Sécurisation des maillages 3D pour l'industrie de la chaussure et la maroquinerie

Sebastien Beugnon

THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En informatique

École doctorale I2S – Information, Structures, Systèmes

Unité de recherche LIRMM – Laboratoire d'Informatique, de Robotique et de Micro-électronique de Montpellier

Sécurisation des maillages 3D pour l'industrie de la chaussure et de la maroquinerie

Présentée par Sébastien BEUGNON

Le 7 novembre 2019

Sous la direction de William PUECH

Devant le jury composé de

Géraldine MORIN, Prof. des Universités, ENSEEITH / IRIT, Toulouse	Président
Guillaume LAVOUE, MCF HDR, INSA / LIRIS, Lyon	Rapporteur
Luce MORIN, Prof. des Universités, INSA / IETR, Rennes	Rapporteur
Jean-Marc CHASSERY, Directeur de Recherche, CNRS / GIPSA-lab, Grenoble	Examinateur
Stéphane BESSY, MCF HDR, Univ. Montpellier / LIRMM, Montpellier	Examinateur
William PUECH, Prof. des universités, Univ. Montpellier / LIRMM, Montpellier	Directeur de thèse
Jean-Pierre PEDEBOY, PDG, STRATEGIES, Rungis	Invité

UNIVERSITÉ DE MONTPELLIER

Remerciements

Tout d'abord, je souhaite remercier les membres du jury Stéphane BESSY, Jean-Marc CHASSERY, Guillaume LAVOUÉ, Géraldine MORIN et Luce MORIN pour l'intérêt porté à mon travail, et en particulier Luce MORIN et Guillaume LAVOUÉ pour avoir accepté d'être rapporteurs et pour avoir fourni des retours ayant grandement contribué à l'amélioration de ce manuscrit de thèse.

Je voudrais remercier tout particulièrement mon directeur de thèse William PUECH de m'avoir offert cette opportunité dans le monde de la recherche. Je le remercie également de m'avoir encadré et dirigé pendant la durée de ma thèse et d'avoir été aussi impliqué dans mes travaux, ainsi que pour ses très nombreuses relectures et corrections. Ses conseils m'ont permis de mieux comprendre et de m'adapter aux enjeux de la recherche.

Je souhaite aussi exprimer ma gratitude envers la société STRATEGIES, et plus précisément à Jean-Pierre PEDEBOY, pour le financement de ma thèse à travers une collaboration CIFRE entre l'entreprise et le LIRMM. Son intérêt pour les innovations, ses perspectives industrielles et son ouverture aux problématiques dans le domaine de la sécurité multimédia ont permis de cerner des besoins et de nouveaux challenges pour mes travaux de recherches.

Je souhaite remercier les personnes qui m'ont permis d'avancer dans mon travail. Je remercie Vincent ITIER de m'avoir conseillé au début de mes travaux et pour nos discussions de recherche. Je remercie Silvère GAUTHIER d'avoir été un collègue de bureau avec lequel nous avons échangé pendant des heures des conversations électriques sur nos sujets de thèse respectifs et sur l'informatique, la 3D et la programmation. Je remercie tout particulièrement Pauline PUTEAUX d'avoir été une collègue de bureau à l'écoute, une collaboratrice consciencieuse et dynamique sur nos projets en commun, mais aussi une amie.

Plus généralement,je souhaite remercier tous les membres de l'équipe ICAR, permanents comme doctorants, pour toutes les aides et les conseils que j'ai obtenus auprès d'eux. En particulier, je les remercie pour la bonne ambiance, le plaisir de venir au laboratoire dans cet environnement exceptionnel, où les pauses-café sont le lieu de discussions inimaginables et les légendaires C&C.

Finalement, je souhaite remercier ma famille qui a toujours été à mes côtés. Je souhaite remercier mon oncle pour son aide sur la correction de ce manuscrit.

Je remercie particulièrement mes parents qui m'ont toujours encouragé à poursuivre les études qui me passionnent et qui m'ont soutenu en toutes circonstances.

Table des matières

Introduction

Contexte de la thèse

Cette thèse est le fruit d'une collaboration dans le cadre d'un contrat CIFRE entre la société STRATEGIES [1] et l'équipe ICAR (Image & Interaction) du LIRMM (Laboratoire d'Informatique, de Robotique et de Micro-électronique de Montpellier), Université de Montpellier, CNRS. La société STRATEGIES commercialise une suite logicielle du nom de *Roman CAD Software* à l'intention des créateurs dans l'industrie de la chaussure, la maroquinerie et les matériaux souples. La figure 1 présente un exemple d'utilisation du logiciel RCS 3D Last permettant de construire une valise à partir de courbes définissant sa forme.

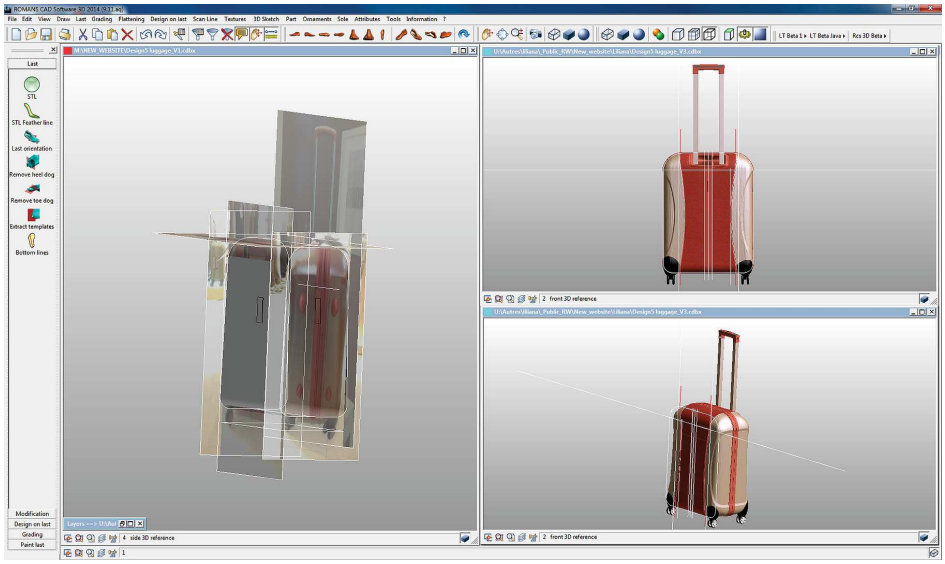


FIGURE 1 – Illustration de la suite *Roman CAD Software*.

Les solutions de conception assistée par ordinateur (CAO) proposées par STRATEGIES offrent des outils avancés pour la conception de chaussures, de sacs de luxe ou d'ameublements utilisant des matériaux souples. Ces outils permettent d'accélérer le processus de création de nouveaux objets de manière numérique jusqu'à la production du produit final en réduisant le temps total de 300 jours à 90 jours environ. Pour modéliser et concevoir des chaussures, les *designers* se servent de la forme interne de la chaussure. Cette forme peut être créée numériquement ou bien physiquement pour être ensuite scannée. Les outils de STRATEGIES permettent également, à partir d'une forme, de concevoir un modèle de chaussure complet en proposant l'ajout de coutures, de textures, de couleurs ou d'éléments propres aux chaussures. Toutes ces informations sont ensuite stockées

1. https ://www.romans-cad.com

avec l'objet 3D produit au sein d'un système de gestion des données techniques (*Product Data Management* ou PDM) proposé par STRATEGIES. Ce système de gestion permet de rendre accessibles toutes les données techniques tout en les sécurisant et en surveillant leur accès. Une fois qu'un modèle a été produit par un artiste et validé pour la production, le modèle 3D peut ensuite être transmis aux usines de production. L'avantage indéniable de la modélisation numérique 3D est de permettre aux artistes de profiter d'un rendu en temps réel de leurs créations et de leurs modifications. Grâce aux solutions proposées par STRATEGIES, les sociétés employant ces outils possèdent un autre avantage qui est de concevoir beaucoup plus rapidement leurs prototypes numériquement et d'imprimer ces derniers grâce aux dernières technologies d'impression 3D. À travers ces logiciels, les objets 3D sont de haute qualité. Ces derniers possèdent un grand nombre de points et d'attributs supplémentaires comme les couleurs, les textures, ou bien les matériaux, le tout dans un format propriétaire. Les travaux de cette thèse s'inscrivent dans la suite de nombreux partenariats de recherche entre la société STRATEGIES et l'équipe ICAR du LIRMM, ainsi que de trois thèses soutenues durant ces 15 dernières années [2, 63, 131]. Au cours de ces 15 ans de collaboration, des recherches ont été menées sur l'aide au découpage automatique de pièces, la numérisation de formes 3D et sur l'insertion robuste ou haute-capacité de données cachées au sein de maillages 3D [3, 64, 71].

Applications

Aujourd'hui, les objets 3D deviennent une part de plus en plus importante des médias numériques visuels, au travers des logiciels de CAO, mais aussi de l'imagerie médicale, des simulations, des effets spéciaux, des applications pour le patrimoine culturel ou des jeux vidéos par exemple. Ces objets 3D peuvent être représentés de différentes manières à l'aide de maillages, de surfaces implicites, de primitives géométriques ou de voxels. Les maillages 3D sont les représentations les plus couramment utilisées qui approximent la surface d'un objet 3D selon un ensemble de points et de polygones. De plus, de par leur simplicité d'utilisation, ils sont notamment utilisés dans les systèmes d'acquisition 3D devenant plus accessibles, ainsi que dans l'impression 3D qui a pris une envergure nettement plus importante ces dernières années.

Les travaux effectués dans les domaines des images ou des vidéos présentent l'insertion de données cachées, le chiffrement sélectif et le partage de secret comme des solutions intéressantes répondant aux différents problèmes de la sécurité multimédia. L'insertion de données cachées 3D permet d'insérer un message secret au sein d'un objet 3D de telle sorte que, statistiquement, l'insertion soit imperceptible. Le chiffrement sélectif 3D préserve le format des objets tout en assurant la confidentialité visuelle de leur contenu. Enfin, le partage de secret permet de construire des protocoles pour le contrôle d'accès aux données de manière collaborative tout en proposant un système de redondance de l'information sécurisée pour assurer un service en continu.

Ainsi, pour la société STRATEGIES les applications possibles de la protection 3D sont multiples. L'insertion de données cachées peut fournir un outil de traçabilité des objets 3D de leur client et ainsi donner une aide à l'identification de fuites. Un autre cas d'applications comme présenté dans les thèses précédentes est de permettre l'enrichissement de contenu en embarquant des logos, des textures ou bien des données techniques dans l'objet 3D. Le chiffrement sélectif quant à lui peut fournir une couche supplémentaire de protection aux objets 3D en sécurisant les données géométriques, en restreignant la visualisation ou interdisant l'exportation sous des formats standards pour l'impression. Cette approche de protection peut également répondre aux problématiques de transmis-

sion afin de sécuriser les échanges entre une société créatrice d'objets 3D et une entreprise de production. Le partage de secret permet de construire un système de redondance des données techniques pour assurer un service sécurisé en continu.

Dans ces travaux de thèse, nous développons de nouveaux systèmes pour sécuriser les objets 3D, en chiffrant sélectivement leur géométrie. Nous créons également de nouvelles méthodes en appliquant les principes du partage de secret aux objets 3D permettant de reconstruire un objet 3D en haute qualité à partir d'un sous-ensemble d'objets 3D chiffrés sélectivement et distribués aux collaborateurs ou à des serveurs de stockage sécurisé. De plus, nous proposons d'améliorer notre approche de partage d'objet 3D secret en ajoutant une structure hiérarchique des collaborateurs afin de rendre plus ou moins accessible la reconstruction du contenu en haute qualité. Enfin, nous étudions la problématique de la confidentialité visuelle des objets 3D sélectivement chiffrés par nos approches.

Plan

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

Dans la première partie, nous présentons un état de l'art dans le domaine de la sécurité multimédia en particulier. Nous présentons les objets 3D, l'insertion de données cachées, le chiffrement sélectif et le partage de secret dans le chapitre 1. Dans le chapitre 2, nous détaillons les méthodes et approches existantes de l'état de l'art pour le partage de secret pour les supports multimédia comme les images et les objets 3D ainsi que les aspects hiérarchiques qui peuvent être ajoutés à ces mêmes méthodes.

Dans la seconde partie de ce manuscrit, nous présentons nos contributions. Dans un premier temps, dans le chapitre 3, nous proposons une nouvelle méthode pour le chiffrement sélectif d'objets 3D en protégeant le contenu géométrique selon les besoins des utilisateurs. À partir de notre approche de chiffrement sélectif, nous proposons dans le chapitre 4 une nouvelle méthode de partage d'objet 3D secret entre plusieurs collaborateurs. Reprenant nos travaux sur le partage d'objet 3D, nous proposons dans le chapitre 5 une nouvelle méthode intégrant les aspects hiérarchiques étudiés au partage d'objet 3D secret. Enfin, dans le chapitre 6, nous analysons, à l'aide de métriques objectives et d'évaluations subjectives, l'évolution des objets 3D chiffrés sélectivement et proposons une nouvelle métrique de confidentialité.

Finalement, nous concluons ce manuscrit avec un bilan de nos différentes contributions et proposons des perspectives sur la protection des objets 3D et l'analyse des objets 3D chiffrés.

Première partie

État de l'art

Chapitre 1

Sécurité multimédia

Sommaire

1.1 Introduction

Avec l'essor des échanges de données et l'évolution des technologies, les contenus multimédia ont pris une place importante dans le trafic mondial de données et sur les applications. Ainsi les données multimédia, à savoir les images, les vidéos, les sons et les objets 3D, sont régulièrement utilisées dans de nombreux domaines et échangées sur les réseaux sociaux. De nos jours, les objets 3D sont utilisés dans un grand nombre d'applications en particulier pour le médical, la simulation, les jeux vidéo, l'animation ou encore les effets spéciaux. La consommation d'objets 3D par le grand public devient un marché lucratif pouvant prendre la forme de plateformes de téléchargement de modèles sous tous formats. Avec cette évolution toujours plus rapide, les créateurs et les propriétaires, au vu des coûts de production, voient leurs créations comme des biens financiers qu'il est nécessaire de protéger du piratage ou des copies illicites.

Dans ce chapitre, en section 1.2 nous introduisons le principal type de données traitées durant cette thèse, à savoir les objets 3D. Dans la section 1.3, nous décrivons le principe d'insertion de données cachées et ses diverses propriétés. En section 1.4, nous abordons le principe de chiffrement sélectif. Enfin en section 1.5, nous présentons de manière générale le principe de partage de secret en décrivant les principales approches connues.

1.2 Objets 3D

Les objets 3D numériques sont décrits mathématiquement par leur surface ou leur volume selon différentes représentations numériques. Les objets 3D peuvent représenter deux types d'objets, à savoir les objets 3D synthétiques créés numériquement et les objets 3D issus du monde réel et ayant été numérisés à l'aide de systèmes d'acquisition. Les objets 3D synthétiques sont produits par des designers, des infographistes, ou des artistes à l'aide d'outils de conception assistée par ordinateur (CAO) qui permettent de générer des objets de très bonne qualité. Quant aux objets 3D numérisés acquis à l'aide de systèmes d'acquisition 2D ou 3D, ces derniers peuvent varier au niveau des informations acquises selon un large spectre de systèmes d'acquisition. Ainsi, en plus de la géométrie ces systèmes peuvent fournir la couleur, la texture, la réflectance, la rugosité ainsi que d'autres propriétés liées aux matériaux de l'objet 3D réel. Sansoni *et al.* ont proposé un état de l'art détaillant les différentes méthodes d'acquisition [113].

En section 1.2.1, nous abordons les différentes représentations des objets 3D alors qu'en section 1.2.2, nous présentons différentes métriques utilisées pour évaluer la qualité des objets 3D.

1.2.1 Représentation

Dans cette section, nous introduisons les différentes représentations possibles des objets 3D. En fonction de leur origine et leur utilisation, certaines représentations sont privilégiées pour, par exemple, simplifier le processus de création. D'autres représentations sont utilisées pour stocker un minimum d'informations afin de permettre la visualisation ou l'évaluation qualitative des objets 3D issus d'un système d'acquisition. Nous distinguons deux grandes catégories de représentations, à savoir les représentations continues et les représentations discrètes. Pour chacune de ces représentations, deux types d'objets 3D sont rencontrés, à savoir les objets surfaciques produisant juste une information d'enveloppe de l'objet 3D et les objets volumiques décrivant également l'intérieur d'un objet 3D pouvant prendre en compte les parties vides, ainsi que les aspects multi-matériaux.

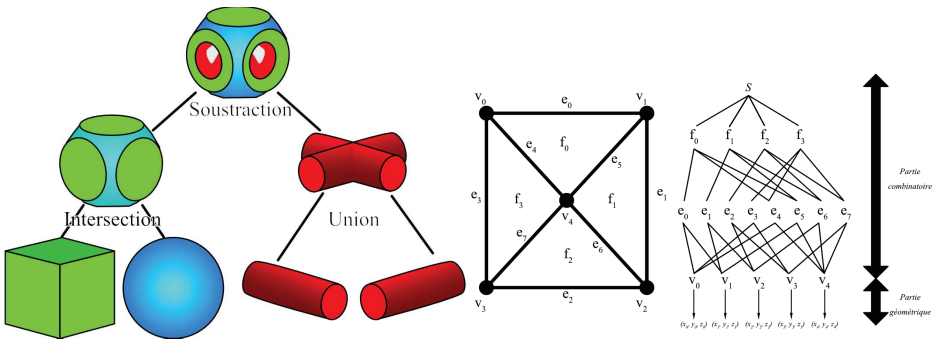
Dans un premier temps, nous expliquons comment les objets 3D sont définis par des représentations continues au sein des outils de CAO. Puis, dans un second temps, nous présentons les représentations discrètes des objets 3D et nous détaillons les représentations basées sur des maillages.

Représentations continues

Les représentations continues consistent à stocker des primitives géométriques définies mathématiquement pour décrire la surface ou le volume d'un objet 3D. Les deux principales représentations sont le modèle volumique CSG (Constructive Solid Geometry) [52] et le modèle surfacique B-Rep (Boundary Representation) [123] permettant de construire un objet 3D par une combinaison de volumes et de surfaces, respectivement :

— Modèle volumique CSG (Constructive Solid Geometry) : les objets 3D représentés selon un modèle CSG, ou aussi appelé arbre CSG [52], sont construits à l'aide de primitives géométriques comme des pavés, des boules, des cylindres ou des cônes. Les opérations booléennes sont définies entre ces primitives afin de représenter le processus de reconstruction. Ainsi intersection, union ou soustraction permettent de définir le processus de construction sous la forme d'un arbre. Les feuilles de l'arbre CSG correspondent aux primitives géométriques. Les feuilles sont connectées par paire à un noeud qui correspond à une opération booléenne appliquée entre les deux primitives géométriques représentées, comme illustré en figure 1.1a.

— Modèle surfacique B-Rep (Boundary Representation) : le modèle B-Rep est une composition de surfaces définies par des primitives géométriques surfaciques comme des plans, des cylindres, des sphères ou bien des surfaces plus complexes. À l'aide de définitions paramétriques, ces primitives sont représentées par un ensemble de paramètres et d'équations mathématiques. En plus de ces primitives géométriques, le modèle B-Rep stocke des contours correspondant aux intersections entre les différentes primitives sous la forme de courbes paramétriques. Stroud présente le modèle B-Rep comme un solide qui est entièrement représenté par son bord [123]. Comme illustré en figure 1.1b, les surfaces des différentes primitives géométriques sont ainsi délimitées par les contours pour former la surface continue de l'objet 3D.



(a) Modèle CSG (b) Modèle B-Rep

FIGURE 1.1 – Exemples de représentation continue : a) Modèle CSG, b) Modèle B-Rep.

Représentations discrètes

Les représentations discrètes permettent de représenter de manière éparse la surface ou le volume d'un objet 3D. Les représentations discrètes sont souvent privilégiées afin de

transmettre des objets 3D numériques sans en fournir la représentation continue complète. La discrétisation est un processus permettant de produire, à partir d'une représentation continue ou d'un nuage de points 3D produit par un système d'acquisition, une surface approximant la forme de l'objet 3D réel. La surface d'un objet 3D peut être représentée par la création d'une connectivité entre différents points 3D sous la forme d'arêtes formant des polygones qui approximent la surface de l'objet 3D représenté. Les systèmes d'acquisition d'objets 3D réels fournissent des nuages de points 3D contenant souvent des erreurs telles qu'un échantillonnage bruité, non uniforme ou des points irréalistes. De plus, certaines conditions empêchent l'acquisition complète d'un objet 3D réel par le fait de ne pas pouvoir bouger ou tourner autour de l'objet réel comme c'est le cas avec les œuvres d'art par exemple. Ainsi il existe une large gamme de représentations discrètes permettant d'approximer la surface ou le volume d'un objet 3D :

— Nuage de points 3D : le nuage de points 3D est la première structure permettant de construire des représentations discrètes. Il s'agit d'un échantillonnage de la forme et de la surface d'un objet 3D ou d'une scène 3D. En raison de sa simplicité d'implémentation, le nuage de points 3D est souvent utilisé dans les systèmes d'acquisition ou la télédétection par laser telle que le LiDAR (*Light Detecting And Radar* [27]). Un nuage de points 3D $\mathcal{P}$ est défini par un ensemble de n points :

$$\mathcal{P} = \{p_1, \ldots, p_n\}, p_i \in \mathbb{R}^3, 1 \leq i \leq n. \tag{1.1}$$

Dans les systèmes d'acquisition, chaque point 3D est défini par sa position selon le repère du système utilisé :

$$p_i = (x_i, y_i, z_i). \tag{1.2}$$

Ces systèmes d'acquisition fournissent généralement, en plus de la position, une normale en chaque point :

$$p_i = (x_i, y_i, z_i, nx_i, ny_i, nz_i). \tag{1.3}$$

De plus, la couleur de l'objet 3D réel au niveau du point acquis peut être récupérée et stockée selon le modèle de couleur RVB (Rouge/Vert/Bleu) :

$$p_i = (x_i, y_i, z_i, nx_i, ny_i, nz_i, r_i, g_i, b_i). \tag{1.4}$$

— Maillage (Mesh) : une fois le nuage de points 3D acquis ou généré, ce dernier est utilisé pour reconstruire une surface fermée. Le maillage 3D est donc une approximation d'une surface continue. Une forte densité de sommets permet d'assurer une approximation de bonne qualité, mais entraîne un fort coût de stockage. Ainsi, il existe un compromis où la qualité de l'approximation dépend de la densité des sommets et la capacité de stockage de ces informations. L'approximation de la surface consiste en une étape de maillage qui aura pour but de fournir une connectivité au nuage de points 3D et de déterminer sa topologie. Si l'échantillonnage est uniformément réparti, alors la surface continue de l'objet 3D réel peut être approximée de manière précise. La création d'une surface à partir de nuages de points 3D peut être basée sur une étape de triangulation, avec par exemple la triangulation de Delaunay [28] ou à partir d'une reconstruction de Poisson par Kazhdan *et al.* [77]. Berger *et al.* ont proposé un état de l'art des techniques de maillage à partir de nuages de points 3D [10].

Botsch *et al.* [22] définissent un maillage 3D $\mathcal{M}$ comme deux composantes, à savoir une composante géométrique $\mathcal{P}$ et une composante de connectivité. Cette dernière

composante peut être représentée sous la forme d'un graphe avec l'ensemble de sommets :

$$\mathcal{V} = \{v_1, \ldots, v_i, \ldots v_{\mathrm{V}}\}, 1 \leq i \leq \mathrm{V}, \tag{1.5}$$

qui forment un ensemble de facettes $\mathcal{F}$ consistant à des triplets d'indice de sommets :

$$\mathcal{F} = \{f_1, \ldots, f_j, \ldots, f_{\mathrm{F}}\}, 1 \leq j \leq \mathrm{F}, \tag{1.6}$$

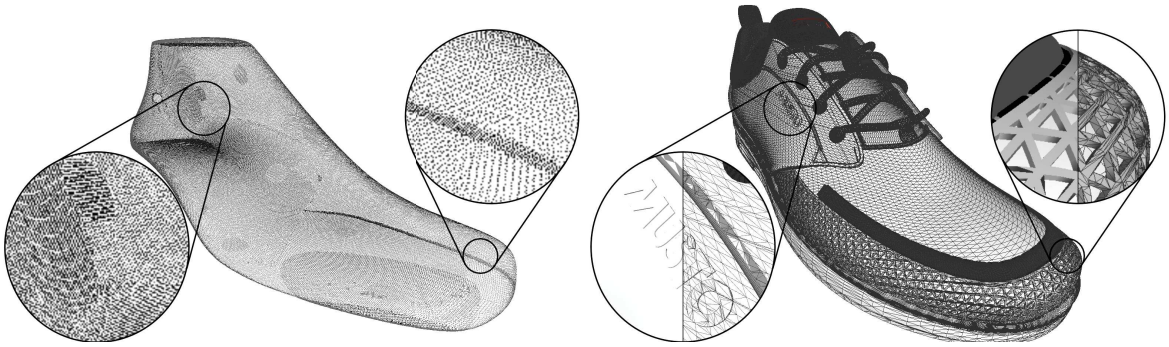
où V correspond au nombre de sommets. La composante géométrique est représentée par un nuage de points $\mathcal{P}$ associant chaque point p_i, une position 3D, à chaque sommet $v_i \in \mathcal{V}$:

$$\mathcal{P} = \{p_1, \ldots, p_{\mathrm{V}}\}, p_i \leftarrow p(v_i) = \begin{pmatrix} x(v_i) \\ y(v_i) \\ z(v_i) \end{pmatrix} \in \mathbb{R}^3. \tag{1.7}$$

Dans certains cas, il est plus efficace de représenter la connectivité du maillage avec des arêtes telles que :

$$\mathcal{E} = \{e_1, \ldots, e_{\mathrm{E}}\}, e_i \in \mathcal{V} \times \mathcal{V}. \tag{1.8}$$

Ainsi, un maillage 3D est noté $\mathcal{M} = (\mathcal{P}, \mathcal{V}, \mathcal{F}, \mathcal{E})$ et ses composantes : l'ensemble de points $\mathcal{P}$, l'ensemble de sommets $\mathcal{V}$, l'ensemble de facettes $\mathcal{F}$ et l'ensemble d'arêtes $\mathcal{E}$. De plus, la plupart des méthodes de traitement utilisent des maillages triangulaires, où $f \in \mathcal{V}^3$, car ces derniers profitent de propriétés avantageuses que cela soit pour le rendu, le traitement ou l'évaluation.



