least Significant Bit), sont basées sur le fait que modifier les bits de poids faible qui codent les pixels d'une image par exemple est imperceptible visuellement. Neeta *et. al* [88] évaluent la technique sur les formats *png* et *bmp*. Il existe d'autres méthodes adaptatives pour l'image, qui se veulent plus sûres comme HUGO (Highly Undetectable steGO) [96], S-Uniward [46], ou ASO (Adaptive Steganography by Oracle) [65].

3.3.2 Stéganalyse

La stéganalyse est l'art de déceler la présence ou non d'un message secret caché dans un média sans pour autant pouvoir lire le message. Dans ce cas, il ne suffit plus

de considérer un gardien seulement passif ou actif, il faut le supposer malicieux. En effet, s'il suspecte un échange entre deux prisonniers, il est plus intéressant de pouvoir prouver que le média partagé contient un message et le cas échéant d'essayer de l'extraire. La surveillance active est rarement la meilleure solution car très couteuse. De plus, la dégradation effectuée par la surveillance active du média visuel peut perturber les échanges classiques. Notons que pour un gardien il est préférable de pouvoir espionner discrètement une communication secrète si celle-ci est avérée. Il peut alors analyser un contenu suspect afin d'essayer d'extraire le message caché et d'estimer les paramètres d'insertion. Si la méthode n'est pas suffisamment sécurisée, et que le gardien a réussi à trouver une faille, alors celui-ci va pouvoir l'exploiter pour espionner les messages échangés. Comme la stéganalyse cherche à différencier les médias contenant un message caché des médias n'en contenant pas, il est possible d'utiliser des filtres. Fridrich et Kodovsky [32] ont proposé une méthode de stéganalyse d'image par "modèles riches". L'idée est que les pixels partagent un nombre important d'informations avec leur voisinage. Les pixels sont décrits par leurs caractéristiques et leurs liens d'adjacences. Les auteurs utilisent un ensemble de classificateurs et leur modèle pour stéganalyser trois méthodes dont HUGO.

3.4 IDC haute capacité

L'insertion de donnée cachées haute capacité définit les méthodes ayant pour but un enrichissement de contenu ou un ajout de méta-données. Dans le cadre de l'IDC pour les maillages 3D, il peut être utile d'insérer une information de texture ou de couleur par exemple. Ceci peut permettre d'éviter de transférer deux fichiers, un pour le maillage, un pour la texture. Cela permet aussi de cacher la texture à un utilisateur qui n'a pas connaissance de l'IDC, tout en lui permettant de visualiser et manipuler le maillage 3D dans son afficheur standard. D'autres cas d'applications existent comme en médecine où les informations d'un patient et de diagnostic peuvent être insérées directement dans l'image médicale, pour éviter la perte ou la séparation de ces informations.

Ce genre d'information étant volumineuse, le principal challenge est d'augmenter le plus possible la capacité des méthodes. Du fait du compromis présenté Fig. 3.4, les méthodes d'IDC haute capacité sont très peu robustes. La plupart des méthodes haute capacité sont basées sur le remplacement de bits ou sur une quantification, comme les méthodes basées LSB. Par exemple, la méthode proposée par Yang *et. al* [132] permet d'insérer plusieurs bits par pixel. Généralement, ces méthodes se placent dans le domaine spatial. Cependant, certains auteurs atteignent également une haute capacité dans les

domaines transformés. Par exemple, Lin et Shiu [76] ont proposés une méthode de compression d'image basée DCT (Transformée en Cosinus Discrète) en utilisant une étape de quantification pour insérer l'information.

3.5 Dissimulation de droits d'auteur

Dans l'insertion de droits d'auteur, nous distinguons le tatouage robuste, le tatouage fragile et le fingerprinting. Contrairement à la stéganographie, un média marqué peut être présenté comme tel. La marque peut servir pour l'identification, en effet elle permet de clamer la propriété d'un média ou d'identifier un utilisateur par sa copie du média. Elle peut aussi servir à des buts de vérification d'intégrité ou d'authenticité d'un média. Ces méthodes sont réparties en 3 sous-catégories présentées dans la Fig. 3.3. Les méthodes de tatouage fragile et les méthodes de tatouage robuste l'identification du propriétaire ou de l'utilisateur (fingerprinting).

3.5.1 Tatouage fragile

Le tatouage fragile est conçu pour vérifier l'authenticité ou l'intégrité d'un média. Dans le cas d'un média dans lequel nous nous attendons à retrouver une marque, si celle-ci n'est pas présente ou dégradée alors l'authenticité n'est pas vérifiée et le média n'est pas de confiance. Lorsque le tatouage se fait pour savoir quelles modifications ont eu lieu, le tatouage est sert à la vérification d'intégrité. La marque est prévue pour être détériorée en fonction des modifications faites au média. Il est alors parfois possible de détecter la zone du maillage 3D modifiée [79] ou les pixels falsifiés [140]. Le tatouage semi-fragile est conçu pour résister quand même à certaines modifications usuelles comme la compression.

3.5.2 Tatouage pour l'ayant droit

Le tatouage robuste permet de marquer un média, avec un identifiant du propriétaire. Ce type de tatouage est prévu pour vérifier la propriété d'un média par extraction de la marque de l'ayant droit. Cette marque, conservée par un tiers, est censée servir pour pouvoir revendiquer la propriété d'un média. Cependant, la législation n'a pas encore tranché sur ce sujet. Néanmoins, ce type de tatouage doit être robuste aux attaques jusqu'à ce que le support soit trop dégradé pour qu'il ne soit encore utilisable ou qu'il n'ait plus de valeur marchande.

3.5.3 Fingerprinting

Dans le cas du fingerprinting, la marque insérée est un identifiant correspondant à un utilisateur. Cette marque peut être insérée dans le but de faire du traçage de traitres dans le cas de piratage ou de divulgation d'information, comme les codes de Tardos [109]. Bien évidemment, l'identifiant caché doit être robuste car les codes permettent la mise en cause des utilisateurs impliqués dans une copie pirate. Par exemple, un média falsifié peut être obtenue par fusion des contenus de plusieurs utilisateurs. Comme l'identifiant des utilisateurs est inséré dans leurs versions, la version pirate contient une partie de tous les identifiants. Le but des décodeurs est alors de retrouver une partie ou tous les utilisateurs impliqués, tout en n'accusant pas d'innocents, comme la méthode proposée par Desoubeaux *et. al* [26].

3.6 Sécurité

La sécurité a été définie par Kalker [58] comme l'incapacité pour des utilisateurs non autorisés d'accéder au canal de tatouage. Cela signifie qu'il est possible de lire le message. Pour Perez-Freire *et. al* [94], la sécurité correspond à la difficulté d'estimer les paramètres secrets de la méthode d'insertion en observant un objet marqué. Il est important de souligner la différence entre la sécurité et la robustesse, comme l'ont montré Cayre *et. al* [13]. En effet, une attaque sur la robustesse peut permettre d'enlever une marque, ce qui est problématique en tatouage mais moins gênant en stéganographie où le message doit rester secret. Dans les deux cas, estimer les paramètres secrets d'une méthode peut permettre une attaque ciblée d'une méthode, par remplacement du message par exemple. Une illustration de l'analyse de la sécurité par Furon et Bas [34]. Ils étudient l'efficacité de la longueur de la clé pour un schéma de tatouage utilisant une méthode QIM avec compensation des distorsions, dans lequel le vecteur de distorsions joue le rôle de la clé secrète.

3.7 Conclusion

Dans cette section nous avons présenté les caractéristiques, les propriétés ainsi que quelques application en insertion de données cachées (IDC). Nous avons présenté les principaux scénarios d'utilisation avec des exemples de la littérature. Ces méthodes reposent généralement sur un compromis entre l'imperceptibilité, la robustesse, la capacité et la sécurité. Il est possible de séparer les méthodes d'IDC en deux grandes catégories, à savoir la stéganographie qui utilise l'IDC comme moyen de communication ou l'IDC

comme moyen d'ajouter de l'information (additionnelle ou de droits d'auteur). Pour résumer, dans la première catégorie le message caché a plus de valeur que le support et inversement pour la seconde catégorie.

À cause du compromis entre la robustesse, l'imperceptibilité et la capacité, en choisir une essentielle pour une méthode d'IDC, implique de réduire les autres fortement ainsi les méthodes les plus robustes ont une très faible capacité et sont facilement détectable. En pratique la majorité les auteurs proposent des méthodes apportant des compromis adaptés à l'application. Par exemple, il existe des méthodes haute capacité qui sont robustes à certaines attaques. Généralement, il est préférable d'utiliser des méthodes aveugles respectant le principe de Kerckhoffs, avec l'objectif de maintenir les paramètres secrets et le message caché inaccessible.

Dans nos travaux de recherche, nous proposons de développer des méthodes d'IDC haute capacité afin d'insérer une grande quantité d'information dans un maillage 3D, tout en produisant le moins de distorsions possible et en offrant un bon niveau de sécurité.

Chapitre 4

Insertion de données cachées 3D

4.1 Introduction

L'insertion de données cachées (IDC) dans des maillages 3D est un domaine d'étude récent. Dans ce chapitre, nous effectuons le lien entre les Chapitres 2 et 3. La représentation de surfaces dans $\mathbb{R}^3$, sous forme de maillage, permet d'utiliser ces maillages 3D comme support d'un message secret. Les maillages pour lesquels les algorithmes sont les plus étudiés, sont des maillages triangulaires et 2-variétés. Cependant, nous considérons également d'autres types de maillages. Nous considérons, ici, les modifications des maillages 3D comme des attaques contre la robustesse d'une méthode d'IDC. D'autre part, les méthodes d'ordonnancement présentées servent comme étapes de synchronisation pour l'IDC. Nous commençons par proposer un aperçu des méthodes de chiffrement pour les maillages 3D. Ces méthodes apportent des solutions pour certains problèmes, comme réduire l'accès à certaines informations à un utilisateur. Nous montrons les avantages et les limites de ces méthodes par rapport à celles d'IDC. La représentation des objets sous forme de maillage permet d'utiliser plusieurs domaines d'insertions, en particulier le domaine spatial, ou les domaines transformés. Les méthodes d'IDC sont classifiées selon leur domaine d'insertion, comme dans les panoramas proposés par Rondao *et. al* [101] et Wang *et. al* [123]. L'insertion dans les domaines transformés garantit une meilleure robustesse que dans le domaine spatial. Néanmoins, le domaine spatial offre une plus grande capacité (voir Table.3.1). Dans cet état de l'art sur l'insertion de données cachées dans des maillages 3D, notre étude porte sur les types d'IDC présentés Fig. 3.3. Cependant, nous analysons plus en détails les méthodes à haute capacité. Contrairement aux définitions strictes de la stéganographie introduites Section 3.3.1, en 3D les auteurs utilisent plus facilement le terme stéganographie

afin de s'opposer au tatouage robuste. Dans ce contexte, la stéganographie est souvent décrite comme une méthode d'IDC à haute capacité produisant peu de distorsions dans un maillage 3D. Ne pas divulguer l'existence du message est le but de la stéganographie. Pour valider cette indétectabilité statistique, il faut procéder à une étape de stéganalyse théorique ou pratique. En effet, l'imperceptibilité d'une marque ne la rend pas indétectable. Généralement, l'indétectabilité implique l'imperceptibilité qui est aussi un point clé de la stéganographie. En effet, l'objet support doit avoir un intérêt et être considéré comme le message final. Dans les parties suivantes nous classifions donc les méthodes qui n'utilisent pas cette définition, comme des méthodes d'IDC à haute capacité. Contrairement à l'IDC dans les images, il n'existe pas de stéganalyses avancées en 3D.

Dans ce chapitre, après avoir présenté la confidentialité visuelle Section 4.2, nous explorons les différentes méthodes d'IDC, pour la haute capacité Section 4.3, et pour le droit d'auteur Section 4.4. Finalement, la Section 4.5 est structurée autour d'une présentation des méthodes d'analyse de la sécurité et des méthodes de stéganalyse.

4.2 Confidentialité visuelle

La confidentialité, l'intégrité, la disponibilité, l'authentification et la non répudiation sont parmi les principaux objectifs des applications de sécurité 3-D. De préférence, un objet protégé doit être dans le même format que l'original. Les méthodes de chiffrement transforment les données d'origines afin qu'elles ne soient pas identifiables et sont utiles pour la confidentialité visuelle. La confidentialité visuelle peut être réalisée par masquage des données 3-D afin de produire un maillage inintelligible pour ceux qui ne partagent pas la clé. Un chiffrement sélectif des données peut être utile pour permettre la visualisation restreinte à des données non sensibles. Après le déchiffrement, le maillage décodée n'est plus protégé. Néanmoins, une approche pratique qui repose sur la permutation des coordonnées des sommets, est présentée dans la Fig. 4.1. Ce chiffrement complet préserve la boîte englobante de l'objet.

Le chiffrement sélectif peut être effectué en chiffrant une partie du maillage ou en chiffrant les plus petites décimales des coordonnées de chaque sommet, comme illustré dans la Fig. 4.2. Les deux méthodes respectent la condition de préservation du format de chiffrement comme expliqué par Bellare *et al.* [5].

Ces méthodes possèdent un intérêt dans certains domaines pour masquer des éléments, comme dans le jeu vidéo ou l'animation. Comme les représentations peuvent produire des maillages non fermés, non manifold, le temps de rendu est affecté et a été

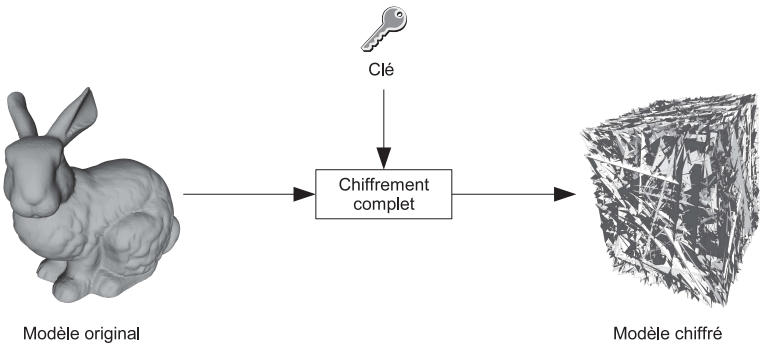


FIGURE 4.1: Schéma classique de chiffrement complet.

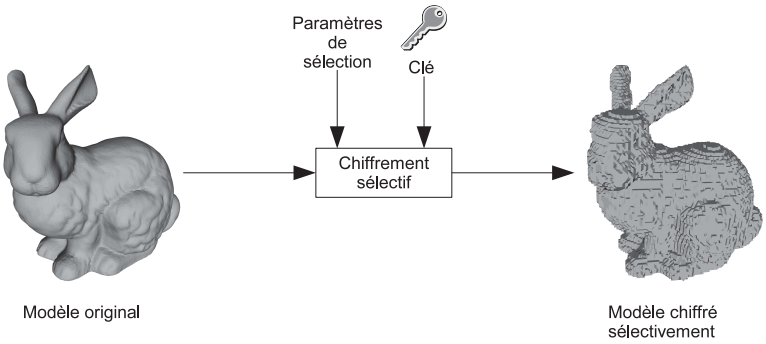


FIGURE 4.2: Schéma classique de chiffrement sélectif.

analysé par Eluard *et al.* [30]. Cependant, ces méthodes ne peuvent être utilisées que pour partager un maillage avec un tiers de confiance. Une fois décrypté le maillage n'est plus protégé.

4.3 IDC 3D haute capacité

4.3.1 Domaine de représentation

L'IDC dans le domaine de représentation consiste à utiliser les redondances dans la représentation du maillage comme support de l'information [133]. Ces méthodes sont sans distorsion puisque ni la géométrie ni la topologie du maillage ne sont affectées. Les maillages sont généralement représentés par une liste de sommets, leurs coordonnées dans $\mathbb{R}^3$ et une liste de polygones. L'ordre d'apparition de ces éléments dans les listes n'affecte pas le rendu 3D. Bogomjakov *et. al* [8] réordonnent les sommets et les facettes ainsi le message est inséré dans les permutations de primitives par rapport à leur ordre de parcours dans le maillage. La méthode peut utiliser n'importe quelle synchronisation déterministe comme "Edgebreaker". Le support de l'information est donc le fichier et non l'information géométrique ou topologique, ce qui est intéressant pour une utilisation en stéganographie. La capacité de leur méthode est calculée comme : $\lfloor \sum_{i=1}^{|V|} log_2 i \rfloor +$

$\lfloor \sum_{i=1}^{|F|} log_2 i \rfloor$, où $|V|$ est le nombre de sommets et $|F|$ le nombre de facettes du maillage. Par la suite, Huang *et. al* [47] et Tu *et. al* [116] améliorent la capacité de la méthode en s'approchant des valeurs théoriques optimales. Ces méthodes sont très sensibles, puisqu'il suffit de changer de représentation ou simplement de réorganiser les primitives pour faire disparaitre la marque. Plus récemment, Lin *et. al* [77], proposent de combiner cette approche avec une approche d'IDC haute capacité [15] pour augmenter la capacité de leur méthode. Ils atteignent une capacité de $(\alpha_p + 2\beta_p + 3n_{layers})|V|$, où α_p est le nombre de permutations possibles des sommets dans le domaine de représentation, $2\beta_p$ le nombre de permutations possibles des triangles dans le domaine de représentation et $3n_{layers}$ le nombres de couches utilisées par la méthode de Chao *et. al* [15]. En pratique, cette méthode offre plus de capacité mais produit des distorsions.

L'insertion dans le domaine de représentation permet une capacité importante et aucune distorsion. Il serait important d'étudier la sécurité de ces méthodes ainsi que l'imperceptibilité au sens stéganographique. Il est également intéressant de pouvoir combiner ces méthodes avec des méthodes qui modifient la topologie. En effet grâce à ce message on peut imaginer détecter un changement de format ou une attaque malicieuse visant l'ordonnancement des primitives dans le fichier.

4.3.2 Domaine spatial

Cette catégorie d'IDC contient les méthodes qui changent la géométrie d'un maillage, *i.e.* qui modifient la position des sommets dans l'espace de plongement de la topologie. Les méthodes dans le domaine spatial permettent en général une grande capacité, mais ont une plus faible robustesse. La plupart des méthodes haute capacité ou de stéganographie appartiennent à cette catégorie. Ces schémas d'insertion se basent souvent sur le concept de QIM, qui permet une grande capacité avec peu de distorsions du vecteur support.

Une des premières méthodes dite haute capacité a été proposée par Cayre et Macq [12], pour de l'insertion dans des maillages triangulaires. L'étape de synchronisation consiste d'abord à ordonner les triangles qui vont servir de support à l'information, grâce à la méthode TSPS, et ensuite à insérer l'information en utilisant une technique basée sur la méthode QIM, illustrée Fig. 4.3.

L'idée principale de la méthode consiste à d'insérer un bit en considérant la projection d'un sommet sur l'arête opposée comme support. La Fig. 3.3 illustre l'insertion du bit "1" dans deux configurations différentes, le sommet est déplacé pour que sa projection soit dans un intervalle qui code la valeur "1". Les auteurs proposent de choisir le premier triangle qui est le départ de la méthode de synchronisation avec deux méthodes.

(a) (b)

FIGURE 4.3: Méthode d'insertion de l'algorithme de Cayre et Macq [12], le bit "1" est inséré en déplaçant le sommet C_0 à la position C_1 pour que sa projection sur l'arête opposée correspondent à l'intervalle qui convient. L'arête opposée est divisée : a) en deux intervalles, b) en quatre intervalles.

La première en choisissant le triangle ayant l'aire minimale et la seconde consiste à effectuer une ACP (Analyse en Composantes Principales) et d'utiliser les triangles qui intersectent les trois axes principaux. Cette méthode permet une certaine robustesse contre des transformations affines. De plus, la capacité de la méthode est d'environ 1 bit par sommet (bps). Néanmoins cette méthode nécessite des maillages triangulaires, pour la synchronisation mais également comme support. L'extension de la méthode aux maillages de polygones n'est pas triviale.

La capacité de la méthode de Cayre et Macq [12] a été améliorée par Wang et Cheng [121] pour atteindre au minimum 3 bps. Les auteurs utilisent une procédure d'insertion multi-niveaux qui insère de la même manière que la méthode de référence, puis dans la hauteur du triangle grâce à un seuil et enfin dans la rotation de l'angle entre la base et la hauteur du triangle. Cette méthode est limitée par la précision numérique en fonction du nombre de divisions choisi. Les auteurs proposent de conserver une capacité entre 3 et 6 bps tout en prenant en compte les distorsions visuelles dans leur nouvelle méthode [17]. Cette méthode étend le type des maillages d'entrée aux maillages polygonaux, et définit une nouvelle synchronisation basée sur la diffusion contagieuse, comme si le polygone d'entrée et une de ses arêtes étaient contaminés. La transmission se faisant par arête commune, le coût en temps pour parcourir tout le maillage est $\mathcal{O}(n)$.

Pour augmenter la capacité, Chao *et. al* [15] utilisent une méthode d'insertion multi-niveaux sur chaque coordonnée d'un sommet. Dans un premier temps les auteurs définissent un ordre pour l'insertion en utilisant la procédure de TSPS. Ensuite, les auteurs identifient des sommets extrémités V_a, V_b et V_c grâce à l'ACP. V_a et V_b sont les sommets aux extrémités de l'axe principal et V_c est le sommet à l'extrémité la plus éloignée du second axe principal. Ils alignent les axes avec le repère cartésien, puis le segment $[V_a, V_b]$ est divisé en région à deux états R_0 et R_1 de façon alternative, comme illustré Fig. 4.4. La région dans laquelle tombe le sommet est notée q_i et r_i indique sa position. Chaque intervalle possède deux états, l'état changé et l'état inchangé. La taille

des intervalles d'état inchangé est calculée pour qu'aucun sommet n'y appartienne. Si la valeur à insérer est k et $q_i \in R_k$ tel que $k \in \{0, 1\}$ alors le sommet n'est pas déplacé. Dans le cas contraire le sommet est déplacé à la région de changement la plus proche.

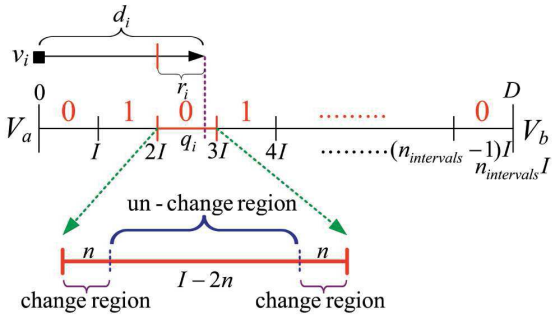


FIGURE 4.4: Illustration proposée par Chao *et. al* [15] d'un niveau d'insertion , ici $q_i \in R_0$.

La Fig. 4.4 présente l'insertion sur un niveau, qui est extensible facilement en multi-niveaux en ajoutant directement d'autres couches. En se servant des trois coordonnées x, y et z d'un sommet, la méthode permet d'insérer 3 bits par sommet et ce dans chaque couche, excepté les 3 sommets utilisés pour la synchronisation, ce qui donne une capacité de $3(|V|-3)n_{layers}$. Cette capacité est bornée par la précision du nombre flottant utilisé pour stocker la position des sommets. Le standard IEEE 754 simple offre 23 bits pour la mantisse, ce qui permet aux auteurs d'affirmer que leur méthode avec ce standard a une capacité maximale théorique de 69 bps.

Dans un scénario de stéganographie, augmenter la taille d'un maillage en ajoutant des sommets est envisageable. La seule contrainte est le coût de stockage, et éventuellement la sécurité si le maillage original peut être obtenu. Outre ces considérations, Li *et. al* [75] ont proposé la première méthode de stéganographie basée sur l'échantillonnage. L'idée se décompose en deux étapes, la génération du message à insérer sur une sphère unitaire dans $\mathbb{R}^3$ et la projection de la sphère sur un maillage de couverture. La première étape consiste à diviser une sphère unitaire en $M \times N$ régions, ces régions sont échantillonnées aléatoirement pour coder le message secret. Par la suite, le centre de la sphère est placé au centre de gravité d'un maillage et les points caractéristiques, notés P codant le message sont projetés sur la surface. Comme la sphère n'est pas uniformément échantillonnée, les auteurs génèrent des points pour compenser les zones moins denses. Cette méthode ne provoque pas de distorsions sur les sommets existants. Cependant l'ajout de sommets a un impact sur la qualité du maillage difficile à évaluer. La méthode transmet seulement le nuage de points comme objet stégo. Tsai [115] fait une liste d'autres inconvénients de cette méthode, par exemple, si la position du centre de gravité du maillage est hors de celui-ci la projection est impossible. Il souligne aussi le coût important en temps du calcul d'intersection de la projection sur les polygones. Sa méthode améliore celle de Li *et. al*

en diffusant l'échantillonnage sur l'ensemble du maillage de couverture, en utilisant une grille unitaire à la place d'une sphère et en projetant les points échantillonnés, notés P, sur le volume englobant de l'objet support. Cette amélioration prend aussi en compte des nuages de points comme objet support en reconstruisant la surface pour avoir une projection adaptée.

Dans le cadre de l'IDC haute capacité et robuste Gao *et. al* [36] ont proposé une méthode semi-aveugle qui a une capacité pratique d'environ 1-2 bps. Ils utilisent les quadrilatères coplanaires convexes comme support de parties du message. L'insertion se fait en modifiant les ratios de longueurs de ces quadrilatères. En outre, ces ratios sont invariants aux transformations affines. L'insertion se fait en modifiant la notation décimale des ratios entre deux bornes notées H et L. Les auteurs utilisent une partie des bits disponibles à chaque étape pour insérer l'ordre de reconstruction du message. La capacité théorique de la méthode, $C = 3(H-L+1)\Omega$, dépend du nombre de quadrilatères Ω, et de la taille de l'intervalle d'insertion. De plus, ils insèrent plusieurs copies du message pour résister aux attaques de types découpage, ce qui réduit la charge utile.

Yang *et. al* [135] ont analysé la corrélation entre un bruit géométrique appliqué à un maillage et un bruit sur la normale des triangles. Leurs résultats montrent un rapport linéaire entre ces deux bruits. Comme il est connu qu'un changement de normale d'une facette provoque des distorsions, les auteurs proposent une méthode d'IDC de haute capacité offrant un compromis entre la capacité et la distorsion des normales. La capacité maximale de leur méthode dépend du degré de dégradation tolérée, en pratique une capacité entre 45 et 60 bps est atteignable sans provoquer trop de distorsions. Une méthode de type LSB adaptative est appliquée pour insérer le message. Chaque sommet a une quantification différente, en fonction du paramètre de tolérance aux distorsions. Cette information est insérée pour être connue à l'étape de décodage. L'ordonnancement des sommets à l'étape d'insertion est donné par la projection des sommets quantifiés sur le premier axe principal. La synchronisation entre l'étape d'insertion et d'extraction est alors permise si l'extraction des taux de quantification est exacte.

Wang and Wang [122] ont proposé deux des seules méthodes à s'intéresser aux points échantillonnés non maillés. Premièrement, ils ordonnent les sommets grâce aux trois axes principaux donnés par une ACP. La première méthode consiste à insérer un bit par coordonnée d'un sommet. L'insertion se fait en modifiant une valeur c_{i+1} par rapport à la valeur médiane $m_{i,i+2}$, entre c_i et c_{i+2}, où $1 \leq i \leq |V|$. En étendant cette méthode, les auteurs proposent d'insérer les valeurs dans des MEPs (Macro Embedding Primitive), les valeurs sont alors déplacées conjointement sur les trois axes principaux. Les auteurs affirment que leur méthode est sécurisée dans le même sens que celle de Cayre et Macq.

TABLEAU 4.1: Récapitulatif des méthodes d'IDC haute capacité dans le domaine spatial.

Méthode	capacité théorique pour un maillage de $\|V\|$ sommets	capacité bps
Cayre et Macq [12]	$\|V\|$	< 1
Wang et Cheng [121]	$3 \times \|V\|$	3
Cheng et Wang [17]	$\|V\| \times \alpha$ un entier donné	3-6
Chao *et. al* [15]	$69 \times \|V\|$	40
Li *et. al* [75]	$\|P\| \times \lfloor \log_2(M \times N) \rfloor$	50
Tsai [115]	$\|P\| \times \lfloor \log_2(M \times M) \rfloor$	> 8
Gao *et. al* [36]	$3 \times (H - L + 1)\Omega$	1-2
Yang *et. al* [135]	$\|V\|\times$ fonction du facteur de tolérance aux distorsions ε	45-60
Wang and Wang [122]	$1.5 \times \|V\|$	< 1.5

Finalement, l'évolution des techniques d'IDC haute capacité dans le domaine spatial, présentée dans le Tableau. 4.1, permet d'offrir une grande capacité supérieure à 40 bps avec de faibles distorsions. Les méthodes basées sur l'ajout de sommets [75, 115] augmentent la taille du fichier, et produisent seulement un nuage de points (qui peut être maillé sans changer la géométrie). Les méthodes [15, 135] possèdent également une grande capacité mais sont néanmoins très sensibles à la désynchronisation.

4.3.3 Domaines transformés

Il existe peu de méthodes d'IDC haute capacité dans les domaines transformés puisque le domaine spatial offre une plus grande capacité. Néanmoins certains auteurs atteignent une capacité assez élevée comme Kaveh et Moin [61] qui proposent une méthode basée sur une transformation en surfacelettes (ST). L'insertion se fait par la modification des coefficients ST. La méthode offre une capacité de $3|V|$, cependant elle n'est que partiellement robuste aux transformations affines et au découpage.

4.4 IDC pour les droits d'auteurs

4.4.1 Tatouage fragile

Les méthodes de tatouage fragile s'appuyant sur la connectivité topologique, sont des techniques qui ne modifient pas la géométrie du maillage. Les changements topologiques sont limités, ce qui offre une faible capacité à ces méthodes. Mao *et. al* [83] ont proposé une méthode pour les maillages triangulaires qui subdivise les triangles pour insérer l'information. La synchronisation est définie à l'aide d'un chemin de triangles.

En principe cette méthode ne crée pas d'erreurs géométriques et a une capacité de 8 bits par arête. Néanmoins, couvrir tout le maillage est compliqué avec cette méthode et le poids du fichier est considérablement augmenté. De plus, une marque insérée dans une zone du maillage peut être détectée comme une zone de plus haute densité. Amat *et. al* [1] utilisent les arbres couvrant de poids minimum (ACPM) pour synchroniser des quadrangles, deux triangles partageant une même arête, coplanaires et convexes. L'insertion se fait en flippant cette arête selon la valeur du bit à insérer. Pour le bit "0" l'arête doit appartenir à l'ACPM, et l'inverse pour la valeur "1". Évidemment ce genre de méthodes ne sont pas robustes aux attaques sur la connectivité.

Yeo et Yeung [137], ont proposé une des premières approches de tatouage fragile. L'insertion modifie la géométrie de l'objet de telle sorte que pour chaque sommet la valeur des fonctions de hachage soit la même. Leur méthode vérifie l'intégrité d'un maillage 3D en comparant deux fonctions de hachage données, sur chaque sommet, si leur valeur est différente alors il y a eu une modification. La synchronisation de la méthode est donnée par une fonction de hachage qui n'est pas robuste et peut poser des problèmes de causalité. Lin *et. al* [79], propose une méthode semi-fragile qui résiste à certaines manipulations comme la quantification et le ré-ordonnancement des sommets. Cette méthode permet d'éviter les problèmes de causalité de la méthode de Yeo et Yeung.

4.4.2 Tatouage robuste

Le tatouage robuste est conçu pour être préservé même après des attaques. À cause du compromis entre la capacité et la robustesse, ces méthodes insèrent peu de charge utile. Cependant pour certaines applications, un identifiant de 64 ou 128 bits peut suffire.

Domaine spatial

La première méthode d'IDC dans des maillages 3D a été proposée par Ohbuchi *et. al* [90], en insérant une marque visible sur le maillage. Une des premières méthodes de tatouage robuste invisible est la méthode de Benedens [6]. Elle est basée sur la construction d'un histogramme des normales des sommets, chaque classe est utilisée pour insérer 1 bit. Cette méthode est robuste aux méthodes de simplification puisqu'elle préserve la forme du maillage. Par contre, l'ajout d'un bruit géométrique perturbe fortement les normales. Les méthodes utilisant les histogrammes ont l'avantage d'être robustes et de ne pas nécessiter d'ordre sur les primitives du maillage. L'utilisation des caractéristiques statistiques géométriques a été proposée par Zafeiriou *et. al* [138]. Les auteurs se placent dans un repère sphérique à partir du centre de gravité et d'une ACP.

Dans une première méthode, les sommets sont synchronisés en fonction de leurs coordonnées θ. Cette méthode n'étant pas robuste à la simplification, les auteurs proposent une seconde méthode qui regroupe les sommets qui appartiennent à un intervalle d'angle θ. L'insertion se fait en utilisant les distances radiales de chaque groupe. Ces distances sont supposées suivre une distribution gaussienne, l'insertion se fait en modifiant la variance à droite ou à gauche. Cette méthode produit peu de distorsions mais réduit la capacité. Les méthodes statistiques sont utilisées ensuite par Cho *et. al* [18], les sommets sont groupés par leur distances radiales, puis les groupes sont normalisés entre $[0, 1]$. Les auteurs proposent deux techniques d'insertion en modifiant la moyenne ou la variance des distributions. Dans le premier cas, un bit est inséré en déplaçant les sommets d'un groupe de façon à ce que la moyenne de la distribution soit supérieure ou inférieure à 0.5. Roandao-Alface *et al.* [102] utilisent la saillance pour définir des zones d'insertion. Cette méthode est conçue pour être robuste au découpage car les zones de fortes courbures ne doivent pas être altérées par une modification du maillage comme la simplification, la compression ou des attaques dans le but d'éviter une perte de qualité. L'insertion se fait suivant la méthode de Cho *et. al* pour chaque zone. En théorie, la méthode résiste au découpage si les zones sont entièrement conservées, ce qui n'est pas automatiquement le cas en pratique. Bors et Luo [9] proposent une méthode statistique qui améliore celle de Cho *et. al* en optimisant les déplacements des sommets de façon à produire le moins de distorsions possible en fonction de l'erreur quadratique moyenne. Rolland-Nevière *et. al* [100] reprennent cette technique pour résoudre le problème de causalité, *i.e.* conserver la même position du centre de gravité. Leur optimisation permet le déplacement des sommets non seulement sur la direction radiale mais autour, ce qui permet un plus grand nombre de possibilités. De plus, leur méthode prend mieux en compte les distorsions visuelles.

Certains auteurs proposent d'autres méthodes comme, Vasic et Vasic [118] qui utilisent un algorithme de sélection de sommets stables, au sens qu'ils sont généralement préservés lors de modifications. L'insertion se fait alors sur ces sommets avec une méthode basée sur QIM et un code correcteur d'erreurs. Motwani *et. al* [87] ont proposé une méthode originale qui utilise un SVM (Support Vector Machine) et une classe de caractéristiques pour déterminer les zones d'insertions du message.

Domaines transformés

Les domaines transformés offrent en général une bonne robustesse aux méthodes de tatouage et sont souvent utilisés en image à l'aide des DFT, DCT ou DWT par exemple. Cependant, en 3D il n'existe pas d'analyse spectrale aussi efficace et robuste et les méthodes proposées sont souvent très coûteuses en temps. Les premières méthodes de

tatouage dans le domaine spectral sont basées sur la modification des coefficients basses fréquences de la matrice laplacienne combinatoire, comme la méthode non-aveugle de Ohbuchi *et. al* [91]. Le message est inséré dans la différence des coefficients entre le maillage tatoué et l'original. Cette méthode est étendue par Lavoué *et. al* [72] qui offre 20% de robustesse en plus. Les nuages de points 3D sont pris en compte dans des travaux [92, 22], cependant les auteurs utilisent une étape de maillage. Une autre transformation MHT (Manifold Harmonics Transform) permet de prendre en compte la forme du maillage en injectant la géométrie dans la laplacienne. Ce qui permet de développer des méthodes de tatouage robustes et aveugles. Par exemple, les méthodes de Liu *et. al* [80] et de Wang *et. al* [126], permettent une grande robustesse aux attaques. Cependant, leurs capacités sont de respectivement 5 et 16 bits par maillage. L'utilisation d'autres domaines est étudié, par exemple, Konstantinides *et. al* [64] ont proposé une méthode robuste basée sur les harmoniques sphériques mais cette méthode est dépendante de la position du centre de gravité ce qui peut poser des problèmes de causalité notamment lors de découpages. Maret *et. al* [84] décrivent un espace invariant aux transformations affines. La capacité de la méthode atteint 0.5 bps, ce qui est une assez haute capacité pour ce genre de méthode. Cependant elle est moins robuste que les méthodes classiques dans les domaines transformés. Wu *et. al* [130] ont présenté une méthode basée sur un ensemble de fonctions de base radiales. Leur tatouage est non-aveugle, rapide et utilisable pour des maillages de grande taille. Certains auteurs se basent sur la décomposition multi-résolution des maillages pour proposer des méthodes de tatouage de type transformée en ondelettes. Praun *et. al* [97] ont proposé une méthode de tatouage robuste non-aveugle basée sur la multi-résolution par contraction d'arête de Hoppe [42]. D'autres méthodes s'appuient sur la transformation en ondelettes de maillage régulier de Lounsbery [81], comme les méthodes de Kanai *et. al* [59], de Ucchedu *et. al* [117] et de Wang *et. al* [124]. Par rapport aux autres méthodes utilisant les domaines transformés, ces méthodes permettent un meilleur contrôle des distorsions, et permettent d'utiliser les différents niveaux de résolutions pour l'insertion. Par exemple, la méthode de Wang *et. al* [124] permet d'insérer plusieurs marques (une robuste, une fragile et une haute capacité). Cependant, ces méthodes sont très sensibles aux attaques sur la connectivité des maillages comme la simplification, le découpage ou le remaillage.

4.5 Stéganalyse et sécurité

La stéganalyse ainsi que la sécurité définie Section 3.6, sont souvent peu analysées en 3D. En sécurité, les auteurs considèrent qu'une complexité théorique élevée permet de maintenir une bonne sécurité. Nous pensons donc qu'une étude cryptographique des méthodes d'IDC est utile comme nous l'avons expliqué dans [51]. Nous montrons qu'il est

possible d'estimer le paramètre de sécurité de la méthode de tatouage de Bors et Luo [9]. La synchronisation des bits insérés est contrôlée par un paramètre qui est considéré comme le seul paramètre de sécurité. Dans cette méthode l'insertion de bits se fait en modifiant l'histogramme des normes de sommets (leur distance au centre). Le paramètre ε contrôle la gamme de rognage des valeurs extrêmes de l'histogramme, pour augmenter la robustesse. Cependant, estimer ce paramètre demande un nombre limité d'estimations, de ce fait la méthode n'est pas assez sécurisée. Nous proposons des pistes d'amélioration, comme un rognage asymétrique de l'histogramme des normes. Et l'ajout de paramètres de sécurité, comme par exemple sur la position du centre.

Dans un contexte de stéganalyse, il est important de pouvoir détecter si un maillage 3D a été marqué. Yang *et. al* [136] proposent une stéganalyse de la méthode de tatouage de Cho *et. al* [18] modifiant la moyenne de groupe de sommets, décrite Section 4.4.2. Dans un premier temps, les auteurs cherchent le nombre de classes de l'histogramme de la distribution des coordonnées radiales. Ensuite, ils montrent que cette distribution pour un modèle marqué est bimodale. Une version stéganographique de la méthode de Cho *et. al* est également proposée. L'origine des coordonnées sphériques est déplacée pour augmenter la variance des coordonnées radiales. L'information est alors insérée dans la différence entre deux classes successives de l'histogramme. Cette méthode permet de baisser le taux d'accusation d'environ 98% pour la méthode de Cho *et. al*, à environ 70% pour la version modifiée de Yang *et. al*. Récemment, Yang et Ivrissimtzis [134] ont proposé une première méthode de stéganalyse pour les maillages 3D basée sur les travaux faits en image. L'idée est d'entraîner un classifieur sur les différences des caractéristiques des maillages stégos et supports. Les auteurs ont analysé leurs résultats sur des méthodes de tatouage 3D, ce qui produit de bons résultats de classification. Néanmoins, ces méthodes ne sont pas conçues pour être indétectables, de plus le nombre de caractéristiques prises en compte est bien plus faible que le nombre de caractéristiques utilisées actuellement en image [32]. En outre, les méthodes de stéganalyse d'image les plus récentes utilisent des réseaux d'apprentissage "deep learning" et proposent des performances comparables [99].

4.6 Conclusion

Dans ce chapitre nous avons présenté un état de l'art actuel des méthodes d'IDC 3D. Nous nous sommes intéressés aux capacités maximales des algorithmes ainsi qu'à leurs méthodes de synchronisation. Après avoir pointé les défauts des méthodes de chiffrement, nous avons montré que les méthodes possédant les plus hautes capacités sont basées sur la géométrie du maillage, ou sur le domaine de représentation. Cependant, l'insertion

dans le domaine de représentation est très liée au format de stockage et peut facilement être détachée du maillage. Par la suite nous nous sommes penchés sur les méthodes de tatouage fragile qui permettent de vérifier l'intégrité des maillages. Nous avons comparé différentes méthodes de tatouage robuste. Ces méthodes sont souvent conçues pour résister à un type de modification ou d'attaque et la résistance au découpage est encore un challenge. De plus elles possèdent une faible capacité. Finalement, nous présentons les analyses récentes concernant la sécurité de ces méthodes.

Dans nos recherches, nous avons souhaité concevoir une méthode d'IDC 3D haute capacité. D'après notre étude de la littérature, les méthodes basées sur la géométrie du maillage permettent d'atteindre nos objectifs. Plus précisément, les méthodes basées QIM, comme la méthode de Chao *et. al* [15], proposent une capacité importante tout en produisant peu de distorsions. Nous nous sommes basés sur ces travaux comme point de référence. Nous nous sommes particulièrement intéressés aux méthodes de synchronisation basées sur la construction de graphe. Ces graphes permettent de définir un ordre sur les éléments des maillages et peuvent être utilisés comme étape de synchronisation. Pour synchroniser les sommets d'un graphe défini sur un nuage de points, les graphes ont un avantage par comparaison aux méthodes synchronisées à l'aide d'ACPs. En effet, il en existe un grand nombre sur un nuage de points et ils peuvent être sécurisés par l'ajout de paramètres secrets. En général, les auteurs considèrent que la sécurité de la méthode réside dans la méthode d'insertion. Nous voulons étudier le comportement de l'ajout de sécurité dès l'étape de synchronisation. Dans nos contributions nous allons donc étudier un schéma d'IDC en deux étapes présentées dans la Fig. 4.5.

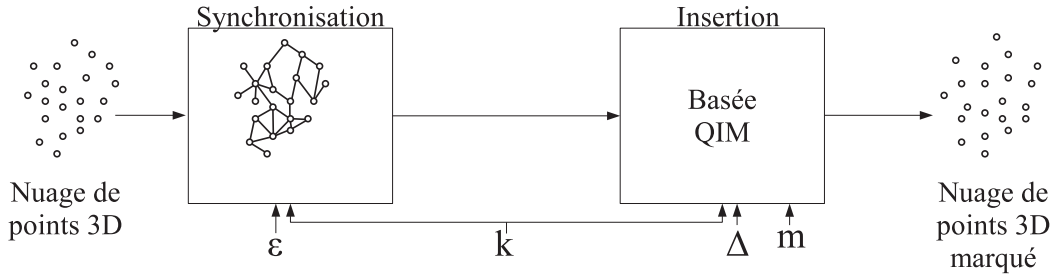


FIGURE 4.5: Schéma d'IDC étudié : ε paramètre de synchronisation, k la clé secrète, Δ le pas de quantification et m le message.

Dans le Chapitre 5, nous étudions la synchronisation des nuages de points 3D à l'aide de chemins hamiltoniens après l'étude des arbres couvrant de poids minimum. Nous y définissons le paramètre ε comme la stabilité de la construction d'une structure sur le nuage de points. Le graphe est stable pour cette valeur, si l'ajout d'un sommet au graphe se fait quelque soit la position du sommet à $\pm\varepsilon$. Dans un second temps, dans le Chapitre 6, nous étudions comment utiliser cette synchronisation afin de proposer une méthode d'IDC haute capacité à l'aide du paramètre de quantification Δ. Enfin, dans

le Chapitre 7, nous proposons de revenir sur l'étape de synchronisation pour proposer une méthode plus sécurisée, d'autre part nous proposons une série d'améliorations afin d'obtenir des meilleures méthodes d'IDC.

Ces travaux on fait l'objet d'une publication dans la conférence internationale IEEE ICIP 2014 [51].

Deuxième partie

Contributions

Chapitre 5

Nouvelle méthode de synchronisation basée sur les chemins hamiltoniens

5.1 Introduction

Dans ce chapitre, nous proposons une nouvelle méthode de synchronisation basée sur les chemins hamiltoniens. Nous nous sommes inspirés des travaux effectués autour des ACPMs (Arbres Couvrant de Poids Minimum) présentés Section 2.4.3.1. Nous y définissons la synchronisation comme le parcours de l'ACPM construit sur les sommets d'un nuage de points 3D. Cette synchronisation possède des propriétés essentielles, que nous recherchons :

— elle est unique,

— elle ne dépend pas de la connectivité du maillage,

— elle donne un ordre grâce à l'algorithme de construction.

Nous présentons et expliquons la nouvelle méthode de synchronisation basée sur la construction d'un chemin hamiltonien dans la Section 5.2. Dans la Section 5.3, nous proposons une solution pour augmenter la stabilité de la méthode suggérée. La Section 5.4 présente les résultats de notre approche et son comportement contre des attaques géométriques. Enfin, la Section 5.5 conclut le chapitre et donne quelques pistes futures des recherches.

5.2 Nouvelle méthode de synchronisation

Dans cette partie, nous proposons une étude et une analyse détaillée de la construction d'ACPM sur des nuages de points 3D. Nous mettons en avant que cette construction est peu stable en raison du nombre important de distances proches entre deux sommets dans un maillage 3D. Pendant la construction d'un chemin, il est possible, qu'à l'étape de comparaison de distances, deux sommets soit presque à la même distance d'un sommet donné, nous appelons cette configuration une "ambiguïté" à ε près. Nous proposons donc d'utiliser un chemin hamiltonien au lieu des ACPMs pour réduire les cas de ambiguïtés.

5.2.1 Analyse des ACPMEs

Nous avons d'ailleurs analysé le comportement d'un ACPME construit grâce à l'algorithme de Prim [98] qui est plus adapté aux graphes denses. La Fig. 5.1 présente le problème de sensibilité des ACPMEs. La Fig. 5.1.a donne une configuration initiale de sommets et la construction d'un ACPME sur ces sommets.

(a) (b)

FIGURE 5.1: Exemple du problème de sensibilité des ACPMs.

Après le déplacement d'un sommet, la Fig. 5.1.b illustre le changement de sélection de l'arête. Nous avons soumis un ACPME, construit sur le nuage de points d'un objet numérique 3D, au déplacement de ses sommets afin de définir une borne de déplacement pour chaque point [54]. Notre approche utilise l'algorithme de Prim pour avoir une construction itérative de l'arbre. À chaque étape, un sous-arbre $T_i = (V_i, E_i)$ de l'ACPM est construit. L'algorithme commence avec un sommet $v_0 \in V$. L'algorithme ajoute le sommet le plus proche $v_i \in V_{i-1}$ à $V \backslash V_{i-1}$ et ajoute dans E_{i-1} la connexion entre v_i et son sommet le plus proche dans V_{i-1} appelé "père" et noté $f(v_i)$. Nous supposons que la perturbation des sommets est simplifiée aux deux hypothèses suivantes :

Hypothèse 5.2.1. À l'étape $i > 0$ de l'algorithme de Prim, nous perturbons seulement la position du sommet v_i, à sa nouvelle position v^*.

Hypothèse 5.2.2. La perturbation géométrique est restreinte à la demi-droite $]f(v_i); v_i)$.

Après la perturbation, $T_i^* = (V_i^*, E_i^*)$ dénote la séquence de construction. Pour maintenir la même connectivité de l'ACPME ($T_k = T_k^*$), il faut vérifier deux conditions :

1. $v^* = v_i^*$, v^* est sélectionné à l'étape i de l'algorithme de Prim.
2. $f(v^*) = f(v_i^*)$, le père de v^* est toujours le père de v_i.

Sous ces hypothèses, nous pouvons définir un radius de déplacement pour chaque sommet. Nous calculons dans un premier temps, les déplacements maximums suivant la demi-droite. Le rapprochement maximal et l'éloignement maximal sont notés respectivement r_i^- et r_i^+. Le rapprochement maximal est la limite à partir de laquelle le sommet courant v_i est sélectionné avant l'étape i au décodage. Cette limite dépend du poids maximal d'une arête dans le sous-arbre, soit $f(v_i) = v_k$ et $V_k = \{v_j : j \leq k\}$, alors nous utilisons les sommets $v_j \in V_i \backslash V_k$ et les arêtes $e_j \in E_{i-1} \backslash E_k$ pour définir :

$$r_i^- = \omega(\{f(v_i), v_i\}) - \max\{\omega(e_j) \ : \ e_j \in E_{i-1} \backslash E_k\}. \tag{5.1}$$

L'éloignement maximal dépend de la distance entre le sous-arbre et le second plus proche sommet $s(v_i) \in V \backslash V_i$:

$$d_i^1 = \omega((f \circ s)(v_i), s(v_i)). \tag{5.2}$$

S'il est déplacé plus loin, le sommet ne sera plus le plus proche du sous-arbre. De plus, pour conserver le même père $f(v_i)$, il faut calculer l'intersection $x(v_k)$ entre la demi-droite et la bissectrice du segment $[f(v_i), v_k]$, $v_k \in V_{i-1}$, $v_k \neq f(v_i)$:

$$d_i^2 = \min\{\omega(f(v_i), x(v_k)) \ : \ v_k \in V_{i-1}, v_k \neq f(v_i)\}. \tag{5.3}$$

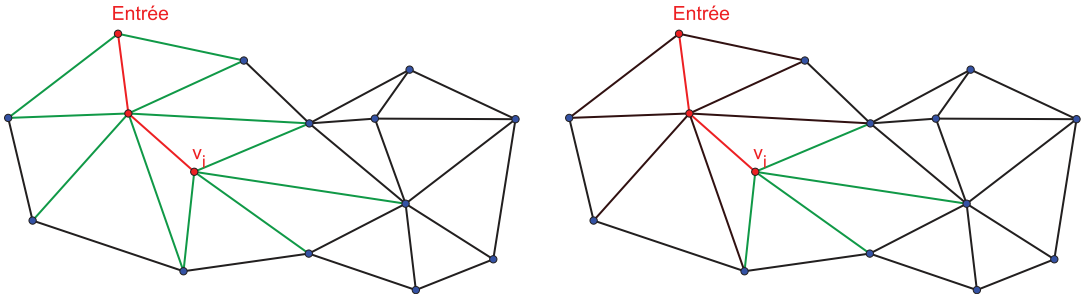
r_i^+ est alors défini comme :

$$r_i^+ = \min\{d_i^1, d_i^2\} - \omega(v_i, f(v_i)), \tag{5.4}$$

Le rayon de déplacement d'un sommet est $r_i = \min\{r_i^-, r_i^+\}$. Nous avons montré dans [54] que si le sommet reste dans ce rayon alors l'arête $e_{\{v_i^*, v_{i+1}^*\}}$ est préservée également. Ces résultats peuvent permettre d'utiliser les sommets les plus robustes comme support d'un message, par exemple dans le cas de l'IDC. Cependant, nous avons montré la complexité de la recherche du rayon de déplacement sous des hypothèses fortes. Nous proposons alors une étude plus simple de la construction d'un chemin hamiltonien.

5.2.2 Chemin hamiltonien

Si l'on considère toutes les stratégies possibles alors pour un graphe de n sommets, il existe n^{n-2} arbres couvrants différents (Formule de Cayley) contre $(n-1)!$ circuits Hamiltoniens. La Fig. 5.2 compare les étapes de construction d'un ACPME Fig. 5.2.a et d'un chemin Hamiltonien Fig. 5.2.b, à partir d'un point d'entrée. La construction du chemin Hamiltonien se fait par sélection de l'arête de poids minimum en distance euclidienne. À l'instant $t = i$, les arêtes en rouge appartiennent à la structure, les arêtes en vert indiquent les arêtes qui sont comparées pour trouver v_{i+1}, une des stratégies peut être de choisir le poids minimum. On constate que le nombre de poids d'arêtes à comparer à chaque étape est plus beaucoup plus important dans le cas des ACPMEs. La construction d'un chemin Hamiltonien, avec une recherche du plus proche voisin, propose moins de candidats possibles que la construction d'un ACPME avec l'algorithme de Prim.



(a) (b)

FIGURE 5.2: État de la construction d'un chemin à l'instant i, les sommets et les arêtes en rouge sont déjà parcourus, les arêtes en vert sont comparées pour trouver le sommet du chemin à l'instant $i+1$, dans le cas : a) d'un ACPME avec l'algorithme de Prim, b) d'un chemin Hamiltonien.

Ce critère est important lorsque l'on cherche à évaluer la robustesse d'un point dans le chemin, *i.e.* si le sommet v_i est déplacé dans l'espace à la position v'_i, l'ordre de parcours des sommets ne doit pas être changé. De plus, pour comparer la complexité pour un graphe complet de n sommets, on sait qu'une implémentation naïve de l'algorithme de Prim donne un ACPME en $\mathcal{O}(n^2)$, tandis que la construction d'un chemin Hamiltonien se fait en $\mathcal{O}(n)$.

Pour un nuage de sommets possédant n sommets, le chemin hamiltonien est construit sur le graphe complet pondéré par les distances euclidiennes des sommets. Ce graphe est noté $G_n = (V_n, E_m, \omega)$ tel que $\omega : E \to \mathbb{R}^+$. L'ensemble $E_m = \{\{v_i, v_j\} | v_i, v_j \in V_n, v_i \neq v_j\}$ représente les arêtes pondérées du graphe et $m = n(n-1)/2$. Le chemin construit à l'étape i est un sous-chemin hamiltonien noté $\mathbf{P_i}$ construit sur l'ensemble des sommets $V_i = \{v_0, \ldots, v_i\}$, et qui passe par les arêtes orientées $E' = \{e\{v_0, v_1\}, \ldots, e\{v_{i-1}, v_i\}\}$.

Tous les sommets sont soit dans V_i, l'ensemble des sommets de $\mathbf{P_i}$, soit dans $V_n \backslash V_i = \{v_{i+1}, \ldots, v_n\}$, l'ensemble des sommets non visités. Ces ensembles sont illustrés dans la Fig. 5.3.

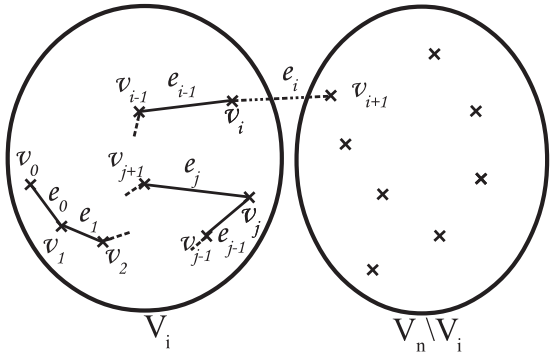


FIGURE 5.3: État des ensembles à l'étape i.

Nous utilisons une clé secrète pour obtenir le sommet de départ v_0 sur les n sommets possibles. Le sommet est donné soit par ses coordonnées, soit à l'aide d'une fonction de hachage. Un chemin hamiltonien passe une et une seule fois par chaque sommet du graphe. Pour le construire, nous choisissons récursivement le plus proche sommet v_{i+1} du sommet courant noté v_i, $i \in [0, n-1]$. La recherche du plus proche sommet se fait en choisissant l'arête de poids minimal en distance euclidienne e_i entre un sommet $v_i \in V_i$ et un des sommets non-visités $v_{i+1} \in V_n \backslash V_i$. L'ensemble de recherche est alors réduit à E'_i, défini comme l'ensemble des arêtes de v_i, tel que $e_k = e\{v_i, v_k\}$, $k \in V_n \backslash V_i$ et $|E'_i| = n - i - 1$. L'arête e_i est alors choisie comme :

$$\omega(e_i) < \omega(e_k), i \neq k, \tag{5.5}$$

où $e_i, e_k \in E'_i$. Clairement, à la fin de la construction, $\mathbf{P_n}$ est un chemin hamiltonien sur $G = (V_n, E_m)$.

5.2.3 Analyse des problèmes d'ambiguïté

Nous avons vu que des ambiguïtés peuvent apparaitre dans la construction d'un chemin hamiltonien. Ces ambiguïtés sont la source d'erreur de reconstruction d'un chemin si la position des sommets varie légèrement. Soit v_i le sommet courant, soit v_j son plus proche voisin et v_k un sommet non-relié au chemin $v_j, v_k \in V_n \backslash V_i$ Nous notons eij l'arête entre v_i et v_j et eik l'arête entre v_i et v_j. Alors pour $k \neq j$, nous exigeons que :

$$\begin{aligned} &\omega(e_{ij}) < \omega(e_{ik}), \\ &|\omega(e_{ij}) - min(\omega(e_{ik}))| > \varepsilon. \end{aligned} \tag{5.6}$$

Si cette condition est respectée, il n'y a pas d'ambiguïté entre les deux sommets, et le sommet choisi devient le nouveau sommet courant. Ce sommet est ajouté au sous-chemin en construction et il est connecté au chemin par l'arête e_{ij}. Lorsque le dernier sommet est ajouté au chemin, nous obtenons un chemin hamiltonien sur l'ensemble des sommets du graphes. Si à une étape de la construction du chemin, la condition de l'équation 5.6 n'est pas respectée, nous proposons une solution pour la contraindre en partie dans la section suivante.

5.3 Augmentation de la stabilité du chemin hamiltonien

Nous choisissons de simplifier le problème en effectuant une synchronisation sur moins de sommets. Pour choisir les sommets à synchroniser, nous proposons une méthode de décimation s'appuyant sur le regroupement de sommets. Les méthodes de décimation sont souvent utilisées pour compresser ou pour simplifier des maillages de modèles 3D. Dans le cas du regroupement des sommets, des travaux antérieurs ont mis l'accent sur la segmentation des maillages 3D guidée par la forme, comme celle de Tierny *et. al* [112] ou guidée par la connectivité comme la méthode de Schroeder *et. al* [105]. Nous ne nous sommes pas intéressés à ces méthodes car elles supposent des maillages avec des complexes (humanoïdes, animaux, *etc.*) ou dépendent de la connectivité. La Fig. 5.4 donne un aperçu de notre méthode qui comporte deux étapes principales.

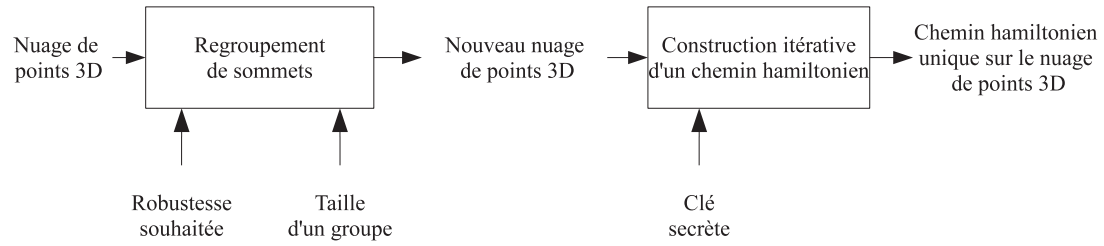


FIGURE 5.4: Schéma de la méthode proposée.

La première, appelée regroupement de sommets, consiste à diviser le nuage de sommets 3D en groupe et à calculer leurs isobarycentres. La deuxième étape consiste à construire un chemin hamiltonien sur l'ensemble des centres, d'une manière itérative en utilisant une clé comme point de départ, tout en vérifiant et maintenant la stabilité du chemin. Cette synchronisation peut être utilisée pour ordonner des zones définies par d'autres méthodes, ou des zones saillantes comme proposées par Lee *et. al* [74].

5.3.1 Regroupement de sommets

Nous proposons d'utiliser les chemins hamiltoniens pour la synchronisation de groupes de sommets. Cette synchronisation est basée sur l'ordre donné par la construction d'un chemin hamiltonien reliant les centres des groupes. Le but est de proposer une méthode de synchronisation plus stable, dans un premier temps nous présentons comment sont groupés les sommets, puis comment est construit le chemin en prenant en compte les cas d'ambiguïtés.

Nous effectuons une étape d'alignement du maillage en utilisant l'ACP (Analyse en Composantes Principales), dans le but de le positionner dans l'espace suivant la même direction. Puis nous calculons le volume englobant du nuage de sommets 3D. Ce volume est alors divisé en un certain nombre de sous-volumes. Cela permet de calculer des groupes identiques et régulièrement répartis, qui sont facilement calculés même sur un maillage modifié. La difficulté de cette approche est alors de définir la taille d'un sous-volume et le nombre de groupes. Un groupe contient tous les sommets inclus dans le sous-volume dont il est issu. De manière générale (*i.e.* dans le cas d'une distribution non-uniforme), l'algorithme produit des sous-volumes vides, chaque sous-volume non vide définit un groupe K_i. La Fig. 5.5, présente les étapes de segmentation et de simplification de notre méthode pour un maillage de 188609 sommets. La première étape consiste à diviser le volume englobant, comme illustré dans la Fig. 5.5.a. Ensuite, les sommets contenus dans un sous-volume appartiennent à un même groupe, ils sont colorés de la même couleur dans la Fig. 5.5.b. Finalement, les isobarycentres des groupes sont calculés et présenté dans la Fig. 5.5.c. Dans la suite nous appelons un centre C_i, l'isobarycentre d'un groupe.

(a) (b) (c)

FIGURE 5.5: Sur un maillage de 188609 sommets : a) Division du volume englobant, b) construction des groupes, c) les barycentres de ces groupes.

Pour obtenir le nombre de groupes, nous choisissons une valeur approximative de sommets que nous souhaitons avoir dans chaque groupe. Comme nous ne disposons pas d'informations *a priori* sur la forme des nuages de points 3D, nous supposons que les $|V|$ sommets sont répartis uniformément dans le volume englobant. Alors le nombre de groupes noté nb_C correspond au nombre de sous-volumes. Nous notons le volume de la boite englobante vol_B. Nous définissons la relation entre deux centres C_i et C_j comme

une arête pondérée notée $e_{ij},\ i \neq j,\ i,j \in [0, nb_C]$. La valeur d'une arrête notée $\omega(e_{ij})$ est donnée par la distance euclidienne entre les deux centres. Le nombre de groupes est alors calculé :

$$nb_C = \frac{vol_B}{\omega(e_{ij}))^3}. \tag{5.7}$$

Réciproquement, on peut chercher à connaître la distance moyenne entre deux centres pour un nombre de groupe donné :

$$\omega(e_{ij}) = \left(\frac{vol_B}{nb_C}\right)^{1/3}. \tag{5.8}$$

Comme nous considérons une distribution uniforme, le nombre de sommets $|K|$ dans chaque groupe est constant pour chaque groupe dans le volume d'un sous-cube vol_C, et de façon triviale nous obtenons :

$$|K| = \frac{|V|}{nb_C}. \tag{5.9}$$

Dans le cas d'une distribution uniforme, la distance $\omega(e_{ij})$ devrait être supérieure aux variations de la géométrie produites par un bruit dans la position des sommets contre lequel le système doit pouvoir résister. Dans la Section 5.4, nous montrons l'influence de ces paramètres dans la robustesse de la synchronisation. Certains groupes possèdent un trop petit nombre de sommets et ne sont pas conservés, car ils sont trop instables. Les sommets des groupes supprimés sont remis en jeu à l'aide d'un algorithme de type k-moyennes, dans un but de stabiliser les bords des groupes. Lorsque l'étape de décimation est finie, on peut limiter la construction d'un chemin Hamiltonien à ces nouveaux sommets.

La Fig. 5.6 illustre la construction du chemin hamiltonien sur les centres. Une clé secrète donne le premier groupe K_1 et son centre est le sommet initial de la construction du chemin. La clé indique la position d'un sous-cube du volume numéroté comme une grille 3D, si il est vide nous déterminons le plus proche non vide. À chaque itération, l'algorithme vérifie s'il y a un autre candidat à une distance proche, inférieure à ε. Ce seuil dépend de la distance moyenne entre deux centres.

Cependant, il ne s'agit pas de simple décimation, nous utilisons les groupes comme éléments principaux pour permettre des changements. En effet, la méthode permet une adaptabilité lors de la construction du chemin. Afin de définir des relations entre les

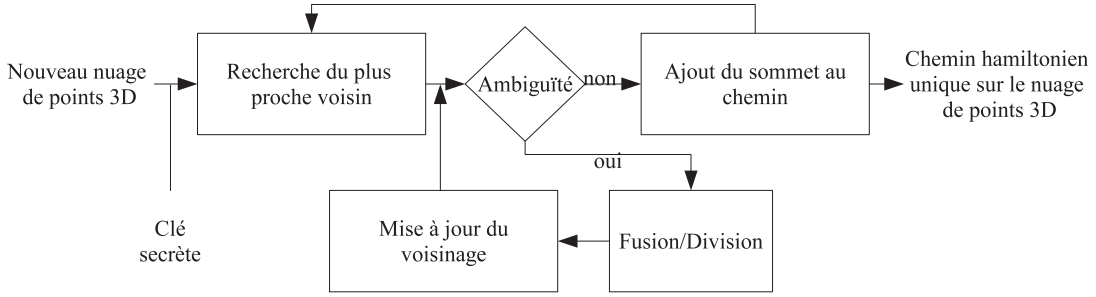


FIGURE 5.6: Schéma de la construction itérative du chemin hamiltonien.

groupes, nous considérons qu'un groupe est voisin à un autre si les sous-cubes auxquels ils appartiennent sont des voisins (*i.e.* les sous-cubes partagent une face et nous considérons un 6-voisinage). L'algorithme maintient les relations entre groupes voisins à chaque changement d'échelle, nous expliquons l'importance pour un groupe de connaître ses voisins, pour résoudre les ambiguïtés qui pourraient survenir.

Lorsqu'il existe une ambiguïté entre deux sommets, nous proposons d'inclure une étape de vérification lors de la construction du chemin hamiltonien. L'idée est de modifier les groupes, appartenant à l'ensemble des groupes E_k construit à l'étape de regroupement. Le but est de supprimer les sommets centre en cause et de les remplacer par de nouveaux sommets. La mise à jour va changer l'organisation des groupes et leur nombre tout en préservant le sous-chemin construit aux étapes précédentes. Nous étudions une ambiguïté entre deux sommets et nous considérons deux cas :

1. Les deux sommets appartiennent à deux groupes voisins.
2. Les deux sommets ne font pas partie de deux groupes voisins.

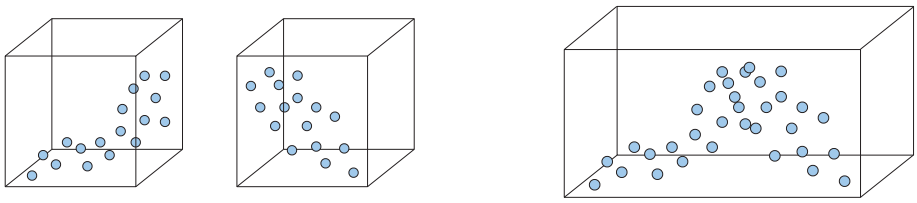
Cette étude peut être étendue à plus de deux sommets, en effet nous proposons un processus itératif prenant en compte le sommet dans une configuration ambiguë le plus proche. Par la suite, les changement apportés changent les relations entre sommets et les cas d'ambiguïté également. Lorsque deux sommets se retrouve être les plus proche à ε près, le changement d'échelle provoque une modification de E_K, noté E'_K. Pour changer d'échelle dans le premier cas, nous fusionnons les deux groupes qui contiennent les candidats et $|E'_K| = |E_K| - 1$. Dans le second cas, nous avons divisé ces groupes et $|E'_K| = |E_K| + 2$.

5.3.2 Fonction de fusion

Fusionner deux groupes K_i, K_j implique de regrouper les sommets en un seul nouveau groupe $K_k = K_i \cup K_j$, puis nous calculons le nouveau centre. Pour conserver le voisinage, notons $\mathcal{N}(K_i)$ l'ensemble des voisins de K_i, tel que :

$$\mathcal{N}(K_k) = (\mathcal{N}(K_i) - \{K_j\}) \cup (\mathcal{N}(K_j) - \{K_i\}). \tag{5.10}$$

La fonction de fusion évite les ambiguïtés en supprimant deux candidats et en calculant un nouveau prétendant unique qui est la moyenne des deux originaux. La Fig. 5.7 illustre un cas de fusion entre deux groupes. Les groupes sont voisins dans le maillage (leurs volumes englobants partagent une face), il n'existe pas de séparation dans l'espace entre les deux volumes englobant autre que celle défini par notre méthode de regroupement.



(a) (b) (c)

FIGURE 5.7: a,b) Groupes originaux, c) Nouveau groupe issu de la fusion.

La fusion est un mécanisme simple, cependant si le seuil ε est grand par rapport à la distance moyenne entre deux centres de gravité, il pourrait il y avoir trop de fusion. En conséquence, dans le pire des cas, la méthode peut fusionner les groupes jusqu'à ce qu'il n'y en ait plus qu'un. Inversement, si le seuil ε est trop petit, il n'y aura pas de fonctions de fusion.