(a) (b)

FIGURE 1.2 – Représentations discrètes d'un objet 3D : a) Nuage de points 3D, b) Maillage.

La figure 1.2.a illustre un nuage de points acquis à l'aide d'un système d'acquisition d'une forme représentant l'intérieur d'une chaussure, alors que la figure 1.2.b représente un maillage 3D produit à la suite d'une discrétisation d'un modèle continu.

1.2.2 Évaluation de la qualité

Les objets 3D peuvent subir de nombreux traitements. Les principaux traitements concernent le filtrage, le remaillage, la déformation, la simplification, la réparation ainsi que la compression [22, 95]. De plus, au moment de l'acquisition, un objet 3D est également altéré [113]. Il est alors nécessaire de pouvoir estimer l'impact de ces traitements afin de comparer le maillage original et le maillage traité. Ainsi, pour quantifier les distorsions et la qualité d'un maillage, des métriques d'évaluation de qualité avec références

peuvent être utilisées. Ces métriques sont rangées en deux catégories, celles basées sur des mesures statistiques que nous notons métriques *objectives* et celles qui sont corrélées au système visuel humain (SVH) que nous notons métriques *subjectives*. La première catégorie est fortement utilisée en raison de leur intégration dans de nombreuses applications, la distance de Hausdorff (HD) [5], la racine carrée de l'erreur quadratique moyenne RMSE (*Root Mean Square Error*) [38], le PSNR (*Peak Signal-to-Noise Ratio*) appartiennent à cette première catégorie. Ces dernières années, les métriques perceptuelles ont eu un fort engouement notamment par le fait qu'elles cherchent à être corrélées au SVH en se reposant sur des évaluations subjectives. Par exemple, la métrique MSDM2 (*Mesh Structural Distortion Measure 2*), proposée par Lavoué [84], se base sur une évaluation multirésolution de la courbure d'un maillage 3D. Wang *et al.* [143], Vása et Rus [137] et Torkhani [130] ont également proposé de nouvelles métriques : FMPD (*Fast Mesh Perceptual Distance*) basée sur la rugosité locale, DAME (*Dihedral Angle Mesh Error*) basée sur la variation des angles dièdres et TPDM (*Tensor-based Perceptual Distance Measure*) basée sur les tenseurs des courbures principales du maillage 3D.

Distance de Hausdorff

Pour comparer deux surfaces d'objets 3D, il est possible d'utiliser la distance de Hausdorff (HD). Cignoni *et al.* [38] ont proposé une première version pour évaluer la qualité d'un maillage. Ensuite, Aspert *et al.* [5] ont réduit le temps d'exécution avec une méthode plus efficace. Grâce à l'échantillonnage de la géométrie des objets 3D par un maillage, ces méthodes fournissent une bonne approximation de la distance. La distance de Hausdorff se base sur la distance entre un point p appartenant à une surface $\mathcal{S}$ et un point p' appartenant à l'autre surface comparée $\mathcal{S}'$:

$$d(p,\mathcal{S}') = \min_{p'\in\mathcal{S}'} \|p-p'\|_2, \tag{1.9}$$

où $\|p-p'\|_2$ est la distance euclidienne dans $\mathbb{R}^3$. La distance entre la surface $\mathcal{S}$ et la surface $\mathcal{S}'$ est alors égale à la plus grande distance entre les points de la surface $\mathcal{S}$ et la surface $\mathcal{S}'$:

$$d(\mathcal{S},\mathcal{S}') = \max_{p\in\mathcal{S}} d(p,\mathcal{S}'). \tag{1.10}$$

Comme la distance précédente n'est pas symétrique, en général, $d(\mathcal{S},\mathcal{S}') \neq d(\mathcal{S}',\mathcal{S})$. Ainsi, pour respecter les propriétés d'une distance, la distance de Hausdorff est définie telle que :

$$\mathrm{HD}(\mathcal{S},\mathcal{S}') = \max(d(\mathcal{S},\mathcal{S}'), d(\mathcal{S}',\mathcal{S})). \tag{1.11}$$

Métrique RMSE

En 2002, Aspert *et al.* ont proposé la métrique RMSE basée sur une distance point-surface (cf. Eq (1.9)) telle que la distance de Hausdorff :

$$\mathrm{RMSE}(\mathcal{S},\mathcal{S}') = \sqrt{\frac{1}{|\mathcal{S}|}\int\int_{p\in\mathcal{S}} d(p,\mathcal{S}')^2 d\mathcal{S}}. \tag{1.12}$$

Comme cette métrique est aussi asymétrique, nous définissons une MRMSE, le maximum RMSE par :

$$\mathrm{MRMSE}(\mathcal{S},\mathcal{S}') = \max(\mathrm{RMSE}(\mathcal{S},\mathcal{S}'), \mathrm{RMSE}(\mathcal{S}',\mathcal{S})). \tag{1.13}$$

Cependant, nous nous intéressons à une autre version de la métrique RMSE approximant le résultat en calculant la distance entre deux sommets de deux maillages à partir d'un appariement connu entre les points des deux surfaces. En effet, dans ce manuscrit, les méthodes proposées ne modifient pas la connectivité des objets 3D ce qui rend possible l'analyse des altérations sommet par sommet. Du fait que l'appariement soit connu et symétrique, notre RMSE est alors symétrique. Ainsi, nous proposons d'utiliser cette approximation pouvant être calculée à partir de la position des sommets ou sur le produit scalaire des normales des sommets entre les deux maillages $\mathcal{M} = (\mathcal{V}, \mathcal{F})$ et $\mathcal{M}' = (\mathcal{V}', \mathcal{F}')$ approximant les surfaces $\mathcal{S}$ et $\mathcal{S}'$, respectivement :

$$\mathrm{RMSE}_v(\mathcal{M}, \mathcal{M}') = \sqrt{\frac{1}{\mathrm{V}} \sum_{i=1}^{\mathrm{V}} \| v_i - v'_i \|_2^2}, \tag{1.14}$$

où $v_i \in \mathcal{V}$, $v'_i \in \mathcal{V}'$ sont deux sommets appareillés, et :

$$\mathrm{RMSE}_n(\mathcal{M}, \mathcal{M}') = \sqrt{\frac{1}{\mathrm{V}} \sum_{i=1}^{\mathrm{V}} \langle n_i . n'_i \rangle^2}, \tag{1.15}$$

où $\langle . \rangle$ est le produit scalaire entre deux vecteurs.

Métrique PSNR

Le PSNR est une métrique de référence en 2D pour évaluer la qualité des images. Certains auteurs l'utilisent en 3D. Ainsi, deux versions ont été définies par Chao *et al.* [31] en 2009, une quantifiant la distorsion sur la position des sommets, l'autre la distorsion sur les normales de ces mêmes sommets. Ces deux métriques se basent sur les métriques RMSE_v et RMSE_n présentées précédemment. Ainsi le PSNR entre deux maillages $\mathcal{M}$ et $\mathcal{M}'$ est défini respectivement, comme :

$$\mathrm{PSNR}_v(\mathcal{M}, \mathcal{M}') = 20 \log_{10} \frac{\mathrm{D}_{\max}}{\mathrm{RMSE}_v(\mathcal{M}, \mathcal{M}')}, \tag{1.16}$$

$$\mathrm{PSNR}_n(\mathcal{M}, \mathcal{M}') = 20 \log_{10} \frac{\mathrm{D}_{\max}}{\mathrm{RMSE}_n(\mathcal{M}, \mathcal{M}')}, \tag{1.17}$$

où $\mathrm{D}_{\max}$ est la longueur de la diagonale de la boîte englobante du maillage de référence.

Cette métrique est une bonne approximation du PSNR pour les images appliquée aux données 3D, car elle est capable de détecter des déplacements très faibles grâce à l'appariement des points. Cependant, elle n'échantillonne pas toute la surface du maillage.

Métrique MSDM2

Le principal défaut des métriques précédentes est de ne mesurer que des distances géométriques sans prendre en compte les aspects visuels liés à la surface de l'objet 3D. Le rajout de bruit dans une zone fortement texturée du maillage (comme des cheveux) ne va pas affecter la perception pour un observateur. En revanche, l'ajout d'un bruit sur une surface plane conduit à une mauvaise appréciation du maillage 3D. Ainsi, les modifications du maillage qui ont pour but de ne pas laisser de traces doivent conserver localement les structures, comme les zones lisses. Tandis que les modifications du maillage qui visent à perturber les informations géométriques, par exemple un chiffrement, doivent déranger certaines les zones lisses. Pour répondre localement à la détection des modifications

dans les structures, une première métrique intitulée MSDM (Mesh Structural Distortion Measure), proposée par Lavoué *et al.* [81] adapte la métrique 2D SSIM (*structural similarity*) [146] aux objets 3D en remplaçant l'intensité des pixels par la courbure moyenne du maillage. Une métrique locale LMSDM entre deux fenêtres locales a et b appartenant respectivement à $\mathcal{M}$ et $\mathcal{M}'$ est définie par :

$$\mathrm{LMSDM}(a,b) = \sqrt[3]{0.4 \times \mathrm{L}(a,b)^3 + 0.4 \times \mathrm{C}(a,b)^3 + 0.2 \times \mathrm{S}(a,b)^3}, \tag{1.18}$$

où L, C et S représentent les fonctions de comparaison de courbures, de contrastes et de structures [81]. La métrique MSDM est égale à la somme de Minkovski des n_w distances locales :

$$\mathrm{MSDM}(\mathcal{M},\mathcal{M}') = \sqrt[3]{\frac{1}{n_w}\sum_{j=1}^{n_w} \mathrm{LMSDM}(a_j,b_j)^3}. \tag{1.19}$$

En 2011, Lavoué [84] a proposé une nouvelle métrique, intitulée MSDM2, présentant deux nouvelles améliorations. La première consiste à pouvoir comparer des maillages triangulaires ayant des connectivités différentes à l'aide d'une étape permettant de déterminer la correspondance entre les sommets du maillage de référence et le maillage à comparer. La seconde amélioration concerne l'évaluation de la différence visuelle par multi-résolution. En effet, de cette manière les résultats de la métrique sont plus corrélés aux évaluations subjectives. Grâce à cette approche, les nombreux scores calculés sur la surface de manière locale sont rassemblés en un seul et unique score global.

Métrique FMPD

La métrique FMPD (*Fast Mesh Perceptual Distance*) a été proposée par Wang *et al.* [143] dans le but d'évaluer la qualité visuelle des maillages 3D. Elle est aussi basée sur l'étude de la courbure du maillage et plus particulièrement de la rugosité (*roughness*) locale qui est dérivée de la courbure gaussienne. Ainsi, la rugosité locale est analysée pour les deux maillages 3D comparés. Les valeurs locales sont ensuite modulées avec attention selon les effets d'un masque visuel. Un score global de rugosité est alors calculé pour chaque maillage pour en déduire ensuite la différence en tant que distance perceptuelle.

Métrique DAME

La métrique DAME (*Dihedral Angle Measure Error*) proposée par Vása et Rus [137] s'intéresse à analyser les distorsions au niveau des angles dièdres des triangles des maillages 3D possédant la même connectivité :

$$\mathrm{DAME}(\mathcal{M},\mathcal{M}') = \frac{1}{||\Omega||}\sum_{\{t_1,t_2\}\in\Omega} ||\mathrm{D}_{t_1,t_2} - \overline{\mathrm{D}_{t_1,t_2}}|| \times m_{t1,t2} \times (w_{t_1} + w_{t_2}), \tag{1.20}$$

où Ω est l'ensemble de toutes les paires de triangles partageant une arête, t_1, t_2 des triangles, D_{t_1,t_2} l'angle dièdre entre les deux triangles t_1 et t_2 dans $\mathcal{M}$,$\overline{\mathrm{D}_{t_1,t_2}}$ l'angle dièdre entre les deux triangles t_1 et t_2 dans $\mathcal{M}'$, $m_{t1,t2}$ le coefficient de masquage visuel, w_{t_1} et w_{t_2} les termes de visibilité des deux triangles. Ainsi, la moyenne des différences entre les angles dièdres est calculée après avoir pondéré ces derniers par un coefficient de masquage visuel tel que :

$$m_{t_1,t_2} = e^{k\times \mathrm{D}_{t_1,t_2}}, \tag{1.21}$$

où $k = 7$ est choisi empiriquement par les expérimentations de Vása et Rus [137].

Le terme de *visibilité* w_{t_1} (w_{t_2}) consiste à calculer la densité de pixels représentant les triangles selon différents angles de vue. Vása et Rus ont proposé de calculer ce terme en générant des images de synthèse de l'objet 3D selon différents angles afin de compter le nombre de pixels dans chaque image représentant un triangle. Les auteurs proposent aussi une approximation de ce terme, beaucoup plus rapide, consistant à calculer le ratio entre l'aire du triangle et l'aire de la surface du maillage.

Métrique TPDM

Torkhani [130] a proposé une nouvelle métrique se basant sur la distance entre les tenseurs de courbure de deux objets 3D. Ainsi, la méthode peut à partir des valeurs *propres* (ou *eigenvalues*, l'amplitude de la courbure), mais aussi des vecteurs *propres* (ou *eigenvectors*, directions de la courbure principale) dériver une distance orientée perceptuelle. Cette métrique prend aussi en compte l'effet de masquage visuel présent dans le SVH en pondérant les distances des tenseurs locaux avec la rugosité locale qui est dérivée de la courbure. Enfin, le score global est calculé à partir d'une somme de Minkowski pondérée par l'aire totale des triangles incidents à chaque sommet.

Récapitulatif des métriques

Dans ce manuscrit, nous utilisons différentes métriques présentées dans ce chapitre afin d'évaluer la qualité d'un objet 3D entre sa version originale et sa version chiffrée, dégradée ou décompressée. La figure 1.3 illustre deux objets 3D, représentés sous forme de maillages triangulaires 3D, où le premier est l'objet 3D original et le second a été dégradé par ajout de bruit. Dans le tableau 1.1, nous résumons les valeurs obtenues pour chacune des métriques entre les deux objets 3D présentés en figure 1.3.

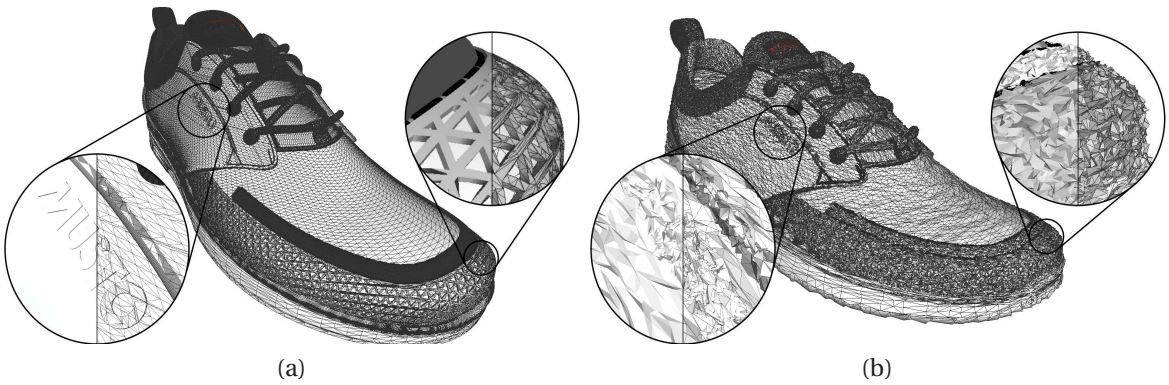


FIGURE 1.3 – Objets 3D : a) Original, b) Dégradé.

TABLEAU 1.1 – Valeurs obtenues pour chacune des métriques présentées entre l'objet 3D original et sa version dégradée présentés en figure 1.3.

Métrique	HD	$RMSE_v$	$RMSE_n$	$PSNR_v$	$PSNR_n$	MSDM2	FMPD	DAME	TPMD
Résultats	0.48831	0.54836	63.813	75.887	14.570	0.61234	0.35873	0.62611	0.88472

1.3 Insertion de données cachées

L'insertion de données cachées (IDC) est une approche permettant d'insérer de l'information supplémentaire au sein d'un support numérique sans être perceptible (visuellement ou statistiquement) tout en respectant le format initial des données. Les données insérées peuvent être par exemple un message secret, des méta-données, un identifiant, une marque ou un autre média. L'IDC consiste donc à modifier le support numérique pour l'enrichir avec une information additionnelle. Lorsque l'IDC est utilisée sur un média, ce dernier doit pouvoir être visualisé ou manipulé tout en respectant le format initial. La taille originale du fichier doit être préservée au mieux afin d'éviter un coût supplémentaire de stockage en mémoire ou de bande passante. De plus, l'augmentation de la taille d'un fichier peut être suspecte. Certaines méthodes naïves utilisent les zones de commentaires pour insérer un message, ce qui en fait ne marque pas vraiment le contenu visuel. De plus dans ce cas, un changement de format peut supprimer le message caché. Dans cette section, nous nous concentrons sur les méthodes qui modifient le contenu visuel de façon imperceptible afin d'insérer des données cachées. Dans la section 1.3.1, nous définissons les principes et les propriétés sur lesquelles repose l'IDC. En section 1.3.2, nous présentons les différentes classes d'insertion de données cachées. Enfin, nous présentons les principales méthodes d'IDC de l'état de l'art utilisant comme média des objets 3D en section 1.3.3.

1.3.1 Principes et propriétés

Comme présenté dans le livre de Cox *et al.* [41], l'insertion de données cachées a pour but de dissimuler, de façon imperceptible, des données au sein d'un support hôte. Cox *et al.* définissent le contexte de l'IDC où les méthodes possèdent deux modules principaux, à savoir l'insertion et l'extraction du message. La figure 1.4 illustre l'étape d'insertion qui est précédée d'une phase de synchronisation à partir d'une image originale. Afin d'assurer la bonne récupération du message, une étape de synchronisation est nécessaire pour obtenir le bon ordre pour l'insertion et l'extraction du message. La synchronisation peut dépendre d'une clé secrète afin d'assurer que les personnes autorisées soient les seules à pouvoir accéder au message. Ainsi dans la figure 1.4, l'insertion permet de produire une image marquée, très proche visuellement et statistiquement de l'image originale. L'étape d'insertion peut dépendre aussi de la clé secrète dans certaines méthodes afin de chiffrer le message secret.

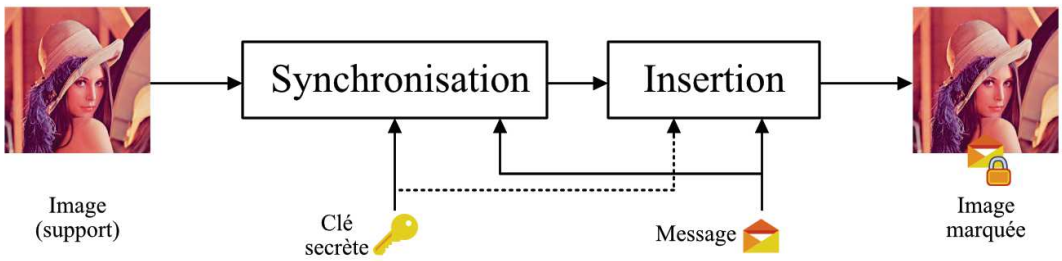


FIGURE 1.4 – Module d'insertion de données cachées précédée d'une phase de synchronisation initialisée avec une clé secrète. L'image est marquée avec un message secret.

Dans une première phase, la clé secrète est utilisée pour sécuriser l'étape de synchronisation et définir l'ordre des zones choisies pour l'insertion. En figure 1.4, il s'agit d'un

ordre sur les pixels. Une fois que la synchronisation est établie, l'insertion modifie la valeur des pixels afin de cacher le message dans l'image originale. La méthode la plus classique consiste à modifier les bits de poids faible codant les pixels. Notons que le message inséré est généralement chiffré en utilisant une clé secrète pour augmenter la sécurité de la méthode.

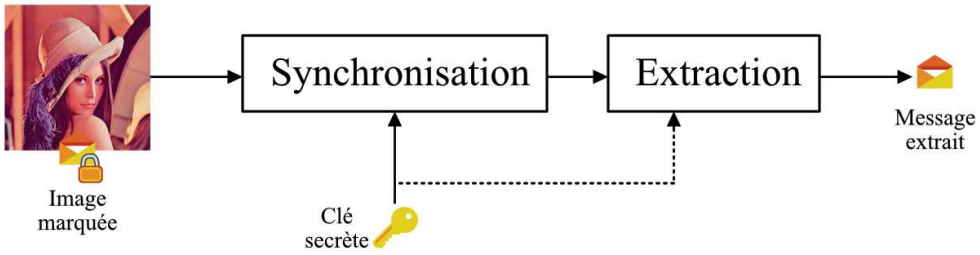
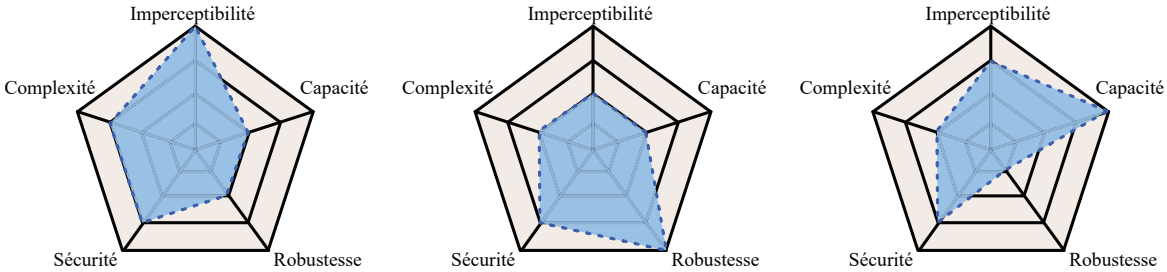


FIGURE 1.5 – Module d'extraction de données cachées. L'information est extraite à partir de l'ordre reconstruit par la clé secrète lors de la synchronisation.

Le module d'extraction est illustré en figure 1.5 et permet d'extraire le message caché du support grâce à la clé qui a été utilisée à l'insertion. L'extraction du message caché se déroule généralement en deux étapes. Une étape de synchronisation qui, comme pour l'insertion, consiste à retrouver les zones d'insertion et établir un ordre de lecture. Puis la seconde étape qui extrait les bits dans l'ordre obtenu par la phase de synchronisation. Les méthodes d'IDC doivent faire face à un compromis entre différentes propriétés telles que :

— La robustesse : la résistance de l'insertion à des traitements sur le support (transformations, filtrages) permettant de toujours récupérer le message ;
— La capacité : la quantité de bits insérables dans le support ;
— L'imperceptibilité : l'invisibilité statistique de l'insertion dans le support ;
— La sécurité : le niveau de protection du message secret ;
— La complexité : la complexité algorithmique de la méthode d'insertion et de son temps d'exécution.



(a) Stéganographie (b) Tatouage (c) Insertion haute-capacité

FIGURE 1.6 – Compromis entre les différentes propriétés d'une application d'insertion de données cachées.

En général, améliorer une de ces propriétés fait décroître les autres. Ainsi, un compromis doit être réalisé comme illustré figure 1.6, souvent entre capacité et robustesse ou robustesse et sécurité. Plus généralement, les propriétés d'une méthode d'IDC varient en fonction de la technique utilisée et du scénario d'utilisation.

1.3.2 Classification des applications

La figure 1.7 illustre les différentes classes d'IDC des applications les plus générales aux plus spécifiques. Chacune de ces classes est définie par rapport aux caractéristiques souhaitant être mises en avant.