5.3.3 Fonction de division

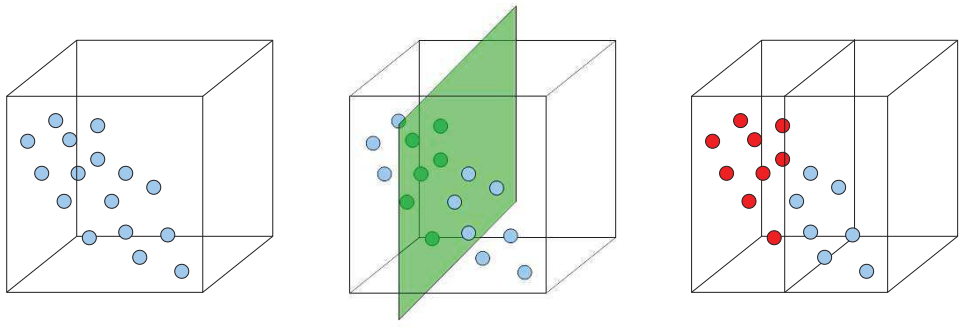
Dans l'étape dite de division, la fonction divise les deux groupes qui provoquent une ambiguïté. Cela évite d'avoir à faire un choix entre deux sommets trop près, et ainsi les erreurs lors de la reconstruction. Contrairement à la fusion, qui est assez simple, nous devons définir la procédure de séparation. Afin d'éviter les ambiguïtés après l'étape de séparation, les distances entre le sommet courant C_i et tous les groupes créés doivent ne pas être trop proches. Nous notons respectivement $K_{k,1}$, $K_{k,2}$ et $K_{j,1}$, $K_{j,2}$ les nouveaux groupes créés par la division de K_k et de K_j. Si le centre de gravité $C_{j,1}$ de $K_{j,1}$ est le sommet choisi à l'étape suivante, alors il doit vérifier la condition :

$$\omega(C_i, C_{j,1}) + \varepsilon < min(\omega(C_i, C_{j,2}), \omega(C_i, C_{k,1}), \omega(C_i, C_{k,2})). \tag{5.11}$$

Si la condition est vérifiée, alors il n'y a plus d'ambiguïté et l'algorithme passe à l'étape de recherche de voisin suivante, comme illustré dans la Fig. 5.6. Néanmoins, si l'équation 5.11 n'est pas respectée parce que le seuil ε est trop élevé après division d'un groupe, notre système pourrait entrer dans une boucle infinie. Pour éviter cela, nous relaxons la contrainte sur le seuil. Quand un groupe est le résultat d'une division, s'il est à nouveau en configuration d'ambiguïté et qu'un autre appel de la fonction de division se produit, la valeur du seuil appliquée sur elle est divisé par deux (*i.e.* $\varepsilon/2$). Afin de satisfaire l'équation 5.11, nous devons trouver le plan qui maximise la différence :

$$\text{diff} = |\omega(C_i, C_{j,1}) - \omega(C_i, C_{j,2})|. \tag{5.12}$$

Pour calculer ce plan, nous introduisons une vérification de la colinéarité entre le vecteur directeur et la norme de la surface du groupe pour éviter un échec de la division. En effet, dans ce cas, un groupe créé est vide et le second est le même que l'original. Elle est définie comme suit : elle tente d'abord de scinder le long de l'axe X, ensuite de l'axe Y et enfin de l'axe Z. L'algorithme s'arrête lorsque la distribution de sommets entre les groupes se situe entre 40% et 60%, et la division se fait le long de cet axe. Nous devons maintenir cet ordre au lieu de chercher le meilleur candidat pour avoir l'algorithme le plus déterministe possible.



(a) (b) (c)

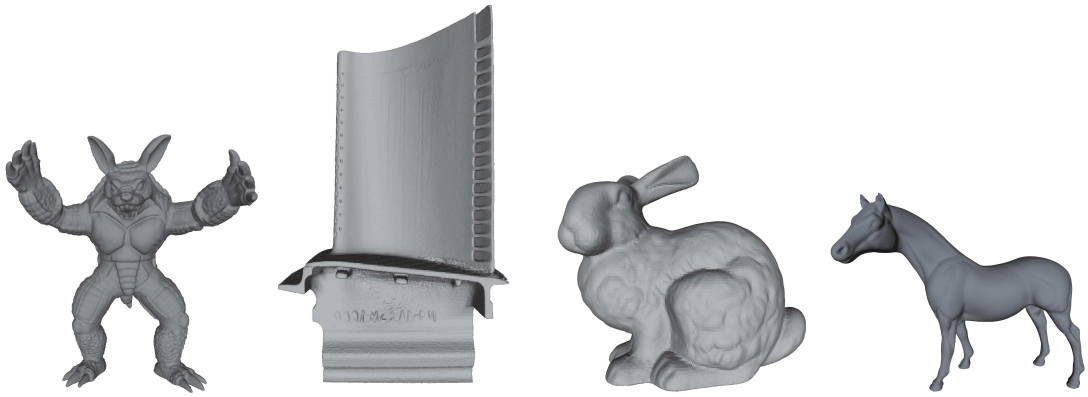
FIGURE 5.8: a) Groupe original, b) division selon le plan donné par l'axe choisi, c) les deux groupes créés.

Le mécanisme illustré Fig. 5.8, calcule le plan de divisions d'un groupe K_i. Cette méthode permet de définir le plan de division 3D grâce au vecteur normal calculé en fonction du choix de l'axe de division et en considérant que le centre d'un groupe fait parti du plan séparateur pour répartir les sommets dans les nouveaux groupes. L'étape suivante consiste à calculer le centre des nouveau groupes. Ensuite, ils sont connectés aux autres en recalculant leur voisinage. Après une fusion ou une division, nous vérifions que les conditions décrites dans l'équation 5.11 sont bien respectées. Par conséquent, l'algorithme ne doit pas ajouter au chemin le centre d'un nouveau groupe créé sans

vérifier qu'il convient. Comme présentée dans la Fig. 5.6, la construction itérative repasse une étape de vérification d'existence d'ambiguïté, avec la nouvelle configuration. Grâce à l'adaptation du seuil, l'algorithme se termine en produisant un chemin hamiltonien déterministe.

5.4 Étude expérimentale

Dans cette section, nous utilisons une base de quatre maillages triangulaires d'objets 3D, présentés Fig. 5.9. Ces maillages sont variés en terme de forme et de nombre de sommets. Nous avons analysé notre méthode de synchronisation sur ces quatre maillages normalisés de façon à ce que la plus longue distance soit égale à 1.



(a) (b) (c) (d)

FIGURE 5.9: Maillages utilisés : a) Armadillo 172974 sommets, b) Blade 220559 sommets, c) Bunny 34834 sommets, d) Horse 48485 sommets.

Nous commençons par présenter la complexité de l'implémentation de l'algorithme. La classification de n sommets en m groupes se fait en temps linéaire $\mathcal{O}(n)$. La construction itérative du chemin hamiltonien est rapide puisque le nombre de sommets est faible. Néanmoins, dans le pire des cas, le temps de calcul est polynomial en $\mathcal{O}(m^3)$.

Une illustration de notre méthode de classification est présentée Fig. 5.10. La Fig. 5.10.a présente les groupes avant la construction du chemin. La construction du chemin se fait alors itérativement et le résultat final est présenté Fig. 5.10.b, où les nouveaux groupes issus de fusions sont en rouge et les deux nouveaux groupes créés après division sont en noir ou en blanc.

Nous voulons que le chemin résiste à un bruit gaussien $\sigma = 10^{-2}$ appliqué sur les sommets d'un maillage. Au delà de cette limite, les distorsions sont trop importantes pour que le maillage ait encore de la valeur. Il est connu que pratiquement toutes les valeurs soient contenues dans trois écarts-types $3.\sigma$ autour de la moyenne $\mu = 0$, exceptées

(a) (b)

FIGURE 5.10: a) Groupement avant la construction itérative du chemin, b) groupement après les fusions en rouge et les divisions en noir et blanc.

quelques valeurs aberrantes. Nous choisissons une distance $\varepsilon = 10^{-1}$, beaucoup plus grande que $3.\sigma$, comme la répartition des sommets n'est pas uniforme dans le volume. Ensuite, nous calculons le nombre de sommets de chaque groupe, grâce à l'équation 5.7. Par exemple, pour le maillage du modèle 3D Armadillo qui a un volume englobant d'environ 0.640, il apparaît qu'un bon choix pour le nombre de groupes est $\frac{0.640}{\varepsilon^3} = 640$, ce qui donne environ 270 sommets par groupe. En théorie, le paramètre de nombre de sommets est exact si les sommets sont répartis uniformément dans la boîte englobante. Comme les formes des objets varies et que certains groupes sont vides, c'est donc plutôt une méthode pour calibrer la méthode. Dans des cas réels, les groupes possèdent en moyenne entre 800 et 1000 sommets, en fonction du nombre total de sommets des modèles 3D de notre base. Pour évaluer les résultats et montrer l'influence du nombre de sommets et de la forme d'un objet, nous comparons tous les modèles avec le paramètre de nombre de sommets par groupe fixé à 200.

TABLEAU 5.1: Résultats obtenus pour différents modèles.

Modèle	# de sommets	# G	# S du chemin	# Fusion	# Division
Bunny	34834	88	98	0	5
Horse	48485	94	104	0	5
Armadillo	172974	229	164	3	19
Blade	220559	291	673	2	192

Le Tableau 5.1 illustre le comportement de notre implémentation sur notre base de test. La seconde colonne donne le nombre de sommets du maillage. La troisième montre le nombre initial de groupes. La quatrième indique le nombre de sommets dans le chemin hamiltonien construit. Les deux colonnes suivantes indiquent le nombre de fusions et de divisions. Logiquement, les modèles plus grands produisent plus de groupes. Le nombre

de fusion/division devient plus grand avec la taille du modèle en raison du choix fait sur la taille fixée des groupes. Notons qu'il y a moins d'étapes de fusion que d'étapes de divisions. Un des principaux défis est de trouver la taille qui maximise le nombre de groupes dans le chemin hamiltonien tout en maintenant une bonne robustesse. Le nombre de groupes obtenus n'est pas prévisible. Nous pouvons l'expliquer par la forme et la densité d'un maillage. Plus il est dense, plus le facteur de seuil va provoquer des ambiguïtés. En effet, la forme de l'objet peut induire un grand volume englobant où les sommets du modèle sont condensés dans la même zone.

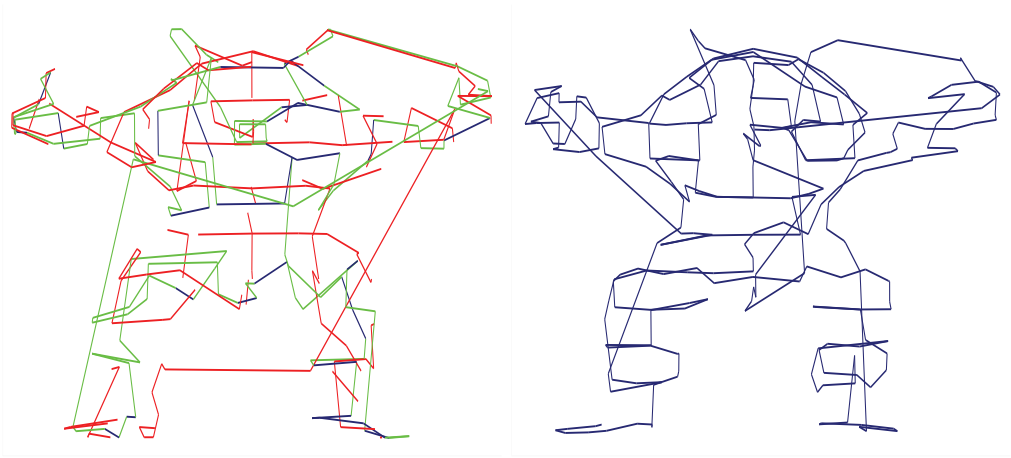
TABLEAU 5.2: Pourcentage d'arêtes communes entre le chemin construit sur un modèle 3D d'origine et celui construit sur le maillage bruité avec un bruit gaussien $\sigma = 10^{-6}$.

Modèle	Bunny	Horse	Armadillo	Blade
Méthode naïve	77%	80%	20%	27%
Méthode proposée	100%	100%	100%	100%

Dans le but de confronter la robustesse du chemin au bruit, nous proposons de comparer les résultats de notre méthode avec ceux d'une méthode naïve. La méthode naïve consiste à construire un chemin hamiltonien sur un ensemble de sommets donnés par un regroupement simple sans considération des ambiguïtés. Le Tableau 5.2 présente les résultats de notre méthode et ceux de la méthode naïve pour une taille de groupe d'environ 200 sommets et un bruit gaussien $\sigma = 10^{-6}$. Ils montrent que notre méthode résiste très bien à ce type de bruit, alors que la méthode naïve ne produit jamais un chemin robuste même avec un faible bruit. Puisque les maillages des objets 3D Bunny et Horse sont plus petits, le chemin hamiltonien construit par la méthode naïve est moins dégradé que ceux construits sur de plus grands modèles. Pour illustrer le contenu du Tableau 5.2, nous montrons un exemple dans la Fig. 5.11 qui est une comparaison entre le procédé naïf et notre méthode.

Les arêtes en bleu sont les arêtes communes, celles en rouge sont celles du chemin obtenu sur le maillage original et en vert celles du chemin obtenu sur le maillage bruité. Nous appliquons un bruit gaussien avec $\sigma = 10^{-6}$ sur l'objet Armadillo, puis nous calculons les chemins hamiltoniens sur le maillage original et le bruité en utilisant la décimation naïve et notre méthode. Le pourcentage d'arêtes communes (en bleu) est de seulement 20% alors qu'avec notre méthode il est de 100%.

Un autre résultat intéressant que nous mettons en avant est que notre méthode est clairement robuste contre les attaques de connectivité, car elle ne repose pas sur les relations entre sommets. Par conséquent, nous avons analysé le comportement de l'algorithme contre une attaque géométrique sur les positions des sommets. Les résultats sont présentés Fig. 5.12. Nous comparons notre méthode avec la méthode naïve, dont les résultats sont notés "DN" (Décimation Naïve). La robustesse est le pourcentage



(a) (b)

FIGURE 5.11: a) Construction d'un chemin hamiltonien à partir d'une décimation naïve, b) construction d'un chemin hamiltonien avec notre approche.

d'arêtes communes entre le chemin hamiltonien calculé sur un maillage 3D et le chemin hamiltonien calculé sur le même maillage 3D bruité.

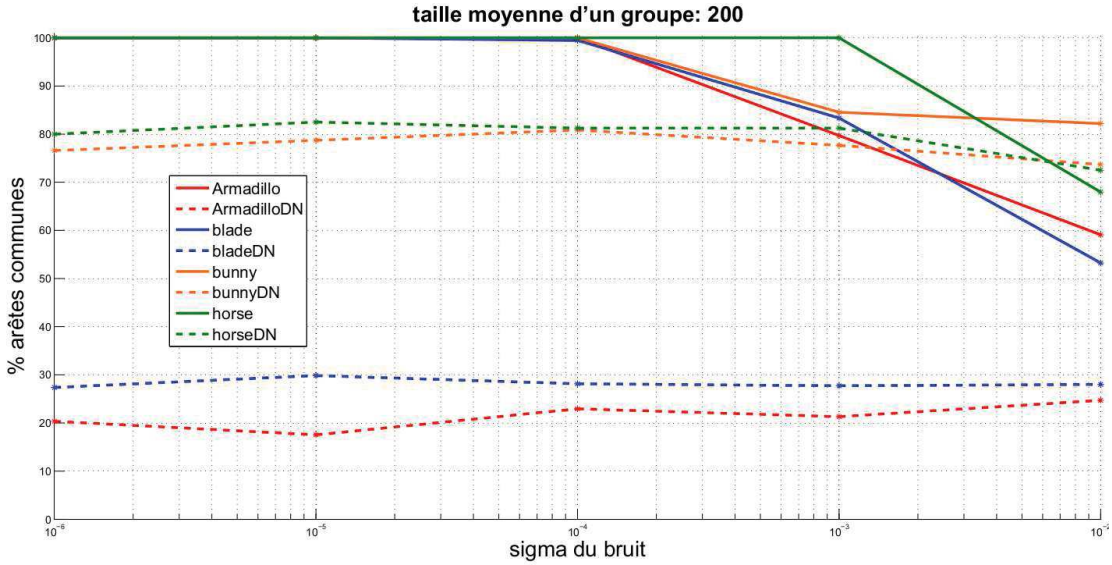


FIGURE 5.12: Courbes de robustesse contre un bruit gaussien.

Les courbes montrent que pour un bruit gaussien $\sigma < 10^{-4}$, le chemin hamiltonien est stable et robuste contre les attaques de déplacement des sommets. Nous voulons atteindre la robustesse contre un bruit d'environ $\sigma = 10^{-2}$, mais nos résultats théoriques sont basés sur une hypothèse forte. Pour un bruit plus important, nous notons qu'il y a encore plus de 50% d'arêtes communes. Comme mentionné précédemment, les paramètres doivent être définis : la taille moyenne d'un groupe est fixé à 200 sommets, ce qui est assez petit pour avoir un bon nombre de groupes pour les petits modèles. Cependant, il est trop petit pour assurer la robustesse des modèles plus grands contre plus de bruit. Ces

courbes montrent que nous devons faire des compromis et adapter les paramètres aux maillages. Cependant, notre méthode est toujours meilleure que la méthode naïve.

5.5 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle méthode robuste et adaptative pour décrire un modèle 3D en utilisant un chemin hamiltonien qui est unique. En effet, notre système multi-échelle évite certaines ambiguïtés. L'approche proposée est efficace contre les attaques géométriques de type ajout de bruit gaussien, nos résultats expérimentaux le montrent. Cette approche peut être utilisée comme méthode de synchronisation afin d'insérer des données cachées dans des modèles 3D. En effet, cette approche présente de bonnes propriétés, est aveugle et ne modifie pas le modèle de maillage 3D. Le procédé proposé peut être amélioré à plusieurs égards. Nous considérons qu'un paramétrage automatique des variables sensibles comme la taille d'un groupe ou le seuil des fonctions fusion/division serait intéressant à définir en fonction, par exemple, du nombre de groupes souhaité. Afin d'obtenir le même chemin entre un maillage original et sa version attaquée, nous souhaiterons également améliorer les règles de construction pour augmenter la robustesse. Cependant, un des principaux challenges reste la robustesse aux attaques de découpage qui perturbent l'ACP et le regroupement des sommets. En perspective, il serait intéressant d'utiliser cette synchronisation afin d'élaborer un tatouage robuste pour insérer des données dans les maillages 3D.

Ces travaux ont fait l'objet de trois publications, une dans une revue internationale Computer-Aided Design [54], une conférence internationale IEEE MMSP [50], et une conférence nationale CORESA [49].

Chapitre 6

Insertion de données cachées haute capacité dans un nuage de points 3D

6.1 Introduction

Dans ce chapitre, la méthode proposée se concentre principalement sur la caractéristique de capacité. Pour rappel, nous avons défini les contraintes entre la capacité et la robustesse dans la Section 3.2. L'approche proposée repose sur une étape d'ordonnancement des sommets grâce à un chemin hamiltonien [43]. Cette phase de synchronisation est essentielle pour effectuer exactement le même trajet entre l'insertion et l'étape d'extraction. Nous proposons de construire un chemin hamiltonien et l'insertion du message de façon conjointe, comme présenté dans la Fig. 6.1. Cette figure illustre le comportement de deux méthodes d'insertion que nous présentons dans ce chapitre. Après un recalage du maillage par une ACP et la construction du graphe complet, l'algorithme commence la construction du chemin, qui est analysée dans la Section 6.2. À partir d'un sommet donné, pour chaque sommet ajouté au chemin, l'algorithme insère une partie du message. L'étape d'insertion se fait sur les trois dimensions spatiales par de légers déplacements des sommets. L'idée est de déplacer le sommet dans une intersection d'intervalles qui correspond aux données à insérer. Cette méthode est basée sur les méthodes de type Quantization Index Modulation (QIM) proposé par Chen et Wornell [16], en particulier celle de Chao *et al.* [15]. Dans la Section 6.3, nous présentons la première méthode qui offre une capacité de 3 bps, en gérant les problèmes de causalité.

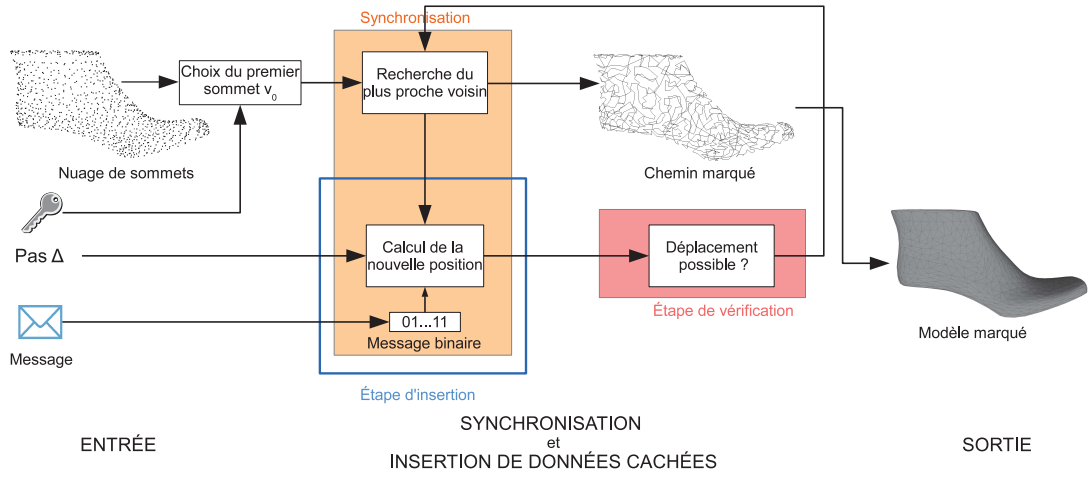


FIGURE 6.1: Schéma général de la méthode d'IDC proposée.

Puis, dans la Section 6.4, nous étendons cette méthode pour obtenir une capacité adaptative qui peut atteindre 24 bps. Nous proposons également une extension permettant une meilleure indétectabilité.

6.2 Analyse de la synchronisation par chemin hamiltonien

Dans cette section, nous analysons la nouvelle synchronisation introduite dans [50], qui ne dépend que de la géométrie du maillage et non des relations entre arêtes. Ainsi, une attaque sur la connectivité ne peut pas entraîner la perte de l'ordre défini sur les sommets. L'idée principale est de construire une structure ordonnée sur les sommets. Cet ordre doit être stable, c'est-à-dire ne pas changer entre deux réalisations du chemin, nous vérifions la stabilité du chemin lors d'un déplacement de sommets. Pour simplifier le problème, nous supposons le déplacement d'un sommet à la fois. Dans un contexte d'IDC, les zones où sont cachées les données doivent être synchronisées suivant le même ordre à l'insertion et à l'extraction. Cependant, le déplacement d'un sommet à l'insertion peut introduire un problème de causalité, qui se produit lorsque l'insertion provoque une désynchronisation. Notre analyse permet de déterminer si les méthodes d'IDC, présentées Section 6.3 et Section 6.4, engendrent ce problème.

Nous analysons la construction d'un chemin hamiltonien, dans le cas d'insertion de données cachées dans un maillage 3D comportant $|V|$ sommets. Si on considère une insertion en déplaçant les sommets, il faut préserver le chemin de façon à ce que les données insérées soit extraites dans leur ordre d'insertion. Nous simplifions le problème en considérant le déplacement d'un sommet à chaque étape de la construction du chemin. Il faut donc maintenir le chemin construit à l'étape précédente soit le sous chemin $\mathbf{P_{i+1}}$, lors du déplacement du sommet v_{i+1}. À l'itération i de la construction du chemin, un

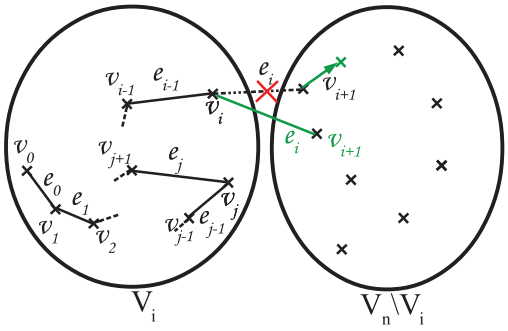


FIGURE 6.2: v_{i+1} est déplacé trop près du sous-chemin.

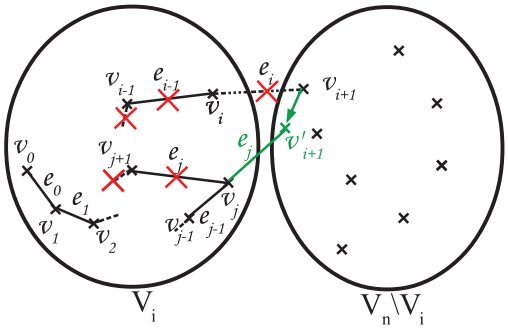


FIGURE 6.3: v_{i+1} est déplacé trop loin de son père.

sommet est soit dans le chemin, soit dans l'ensemble des sommets à ajouter. Un sommet $v_j, j \in [0, |V|-1]$ est dans $\mathbf{P_i}$ si $j \in [0, i]$. Ainsi, il y a deux cas pour un sommet v_j, $j \neq i+1$:

(i). $j \in [0, i] \Leftrightarrow v_j, \in V_i$,

(ii). $j \in [i+2, n] \Leftrightarrow v_j, \in V_n \backslash V_i$.

Pour le premier cas, (i), le sommet v_{i+1} pourrait être déplacé trop près du sous-chemin P_i et un sommet de l'ensemble des sommets utilisés $v_j \in V_i, j \neq i$ pourrait devenir son prédécesseur, comme illustré dans la Fig. 6.2. Le sommet v_{i+1} est déplacé à une nouvelle position v'_{i+1} proche de v_j, alors la distance $d_i = \|v_i, v'_{i+1}\|_2$ de e_i devient supérieure à la distance $d_j = \|v_j, v'_{i+1}\|_2$ de e_j. En conséquence, à l'étape de décodage, l'arête e_j entre v_j et v'_{i+1} serait choisie et celle-ci changerait le chemin hamiltonien. La contrainte à respecter est donnée par :

$$\|v_j, v_{j+1}\|_2 < \|v_j, v'_{i+1}\|_2, \ \forall v_j \in V_i, \ j \neq i, \tag{6.1}$$

Dans le second cas (ii), le sommet v_{i+1} est déplacé à une nouvelle position v'_{i+1} qui est trop loin de son prédécesseur dans le chemin appelé père. Dans cette situation, un autre sommet $v_j \in V_n \backslash V_i$ devient le plus proche de v_i. Ce cas est illustré dans la Fig. 6.3 qui montre que l'autre sommet sera choisi à l'étape d'extraction. Il faut donc que :

$$\|v_i, v'_{i+1}\|_2 < \|v_i, v_j\|_2, \ \forall v_j \in V_n \backslash V_i, \ j \neq i+1. \tag{6.2}$$

Nous pouvons définir un rayon de déplacement pour chaque sommet, on pose r^- la distance maximale du rapprochement du sommet vers le sous-chemin au delà de laquelle le sommet courant est choisi avant dans la construction du chemin à l'extraction. Réciproquement, on note r^+ la distance maximale d'éloignement par rapport au

sous-chemin, à partir de laquelle le sommet ne sera pas choisi à la même itération à l'extraction. Le rayon est alors donné par :

$$r_{i+1} = min(r^{+}_{i+1}, r^{-}_{i+1}), \tag{6.3}$$

où r^{-} et r^{+} sont obtenus par :

$$r^{-}_{i+1} = min(\|v_j, v_{i+1}\|_2 - \|v_j, v_{j+1}\|_2),\ v_j \in V_i,\ j \neq i, \tag{6.4}$$

$$r^{+}_{i+1} = min(\|v_i, v_j\|_2) - \|v_i, v_{i+1}\|_2,\ v_j \in V_n \backslash V_i,\ j \neq i{+}1. \tag{6.5}$$

Ce rayon est intéressant pour définir des sommets plus sensibles ou plus robustes comme dans [54]. On peut également chercher un chemin plus robuste en fonction du point d'entrée. Néanmoins, ce rayon est très restrictif sur la direction du déplacement, nous proposons de contrôler sommet par sommet chaque déplacement. Cette étape consiste à vérifier les équations 6.1 et 6.2 pour chaque déplacement. Si les équations sont respectées, alors l'algorithme continue, au contraire si un déplacement n'est pas valide il faut établir une stratégie.

Du coup, en utilisant cette synchronisation nous constatons que le déplacement des sommets, est limité. En effet, pour que le déplacement d'un sommet soit validé, il faut qu'il se fasse en respectant les conditions des équations 6.1 et 6.2. Le déplacement peut être borné pour réduire le nombre d'erreurs. Nous notons *md* la distance moyenne entre deux sommets du chemin hamiltonien :

$$md = \sum_{i=0}^{n} \omega(e_i), e_i \in \mathbf{P}_{|\mathbf{V}|}. \tag{6.6}$$

Pour estimer cette valeur avant la construction du chemin, nous pouvons calculer la valeur moyenne des arêtes d'un maillage. Ensuite, nous pouvons borner le déplacement par cette distance moyenne. Si le déplacement est de l'ordre de grandeur de *md*, alors nous pouvons dire que la synchronisation va interdire beaucoup de déplacements et donc produire beaucoup d'erreurs.

6.3 IDC basée sur la construction d'un chemin hamiltonien

Dans cette partie nous présentons une méthode d'IDC de haute capacité basée sur les analyses précédentes. Dans cette partie, l'objectif est de réaliser une méthode d'IDC

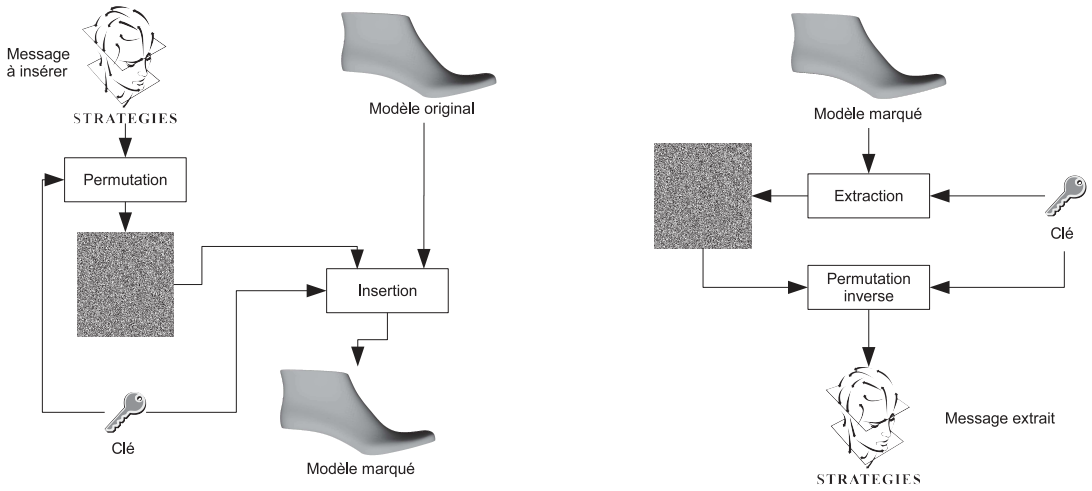


FIGURE 6.4: Schéma d'insertion et d'extraction d'un logo.

permettant l'insertion d'un logo de taille importante, comme illustré dans la Fig. 6.4. Dans un premier temps, le logo est chiffré. Il est ensuite inséré dans un maillage à l'aide d'une clé secrète. Cette clé permet l'extraction du logo et la vérification de la propriété du maillage. La clé secrète permet de générer aléatoirement une position 3D dans l'espace du volume englobant, le maillage étant recalé cette position reste inchangée après des transformations affines. Le premier sommet est alors obtenus en cherchant le sommet du maillage le plus proche de cette position. L'utilisation d'un point d'entrée aléatoire ne garantie pas d'avoir le chemin le plus stable, néanmoins il permet de produire des chemins différents dans le maillage ce qui le rend moins évident à retrouver sans clé, que l'utilisation d'un parcours unique, comme un chemin de poids minimum.

Nous proposons d'utiliser la construction du chemin hamiltonien pour obtenir une relation entre deux sommets. Cette relation est donnée par les arêtes du chemin. Pour chaque itération i de la construction du chemin, les données seront insérées en déplaçant légèrement le sommet v_{i+1}, par rapport à son prédécesseur v_i, à sa nouvelle position v'_{i+1}. Le processus d'insertion est joint à la synchronisation proposée, cela afin de connaître la nouvelle position du sommet à chaque itération et de maintenir le chemin inchangé. Nous proposons de déplacer le sommet sur les trois coordonnées ρ, θ et ϕ après avoir converti les coordonnées cartésiennes en coordonnées sphériques. La conversion se fait par rapport au sommet précédent, nous calculons le vecteur p_i entre $v_i(x_i, y_i, z_i)$ et $v_{i+1}(x_{i+1}, y_{i+1}, z_{i+1})$, tel que :

$$p_i = \begin{bmatrix} x_{i+1} - x_i \\ y_{i+1} - y_i \\ z_{i+1} - z_i \end{bmatrix} \tag{6.7}$$

Nous calculons ensuite les coordonnées sphériques de $p_i(\rho_i, \theta_i, \phi_i)$. Après l'insertion

et la conversion en coordonnées cartésiennes, la nouvelle valeur de v_{i+1}, notée v'_{i+1}, est calculée à l'aide de la nouvelle valeur de $p_i(p'_{x_i}, p'_{y_i}, p'_{z_i})$:

$$v'_{i+1} = \begin{bmatrix} p'_{x_i} + x_i \\ p'_{y_i} + y_i \\ p'_{z_i} + z_i \end{bmatrix}. \tag{6.8}$$

Nous présentons le processus d'insertion sur ρ puisque le même processus peut être appliqué sur toutes les coordonnées. Pour définir la valeur d'une arête e_i qui correspond à ρ_i, la valeur de ρ_i est alors divisée à l'aide d'un pas donné Δ. Chaque intervalle correspond à un bit du message caché, qui est alternativement un 0 ou un 1. Nous pouvons extraire la valeur b_i d'un sommet v_{i+1} par :

$$b_i = \left\lceil \left(\frac{\rho_i}{\Delta} \right) \right\rceil modulo(2), \tag{6.9}$$

où $\rho_i = \|v_i, v_{i+1}\|_2$, est la distance euclidienne entre v_i et v_{i+1}. La Fig. 6.5 illustre la division de la distance et la valeur correspondante à la position initiale du sommet v_{i+1}.

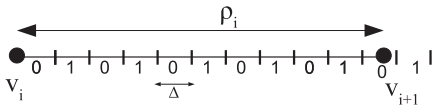


FIGURE 6.5: Division de la longueur d'une arête pour l'insertion sur ρ_i.

Puisque nous connaissons la valeur de b_i nous pouvons la comparer avec le bit m_i à insérer. Il existe alors deux cas, le bit du message et le bit lu sont soit identiques soit différents. S'ils sont égaux, nous devons calculer la nouvelle valeur ρ'_i de la coordonnée du sommet v_{i+1}. Cette valeur correspond au centre de l'intervalle courant et la position du sommet est mise à jour et notée v'_{i+1}. Si les bits sont différents, nous déplaçons le sommet v_{i+1} en modifiant la valeur de ρ_i centre de l'intervalle adjacent le plus proche, à sa nouvelle position v'_{i+1}. La Fig. 6.6 illustre les deux cas, en bleu, l'égalité, en rouge, la différence.

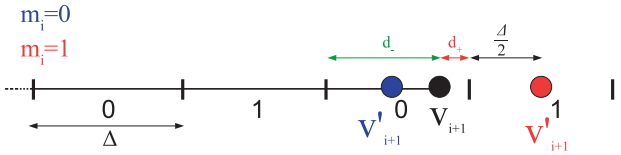


FIGURE 6.6: Déplacement du sommet en fonction de ρ_i.