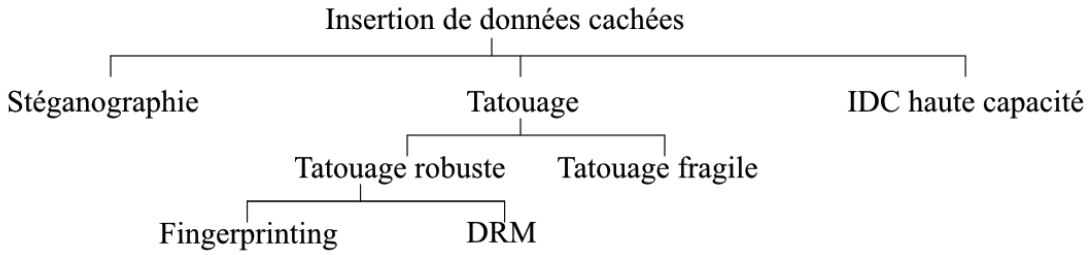


FIGURE 1.7 – Classification des méthodes d'insertion de données cachées [101].

Les techniques d'insertion jouent un rôle essentiel car elles déterminent les propriétés de la méthode d'IDC. Ainsi, l'insertion peut être :

— **Additive** : la technique consiste à insérer le message sous la forme d'un bruit. Cela nécessite le support original pour analyser la différence entre le média de support et le média marqué pour extraire le message.

— **Substitutive** : le message est inséré en remplaçant l'information redondante du support de manière à provoquer une altération la plus faible possible. Cette technique est la plus utilisée.

Stéganographie

La stéganographie est l'art de cacher un message dans un média hôte de manière imperceptible visuellement et statistiquement. Elle vise à permettre la réalisation de communication secrète. Dans le cas d'une 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 cette dernière doit être échangée. Tandis que les méthodes à clés privées/publiques dites *asymétriques* considèrent que chaque personne possède une paire de clés : une clé privée qu'il conserve et une clé publique qu'il partage. Par exemple, Alice souhaite envoyer un message à Bob. Pour faire cela, elle utilise la clé publique de Bob pour insérer son message dans un support et Bob, grâce à sa clé privée, peut extraire ou décoder le message d'Alice. Ainsi, Simmons [118] pose le problème des prisonniers, dans lequel Alice et Bob sont deux prisonniers souhaitant échanger des messages secrets. Les échanges sont contrôlés par Eve qui joue le rôle de gardienne. Si le message est chiffré ou semble suspect, alors il n'est pas remis au destinataire. Grâce à la stéganographie, Alice et Bob peuvent se transmettre sur un canal public des média servant de support pour le message secret qu'ils souhaitent se transmettre.

Tatouage

Le tatouage permet l'insertion de données cachées pouvant servir à la gestion des droits d'auteur [141]. Le tatouage peut en effet servir pour l'identification; permettant de clamer la propriété d'un média ou d'identifier l'auteur d'une fuite par sa copie du média [160]. Par conséquent, il peut aussi permettre de rendre public un document multimédia. Ainsi, le tatouage permet la dissimulation de droits d'auteur. Nous pouvons distin-

guer plusieurs catégories de méthodes présentées dans la figure 1.7, à savoir le tatouage robuste, le tatouage fragile et le *fingerprint*.

Le tatouage robuste permet de marquer un média de manière à être robuste aux attaques (modifications du média) jusqu'à ce que le support soit trop dégradé pour être réellement utilisable ou que sa valeur ne soit plus intéressante [94, 109, 142]. Ainsi, à l'aide de ce type de tatouage, l'identifiant du propriétaire peut être inséré, permettant par la suite de pouvoir vérifier la propriété du média par extraction du tatouage. Actuellement, la législation n'a pas encore tranché sur ce sujet.

Le tatouage fragile a pour but de permettre la vérification de l'authenticité ou de l'intégrité d'un média [53, 150]. Dans le cas où nous cherchons à vérifier l'authenticité d'un média, si le tatouage est absent ou dégradé alors l'authenticité du média n'est pas vérifiée et le média n'est plus de confiance. Le tatouage fragile peut servir également à détecter si le média n'a pas été modifié en y insérant une marque de manière à être facilement détériorée par les modifications faites au média.

Enfin, le *fingerprinting* est l'insertion d'un identifiant correspondant non pas au propriétaire mais à un utilisateur. Dès lors, ce tatouage permet de faire du traçage de traîtres dans le cas de piratage ou de divulgation d'information [36, 44]. L'insertion de l'identifiant caché doit être robuste afin de permettre la mise en cause des utilisateurs impliqués dans une reproduction ou une diffusion illégale.

Insertion de données cachées haute-capacité

Grâce à l'IDC, il est possible de réaliser de l'enrichissement de contenu et de l'ajout de méta-données [29]. C'est pourquoi l'insertion de données cachées haute-capacité cherche à fournir l'espace suffisant pour stocker ces nouvelles informations. Par exemple, en 3D ce stockage peut servir à embarquer les informations de texture, de *normal map* ou de couleur d'un objet 3D. Cela permet aussi d'éviter le transfert de plusieurs fichiers. Notons qu'un utilisateur standard n'ayant pas accès à la clé secrète ou pas connaissance de l'IDC peut visualiser et manipuler l'objet 3D dans un afficheur classique. La haute-capacité permet d'étendre les domaines d'application par ce nouvel espace de stockage comme par exemple en médecine où les informations d'un patient et de diagnostic peuvent être marquées directement dans l'image médicale, pour relier les informations entre elles.

Ainsi, le principal challenge de l'IDC haute-capacité est d'augmenter le plus possible la capacité des méthodes. Par exemple, la méthode proposée par Yang *et al.* [156] permet d'insérer dans une image plusieurs bits par pixel en remplaçant les bits de poids faible des pixels.

1.3.3 Insertion dans les objets 3D

Il existe plusieurs méthodes d'insertion de données cachées dans les objets 3D [31, 64, 69, 100, 109, 141]. Généralement, les auteurs séparent les méthodes en fonction du domaine d'insertion. Ainsi, dans le domaine spatial, les modifications faites afin de coder un message secret peuvent être sur la position des sommets [64, 69], les valeurs au sein de l'histogramme des normales [9], les valeurs au sein de l'histogramme des distances radiales [160] ou les moments locaux [21]. Tandis que dans les domaines transformés, le message secret peut être codé en modifiant les caractéristiques de l'objet 3D dans des domaines ayant des transformées inverse comme le domaine laplacien [99], le domaine fréquentiel [100] ou bien d'autres domaines spectraux [30, 148].

Ces domaines d'insertion ont chacun leurs avantages et leurs inconvénients selon des facteurs bien précis présentés dans le tableau 1.2.

TABLEAU 1.2 – Caractéristiques selon le domaine d'insertion choisi

Facteurs	Domaine spatial	Domaines transformés
Capacité	Élevé	Faible
Robustesse	Faible	Élevé
Qualité perceptuelle	Contrôlable	Peu de contrôle
Complexité	Faible	Élevé

Dans le cadre d'une précédente collaboration avec la société STRATEGIES, Itier *et al.* [64–72] ont proposé une méthode insérant un message caché au sein d'un maillage 3D dans le domaine spatial en utilisant pour ordre de synchronisation un chemin hamiltonien. Ce chemin est construit sur l'ensemble du nuage de points tout en modifiant la position des sommets selon le point précédent dans le chemin pour coder un message. Les coordonnées sphériques sont alors quantifiées pour pouvoir coder trois octets par sommet, il s'agit alors d'une méthode de modulation d'indices quantifiés (*Quantization Index Modulation* ou QIM). La figure 1.8 présente le processus d'IDC haute-capacité proposé par Itier et Puech [64] avec l'objet 3D original (cf. figure 1.8.a). Le nuage de points illustré figure 1.8.b, correspond à celui utilisé pour construire le chemin hamiltonien et insérer le message. Nous constatons par ailleurs que le chemin hamiltonien déduit avant insertion (cf. figure 1.8.c) et celui après insertion (cf. figure 1.8.d) possédent de nombreuses arêtes en commun. Nous observons également que l'objet 3D marqué illustré en figure 1.8.e est visuellement similaire à l'original malgré les légers déplacements des sommets illustrés en couleur en figure 1.8.f.

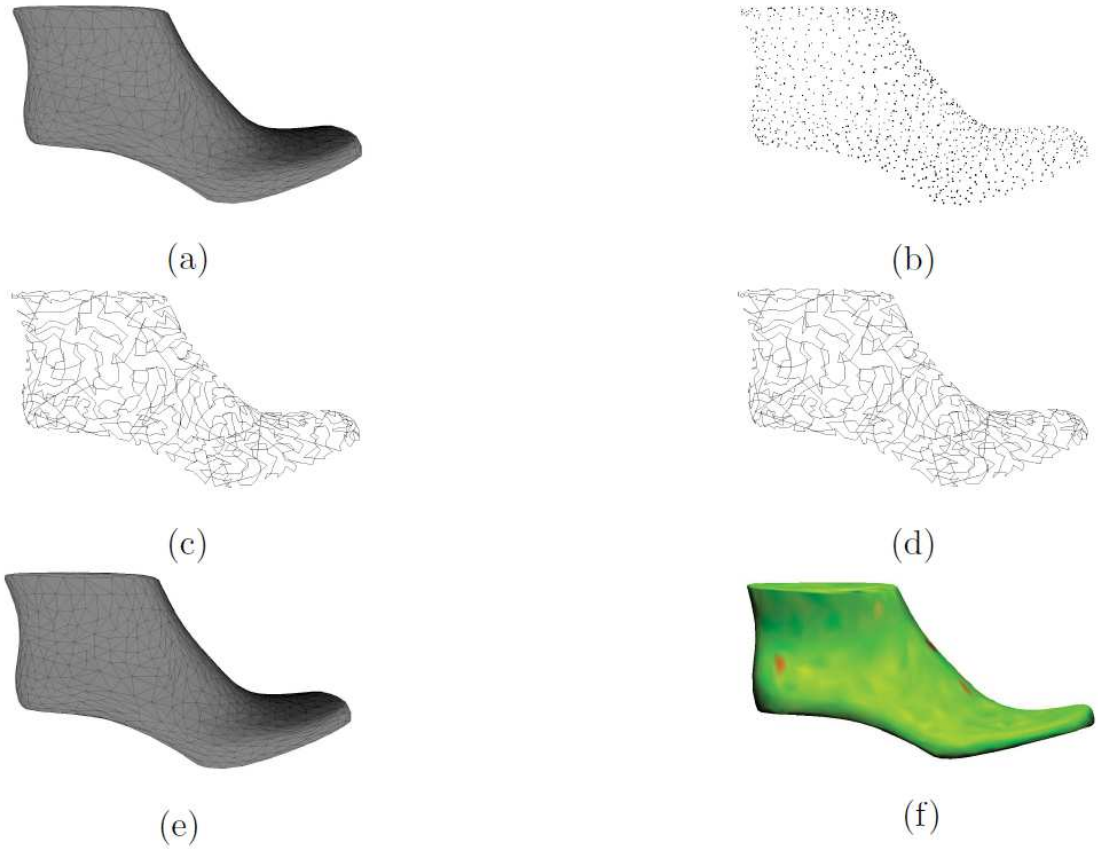


FIGURE 1.8 – Processus d'insertion de données cachées haute-capacité proposé par Itier et Puech [64] : a) Objet 3D original avec 1002 sommets (soit 3006 octets de capacité), b) nuage de points, c) chemin hamiltonien avant insertion, d) chemin hamiltonien marqué, e) objet 3D marqué, f) comparaison entre l'objet 3D initial et celui marqué par RMSE_v.

1.4 Chiffrement multimédia

Les données multimédia regroupent un ensemble large de données textuelles (textes, web) mais aussi visuelles (image, audio, vidéo et objets 3D). La gestion d'accès à ces données est souvent réalisée à l'aide de système complexe dédié à la gestion de droits numériques (*Digital Right Management* ou DRM) pour assurer la propriété et la protection du contenu. L'approche classique à cette protection est d'appliquer naïvement les méthodes de chiffrement sur les données sans prendre en compte la structure du média traité.

Au contraire, le chiffrement multimédia prend en compte les spécificités des structures internes représentant le média afin d'en tirer le meilleur parti. Cela permet d'avoir un format, après chiffrement cohérent et dont le contenu n'est plus reconnaissable visuellement. Le chiffrement sélectif est une catégorie au sein des méthodes de chiffrement multimédia où des informations précises au sein du média à chiffrer sont sélectionnées pour le chiffrement selon un certain niveau de confidentialité.

En section 1.4.1 nous présentons tout d'abord les grands principes de la cryptographie passée et moderne. En section 1.4.2 nous détaillons les différentes approches en cryptographie moderne. Puis, en section 1.4.3 nous détaillons le chiffrement sélectif. Enfin en section 1.4.4 nous présentons quelques méthodes de l'état de l'art sur le chiffrement sélectif à destination des objets 3D.

1.4.1 Histoire de la cryptographie

La cryptographie, ou la science du secret, a très longtemps été le seul moyen efficace pour répondre aux besoins de sécurisation des messages contre les accès illégaux [122]. L'objectif de la cryptographie est d'assurer la confidentialité en chiffrant l'information et en la rendant incompréhensible à une personne ne possédant pas la clé permettant de déchiffrer l'information [78]. Ainsi, la confidentialité des données est obtenue par l'utilisation d'algorithme de chiffrement. Ces systèmes réalisant les opérations de chiffrement sont appelées cryptosystèmes, ils transforment un texte clair en texte chiffré, de telle façon que seuls les utilisateurs autorisés soient en mesure de récupérer l'information à partir de la donnée chiffrée.

Dès l'Antiquité, l'usage de la cryptographie était destiné avant tout au domaine militaire. Kahn présente dans son ouvrage *The Codebreakers* la longue histoire de cette science en commençant par la "Scytale" spartiate (500 ans avant J.C.) [75].

FIGURE 1.9 – La Scytale spartiate.

La scytale, illustrée en figure 1.9, est un bâton de bois de diamètre fixé par les personnes communiquant secrètement autour duquel on entoure une bande de cuir. Le diamètre du bâton servait de clé de chiffrement et de déchiffrement des messages secrets. L'expéditeur écrit son message secret sur la bande de cuir une fois cette dernière enroulée. En déroulant la bande de cuir, il rend le message incompréhensible. Seul un bâton

de la même taille que celui utilisé par l'expéditeur permet le déchiffrement du message. Le mécanisme de la Scytale est une technique de chiffrement par **transposition** car ce dernier ne fait que mélanger les lettres sans les modifier.

Une autre technique de chiffrement célèbre est *le chiffre* ou *code de César* qui consiste en un chiffrement par **substitution** de lettres. Ainsi, chaque lettre est remplacée par une seule autre, selon un certain décalage circulaire dans l'alphabet. César utilisait un décalage circulaire de trois positions. Par exemple, le mot CRYPTOGRAPHIE devient FUBSWRJUDSKLH selon l'approche de César. Formellement en cryptographie un vocabulaire spécifique définit ce qu'est un cryptosytème. Ce dernier est composé de deux opérations, à savoir le chiffrement et le déchiffrement. L'opération de transformation appelée chiffrement fait généralement appel à une clé secrète ou clé de chiffrement. Ainsi, nous notons $\mathcal{M}$ le message initial, $\mathcal{E}$ l'opération de chiffrement et $\mathcal{K}$ la clé de chiffrement tels que :

$$\mathcal{M}_{\mathcal{K}} = \mathcal{E}(\mathcal{M}, \mathcal{K}), \tag{1.22}$$

où $\mathcal{M}_{\mathcal{K}}$ est le message chiffré à l'aide de la clé secrète $\mathcal{K}$. L'opération de déchiffrement notée $\mathcal{D}$ nécessite une clé de déchiffrement $\mathcal{K}_d$ donne :

$$\mathcal{M} = \mathcal{D}(\mathcal{M}_{\mathcal{K}}, \mathcal{K}_d) \tag{1.23}$$

1.4.2 Cryptographie moderne

Jusqu'au 20ème siècle, les techniques de cryptographie se basaient sur les techniques précédemment présentées. Cependant, le fait de connaître la technique employée permet de réaliser un déchiffrement et de rapidement briser le cryptosystème. C'est pourquoi la cryptographie moderne, et en particulier grâce à l'informatique, s'est fortement développée pour assurer une sécurité maximale. Ainsi, dans les années 80, le développement de communications entre ordinateurs a nécessité la création d'une première norme de chiffrement, le DES (*Data Encryption Standard*) [43]. Les principes de la cryptographie moderne sont établis par Kerckhoffs en 1883 [78] :

— La sécurité repose sur le secret de la clé et non sur le secret de la méthode.

— Le déchiffrement sans la clé doit être matériellement, sinon mathématiquement impossible.

— Trouver la clé à partir du texte clair et du texte chiffré est impossible.

La cryptographie moderne se divise en deux catégories : la cryptographie symétrique et la cryptographie asymétrique.

Cryptographie symétrique

Souvent appelées chiffrement à clé privée (ou secrète), les méthodes de cryptographie symétrique utilisent la même clé pour le chiffrement et le déchiffrement comme illustré figure 1.10. La clé doit rester secrète, car toute la sécurité du cryptosystème est dépendante du fait que l'expéditeur et le destinataire sont les seuls à la connaître. Lorsque le cryptosystème est jugé "sécurisé" d'un point de vue technique, la taille de la clé peut déterminer la difficulté pour briser un tel chiffrement. En effet, le temps de chiffrement augmente avec la taille de la clé et cela de manière exponentielle. Les processeurs actuels permettent de traiter rapidement des quantités de données importantes, toutefois il ne faut pas sous-estimer l'impact des opérations de chiffrement et de déchiffrement sur l'expérience utilisateur. Il existe deux catégories de systèmes à clé privée : le chiffrement par bloc et le chiffrement par flot.

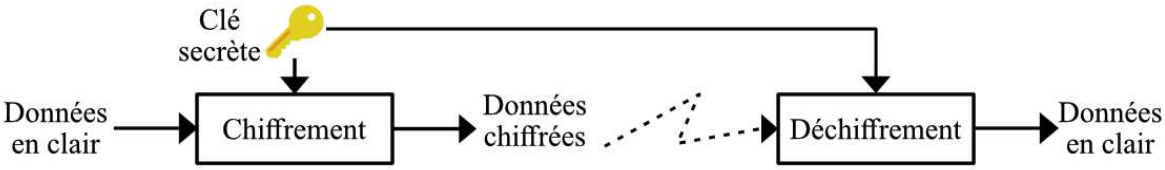


FIGURE 1.10 – Principe du chiffrement symétrique.

Le chiffrement par bloc Les techniques de chiffrement par bloc consistent à diviser les données en clair $\mathcal{M}$, en T blocs de taille fixe de n bits (souvent 64, 128 ou 256 bits) et chiffrer un bloc à la fois avec la même clé. Si la longueur de $\mathcal{M}$ n'est pas multiple de n, une technique de bourrage (*padding*) est utilisée sur les blocs afin de les compléter. Il existe au moins cinq modes opératoires de chiffrement par bloc [47] :

— Le mode ECB (*Electronic Code Book*) : le principe est de chiffrer chaque bloc d'une manière indépendante des autres blocs comme illustrés en figure 1.11. Le chiffrement des blocs peut se faire en parallèle afin d'accélérer les opérations de chiffrement.

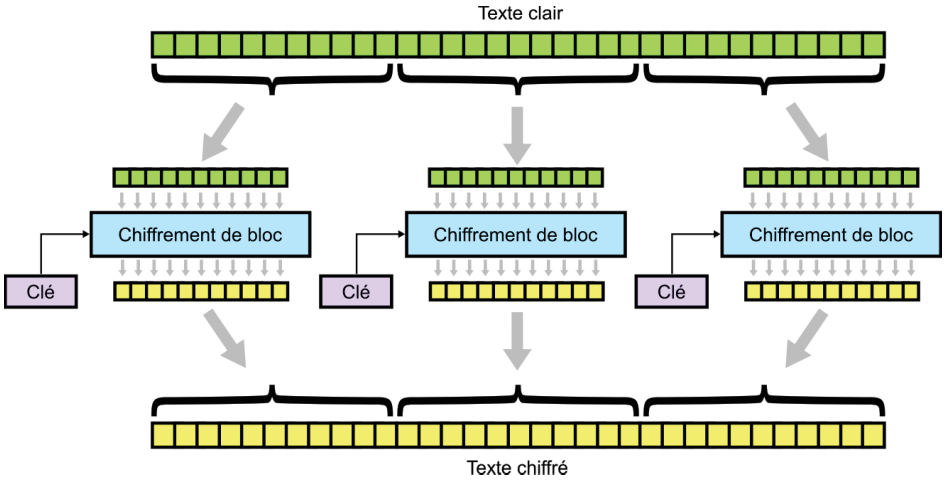


FIGURE 1.11 – Mode ECB.

— Le mode CBC (*Cipher Block Chaining*) : contrairement au mode ECB, une dépendance est faite entre les blocs successifs chiffrés pour augmenter la variabilité des blocs chiffrés. Pour se faire, le bloc b_t à chiffrer, où $t \in [\![0\,;\,\mathrm{T}[\![$ va dépendre du bloc chiffré précédent b'_{t-1} et de la clé secrète $\mathcal{K}$ tels que :

$$b'_t = \mathcal{E}(b_t \oplus b'_{t-1}, \mathcal{K}), \tag{1.24}$$

où b'_{-1} est un vecteur d'initialisation choisi pseudo aléatoirement et $\oplus$ l'opérateur binaire XOR comme illustré figure 1.12.

— Le mode CFB (*Cipher FeedBack*) : le bloc chiffré b'_{t-1} est à nouveau chiffré puis combiné (à l'aide de l'opérateur binaire XOR) avec le bloc en clair courant b_t pour donner le bloc chiffré courant (cf. figure 1.13) tel que :

$$b'_t = b_t \oplus \mathcal{E}(b'_{t-1}, \mathcal{K}). \tag{1.25}$$

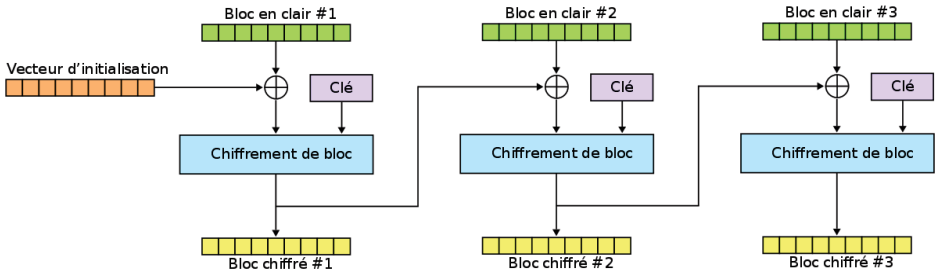


FIGURE 1.12 – Mode CBC.

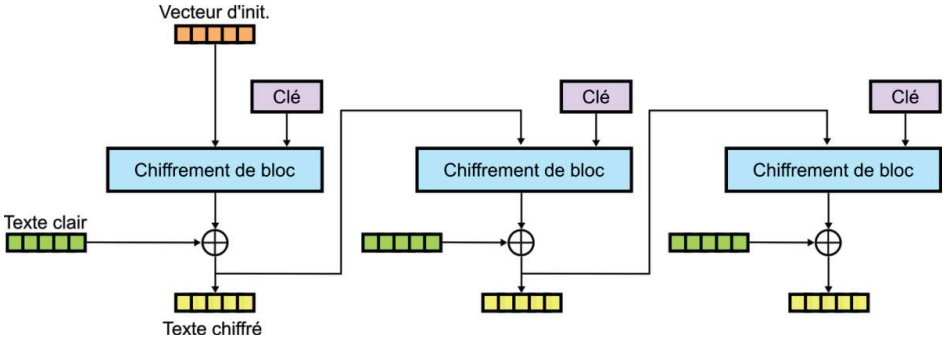


FIGURE 1.13 – Mode CFB.

— Le mode OFB (Output FeedBack) : ce mode opératoire illustré en figure 1.14 est similaire au mode CFB à l'exception de l'ordre d'opérations. En effet, pour obtenir le bloc chiffré b'_t, le bloc en clair b'_t va être combiné avec le bloc chiffré c_t dépendant du vecteur d'initialisation généré tel que :

$$c_t = \mathcal{E}(c_{t-1}, \mathcal{K}), \tag{1.26}$$

où c_{-1} est le vecteur d'initialisation de départ. Ainsi le chiffrement d'un bloc par le mode OFB est équivalent à :

$$b'_t = b_t \oplus c_t. \tag{1.27}$$

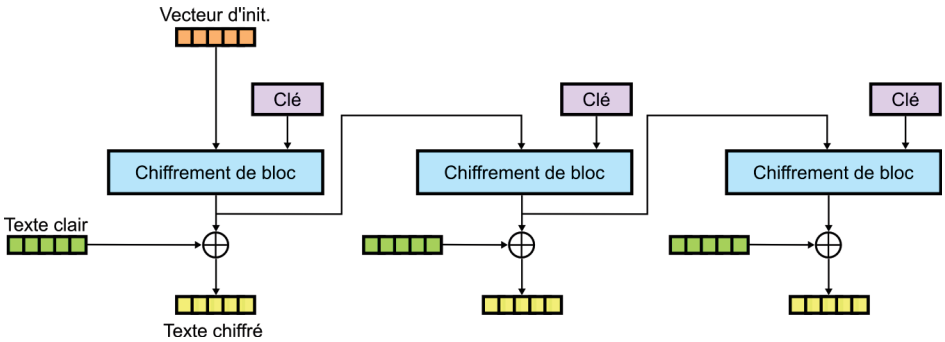


FIGURE 1.14 – Mode OFB.