Pour trouver le centre du plus proche intervalle adjacent qui minimise $\|v_{i+1}, v'_{i+1}\|_2$, nous calculons d_+ et d_- qui sont les distances entre le sommet v_i des limites de l'intervalle :

$$d_+ = \lceil (\frac{\rho_i}{\Delta}) \rceil \times \Delta - \rho_i, \tag{6.10}$$

$$d_- = \rho_i - \lfloor (\frac{\rho_i}{\Delta}) \rfloor \times \Delta. \tag{6.11}$$

Puis, nous obtenons d_m :

$$d_m = min(d_+, d_-) + \frac{\Delta}{2}. \tag{6.12}$$

La dernière étape est d'effectuer la transformée inverse de v_{i+1} des coordonnées sphériques en coordonnées cartésiennes. De plus, en utilisant la même stratégie sur θ et ϕ nous pouvons insérer 3 bps. Comme nous pouvons répéter l'insertion pour $i \in [0, |V|-1]$, il est possible d'insérer $3 \times (|V|-1)$ bits dans le maillage d'un objet 3D de $|V|$ sommets. Néanmoins, un problème peut se produire quand nous nous déplaçons un sommet, en effet le déplacement peut changer la synchronisation à l'étape d'extraction. Pour résoudre ce problème, nous proposons d'utiliser les résultats de la Section 6.2.

6.3.1 Algorithme de la méthode

La Fig. 6.1 illustre le schéma d'insertion proposé, il possède trois grandes étapes, l'ordonnancement, l'insertion et la vérification du déplacement. Ces étapes sont réitérées pour chaque sommet $v_i, i \in [0, |V|-1]$. L'algorithme 1 décrit la méthode d'IDC proposée. En entrée de l'algorithme, il faut un sommet de départ v_0 qui est donné par la clé secrète k comme expliqué dans l'introduction. L'algorithme reçoit également le message à insérer $\mathbf{M}$ et une structure de liste P stockant le chemin hamiltonien sur le graphe G_n. La première étape consiste en la recherche du plus proche voisin qui est effectuée par la fonction $recherchePlusProcheVoisin()$ à la ligne 4 de l'algorithme 1. La seconde étape est l'insertion décrite dans la Section 6.3 et appelée $nouvellePosition()$ ligne 6. La dernière grande étape est une étape de vérification pour déterminer si un problème de désynchronisation est induit. Ce résultat est donné par la fonction $verficication()$ à la ligne 5. Comme expliqué Section 6.2, si une désynchronisation peut se produire, nous devons avoir une stratégie. Si le déplacement d'un sommet n'est pas compatible avec les équations 6.1 et 6.2, nous choisissons d'interdire le déplacement, pour que le sommet soit choisi à la même étape lors de la reconstruction du chemin. Dans ce cas, le message ne peut pas être dissimulé dans cette arête.

Algorithme 1 Insertion jointe à la synchronisation

Entrée : $P = (E, V_{in}), V_{out}, k, M, G_n = (E_m, V_n)$

$V_{out} = V_n$
$v_1 \leftarrow v_k \in V_{out}$
tant que $V_{out} \neq \emptyset$ **faire**
  $v_2 \leftarrow rechercheProcheVoisin(v_1)$
  si $verficication()$ **alors**
    $v_2 \leftarrow nouvellePosition(v_2, M)$
  fin si
  $V_{in} \leftarrow \{v_1\}$
  $V_{out} \leftarrow V_{out} \setminus \{v_1\}$
  $E \leftarrow \{v_1, v_2\}$
  $v_1 \leftarrow v_2$
fin tant que

6.3.2 Code correcteur d'erreurs

Dans la Section 6.3.1, nous nous permettons donc de perdre trois bits afin de conserver la synchronisation entre l'insertion et l'extraction. La perte de bits du message produit une perte d'informations et un bruit sur le logo extrait. Nous proposons d'utiliser un CCE (Code Correcteur d'Erreurs) pour corriger les erreurs commises lors de l'insertion. Nous avons choisi le code de Golay $(23, 12, 7)$ [38], qui est un code classique de la littérature simple à utiliser. Il ajoute 11 bits correcteurs pour 12 bits de message, afin de produire le message à insérer. Après l'étape d'extraction, le message est décodé en utilisant une méthode rapide proposée par Chang *et al.* [14] qui permet de corriger jusqu'à 3 bits par bloc de 23 bits du message. Bien entendu, la taille du message utile est réduite et sa taille s_m donnée par :

$$s_m = \lfloor \frac{3(|V| - 1)}{23} \rfloor \times 12. \tag{6.13}$$

En outre, en utilisant une clé pseudo-aléatoire pour brouiller le message par permutation, la distribution des erreurs est uniforme sur le message déchiffré. Ne pas avoir trop d'erreurs dans le même bloc améliore le taux de correction. En effet, des erreurs sont générées par l'étape de vérification qui dépend du voisinage d'un sommet. Puisque le maillage peut avoir des zones de densités différentes, nous pouvons dire que les erreurs ne sont pas réparties uniformément dans le maillage marqué.

6.3.3 Extraction du message caché

La Fig. 6.7 illustre le schéma d'extraction. Connaissant le sommet initial v_0, le pas choisi Δ et la clé pseudo-aléatoire utilisée pour chiffrer le message, il est facile d'extraire le

message inséré en construisant le chemin hamiltonien et de “lire” les trois bits contenus dans chaque sommet en utilisant l'équation 6.9. Ensuite, l'ensemble du message est déchiffré en utilisant la clé secrète. Une fois déchiffré, il peut être corrigé si un CCE a été utilisé.

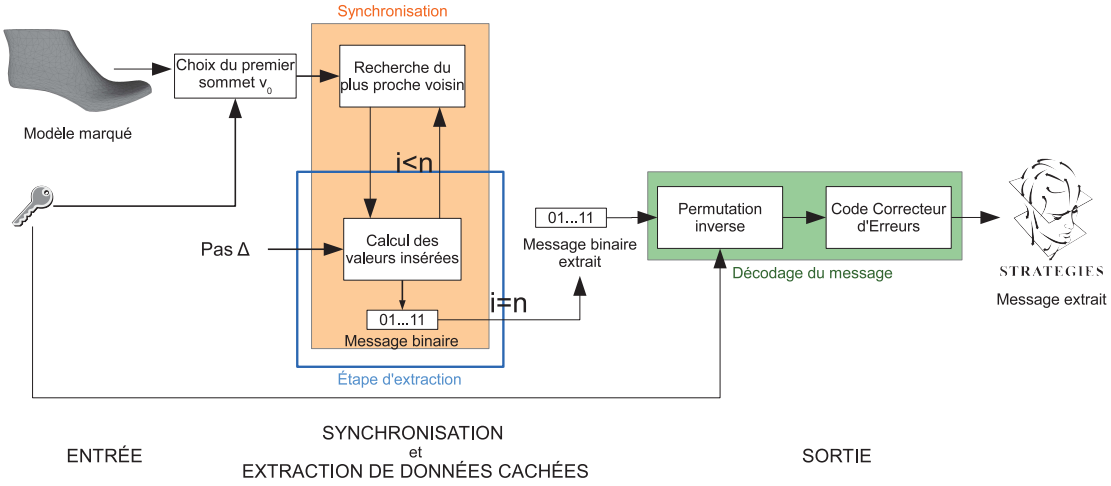


FIGURE 6.7: Extraction du message caché.

Établir la sécurité de la méthode seulement sur le premier sommet n'est pas suffisant pour assurer un niveau de sécurité élevé. Mais nous pensons que la complexité d'estimation des paramètres ainsi que le chiffrement du message sont suffisants pour assurer un bon niveau de sécurité. Néanmoins, nous pensons qu'il est intéressant d'évaluer plus précisément la sécurité de notre méthode, en se basant sur les travaux de Cayre *et al.* [13]. Cet objectif majeur est présenté dans le cas généralisé dans la Section 6.4.

6.3.4 Résultats expérimentaux

Dans cette section, nous présentons les résultats de l'insertion d'un logo dans un maillage grâce à la méthode d'IDC proposée. Dans un premier temps, nous présentons le comportement du processus d'IDC sur un exemple complet. Ensuite, nous discutons des améliorations proposées à l'aide d'exemples. Enfin, une expérimentation complète sur de nombreux objets différents est proposée.

6.3.4.1 Exemple d'application

Dans cette section, nous présentons un exemple détaillé de la méthode appliquée à la forme de l'intérieur d'une chaussure. La Fig. 6.8 présente le maillage qui est utilisé pour illustrer les résultats. Cet objet est une version simplifiée du maillage original [1] et

1. STRATEGIES S.A. http ://www.cadwin.com

FIGURE 6.8: Maillage ShoeS avec 45002 sommets.

possède 45002 sommets et 90000 facettes. Le maillage est normalisé afin de comparer les résultats obtenus sur différents maillages, avec les mêmes paramètres. La normalisation est donnée par un facteur d'échelle k qui dépend de la taille du volume englobant :

$$k = max\{x_{max} - x_{min}, y_{max} - y_{min}, z_{max} - z_{min}\}, \tag{6.14}$$

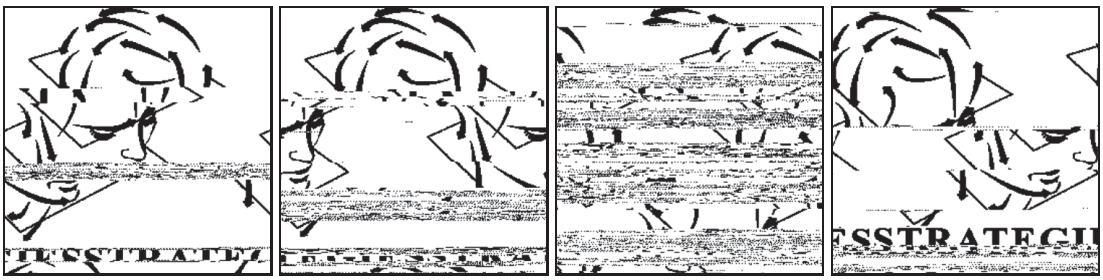
il fixe la taille du plus grand coté du volume englobant à 1. La capacité du système d'IDC de base sans l'ajout de CCE, permet de cacher un logo binaire carré de taille :

$$s = \lfloor\sqrt{3(|V| - 1)}\rfloor \times \lfloor\sqrt{3(|V| - 1)}\rfloor. \tag{6.15}$$

La Fig. 6.9 présente le logo à insérer, et sa taille est donnée par l'équation 6.15. Cette image représente en fait le logo de la société STRATEGIES[1].

FIGURE 6.9: Logo redimensionné : 367×367 pixels.

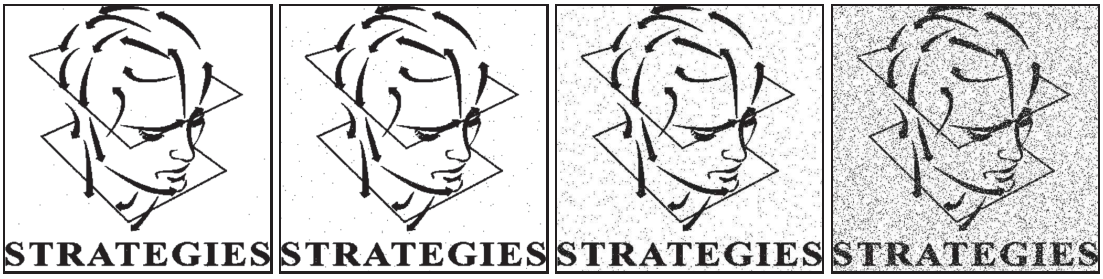
Pour bien comprendre l'aspect majeur de la synchronisation, la Fig. 6.10 présente les résultats du procédé sans l'étape de vérification. Nous utilisons la méthode proposée avec un pas $\Delta = 10^{-6}$ pour insérer le logo dans le maillage, il est ensuite extrait. Nous proposons d'illustrer ce résultat par quatre exemples du processus d'insertion/extraction avec quatre sommets de départ différents et donc quatre clés. Notons l'importance de la synchronisation et également du choix du premier sommet.



clé : #1 clé : #2 clé : #3 clé : #4

FIGURE 6.10: Logos extraits pour différentes clés, pas : $\Delta = 10^{-6}$.

Ces résultats nous montre, l'importance de vérifier l'ordonnancement avec l'étape autorisant un déplacement. Nous présentons d'autres expériences utilisant l'étape de vérification pour maintenir la synchronisation. La Fig. 6.11 présente les logos binaires extraits. Nous analysons leur qualité en comparant ceux extraits avec l'original.



pas : $\Delta = 10^{-6}$ pas : $\Delta = 10^{-5}$ pas : $\Delta = 10^{-4}$ pas : $\Delta = 10^{-3}$

FIGURE 6.11: Logos extraits avec préservation du chemin, en fonction du pas d'insertion Δ.

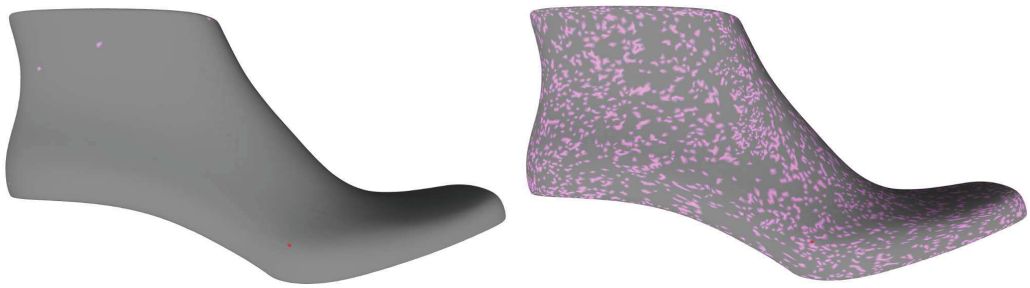
Le Tableau 6.1 présente les valeurs de CCN [39] (Corrélation Croisée Normalisée) entre le logo original et ceux extraits. Plus la valeur est proche de 1 plus les images sont corrélées, à l'inverse plus elle se rapproche de 0 moins elles sont corrélées. Dans ce contexte, nous avons choisi de présenter les résultats en termes de CCN, car c'est une métrique classique de comparaison d'image au contraire du taux d'erreurs BER (Bit Error Rate) qui est plus intéressant en traitement du signal pour comparer les flux de bits.

TABLEAU 6.1: CCN entre le logo original est les logos extraits en fonction du pas d'insertion Δ.

Pas Δ	10^{-6}	10^{-5}	10^{-4}	10^{-3}
CCN	0.9997	0.9973	0.9755	0.8032

Nous constatons, dans la Fig. 6.11, que le message est bruité, ce qui est dû à l'étape de vérification qui interdit certains déplacements de sommets. Cependant, cette vérification garde la synchronisation inchangée et le logo reste reconnaissable. De plus, le Tableau 6.1

présente des bons résultats en terme de CCN pour des pas Δ inférieurs à 10^{-3}. Toutefois, le logo est encore reconnaissable pour $\Delta = 10^{-3}$. La limite de la valeur du pas dépend de la distance moyenne md, entre deux sommets du chemin hamiltonien. Dans cet exemple le maillage présenté dans la Fig. 6.8 a une distance moyenne de $md = 4.40368 \times 10^{-3}$, calculé avec l'équation 6.6. En fixant le pas avec un même ordre de grandeur $\Delta = 10^{-3}$, les erreurs sont très fréquentes. La Fig. 6.12 illustre les difficultés à déplacer un sommet lorsque le pas est trop proche de md.



pas : $\Delta = 10^{-6}$ pas : $\Delta = 10^{-3}$

FIGURE 6.12: Déplacement des sommets en fonction du pas Δ, les sommets ne pouvant pas être déplacés en rose.

Nous pouvons voir que de nombreux sommets ne peuvent pas être déplacés et qu'ils ne sont pas distribués avec la même probabilité sur tout le maillage. Le pourcentage de sommets non-déplacés pour différentes valeurs du pas Δ est donné dans le Tableau 6.2. Nous notons que le nombre de sommets ne pouvant être déplacés est directement influencé par la valeur de Δ.

TABLEAU 6.2: Pourcentage de sommets non-déplacés en fonction de Δ.

Pas Δ	10^{-6}	10^{-5}	10^{-4}	10^{-3}
Pourcentage de sommets non-déplacés	0.035 %	0.253 %	1.857 %	16.957 %

6.3.4.2 Méthode améliorée avec codes correcteurs d'erreurs

Pour améliorer ces résultats, nous ajoutons au message à insérer le CCE présenté dans la Section 6.3.2, cela afin de réduire le bruit. La charge utile change puisqu'une partie de la capacité maximale est utilisée pour ajouter de l'information redondante. Le nombre de bits du message est alors donné par l'équation 6.13. La taille s_m du logo carré à insérer est de :

$$s_m = \lfloor \sqrt{\lfloor \frac{3(|V|-1)}{23} \rfloor \times 12} \rfloor \times \lfloor \sqrt{\lfloor \frac{3(|V|-1)}{23} \rfloor \times 12} \rfloor. \tag{6.16}$$

FIGURE 6.13: Logo redimensionné : 265 × 265 pixels.

La Fig. 6.13 présente le logo redimensionné avec l'équation 6.16, nous l'insérons pour différentes valeurs du pas dans le maillage de forme de chaussure présenté dans la Fig. 6.8. La Fig. 6.14 donne les résultats de l'extraction du logo binaire caché. Pour les différentes valeurs du pas Δ, nous extrayons le message et nous corrigeons les erreurs pour obtenir le logo. Pour une comparaison visuelle, nous présentons les résultats de l'extraction avec ou sans correction pour chaque valeur de Δ. Les logos notés (a) ne sont pas corrigés, tandis que ceux notés (b) correspondent au message décodé avec le CCE.

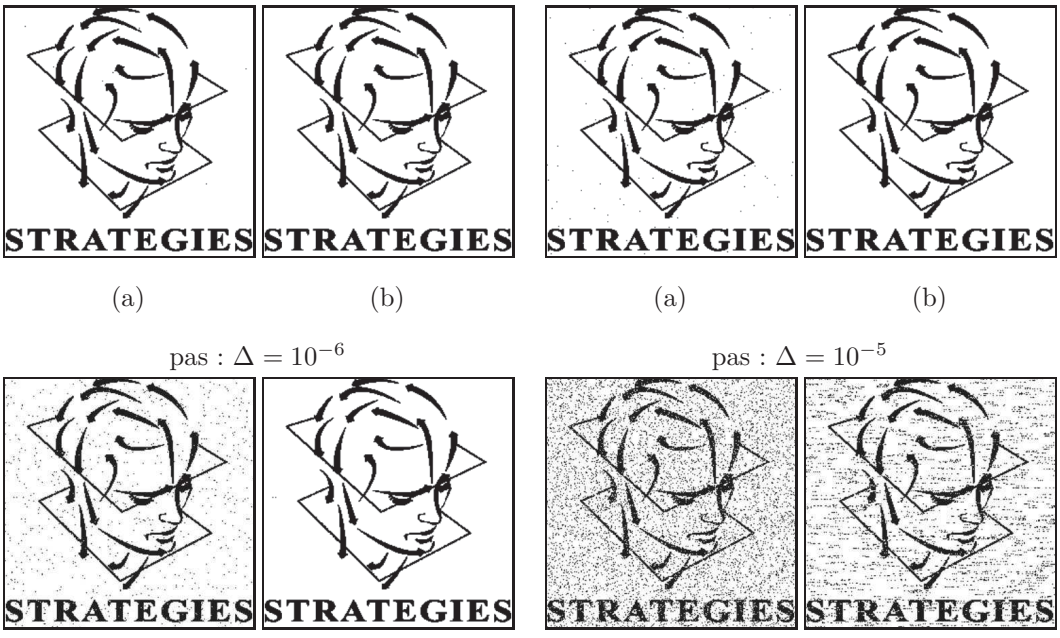


FIGURE 6.14: Logos extraits en fonction du pas Δ : a) sans CCE, b) avec CCE.

Pour appuyer ces résultats visuels, le Tableau 6.3 présente la CCN entre le logo original et les logos extraits avec ou sans CCE.

Nous constatons que sans l'ajout du CCE, les résultats sont corrects pour $\Delta < 10^{-3}$ mais ils sont presque parfaits en utilisant le CCE. Pour $\Delta = 10^{-3}$, la CCN chute à cause

TABLEAU 6.3: CCN entre le logo original et les logos extraits.

Pas Δ	10^{-6}	10^{-5}	10^{-4}	10^{-3}
sans CEE	0.9998	0.9974	0.9764	0.8051
avec CEE	1	1	0.9999	0.8796

du nombre de sommets non-déplacés puisque la valeur du pas atteint l'ordre de grandeur de la moyenne du poids des arêtes du chemin hamiltonien. La Fig. 6.15 illustre le ratio entre la CCN et le pourcentage de sommets non-déplacés. Évidemment, le pourcentage de sommets non-déplacés est corrélé avec la valeur du pas Δ. Nous pouvons également noter que l'ajout de CCE tend à stabiliser la courbe et elle décroit moins vite que sans l'utilisation du CCE.

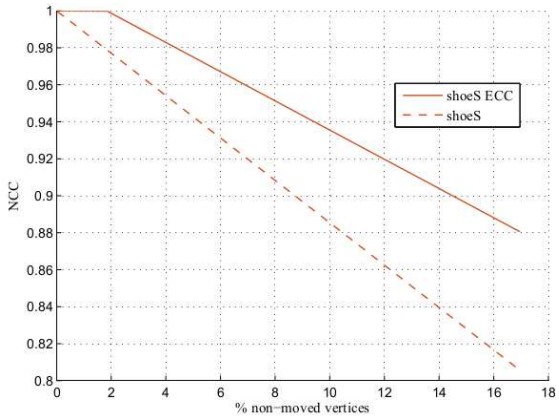


FIGURE 6.15: CCN en fonction du nombre de sommets non-déplacés.

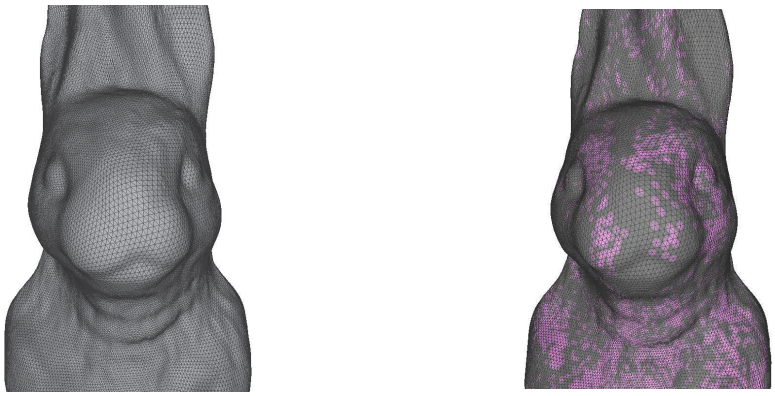
Nous nous sommes également intéressés aux distorsions qui sont produites sur le maillage par l'IDC. Le Tableau 6.4 présente la différence entre le maillage initial et ceux marqués. Les résultats sont présentés à l'aide de trois métriques présentées dans le Chapitre 3, à savoir la distance de Hausdorff, l'erreur quadratique moyenne RMSE (Root Mean Square Error) en utilisant Metro [19] et le MSDM2 de Lavoué [71] qui est mieux corrélé avec la vision humaine.

TABLEAU 6.4: Différences entre le maillage initial et les maillages marqués.

Step Δ	distance de Hausdorff	RMSE	MSDM2
10^{-6}	1.10^{-6}	0	0.00259
10^{-5}	10.10^{-6}	0	0.00310
10^{-4}	82.10^{-6}	2.10^{-6}	0.03556
10^{-3}	724.10^{-6}	16.10^{-6}	0.16246

Ces résultats montrent que le schéma d'insertion proposé produit des distorsions presque imperceptibles. En effet, les distances obtenues par les méthodes classiques sont très faibles et les distances données par la métrique MSDM2 sont proches de zéro, ce qui

correspond à une distorsion presque invisible pour le Système Visuel Humain (SVH). L'invisibilité du procédé est illustrée sur un exemple dans la Fig. 6.16.



(a) (b)

FIGURE 6.16: a) Maillage original de l'objet 3D "Rabbit", b) maillage marqué avec les sommets non-déplacés en rose.

De plus, dans la Fig. 6.17, nous analysons la distribution des déplacements des sommets pour chaque valeur du pas Δ.

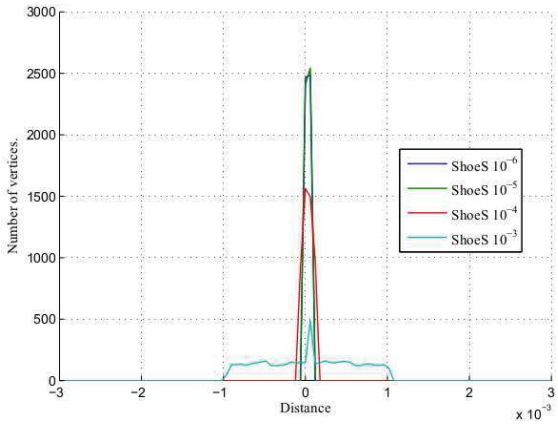


FIGURE 6.17: Distribution des déplacements en fonction de la valeur du pas Δ.

La distance de déplacement est définie comme négative si sa direction est orientée vers le centre du maillage, elle est positive sinon. Notons que les déplacements des sommets sont distribués également autour de l'axe y. En outre, une valeur importante du pas Δ produit une variance plus large, mais plus de déplacements ont une valeur nulle puisque plus de sommets ne sont pas déplacés.

6.3.4.3 Expérimentation sur une base hétérogène

Dans cette partie, nous présentons une expérimentation complète sur différents maillages provenant de différentes sources, Horse[2], Bunny[3], Hand[2], ShoeS[1], Dragon[3], Rabbit[2], Armadillo[3], et Shoe[1]. Le maillage "ShoeS" est la version simplifiée, présentée dans Section 6.3.4.1 et obtenue à partir du maillage "Shoe". Les maillages utilisés pour analyser le comportement général de la méthode ont des formes et des densités différentes. Ils sont présentés dans le Tableau 6.5, la première colonne donne le nombre de sommets, la seconde, la distance moyenne entre deux sommets dans le chemin hamiltonien, et la troisième le temps de l'ensemble du processus. Comme nous pouvons voir la distance moyenne est proportionnelle au nombre de sommets. Cela est dû à la normalisation des maillages calculés par l'équation 6.14. Cependant, notre implémentation a une complexité cubique en temps, notamment à cause de l'étape de vérification.

TABLEAU 6.5: Présentation des maillages.

Maillage	Horse	Bunny	Hand	ShoeS	Dragon	Rabbit	Armadillo	Shoe
# sommets	5002	34834	36616	45002	50000	70658	172974	188609
$md{\times}10^{-3}$	14.4384	6.2245	11.3314	4.4227	8.9123	6.0781	3.4081	2.2879
Tempsd'exécution	$4.5s$	$8'34s$	$9'58s$	$14'53s$	$24'25s$	$36'57s$	$3h52'$	$4h42'$

Les résultats en terme de CCN sont présentés dans la Fig. 6.18 qui permet de comparer les valeurs de CCN entre les logos insérés et les logos extraits pour différentes valeurs du pas. Pour chaque maillage, nous traçons deux courbes, une en pointillé qui correspond au logo extrait sans l'utilisation du CCE, l'autre en trait continu de la même couleur correspond au logo extrait après l'application du CCE.

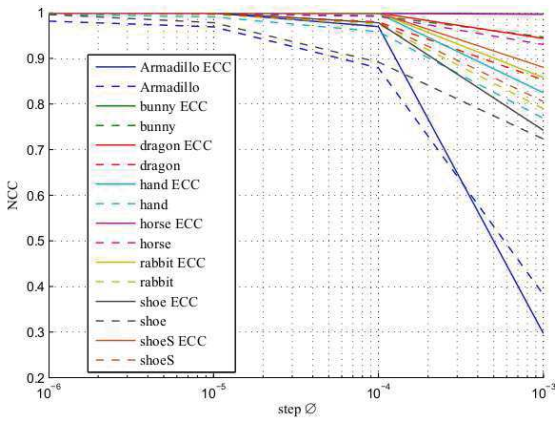


FIGURE 6.18: CCN entre le logo initial et les logos extraits en fonction du pas Δ.

Nous constatons que l'utilisation du CCE est presque toujours préférable. Par ailleurs, la CCN est supérieure à 0.97 pour $\Delta < 10^{-3}$. Pour les petits maillages, les

2. http ://www-rech.telecom-lille1.eu/madras
3. http ://www-graphics.stanford.edu

résultats restent bons avec $\Delta = 10^{-3}$ en raison de leur densité. En effet, leur valeur *md* est plus élevée que la valeur du pas, de sorte que leurs sommets peuvent être déplacés facilement. Au contraire, les maillages comportant plus de sommets, pour lesquels l'algorithme interdit plus de déplacements, ont une faible CCN. Cependant, comme la capacité est plus importante, la taille du logo caché est importante, le logo extrait est encore reconnaissable mais il est très bruité. Comme nous le montrons dans la Section 6.3.4.2, la valeur de la CCN est en corrélation avec le nombre de sommets non-déplacés. Le pourcentage de sommets non-déplacés dépend de la valeur du pas Δ et la Fig. 6.19 présente ces pourcentages pour les maillages étudiés.

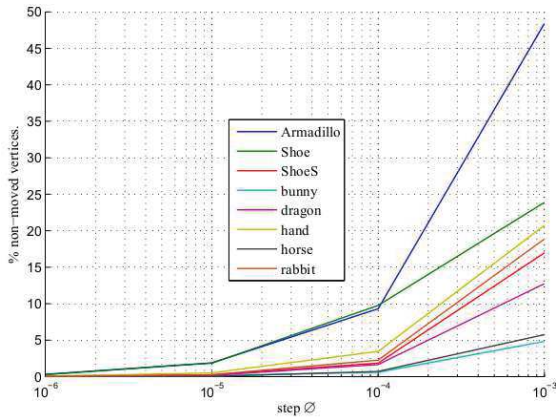


FIGURE 6.19: Pourcentage de sommets non-déplacés en fonction du pas Δ.

Lorsque nous étudions les résultats présentés Fig. 6.18 et Fig. 6.19, nous constatons que la CCN est inversement proportionnelle au pourcentage de sommets non-déplacés, ce qui confirme les hypothèses de corrélation. Afin de montrer comment la méthode d'insertion modifie le maillage, la Fig. 6.20 présente les distorsions mesurées par la distance de Hausdorff entre le maillage d'origine et celui marqué.

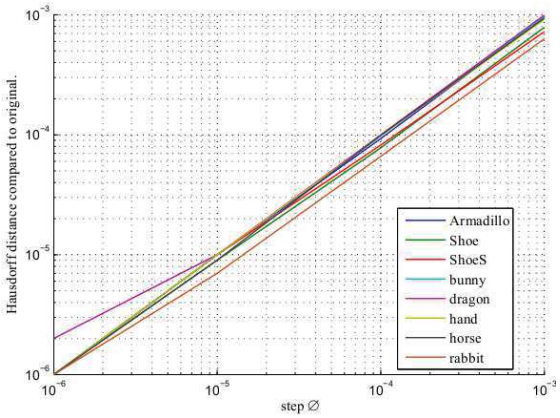


FIGURE 6.20: Distorsions mesurées en distance de Hausdorff en fonction du pas Δ.

En utilisant cette métrique géométrique, nous pouvons voir que les distorsions sont de l'ordre de grandeur de la valeur du pas. Ce résultat est très intéressant car il permet

de choisir la distorsion d'un maillage en fonction du pas choisi. Ici, la surface de l'objet n'est pas visiblement déformée. Afin de comparer notre méthode avec l'état de l'art, le Tableau 6.6 compare les résultats de la méthode en terme de capacité, de distance de Hausdorff et de $PSNR_1$ [15], pour le maillage de l'objet "Bunny" qui comporte 34834 sommets.

TABLEAU 6.6: Comparaison avec l'état de l'art sur le modèle "Bunny".

	Capacité	Distance de Hausdorff	$PSNR_1$
Chao *et al.* [15]	940464	×	100.57
Gao *et al.* [36]	51408	548.10^{-6}	×
Méthode proposée avec $\Delta = 1.10^{-6}$	54289	1.10^{-6}	127.3

Nous notons que pour une capacité équivalente, notre méthode déforme moins le modèle que la méthode de Gao *et al.*. Nous pouvons voir que notre capacité est inférieure à celle de Chao *et al.* mais pour un PSNR supérieur. Nous avons produit des résultats pour d'autres modèles avec $\Delta = 1.10^{-6}$, ces résultats sont présentés dans le Tableau 6.7.

TABLEAU 6.7: Capacité, distance de Hausdorff et $PSNR_1$ pour des maillages avec $\Delta = 1.10^{-6}$.

Modèle	# de sommets	Capacité	Distance de Hausdorff	$PSNR_1$
Horse	5002	7744	1.10^{-6}	127.635
Dragon	50000	77841	2.10^{-6}	132.247
Rabbit	70658	110224	1.10^{-6}	130.002
Venus	100759	157609	1.10^{-6}	133.582
Buddha	144628	225625	1.10^{-6}	111.985
Shoe	188209	294849	1.10^{-6}	126.087

Ces résultats montrent que notre méthode d'IDC produit de faibles distorsions tout en conservant une capacité intéressante, et que la distance de Hausdorff correspond bien au pas d'insertion.

6.4 Insertion de données cachées à très haute capacité

Dans cette section, nous présentons une méthode à très haute capacité basée sur la généralisation de la méthode présenté dans la Section 6.3. Nous fixons les bornes de la capacité de la méthode. Puis nous présentons deux façons de coder l'information dans un intervalle. Notre étude expérimentale montre les performances de la méthode en terme de capacité et d'imperceptibilité. Finalement nous analysons la sécurité de la méthode contre un potentiel attaquant.

6.4.1 Généralisation de la méthode d'insertion

Dans cette partie, nous présentons comment insérer un message $\mathbf{M}$ de taille $|\mathbf{M}| < n-1$ sur un alphabet $\mathcal{S} = \{s_0, ..., s_q\}$, dans un maillage de $|V|$ sommets. En utilisant la synchronisation définie dans la Section 6.2, le message $\mathbf{M}$ est inséré en déplaçant un sommet $v \in V$ à une nouvelle position notée v'. Notre synchronisation fournit un chemin $\mathbf{P}_{|}\mathbf{V}|$ sur le graphe $G_{|}V|$. Nous avons vu dans la Section 6.3 que la relation entre deux sommets du chemin u, v est donnée par l'arête $e\{u, v\}$ et son poids $\omega(e) \in \mathbb{R}^+$ est donné par la distance euclidienne $\|u, v\|_2$. A partir d'un sommet de départ v_0, le chemin est construit conjointement avec l'insertion. Pour chaque itération $i, 0 \leq i < n$ de la construction du chemin, le sommet v_{i+1} est déplacé par rapport à son prédécesseur v_i. Cette méthode nous permet d'insérer des données sur les $n-1$ arêtes du chemin $\mathbf{P_n}$. Par ailleurs, nous avons proposé dans la Section 6.3 de déplacer le sommet sur les trois composantes ρ, θ et ϕ après avoir converti les coordonnées cartésiennes en coordonnées sphériques. Cette nouvelle méthode définit un intervalle de valeur dans lequel les coordonnées du sommet peuvent être changées. Ces intervalles sont partitionnés en sous-intervalles et chaque sous-intervalle correspond à un mot s_j. Ensuite, pour insérer une partie du message, nous devons assigner la valeur de la coordonnée à une valeur appartenant au sous-intervalle correspondant. Cette méthode a une capacité théorique intéressante, et nous montrons la limite et les valeurs intéressantes du nombre de sous-intervalles.