— Le mode CTR (CounTeR) : ce mode consiste comme illustré en figure 1.15 à chiffrer chaque bloc en fonction de sa position indépendamment des autres blocs tels que :

$$b'_t = b_t \oplus \mathcal{E}(v_t, \mathcal{K}), \tag{1.28}$$

où v_t est la valeur du compteur pour le bloc t, c'est-à-dire $v_t = t$. Cela permet de réaliser un chiffrement pré-calculable lorsque nous avons connaissance du nombre

de blocs, ou bien un chiffrement par flot si ce dernier est inconnu. De plus, ce mode propose un accès aléatoire aux données, car le déchiffrement de chaque bloc ne dépend que de son compteur. Les compteurs utilisés v_t peuvent être générés pseudo-aléatoirement à l'aide de clé secrète.

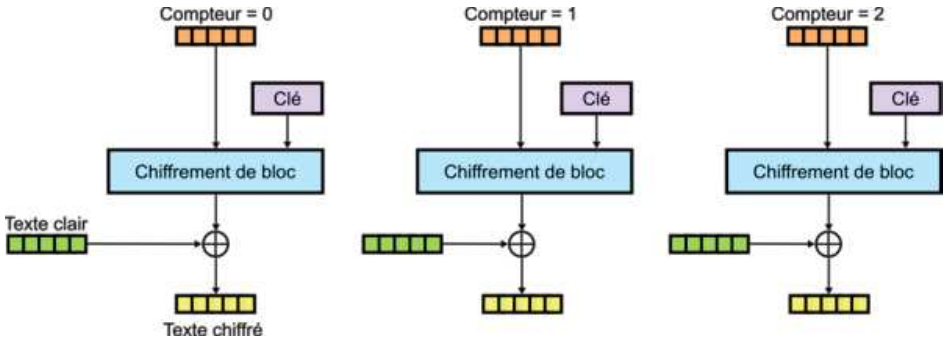


FIGURE 1.15 – Mode CTR.

Il existe plusieurs algorithmes de chiffrement par bloc. Parmi eux, l'algorithme de chiffrement DES (*Data Encryption Standard*) apparu en 1976, est maintenant inutilisé, car devenu trop vulnérable. Le nouveau standard en matière de chiffrement par bloc est l'algorithme de chiffrement AES (*Advanced Encryption Standard*) qui a été développé pour remplacer le DES [42]. Le chiffrement AES utilise des clés de taille allant de 128 à 192 et dernièrement 256 bits [120].

Le chiffrement par flot Les algorithmes de chiffrement par flot sont une classe importante d'algorithmes de chiffrement symétrique [111]. Ils considèrent le message en clair comme un flux de caractères (généralement des bits ou des octets) et effectuent le chiffrement sur chaque caractère. À chaque chiffrement d'un nouveau caractère, la méthode évolue grâce à une fonction de mise-à-jour. Contrairement aux algorithmes de chiffrement par bloc pouvant chiffrer simultanément des blocs de données à l'aide d'une transformation de taille fixe, le chiffrement par flot est extrêmement rapide. En effet le chiffrement par flot peut chiffrer en continu sans attendre la réception complète du message à chiffrer. Dans certains cas, le chiffrement par flot peut être utilisé lorsque l'information à traiter se trouve en petites quantités de symboles à la fois comme par exemple sur les systèmes embarqués ou les objets connectés [121]. Le principe de chiffrement par flot a sa structure présentée par Vernam [138]. Vernam présente le chiffrement par flot d'une approche utilisant une clé secrète pour initialiser un flot d'aléatoire appelé flot de chiffrement qui peut être généré bit par bit ou octet par octet suivant le système. La génération du flot de chiffrement est réalisée à partir d'un générateur de nombres pseudo-aléatoire (GNPA) cryptographiquement sécurisé tel que :

$$k_i = \mathrm{GPA}(\mathcal{K}). \tag{1.29}$$

Le chiffrement de Vernam est ensuite défini tel que l'opération de chiffrement est :

$$c_i = m_i \oplus k_i, \tag{1.30}$$

où m_i sont les symboles (bits ou octets) du message en clair à chiffrer, c_i les symboles du message chiffré et $\oplus$ l'opérateur binaire XOR ("Ou exclusif").

Cryptographie asymétrique

La distribution des clés secrètes est un problème essentiel en cryptographie : pour que n personnes puissent communiquer de manière confidentielle avec chacune des personnes il faut $\frac{n(n-1)}{2}$ clés secrètes. En 1976, Diffie et Hellman proposent une nouvelle façon de chiffrer qui contourne ce problème [46]. Ils présentent le concept de clé privée/publique et d'algorithme asymétrique. La clé de chiffrement peut être une clé publiée largement aux utilisateurs (clé publique). Seule la clé privée, utilisée pour le déchiffrement, doit rester secrète et connue de son seul propriétaire comme présenté en figure 1.16.

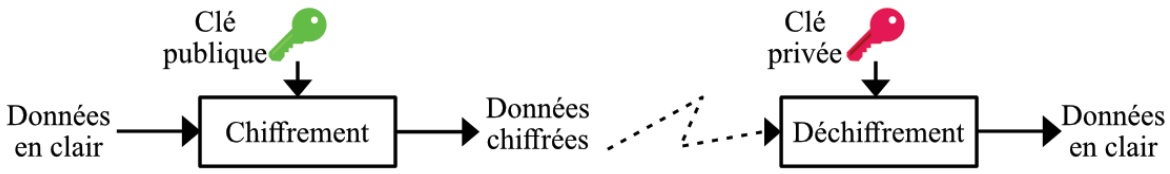


FIGURE 1.16 – Principe du chiffrement asymétrique.

Les clés sont mathématiquement reliées du fait que l'opération de déchiffrement correspond à l'inverse de l'opération de chiffrement. Les propriétés requises pour la réalisation d'un tel cryptosystème sont plus fortes que celles en cryptographie symétrique. En 1978, Rivest *et al.* proposent le premier système à clé publique nommé RSA du nom de ses auteurs *Rivest, Shamir et Adleman* [108]. Leur méthode se base sur le problème de factorisation des grands nombres. La sécurité de leur méthode dépend donc de la difficulté à résoudre ce problème. C'est pourquoi, la taille des clés secrètes peut varier entre 1024 et 4096 bits.

1.4.3 Chiffrement sélectif

Le chiffrement sélectif est une catégorie au sein des méthodes de chiffrement multimédia préservant le format du fichier (images, vidéos, objets 3D) et sélectionnant une partie des informations de ce dernier pour les chiffrer [1, 104, 105, 115, 136]. Contrairement à la cryptographie classique ne prenant pas en compte le contenu du support chiffré, le support est traité de manière à préserver ses structures internes. Ainsi, comme illustré en figure 1.17, le chiffrement sélectif sélectionne un sous-ensemble des données du support de manière locale ou globale selon un niveau de confidentialité permettant de plus ou moins rendre le contenu visuellement confidentiel ou dégradé. Le sous-ensemble de données sélectionnées est ensuite chiffré avec une clé secrète. Le chiffrement sélectif vise aussi, afin de réduire le temps de traitement, à réduire la quantité d'information à chiffrer tout en maintenant un niveau de sécurité suffisant. La protection contre les accès non autorisés au contenu 3D a reçu récemment de l'attention. A. Pommer définit trois niveaux de chiffrement sélectif répondant à des cas d'utilisation spécifiques de protection 3D [1] :

— **Confidentialité visuelle** : la forme et le contenu de l'objet 3D sont visuellement protégés. Des informations comme le format des données peuvent être révélées, mais un attaquant ne peut pas être capable de calculer la moindre information visuelle.

— **Chiffrement suffisant** : seule la forme de l'objet 3D est reconnaissable, mais pas son contenu qui est toujours protégé visuellement.

— **Chiffrement transparent** : la forme et le contenu sont accessibles, cependant la haute qualité de l'objet 3D est protégée. Un adversaire ne pourra récupérer qu'une version basse qualité de l'objet 3D.

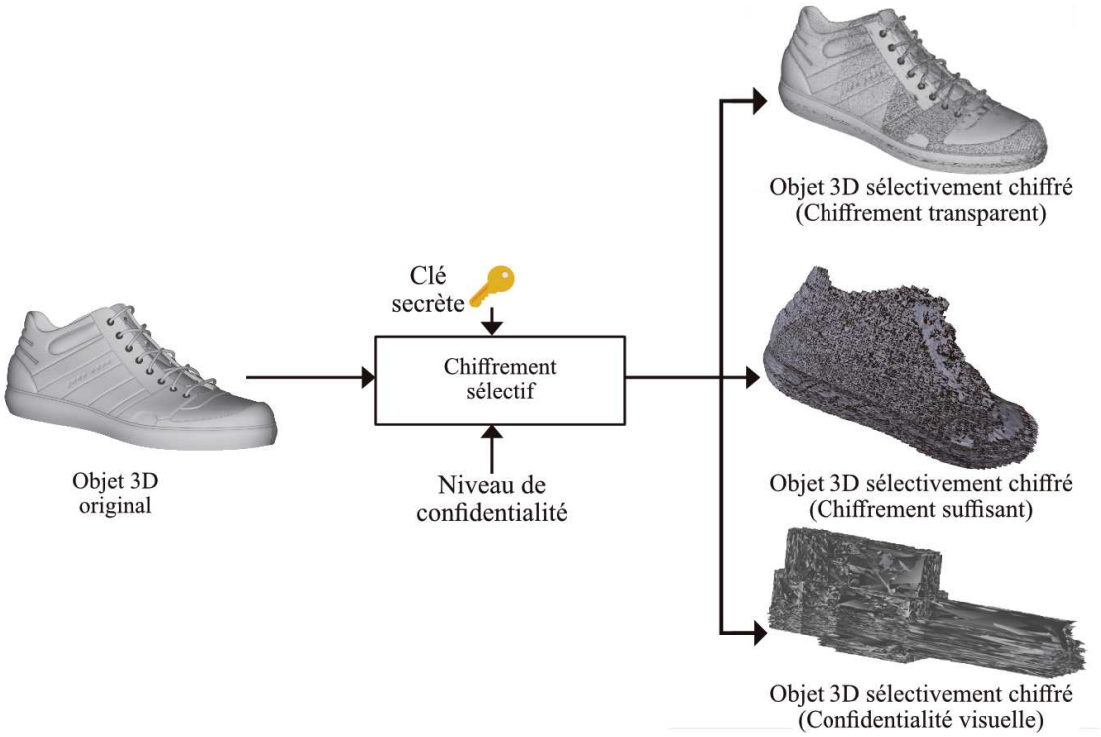


FIGURE 1.17 – Schéma du chiffrement sélectif.

1.4.4 Chiffrement d'objets 3D

En 2009, Gschwandtner et Uhl [55] ont proposé une méthode de protection 3D se basant sur des maillages 3D progressifs. Un maillage progressif est un maillage qui est défini en plusieurs couches de raffinement. Ces couches contiennent des informations sur la géométrie, la connectivité et d'autres attributs permettant de raffiner et d'améliorer la qualité du maillage initial. Dans la proposition de Gschwandtner et Uhl, les couches de raffinement sont chiffrées afin de protéger la haute qualité de l'objet 3D. De cette manière, selon ses droits d'accès, un utilisateur peut déchiffrer les différentes couches de raffinement pour améliorer la qualité de l'objet 3D.

Récemment, Éluard *et al.* ont présenté des méthodes de chiffrement sélectif préservant la géométrie [50]. Ces méthodes réalisent des permutations de sommets ou de coordonnées pour chiffrer le maillage 3D et assurer la confidentialité visuelle de l'objet.

La figure 1.18 représente des objets 3D chiffrés selon trois méthodes de chiffrement sélectif 3D proposées par Éluard *et al.*. La figure 1.18.a présente l'objet 3D original, tandis que les figures 1.18.b-e représentent l'objet 3D chiffré selon l'approche *Coordinate Shuffling* (CS) où un pourcentage précis (1, 5, 10, et 100%) de coordonnées sont permutées selon une clé secrète. Nous remarquons que 1% des coordonnées permutées permettent d'atteindre un niveau de confidentialité très élevé. L'approche nommée *Dithering* applique un bruit additif sous la forme d'un vecteur généré pseudo-aléatoirement à partir d'une clé secrète comme illustrée en figure 1.18.f-i. Ainsi, le vecteur généré modifie la position du sommet selon une direction et avec une certaine force α contrôlant la magnitude

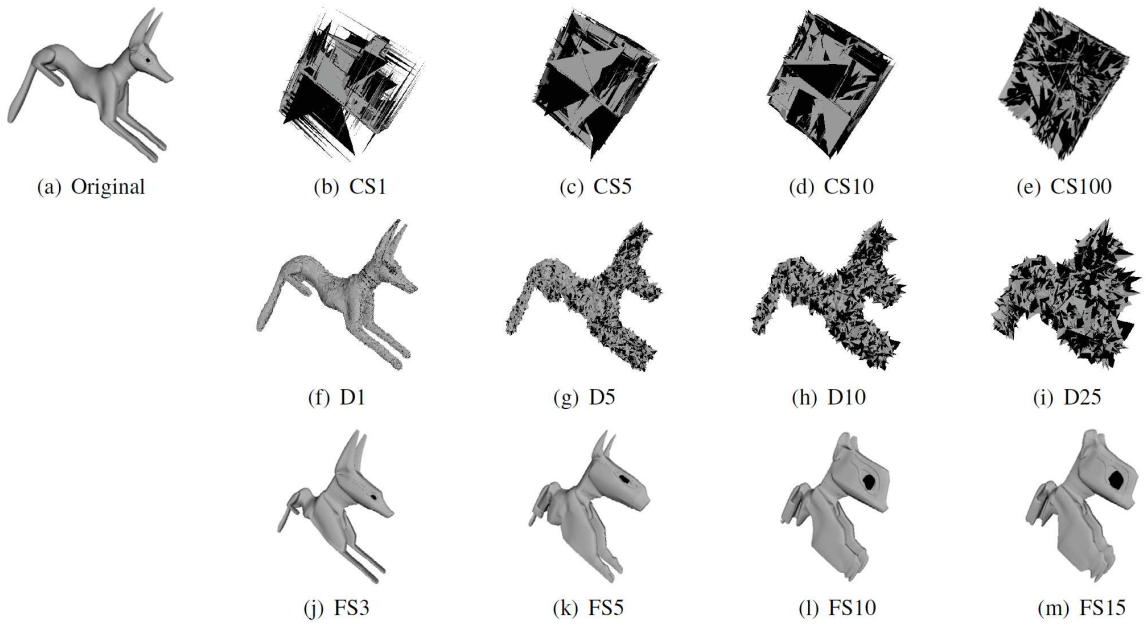


FIGURE 1.18 – Résultats des méthodes de protection préservant la géométrie proposées par Éluard *et al.* [50].

du vecteur, cela permet ainsi de varier entre un chiffrement transparent et suffisant. Enfin, la dernière approche nommée *Fragment Scaling* (FS) consiste à partitionner l'objet 3D selon une subdivision de l'espace et de modifier l'échelle de ces subdivisions pour déformer l'objet 3D sans pour autant créer des artéfacts importants. Ainsi, en figures 1.18.j-m, nous observons les distorsions entraînées par cette approche qui conserve la forme globale de l'objet.

1.5 Partage de secret

Le partage de secret (*Secret Sharing*) est un concept qui a été développé indépendamment par Shamir [116] et Blakley [20] en 1979 pour résoudre les problèmes liés aux méthodes classiques de chiffrement (cf. section 1.4). En effet, ces dernières ont pour défaut de dépendre d'une clé et d'un seul conteneur dans lequel le secret est inséré. Ce conteneur peut être perdu, détruit ou alors altéré durant une attaque rendant la récupération du secret impossible. C'est pour répondre à cette problématique que le principe de partage de secret a été conçu. Depuis les années 1980, les cas d'applications du partage de secret ont évolué de leur but initial. Ainsi, le partage de secret permet de répondre à de nouvelles problématiques comme le calcul multipartite sécurisé, la résolution du problème d'accord byzantin, le contrôle d'accès et le chiffrement par attribut [6].

Dans un premier temps, en section 1.5.1 nous décrivons les grands principes du partage de secret et présentons les méthodes de Shamir [116] et Blakley [20]. Nous détaillons ensuite l'ensemble des propriétés développées au sein des méthodes de partage de secret en section 1.5.2.

1.5.1 Principes

Comme présenté dans l'état de l'art de Beimel [6], le partage de secret rassemble un grand nombre d'outils importants de la cryptographie servant de base pour la construction de nombreux protocoles sécurisés. Comme indiqué en introduction de cette section, le partage de secret est introduit indépendamment par Shamir [116] et Blakley [20] en 1979. Le partage de secret est présenté comme une approche de chiffrement sans clé où un secret peut être distribué entre plusieurs utilisateurs et reconstruit lorsque un sous-ensemble de ces utilisateurs se réunissent pour récupérer le secret. Celui partageant le secret est nommé donneur (*dealer*).

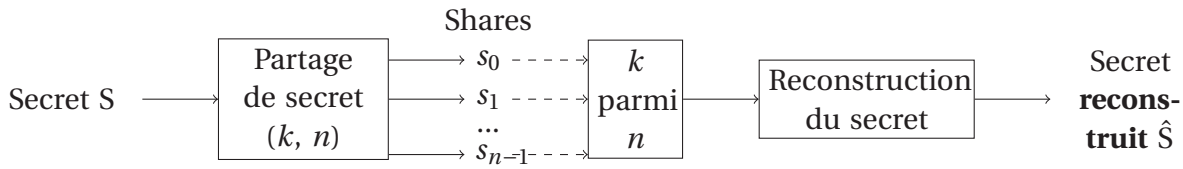


FIGURE 1.19 – Processus de partage et de reconstruction du secret S.

Comme illustré en figure 1.19, durant la phase de partage, à partir du secret S, le donneur génère des parts (*shares*) noté s_i. Chaque part est affectée à un utilisateur parmi n tel que $i \in [\![0\,;\,n[\![$. Dans la suite de ce manuscrit, nous utilisons le terme *share* pour désigner des parts distribuées aux utilisateurs. Les *shares* ne sont pas des parties disjointes du secret S, mais des informations calculées à partir du secret S permettant de reconstruire ce dernier. Ainsi, ces *shares* ne fournissent aucune information individuellement sur le secret S. Pour reconstruire le secret, il est nécessaire de regrouper au moins k *shares* parmi celles des n utilisateurs. N'importe quel groupe d'utilisateurs ayant strictement moins de k membres parmi les n ne peut pas reconstruire le secret. Cette première définition du partage de secret porte le nom de méthode seuillée-(k, n).

Cette définition a ensuite été étendue et généralisée par Ito *et al.* [73]. Ainsi, une méthode de partage de secret est définie par le triplet $(\mathrm{S}, \mathcal{U}, \mathcal{A})$, où S est le secret à protéger, $\mathcal{U}$ l'ensemble d'utilisateurs $(u_0, u_1, \ldots, un_1)$ autorisés et $\mathcal{A}$ la structure d'accès générale (*general access structure*). Une structure d'accès générale est une collection de sous-ensembles de $\mathcal{U}$ autorisés à reconstruire le secret S. Ainsi, cette définition retire le principe de seuil minimum pour généraliser à des ensembles d'utilisateurs de tailles différentes. Ito *et al.* [73] rappellent aussi que la structure d'accès générale doit satisfaire la propriété suivante :

$$\mathrm{A} \in \mathcal{A}, \mathrm{A} \subseteq \mathrm{A}' \implies \mathrm{A}' \in \mathcal{A}. \tag{1.31}$$

Cette propriété signifie que pour tout groupe d'utilisateurs A' comportant un sous-ensemble d'utilisateurs A appartenant à la structure d'accès $\mathcal{A}$, alors le groupe d'utilisateurs A' est autorisé à reconstruire le secret S. La méthode proposée par Ito *et al.* [73] est non raffinée ; en effet, pour chaque ensemble de la structure d'accès $\mathcal{A}$, le secret S est partagé parmi les membres de l'ensemble. De plus, comme l'explique Benaloh [8], dans le pire des cas chaque utilisateur appartenant à $\mathcal{U}$ reçoit $2^{|\mathcal{U}|}$ *shares*.

Dans la suite de cette section, nous nous attardons principalement sur les méthodes de Shamir [116] et Blakley [20].

Méthode de Shamir

La méthode de Shamir présentée en 1979 considère le secret S comme un élément d'un corps fini [116]. Pour protéger ce secret, l'approche de Shamir consiste à utiliser des polynômes sur un corps fini $\mathbb{F}_q$ tel que $|\mathbb{F}_q| = q$ avec $|.|$ le cardinal d'un corps fini et q un nombre premier respectant :

$$1 < k \leq n < q, \tag{1.32}$$

$$0 \leq \mathrm{S} < q, \tag{1.33}$$

avec k le nombre minimum d'utilisateurs requis pour reconstruire le secret et n le nombre maximum d'utilisateurs devant posséder une *share* .

Avec les paramètres (k, n), cette méthode distribue un ensemble de *shares* pour chacun des n utilisateurs et permet la reconstruction du secret lorsqu'au moins k des n utilisateurs regroupent leur *share* pour résoudre un problème d'interpolation polynomiale. Avec seulement un sous-ensemble de $(k-1)$ *shares*, aucune information sur le secret ne peut être révélée. Durant l'étape de partage, chaque utilisateur reçoit un unique identifiant x_j, où $0 < x_j < q$ et $j \in [\![0\,;\,n[\![$. Un polynôme de degré $(k-1)$ est ensuite construit, tel que $(k-1)$ entiers sont choisis aléatoirement pour former l'ensemble $\mathbf{a} = \{a_i \in \mathbb{F}_q\}$ avec $i \in [\![1\,;\,k[\![$, $a_i < q$ et $a_0 = \mathrm{S}$:

$$f(x) = \sum_{i=0}^{k-1} a_i \times x^i. \tag{1.34}$$

Ainsi, $f(0)$, qui est égale à a_0, correspond à la valeur du secret S. Chacun des n utilisateurs reçoit la paire d'information $s_j = (x_j, y_j = f(x_j))$.

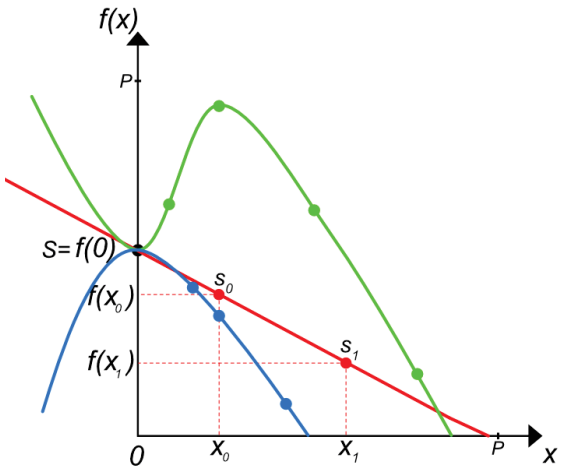


FIGURE 1.20 – Exemple de la méthode de Shamir [116] avec les paramètres : (2, n) en rouge, (3, n) en bleu et (4, n) en vert.

Comme illustré en figure 1.20, le secret S peut être reconstruit en utilisant une interpolation polynomiale avec au moins k *shares*. Ces *shares* peuvent être interprétées comme des points 3D appartenant au polynôme utilisé. Par exemple sur la figure 1.20, les points $s_0 = (x_0, f(x_0))$ et $s_1 = (x_1, f(x_1))$ peuvent être utilisées pour interpoler le polynôme de degré 1 $f(x)$ représenté par la ligne rouge. Toujours dans la figure 1.20, la courbe bleue représente un polynôme de degré 2, qui peut reconstruire le secret S lorsqu'au moins 3 points sont utilisés pour interpoler le polynôme, tandis que la courbe verte est un polynôme de degré 3 permettant la reconstruction du secret S à partir de 4 points. Une interpolation est alors appliquée pour déterminer la valeur de $f(0)$ du polynôme qui correspond au terme

a_0 et par définition au secret S. Pour reconstruire le secret S, un groupe d'au moins k utilisateurs peut déterminer, par interpolation de Lagrange, les coefficients utilisés dans le polynôme $f(x)$ durant l'étape de partage :

$$f(x) = \sum_{i=0}^{k-1} y_i \times \prod_{u=0, i \neq u}^{k-1} \frac{x - x_u}{x_i - x_u} \mod q. \tag{1.35}$$

Méthode de Blakley

Durant la même année, Blakley a proposé une méthode de partage de secret en utilisant de la géométrie hyperplanaire [20]. Ce dernier définit le secret S comme un point dans un espace à k-dimension tel que $\mathrm{S} = (x_0, x_1, \ldots, x_{k-1})$. Il distribue alors aux utilisateurs des hyperplans de dimension k tels que le point S se trouve dans chaque hyperplan distribué. Formellement, un hyperplan de dimension k noté H est défini par l'équation suivante :

$$b = \sum_{i=0}^{k-1} a_i \times x_i, \tag{1.36}$$

où a_i avec $i \in [\![0\ ;\ k[\![$ est le i-ème coefficient de l'hyperplan de dimension k défini par l'ensemble $\mathbf{a} = \{a_i\}$ et b le coefficient.

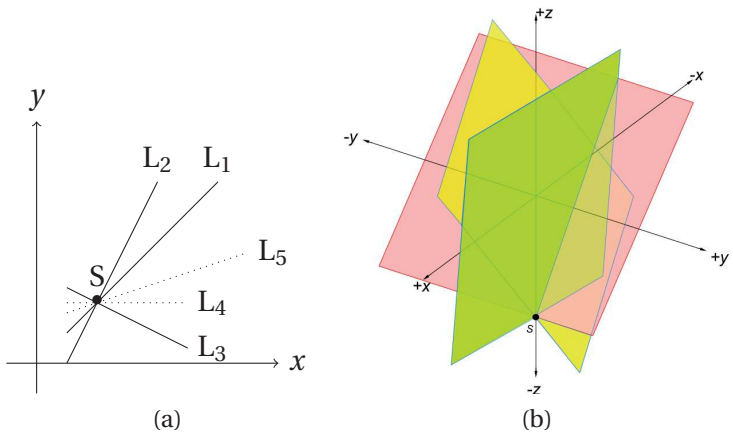


FIGURE 1.21 – Exemples de la méthode de Blakley [20] avec les paramètres : a) $(2, n)$ et b) $(3, n)$.

Grâce à la géométrie hyperplanaire, le point secret est le point d'intersection de n'importe quel groupe de k ou plus hyperplans. La figure 1.21 illustre comment les hyperplans s'intersectent ensemble en un seul point pour $k = 2$ (cf. figure 1.21a) et pour $k = 3$ (cf. figure 1.21b), respectivement. Les *shares* distribuées sont donc les hyperplans de dimension k et plus exactement les coefficients de l'équation représentant l'hyperplan $\mathrm{H}_j = (\{a_{j,i}\}, b_j)$ avec $j \in [\![0\ ;\ n[\![$.

1.5.2 Propriétés

De par leur construction ou l'organisation de la structure d'accès $\mathcal{A}$, certaines méthodes de partage de secret possèdent des propriétés particulièrement intéressantes [6, 7, 24–26, 119, 125], à savoir les méthodes peuvent être considérées comme parfaites, idéales, vérifiables ou bien hiérarchiques.

Parfait (*Perfect*)

Une méthode de partage de secret est dite *parfaite* lorsque pour n'importe quel ensemble d'utilisateurs autorisés A où $A \subseteq \mathcal{U}$, mais n'appartenant pas à la structure d'accès $\mathcal{A}$ (c'est-à-dire la liste des groupes d'utilisateurs autorisés), alors aucune information sur le secret S n'est révélée dans le sens théorique de l'information. Ainsi des méthodes comme celle de Shamir [116] peuvent être considérées comme *parfaites* [26, 92, 116].

Ce n'est pas le cas de celle de Blakley [20]. En effet, prenons l'exemple tel que $(k = 3, n)$. Si un attaquant arrive à obtenir $k-1$ *shares* , c'est-à-dire deux plans, il peut dès alors déterminer une droite d'intersection entre les deux plans. L'attaquant peut donc en déduire que le point d'intersection S se trouve sur cette droite.

Idéal (*Ideal*)

La définition d'une méthode de partage de secret dite *idéale* a été établie par Brickell [24] en 1989. Ce dernier définit qu'une méthode de partage de secret est *idéale* lorsque pour n'importe quel secret S et n'importe quelle *share* T générée à partir de S les *shares* ont la même taille (en bits) que le secret en entrée. Ainsi, la méthode de Shamir [116] est *idéale*. La recherche de méthodes de partage de secret *idéales* est un sujet important. En opposition aux méthodes dites *idéales*, certaines méthodes se sont concentrées sur le fait d'essayer de réduire la taille des *shares* pour des raisons de stockage et de temps de transmission [80].

Vérifiable (*Verifiable*)

Certaines méthodes de partage de secret [8, 37, 57, 161] intègrent des mécanismes de détection de fautes permettant de savoir si un utilisateur malveillant n'a pas donné une mauvaise *share* pour empêcher la reconstruction du secret. Ces mécanismes peuvent permettre la correction des fautes [8], mais aussi la détection des utilisateurs malveillants [57].

Hiérarchique (*Hierarchical*)

Plusieurs méthodes de partage de secret ont proposé de hiérarchiser les utilisateurs en groupes ou niveaux [7, 51, 98, 119, 125]. Avec ce type de méthode, les utilisateurs ont un niveau d'accès au secret différent alors qu'avec une approche classique ils possèdent le même niveau dans les méthodes seuillées. Ces méthodes peuvent alors servir de base pour des approches de chiffrement par attribut.

En 1998, Simmons considère un système où les utilisateurs sont répartis en différents groupes et à chaque groupe est assigné un seuil devant être atteint pour chaque groupe afin de permettre la reconstruction du secret [119]. Simmons détaille par ailleurs deux types de hiérarchie pour les méthodes de partage hiérarchique de secret : les méthodes de partage multi-niveaux (*multilevel threshold secret sharing scheme*) et les méthodes de partage compartimenté (*compartmented threshold secret sharing scheme*).

Nous développons les aspects hiérarchiques des méthodes de partage de secret en section 2.4 du chapitre 2.

1.6 Conclusion

Dans ce chapitre, nous avons présenté différentes représentations des objets 3D en définissant en détail les principales représentations discrètes que nous allons sécuriser

comme les nuages de points et les maillages 3D. Nous avons présenté des méthodes d'évaluation des distorsions de maillages 3D modifiés par rapport à leur maillage de référence. Dans nos travaux, ces métriques vont nous permettre d'évaluer et valider nos résultats expérimentaux.

Dans une seconde partie de ce chapitre, nous avons présenté de nombreuses approches possibles en sécurité multimédia, en commençant tout d'abord par l'insertion de données cachées. L'insertion de données cachées varie en fonction des besoins d'invisibilité, de robustesse, de capacité, de sécurité et de complexité. Nous avons ensuite présenté les différentes applications de l'insertion de données cachées dans les objets 3D ainsi que la méthode proposée par Itier *et al.* [70] lors de la thèse précédente en collaboration avec la société STRATEGIES [63].

Nous avons poursuivi la présentation des différentes techniques de sécurité multimédia en définissant les approches de chiffrement sélectif. Tout d'abord, nous avons introduit, avec un bref rappel, la cryptographie classique pour ensuite continuer sur le chiffrement de supports multimédia préservant le format du média et permettant son utilisation malgré la présence de données chiffrées. Nous avons alors défini le chiffrement sélectif, qui est une catégorie spécifique parmi les méthodes de chiffrement, où le contenu est partiellement ou globalement chiffré selon le choix de l'utilisateur en charge du chiffrement. Nous avons également présenté des méthodes de chiffrement appliquées au domaine 3D.

Enfin, nous avons détaillé le concept de partage de secret proposé par Shamir et Blakley permettant de protéger une information secrète au sein d'un groupe de n utilisateurs afin que n'importe quel groupe de k utilisateurs puissent reconstruire le secret. Nous avons présenté deux méthodes de partage de secret de Shamir [116] et de Blakley [20] ainsi que les principales propriétés entourant les méthodes de partage de secret sur lesquelles nous nous sommes appuyés pour développer les travaux présentés dans ce manuscrit.

Chapitre 2

Partage de données multimédia secrètes

Sommaire

2.1 Introduction

Durant les années 90, le partage de secret a vu son champ d'action étendu à de nouveaux domaines de la sécurité. Ainsi, avec l'évolution du chiffrement présentée en section 1.4 du chapitre 1, le partage de secret a commencé à être adapté aux données multimédia et plus particulièrement aux images [96]. Cet engouement pour la protection et le partage d'image secrète reprend dès 2003 avec l'utilisation de la méthode de Shamir [116] tout en préservant le format des images [128]. Ce n'est qu'à partir des années 2010 que le partage de secret est adapté aux objets 3D [48].

Dans ce chapitre, nous présentons en section 2.2 le partage d'image secrète qui est l'extension du partage de secret aux images en 2D en reprenant les principes du chiffrement multimédia. En section 2.3, nous détaillons l'état de l'art des méthodes de partage d'objet 3D secret, un domaine très peu étudié à ce jour.

2.2 Partage d'image secrète

Comme présenté dans la section 1.5 du chapitre 1, le concept de partage de secret provient de méthodes destinées initialement à la gestion de clés cryptographiques secrètes [20, 116]. En 1994, Naor et Shamir ont proposé d'adapter le concept de partage de secret au domaine de l'imagerie 2D [96]. Ainsi, ces auteurs ont défini le partage de secret visuel (*Visual Secret Sharing*) où le secret est le contenu d'une image. En section 2.2.1 nous présentons tout d'abord le principe de partage d'image secrète. La section 2.2.2 décrit la cryptographie visuelle (*Visual Cryptography*), une des premières approches en matière de partage de secret visuel. Tandis qu'en section 2.2.3 nous présentons le partage d'image secrète utilisant des approches polynomiales assurant la reconstruction de l'image secrète avec la meilleure qualité possible. Enfin, nous détaillons en section 2.2.4 les principales propriétés des méthodes de partage d'image secrète.

2.2.1 Principe

Tout comme les méthodes de partage de secret, celles pour le partage d'image secrète se divisent en deux parties, à savoir le partage et la reconstruction. Comme illustrée en figure 2.1, la méthode prend en entrée une image secrète et les paramètres k et n, où k est le nombre minimum de *shares* à regrouper pour reconstruire l'image secrète et n le nombre de *shares* à générer. La particularité du partage d'image secrète est que les *shares* sont des images ; en effet, comme la figure 2.1 l'illustre, en sortie de l'étape de partage, la méthode fournit 4 *shares* (s_1, s_2, s_3 et s_4) à distribuer à chacun des utilisateurs. Et lorsque au moins 3 de ces *shares* sont regroupées, alors il est possible de reconstruire l'image secrète I. Gé-

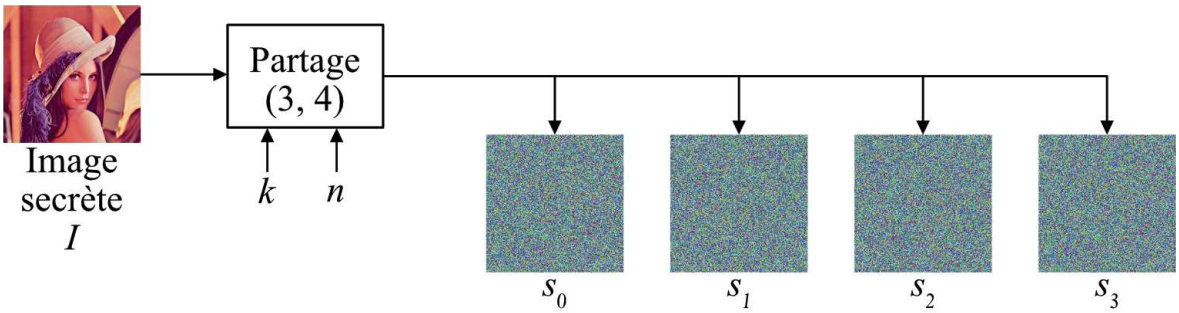


FIGURE 2.1 – Partage d'une image secrète en 4 *shares*.

néralement, l'image secrète I est divisée en plusieurs blocs qui sont ensuite partagés et regroupés pour former les images s_i de sortie, avec $i \in [\![0\ ;\ n[\![$. Ce traitement par bloc est utilisé dans le but de casser la corrélation spatiale des pixels. La reconstruction quant à elle prend en entrée des *shares* ayant la forme d'images pour reconstruire l'image originale ou alors son contenu visuel. La figure 2.2 illustre la reconstruction de l'image secrète de la figure 2.1. Pour n'importe quelle combinaison de trois *shares* parmi les quatre générées par le partage, la méthode permet de reconstruire le contenu de l'image secrète Î.

Comme détaillé dans *"Visual Cryptography & Secret Image Sharing"* [39], il existe deux

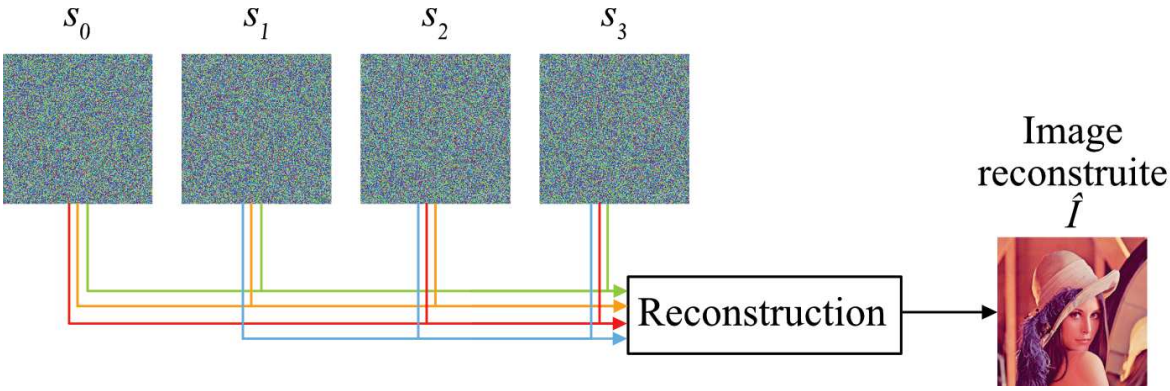


FIGURE 2.2 – Reconstruction d'une image secrète.

catégories de méthodes de partage visuel, à savoir la cryptographie visuelle (*Visual Cryptography*) proposée par Naor et Shamir [96] en 1994 et le partage d'image secrète (*Secret Image Sharing*) proposé par Thien et Lin [128] en 2002.

2.2.2 Cryptographie visuelle

La sécurité visuelle proposée par Naor et Shamir [96] consiste à obtenir un contenu visuel secret en superposant deux *shares*. Au départ destiné aux images binaires (noir ou blanc), les *shares* sont construites en utilisant des répétitions de pixels binaires aléatoires comme illustré en figure 2.3.

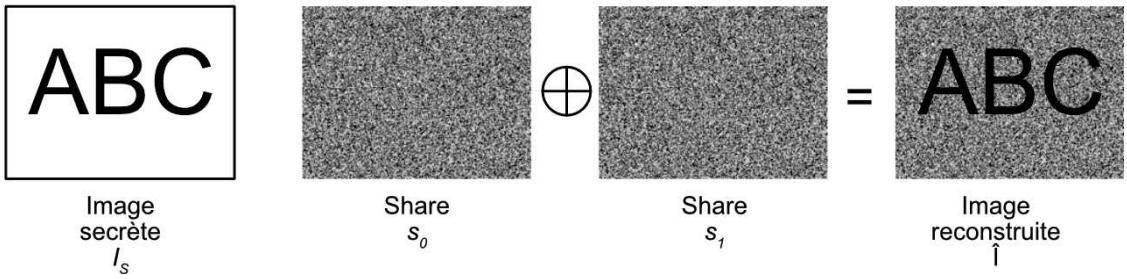


FIGURE 2.3 – Cryptographie visuelle, où ⊕ correspond à l'opérateur binaire *XOR*.

Ainsi, la méthode proposée par Naor et Shamir consiste à construire deux *shares*. La méthode protège les pixels noirs de l'image secrète et laisse aléatoire la couleur des pixels blancs. Le secret est ensuite reconstruit par l'opération *XOR* entre les deux *shares* et son contenu est uniquement reconnaissable par le système visuel humain (SVH). Le fait que cette méthode possède une très faible complexité la rend très avantageuse dans des cas d'utilisation nécessitant des temps de réponse très court. L'approche de cryptographie visuelle a été longuement étudiée, notamment sur la manière de faire varier les différents

seuils k et n, afin de résoudre le problème de contraste présenté par Naor et Shamir sur leur approche [97]. Cimato et Yang [39] divisent les méthodes de cryptographie visuelle en trois groupes, à savoir la cryptographie visuelle basée *XOR* (*XOR-based Visual Cryptography*), la cryptographie basée sur des grilles aléatoires (*Random Grids*) et la cryptographie visuelle probabiliste (*Probalistic Visual Cryptography*). La cryptographie visuelle basée *XOR* correspond à celle proposée par Naor et Shamir et d'autres auteurs [96, 97]. Cependant, les différentes méthodes de cryptographie visuelle basée *XOR* possèdent en général un défaut d'expansion de la taille des *shares*. En effet, afin d'augmenter les seuils de *shares* minimums requis ou à générer, un pixel de l'image secrète est représenté par un bloc de pixels dans chaque *share* de taille m. Ainsi, plus il y a de *shares* à générer ou plus un seuil de reconstruction est élevé, plus la taille des *shares* est grande. C'est pour répondre au problème d'expansion des *shares* que les grilles aléatoires et la cryptographie visuelle probabiliste ont été proposées. Les grilles aléatoires définies par Kafri et Keren sont des matrices de pixels [74]. Chaque pixel est soit transparent, soit opaque, et le choix entre les deux est déterminé aléatoirement. De ce fait, la corrélation spatiale entre les pixels est supprimée. Quand au moins deux grilles sont superposées, les zones cachant le secret apparaissent en fonction de la différence dans la transmission lumineuse de l'image secrète. Ainsi, le secret visuel est reconnaissable par un SVH. Récemment, Shyu a proposé une généralisation des méthodes basées sur les crypto-systèmes à base de grilles aléatoires permettant d'obtenir le partage de secret visuel [117]. La cryptographie visuelle probabiliste répond aux problèmes d'expansion des *shares* présents dans les méthodes précédentes et permet d'assurer la sécurité de l'image secrète [40, 153]. En 2010, Weir et Yan ont proposé un état de l'art complet des différentes techniques de cryptographie visuelle [147].

2.2.3 Partage d'image secrète (à base de polynôme)

Dans l'état de l'art, nous distinguons les méthodes issues de la cryptographie visuelle où l'étape de décodage est quasi-instantanée de celles requérant une plus forte complexité pour la reconstruction. Ces méthodes sont plus coûteuses en temps et ressources mais permettent d'obtenir une meilleure qualité dans la reconstruction de l'image secrète. De plus, elles sont plus souvent adaptées aux images en niveaux de gris ou en couleur qu'aux images binaires. En 2002, Thien et Lin ont proposé d'appliquer la méthode de Shamir directement sur des blocs de pixels d'une image secrète [128]. La méthode est définie sur des images en niveaux de gris (ou RVB) où la valeur des pixels de chaque canal couleur varie entre 0 et 255. Tout d'abord, afin de fonctionner avec la méthode Shamir, un corps fini défini par le nombre premier q est utilisé tel que :

$$|\mathbb{F}_q| = q = 251. \tag{2.1}$$

L'image secrète est donc traitée afin que toutes les valeurs des pixels soient comprises entre 0 et 250. La valeur 251 est attribué à q, car il s'agit du plus grand entier premier représentable sur un seul octet. Ensuite, au lieu d'utiliser seulement le terme a_0 de l'équation (1.34), tous les coefficients sont utilisés comme coefficients du polynôme pour stocker des valeurs d'intensité des pixels. Ainsi, en fonction de la valeur de k, plus ou moins de pixels sont utilisés par le polynôme. L'image est alors divisée en blocs de k pixels qui fournissent les valeurs des coefficients du polynôme. Les valeurs générées par les polynômes sont alors distribuées dans chaque *share* comme valeur de pixel. La taille des *shares* est alors divisée par $\frac{1}{k}$ par rapport à l'image secrète. Cette réduction de taille permet un stockage réduit et offre une transmission plus rapide sur les réseaux.

2.2.4 Propriétés

Les méthodes de partage de secret visuel possèdent de nombreuses propriétés, que cela soit pour la qualité de la reconstruction [128, 154], la réduction ou l'expansion de la taille des *shares* [128, 145], le stockage de multiples images secrètes [132, 151], l'utilisation de l'insertion de données cachées [154] ou encore la progressivité de l'apparition du secret [144].

Reconstruction sans perte

Du fait des méthodes cryptographiques utilisées, l'utilisation de corps fini est parfois indispensable pour assurer la sécurité de ces méthodes. Ainsi, pour des images en niveau de gris ou en RVB, la plupart des méthodes utilisent un corps fini (cf. Eq. (2.1)). Or, l'intervalle de valeurs que peuvent prendre les pixels de l'image secrète varie entre 0 et 255. Une perte d'information est donc possible. L'image secrète est pré-traitée pour que les valeurs de pixels soient comprises entre 0 et 250. Ainsi, la plupart des méthodes [91, 92, 128, 129, 133] proposent une reconstruction de l'image secrète avec pertes. C'est pourquoi Yang *et al.* ont proposé d'utiliser les corps de Galois pour éviter la troncature des valeurs de pixels [154]. Les corps de Galois redéfinissent les opérations arithmétiques afin que les calculs restent au sein d'un corps. L'intérêt des corps de Galois est qu'il est possible d'utiliser des nombres non premiers pour les définir tant que les nombres sélectionnés peuvent se décomposer en nombres premiers. Ainsi, nous pouvons utiliser un corps de Galois pour les valeurs de pixels tel que :

$$\mathrm{GF}(256) = \mathrm{GF}(2^8). \tag{2.2}$$

Ce corps de Galois est fortement utilisé en cryptographie et pour des codes d'erreurs, car il permet de manipuler facilement un octet.

Réduction ou expansion de la taille des *shares*

Thien et Lin ont été les premiers à proposer de réduire la taille des *shares* en profitant de la forme des polynômes utilisés [128]. En utilisant les autres coefficients du polynôme utilisé pour générer la valeur des pixels des *shares*, les pixels sont alors traités par bloc de k pixels dans l'image secrète. À chaque bloc de pixels de l'image secrète correspond un pixel dans les *shares*. Les auteurs ont été ainsi capables de réduire la taille des *shares* à seulement $\frac{1}{k}$ fois la taille de l'image secrète. Récemment, Chen *et al.* ont travaillé sur la réduction de la taille des *shares* pour d'autres structures d'accès générales [34].

Insertion de données cachées

Comme les images partagées aux utilisateurs sont des images ressemblant à un bruit blanc, dans le cadre d'une communication secrète un contrôleur peut facilement détecter une image suspecte sur les réseaux et bloquer sa transmission [75]. Ainsi, afin de transférer les *shares* sur les réseaux de manière moins suspicieuse, Thien et Lin [129] ont profité de leur approche de réduction de taille des *shares* pour cacher ces dernières dans des images hôtes. Cette propriété est aussi appelée *Meaningful shadows*, où les *shadows* sont les images distribuées aux utilisateurs qui visuellement ressemblent à des images anonymes. De nombreux autres auteurs ont travaillé intensivement sur la manière d'insérer des *shares* le plus discrètement possible [91, 153, 154].

Dans ce manuscrit, nous travaillons principalement sur des méthodes proposant un partage d'image secrète basé sur des polynômes. Ainsi, dans le chapitre 4 nous présentons une de nos contributions en matière de partage d'image secrète en section 4.2, ainsi qu'une contribution sur le partage d'objet 3D secret en section 4.3. L'image secrète est donc traitée afin que toutes les valeurs des pixels soient comprises entre 0 et 250.

2.3 Partage d'objet 3D secret

Durant cette dernière décennie, avec l'augmentation des applications utilisant des données 3D, comme la numérisation, la création et l'impression 3D, les objets 3D sont devenus des atouts financiers importants pour leurs ayant-droits. Ainsi, en plus de protéger la propriété intellectuelle de leur créateur, l'utilisation massive de services de stockage sur des systèmes distribués accentue le besoin de protéger ces contenus. Dans un premier temps, en section 2.3.1 nous décrivons le principe de partage d'objet 3D secret. En section 2.3.2, nous présentons des méthodes de partage d'objet 3D secret générant des *shares* binaires. Tandis qu'en section 2.3.3 nous détaillons des méthodes de partage d'objet 3D secret fournissant des *shares* représentées par des objets 3D. Enfin, en section 2.3.4 nous détaillons les différentes propriétés et applications des méthodes de partage d'objet 3D secret.

2.3.1 Principe

Tout comme le partage d'image secrète, le partage d'objet 3D se base sur les méthodes connues de partage de secret [20, 116]. L'état de l'art sur le partage d'objet 3D est très restreint [4, 48, 49, 85, 134]. Ces méthodes servent principalement à protéger le contenu 3D secret, mais aussi à réduire le coût de stockage et de transmission des *shares*. Notons aussi un cas particulier d'utilisation où le partage d'objet 3D secret sert à simplifier le transfert ainsi que des applications pour le *streaming* de plusieurs objets 3D [85]. Les méthodes de partage d'objet 3D possèdent également une étape de partage ainsi qu'une étape de reconstruction. Cependant nous pouvons distinguer deux grandes catégories de partage, à savoir les méthodes ne préservant pas le format des objets 3D et fournissant en sortie des *shares* binaires [4, 48, 49] et des méthodes fournissant des objets 3D en tant que *shares* [85, 134].