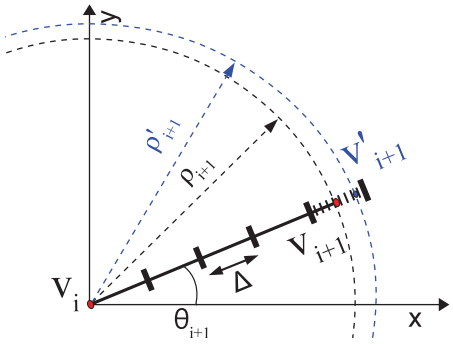


FIGURE 6.21: Déplacement de ρ_{i+1}, projeté sur un plan.

La Fig. 6.21 présente une projection dans un plan de $\mathbb{R}^2$, dans laquelle nous présentons notre méthode d'insertion sur la composante ρ. L'idée principale est de définir un intervalle de mobilité noté Δ dans lequel un composant ρ_{i+1} est déplacé à sa nouvelle valeur ρ'_{i+1}. L'intervalle est divisé en un nombre donné de sous-intervalles notés $\delta_x, x \in \mathbb{R}^+$, pour insérer une valeur s_j du message $\mathbf{M}$. De plus, en supposant que $x = q$, la méthode nous permet d'insérer une valeur $s_j, 0 \leq j < q$ en assignant la valeur ρ_{i+1} au sous-intervalle correspondant δ_j. La Fig. 6.22 illustre un exemple pour 8 sous-intervalles,

où le sommet v_{i+1} est déplacé à sa nouvelle position v'_{i+1} pour coder la valeur du mot s_6.

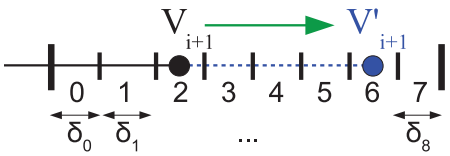


FIGURE 6.22: Déplacement de v_{i+1} à sa nouvelle position, dans le sous-intervalle correspondant à valeur du mot s_6.

Afin de trouver la borne inférieure b_l d'un intervalle, nous calculons :

$$b_l = \lfloor \frac{\rho_{i+1}}{\Delta} \rfloor \times \Delta. \tag{6.17}$$

La limite supérieure est simplement donnée par $b_u = b_l + \Delta$. Puis, comme nous souhaitons fixer une capacité par coordonnée à l'initialisation de la méthode, nous savons exactement les tailles de chaque sous-intervalle, qui sont données par une méthode uniforme ou une méthode de codage arithmétique statique, présentées respectivement dans les Section 6.4.3 et Section 6.4.4. Ainsi, les limites inférieures de chaque sous-intervalle correspondent à une valeur à coder s_w, où $w \in [0, x-1]$ est choisi en fonction de la capacité souhaitée. La nouvelle valeur c'_{i+1}, d'une coordonnée c_{i+1} d'un sommet v_{i+1} est :

$$c'_{i+1} = \begin{cases} b_l + w_b & \text{si } c_{i+1} < b_l + w_b \\ b_l + w_{b+1} - \gamma & \text{sinon,} \end{cases} \tag{6.18}$$

où γ est égal à $\frac{1}{k}$ de la taille du sous-intervalle $k \in \mathcal{N}, k > 1$.

Pour distribuer les valeurs dans les intervalles, nous pouvons aussi calculer :

$$c'_{i+1} = b_l + w_b + A, \tag{6.19}$$

où A est une variable aléatoire réelle suivant une distribution uniforme sur l'ensemble $[0, w_{b+1} - wb[$. Cette méthode produit plus de distorsions puisque, nous n'essayons plus de rapprocher la nouvelle valeur de l'ancienne.

Comme mentionné précédemment, les équations 6.18 et 6.19 sont valables pour chaque coordonnée sphérique du sommet, en fonction des coordonnées du père du sommet dans le chemin.

6.4.2 Analyse de la capacité

Dans la suite de cette section, nous expliquons comment x, le nombre de sous-intervalles, est défini et comment choisir la capacité de la méthode.

La capacité de la méthode dépend du nombre de divisions x de l'intervalle de déplacement. Pour évaluer x, nous distinguons deux paramètres importants, la taille de l'intervalle Δ et l'erreur numérique. En effet, ils sont liés en raison de la précision que nous pouvons atteindre avec la norme IEEE-754 qui standardise le codage des flottants. Le problème est de savoir comment définir Δ pour produire le moins de déformations sur le maillage, et la façon de choisir x en fonction de la limite de précision. La valeur de Δ dépend fortement du maillage (forme et densité), du chemin hamiltonien (voir la distance moyenne d'une arête Section 6.2) et la façon dont il est normalisé. Nous choisissons de laisser ce paramètre à l'appréciation des utilisateurs pour les résultats présentés dans la Section 6.4.5. Nous posons x, comme la valeur maximale qui peut être codée sur un nombre donné de bits c, alors $c = log_2(x)$. Dans le cas applicatif il est intéressant de pouvoir fixer une capacité et donc à partir d'une capacité souhaiter trouver le nombre de sous-intervalles. La capacité théorique totale de notre approche pour un maillage de $|V|$ sommets est donc la suivante :

$$cp = 3 \times log_2(x) \times (|V| - 1). \tag{6.20}$$

Si nous choisissons $c = 1$, nous avons un système binaire comparable à la méthode présentée dans la Section 6.3. C'est-à-dire que pour un message sur un alphabet binaire $\mathcal{S} = \{0, 1\}$, les sous-intervalles δ_0 ou δ_1 correspondent à un bit du message $\mathbf{M}$. D'un point de vue pratique, il semble qu'une valeur de $c = 8$, est intéressante car l'alphabet $\mathcal{S}$ peut être défini comme l'ensemble des mots codés sur un octet. En outre, cette valeur permet d'insérer un octet du message $\mathbf{M}$, sur chaque composante de chaque sommet. La capacité est alors de 24 bits par arête du chemin et notre idée est d'utiliser cette grande capacité pour insérer des objets complexes comme des images en couleur par exemple. Par ailleurs, pour avoir un ordre de grandeur de Δ, étant donné $c = 8$, si nous choisissons $\Delta = 10^n$, nous avons besoin d'une précision de mesure de 10^{n-3}. Cette valeur est réaliste pour la précision des nombres flottants qui sont utilisés par les formats de fichier binaire comme le STL [4] ou le PLY [5] par exemple, qui sont utilisés quotidiennement dans le domaine du traitement des maillages 3D dans le milieu industriel.

Dans les sections suivantes, nous présentons comment diviser l'intervalle Δ en x sous-intervalles. Dans la Section 6.4.3, nous expliquons le cas générique dans lequel nous

4. développé par 3D Systems
5. développé par l'université de Stanford

n'avons pas de connaissance *a priori* sur la distribution de messages et nous supposons qu'elle est uniforme. Puis, dans la Section 6.4.4, nous sommes en mesure d'utiliser la distribution de probabilité du message afin de subdiviser l'intervalle à l'aide du codage arithmétique statique.

6.4.3 Codage uniforme

Nous supposons que nous n'utilisons pas l'information du message, cela signifie que nous savons seulement que le message appartient à l'ensemble des mots sur un alphabet $\mathbf{M} \in \mathcal{S}^*$ et que sa taille est $|\mathbf{M}|$. Nous devons simplement supposer que chaque valeur s_j, a la même probabilité d'apparaître dans $\mathbf{M}$, alors : $p(s_j) = \frac{|\mathbf{M}|_{s_j}}{|\mathbf{M}|} = \frac{1}{|\mathcal{S}|}$. Ainsi, la probabilité de chaque valeur possible est $p(s_j) = \frac{1}{x}$, nous pouvons donc en déduire la taille d'un sous-intervalle δ_{s_j}, $|\delta_{s_j}| = \frac{\Delta}{x}$. Par exemple, la Fig. 6.23 présente comment Δ est divisé avec $c = 2$. Cette méthode est simple, mais elle possède quelques inconvénients comme le fait que certaines valeurs possibles s_j peuvent ne pas appartenir au message $\mathbf{M}$, $|\mathbf{M}|_{s_j} = 0$. Un autre problème concerne la sécurité, en effet la répartition uniforme des sommets dans l'intervalle peu être estimé et est présenté en détail dans la Section 6.4.6.

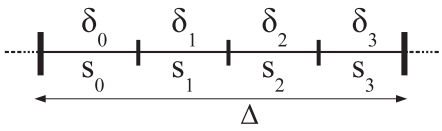


FIGURE 6.23: Exemple de division de Δ avec $c = 2$ en utilisant un codage uniforme, avec $p(s_i) = \frac{1}{4}$.

6.4.4 Codage adaptatif statique

Dans cette section, nous proposons de transformer la méthode uniforme en l'adaptant, en fonction des probabilités d'apparitions dans $\mathbf{M}$. La méthode est basée sur le Codage Arithmétique Statique (CAS), introduit par Langdon [69]. Le principe du CAS est de représenter une séquence de symboles par un intervalle d'un nombre réel entre 0 et 1. Chaque valeur dans cet intervalle correspond à un mot unique à coder. Le CAS commence à calculer les probabilités, il associe chaque symbole au sous-intervalle correspondant. La méthode proposée tient compte de la distribution du message d'origine pour adapter la valeur de l'intervalle de division Δ en q sous-intervalles, où $q \leq x$. En effet, nous savons exactement quelles lettres $s_k \in \mathcal{S}$ sont utilisées, et dans quelle proportion. Par conséquent, la probabilité de chaque valeur $s_j \in [0, q-1]$ est donnée par sa probabilité d'appartenance au message $\mathbf{M}$ donnée par $p(s_j) = \frac{|\mathbf{M}|_{s_j}}{|\mathbf{M}|}$. Chaque sous-intervalle δ_{s_j}, est d'une longueur qui dépend de la probabilité $p(s_j)$, $|\delta_{s_j}| = \frac{p(s_j)}{\Delta}$. Par

exemple, la Fig. 6.24 présente comment Δ est divisé avec $c = 2$, $p(s_0) = 0.15$, $p(s_1) = 0.4$, $p(s_2) = 0.2$, $p(s_3) = 0.25$.

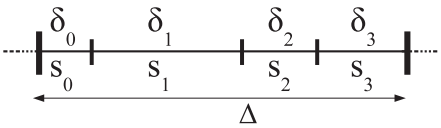


FIGURE 6.24: Exemple de division de Δ avec $c = 2$ en utilisant un CAS, avec $p(s_0) = 0.15$, $p(s_1) = 0.4$, $p(s_2) = 0.2$, $p(s_3) = 0.25$.

Notons que, comme nous voulons garder notre méthode aveugle c'est-à-dire sans *a priori* sur le maillage ou le message, nous devons insérer la distribution de probabilité dans le début du message. Ce qui affecte la capacité parce que cette insertion nécessite le même nombre de sommets que la taille de $\mathcal{S}$. Par rapport à la capacité de la méthode, cette perte est négligeable. Nous pouvons également considérer que la clé donne cette distribution de probabilité pour plus de sécurité.

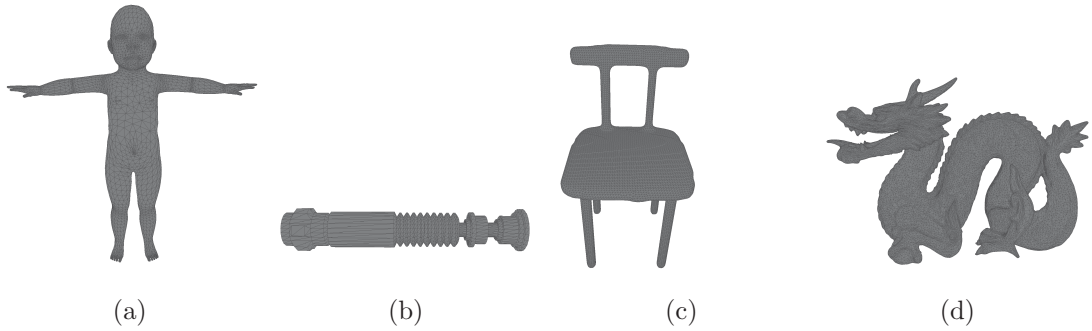
6.4.5 Résultats expérimentaux

Dans cette section, nous présentons des résultats expérimentaux. Dans un premier temps, nous introduisons les données considérées et le protocole. Ensuite, nous présentons l'ensemble des résultats. Dans nos expériences, nous considérons une base de données 31 maillages d'objets 3D [6], ceux-ci fournissent une diversité représentative de formes et tailles. Le nombre de sommets des maillages 3D est compris entre 1000 et 200000. Dans le but d'utiliser Δ comme un paramètre commun, nous normalisons la base en fixant la distance moyenne des arêtes à la valeur 1. Pour illustrer la diversité des formes et des maillages nous présentons quatre maillages représentatifs de la base de données dans la Fig. 5.9 et la Fig. 6.25.

Cependant, l'écart-type peut varier fortement entre deux maillages, comme illustré dans la Fig. 6.26. En moyenne le poids d'une arête est de 1 à cause de la normalisation, l'écart-type moyen est de 0.47, les valeurs minimales de l'écart-type sont autours de 0.15 et les valeurs maximales supérieures à 1 pour les maillages de CAD (Crank et Cad).

Dans la Fig. 6.27 nous présentons un exemple complet de la méthode proposée. Nous utilisons un objet 3D de forme de chaussure qui est décimé afin d'avoir environ 1000 sommets (dans le but de présenter des chiffres lisibles et de voir à quoi le chemin hamiltonien ressemble). Cet objet nous permet d'insérer $1001 \times 24 = 24024$ bits. Ce qui équivaux à environ une image RVB de 31×31 pixels ($31^2 \times 3 \times 8 = 23064$ bits).

6. Données fournies par Strategies S.A, le projet MADRAS, l'université de Stanford, et LGMA



(a) (b) (c) (d)

FIGURE 6.25: Échantillons utilisés : a) Baby 5075 sommets, b) CAD 1426 sommets, c) Chair1 12326 sommets, d) Dragon 50000 sommets.

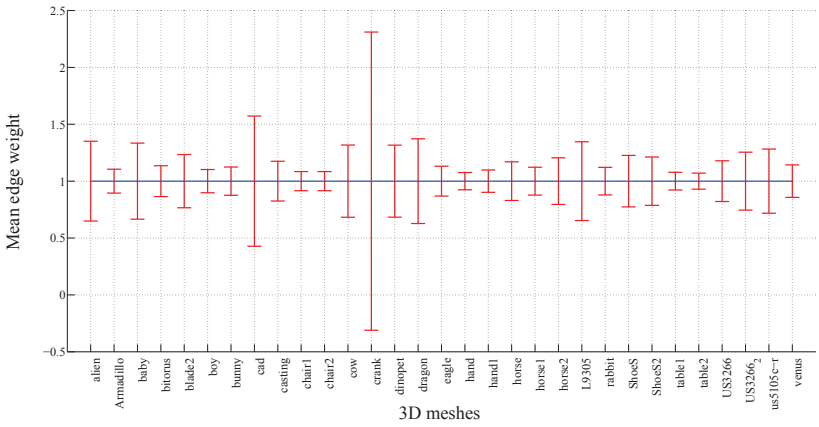


FIGURE 6.26: L'écart-type du poids des arêtes de notre base de maillages 3D.

À partir d'un maillage original présenté dans la Fig. 6.27.a, nous considérons seulement sa géométrie *i.e.* les positions des sommets qui correspondent au nuage de sommets illustré dans la Fig. 6.27.b. Ensuite, dans la Fig. 6.27.c nous présentons le chemin construit pendant l'étape d'IDC. Le chemin dans la Fig. 6.27.d, est le parcours effectué à l'aide la clé secrète pour extraire les données cachées du maillage marqué. Nous constatons que le même ordre de parcours des sommets est effectué, ce qui est essentiel pour extraire tout le message inséré. Après, l'IDC dans le chemin hamiltonien, le maillage 3D est reconstruit grâce à l'information de connectivité, le maillage est présenté dans la Fig. 6.27.e. Le maillage initial et le maillage marqué sont comparés dans la Fig. 6.27.f. Pour l'ensemble de sommets une distance signée est calculée entre la position de deux sommets associés des deux maillages. Les distances faibles tendent vers le vert, les distances négatives plus importantes tendent vers le bleu, et les distances importantes positives tendent vers le rouge.

Nous constatons que les distorsions ne sont pas visibles, et qu'elles sont assez faibles et uniformes sur la surface, à l'exception de sommets particuliers. Les distorsions entre les deux maillages sont autour de 6.4×10^{-5} en terme de distance de Hausdorff. Sur

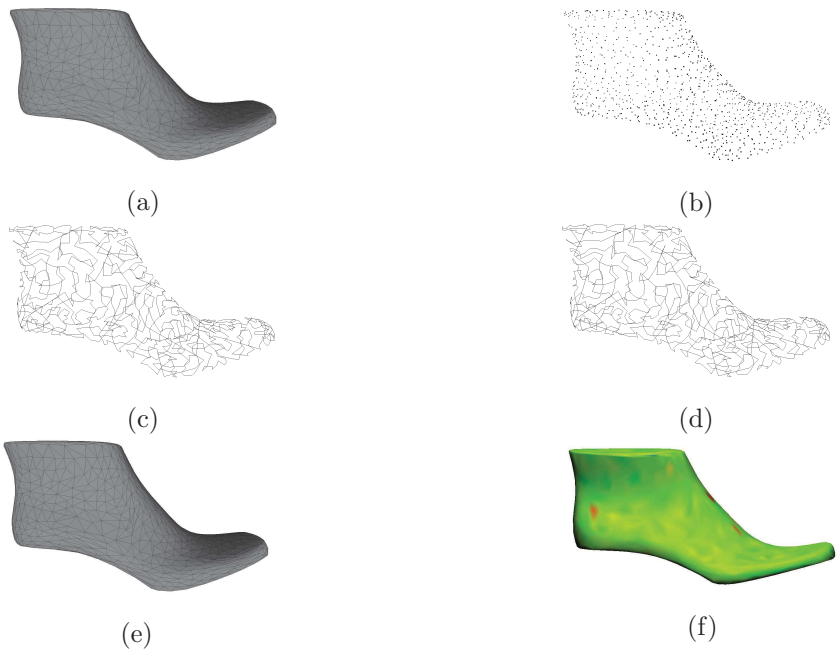


FIGURE 6.27: Exemple complet, a) Maillage initial avec 1002 sommets, b) nuage de points, c) chemin hamiltonien sans insertion, d) chemin hamiltonien marqué, e) maillage marqué avec la méthode CAS, f) comparaison entre le maillage initial et celui marqué.

cet exemple, tous les sommets sont déplacés ce qui conserve la synchronisation. Ainsi le message extrait ne comporte pas d'erreur et l'image extraite est exactement la même que celle insérée.

Dans le but de présenter des résultats plus représentatifs nous donnons un exemple de la méthode proposée pour le modèle "Bunny" qui comporte 34834 sommets. L'exemple complet est développé dans la Fig. 6.28. Nous utilisons un intervalle de déplacement fixé à $\Delta = 10^{-4}$ qui produit de faibles distorsions et des résultats comparables avec d'autres techniques de l'état de l'art. Nous fixons également la valeur $\gamma = \frac{1}{10}$ dans l'équation 6.18, lorsque la valeur d'une coordonnée est tirée jusqu'à la limite supérieure de son sous-intervalle. Le Tableau 6.8 présente les différences entre le message extrait pour chaque méthode en termes de PSNR et de BER (bit error rate).

TABLEAU 6.8: Qualité du message extrait, pour le maillage "Bunny".

	Méthode uniforme	Méthode CAS
# de sommets non-déplacés	1 (0.003%)	2 (0.006%)
PSNR (dB)	58.50	52.61
BER ($\times 10^{-3}$)	0.017	0.042

Nous constatons que, pour éviter les effets de désynchronisation, la méthode perd un faible pourcentage de l'information. Néanmoins, l'image extraite est de bonne qualité avec une valeur de PSNR d'environ 50 dB. Nous tenons à souligner qu'il est possible de

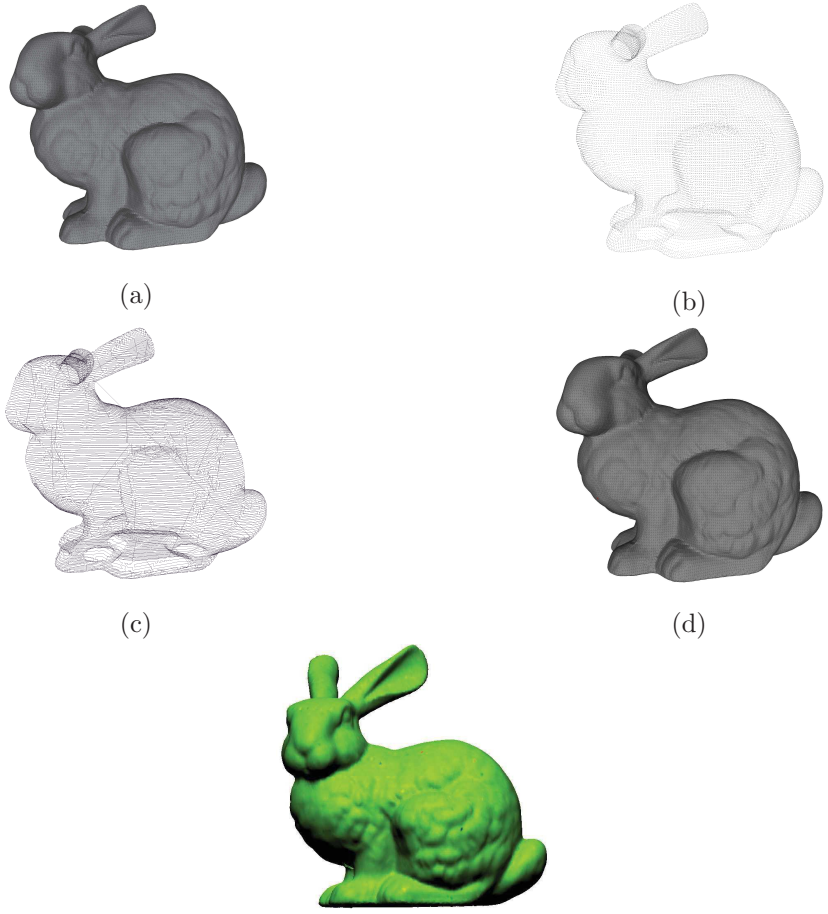


FIGURE 6.28: a) Maillage original de 34834 sommets, b) nuage de sommets, c) chemin hamiltonien construit sur le nuage de sommets, d) maillage marqué avec la méthode de CAS, e) comparaison des distorsions.

réparer les pixels faux par insertion d'un code correcteur d'erreurs ou d'utiliser des outils de traitement d'image *a posteriori*. La Fig. 6.29 présente les images extraites, l'image originale qui est insérée dans le maillage est présentée dans la Fig. 6.29.a. Ici, nous utilisons une image classique du traitement d'images, l'image "mandrill". La Fig. 6.29.b est l'image extraite avec la méthode uniforme, et la Fig. 6.29.c est l'image extraite avec la méthode CAS. La capacité maximale du maillage est de $|V| \times 24 = 836016$ bits et nous voulons insérer une image carrée. L'image RGB est donc redimensionnée à une taille de 186×186 pixels, ce qui correspond à une charge utile de $186 \times 186 \times 3 \times 8 = 830304$ bits. De plus, nous choisissons d'utiliser la distribution CAS comme la clé, afin d'avoir une comparaison équitable en terme de charge utile. Nous pouvons également intégrer la distribution dans des sommets identifiés dans le chemin pour la transmettre de façon cachée, il faut alors compter $256 \times 3 = 768$ sommets pour contenir cette distribution.

Nous pouvons voir dans la Fig. 6.29, que les images sont très similaires visuellement, mais certains pixels sont faux. Dans le but de comparer notre méthode avec celles de

(a)

(b)

(c)

FIGURE 6.29: a) Image insérée, b) Image extraite avec la méthode uniforme, c) Image extraite avec la méthode CAS.

l'état de l'art, le Tableau 6.9 présente les résultats de la méthode en terme de capacité de distance de Hausdorff [19] et de $PSNR_1$ [15].

TABLEAU 6.9: Comparaison avec l'état de l'art sur le maillage Bunny.

	Capacité (bits)	Distance de Hausdorff $\times 10^{-3}$	$PSNR_1$ (dB)
Chao *et al.* [15]	940464	×	100.57
Gao *et al.* [36]	51408	0.55	70.02
Itier *et al.* [52]	54289	1.00	127.30
Méthode proposée avec $\Delta = 1.10^{-4}$	830304	0.27	127.24

Nous proposons d'évaluer les résultats sur toute la base de maillages, afin d'analyser les déformations et la qualité du message extrait. Nous fixons les paramètres de l'algorithme à $\Delta = 10^{-4}$ et $\gamma = \frac{1}{10}$, ces paramètres sont expérimentalement choisis pour être le meilleur compromis entre la distorsion et le nombre d'erreurs. Le Tableau 6.10 présente les résultats pour la méthode CAS, la méthode uniforme produit des résultats comparables les variations dépendant de la forme des maillages. Nous utilisons deux métriques pour mesurer les distorsions du maillage la métrique $PSNR_1$ [15] et la métrique $MSDM2$ [71]. Afin d'évaluer la qualité du message extrait, nous proposons d'utiliser le BER pour comparer le flux de bits. Comme nous avons choisi d'insérer une image dans chaque maillage, nous évaluons également la qualité des images extraites avec le PSNR. Évidemment, la taille de l'image *i.e.* la taille du maillage, influence les résultats,

Tableau 6.10: Résultats de la méthode SAC.

Maillage	# de sommets	Charge utile (en pixels)	Distorsion du maillage		Qualité du message	
			$PSNR_1$ (dB)	MSDM2 ($\times 10^{-3}$)	PSNR (dB)	BER ($\times 10^{-3}$)
alien	7401	7056	123.219	3.75	48.03	0.164
Armadillo	172974	172225	134.44	1.27	23.49	2.632
baby	5075	4761	123.375	0.95	$+\infty$	0
bitorus	3000	2704	116.862	0.55	31.34	4.024
blade2	24738	24336	123.375	3.52	48.6	0.067
boy	8441	8100	123.232	0.30	39.07	0.489
bunny	34834	34225	127.236	0.41	52.61	0.042
cad	1426	1156	117.206	1.33	28.38	5.121
casting	5096	4761	118.308	3.48	32.77	2.411
chair1	12326	11881	125.168	4.54	5.49	457.288
chair2	13463	12996	124.716	4.35	2.39	466.227
cow	2904	2601	118.587	25.80	41.65	0.461
crank	50004	49729	127.694	2.14	39.29	0.147
dinopet	4500	4225	118.968	1.88	35.6	1.486
dragon	50000	49729	128.618	1.84	55.07	0.031
eagle	1000	729	112.824	1.32	$+\infty$	0
hand	36619	36100	125.463	1.72	36.11	0.807
hand1	26000	25600	126.428	2.52	30.02	2.913
horse	112642	112225	134.965	2.09	48.68	0.047
horse1	20000	19600	126.58	0.69	45.37	0.149
horse2	2450	2116	117.893	0.53	33.10	2.098
L9305	31088	30625	127.679	2.71	40.44	0.438
rabbit	70658	70225	131.27	0.41	46.34	0.103
ShoeS	45002	44521	128.954	1.80	47.12	0.062
ShoeS2	5002	4624	120.147	2.20	$+\infty$	0
table1	10082	9801	123.722	6.70	8.67	417.223
table2	13579	13225	123.594	7.72	28.45	3.515
US3266	199093	198025	136.006	2.15	42.42	0.334
US3266-2	50002	49729	128.927	2.89	45.81	0.134
us5105c-r	83698	82944	132.52	5.95	38.43	0.705
venus	100759	100489	130.454	1.56	44.02	0.189

mais cette métrique permet de valider la qualité de l'image. Dans ces expériences, nous proposons d'insérer les distributions du message dans les 256 premiers sommets, dans le but de disposer d'une méthode d'IDC totalement aveugle.

Dans le Tableau 6.10, qui présente les résultats avec la méthode CAS, les distorsions du maillages sont très faibles comme constaté sur les premiers exemples Fig. 6.27 et Fig. 6.28. Les distorsions en terme de MSDM2, correspondent à des distorsions imperceptibles pour le système visuel humain. Nous notons que trois maillages (*chair1, chair2, table1*), ne permettent pas une bonne extraction du message. Ces maillages sont ceux qui possèdent la plus petite variance du poids de leurs arêtes, c'est-à-dire qu'elles ont

une longueur régulière. La majorité de leurs sommets se trouve à des distances similaires et par conséquent l'étape de contrôle présentée dans la Section 6.2, interdit les déplacements. En revanche, nous soulignons que chaque sommet qui n'est pas déplacé, augmente la qualité du maillage. Cependant, en général, la méthode proposée a un bon compromis entre la fidélité, la capacité et la qualité du message extrait.

Tableau 6.11: Moyenne des résultats sur la méthode CAS et la méthode uniforme.

Méthode	Capacité (bps)	Distorsion du maillage		Qualité du message extrait	
		$PSNR_1$ (dB)	MSDM2 ($\times 10^{-3}$)	PSNR (dB)	BER ($\times 10^{-3}$)
CAS	22.89	125.46	0.00320	36.15	44.17
Uniforme	22.89	126.52	0.00301	38.39	40.38

Le Tableau 6.11, présente les résultats moyens pour la méthode CAS et la méthode uniforme. Les résultats obtenus avec la méthode uniforme sont très proches des résultats obtenus la méthode CAS. Pour la méthode uniforme, nous n'avons pas besoin d'insérer la distribution. Cependant pour une comparaison entre les deux méthodes nous choisissons d'insérer la même charge utile, c'est-à-dire des images de même taille. Comme la distribution des sommets n'est pas insérée dans la méthode uniforme, il y a donc moins de sommets déplacés, et le maillage est généralement moins déformé par cette méthode. Les différences dépendent également de la forme et de la variance des poids des arêtes de chaque maillage. En outre, nous pouvons voir que la charge utile moyenne obtenue sur les résultats expérimentaux est de 22.89 bits par sommet, ces résultats sont proches de la capacité optimale théorique de 24 bits par sommet. De plus, le nombre de bits de l'image ne correspond pas exactement à la capacité exacte disponible puisque nous souhaitons insérer une image. Pour les autres applications, telles que l'intégration d'un message binaire, la charge utile est plus proche de la capacité théorique, à l'exception des pertes du message en raison de sommets non-déplacés.

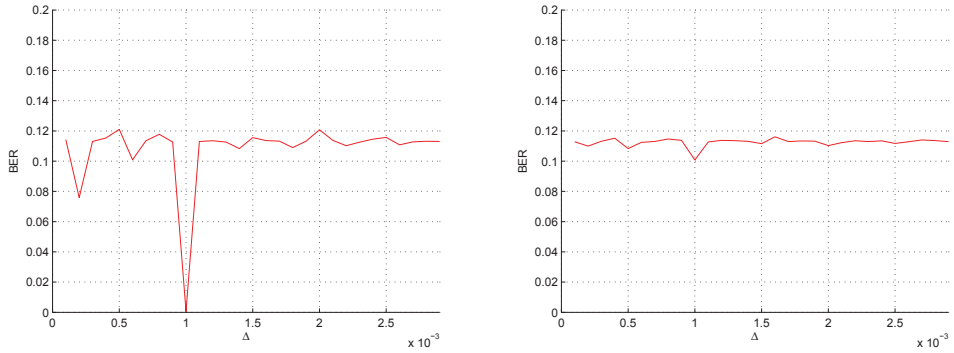
6.4.6 Analyse de la sécurité

Dans cette partie, nous sommes intéressés par les aspects sécurité de la méthode. En effet, le message inséré doit être accessible aux seuls utilisateurs autorisés. Nous considérons trois cas énumérés par par Perez-Freire et Perez-Gonzalez [94] : les attaques de messages connus KMA (Known Message Attack), les attaques à message constant CMA (Constant Message Attack) et les attaques seulement sur le support marqué WOA (Watermarked Only Attack). Le cas le plus étudié est le WOA, en effet il est le plus fréquent.

Pour le cas WOA, le plus simple est de faire une attaque par force brute en testant tous les sommets du maillage comme sommet de départ v_0 du chemin. En supposant

que tous les autres paramètres sont connus, l'attaque réussirait, pour un coût en temps en fonction du nombre de sommets. Le cas KMA a plus de chance de réussir parce que nous ne pouvons prédire la position d'un sommet qu'en connaissant le message. Cela signifie que pour chaque partie du message nous pouvons prédire où les coordonnées du sommet ont été déplacés dans leurs intervalles Δ. Le cas CMA est plus difficile. Cette attaque est inutile si nous supposons que seulement le premier sommet est secret, puisque pour extraire le message, un adversaire peut utiliser une attaque de type WOA. Dans l'autre cas, si Δ est secret, un adversaire peut l'estimer si il connait le nombre de sous-intervalles, en trouvant la valeur qui conduit à la propriété suivante : chaque coordonnée de chaque sommet du maillage, correspondant au même message, aura la même position dans son intervalle.

En considérant que le cas WOA est le plus fréquent, et que la méthode respecte le principe de Kerckhoffs, le but de l'attaquant est d'estimer la position du sommet initial, ainsi que la valeur du pas de quantification Δ. Nous supposons que la capacité est connue, c'est à dire le nombre de subdivisions. En utilisant une recherche exhaustive, un attaquant peut trouver tous les chemins hamiltoniens possibles sur le graphe des sommets. Ensuite, pour chaque chemin, il peut estimer Δ. Bien que nous pensons que le système est sûr dans un sens cryptographique, en raison de la complexité, nous proposons d'analyser les attaques d'estimation de Δ. En supposant que le chemin est connu, nous essayons d'estimer la valeur de Δ pour les méthodes uniforme et CAS. Les probabilités ne sont pas insérées dans le but de maximiser la sécurité. Un message est inséré dans le maillage de l'objet 3D "Dinopet", avec $\Delta = 10^{-3}$. La Fig. 6.30, illustre le taux d'erreurs BER du message extrait en fonction de l'estimation de Δ. En commençant par 10^{-4}, la valeur de Δ est incrémentée de 10^{-4} jusqu'à 3.10^{-3}. La distribution de probabilité est considérée comme uniforme. En effet, on ne peut pas estimer la distribution du message caché.