2.3.2 Méthodes sans préservation du format

Les premières méthodes de partage d'objet 3D se sont intéressées à directement appliquer les méthodes de partage de secret sur les données 3D qu'il s'agisse de géométrie ou de connectivité des objets 3D [4, 48, 49].

Première méthode proposée par Elsheh et Hamza [48]

En 2010, Elsheh et Hamza [48] ont proposé une première approche consistant à protéger la position des sommets et leur connectivité en utilisant la méthode proposée par Blakley [20]. La méthode propose ainsi un partage où le seuil k est égal à 3. Les coordonnées sont interprétées directement comme des éléments d'un corps fini déterminé par un nombre premier q suffisamment grand pour pouvoir représenter les coordonnées tout en respectant l'équation (2.1). Pour chaque sommet, deux coefficients a_i et b_i sont alors générés aléatoirement, où $i \in [\![0\,;\,n[\![$ et n est le nombre de *shares* à générer. Le coefficient c_i

est déterminé en utilisant l'équation :

$$z = a_i \times x + b_i \times y + c_i, \tag{2.3}$$

tel que x, y, z sont les coordonnées du sommet en cours de partage.

Le même processus peut être réalisé sur la connectivité où chaque facette triangulaire du maillage 3D est représentée par un triplet d'indices de sommets (v_u, v_w, v_t) tel que $u, w, t \in [\![0 ; V[\![$, où V est le nombre de sommets.

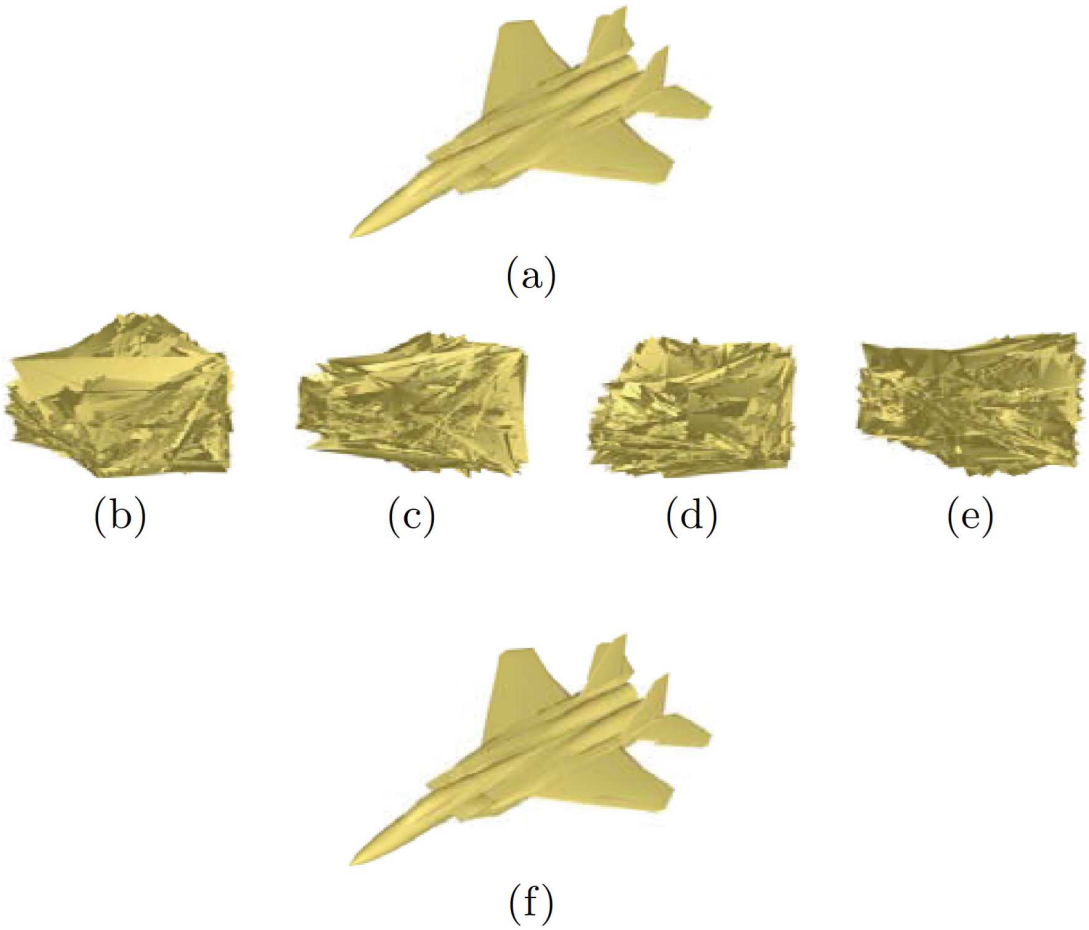


FIGURE 2.4 – Processus de partage et de reconstruction proposé par Elsheh et Hamza [48].

La figure 2.4 illustre le processus de partage selon la méthode de Elsheh et Hamza [48]. La figure 2.4.a représente l'objet 3D secret, les figure 2.4.b-e représentent les hyperplans transmis pour chaque *share* distribuée aux utilisateurs et la figure 2.4.f représente l'objet 3D reconstruit à partir de 3 *shares* parmi les 4. Comme illustré en figures 2.4.b-e, les utilisateurs reçoivent en tant que *shares* un ensemble d'équations d'hyperplans de dimension 3, représenté par les coefficients a_i, b_i et c_i, permettant de reconstruire la géométrie et la connectivité de l'objet 3D.

Seconde méthode proposée par Elsheh et Hamza [49]

Elsheh et Hamza ont ensuite proposé une seconde méthode basée sur la méthode de Shamir [116] en utilisant la même approche que Thien et Lin [128] pour les images 2D où la méthode fournit des *shares* ayant une taille inférieure à celle de l'image secrète [49].

Comme pour leur première méthode [48], les auteurs ont proposé d'appliquer directement la méthode de partage de secret avec $k = 3$ sur les données en prenant soin d'adapter le corps fini par un nombre premier plus grand que le nombre de sommets. Cette méthode a pour défaut d'augmenter le nombre de sommets de l'objet 3D afin que les indices des sommets soient valides. De plus, comme pour l'approche de Thien et Lin [128], la méthode sélectionne non pas un bloc de k pixels, mais k coordonnées ou indices. La taille des *shares* est donc réduite à $\frac{1}{3}$ de celle de l'objet 3D secret. Enfin, Elsheh et Hamza ont montré qu'il était possible d'appliquer une compression supplémentaire aux *shares* générées à l'aide de la compression par codage entropique de Huffman [61] ou celle du logiciel ZLIB [45].

2.3.3 Méthodes avec préservation du format

Récemment, Tsai [134] et Lee *et al.* [134] ont proposé l'utilisation de l'insertion de données cachées pour reproduire l'approche proposée par Thien et Lin [129] en cachant les *shares* dans des images hôtes. Contrairement aux méthodes précédentes, le seuil minimum pour reconstruire l'objet 3D secret k n'est plus bloqué à 3. L'utilisation de l'insertion de données cachées pour insérer les *shares* dans des objets 3D hôtes requiert dans tous les cas une compression des données à insérer, mais aussi des objets 3D hôtes suffisamment denses pour avoir une capacité suffisante. Ainsi des méthodes de ré-échantillonnage et de subdivision de la surface sont utilisées afin d'augmenter la capacité naturelle de l'objet 3D hôte pour stocker la *share* à insérer.

Méthode de Tsai [134]

La méthode de Tsai [134] ne garde que la géométrie de l'objet 3D secret qui est préalablement compressée selon des méthodes de division de l'espace et de quantification. Ces données compressées sont ensuite partagées à l'aide de la méthode de Shamir [116] et insérées dans des objets 3D à l'aide d'une méthode d'insertion de données cachées suffisamment robustes aux modifications.

Méthode de Lee *et al.* [85]

La méthode proposée par Lee *et al.* [85] n'a pas pour but d'assurer la sécurité des objets 3D mais de permettre une transmission moins coûteuse pour afficher un ensemble de n objets 3D pour des solutions de *streaming* 3D. En effet, leur méthode propose de partager plusieurs objets 3D en basse qualité dans le but d'afficher les n objets 3D dès que k *shares* sont téléchargées en haute qualité. Ainsi, les *shares* sont insérées dans les objets 3D en haute qualité, plus exactement les codes de Reed et Solomon [107], permettant de reconstruire l'ensemble des n objets 3D en basse qualité. Les objets 3D originaux sont décimés jusqu'à un certain niveau puis compressés à l'aide de ZLIB [45]. Les données compressées sont ensuite traitées pour être transformées en codes de Reed et Solomon [107] qui peuvent être assimilés à des *shares* selon la méthode de Shamir [116].

La figure 2.5 représente les cas d'utilisation de la méthode proposée par Lee *et al.* [85]. La première ligne correspond au cas d'utilisation, où un objet 3D a été perdu (lost-a') et un autre a été altéré (ruined-c') durant la transmission. Cependant, grâce à la présence des objets 3D (b') et (d'), il est toujours possible de récupérer les objets 3D (a') et (c') avec une qualité plus faible (a") et (c"), respectivement. La seconde ligne présente le cas où les deux objets 3D (c') et (d') sont perdus, mais grâce aux objets (a') et (b') la méthode peut reconstruire les objets 3D (c") et (d") pour les remplacer. Enfin, la troisième ligne présente

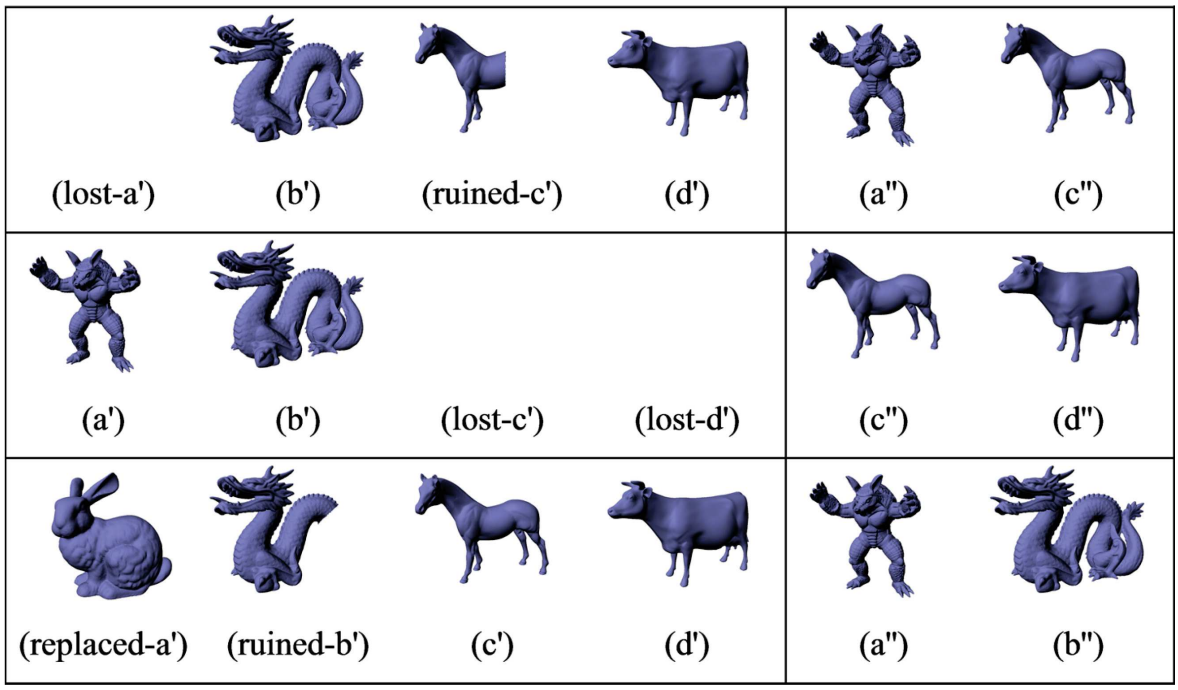


FIGURE 2.5 – Processus de reconstruction *Cross-Recovery* ($k = 2$, $n = 4$) proposé par Lee *et al.* [85].

le cas où un objet 3D a été malicieusement remplacé, ici l'objet 3D (a') a été échangé par l'objet 3D (replaced-a') représentant le modèle "Bunny". Malgré ce remplacement, et l'objet 3D déformé (ruined-b'), la méthode de Lee *et al.* permet de récupérer l'objet 3D (a").

2.3.4 Propriétés et applications

Plusieurs applications ont été proposées dans la littérature reprenant les grands usages du partage de secret, comme par exemple le système de redondance permettant de reconstruire l'objet 3D secret lorsque k *shares* sont présentes [49]. Anbarasi et Mala [4] ont proposé une extension de la méthode de Elsheh et Hamza [48] permettant de cacher plusieurs objets 3D au sein d'une même *share*. De plus, un système de vérification des *shares* est présent afin de détecter si certains utilisateurs ont fourni une fausse *share*.

Le tableau 2.1 récapitule les différentes propriétés de l'état de l'art du partage d'objet 3D secret. Ainsi les méthodes de partage correspondent aux méthodes de partage de secret binaire sur lesquelles sont développées celles pour les objets 3D. Nous notons l'utilisation des méthodes de Blakley [20], de Shamir [116], de Thien et Lin [128] et de Reed et Solomon [107]. Les méthodes se distinguent aussi par les données partagées, à savoir la géométrie et la connectivité [4, 49], uniquement la géométrie [134] ou bien des objets 3D décimés [85]. Les données partagées peuvent passer tout d'abord par un processus de compression binaire utilisant des codeurs entropiques comme Huffman [61] ou hybride [45]. Dans le cas où les objets 3D sont partagés par insertion de données cachées [85], des traitements comme le remaillage, la décimation ou la compression de la connectivité [110] peuvent servir à réduire le coût mémoire des données à partager. En fonction des méthodes de partage de secret utilisées, les paramètres k et n peuvent être limités selon leur utilisation au sein du partage d'objet 3D secret ou des corps finis, notamment pour [116]. Une des propriétés proposées par les méthodes de partage d'objet 3D secret est de pouvoir partager simultanément plusieurs objets 3D secrets [4, 85]. Une autre propriété issue du partage d'image secrète consiste à rendre les *shares* significa-

tives, c'est-à-dire qu'elles ne consistent pas seulement à être des données binaires illisibles mais qu'elles doivent représenter une information visuelle. Ainsi, Tsai [134] et Lee *et al.* [85] proposent d'utiliser un groupe d'objets 3D servant d'hôte pour leur méthode. Ils sont aussi, par extension, les seuls à proposer de fournir des objets 3D en sortie de leur méthode de partage préservant ainsi le format d'entrée. Contrairement aux approches de Elsheh et Hamza [48, 49] et Anbarasi et Mala [4], celles de Tsai [134] et Lee *et al.* [85], malgré les compressions, nécessitent des objets 3D remaillés avec plus de sommets pour pouvoir stocker les données partagées.

TABLEAU 2.1 – Comparatif des méthodes de partage d'objet 3D secret issues de l'état de l'art.

Propriétés ou paramètres	Elsheh et Hamza [49]	Anbarasi et Mala [4]	Tsai [134]	Lee *et al.* [85]
Méthode de partage	Blakley ou Thien & Lin	Shamir	Shamir	Reed-Solomon
Données partagées	Géométrie, Connectivité	Géométrie, Connectivité	Géométrie	Objets 3D décimés
Compression (Avant partage)	Sans perte (Huffman + ZLIB)	Sans perte (Huffman + ZLIB)	Subdivision spatiale	Décimation, EdgeBreaker, LZMA
k	3	$[\![2\,;\,n[\![$	$[\![2\,;\,n[\![$	$[\![2\,;\,n[\![$
n	P (premier)	P (premier)	255	Nombre d'objets 3D dans le groupe hôte
Multiple	Non	Oui	Non	Oui
Shares significatives	Non	Non	Oui	Oui
Préservation du format	Non	Non	Oui	Oui
Taille des *shares*	Identique ou $\frac{1}{k}$	Identique	Grande	Grande
Sortie	n fichiers binaires	n fichiers binaires	n objets 3D stéganographiés	n objets 3D stéganographiés

2.4 Aspect hiérarchique

Pour certaines applications, il peut être nécessaire de définir une hiérarchie parmi les utilisateurs afin de contrôler plus rigoureusement l'accès au contenu secret. Par exemple, au sein d'un établissement bancaire possédant un coffre-fort, il est naturel de s'attendre à ce que l'accès au coffre requière la présence d'employés de la banque dont certains appartiennent à la direction. Ainsi, nous pouvons imaginer un scénario où la présence d'au moins trois employés soit nécessaire pour l'ouverture du coffre et qu'un d'entre eux appartienne à la direction. Ou bien, dès lors que deux cadres sont présents, l'accès au coffre est possible. Cet exemple démontre qu'il est parfois nécessaire de recourir à une hiérarchisation des utilisateurs pour construire une structure de contrôle d'accès variant selon les besoins des cas d'utilisation. De telles conditions nécessitent d'étendre la définition du

partage de secret pour reconstruire le secret en fonction des utilisateurs présents à la reconstruction. Brièvement présenté en section 1.5.2 du chapitre 1, cet aspect hiérarchique a été introduit par Simmons [119] en 1988.

En section 2.4.1, nous définissons la notion de partage hiérarchique de secret (*Hierarchical Secret Sharing*). Nous présentons deux types de hiérarchies définies utiles par Simmons [119], à savoir les méthodes de partage hiérarchique compartimenté en section 2.4.2 et les méthodes de partage multi-niveaux en section 2.4.3. Enfin en section 2.4.4 nous détaillons la méthode de Tassa qui est une méthode de partage hiérarchique multi-niveaux et compartimenté [126] ainsi que la méthode de Belenkiy proposant de reconstruire le secret quand au moins k_ℓ utilisateurs de niveau ℓ ou plus sont présents [7].

2.4.1 Définition

Une méthode de partage hiérarchique (*Hierarchical Secret Sharing*) est définie par plusieurs paramètres noté $(\mathrm{L}, \mathbf{k}, \mathbf{n})$ où L est le nombre de niveaux dans la hiérarchie, $\mathbf{k} = \{k_\ell\}$ le nombre minimum d'utilisateurs par niveau pour la reconstruction $\ell \in [\![0\,;\,\mathrm{L}[\![$ et $\mathbf{n} = \{n_\ell\}$ le nombre maximum possible d'utilisateurs par niveau. Cette approche est apparue suite au besoin de limiter l'accès au secret à un ensemble autorisé d'utilisateurs. Ces utilisateurs appartiennent à des groupes d'utilisateurs formés en fonction de leur position, rôle ou niveau dans la hiérarchie :

$$\begin{cases} u \in \mathrm{U}, \\ \mathbb{L}(u) = \ell, \text{ où } \ell \in [\![0\,;\,\mathrm{L}[\![, \\ u \in \mathrm{U}_{\mathbb{L}(u)}, \\ \mathrm{U}_{\mathbb{L}(u)} \subseteq \mathrm{U}, \\ \mathrm{U} = \bigcup_{\ell=0}^{\mathrm{L}-1} \mathrm{U}_\ell, \end{cases} \tag{2.4}$$

où u est un utilisateur appartenant au groupe d'utilisateurs autorisés U et le groupe d'utilisateurs de niveau $\mathbb{L}(u)$ noté $\mathrm{U}_{\mathbb{L}(u)}$.

Dans la littérature, la hiérarchie définit la manière dont les utilisateurs et les groupes organisent ensemble la structure d'accès au secret. Les méthodes de partage hiérarchique de secret quant à elles correspondent à l'implémentation de ces hiérarchies à travers des approches de partage de secret. Plusieurs types de hiérarchies sont définies par leurs utilisations et objectifs [119].

2.4.2 Partage hiérarchique compartimenté

La première catégorie correspond aux méthodes de partage hiérarchique compartimenté (*Compartmented Hierarchical Secret Sharing* ou CHSS scheme) présentée par Simmons en 1988 [119]. Cette catégorie correspond à une structure d'accès comportant plusieurs groupes d'utilisateurs qui doivent atteindre un consensus dans chaque groupe d'utilisateurs afin de reconstruire le secret. Dans le cas de méthodes de partage hiérarchique compartimenté, les groupes d'utilisateurs sont définis de la manière suivante :

$$\begin{cases} u \in \mathrm{U}, \\ \mathbb{L}(u) = \ell, \text{ où } \ell \in [\![0\,;\,\mathrm{L}[\![, \\ u \in \mathrm{U}_{\mathbb{L}(u)}, \\ \forall \ell, \ell' \in [\![0\,;\,\mathrm{L}[\![, \ell \neq \ell' \iff \mathrm{U}_\ell \cap \mathrm{U}_{\ell'} = \emptyset. \end{cases} \tag{2.5}$$

Chaque utilisateur u appartient à un groupe U_ℓ, où $\ell = \mathbb{L}(u)$. Les différents groupes d'utilisateurs sont distincts les uns des autres. Dans une hiérarchie compartimentée, chaque groupe U_ℓ doit atteindre un certain seuil assigné au groupe pour reconstruire le secret. Comme les groupes sont distincts les uns des autres, les utilisateurs du groupe U_ℓ ne comptent pas pour le seuil d'un groupe $\mathrm{U}_{\ell'}$. C'est pourquoi, si au moins un seuil k_ℓ n'est pas atteint dans un groupe U_ℓ, alors le secret ne peut pas être révélé. La figure 2.6 illustre un exemple de méthode de partage hiérarchique compartimenté avec les paramètres $\mathrm{L} = 4$, $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 2, k_3 = 4)$ et $\mathbf{n} = (n_0, n_1, n_2, n_3)$. Ces paramètres indiquent que quatre groupes disjoints d'utilisateurs ont besoin d'obtenir chacun un consensus pour reconstruire le secret. Dans ce cas précis, la méthode de partage hiérarchique compartimenté requiert 2 utilisateurs de l'ensemble U_0, 3 de l'ensemble U_1, 2 de l'ensemble U_2 et 4 de l'ensemble U_3.

U_ℓ	k_ℓ
U_0	2
U_1	3
U_2	2
U_3	4

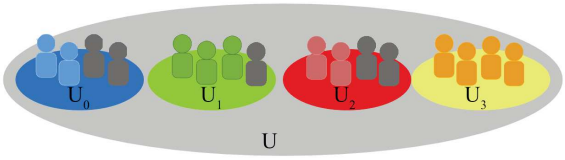


FIGURE 2.6 – Exemple de méthode de partage hiérarchique compartimenté d'un secret avec les paramètres $\mathrm{L} = 4, \mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 2, k_3 = 4)$ et $\mathbf{n} = (n_0, n_1, n_2, n_3)$.

L'implémentation la plus naïve d'une méthode de partage hiérarchique compartimenté consiste à utiliser séquentiellement une méthode de partage de secret. Tout d'abord, le secret S est partagé en utilisant les paramètres $(k = \mathrm{L}, n = \mathrm{L})$ afin de générer L *shares* $(s_0, s_1, \ldots, s_{\mathrm{L}-1})$. Puis, chacune de ces L *shares* est à nouveau partagée entre les membres d'un même groupe U_ℓ avec les paramètres (k_ℓ, n_ℓ) comme illustré en figure 2.7.

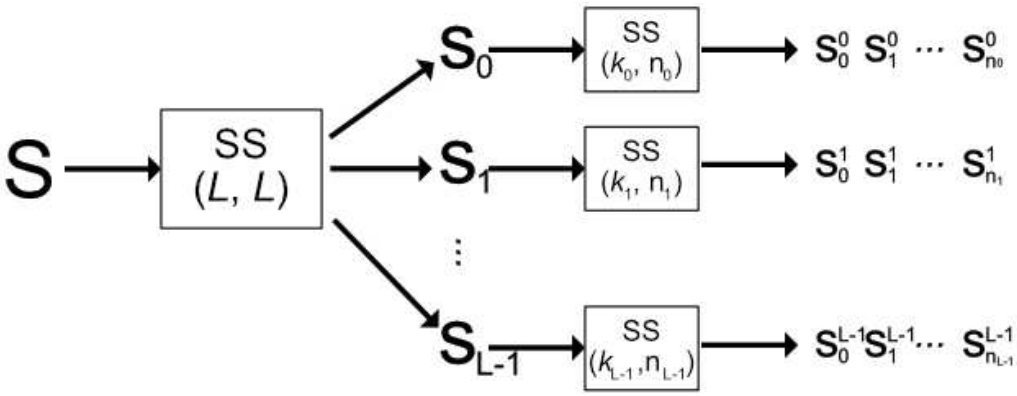


FIGURE 2.7 – Processus possible de partage hiérarchique compartimenté de secret.

2.4.3 Partage hiérarchique multi-niveaux

Une seconde catégorie regroupe les méthodes de partage hiérarchique multi-niveaux donnant une structure d'accès où les utilisateurs sont classés en fonction d'un niveau d'accès. Comme proposé par Simmons [119], le niveau définit la position des utilisateurs dans la hiérarchie et leur capacité à reconstruire le secret avec moins d'utilisateurs. Contrairement aux méthodes de partage hiérarchique compartimenté séparant les utilisateurs dans des groupes disjoints, les méthodes de partage hiérarchique multi-niveaux incluent les groupes les uns dans les autres :

$$\begin{cases} u \in \mathrm{U}, \\ \mathbb{L}(u) = \ell, \text{ où } \ell \in [\![0\,;\,\mathrm{L}[\![, \\ u \in \mathrm{U}_{\mathbb{L}(u)}, \\ \forall \ell, \ell' \in [\![0\,;\,\mathrm{L}[\![, \ell < \ell' \iff \mathrm{U}_\ell \subseteq \mathrm{U}_{\ell'}. \end{cases} \tag{2.6}$$

Une particularité des méthodes de partage hiérarchique multi-niveaux est qu'il existe un ordre entre les différents niveaux tel que :

$$\forall \ell, \ell' \in [\![0\,;\,\mathrm{L}[\![, \ell < \ell' \iff k_\ell < k_{\ell'}. \tag{2.7}$$

Nous notons le niveau le plus élevé 0 et le niveau le plus faible $(\mathrm{L}-1)$. Lors de l'étape de reconstruction la méthode doit d'abord vérifier si le seuil pour le niveau ℓ est atteint par la présence d'au moins k_ℓ utilisateurs appartenant au groupe U_ℓ. Si ce nombre d'utilisateurs est suffisant au niveau ℓ alors le secret est directement reconstruit grâce à cet ensemble d'utilisateurs. Sinon, les utilisateurs de niveau ℓ sont considérés comme des utilisateurs de niveau $(\ell+1)$ et peuvent ainsi être comptabilisés pour le seuil $k_{\ell+1}$.

La figure 2.8 illustre une hiérarchie multi-niveaux, les utilisateurs de plus haut niveau peuvent reconstruire entre eux plus rapidement le secret. Par exemple, les utilisateurs appartenant au groupe U_0 comparés aux utilisateurs appartenant aux groupes U_1, U_2 ou U_3 peuvent reconstruire le secret avec moins d'utilisateurs du groupe U_0. Cependant, cela fonctionne uniquement avec des utilisateurs de même niveau, c'est-à-dire appartenant au même groupe. Ainsi, si le nombre d'utilisateurs de U_0 présents à la reconstruction est inférieur au seuil k_0, alors le secret reste protégé. Néanmoins, ils peuvent participer à la reconstruction avec des utilisateurs de niveau plus faible.

U_ℓ	k_ℓ
U_0	2
U_1	3
U_2	5
U_3	8

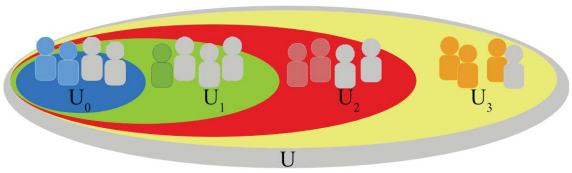


FIGURE 2.8 – Exemple de méthode de partage hiérarchique multi-niveaux d'un secret avec les paramètres $\mathrm{L} = 4, \mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 5, k_3 = 8)$ et $\mathbf{n} = (n_0, n_1, n_2, n_3)$.

2.4.4 Travaux précédents

Dans cette section, nous détaillons deux méthodes de partage hiérarchique de secret proposant des hiérarchies différentes et utilisant comme base la méthode de partage de secret de Shamir [116], à savoir la méthode de Tassa [126] et la méthode de Belenkiy [7].

Méthode de Tassa [126]

En 2007, Tassa [126] a proposé une méthode de partage hiérarchique de secret mélangeant les deux types de hiérarchies présentées en section 2.4.2 et section 2.4.3 et en utilisant la méthode de partage de secret de Shamir [116]. Le but de ce nouveau type de hiérarchie est de permettre la reconstruction quand suffisamment d'utilisateurs à tous les niveaux de la hiérarchie participent à cette reconstruction, c'est-à-dire un nombre minimum d'utilisateurs par niveau. La particularité est que les utilisateurs de plus haut niveaux participent aux consensus des niveaux inférieurs. La méthode proposée par Tassa

est basée sur un polynôme de degré $(k_{L-1}-1)$ (cf. Eq. (1.34)) et cache le secret S dans le coefficient a_0 du polynôme utilisé. La méthode de Tassa transmet alors la paire d'informations $(u_{\ell,i}, g_\ell(u_{\ell,i}))$ au i-ème utilisateur appartenant au niveau $\ell \in [\![0\,;\,L]\!]$, où $i \in [\![0\,;\,n_\ell]\!]$, $\mathbb{L}(u_{\ell,i}=\ell)$ est le niveau de l'utilisateur $u_{\ell,i}$ et g_ℓ le polynôme de Tassa associé au niveau ℓ. Le polynôme de Tassa g_ℓ associé au niveau ℓ est le polynôme de Shamir f dérivé $k_{\ell-1}$ fois et noté $f^{k_{\ell-1}}$:

$$g_\ell(u) = f^{k_{\ell-1}}(u) = \frac{d^{k_{\ell-1}} f}{du^{k_{\ell-1}}}(u), \tag{2.8}$$

avec $k_{-1}=0$.

Par exemple, pour trois niveaux de hiérarchie avec les seuils $\mathbf{k}=(k_0,k_1,k_2)$, pour reconstruire le secret, il est alors nécessaire de regrouper au moins k_0 utilisateurs de niveau 0, k_1 utilisateurs de niveau 1 et k_2 utilisateurs de niveau 2. Les polynômes utilisés pour chaque niveau de la hiérarchie sont :

$$\begin{cases} \forall u \in \mathrm{U}, \\ f(u) = \sum_{i=0}^{k_2-1} a_i \times u^i, \\ \mathbb{L}(u)=0 \Rightarrow g_0(u) = f^{k_{-1}}(u), \\ \mathbb{L}(u)=1 \Rightarrow g_1(u) = f^{k_0}(u), \\ \mathbb{L}(u)=2 \Rightarrow g_2(u) = f^{k_1}(u). \end{cases} \tag{2.9}$$

Si nous prenons $\mathbf{k}=(2,3,4)$ pour les seuils k_0, k_1 et k_2, en remplaçant les variables par leurs valeurs, alors les polynômes obtenus sont :

$$\begin{cases} \forall u \in \mathrm{U}, \\ f(u) = a_0 + a_1 \times u + a_2 \times u^2 + a_3 \times u^3, \\ g_0(u) = f^{k_{(-1)}}(u) = f^0(u) = f(u), \\ g_1(u) = f^{k_0}(u) = f^2(u) = \frac{d^2 f}{du^2}(u) = 2 \times a_2 + 6 \times a_3 \times u, \\ g_2(u) = f^{k_1}(u) = f^3(u) = \frac{d^3 f}{du^3}(u) = 6 \times a_3. \end{cases} \tag{2.10}$$

Le polynôme g_0 utilisé pour les utilisateurs de niveau 0 est égal au polynôme de Shamir f de degré $(k_{L-1}-1)$ sans dérivation, tandis que le polynôme g_1 destiné aux utilisateurs de niveau 1 est égal à f dérivé deux fois selon l'équation (2.8), car $k_0=2$. Pour le polynôme utilisé pour les utilisateurs de niveau 2, ce dernier est égal à f dérivé 3 fois, selon l'équation (2.8), car $k_1=3$.

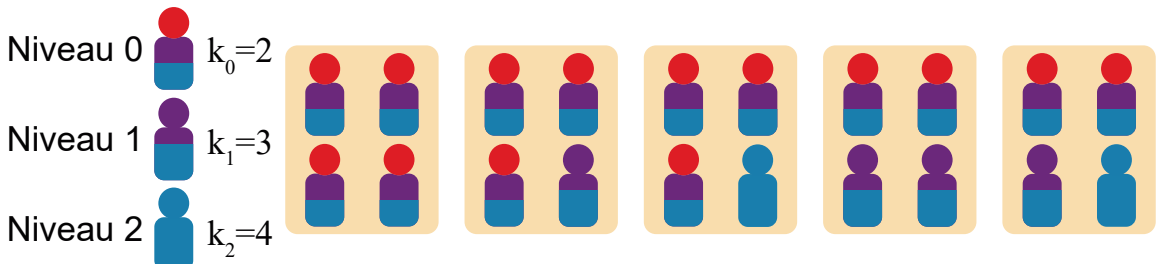


FIGURE 2.9 – Liste de groupes pouvant reconstruire le secret partagé avec la méthode de Tassa [126] avec les paramètres $L=3$ et $\mathbf{k}=(2,3,4)$.

La figure 2.9 illustre la liste des groupes pouvant reconstruire le secret selon la méthode de Tassa [126] avec les seuils $\mathbf{k}=(k_0=2, k_1=3, k_2=4)$. Les utilisateurs de niveaux

0, 1, et 2 sont nécessaires pour la reconstruction du secret selon la hiérarchie proposée par Tassa. Cependant, comme illustré en figure 2.9, les utilisateurs de niveau 0 peuvent agir comme des utilisateurs de niveau 1 et niveau 2 pour participer à la reconstruction. Le dernier groupe d'utilisateurs illustré en figure 2.9 représente la configuration minimale nécessaire à la reconstruction du secret telle que deux utilisateurs de niveau 0, un utilisateur de niveau 1 et un utilisateur de niveau 2 soient présents. Alors que cette hiérarchie requiert deux utilisateurs de niveau 0, trois utilisateurs de niveau 1 et quatre utilisateurs de niveau 2, cette configuration est possible, car les utilisateurs de niveau 0 comptent aussi comme des utilisateurs de niveaux 1 et 2. Les utilisateurs de niveau 1 comptent aussi comme des utilisateurs de niveau 2. En jouant sur le nombre d'utilisateurs par niveau, il est possible de rendre la présence de certains utilisateurs nécessaire à la reconstruction. Pour reconstruire le secret, il est nécessaire d'utiliser une interpolation de Hermite-Birkhoff [114]. L'interpolation de Hermite-Birkhoff permet de déterminer l'unique solution d'un système linéaire formé par un ensemble réduit et valide d'utilisateurs pour la reconstruction. Formellement, quand une méthode de partage hiérarchique de secret est utilisée selon Tassa [126], une relation matricielle pour chaque groupe d'utilisateurs autorisés à reconstruire le secret est obtenue telle que :

$$\mathrm{X} = \begin{bmatrix} 1 & u_{i_0} & u_{i_0}^2 & u_{i_0}^3 \\ 1 & u_{i_1} & u_{i_1}^2 & u_{i_1}^3 \\ 0 & 0 & 2 & 6u_{i_2} \\ 0 & 0 & 0 & 6 \end{bmatrix}, \mathrm{A}^{\mathrm{T}} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}, \tag{2.11}$$

$$\mathrm{X} \times \mathrm{A}^{\mathrm{T}} = b \Rightarrow \mathrm{X} \times \mathrm{A}^{\mathrm{T}} = \begin{bmatrix} g_0(u_{i_0}) \\ g_0(u_{i_1}) \\ g_1(u_{i_2}) \\ g_2(u_{i_3}) \end{bmatrix}, \tag{2.12}$$

où i_0, i_1, i_2 et $i_3 \in [\![0\,;\,|\mathbf{n}|[\![$ avec $|\mathbf{n}| = |\bigcup_{\ell=0}^{\mathrm{L}-1} \mathrm{U}_\ell| = \sum_{\ell=0}^{\mathrm{L}-1} n_\ell$ le nombre d'utilisateurs (tous groupes confondus) participant au partage du secret, u_{i_0}, u_{i_1} deux utilisateurs de niveau 0, u_{i_2} un utilisateur de niveau 1 et u_{i_3} un utilisateur de niveau 2, A est le vecteur de coefficients du polynôme, X est la matrice des polynômes calculés (sans coefficients) et dérivés (ou non) pour chaque utilisateur appartenant au groupe de reconstruction et b le résultat des polynômes pour chaque utilisateur. Le groupe d'utilisateurs $(u_{i_0}, u_{i_1}, u_{i_2}, u_{i_3})$ est un groupe pouvant reconstruire le secret selon la méthode de Tassa [126].

En résolvant le système linéaire $\mathrm{X} \times \mathrm{A}^{\mathrm{T}} = b$, l'ensemble des coefficients A utilisés dans les polynômes peut en être déduit.

Méthode de Belenkiy [7]

La méthode de partage hiérarchique de secret de Belenkiy est basée sur celle de Tassa afin de produire une nouvelle hiérarchie multi-niveaux moins contraignante que la précédente. Dans cette hiérarchie, dès que le nombre d'utilisateurs pour un niveau ℓ atteint la valeur associée au seuil k_ℓ, alors le secret peut être reconstruit. Pour réaliser cette hiérarchie, Belenkiy a proposé de cacher le secret dans le dernier coefficient du polynôme de Shamir a_{k-1} au lieu de a_0 (*cf.* Eq. (1.34)). De plus, puisque les utilisateurs de niveaux supérieurs reçoivent des dérivés du polynôme de Shamir, ils sont alors capables de commencer la reconstruction du secret dès que leur nombre franchit le seuil associé à leur niveau. Si le seuil associé k_ℓ n'est pas atteint, alors les utilisateurs de niveau ℓ peuvent participer à la reconstruction pour le niveau suivant $(\ell+1)$ avec le seuil associé $k_{(\ell+1)}$. Nous notons

h_ℓ le polynôme de Belenkiy utilisé pour distribuer les *shares* pour les utilisateurs de niveau ℓ. Ainsi, en reprenant le même exemple que celui présenté précédemment avec les paramètres $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 4)$, un polynôme de degré 3 (car $k_{L-1} = 4$) est utilisé. La valeur du coefficient $a_{(k_{L-1}-1)} = a_3$ est assignée à celle du secret et pour chaque niveau ℓ, le polynôme est dérivé $(k_{L-1} - k_\ell)$ fois :

$$\begin{cases} \forall u \in \mathbb{U}, \\ \mathbb{L}(u) = \ell \in [\![0\,;\,L[\![\\ \Delta = k_{L-1} - k_\ell, \\ h_\ell(u) = f^\Delta(u) = \frac{d^\Delta f}{du^\Delta}(u). \end{cases} \tag{2.13}$$

Dans ce cas, il est possible pour un groupe de k_ℓ utilisateurs, où ℓ est le niveau du seuil franchi, de reconstruire le dernier coefficient $a_{(k_{L-1}-1)}$ contenant le secret comme illustré en figure 2.10. La figure 2.10 représente l'ensemble de groupes d'utilisateurs pouvant re-

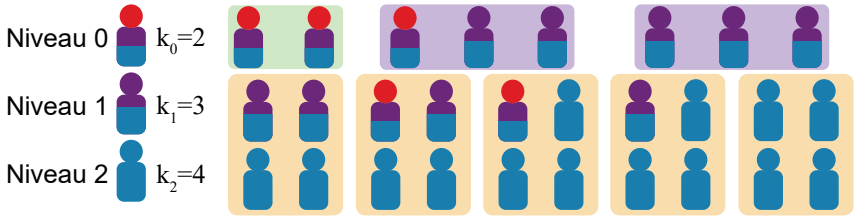


FIGURE 2.10 – Liste de groupes pouvant reconstruire le secret partagé avec la méthode de Belenkiy [7] avec les paramètres $L = 3$ et $\mathbf{k} = (2, 3, 4)$.

construire le secret selon la méthode de Belenkiy pour trois niveaux de hiérarchie et les seuils $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 4)$. Les utilisateurs appartenant à U_0 peuvent reconstruire le secret avec au moins deux membres, tandis que les utilisateurs de U_1 ont besoin d'être au moins trois membres, et enfin les utilisateurs du groupe U_2 ont besoin d'être au moins quatre membres pour reconstruire le secret selon la hiérarchie proposée par Belenkiy. Comme illustré en figure 2.10, si un utilisateur de niveau 0 est seul, alors il ne peut pas reconstruire le secret. Cependant l'utilisateur peut agir en tant qu'utilisateur de niveau 1 et de niveau 2 pour participer à la reconstruction avec les autres utilisateurs de ces niveaux. Le dernier groupe illustré en figure 2.10 représente la configuration minimale avec quatre utilisateurs de niveau 2 qui sont capables de reconstruire le secret.

Grâce à l'approche multi-niveaux, les utilisateurs de niveaux supérieurs peuvent reconstruire le secret plus rapidement ou sinon ils peuvent aider à reconstruire le secret avec les utilisateurs de niveaux inférieurs. Pour reconstruire le secret partagé utilisant la méthode de Belenkiy, il est nécessaire, comme pour la méthode de Tassa, d'utiliser une interpolation de Hermite-Bikhoff [114]. Avec le même exemple que celui en figure 2.10, les utilisateurs de niveau 0 (u_{i_0} et $u_{i_1} \in U_0$) peuvent reconstruire le secret grâce à leur paire d'informations $(u_{i_0}, h_0(u_{i_0}))$ et $(u_{i_1}, h_0(u_{i_1}))$ selon la relation matricielle suivante :

$$X = \begin{bmatrix} 2 & 6u_{i_0} \\ 2 & 6u_{i_1} \end{bmatrix}, A^T = \begin{bmatrix} a_2 \\ a_3 \end{bmatrix}, \tag{2.14}$$

$$X \times A^T = b \Rightarrow X \times A^T = \begin{bmatrix} h_0(u_{i_0}) \\ h_0(u_{i_1}) \end{bmatrix}. \tag{2.15}$$

En solvant le système linéaire $X \times A^T = b$, une unique solution est possible pour les coefficients a_2 et a_3, où a_3 contient la valeur du secret.

2.5 Conclusion

Dans ce chapitre, nous avons présenté le partage d'image secrète et ses différentes approches pour le partage de secret visuel. Nous avons ainsi présenté des propriétés que peuvent avoir les méthodes de partage d'image secrète qui sont ensuite utilisées dans nos travaux. Nous avons décrit les différentes approches existantes du partage d'objets 3D et détaillé l'état de l'art particulièrement restreint.

Enfin, nous avons poursuivi la présentation des méthodes de partage de secret en détaillant le partage hiérarchique de secret. De plus, nous avons présenté deux méthodes de partage hiérarchique de secret sur lesquelles nous nous sommes appuyés pour développer les travaux présentés dans ce manuscrit.

Dans ce manuscrit, nous travaillons sur des méthodes de partage d'image secrète basé sur des polynômes. Ainsi, dans le chapitre 4 nous présentons une de nos contributions en matière de partage d'image secrète, ainsi qu'une contribution sur le partage d'objet 3D secret. Tandis que dans le chapitre 5 nous présentons une de nos contributions en matière de partage hiérarchique de régions d'intérêt au sein d'une image issue de réseaux sociaux et une contribution sur le partage hiérarchique d'objet 3D secret avec une nouvelle hiérarchie.

Deuxième partie

Contributions

Chapitre 3

Chiffrement sélectif d'objet 3D

Sommaire

3.1 Introduction

Dans ce chapitre, nous proposons une nouvelle méthode de chiffrement sélectif des objets 3D. Pour cela, nous nous sommes inspirés des travaux effectués autour du contrôle d'accès des images, des vidéos et des objets 3D présentés en section 1.4 du chapitre 1. Nous présentons notre approche de chiffrement sélectif d'objets 3D basée sur le chiffrement des coordonnées des sommets d'un nuage de points 3D. L'idée principale est de chiffrer les coordonnées en sélectionnant spécifiquement les bits à chiffrer.

Ce chiffrement possède des propriétés essentielles, telles que :

— il préserve le format du fichier initial ;

— il ne dépend pas de la connectivité d'un maillage ;

— il permet à l'utilisateur de choisir le niveau de confidentialité de l'objet 3D chiffré.

Nous présentons notre contribution, à savoir notre méthode de chiffrement sélectif des objets 3D qui est décrite avec les spécifications nécessaires dans la section 3.2. Dans la section 3.3, nous évaluons ensuite les résultats expérimentaux obtenus d'un point de vue statistique à partir de métriques présentées en section 1.2.2 du chapitre 1. Enfin, la section 3.4 conclut le chapitre et présente quelques pistes futures de recherche.

3.2 Méthode proposée

Dans notre méthode, nous proposons de chiffrer seulement une partie de la représentation binaire des coordonnées des sommets de l'objet 3D afin de pouvoir contrôler les impacts géométriques du chiffrement.

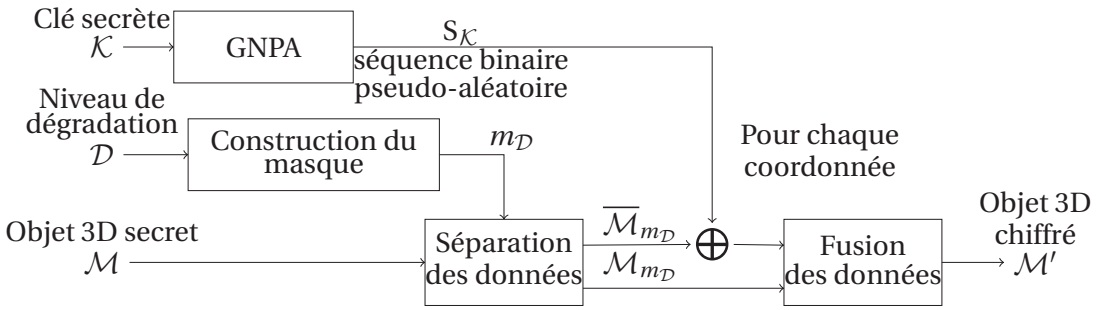


FIGURE 3.1 – Processus de chiffrement sélectif 3D.

Ainsi, comme illustrée en figure 3.1, notre méthode de chiffrement requiert en entrée 2 paramètres pour chiffrer l'objet 3D secret $\mathcal{M}$, à savoir une clé secrète $\mathcal{K}$ et un niveau de dégradation $\mathcal{D}$. La méthode proposée est divisée en deux tâches principales, à savoir la sélection des bits à chiffrer au sein des coordonnées des sommets de l'objet 3D secret et le chiffrement de ces mêmes données. Notre méthode débute par la construction d'un masque de synchronisation noté $m_{\mathcal{D}}$ qui, en fonction du niveau de dégradation $\mathcal{D}$, va sélectionner une partie des données géométriques de l'objet 3D à chiffrer $\overline{\mathcal{M}}_{m_{\mathcal{D}}}$ et laisser une partie à préserver en clair $\mathcal{M}_{m_{\mathcal{D}}}$. Pendant ce temps, la clé secrète est utilisée pour initialiser un Générateur de Nombres Pseudo-Aléatoire (GNPA) dans le but de produire une séquence binaire pseudo-aléatoire. À l'aide du masque de synchronisation construit, certains bits des coordonnées des sommets de l'objet 3D sont sélectionnés et chiffrés à l'aide de la séquence binaire pseudo-aléatoire en appliquant entre eux une opération de "Ou exclusif" (XOR). Cela s'apparente à du chiffrement par flot. Une fois les bits sélectionnés chiffrés, ils se substituent aux bits originaux pour construire l'objet 3D chiffré.

Nous rappelons tout d'abord la notation des valeurs flottantes servant à représenter les coordonnées des sommets dans la section 3.2.1. En section 3.2.2, nous présentons l'approche utilisée pour sélectionner les bits à chiffrer au sein des coordonnées à l'aide du niveau de dégradation $\mathcal{D}$ choisi. Ensuite, nous décrivons le chiffrement des données sélectionnées préalablement en fonction de la clé secrète $\mathcal{K}$ dans la section 3.2.3. Enfin, nous terminons par le processus de déchiffrement d'un objet 3D protégé par notre méthode de chiffrement sélectif en section 3.2.4.

3.2.1 Représentation des valeurs flottantes

Les valeurs flottantes sont représentées de manières bien diverses sur les machines modernes [54, 62]. Pour rappel, les coordonnées d'un sommet dans un objet 3D sont définies par des valeurs flottantes. La norme *IEEE 754* [62] est la représentation la plus commune des valeurs flottantes. Elle définit plusieurs niveaux de précision pour les valeurs flottantes reproduisant la même structure générale contenant trois données distinctes, à savoir le signe s, l'exposant e et la mantisse m.

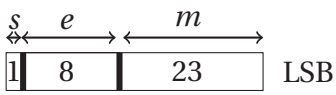


FIGURE 3.2 – Représentation d'une valeur flottante sur 32 bits selon la norme *IEEE 754* [62].

La figure 3.2 présente les trois informations constituant une valeur flottante. Chaque partie possède une quantité spécifique de bits : le bit de signe (en tant que bit de poids fort) permet de déterminer s'il s'agit d'un nombre positif (0) ou négatif (1). Les 8 bits suivants de l'exposant représentent l'exposant décalé. Enfin, les 23 bits restants correspondent à la mantisse. La valeur flottante représentée par cette représentation binaire est calculée par :

$$v = (-1)^s \times m \times 2^{e-127}. \tag{3.1}$$

Dans la suite de ce manuscrit, nous nous basons sur la précision simple de la norme *IEEE 754*. Cependant, notre méthode de chiffrement sélectif peut être adaptée à n'importe quelle précision. Le tableau 3.1 récapitule les différents niveaux de précision proposés à ce jour dans la norme *IEEE 754* [62].

TABLEAU 3.1 – Récapitulatif des différentes précisions des valeurs flottantes selon la norme IEEE 754 [62].