(a) (b)

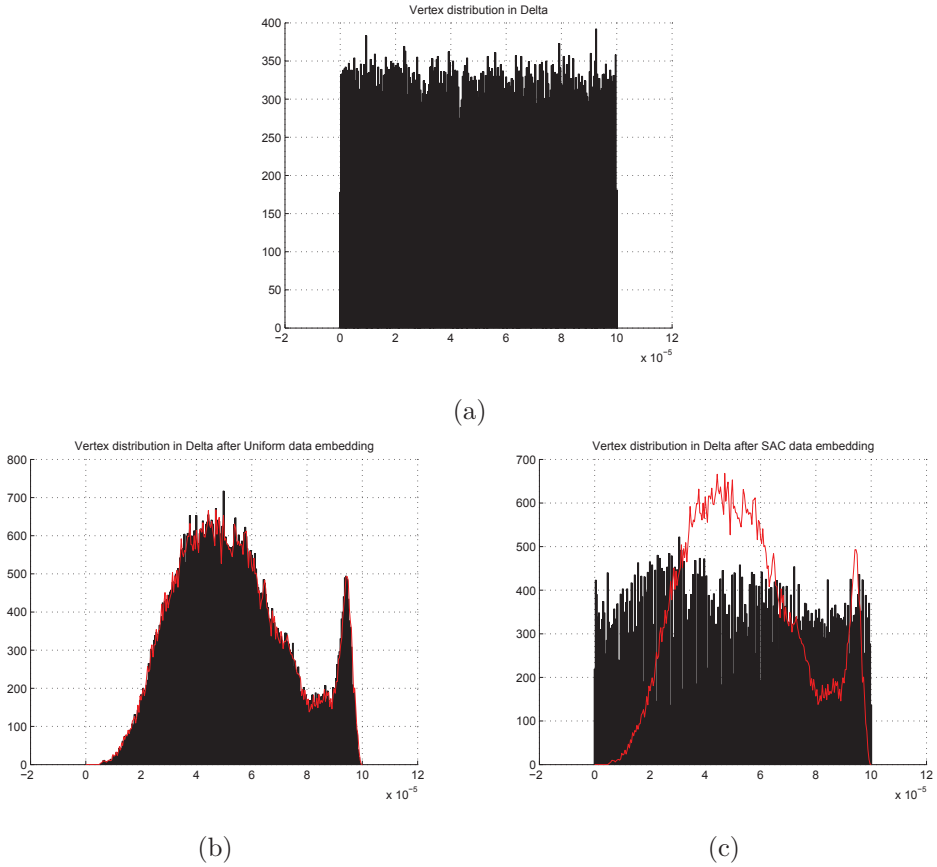
FIGURE 6.30: Taux d'erreurs BER en fonction de l'estimation de Δ : a) méthode uniforme, b) méthode CAS.

Dans la Fig. 6.30.a, nous pouvons voir que pour extraire un message inséré avec une méthode uniforme, nous devons faire une estimation exacte de Δ, afin de récupérer le message correct. Contrairement à la méthode CAS, dans la Fig. 6.30.b, le message ne peut pas être extrait exactement, même avec la valeur correcte de Δ. Ce résultat est dû au secret de la distribution du message. Nous notons qu'il existe un pic correspondant à la valeur de Δ utilisée pour l'insertion. Néanmoins, un attaquant a encore à estimer la distribution de probabilité du message si il veut l'extraire. Nous pensons que la somme des difficultés rencontrées par un attaquant, rend le problème complexe, et enfin, il est beaucoup plus facile de rendre inaccessible le message par désynchronisation ou ajout de bruit géométrique, que d'accéder au message.

En outre, la stéganalyse des méthodes d'IDC 3D, est récente ou conçue pour des méthodes spécifiques. Pour étudier l'effet de la méthode proposée sur la distribution de la position des sommets, nous proposons d'analyser comment le message affecte le déplacement des sommets dans l'intervalle Δ. Pour illustrer notre exemple, nous appliquons la méthode sur le maillage qui est une forme de chaussure ("us5105c-r") possédant 83698 sommets, et présenté dans la Fig. 6.31. Nous fixons les paramètres à $\Delta = 10^{-4}$ et l'image Mandrill est le message à cacher. Nous calculons la nouvelle position du sommet de façon aléatoire avec l'équation 6.19.

FIGURE 6.31: Forme de chaussure avec 83698 sommets.

La Fig. 6.32. illustre l'histogramme de la position d'une coordonnée des sommets dans leur intervalle Δ. Premièrement, nous constatons dans la Fig. 6.32.a que la valeur initiale des coordonnées de chaque sommet est uniformément distribué dans l'intervalle des valeurs possibles. Dans la Fig. 6.32.b et la Fig. 6.32.c nous présentons la même distribution après l'insertion dans les cas (b) de divisions uniformes, (c) de divisions CAS. En rouge, nous avons ajouté l'histogramme normalisé de la distribution du message à insérer sur cette coordonnée. Cette distribution correspond ici aux valeurs de la première composante de couleur de l'image, *i.e.* le canal rouge. Nous constatons dans la Fig. 6.32.b, que la distribution des valeurs des coordonnées des sommets correspond à la distribution des valeurs du message pour le procédé d'insertion uniforme. Alors que, la distribution des valeurs de coordonnées du sommet semble plus uniforme et est plus sécurisé comme illustré dans la Fig. 6.32.c.



(a)

(b) (c)

FIGURE 6.32: a) Distribution originale des valeurs ρ_i des sommets dans Δ, b) distribution des valeurs ρ_i des sommets dans Δ après l'insertion uniforme, c) distribution des valeurs ρ_i des sommets dans Δ après l'insertion CAS.

Les résultats de la méthode CAS en terme de distribution des positions des sommets et la difficulté d'estimer l'intervalle est une perspective intéressante pour une méthode de stéganographie indétectable.

6.5 Conclusion

Dans ce chapitre, nous nous sommes intéressés à développer des méthodes d'IDC haute capacité. Nous avons utilisé la géométrie du maillage, ce qui permet une plus grande capacité. Dans le but d'utiliser la position dans l'espace de tous les sommets comme support du message, nous avons présenté une méthode de synchronisation. Celle-ci est basée uniquement sur la géométrie du maillage qui ordonne les sommets en construisant un chemin hamiltonien. Cette méthode a été analysée en détail et nous montrons comment conserver cet ordre entre deux réalisations. Nous avons proposé deux méthodes d'IDC haute capacité utilisant cet ordonnancement. Dans ces méthodes, l'insertion se fait conjointement avec la synchronisation. L'insertion se fait en déplaçant un sommet par rapport à son prédécesseur dans le chemin. En considérant les arêtes du

chemin comme support du message, la méthode peut déplacer $n-1$ sommets, où n est le nombre de sommets du maillage.

Dans la première méthode, pour les composantes sphériques v_ρ, v_θ et v_ϕ de chaque sommet support, l'algorithme divise les valeurs en intervalles à l'aide d'un pas de quantification. Chaque intervalle correspond alternativement à un bit 0 ou 1. Pour insérer le message, les valeurs des coordonnées sont assignées dans l'intervalle correspondant. Dans ce cas, la méthode atteint une capacité proche de 3 bits par sommet. La méthode est invisible pour un pas Δ faible, et permet un contrôle des distorsions. En outre, la méthode permet d'obtenir une bonne sécurité puisque la charge utile ne peut pas être extraite sans la connaissance de la clé secrète. La méthode proposée possède des propriétés intéressantes, mais nous avons des pistes possibles pour nos travaux futurs. La première amélioration consisterait à calculer l'étape Δ de façon fiable et automatiquement, en fonction du maillage. La normalisation proposée dans nos premiers résultats expérimentaux ne permet pas une comparaison parfaite entre des maillages différents. Nous avons donc normalisé les maillages de façon à ce que la longueur moyenne d'une arête soit égale à 1 dans nos résultats suivants. Nous pensons qu'une étude approfondie de l'impact des normalisations sur les résultats serait productive. Nous voulons également améliorer les performances en terme de complexité en temps.

Dans notre seconde méthode, nous avons fortement augmenté la capacité. Nous définissons un intervalle de déplacement, qui est subdivisé pour atteindre une capacité d'environ 24 bps. Notre méthode est conçue pour insérer par exemple, une image couleur dans le maillage d'un objet 3D, comme une texture ou un message secret. Cependant, la méthode peut être utilisée pour insérer tout type de données. Nous proposons deux méthodes, une prenant en compte la distribution du message, l'autre la considérant comme uniforme. Nous pensons que ces méthodes sont plus sécurisées, dans le sens où le message n'est pas accessible sans la clé puisque le problème est complexe en fonction du nombre de sommets et des paramètres d'insertion à estimer. Nous présentons comment utiliser la distribution du message pour obtenir une méthode plus indétectable. Cependant une stéganalyse pourrait permettre de valider si la méthode peut être utilisée comme méthode de stéganographie haute capacité. Une autre amélioration consisterait à rendre le procédé moins déterministe. Le choix du sommet suivant dans la construction du chemin peut dépendre de la clé secrète pour générer un chemin qui n'est pas basé sur le plus proche voisin.

Ces travaux on fait l'objet de deux publications dans des conférences internationales, la première méthode dans la conférence internationale SPIE EI 2015 [52] et la seconde dans la conférence internationale IEEE ICIP 2015 [53]. Une version étendue est en cours de révision dans la revue internationale IEEE Transaction on Multimedia [56].

Chapitre 7

Synchronisation de données cachées haute capacité avec sécurisation de l'étape de synchronisation

7.1 Introduction

Les chemins hamiltoniens sont intéressants pour traverser un nuage de points, cependant sa sécurité peut être insuffisante. En effet, si nous pouvons trouver le point d'entrée, il est facile de reconstruire tout le parcours. Dans ce chapitre nous proposons une nouvelle méthode de synchronisation basée sur des sauts aléatoires sur la surface du maillage de l'objet 3D ce qui permet d'augmenter la sécurité du système. En effet, l'objectif est de disposer d'une méthode d'IDC haute capacité et sécurisée. L'étape d'insertion s'appuie sur la méthode d'insertion présentée dans la Section 6.4. Cette nouvelle synchronisation nous a conduit à redéfinir le problème de causalité et nous apportons une nouvelle solution qui évite une perte de bits du message. Nous nous sommes aussi focalisés à développer une méthode d'IDC permettant de réduire les cas engendrant les problèmes de désynchronisation. Cette méthode permet de minimiser les distorsions sur la surface du maillage et est efficace afin de réduire le nombre de configurations problématiques appelées ambiguïtés. Cette méthode peut être utilisée dans un premier temps, avant de choisir une stratégie permettant la conservation de l'ordre défini, comme le non déplacement proposé dans le Chapitre 6.

La nouvelle méthode d'IDC est introduite dans la Section 7.2. Nous analysons et proposons une solution au problème de causalité et nous discutons de sa sécurité. Des améliorations de l'ensemble des méthodes proposées permettant la diminution du nombre d'ambiguïtés et la réduction des distorsions sont présentées dans la Section 7.3.

7.2 IDC sécurisée basée sur 3 classes de sommets

Dans cette section, nous décrivons les principales étapes de la nouvelle méthode d'IDC proposée qui sont illustrées dans la Fig. 7.1.

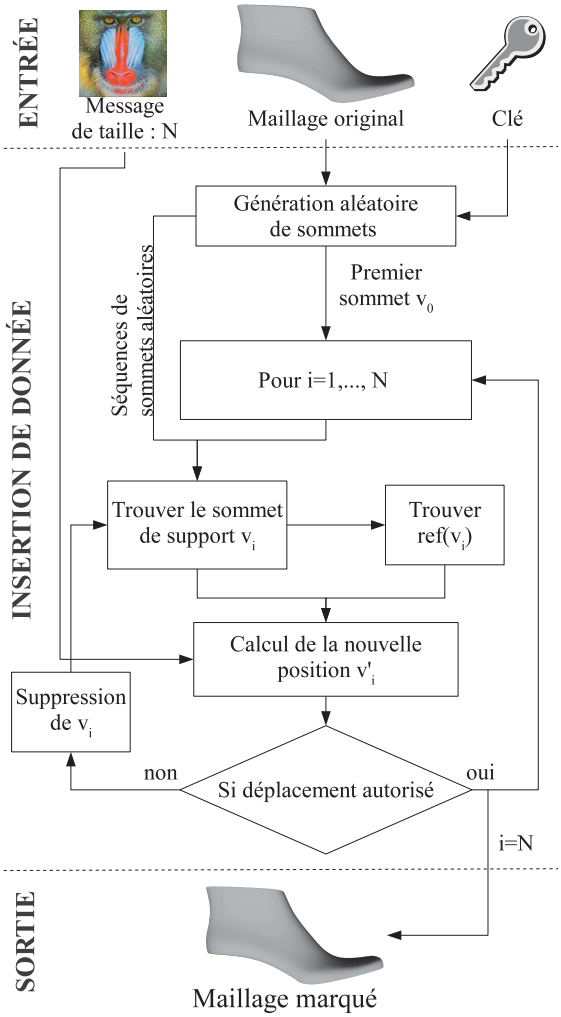


FIGURE 7.1: Schéma de la méthode de dissimulation de données proposée.

La synchronisation consiste à construire un chemin sur les sommets pour les ordonner comme décrit dans le Chapitre 6. Nous proposons une synchronisation basée sur des sauts aléatoires sur le maillage en fonction d'une clé secrète. Les sommets supports de l'information sont alors sélectionnés suivant cet ordre afin d'assurer la sécurité et

pour diffuser le message sur l'ensemble du maillage. L'insertion est effectuée par le déplacement d'un sommet v_i en fonction de la position de son sommet de référence, qui est défini comme son sommet le plus proche. La méthode d'insertion est basée sur les méthodes présentées dans le Chapitre 6, en fixant la capacité à 24 bits par sommets pour avoir une méthode haute capacité. Le message est inséré partie par partie, lors de la synchronisation de façon conjointe. Pour éviter les problèmes de causalité lors de l'insertion, nous proposons une étape de vérification. Le processus de dissimulation de données ne dépend pas de la connectivité des sommets, mais seulement de leurs positions dans l'espace. La Fig. 7.1, donne un aperçu de la méthode proposée pour l'insertion d'un message, qui peut être une image par exemple. La boucle correspond au processus itératif qui inclut à la fois la synchronisation présentée dans la Section 7.2.1 et l'étape consistant à insérer la charge utile. Pour chaque pixel d'une image couleur *i.e.* 24 bits, la méthode proposée sélectionne un sommet de support en fonction d'un sommet généré de façon pseudo aléatoire. Puis l'insertion, présentée dans la Section 7.2.2 se fait en déplaçant le sommet support relativement à son sommet de référence. Ensuite, si le déplacement est autorisé par la vérification présentée dans la Section 7.2.3, l'algorithme passe à l'étape suivante d'insertion. Au contraire, si le déplacement est interdit, pour conserver l'ordonnancement défini, nous proposons de supprimer le sommet de support courant. Cette étape est expliquée dans la Section 7.2.4. Enfin, lorsque l'intégralité du message est insérée, les trous éventuels dans la surface de l'objet 3D, causée par la suppression des sommets, sont fermés par remaillage. Nous expliquons l'intérêt de la méthode pour la sécurité dans la Section 7.2.5. Finalement, la Section 7.2.6 propose des expérimentations.

7.2.1 Synchronisation basée sur un chemin de sauts aléatoires

Le but de cette nouvelle méthode de synchronisation est de produire un chemin sur les sommets grâce à des sauts aléatoire sur la surface d'un objet. Les sauts aléatoires permettent de rendre le choix du prochain sommet à ajouter au chemin non prévisible au contraire de la sélection par plus proche voisin. Ils permettent également de diffuser le message et donc les distorsions sur le maillage, en particulier si le message n'occupe pas toute la capacité disponible. Ce qui est préférable que de condenser toute la marque dans une zone. L'idée est de générer une séquence aléatoire de sommets dans l'espace du volume englobant de l'objet. Cette séquence est utilisée pour obtenir le sommet suivant lors de la construction du chemin. Le chemin est créé itérativement pour définir un ordre pour l'insertion de l'information à l'aide d'une clé secrète. En fait, la synchronisation permet d'ordonner des paires de sommets dans le maillage. Ce couple de sommets sert de support d'information, comme présenté et défini dans la Section 7.2.2, un des sommets

est déplacé, l'autre non et est appelé sommet de référence. La position du premier sommet du chemin est choisi en utilisant une clé spécifique à l'utilisateur et le sommet choisi est considéré comme un sommet de référence. La Fig. 7.2 représente une étape d'ordonnancement des sommets, le sommet v_i et sa référence $ref(v_i)$ est trouvée par rapport à son père dans le chemin $f(v_i)$ et du sommet aléatoire $rd(v_i)$ généré avec le rayon r_i.

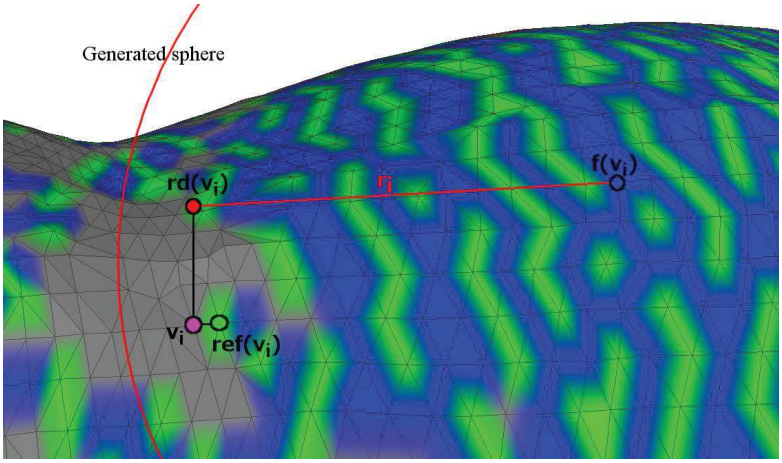


FIGURE 7.2: Vue d'ensemble de la construction du chemin. Le sommet précédent $f(v_i)$ est en bleu. Le sommet aléatoire généré $rd(v_i)$, centré sur $f(v_i)$ et de rayon aléatoire r_i est en rouge. Le sommet courant v_i est en magenta et son sommet de référence $ref(v_i)$ en vert. Avec en bleu les sommets supports, en vert les références et en gris ceux qui ne sont pas utilisés.

Tant qu'il reste des sommets à ordonner, l'algorithme génère une sphère centrée sur le sommet de référence et de rayon aléatoire. C'est ce rayon qui permet de définir la longueur moyenne d'un saut. Sur cette sphère, un point aléatoire $rd(v)$ est calculé et sert pour trouver le sommet suivant sur le maillage. Le sommet suivant v correspond au plus proche sommet non-utilisé du maillage à partir du point généré aléatoirement, et son plus proche sommet est son sommet de référence. L'emplacement du point sur la sphère est calculé avec la méthode de Marsaglia [85], illustrée dans la Fig. 7.3.

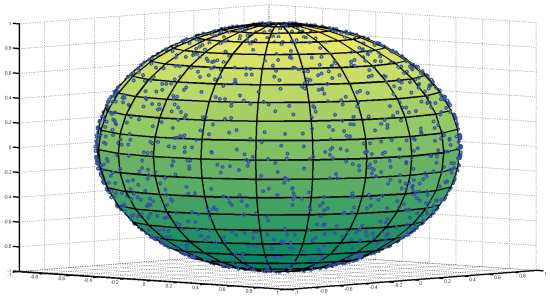


FIGURE 7.3: Distribution aléatoire de 1000 points sur la sphère unitaire calculée avec la méthode de Marsaglia [85]

Cette méthode génère des points répartis uniformément sur une sphère à l'aide de valeurs aléatoires obtenues en utilisant une clé secrète générée aléatoirement. Nous utilisons également une variable aléatoire pour déterminer le rayon de chaque sphère. Il dépend de la distance la plus longue sur le maillage D_{max}, et est calculé comme :

$$r_i = k \cdot D_{max}, k \in [0, 1]. \tag{7.1}$$

Soit deux valeurs uniformes g_1 et g_2 sur l'intervalle $[-r_i, r_i]$, pour une valeur $s = g_1^2 + g_2^2$, tel que $s < r_i$, le sommet rd_i sur la sphère de centre v et de rayon r_i, est calculé :

$$rd_i = \begin{bmatrix} 2g_1\sqrt{r_i - s} + v_x \\ 2g_2\sqrt{r_i - s} + v_y \\ r_i - 2s + v_z \end{bmatrix}. \tag{7.2}$$

En utilisant une petite valeur de k, nous pourrions obtenir une trajectoire prévisible en choisissant le sommet le plus proche à chaque étape. Ce qui revient à faire un chemin qui ne permet pas de diffuser le message sur le maillage et est moins sécurisé. Au contraire, la synchronisation proposée nous permet de distribuer le message sur l'ensemble du maillage, en faisant des sauts en fonction du rayon de la sphère.

7.2.2 Méthode d'IDC basée sur la synchronisation sécurisée

Dans cette section, nous proposons d'insérer un message dans un objet 3D en utilisant comme support la paire de sommets support/référence définie par la synchronisation. Le but est d'avoir une haute capacité tout en assurant un haut niveau de sécurité. La méthode d'insertion entre deux sommets se base sur celle présentée dans le Chapitre 6. Les coordonnées de chaque sommet sont converties en coordonnées sphériques. Un intervalle de distance Δ est utilisé comme intervalle de déplacement du sommet. Pour insérer un octet par coordonnées, l'intervalle Δ est divisé en 256 sous-intervalles. Ainsi, il est possible d'insérer 24 bits dans une paire de sommets. L'insertion se fait à chaque étape de la construction du chemin, pour chaque paire de sommets ajoutée, nous insérons une partie du message. Ce processus est itératif, afin de ne pas perturber un chemin qui a été construit précédemment. En raison de l'utilisation de sauts, les distances entre un sommet v_i et son prédécesseur $f(v_i)$ dans le chemin peuvent être relativement grandes. Nous proposons d'utiliser un sommet comme référence $ref(v_i)$ pour l'insertion. Le déplacement du sommet support courant v_i se fait pas rapport au sommet de référence. Il est défini comme étant le sommet le plus proche de v_i. Ce sommet est utilisé comme référence pour la conversion en coordonnées sphériques et il est utilisé

pour trouver le nouvel emplacement v_i' de v_i. Pour insérer une valeur $n \in [0, 255]$, dans une des coordonnées du sommet $v_i(c_1, c_2, c_3)$, nous utilisons la fonction suivante :

$$c_1' = \left\lfloor \frac{c_1}{\Delta} \right\rfloor \Delta + n\frac{\Delta}{256}. \tag{7.3}$$

Nous définissons trois classes différentes pour caractériser les sommets du maillage d'un objet 3D : les sommets supports qui sont déplacés pour insérer des données, les sommets références et les sommets qui ne sont pas utilisés. Un sommet de référence peut être utilisé par plusieurs sommets de support. L'insertion est jointe à la synchronisation avec un fonctionnement itératif. Le sommet courant v_i est déplacé par rapport à son sommet référence, $ref(v_i)$. L'utilisation de sommets de référence nous permet d'avoir une plus grande sécurité ainsi qu'une plus faible distorsion puisque les sommets ne sont pas déplacés. La capacité de dissimulation de données n'est pas trop affectée car la même référence de sommet peut être utilisée pour le déplacement de plusieurs sommets. En effet, selon nos résultats expérimentaux, nous pouvons utiliser la même référence pour trois sommets de support en moyenne. Nous pouvons donc dire qu'en moyenne un maillage à une capacité de $\frac{2}{3}24(|V| - 1)$ bits. De plus, la méthode proposée permet l'insertion des données dans des représentations d'objets denses en sommets, ce qui augmente de manière significative la charge utile qui peut être insérée dans ces maillages.

Pour extraire le message d'un maillage marqué, il faut connaître la clé secrète afin d'en extraire le message correctement puisque la méthode suit le principe de Kerckhoffs. L'ordre des sommets est récupéré grâce à la clé secrète. Ensuite, pour chaque paire de sommets support et référence et avec connaissance du pas Δ nous pouvons extraire l'information :

$$x = 256(c_j - \left\lfloor \frac{c_j}{\Delta} \right\rfloor \Delta). \tag{7.4}$$

Lorsque tout le message est extrait, le message est reconstitué en concaténant les parties en suivant l'ordre du chemin.

7.2.3 Analyse du problème de causalité

Lorsque l'on considère une insertion jointe à la synchronisation par sauts aléatoires, il faut vérifier que l'ordre de parcours et les classes des sommets sont maintenus durant tout le processus d'insertion. En effet, le déplacement des sommets, peut influer sur l'ordre des sommets précédemment ajoutés au chemin ou alors changer la classe d'un sommet. C'est un problème crucial, car dans de telles situations, nous ne sommes pas en

mesure d'effectuer la même traversée du maillage et donc de récupérer les informations insérées. Nous présentons alors les situations pouvant se présenter à l'étape d'insertion, nous devons vérifier deux cas :

1. Il faut conserver le sommet v_i, et sa référence à l'étape i.
2. Il faut maintenir le sous-chemin $\mathbf{P_i}$.

Ces conditions se basent sur les même idées que celles présentées dans le Chapitre 6. Cependant, l'utilisation des classes de sommets ainsi que les sauts changent les contraintes à vérifier. Dans un premier temps il faut que le sommet courant v_i, ne soit pas déplacé trop loin ni de sa valeur aléatoire correspondante rd_{v_i}, ni de sa référence $ref(v_i)$. Il faut également nous assurer que le sommet courant v_i ne soit pas choisi à une étape précédente lors d'un second parcours. C'est la raison pour laquelle les sommets supports des couples de sommets dans le sous-chemin sont étiquetés comme des références après avoir été déplacés. Notons qu'un problème au niveau du sous-chemin a moins de chance de se produire puisque les sommets sont éloignés par le saut. Dans un souci de clarté nous différencions trois cas et fixons des règles :

— Pour la vérification du sommet courant v_i par rapport au sous-chemin $\mathbf{P_i}$, il faut que $\forall v_k \in V_i$:

$$\|v_k, rd_k\|_2 < \|rd_k, v'_i\|_2, \tag{7.5}$$

et que :

$$\|v_k, ref(v_k)\|_2 < \|v'_i, v_k\|_2, \tag{7.6}$$

c'est-à-dire qu'il faut que le sommet ne soit pas choisi comme sommet support ou référence avant l'étape i.

— Pour la vérification du sommet courant v_i par rapport aux sommets non-utilisés, il faut que $\forall v_j \in V_n \backslash V_i,\ j \neq i+1$:

$$\|v_i, rd_i\|_2 < \|rd_i, v_j)\|_2,\ v_j \neq v_i, \tag{7.7}$$

$$\|v_i, ref(v_i)\|_2 < \|vi, v_j)\|_2,\ v_j \neq ref(v_i), \tag{7.8}$$

c'est-à-dire qu'il faut que le sommet courant soit choisi à l'étape i.

— Pour la vérification du couple support/référence courant, il faut que :

$$\|rd_i, v'_i\|_2 < \|rd_i, ref(v_i)\|_2, \tag{7.9}$$

c'est-à-dire le cas pour lequel il n'y a pas d'inversion de classification entre le sommet support et son sommet de référence.

Nous avons défini formellement les relations et les contraintes dépendantes de la méthode d'ordonnancement. Nous illustrons, sur un exemple de configuration ces relations de distances dans la Fig. 7.4. Cette figure est une projection en 2D illustrant les conditions à vérifier, afin de respecter les équations précédentes. Dans cette figure, nous

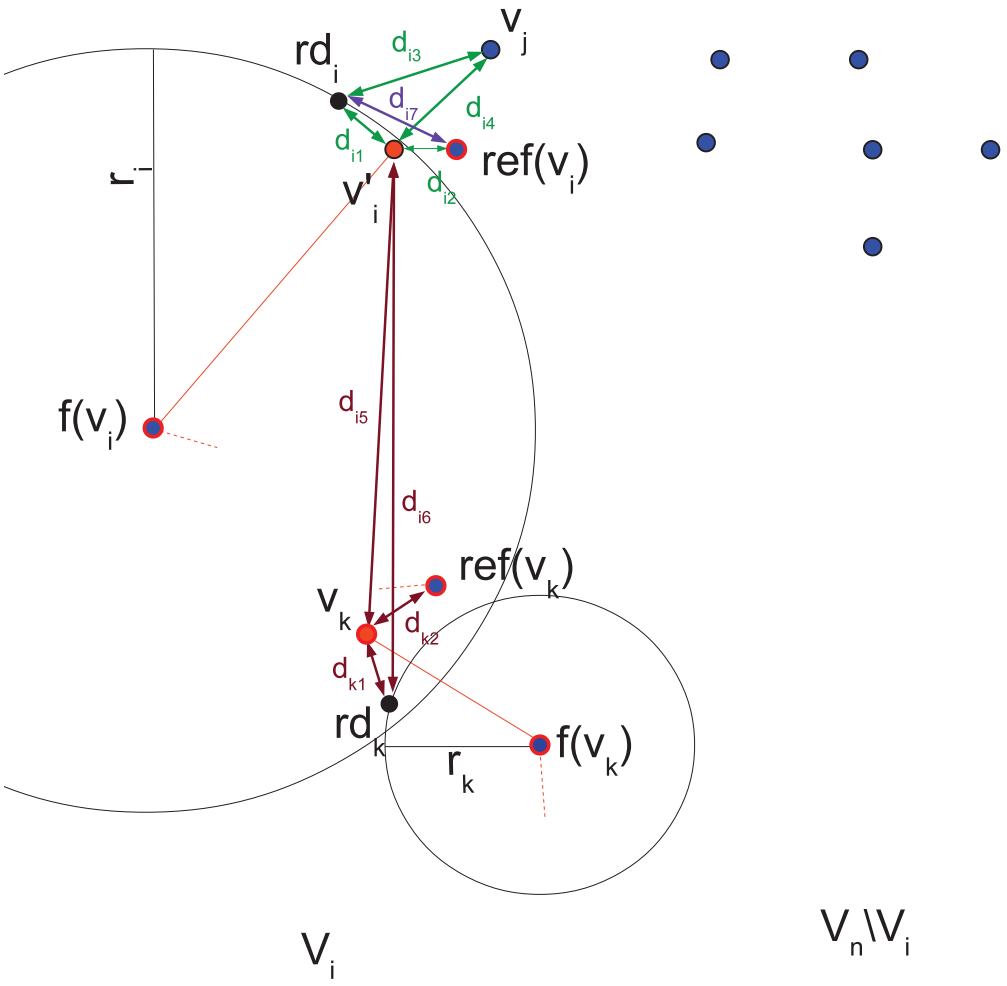


FIGURE 7.4: Schéma des distances.

présentons en bleu les sommets non-utilisées, avec un bord rouge les sommets références, et en orange les sommets supports du chemin. Nous listons les différents cas à respecter pour conserver l'ordonnancement.

1. La paire courante : $d_{i1} < d_{i7}$.
2. À l'étape i : $d_{i1} < d_{i3}$, $d_{i2} < d_{i4}$.
3. Le sous-chemin $\mathbf{P_i}$: $d_{k1} < d_{i6}$, $d_{k2} < d_{i5}$.

Les distances présentées dans la Fig. 7.4 sont colorées en fonction des trois cas proposés, violet pour la paire courante (cas 1), vert pour l'étape i (cas 2) et bordeau pour le sous-chemin (cas 3).

Le chemin de sommets choisis doit être le même pour les étapes d'insertion et d'extraction de la procédure de dissimulation de données. Nous avons défini les conditions à vérifier à chaque étape de déplacement d'un sommet. Si ces conditions ne sont pas respectées, nous produisons un problème de causalité puisque l'insertion modifie la synchronisation. En effet, un petit nombre d'erreurs peut changer de manière significative le chemin sur les sommets et donc la synchronisation. Au contraire lorsque les conditions sont vérifiées, nous pouvons affirmer que l'ordre de parcours du maillage reste inchangé lorsqu'il est recalculé avec les mêmes paramètres. Les situations qui conduisent à un problème de causalité, dépendent de la forme du modèle 3D, de la densité des sommets et de la régularité du maillage. Il faut donc envisager une stratégie pour éviter ou résoudre ces problèmes.

7.2.4 Résolution du problème de causalité

Nous avons vu dans le Chapitre 6 que lorsqu'un déplacement n'est pas autorisé, nous pouvons simplement ne pas déplacer le sommet au coût d'une perte d'une partie du message. Ici, nous proposons une autre stratégie pour éviter de perdre la synchronisation. Nous proposons de supprimer le sommet support ne pouvant être déplacé et de combler le trou résultant. En règle générale, nous pouvons considérer dans les maillages denses et comportant de nombreux sommets, que le voisinage des sommets est une surface plane. Donc la suppression d'un sommet ne crée pas de distorsions importantes. Ensuite, l'ouverture laissée dans le maillage est refermée en la divisant en faces triangulaires simples. Afin de minimiser les distorsions de la surface de l'objet, les arêtes créées sont placées aussi près que possible de l'emplacement du sommet supprimé. Cette stratégie est efficace et simple à mettre en œuvre pour tout type de maillage. Néanmoins elle est particulièrement adaptée dans le cas de maillages triangulaires 2-variétés. En effet, le remaillage est simple et l'orientation des facettes permet d'obtenir des normales dans la bonne direction pour le rendu. Dans le cas de formats sans définitions explicites des normales, il suffit d'utiliser une orientation des facettes compatible avec les facettes existantes, ce qui est simple puisque ce type de maillage est orientable. Pour combler l'ouverture dans le maillage, nous proposons l'utilisation d'un algorithme récursif qui divise en deux parties le trou par l'ajout d'une arête, jusqu'à ce que les parties correspondent à des faces triangulaires. Pour trouver l'arête qui divise en deux le trou, nous recherchons l'arête passant le plus près possible de l'ancienne position du sommet supprimé tout en essayant de conserver des surfaces identiques. Le ratio entre les aires A_1

et A_2 sert à pondérer la distance d_1 entre la droite passant par une nouvelle arête d_e et le sommet supprimé v. Soit $\mathbf{v}_s$ la représentation vectorielle du sommet alors la distance d_1 est calculée comme :

$$d_1(d_e, \mathbf{v}) = \frac{|(\mathbf{v}_s - \mathbf{v}_i)(\mathbf{v_s} - \mathbf{v}_j)|}{|\mathbf{v}_j - \mathbf{v}_i|},\ \mathbf{v}_i, \mathbf{v}_j \in d_e. \tag{7.10}$$

Le ratio des aires est défini comme :

$$r_A = \frac{A_1}{A_2}, A_1 \leq A_2, \tag{7.11}$$

pour que sa valeur soit comprise dans l'intervalle $[0, 1]$. Soit V_t l'ensemble des sommets à la frontière du trou créé par la suppression d'un sommet, alors nous cherchons les sommets de l'arête $e(v_i, v_j),\ i \neq j,\ v_i, v_j \in V_t$ qui minimise la distance pondérée :

$$\underset{v_i, v_j \in V_t}{\operatorname{argmin}} \frac{d_1((v_i, v_j), \mathbf{v})}{r_A}. \tag{7.12}$$

Généralement, les ambiguïtés entre sommets se trouvent dans des zones plutôt régulières ou semi-régulières du maillage dans lesquelles les distances sont similaires et souvent planes. À l'opposé, l'insertion ne pose pas de problème dans les zones des zones plus texturées. La suppression d'un sommet dans les zones planes n'affecte pas la surface, ni le rendu. Par contre la connexité au niveau de la partie du maillage comblée est visible notamment si la zone du maillage est régulière ou semi-régulière.

7.2.5 Analyse de la sécurité

Dans cette section, nous analysons un certain niveau de sécurité de l'approche proposée pour l'IDC haute capacité et sécurisée. Tout d'abord, en se basant sur le principe de Kerckhoff, la sécurité repose sur le secret de la clé. La clé secrète est utilisée comme graine pour la génération aléatoire de l'ordonnancement des sommets. Les clés secrètes assurent qu'un potentiel attaquant n'est pas en mesure de récupérer la séquence de sommets et l'ordre d'insertion. En outre, pour un attaquant cherchant à trouver les paires support/référence, il est difficile de classer les sommets dans une des deux catégories. La complexité augmente si nous choisissons de ne pas utiliser tous les sommets et donc d'affecter des sommets à une troisième classe dite des sommets non-utilisés. Nous supposons différents scénarios d'attaque tels que ceux proposés par Perez-Freire et Perez-Gonzalez [94] : KMA, CMA et WOA décris dans la Section 6.4.6. L'attaquant, en ayant l'objet d'origine, peut tenter de faire une attaque de quantification. Toutefois, afin de trouver le message caché, il faudrait connaître l'ordre des sommets déplacés. Si l'attaquant connait les sommets qui sont déplacés, il peut deviner leurs sommets de référence

en trouvant leur plus proche voisin. Ensuite, il peut faire une attaque de quantification pour trouver la valeur intégrée dans chaque sommet en considérant une insertion uniforme. La difficulté consiste ici, à synchroniser les valeurs extraites. Néanmoins, dans un scénario d'attaque plus pertinente, le WOA qui considère que l'attaquant possède seulement le maillage marqué, nous pensons qu'il est impossible de déterminer où le message est inséré.

7.2.6 Résultats expérimentaux

Les résultats sont produits en utilisant une base de données de maillages 3D qui présentent différentes formes et tailles. Les maillages des objets 3D sont d'abord normalisés, afin de rendre le poids moyen des arêtes égal à 1. Nous fixons le paramètre Δ à 10^{-4} dans l'équation 7.3, qui est expérimentalement, un bon compromis entre l'imperceptible et la charge de données pouvant être insérée dans chaque sommet. Nous utilisons un générateur aléatoire pour l'algorithme de Marsaglia [85]. La Fig. 7.5 illustre la position des sommets aléatoires, représentés par des cercles colorés, autour du maillage de l'objet 3D "Alien". Dans la suite, nous insérons des images en couleur dans l'objet 3D puisque nous pouvons insérer 3 octets dans chaque sommet sélectionné. Comme les images sont en deux dimensions, nous devons intégrer la hauteur et la largeur de l'image dans les deux sommets du chemin, les premiers par exemple ou à une position donnée par la clé.

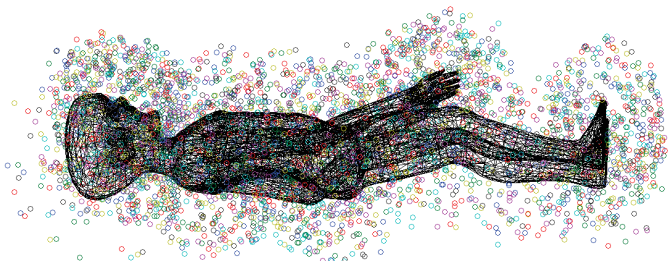


FIGURE 7.5: Positions des sommets aléatoires représentés par des cercles colorés, générés pour le maillage "Alien" possédant 7401 sommets.

Afin de choisir en toute sécurité des sommets supports des données, nous générons une séquence de sphères. Le rayon de ces sphères est choisi comme un nombre généré de façon aléatoire à partir d'une distribution uniforme. Dans la courbe de la Fig. 7.6 nous analysons l'impact de la longueur du rayon sur le nombre de sommets créant des situations d'ambiguïtés et donc qui doivent être supprimés pour conserver l'ordonnancement. La longueur du rayon est choisie en pourcentage $k \in [0.1, 0.9]$ de la longueur maximale dans le maillage D_{max}, selon l'équation 7.1.

Nous évaluons l'influence du rayon des sphères produites sur le nombre de sommets qui sont supprimés par l'algorithme. L'analyse porte sur dix maillages de différentes

formes et tailles. Nous avons standardisé les résultats en calculant la moyenne centrée réduite du pourcentage de sommets supprimés pour chaque maillage. C'est à dire que les résultats de chaque sommet sont obtenus en retranchant la moyenne à chaque donnée et en divisant ces valeurs par l'écart-type de la série. La Fig. 7.6, illustre la moyenne des valeurs normalisées obtenues en fonction du rayon des sphères pour tous les maillages.

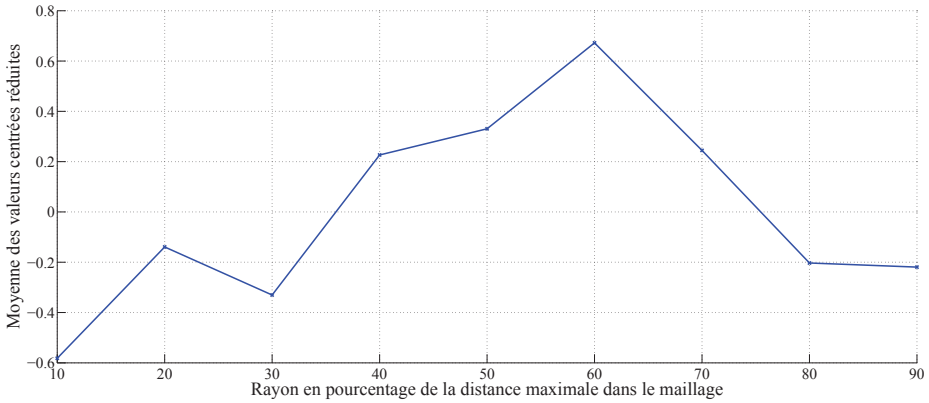


FIGURE 7.6: Moyenne centrée réduite du pourcentage de sommets supprimés en fonction du rayon des sphères, calculée comme le pourcentage de la distance maximale du maillage, D_{max}.

Dans nos hypothèses nous pensions que la courbe devait suivre une loi normale centrée sur 50% de la distance maximale. Nous constatons qu'un pic se trouve à 20%, et que le pic maximal correspond à 60%, nous pensons que ces résultats pourraient être lissés sur un plus grand nombre d'échantillons. D'autre part, nous pouvons voir que l'utilisation d'un petit rayon conduit à peu de sommets supprimés, cependant comme souligné dans la Section 7.2.1 un rayon trop faible revient à faire un chemin basé sur la recherche du plus proche voisin. Selon ces résultats, nous avons constaté qu'un intervalle de rayon $k \in [0.15, 0.35]$ présente un faible nombre d'erreurs et permet au chemin de faire des sauts intéressants dans le maillage. Nous présentons un exemple pour l'insertion de l'image couleur "Mandrill" présentée Fig. 7.7.b dans le maillage de l'objet "Venus", présenté dans la Fig. 7.7.a. Pour nos expériences, nous avons considéré la capacité du maillage comme $c = \frac{24(|V|-1)}{2}$, *i.e* la capacité de la méthode d'insertion pour un sommet sur la moitié des sommets du maillage. Cette approximation permet d'être sûr que la limite n'est pas atteinte pour le maillage. Cependant, expérimentalement il est possible d'utiliser environ les deux tiers des sommets. Dans cet exemple, la capacité est de 11.95 bits par sommet, mais 15637 sommets (environ 15%) ne sont pas utilisés et il y a 0.69 références par sommet en moyenne. Pour atteindre la capacité optimale du maillage, nous pouvons insérer un message tant qu'il y a des sommets supports disponibles.

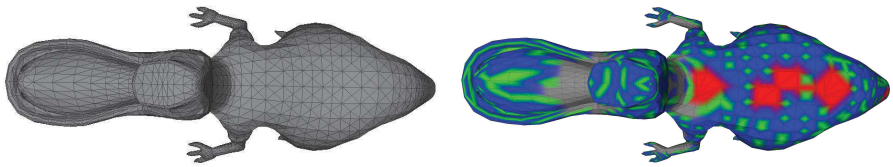
La Fig. 7.8, présente le résultat lors de l'intégration d'un message dans le maillage de l'objet "Dinopet". Dans la Fig. 7.8.a, nous montrons le maillage Dinopet original tandis

(a)

(b)

FIGURE 7.7: a) Maillage original avec 100759 sommets, b) Image insérée 224 × 224 pixels : 1204224 bits.

que l'objet marqué est présenté dans la Fig. 7.8.b. Nous pouvons voir que les sommets supprimés se trouvent principalement dans les zones régulières du maillage. Le maillage est relativement petit (4500 sommets), de sorte que le remplissage des trous, résultant de la suppression des sommets, est parfois visible. De plus, il ne semble pas toujours naturel surtout lorsque les sommets supprimés sont adjacents et créent un trou plus large.

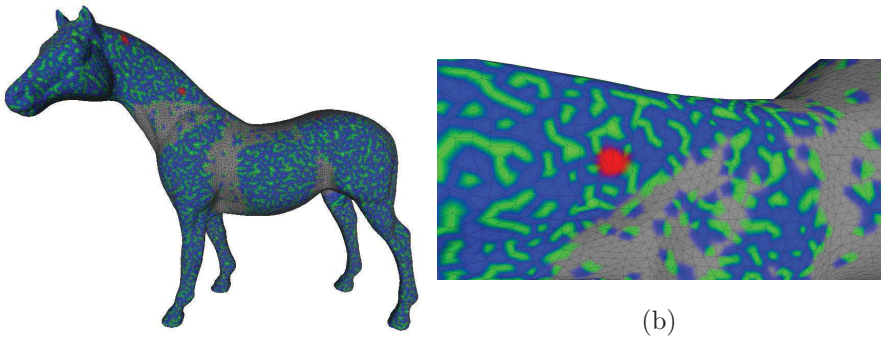


(a) (b)

FIGURE 7.8: a) Maillage original avec 4500 sommets, b) maillage marqué avec 4493 sommets, en bleu : les sommets supports, en vert : les sommets références, en rouge : les trous correspondant aux sommets retirés qui sont remaillés et en gris : les sommets non-utilisés.

Cependant, sur des maillages plus grands et moins réguliers, le remplissage est presque imperceptible. Nous constatons sur le maillage marqué de l'objet "Horse" illustré dans la Fig. 7.9.a, qu'après la fermeture du trou, les distorsions qui en résultent sont imperceptibles. En outre, dans le grossissement sur la surface du maillage présenté dans la Fig. 7.9.b, nous remarquons que le remplissage est lisse et ne fausse pas la surface.

Afin d'évaluer l'algorithme d'IDC proposé, nous utilisons deux métriques présentées dans le Chapitre 3, à savoir la distance de Hausdorff, et la distance MSDM2 [71]. Contrairement à la distance de Hausdorff, la distance MSDM2 est plus en corrélation avec le SVH et nous sert à montrer que malgré une plus grande distance de Hausdorff la méthode reste imperceptible. Le Tableau 7.1 présente les résultats des distorsions produites par la méthode sur plusieurs maillages. Nous n'avons pas pris en compte les maillages avec de grandes zones planes et régulières, parce que la méthode proposée a tendance à supprimer énormément de sommets et pourrait introduire des distorsions dans ces objets.



(a) (b)

FIGURE 7.9: a) Maillage marqué possédant 19995 sommets, b) Gros plan sur la surface du maillage, en bleu : les sommets supports, en vert : les sommets références, en rouge : les trous fermés et en gris : les sommets non-utilisés.

Cependant, nous pensons que les maillages de ce type peuvent être utilisés après avoir subi une opération de simplification.

TABLEAU 7.1: Distortion results.

Modèle	Nombre de sommets	Nombre de sommets supprimés	Distance de Hausdorff $\times 10^{-3}$	MSDM2 $\times 10^{-2}$	Capacité bps
Bitorus	3000	28 (0.930%)	1.581	9.69	11.55
Dinopet	4500	7 (0.160%)	4.659	8.10	11.78
Casting	5096	51 (1.0%)	0.272	14.51	11.77
Horse	20000	9 (0.045%)	1.347	2.59	12
Blade	24738	4 (0.016%)	1.550	2.23	11.95
Bunny	34834	44 (0.126%)	1.796	6.844	11.83
Shoe	45002	12 (0.027%)	0.152	1.28	11.99
Rabbit	70658	47 (0.067%)	0.816	1.07	11.87
Shoe2	83698	149 (0.178%)	0.429	12.77	11.93
Venus	100759	64 (0.064%)	0.800	4.21	11.95

Dans le Tableau 7.1, nous présentons les résultats lors de l'insertion des données pour un ensemble de dix maillages d'objets 3D, possédant une forme et des propriétés différentes. La distance de Hausdorff semble corrélée à la forme de l'objet et à sa taille. Par exemple dans les petits maillages possédant des courbures, comme le maillage de l'objet "Dinopet", la suppression d'un sommet implique l'aplatissement de la surface. Lors de l'insertion des données dans le maillage de l'objet "Casting", nous avons beaucoup de sommets supprimés en raison de la régularité habituellement trouvée dans les objets de type CAO. Il a cependant de faibles distorsions, puisque la suppression d'un sommet n'a presque aucun effet. Au contraire, la suppression d'un sommet sur un bord de ce type d'objet, produit des distorsions visuelles, qui sont représentées par le score

MSDM2. En général, ces résultats montrent que les distorsions résultantes de la fermeture des trous ne sont pas significatives pour le système visuel humain, selon la mesure MSMD2.

Bien que le domaine de l'évaluation de la sécurité en IDC 3D en est encore à ses débuts, nous estimons que les autres méthodes de grande capacité ne fournissent pas une sécurité importante. Nous montrons, en outre que la méthode proposée a une grande capacité pour de faibles distorsions. Nous comparons la méthode proposée avec d'autres méthodes haute capacité [15, 36, 52] en termes de capacité, et grâce aux métriques présentées dans le Chapitre 2, la distance de Hausdorff et le $PSNR_1$ [15]. Les résultats obtenus sur le maillage de l'objet "Bunny " sont présentés dans le Tableau 7.2 (les cellules vides correspondent à des résultats qui ne sont pas donnés par les auteurs).

Tableau 7.2: Comparaisons avec d'autres méthodes sur le modèle Bunny.

Méthode	Capacité	Distance de Hausdorff $_{\times 10^{-6}}$	$PSNR_1$
[15]	940464	×	100.57
[36]	51408	548	×
[52]	54289	1	127.3
Méthode proposée avec $\Delta = 1.10^{-4}$	411864	1796	82.55

Nous pouvons voir que notre méthode produit plus de distorsions que les autres méthodes en raison de la suppression de sommets. Néanmoins, les distorsions sont encore faibles et les métriques nous permettent de dire qu'elles ne sont pas visibles lors de l'insertion dans de grands maillages. La méthode proposée fournit donc un compromis entre une meilleure sécurité théorique et une méthode d'IDC de haute capacité.

7.3 Améliorations des méthodes proposées

Dans cette section, nous proposons des améliorations de l'ensemble des méthodes d'IDC présentées dans ce manuscrit. Nous avons proposé deux types d'insertion : une méthode uniforme et une méthode basée sur le codage arithmétique statique. Nous avons proposé de synchroniser les sommets du maillage à l'aide de deux méthodes, la construction d'un chemin hamiltonien avec une approche "plus proche voisin" et la construction d'un chemin à l'aide de sauts aléatoires. Pour résoudre les cas d'ambiguïté, qui surviennent à cause de l'insertion pendant la synchronisation, nous avons proposé deux stratégies. La première consiste à ne pas déplacer un sommet et la seconde supprime le sommet en cause. Ces méthodes peuvent être combinées différemment en fonction du besoin. Pour ces méthodes, nous proposons premièrement de réduire les distorsions

dues au déplacement des sommets dans la Section 7.3.1. Ensuite, nous proposons de réduire le nombre d'ambiguïtés et donc l'application d'une stratégie d'évitement dans la Section 7.3.2. La Section 7.3.3, présente un moyen efficace de diminuer les distorsions tout en réduisant le nombre de ambiguïtés. Finalement, la Section 7.3.4 présente des résultats expérimentaux.

7.3.1 Réduction des distorsions

Dans cette section, nous proposons de réduire les distorsions lors de l'étape d'insertion. En effet en réduisant l'écart entre les valeurs des coordonnées d'un sommet et ses nouvelles coordonnées après insertion, le déplacement du sommet produit est plus faible. Et nous pouvons supposer qu'un déplacement plus faible implique en général moins de distorsions. Il est impossible de diminuer l'intervalle de déplacement Δ en dessous d'une certaine limite. Donc nous proposons de réduire l'écart entre les valeurs en utilisant les intervalles de déplacement adjacents. Nous utilisons les intervalles adjacents comme dans la méthode d'IDC offrant une capacité de 3 bps proposée dans la Section 6.3. En effet, le déplacement peut être réduit avec cette stratégie, comme illustré sur la Fig. 7.10. Dans la méthode généralisée de la Section 6.4, nous fixons les déplacements à une valeur moyenne de $\frac{\Delta}{2}$, ce qui est intéressant pour prédire les distorsions du maillage mais nous n'utilisons pas les intervalles adjacents. De plus, avec la technique du déplacement dans un intervalle adjacent, les résultats ne sont que très légèrement améliorés. Dans la Fig. 7.10, nous voyons que pour insérer le message a, l'écart des valeurs dans l'intervalle de déplacement Δ_0 est supérieur à l'écart des valeurs dans l'intervalle de déplacement adjacent Δ_1.

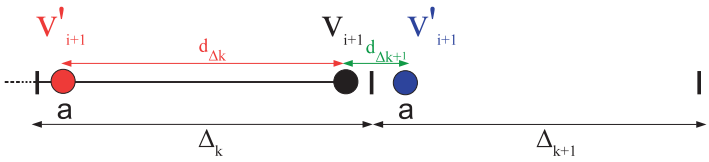


FIGURE 7.10: Comparaison des déplacements, en rouge déplacement classique, en vert déplacement dans l'intervalle adjacent.

Il est clair que si la valeur d'une coordonnée doit être déplacée de plus de $\frac{\Delta}{2}$ dans son intervalle, il est préférable de choisir sa nouvelle valeur dans un intervalle adjacent. Cette technique permet de réduire certains écarts entre les valeurs. Mais le plus intéressant est que nous avons constaté que cette méthode permet d'éviter des cas d'ambiguïtés. En effet, le déplacement dans une autre direction du sommet peut permettre de respecter les conditions de conservation de la synchronisation.

7.3.2 Diminutions du nombre d'ambiguïtés

Afin de diminuer les distorsions, nous pouvons utiliser les intervalles adjacents pour l'insertion d'une valeur. Cependant, un autre cas d'application consiste à utiliser ces intervalles adjacents lorsqu'un déplacement est interdit par l'étape de vérification. En effet, nous pouvons considérer qu'il est plus important de ne pas utiliser de stratégie de conservation du chemin, que de produire plus de distorsions en utilisant les intervalles adjacents. Pour l'insertion, si nous autorisons les intervalles adjacents et que nous acceptons plus de distorsions, nous pouvons essayer de trouver trois nouvelles coordonnées pour $v_i(\rho_i, \theta_i, \phi_i)$, dont la nouvelle position v_i' est validée par l'étape de vérification. Pour cela, nous redéfinissons la valeur d'une coordonnée comme :

$$c' = \left\lfloor \frac{c}{\Delta} \right\rfloor \Delta + n\frac{\Delta}{256} + k_c\Delta,\ k_c \in \mathbb{Z},\ c \in \{\rho, \theta, \phi\}, \tag{7.13}$$

où n correspond à la valeur à insérer. Pour limiter les recherches et les déplacements, nous choisissons de limiter le nombre d'intervalles tel que $k_c \in [-2, 2]$. Les intervalles sont illustrés sur la Fig. 7.11.

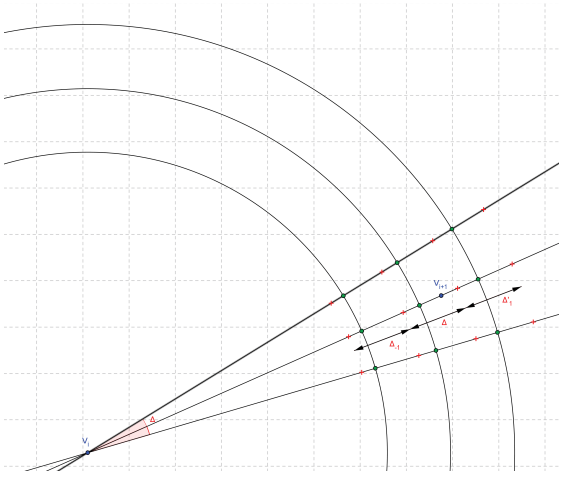


FIGURE 7.11: Illustration des valeurs possibles de la coordonnée ρ en rouge, dans les intervalles de déplacement adjacents.

La première intuition est alors de changer d'intervalle à chaque fois que la nouvelle position d'un sommet est interdite. Cependant cette méthode ne garantie pas de bonnes performances en terme de distorsions.

7.3.3 Stratégie d'optimisation

Lors de l'insertion de bits du message dans chaque sommet, nous devons calculer les nouvelles valeurs de chacune des coordonnées. Notre idée consiste à trouver les valeurs k dans l'équation 7.13 permettant de minimiser les distorsions du maillage. Le but est

de minimiser une distance notée d en fonction de l'ancienne position du sommet et la nouvelle position du sommet qui dépend du triplet $u = (k_\rho, k_\theta, k_\phi),\ u \in \mathbb{Z}^3$. Nous cherchons alors :

$$\underset{u \in \mathbb{Z}^3}{\operatorname{argmin}}\, d(v, v'(u)), \tag{7.14}$$

Cette équation nous permet de classer les positions des sommets, de celle produisant le moins de distorsions à celle en produisant le plus, en fonction de la définition de la distance d. Cette technique permet également d'éviter les problèmes de causalité en prenant la position suivante du sommet si le déplacement est interdit pour une position. En limitant l'intervalle de recherche à $k \in [-2, 2]$ par exemple, nous simplifions le problème et limitons les distorsions possibles. Cependant, il se peut que toutes les positions possibles pour un sommet soient interdites par l'étape de vérification. Il faut alors employer une stratégie, soit le non-déplacement, soit la suppression du sommet.

Pour le choix de la distance d, la première stratégie consiste à tester quelles valeurs de k_c dans l'équation 7.13, produisent les plus faibles écarts entre la valeur c et c'. Nous pouvons supposer qu'en prenant les trois valeurs minimisant les écarts, la nouvelle position du sommet est la position minimisant les distorsions. Nous définissons alors d simplement comme la distance euclidienne :

$$d(v, v') = \left\| v, v' \right\|_2 . \tag{7.15}$$

Cependant, nous considérons que pour ce problème de minimisation des distorsions, il est préférable d'évaluer les distorsions sur le maillage directement que de minimiser les déplacements. Dans ce but, nous proposons de minimiser le déplacement par rapport au plan tangent calculé à partir de la position initiale du sommet, comme illustré dans la Fig. 7.12.

Dans un premier temps, nous calculons la normale du plan tangent p, au sommet à déplacer v. Cette normale est calculée comme la moyenne des normales pondérées des facettes adjacentes au sommet v. Une comparaison des algorithmes de calcul de normales de sommets a été proposée par Jin *et. al* [57], nous utilisons la moyenne pondérée également puisque son exécution est rapide et que pour une bonne résolution du maillage, ses performances sont correctes. Nous calculons alors la distance d entre un point v' et un plan définit par sa normale n en un point v comme :

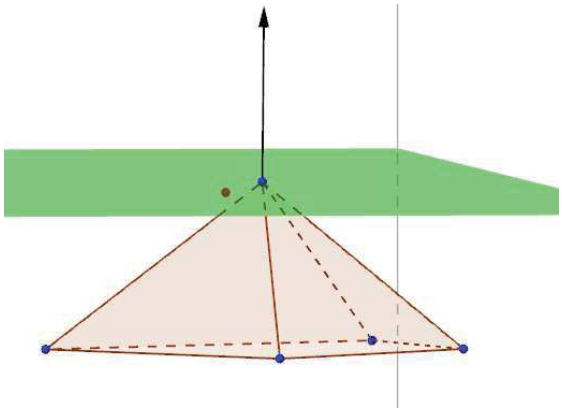


FIGURE 7.12: Illustration de la nouvelle position d'un sommet en rouge, minimisant la distance avec le plan tangent de son ancienne position.

$$d(v, v') = \frac{|v'_x n_x + v'_y n_y + v'_z n_z|}{\sqrt{n_x^2 + n_y^2 + n_z^2}}. \tag{7.16}$$

Cette optimisation est donc un moyen de réduire les distorsions et d'éviter les problèmes de désynchronisation sans pour autant les éliminer complètement.

7.3.4 Résultats expérimentaux

Dans cette partie nous utilisons la base de maillages présentée dans le Chapitre 6. Pour comparer les résultats et montrer le gain de performances de cette optimisation, nous proposons de reprendre le Tableau 6.10 présenté dans la Section 6.4.5. Nous utilisons la méthode d'optimisation basée sur la réduction de la distance au plan tangent avec la stratégie de non déplacement du sommet en cas d'interdiction de déplacement. La stratégie de suppression garantie d'obtenir le bon résultat, ce qui fausserait la comparaison.

Nous évaluons donc les résultats sur toute la base de maillages, afin d'analyser les déformations et la qualité du message extrait. Les paramètres de l'algorithme sont conservés à savoir $\Delta = 10^{-4}$ et $\gamma = \frac{1}{10}$. Le Tableau 7.3 présente les résultats pour la méthode CAS (Codage Arithmétique Statique) et nous utilisons les mêmes métriques pour mesurer les distorsions du maillage la métrique $PSNR_1$ [15] et la métrique $MSDM2$ [71]. La qualité du message est toujours donnée par le BER et le PSNR.

Nous pouvons voir que dans la plupart des maillages, il n'y a aucun problème lors de l'insertion et tout le message peut être inséré correctement. Nous constatons également que les distorsions sont généralement plus faibles. Cependant, certains maillages possèdent des distorsions plus importantes que dans l'expérimentation

TABLEAU 7.3: Résultats de la méthode SAC avec optimisation basée sur la réduction de la distance au plan tangent.

Maillage	# de sommets	Charge utile (en pixels)	Distorsion du maillage		Qualité du message	
			$PSNR_1$ (dB)	MSDM2 ($\times 10^{-3}$)	PSNR (dB)	BER ($\times 10^{-3}$)
alien	7401	7056	117.87	3.79	$+\infty$	0
Armadillo	172974	172225	128.81	1.78	45.68	0.0278
baby	5075	4761	117.01	0.39	$+\infty$	0
bitorus	3000	2704	112.14	0.30	$+\infty$	0
blade2	24738	24336	128.75	3.07	$+\infty$	0
boy	8441	8100	118.22	0.15	$+\infty$	0
bunny	34834	34225	121.93	0.25	$+\infty$	0
cad	1426	1156	111.35	3.55	$+\infty$	0
casting	5096	4761	112.54	2.89	$+\infty$	0
chair1	12326	11881	118.16	7.89	30.71	0.0032
chair2	13463	12996	118.85	4.88	31.35	0.0023
cow	2904	2601	113.44	25.35	$+\infty$	0
crank	50004	49729	122.09	7.83	$+\infty$	0
dinopet	4500	4225	116.10	1.50	$+\infty$	0
dragon	50000	49729	123.31	1.28	$+\infty$	0
eagle	1000	729	111.23	0.63	$+\infty$	0
hand	36619	36100	121.95	8.49	59.19	0.0120
hand1	26000	25600	120.53	2.97	49.76	0.0088
horse	112642	112225	128.62	1.24	$+\infty$	0
horse1	20000	19600	120.99	0.19	$+\infty$	0
horse2	2450	2116	111.76	0.25	$+\infty$	0
L9305	31088	30625	122.23	4.55	$+\infty$	0
rabbit	70658	70225	125.55	0.26	71.67	0.0018
ShoeS	45002	44521	122.63	1.39	$+\infty$	0
ShoeS2	5002	4624	114.42	1.07	$+\infty$	0
table1	10082	9801	118.79	5.10	33.98	0.0014
table2	13579	13225	118.28	5.66	$+\infty$	0
US3266	199093	198025	129.49	2.36	$+\infty$	0
US3266-2	50002	49729	123.51	1.61	$+\infty$	0
us5105c-r	83698	82944	126.39	4.06	$+\infty$	0
venus	100759	100489	125.62	1.37	56.44	0.0027

précédente. Ces résultats sont dus au fait que dans ces maillages très peu de sommets étaient déplacés et donc le message était partiellement inséré. Ici nous constatons que les distorsions sont plus importantes puisque presque tous les sommets sont support d'information et donc déplacés. Le message extrait est de meilleure qualité puisque les taux d'erreurs sont très faibles. De plus, dans le cas d'une image le fait que le PSNR soit élevé indique que l'image extraite est presque identique à l'originale. Finalement, nous attendons que cette méthode se comporte de la même manière par rapport aux analyses statistiques concernant la sécurité. En effet, cette nouvelle méthode est conçue sur le

schéma d'IDC basée sur le CAS qui permet d'éviter l'estimation des paramètres d'insertion, ainsi que la distribution uniforme des valeurs des coordonnées dans les intervalles de déplacement.

7.4 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle synchronisation pour l'IDC haute capacité pour des maillages d'objets 3D. Nous proposons une synchronisation basée sur la construction d'un chemin hamiltonien et de sauts aléatoires sur le maillage. L'idée est de diffuser le message sur le maillage, qui n'exploite pas forcément toute la capacité disponible. Le chemin permet d'ordonner des paires de sommets qui contiennent l'information cachée. La méthode d'IDC proposée accroit la sécurité grâce à l'utilisation d'une clé secrète, qui est utilisée pour choisir le sommet initial et le rayon du saut. Les couples de sommets sur le maillage 3D sont classés comme sommets de référence qui ne sont pas déplacés ou comme sommets supports qui eux le sont. Lors de l'insertion jointe à la synchronisation dans un algorithme itératif, peu de sommets interfèrent avec la stabilité du chemin. Nous proposons de supprimer ces sommets du maillage de l'objet 3D. Les trous dans la forme 3D sont fermés par remaillage triangulaire. Les distorsions du maillage après la suppression des sommets qui produisent des problèmes, ne sont pas importantes. Cette méthode offre une capacité d'insertion de données d'environ 12 bits par sommet, tout en produisant de faibles distorsions.

Plus généralement, nous présentons comment améliorer toutes les méthodes d'IDC proposées dans ces travaux. Indépendamment de la méthode d'ordonnancement et quelque soit la stratégie de conservation de la synchronisation choisie, nous proposons une méthode permettant de réduire les distorsions. Cette méthode est basée sur l'utilisation d'un intervalle de déplacement adjacent à celui donné par la méthode d'insertion. À partir de cette idée, nous avons utilisé plusieurs des intervalles adjacents dans le but de trouver une position du sommet qui respecte les conditions de maintient de la synchronisation. Finalement, nous expliquons comment choisir parmi les différentes positions respectant les conditions, la position minimisant les distorsions.

Ces travaux on fait l'objet d'une publication dans une conférence internationale, IS&T EI 2016 [55].

Chapitre 8

Conclusion et perspectives

Dans ce chapitre, nous présentons un bilan du travail effectué ainsi que nos pistes de recherches. Dans la Section 8.1, nous récapitulons le contenu du manuscrit et nous dressons un bilan des contributions apportées. Finalement, dans la Section 8.2 nous présentons quelques challenges à considérer ainsi que nos idées pour y contribuer.

8.1 Conclusion

Dans cette thèse, nous avons proposé de nouvelles méthodes de synchronisation pour l'insertion de données cachées haute capacité. Après avoir expliqué et défini les différentes notions concernant les maillages 3D et la sécurité des médias visuels, nous avons dressé un état de l'art détaillé des méthodes de références à partir desquelles nos travaux ont été abordés. Dans le cadre applicatif fixé par la société STRATEGIES, en fonction des scénarios d'utilisation envisagés et des contraintes, nous nous sommes particulièrement intéressés à l'insertion de données haute capacité en utilisant la géométrie du maillage comme support du message à insérer. Dans ce contexte général, nous avons voulu proposer des solutions permettant d'ajouter des méta-informations dans un maillage, un numéro d'utilisateur, et si possible de vérifier son intégrité. Un schéma d'insertion de données cachées classique consiste à définir une étape de synchronisation pour ordonner les éléments du maillage, suivie d'une étape d'insertion des données. Dans nos conditions, nous utilisons le plongement des sommets dans l'espace, c'est-à-dire leur position donnée par des coordonnées, comme éléments de base sur lequel nous voulons définir un ordre et que nous souhaitons utiliser comme support du message à cacher.

Notre première contribution à donc consisté à de définir une synchronisation sur la géométrie du maillage, après avoir analysé en détail les ACPMs (Arbres Couvrants de

Poids Minimum). Pour cela, nous avons étudié les chemins hamiltoniens. Ces chemins nous permettent de définir un ordre des sommets stable à ε près. Nous avons alors utilisé une méthode de regroupement de sommets afin d'augmenter la stabilité du chemin. Nos résultats montrent que cette solution permet d'améliorer la résistance à l'ajout de bruit. De plus, cette étude nous a permis de comprendre la construction des chemins hamiltoniens avec leurs avantages et leurs faiblesses.

Dans notre seconde contribution, du fait de la sensibilité de la construction des chemins, lorsque nous déplaçons un sommet pour l'insertion, la synchronisation peut être perdue. Nous proposons donc de joindre les étapes de synchronisation et d'insertion pour prendre en compte ces problèmes de causalité au moment où ils pourraient survenir. L'insertion de données cachées que nous avons proposée se fait en déplaçant un sommet suivant les trois dimensions de l'espace dans un intervalle de déplacement borné afin de limiter les distorsions. Puisque la synchronisation et l'insertion sont réalisées conjointement, nous pouvons éviter un déplacement qui impliquerait un problème de causalité. L'insertion d'une partie du message se fait alors en modifiant la valeur d'une coordonnée afin qu'elle corresponde à un sous-intervalle de valeurs qui codent cette partie du message dans l'intervalle de déplacement. La capacité d'un intervalle est bornée par la précision du format de stockage des valeurs, mais peut atteindre 8 bits par coordonnée, ce qui donne en théorie 24 bits par sommet. Cependant, éviter un déplacement pour conserver la synchronisation implique de ne pas insérer une partie du message. Nous montrons que ce problème peut être résolu en utilisant des codes correcteurs d'erreurs. Nous avons ensuite analysé la sécurité de cette méthode. Le fait d'utiliser une répartition uniforme des valeurs du message dans les intervalles, permet de déceler la présence ou non d'un message. Nous avons donc proposé de changer cette répartition afin d'uniformiser la distribution des valeurs des sommets. Pour cela, nous avons proposé d'utiliser un codage arithmétique statique afin de redéfinir la taille des sous-intervalles. Cette méthode présente de bons résultats en terme de détection d'un message.

Dans notre troisième contribution, nous avons conclu qu'un des défauts en matière de sécurité des méthodes précédentes proposées est principalement la synchronisation. En effet, la construction d'un chemin hamiltonien est déterministe et si le secret du premier sommet n'est pas conservé, il est aisé de reconstruire correctement le chemin complet des sommets. Nous proposons donc d'utiliser un chemin aléatoire sur le maillage défini à l'aide d'une génération pseudo-aléatoire de sommets à partir d'une clé secrète. En outre, nous avons constaté que l'insertion de données n'utilisant pas la capacité maximale du maillage, cache le message dans une zone autour du point d'entrée. Ce qui n'est pas efficace d'un point de vue sécurité et pour la distorsion du maillage. Cette nouvelle méthode proposée permet d'effectuer des sauts dans le maillage afin de diffuser le message inséré. Cette nouvelle synchronisation sécurisée est utilisée pour de l'insertion haute

capacité offrant une capacité d'environ 12 bits par sommet, puisque nous ne pouvons pas utiliser l'ensemble des sommets comme support. Nous proposons également une nouvelle méthode de résolution du problème de causalité pour les maillages triangulaires denses. L'idée est que la suppression d'un sommet sur ce type de maillage est pratiquement invisible dans les zones planes. Par conséquent, la suppression d'un sommet permet de conserver une synchronisation sur des sommets tous porteurs d'information. Finalement, nous montrons comment réduire le problème de causalité en utilisant des intervalles de déplacement adjacents, ce qui permet de choisir une position pour un sommet parmi plusieurs contenant l'information. Ce choix est fait afin d'éviter la désynchronisation et de minimiser les distorsions.

En conclusion, dans ces travaux de recherche nous avons apporté plusieurs solutions à notre problème répondant à divers scénarios d'utilisation. Bien qu'intéressantes, ces méthodes présentent tout de même des défauts, c'est pour cela que nous proposons des pistes afin de les améliorer.

8.2 Perspectives

Dans cette section, nous présentons dans un premier temps des perspectives d'insertion de données cachées 3D, puis nous listons des améliorations possibles à apporter à nos travaux.

Contexte général :

L'IDC dans les objets 3D souffre de la comparaison avec l'IDC dans des images ou dans des vidéos. Les enjeux, comme la synchronisation ou l'utilisation des domaines transformés, sont dépendants du volume des maillages 3D et trouver une méthode à la fois robuste et rapide est important dans ces domaines. D'autre part, des travaux récents ne s'intéressent principalement qu'au développement de nouvelles méthodes d'IDC et de stéganographie des maillages 3D, et la sécurité n'est pas souvent analysée. Dans le cas de la stéganographie 3D, très peu de méthodes ont été proposées. Cependant, la réalisation d'une méthode de stéganalyse et d'une plateforme de tests comparable à ce qui se fait en 2D est un objectif important. En effet, il peut être intéressant de vérifier la présence d'un message dans un maillage 3D, aussi bien pour une méthode de stéganographie que pour une méthode de tatouage robuste. Dans le cas de l'IDC haute capacité, les méthodes de la littérature et celles proposées atteignent des capacités importantes. Néanmoins, ces méthodes sont rarement contraintes par le maillage et ses propriétés. Il pourrait être

intéressant de proposer une méthode basée sur l'optimisation minimisant les distorsions du maillage de façon globale, comme ce qui est fait en tatouage robuste de maillages 3D.

Perspectives liées à nos travaux de recherches :

Nous pensons que la synchronisation basée sur les graphes possède des propriétés intéressantes et qu'il serait intéressant d'étudier l'ordonnancement d'autres éléments des maillages comme sommets du graphe. Nous avons utilisé les ACPMs et les chemins hamiltoniens mais d'autres structures peuvent également être analysées. L'augmentation de la stabilité par regroupement, peut mener à des méthodes d'IDC. Cependant, nous estimons que cette méthode possède des inconvénients à résoudre en priorité. La dépendance au volume englobant et la supposition de la répartition uniforme des sommets dans ce volume sont problématiques dans les calculs des estimations du comportement. De plus ces approches ne sont pas robustes aux attaques par découpage. Nous souhaitons également analyser le comportement et la stabilité de ces chemins sur d'autres transformations du maillage comme par exemple l'utilisation des résolutions inférieures d'un maillage ou une autre méthode de regroupement des sommets.

Dans les méthodes d'IDC proposées dans nos travaux, un des principaux inconvénients a été de s'occuper du problème de causalité. Nous réfléchissons alors à des critères de construction de structures évitant ces ambiguïtés. D'autres part, le développement d'une méthode adaptative à la taille du message, permettant de sub-diviser un intervalle selon les besoins et selon les distorsions semble être une piste intéressante. Lors d'un problème de causalité, il est important de considérer un déplacement permettant de perdre le moins de message possible en fonction des distorsions locales. Par exemple, une approche intéressante serait de déplacer un sommet dans un plan ou suivant une direction qui respecte les contraintes de déplacement et les limites de distorsions. Ce processus pourrait se faire en conservant une partie du message, si conserver son intégralité est impossible. Le message peut dans ce cas contenir un code correcteur d'erreurs.

Notre méthode de minimisation consiste à minimiser la distance au plan tangent d'un sommet, calculé par son 1-voisinage. Cette mesure de distorsions prend en compte une information très locale à la surface. Il peut être intéressant d'étudier une minimisation en fonction d'autres métriques. Enfin, nous pensons qu'il est possible de générer un chemin pseudo-aléatoire sur la géométrie du maillage capable de se resynchroniser en cas d'attaques, peux pas voire de découpage sur la géométrie du maillage.

Liste des contributions

Conférence nationale :

— [49] V. Itier, W. Puech, G. Gesquière, and J.-P. Pedeboy. Construction d'un chemin hamiltonien unique et robuste dans un nuage de points 3D. In CORESA, pages 78–83, 2013.

Conférence internationale :

— [51] V. Itier, W. Puech, J.-P. Pedeboy, and G. Gesquière. Construction of a unique robust hamiltonian path for a vertex cloud. In IEEE 15th International Workshop on Multimedia Signal Processing, pages 105–110, 2013.

— [51] V. Itier, W. Puech, and A.G. Bors. Cryptanalysis aspects in 3-D watermarking. In IEEE International Conference on Image Processing (ICIP), pages 4772–4776, 2014.

— [52] V. Itier, W. Puech, G. Gesquière, and J.-P. Pedeboy. Joint synchronization and high capacity data hiding for 3D meshes. SPIE Electronic Imaging, 9393 :939305–939305–15, 2015.

— [53] V. Itier, W. Puech, and J.-P. Pedeboy. High capacity data-hiding for 3D meshes based on static arithmetic coding. In IEEE International Conference on Image Processing (ICIP), 2015.

— [55] V. Itier,W. Puech, A.G. Bors, and J.-P. Pedeboy. Secure high capacity data hiding for 3D meshes. In IS&T Electronic Imaging, 2016.

Revue internationale :

— [54] V. Itier, N. Tournier, W. Puech, G. Subsol, and J.-P. Pedeboy. Analysis of an EMST-based path for 3D meshes. Computer-Aided Design, 64(0) :22 – 32, 2015.

— [56] V. Itier, W. Puech, and J.-P. Pedeboy. High capacity data hiding for 3D meshes. In Multimedia Tools and Applications. Springer, soumis.

Bibliographie

[1] P. Amat, W. Puech, S. Druon, and J.P. Pedeboy. Lossless 3D steganography based on MST and connectivity modification. *Signal Processing : Image Communication*, 25(6) :400–412, 2010.

[2] N. Aspert, D. Santa-Cruz, and T. Ebrahimi. Mesh : measuring errors between surfaces using the hausdorff distance. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 1, pages 705–708, 2002.

[3] C.L. Bajaj, V. Pascucci, and G. Zhuang. Progressive compression and transmission of arbitrary triangular meshes. In *Visualization '99. Proceedings*, pages 307–537, 1999.

[4] B. G. Baumgart. Winged edge polyhedron representation. Technical report, DTIC Document, 1972.

[5] M. Bellare, T. Ristenpart, P. Rogaway, and T. Stegers. Format-preserving encryption. In *Selected Areas in Cryptography*, volume 5867 of *LNCS*, pages 295–312, 2009.

[6] O. Benedens. Geometry-based watermarking of 3D models. *IEEE Computer Graphics and Applications*, 19(1) :46–55, 1999.

[7] M. Berger, Tagliasacchi A., L. M. Seversky, P. Alliez, J. A. Levine, A. Sharf, and C. T. Silva. State of the art in surface reconstruction from point clouds. In *Eurographics 2014 - State of the Art Reports*, volume 1, pages 161–185, 2014.

[8] A. Bogomjakov, C. Gotsman, and M. Isenburg. Distortion-free steganography for polygonal meshes. In *Computer Graphics Forum*, volume 27, pages 637–642, 2008.

[9] A. G. Bors and M Luo. Optimized 3D watermarking for minimal surface distortion. *IEEE Transactions on Image Processing*, 22(5) :1822–1835, 2013.

[10] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Levy. *Polygon Mesh Processing*. Ak Peters Series. Taylor & Francis, 2010.

[11] J. Cao, J. Huang, and J. Ni. A new spread spectrum watermarking scheme to achieve a trade-off between security and robustness. In R. Böhme, P.W.L. Fong, and Reihaneh S.-N., editors, *Information Hiding*, volume 6387 of *Lecture Notes in Computer Science*, pages 262–276. Springer, 2010.

[12] F. Cayre and B. Macq. Data hiding on 3-D triangle meshes. *IEEE Transactions on Signal Processing*, 51(4) :939–949, 2003.

[13] F. Cayre, C. Fontaine, and T. Furon. Watermarking security : theory and practice. *IEEE Transactions on Signal Processing,*, 53(10) :3976–3987, 2005.

[14] H.-C. Chang, H.-P. Lee, T.C. Lin, and T.K. Truong. A weight method of decoding the (23, 12, 7) golay code using reduced table lookup. In *International Conference on Communications, Circuits and Systems.*, pages 1–5, 2008.

[15] M.-W. Chao, C.-H. Lin, C.-W. Yu, and T.-Y. Lee. A high capacity 3D steganography algorithm. *IEEE Transactions on Visualization and Computer Graphics*, 15(2) :274–284, 2009.

[16] B. Chen and G. W. Wornell. Quantization index modulation methods for digital watermarking and information embedding of multimedia. *Journal of VLSI signal processing systems for signal, image and video technology*, 27(1/2) :7–33, 2001.

[17] Y.-M. Cheng and C.-M. Wang. An adaptive steganographic algorithm for 3D polygonal meshes. *The Visual Computer*, 23(9-11) :721–732, 2007.

[18] J.-W. Cho, R. Prost, and H.-Y. Jung. An oblivious watermarking for 3-D polygonal meshes using distribution of vertex norms. *Signal Processing, IEEE Transactions on*, 55(1) :142–155, 2007.

[19] P. Cignoni, C. Rocchini, and R. Scopigno. Metro : Measuring error on simplified surfaces. Technical report, Centre National de la Recherche Scientifique, 1996.

[20] M. Corsini, E. Drelie Gelasca, and T. Ebrahimi. A multi-scale roughness metric for 3D watermarking quality assessment. In *Workshop on Image Analysis for Multimedia Interactive Services*. SPIE, 2005.

[21] M. Corsini, E.D. Gelasca, T. Ebrahimi, and M. Barni. Watermarked 3-D mesh quality assessment. *IEEE Transactions on Multimedia*, 9(2) :247–256, 2007.

[22] D. Cotting, T. Weyrich, M. Pauly, and M. Gross. Robust watermarking of point-sampled geometry. In *Proceedings Shape Modeling Applications*, pages 233–242, 2004.

[23] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker. *Digital Watermarking and Steganography*. Morgan Kaufmann, 2007.

[24] I. J. Cox, F.T. Kilian, J.and Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. *IEEE Transactions on Image Processing*, 6(12) : 1673–1687, 1997.

[25] B. Delaunay. Sur la sphère vide. a la mémoire de georges voronoï. *Bulletin de l'Académie des Sciences de l'URSS. Classe des sciences mathématiques et naturelles*, pages 793–800, 1934.

[26] M. Desoubeaux, C. Herzet, W. Puech, and G. Le Guelvouit. Enhanced Blind Decoding of Tardos Codes with New Map-Based Functions. In *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pages 283–288, 2013.

[27] N.V. Dharwadkar, B.B. Amberker, and A. Gorai. Non-blind watermarking scheme for color images in RGB space using DWT-SVD. In *International Conference on Communications and Signal Processing*, pages 489–493, 2011.

[28] T. Dieck. *Algebraic Topology*. EMS textbooks in mathematics. European Mathematical Society, 2008.

[29] J.J. Eggers, R. Bauml, R. Tzschoppe, and B. Girod. Scalar costa scheme for information embedding. *IEEE Transactions on Signal Processing*, 51(4) :1003–1019, 2003.

[30] M. Eluard, Y. Maetz, and G. Doerr. Impact of geometry-preserving encryption on rendering time. In *IEEE International Conference on Image Processing (ICIP)*, pages 4787–4791, 2014.

[31] J. Fridrich. *Steganography in Digital Media : Principles, Algorithms, and Applications*. Cambridge University Press, 1st edition, 2009.

[32] J. Fridrich and J. Kodovsky. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3) :868–882, 2012.

[33] T. Furon. A constructive and unifying framework for zero-bit watermarking. *IEEE Transactions on Information Forensics and Security*, 2(2) :149–163, 2007.

[34] T. Furon and P. Bas. A new measure of watermarking security applied on QIM. In *Information Hiding*, volume 7692, pages 207–223. Springer, 2013.

[35] T. Furon and M. Desoubeaux. Tardos codes for real. In *IEEE Workshop on Information Forensics and Security*, pages 24–29, 2014.

[36] X. Gao, C. Zhang, Y. Huang, and Z. Deng. A robust high-capacity affine-transformation-invariant scheme for watermarking 3D geometric models. *ACM Transaction on Multimedia Computing, Communications and Applications*, 8(2S) : 34 :1–34 :21, 2012.

[37] H. Garg, S. Agrawal, and G. Varshneya. A non-blind image based watermarking for 3-D polygonal mesh using its geometrical properties. In *International Conference on Contemporary Computing (IC3)*, pages 313–318, 2013.

[38] M. J. E. Golay. Notes on digital coding. *IEEE Proceedings*, 37 :657, 1949.

[39] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, 3rd edition, 2006.

[40] S. Gumhold, Z. Kami, M. Isenburg, and H.-P. Seidel. Predictive point-cloud compression. In *ACM SIGGRAPH 2005 Sketches*, page 137, 2005.

[41] T. Gurung, M. Luffel, P. Lindstrom, and J. Rossignac. Lr : Compact connectivity representation for triangle meshes. *ACM Transactions on Graphics*, 30(4) :67 :1–67 :8, 2011.

[42] Hoppe H. Progressive meshes. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 99–108. ACM, 1996.

[43] W. R. Hamilton. On the icosian calculus. *Proceedings of the Royal Irish Academy*, 6 :462, 1853.

[44] H. He and J. Zhang. Cryptanalysis on majority-voting based self-recovery watermarking scheme. *Telecommunication Systems*, 49(2) :231–238, 2012.

[45] C. M. Hoffmann. *Geometric and Solid Modeling : An Introduction*. Morgan Kaufmann Publishers Inc., 1989.

[46] V. Holub and J. Fridrich. Digital image steganography using universal distortion. In *ACM Workshop on Information Hiding and Multimedia Security*, IH&MMSec '13, pages 59–68, 2013.

[47] N.-C. Huang, M.-T. Li, and C.-M. Wang. Toward optimal embedding capacity for permutation steganography. *IEEE Signal Processing Letters*, 16(9) :802–805, 2009.

[48] Y.-H. Huang and Y.-Y. Tsai. A reversible data hiding scheme for 3D polygonal models based on histogram shifting with high embedding capacity. *3D Research*, 6(2) :20, 2015.

[49] V. Itier, W. Puech, G. Gesquière, and J.-P. Pedeboy. Construction d'un chemin hamiltonien unique et robuste dans un nuage de points 3D. In *CORESA*, pages 78–83, 2013.

[50] V. Itier, W. Puech, J.-P. Pedeboy, and G. Gesquière. Construction of a unique robust hamiltonian path for a vertex cloud. In *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pages 105–110, 2013.

[51] V. Itier, W. Puech, and A.G. Bors. Cryptanalysis aspects in 3-D watermarking. In *IEEE International Conference on Image Processing (ICIP)*, pages 4772–4776, 2014.

[52] V. Itier, W. Puech, G. Gesquière, and J.-P. Pedeboy. Joint synchronization and high capacity data hiding for 3D meshes. *Proc. SPIE*, 9393 :939305–939305–15, 2015.

[53] V. Itier, W. Puech, and J.-P. Pedeboy. High capacity data-hiding for 3D meshes based on static arithmetic coding. In *IEEE International Conference on Image Processing (ICIP)*, 2015.

[54] V. Itier, N. Tournier, W. Puech, G. Subsol, and J.-P. Pedeboy. Analysis of an EMST-based path for 3D meshes. *Computer-Aided Design*, 64 :22–32, 2015.

[55] V. Itier, W. Puech, A.G. Bors, and J.-P. Pedeboy. Secure high capacity data hiding for 3D meshes. In *IS&T Electronic Imaging*, in submission.

[56] V. Itier, W. Puech, and J.-P. Pedeboy. High capacity data hiding for 3D meshes. In *Multimedia Tools and Applications*. Springer, in submission.

[57] S. Jin, R. R. Lewis, and D. West. A comparison of algorithms for vertex normal computation. *The Visual Computer*, 21(1-2) :71–82, 2005.

[58] T. Kalker. Considerations on watermarking security. In *IEEE Workshop on Multimedia Signal Processing*, pages 201–206, 2001.

[59] S. Kanai, H. Date, and T. Kishinami. Digital watermarking for 3D polygons using multiresolution wavelet decomposition. In *Proceedings of Sixth IFIP WG*, volume 5, pages 296–307, 1998.

[60] Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 279–286. ACM, 2000.

[61] H. Kaveh and M.-S. Moin. A high-capacity and low-distortion 3D polygonal mesh steganography using surfacelet transform. *Security and Communication Networks*, 8(2) :159–167, 2015.

[62] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, SGP '06, pages 61–70. Eurographics Association, 2006.

[63] A. Kerckhoffs. La cryptographie militaire. *Journal des sciences militaires*, IX : 5–38, 1883.

[64] J. M. Konstantinides, A. Mademlis, P. Daras, P. A. Mitkas, and M. G. Strintzis. Blind robust 3-D mesh watermarking based on oblate spheroidal harmonics. *Transaction on Multimedia*, 11(1) :23–38, 2009.

[65] S. Kouider, M. Chaumont, and W. Puech. Adaptive steganography by oracle (ASO). In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2013.

[66] J.B. Kruskal Jr. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proceedings of the American Mathematical society*, 7(1) :48–50, 1956.

[67] D. Kundur and D. Hatzinakos. Digital watermarking for telltale tamper proofing and authentication. *Proceedings of the IEEE*, 87(7) :1167–1180, 1999.

[68] M. Kutter, S. Voloshynovskiy, and A. Herrigel. The watermark copy attack. *Proceedings of SPIE*, 3971 :371–380, 2000.

[69] G. G. Langdon. Arithmetic coding. *IBM Journal of Research and Development*, 23 :149–162, 1979.

[70] G. Lavoué, E. Drelie Gelasca, F. Dupont, A. Baskurt, and T. Ebrahimi. Perceptually driven 3D distance metrics with application to watermarking. In *SPIE Applications of Digital Image Processing*, volume 6312, pages 63120L.1–63120L.12, 2006.

[71] G. Lavoué. A multiscale metric for 3D mesh visual quality assessment. *Computer Graphics Forum*, 30(5) :1427–1437, 2011.

[72] G. Lavoué, F. Denis, and F. Dupont. Subdivision surface watermarking. *Computers & Graphics*, 31(3) :480 – 492, 2007.

[73] G. Lavoué, M Tola, and F Dupont. MEPP - 3D mesh processing platform. In *GRAPP & IVAPP 2012 : Proceedings of the International Conference on Computer Graphics Theory and Applications and International Conference on Information Visualization Theory and Applications*, pages 206–210, 2012.

[74] C. H. Lee, A. Varshney, and D. W. Jacobs. Mesh saliency. *ACM Transactions on Graphics*, 24(3) :659–666, 2005.

[75] M. T. Li, N. C. Huang, and C. M. Wang. A novel high capacity 3D steganographic algorithm. *International Journal of Innovative Computing, Information and Control*, 7(3) :1055–1074, 2011.

[76] C.-C. Lin and P.-F. Shiu. High capacity data hiding scheme for DCT-based images. *Journal of Information Hiding and Multimedia Signal Processing*, 1(3) :220–240, 2010.

[77] C.-H Lin, M.-W Chao, J.-Y Chen, C.-W Yu, and W.-Y and Hsu. A high-capacity distortion-free information hiding algorithm for 3D polygon models. *nternational Journal of Innovative Computing, Information and Control*, 20(3) :1321–1335, 2013.

[78] C.-Y. Lin, M. Wu, J.A. Bloom, I. J. Cox, M.L. Miller, and Lui Y. M. Rotation, scale, and translation resilient watermarking for images. *IEEE Transactions on Image Processing*, 10(5) :767–782, 2001.

[79] H.-Y. S. Lin, H.-Y.M. Liao, C.-S. Lu, and J.-C. Lin. Fragile watermarking for authenticating 3-D polygonal meshes. *IEEE Transactions on Multimedia*, 7(6) : 997–1006, 2005.

[80] Yang Liu, Balakrishnan Prabhakaran, and Xiaohu Guo. A robust spectral approach for blind watermarking of manifold surfaces. In *Workshop on Multimedia and Security*, MM&Sec '08, pages 43–52. ACM, 2008.

[81] M. Lounsbery, T. D. DeRose, and J. Warren. Multiresolution analysis for surfaces of arbitrary topological type. *Transactions on Graphics*, 16(1) :34–73, 1997.

[82] M. Luo and A.G. Bors. Surface-preserving robust watermarking of 3-D shapes. *IEEE Transactions on Image Processing*, 20(10) :2813–2826, 2011.

[83] X. Mao, M. Shiba, and A. Imamiya. Watermarking 3D geometric models through triangle subdivision. *Proceedings of SPIE*, 4314 :253–260, 2001.

[84] Y. Maret and T. Ebrahimi. Data hiding on 3D polygonal meshes. In *Workshop on Multimedia and Security*, MM&Sec '04, pages 68–74. ACM, 2004.

[85] G. Marsaglia. Choosing a point from the surface of a sphere. *The Annals of Mathematical Statistics*, 43(2) :645–646, 1972.

[86] B. Merry, P. Marais, and J. Gain. Compression of dense and regular point clouds. In *International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa*, AFRIGRAPH '06, pages 15–20. ACM, 2006.

[87] R. Motwani, M.C. Motwani, F.C. Harris, and S.M Dascalu. An eigen-normal approach for 3D mesh watermarking using support vector machines. *Journal of Electronic Science and Technology*, 8(3) :237–243, 2010.

[88] D. Neeta, K. Snehal, and D. Jacobs. Implementation of LSB steganography and its evaluation for various bits. In *IEEE International Conference on Digital Information Management*, pages 173–178, 2007.

[89] Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su. Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(3) :354–362, 2006.

[90] R. Ohbuchi, H. Masuda, and M. Aono. Watermaking three-dimensional polygonal models. *International Conference on Multimedia*, 16 :261–272, 1997.

[91] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama. Watermarking 3D polygonal meshes in the mesh spectral domain. In *Proceedings of Graphics Interface*, GI '01, pages 9–17, 2001.

[92] R. Ohbuchi, A. Mukaiyama, and S. Takahashi. Watermarking a 3D shape model defined as a point set. In *International Conference on Cyberworlds*, pages 392–399, 2004.

[93] B. O'Neill. *Elementary Differential Geometry, Revised 2nd Edition.* Elementary Differential Geometry Series. Elsevier Science, 2006.

[94] L. Perez-Freire and F. Perez-Gonzalez. Spread spectrum watermark security. *IEEE Transactions on Information Forensics and Security*, 4(1) :2–24, 2009.

[95] F.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn. Information hiding-a survey. *Proceedings of the IEEE*, 87(7) :1062–1078, 1999.

[96] T. Pevný, T. Filler, and P. Bas. Using high-dimensional image models to perform highly undetectable steganography. In Rainer B., P.W.L. Fong, and R. Safavi-Naini, editors, *Information Hiding*, volume 6387 of *Lecture Notes in Computer Science*, pages 161–177. Springer, 2010.

[97] E. Praun, H. Hoppe, and A. Finkelstein. Robust mesh watermarking. In *Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, pages 49–56. ACM, 1999.

[98] R.C. Prim. Shortest Connection Networks and Some Generalizations. *Bell System Technical Journal*, 36 :1389–1401, 1957.

[99] Y. Qian, J. Dong, W. Wang, and T. Tan. Deep learning for steganalysis via convolutional neural networks. In *IS&T/SPIE Electronic Imaging*, pages 94090J–94090J. International Society for Optics and Photonics, 2015.

[100] X. Rolland-Neviere, G. Doerr, and P. Alliez. Triangle surface mesh watermarking based on a constrained optimization framework. *IEEE Transactions on Information Forensics and Security*, 9(9) :1491–1501, 2014.

[101] P. Rondao Alface and B. Macq. From 3D mesh data hiding to 3D shape blind and robust watermarking : A survey. In Y.Q. Shi, editor, *Transactions on Data Hiding and Multimedia Security II*, volume 4499, pages 91–115. Springer, 2007.

[102] P. Rondao Alface, B. Macq, and F. Cayre. Blind and robust watermarking of 3D models : How to withstand the cropping attack ? In *IEEE International Conference on Image Processing*, volume 5, pages 465–468, 2007.

[103] J. Rossignac. Edgebreaker : Connectivity compression for triangle meshes. *IEEE Transactions on Visualization and Computer Graphics*, 5(1) :47–61, 1999.

[104] G. Sansoni, M. Trebeschi, and F. Docchio. State-of-the-art and applications of 3D imaging sensors in industry, cultural heritage, medicine, and criminal investigation. *Sensors*, 9(1) :568, 2009.

[105] W. J. Schroeder, J. A. Zarge, and W. E. Lorensen. Decimation of triangle meshes. *SIGGRAPH Computer Graphics*, 26(2) :65–70, 1992.

[106] J. Shuangshuang, R. R. Lewis, and D. West. A comparison of algorithms for vertex normal computation. *The Visual Computer*, 21(1-2) :71–82, 2005.

[107] G.J. Simmons. The prisoners' problem and the subliminal channel. In D. Chaum, editor, *Advances in Cryptology*, pages 51–67. Springer, 1984.

[108] V. Surazhsky, T. Surazhsky, D. Kirsanov, S. J. Gortler, and H. Hoppe. Fast exact and approximate geodesics on meshes. *ACM Transactions on Graphics*, 24(3) : 553–560, 2005.

[109] G. Tardos. Optimal probabilistic fingerprint codes. In *ACM Symposium on Theory of Computing*, pages 116–125, 2003.

[110] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *International Conference on Computer Vision*, pages 902–907, 1995.

[111] J. Tian. Reversible data embedding using a difference expansion. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(8) :890–896, 2003.

[112] J. Tierny, J.-P. Vandeborre, and M. Daoudi. Topology driven 3D mesh hierarchical segmentation. In *IEEE International Conference on Shape Modeling and Applications*, pages 215–220, 2007.

[113] F. Torkhani, K. Wang, and J.-M. Chassery. A curvature-tensor-based perceptual quality metric for 3D triangular meshes. *Machine Graphics & Vision*, pages 1–25, 2014.

[114] N. Tournier, W. Puech, G. Subsol, and J.-P. Pedeboy. Finding Robust Vertices for 3D Synchronization Based on Euclidean Minimum Spanning Tree. In *IS&T/SPIE Electronic Imaging*, pages 78640W–78640W. International Society for Optics and Photonics, 2011.

[115] Y.-Y. Tsai. An efficient 3D information hiding algorithm based on sampling concepts. *Multimedia Tools and Applications*, pages 1–17, 2015.

[116] S.-C. Tu, W.-K. Tai, M. Isenburg, and C.-C. Chang. An improved data hiding approach for polygon meshes. *The Visual Computer*, 26(9) :1177–1181, 2010.

[117] F. Uccheddu, M. Corsini, and M. Barni. Wavelet-based blind watermarking of 3D models. In *Workshop on Multimedia and Security*, MM& Sec '04, pages 143–154. ACM, 2004.

[118] B. Vasic and B. Vasic. Simplification resilient LDPC-coded sparse-QIM watermarking for 3D-meshes. *IEEE Transactions on Multimedia*, 15(7) :1532–1542, 2013.

[119] J. Vollmer, R. Mencl, and H. Müller. Improved laplacian smoothing of noisy surface meshes. In *Computer Graphics Forum*, pages 131–138, 1999.

[120] S. Voloshynovskiy, S. Pereira, V. Iquise, and T. Pun. Attack modelling : Towards a second generation watermarking benchmark. *Signal Processing*, 81(6) :1177–1214, 2001.

[121] C.-M. Wang and Y.-M. Cheng. An efficient information hiding algorithm for polygon models. *Computer Graphics Forum*, 2005.

[122] C.-M. Wang and P.-C. Wang. Steganography on point-sampled geometry. *Computers and Graphics*, 30(2) :244 – 254, 2006. ISSN 0097-8493.

[123] K. Wang, G. Lavoué, F. Denis, and A. Baskurt. A comprehensive survey on three-dimensional mesh watermarking. *IEEE Transactions on Multimedia*, 10(8) : 1513–1527, 2008.

[124] K. Wang, G. Lavoue, F. Denis, and A. Baskurt. Hierarchical watermarking of semi-regular meshes based on wavelet transform. *IEEE Transactions on Information Forensics and Security*, 3(4) :620–634, 2008.

[125] K. Wang, G. Lavoué, F. Denis, and A. Baskurt. Robust and Blind Watermarking of Polygonal Meshes Based on Volume Moments. Technical Report RR-LIRIS-2009-001, LIRIS, 2009.

[126] K. Wang, M. Luo, A.G. Bors, and F. Denis. Blind and robust mesh watermarking using manifold harmonics. In *IEEE International Conference on Image Processing (ICIP)*, pages 3657–3660, 2009.

[127] M.-S. Wang and W.-C. Chen. A majority-voting based watermarking scheme for color image tamper detection and recovery. *Computer Standards & Interfaces*, 29 (5) :561–570, 2007.

[128] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment : from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4) :600–612, 2004.

[129] K. Weiler. Edge-based data structures for solid modeling in curved-surface environments. *IEEE Computer Graphics and Applications*, 5(1) :21–40, 1985.

[130] J. Wu and L. Kobbelt. Efficient spectral watermarking of large meshes with orthogonal basis functions. *The Visual Computer*, 21(8-10) :848–857, 2005.

[131] T. Yamasaki, Y. Nakai, and K. Aizawa. An object-based non-blind watermarking that is robust to non-linear geometrical distortion attacks. In *IEEE International Conference on Image Processing (ICIP)*, pages 3669–3672, 2009.

[132] H. Yang, X. Sun, and G. Sun. A high-capacity image data hiding scheme using adaptive LSB substitution. *Radioengineering*, 18(4) :509, 2009.

[133] Y. Yang. *Information Analysis for Steganography and Steganalysis in 3D Polygonal Meshes*. PhD thesis, Durham University, 2013.

[134] Y. Yang and I. Ivrissimtzis. Mesh discriminative features for 3D steganalysis. *Transactions on Multimedia Computing, Communications, and Applications*, 10 (3) :27 :1–27 :13, 2014.

[135] Y. Yang, N. Peyerimhoff, and I. Ivrissimtzis. Linear correlations between spatial and normal noise in triangle meshes. *IEEE Transactions on Visualization and Computer Graphics*, 19(1) :45–55, 2013.

[136] Y. Yang, R. Pintus, H. Rushmeier, and I. Ivrissimtzis. A steganalytic algorithm for 3D polygonal meshes. In *IEEE International Conference on Image Processing (ICIP)*, pages 4782–4786, 2014.

[137] B.-L. Yeo and M.M. Yeung. Watermarking 3D objects for verification. *IEEE Computer Graphics and Applications*, 19(1) :36–45, 1999.

[138] S. Zafeiriou, A. Tefas, and I. Pitas. Blind robust watermarking schemes for copyright protection of 3D mesh objects. *IEEE Transactions on Visualization and Computer Graphics*, 11(5) :596–607, 2005.

[139] J. Zhang, C. Zheng, and X. Hu. Triangle mesh compression along the hamiltonian cycle. *Visual Computer*, 29(6-8) :717–727, 2013.

[140] X. Zhang and S. Wang. Statistical fragile watermarking capable of locating individual tampered pixels. *IEEE Signal Processing Letters*, 14(10) :727–730, 2007.

Abstract

This thesis addresses issues relating to the protection of 3D object meshes. For instance, these objects can be created using CAD tool developed by the company STRATEGIES. In an industrial context, 3D meshes creators need to have tools to verify the integrity of their meshes, or check permissions for 3D printing for example. In this context we study data hiding on 3D meshes. This approach allows us to insert information in a secure and imperceptible way in a mesh. This may be an identifier, a meta-information or a third-party content, for instance, in order to transmit secretly a texture. Data hiding can address these problems by adjusting the trade-off between capacity, imperceptibility and robustness. Generally, data hiding methods consist of two stages, the synchronization and the embedding. The synchronization stage consists of finding and ordering available components for insertion. One of the main challenges is to propose an effective synchronization method that defines an order on mesh components. In our work, we propose to use mesh vertices, specifically their geometric representation in space, as basic components for synchronization and embedding. We present three new synchronisation methods based on the construction of a Hamiltonian path in a vertex cloud. Two of these methods jointly perform the synchronization stage and the embedding stage. This is possible thanks to two new high-capacity embedding methods (from 3 to 24 bits per vertex) that rely on coordinates quantization. In this work we also highlight the constraints of this kind of synchronization. We analyze the different approaches proposed with several experimental studies. Our work is assessed on various criteria including the capacity and imperceptibility of the embedding method. We also pay attention to security aspects of the proposed methods.

keywords: Data-hiding, 3D mesh, synchronization, high capacity, security.

Résumé

Cette thèse aborde les problèmes liés à la protection de maillages d'objets 3D. Ces objets peuvent, par exemple, être créés à l'aide d'outil de CAD développés par la société STRATEGIES. Dans un cadre industriel, les créateurs de maillages 3D ont besoin de disposer d'outils leur permettant de vérifier l'intégrité des maillages, ou de vérifier des autorisations pour l'impression 3D par exemple. Dans ce contexte nous étudions l'insertion de données cachées dans des maillages 3D. Cette approche permet d'insérer de façon imperceptible et sécurisée de l'information dans un maillage. Il peut s'agir d'un identifiant, de méta-informations ou d'un contenu tiers, par exemple, pour transmettre de façon secrète une texture. L'insertion de données cachées permet de répondre à ces problèmes en jouant sur le compromis entre la capacité, l'imperceptibilité et la robustesse. Généralement, les méthodes d'insertion de données cachées se composent de deux phases, la synchronisation et l'insertion. La synchronisation consiste à trouver et ordonner les éléments disponibles pour l'insertion. L'un des principaux challenges est de proposer une méthode de synchronisation 3D efficace qui définit un ordre sur les composants des maillages. Dans nos travaux, nous proposons d'utiliser les sommets du maillage, plus précisément leur représentation géométrique dans l'espace comme composants de base pour la synchronisation et l'insertion. Nous présentons donc trois nouvelles méthodes de synchronisation de la géométrie des maillages basées sur la construction d'un chemin hamiltonien dans un nuage de sommets. Deux de ces méthodes permettent de manière conjointe de synchroniser les sommets et de cacher un message. Cela est possible grâce à deux nouvelles méthodes d'insertion haute capacité (de 3 à 24 bits par sommet) qui s'appuient sur la quantification des coordonnées. Dans ces travaux nous mettons également en évidence les contraintes propres à ce type de synchronisation. Nous discutons des différentes approches proposées dans plusieurs études expérimentales. Nos travaux sont évalués sur différents critères dont la capacité et l'imperceptibilité de la méthode d'insertion. Nous portons également notre attention aux aspects sécurité des méthodes.

mots-clés : Insertion de données cachées, maillage 3D, synchronisation, haute capacité, sécurité.