Nom	Taille (en bits)	Taille exposant (en bits)	Taille mantisse (en bits)
Demi-précision	16	5	10
Simple-précision	32	9	23
Double-précision	64	11	52
Quadruple-précision	128	15	112
Octuple-précision	256	19	236

3.2.2 Sélection des données à chiffrer

Cette étape consiste à réaliser deux tâches particulières. Tout d'abord, la méthode génère une séquence binaire pseudo-aléatoire $S_{\mathcal{K}}$ à partir d'un GNPA et de la clé secrète $\mathcal{K}$

fournie par l'utilisateur. Ces valeurs pseudo-aléatoires sont ensuite utilisées séquentiellement lors du chiffrement des coordonnées. Notre méthode construit alors un masque de chiffrement basé sur le niveau de dégradation $\mathcal{D}$. Cette variable définit le niveau du chiffrement sélectif. Plus le niveau de dégradation est petit, plus les déformations géométriques sont faibles sur l'objet 3D chiffré. Le masque permet ainsi de localiser la zone de la représentation binaire des valeurs flottantes à chiffrer et de préciser combien de bits doivent être chiffrés. Pour cette méthode, nous avons développé une stratégie pour générer le masque $m_{\mathcal{D}}$ consistant à déplacer une fenêtre glissante sur les bits de la valeur flottante pour sélectionner ceux à chiffrer. Nous notons cette stratégie, *D-Sliding Window mask* (ou masque *D-SW*) où le niveau de dégradation est défini par la paire :

$$\mathcal{D} =< p, l >, \tag{3.2}$$

où le paramètre $p \in [\![0\ ;\ 22]\!]$ du niveau de dégradation $\mathcal{D}$ indique la position du premier bit à chiffrer dans le masque de chiffrement à partir du bit de poids faible, tandis que $l \in [\![1\ ;\ p+1]\!]$ définit le nombre de bits à chiffrer.

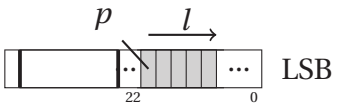


FIGURE 3.3 – Masque de sélection des bits à chiffrer en fonction du niveau de dégradation $\mathcal{D}$ choisi.

La figure 3.3 illustre l'application du masque *D-SW* permettant aux utilisateurs de sélectionner les bits à chiffrer. Plus la valeur de p est grande, plus les distorsions géométriques induites dans l'objet 3D chiffré seront importantes. De plus, en fixant la contrainte $l = p+1$, il est possible de créer un masque sélectionnant tous les bits les moins significatifs (LSB) pour le chiffrement. Nous notons ce masque *D-LSB*, où $\mathcal{D} =< p, p+1 >$.

3.2.3 Chiffrement sélectif des coordonnées des sommets

Durant l'étape de chiffrement de l'objet 3D, comme illustré en figure 3.1, pour tous les sommets de l'objet 3D, les coordonnées sont séparées en deux valeurs grâce au masque $m_{\mathcal{D}}$. Par exemple pour la coordonnée x_i du i-ème sommet :

— $x_{i_{m_{\mathcal{D}}}}$, est la partie préservée de la coordonnée x_i telle que : $x_{i_{m_{\mathcal{D}}}} = x_i \wedge \neg m_{\mathcal{D}}$;

— $\overline{x_{i_{m_{\mathcal{D}}}}}$, est la partie à chiffrer de la coordonnée x_i telle que : $\overline{x_{i_{m_{\mathcal{D}}}}} = x_i \wedge m_{\mathcal{D}}$,

où, $. \wedge .$ est l'opérateur binaire "Et" (AND) et $\neg.$ l'opérateur binaire unaire "Négation" (NOT) inversant la valeur de chaque bit.
À partir de la séquence binaire pseudo-aléatoire $S_{\mathcal{K}}$ générée à partir du GNPA, une valeur r est extraite de la séquence pseudo-aléatoire $S_{\mathcal{K}}$ et le masque $m_{\mathcal{D}}$ est appliqué tel que :

$$r_{m_{\mathcal{D}}} = r \wedge m_{\mathcal{D}}. \tag{3.3}$$

Nous utilisons alors une approche de chiffrement par flot présentée en section 1.4.2 pour chiffrer $\overline{x_{i_{m_{\mathcal{D}}}}}$ telle que la valeur intermédiaire $e = r_{m_{\mathcal{D}}}$ est calculée. Le résultat du chiffrement se substitue à la part à chiffrer $\overline{x_{i_{m_{\mathcal{D}}}}}$ tout en conservant la part en clair $x_{i_{m_{\mathcal{D}}}}$ pour fournir la coordonnée chiffrée sélectivement x'_i telle que :

$$x'i = e \vee x_{i_{m_{\mathcal{D}}}}, \tag{3.4}$$

où $. \vee .$ est l'opérateur binaire "Ou" (OR).

Le même processus est répété pour les coordonnées y_i et z_i du sommet traité en utilisant de nouvelles valeurs issues de la séquence pseudo-aléatoire $S_{\mathcal{K}}$. Ces différentes opérations, constituant le chiffrement sélectif, sont résumées par :

$$c'_i = ((c_i \oplus r) \wedge m_{\mathcal{D}}) \vee (c_i \wedge \neg m_{\mathcal{D}}), \tag{3.5}$$

où c'_i est la variable à remplacer par les coordonnées du i-ème sommet et $oplus$ l'opérateur binaire "Ou exclusif" (XOR). Une fois que toutes les coordonnées de tous les sommets sont chiffrés, l'objet 3D chiffré, noté $\mathcal{M}'$, est construit.

3.2.4 Déchiffrement d'un objet 3D

Pour déchiffrer l'objet 3D chiffré $\mathcal{M}'$, nous ré-appliquons l'algorithme de chiffrement présenté en figure 3.1. En effet, en utilisant la même clé secrète $\mathcal{K}$ et le même niveau de dégradation $\mathcal{D}$ en entrée, nous pouvons alors déchiffrer l'objet 3D chiffré et obtenir un objet 3D en clair correspond exactement à l'objet 3D original sans perte.

3.3 Résultats expérimentaux

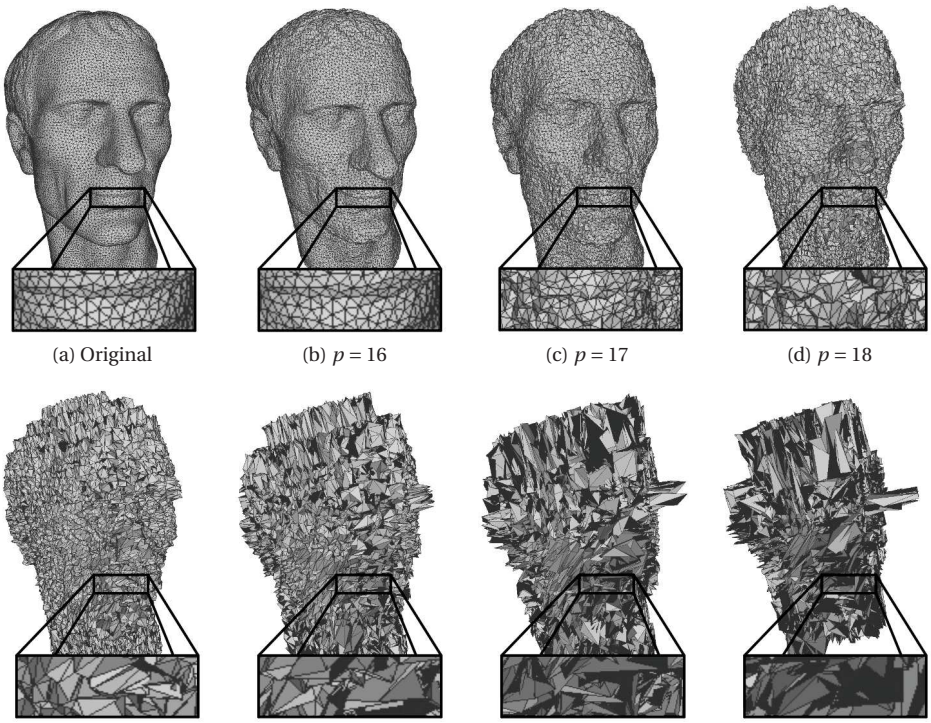
Dans cette section, nous présentons des résultats expérimentaux de notre méthode de chiffrement sélectif d'objets 3D. Tout d'abord, nous présentons un cas d'application de notre méthode sur un objet 3D pour deux niveaux de dégradation en section 3.3.1. Nous réalisons ensuite en section 3.3.2 une analyse statistique ainsi qu'une comparaison des résultats générés. Nous analysons alors la sécurité de notre méthode face à diverses attaques géométriques en section 3.3.3. Enfin, dans la section 3.3.4 nous comparons notre méthode de chiffrement sélectif 3D à des méthodes de l'état de l'art présentées en section 1.4.3 du chapitre 1.

3.3.1 Application

La figure 3.4 présente les résultats obtenus par notre méthode utilisant le masque *D-SW* avec le niveau de dégradation $\mathcal{D} =< p, l >$ avec $p \in [\![16 ; 22]\!]$, où p est la position du premier bit à chiffrer et $l = 1$ le nombre de bits à chiffrer.

À partir de la figure 3.4.a représentant l'objet 3D original, les figures 3.4.b-i représentent les objets 3D chiffrés avec un niveau de dégradation $\mathcal{D}$ spécifique. Nous remarquons que notre méthode autorise les utilisateurs à chiffrer visuellement plus ou moins fortement un objet 3D afin qu'il reste reconnaissable, mais géométriquement déformé. Nous notons aussi qu'à partir d'un niveau de dégradation spécifique $p = 20$ en figure 3.4.f, il n'est plus possible de reconnaître visuellement le contenu de l'objet 3D chiffré. En effet, pour cet exemple, avec p inférieur ou égal à 18 nous obtenons un chiffrement transparent, avec p entre 19 et 20 nous avons un chiffrement suffisant et avec p au-delà de 21 nous obtenons une confidentialité visuelle. Ainsi, comme illustré en figure 3.4, même si nous chiffrons une infime partie de la géométrie de l'objet 3D, soit 1 bit par coordonnée et par extension 3 bits sur les 96 bits composant les coordonnées x, y et z de chaque sommet, les objets 3D chiffrés générés par notre méthode ont leur contenu protégé visuellement.

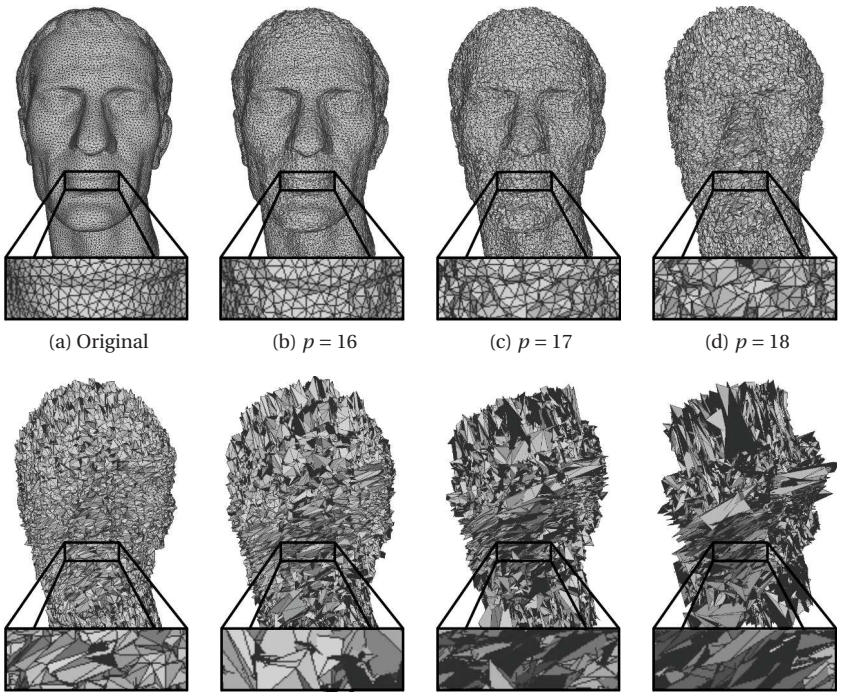
La figure 3.5 illustre l'utilisation de notre méthode avec un masque *D-LSB* où le niveau de dégradation $\mathcal{D} =< p, p+1 >$ tel que $p \in [\![17 ; 23[\![$. A partir de la figure 3.5.a représentant l'objet 3D secret original, les figures 3.5.b-i représentent des objets 3D chiffrés sélectivement selon un niveau de dégradation $\mathcal{D}$ spécifique. Nous notons que l'utilisation du



(a) Original (b) $p = 16$ (c) $p = 17$ (d) $p = 18$

(e) $p = 19$ (f) $p = 20$ (g) $p = 21$ (h) $p = 22$

FIGURE 3.4 – Chiffrement sélectif en fonction du niveau de dégradation $\mathcal{D} =< p, 1 >$.



(a) Original (b) $p = 16$ (c) $p = 17$ (d) $p = 18$

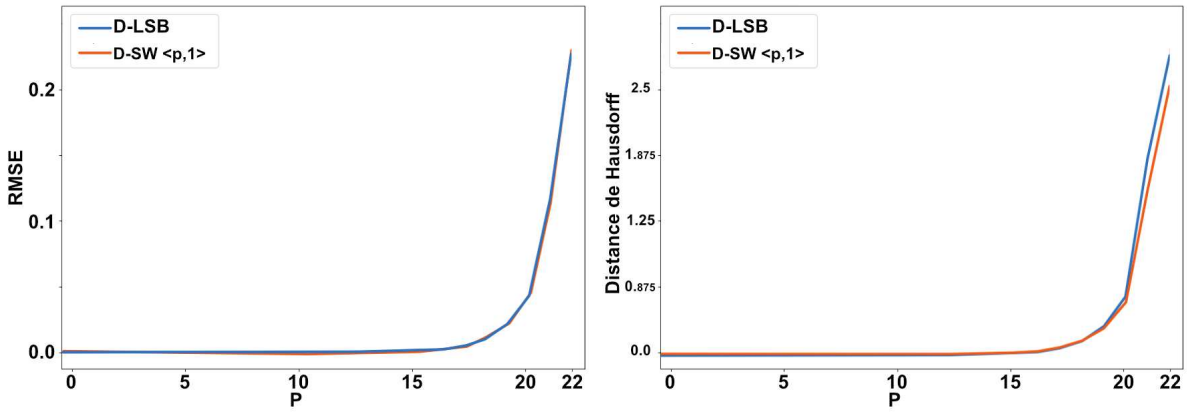
(e) $p = 19$ (f) $p = 20$ (g) $p = 21$ (h) $p = 22$

FIGURE 3.5 – Chiffrement sélectif en fonction du niveau de dégradation $\mathcal{D} =< p, p+1 >$.

masque *D-LSB* consistant à chiffrer les p bits les moins significatifs nous permet d'obtenir des résultats visuellement similaires à ceux présentée en figure 3.4. De plus, nous notons aussi expérimentalement que les distorsions géométriques induites par le chiffrement sélectif commencent à être visuellement détectables à partir de $p = 17$.

3.3.2 Analyse statistique

Afin d'analyser notre approche, nous avons réalisé plusieurs expérimentations permettant de comparer les objets 3D chiffrés sélectivement selon les deux stratégies de masque. Nous utilisons pour ces expérimentations la base de données de maillage de Princeton [33] contenant 380 objets. Les métriques utilisées pour comparer les objets 3D sont le RMSE et la distance de Hausdorff (HD) présentées en section 1.2 du chapitre 1.



(a) RMSE (b) Distance d'Hausdorff

FIGURE 3.6 – Résultats des métriques (*RMSE* et *HD*) sur la base de données de maillages de Princeton [33] en fonction du paramètre p, pour les masques *D-LSB* et *D-SW*.

La figure 3.6 présente les dégradations des objets 3D chiffrés en termes de RMSE et de HD en fonction du paramètre p pour les deux types de masque, à savoir les masques *D-LSB* et *D-SW*. Précédemment, nous avons observé visuellement que nous sommes capables, en fonction du paramètre p, de générer un objet 3D chiffré sélectivement selon la classification donnée en section 1.4, à savoir d'une confidentialité visuelle totale à un chiffrement transparent, en passant par un chiffrement suffisant.

Pour le masque *D-LSB*, nous constatons que les valeurs de RMSE et de HD, représentées par les courbes bleues en figure 3.6.a, et en figure 3.6.b respectivement, sont quasi nulles pour $p \in [\![1 ; 17]\!]$, ce qui signifie que les distorsions géométriques sont très faibles. Lorsque $p \in [\![18 ; 22]\!]$, les valeurs des deux métriques augmentent de manière exponentielle jusqu'à atteindre une valeur aux alentours de 0.22 pour le RMSE et 2.6 pour la distance de Hausdorff. En même temps, les distorsions géométriques deviennent de plus en plus présentes jusqu'à rendre totalement confidentiel l'objet 3D pour $p = 22$. Ainsi, les valeurs les plus élevées pour le RMSE et la distance de Hausdorff correspondent au niveau de confidentialité le plus élevé.

Entre-temps pour le masque *D-SW*, nous dénotons aussi la même dynamique avec les métriques RMSE et HD. En effet, malgré l'utilisation d'un masque *D-SW*, où un seul bit est chiffré par coordonnée, nous observons que les valeurs de RMSE, illustrées par la courbe orange en figure 3.6.a, sont fortement identiques à celles pour le masque *D-LSB*. La même dynamique est retrouvée avec la distance de Hausdorf illustrée en figure 3.6.b par la courbe orange, où la valeur maximale est atteinte en $p = 22$ avec 2.5 environ.

Ainsi, nous pouvons déduire que la position du premier bit significatif chiffré dans la représentation de valeurs flottantes impacte principalement la présence des distorsions géométriques au sein des objets 3D chiffrés sélectivement. De plus, étant donné que les métriques RMSE et HD ont des résultats relativement similaires, comme illustrés en figure 3.6, nous notons que réduire le nombre de bits chiffrés ne change pas les résultats des métriques. Ainsi, dans le cas d'un masque *D-SW* avec un bit chiffré par coordonnée, notre méthode ne chiffre que 3.125% de la géométrie de l'objet 3D pour en protéger le contenu.

3.3.3 Analyse de la sécurité

Tout d'abord, nous étudions la sensibilité de la clé secrète de notre méthode. Pour cela, nous analysons la sensibilité de notre paramètre de contrôle, à savoir le niveau de dégradation. Nous discutons de la fragilité liée aux méthodes chiffrant sélectivement, face aux attaques par force brute et aux attaques par traitement de maillage.

Pour tester la sensibilité de la clé secrète $\mathcal{K}$, nous chiffrons un objet 3D $\mathcal{M}$ en utilisant la clé secrète $\mathcal{K}$. Nous générons alors un ensemble de clés $\mathbb{K}$ tel que :

$$d_{\mathrm{H}}(\mathcal{K}, k) = 1, k \in \mathbb{K}, \tag{3.6}$$

où d_{H} est la distance de Hamming.

À partir de cet ensemble de clés ne différant que d'un seul bit par rapport à la clé secrète $\mathcal{K}$, nous appliquons des déchiffrements de l'objet 3D chiffré $\mathcal{M}'$. Nous comparons alors les objets 3D déchiffrés avec l'ensemble de clés $\mathbb{K}$ à l'objet 3D secret $\mathcal{M}$.

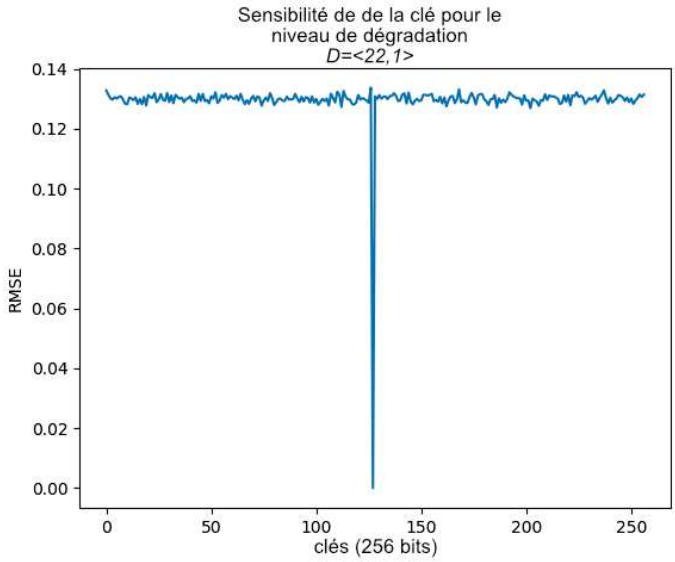


FIGURE 3.7 – Résultats du RMSE pour les objets 3D déchiffrés avec 255 clés secrètes ne différant que d'un bit de clé secrète $\mathcal{K}$ par rapport à la clé secrète $\mathcal{K}$ avec $\mathcal{D} =< 22, 1 >$.

En figure 3.7, nous observons que seule la clé originale permet de récupérer l'objet 3D secret sans perte (RMSE = 0). Tandis que les déchiffrements utilisant les autres clés ont généré des objets 3D ayant tous un RMSE proche de 0.13 et restent éloignés de l'original.

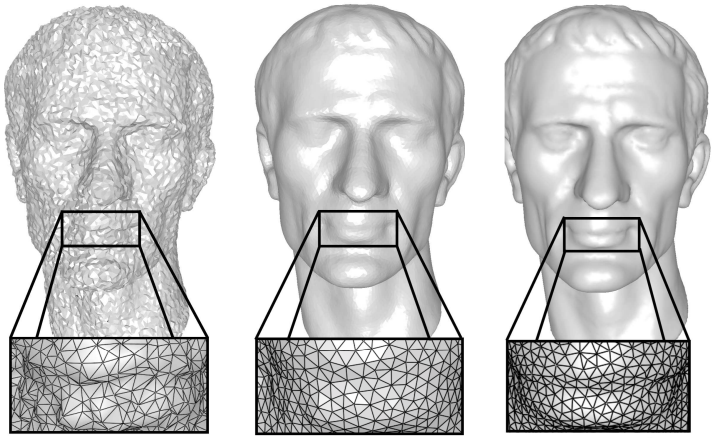
Lorsqu'un utilisateur tente de déchiffrer un objet 3D chiffré avec la bonne clé mais avec un niveau de dégradation $\mathcal{D}_{bad}$ incorrect, alors l'objet 3D n'est pas révélé. Dans le cas d'un masque *D-LSB*, lorsque $\mathcal{D}_{bad} < \mathcal{D}$ où $\mathcal{D}$ est le niveau de dégradation correct pour le déchiffrement, $\mathcal{D}_{bad}$ bits sont déchiffrés, mais pas les plus significatifs du masque. Comme expliqué précédemment, avec un masque *D-SW* le chiffrement d'un seul bit suffit

à protéger visuellement le contenu autant que le masque *D-LSB*. De plus, si $\mathcal{D}_{bad} > \mathcal{D}$, le masque *D-LSB* est déchiffré, mais les $(\mathcal{D}_{bad} - \mathcal{D})$ bits les plus significatifs sont chiffrés rendant l'objet 3D encore reconnaissable. Le même problème est observable avec le masque *D-SW*. Toujours en matière de sécurité, comme il s'agit d'une méthode de chiffrement sélectif, la quantité de bits à chiffrer est inférieure à celle d'un chiffrement complet. Cela rend la méthode plus rapide aux étapes de chiffrement et déchiffrement, mais aussi plus sensible à diverses attaques. Ainsi, au lieu de chercher à trouver la clé secrète $\mathcal{K}$, une stratégie possible peut consister à essayer de reconstruire le contenu chiffré par force brute. Naïvement, l'attaquant va chercher la valeur des bits chiffrés pour chaque coordonnée de chaque sommet de l'objet 3D chiffré. Cela correspond à trouver la combinaison correcte parmi $2^{3\times \mathrm{V}\times l}$, où V est le nombre de sommets. Il existe des approches plus intelligentes cherchant à reconstruire le contenu chiffré en fonction du contenu préservé au voisinage. Comme présenté par Said [112], il est possible d'attaquer le chiffrement partiel. L'auteur a ainsi défini un système de cryptanalyse pour les images partiellement chiffrés en utilisant les données préservées du chiffrement. Si nous prenons en considération son approche pour notre méthode et utilisons les informations publiques comme le niveau de dégradation ou la connectivité de l'objet 3D, il est possible pour un attaquant de construire une heuristique permettant d'assister le choix de la valeur des bits pour chaque coordonnée de tous les sommets. Un attaquant peut au lieu de s'attaquer à tous les bits chiffrés, s'intéresser itérativement au bit chiffré le plus significatif pour réduire le paramètre p du niveau de dégradation $\mathcal{D}$. Ainsi, la recherche de la bonne combinaison se réduit à $2^{3\times \mathrm{V}}$. En répétant l fois la recherche de la bonne combinaison pour le bit chiffré le plus significatif de chaque coordonnée, un attaquant peut déchiffrer progressivement un objet 3D. À chaque fois, l'attaquant doit conserver, pour l'itération suivante, l'objet 3D le plus "cohérent" visuellement. En effet, comme l'adversaire ne possède aucune donnée sur l'objet 3D secret, il ne peut pas évaluer la distance entre l'objet 3D qu'il a déchiffré et l'objet 3D secret. Il est possible de réaliser un autre type d'attaque en essayant de traiter directement l'objet 3D chiffré. Comme notre méthode génère des objets 3D chiffrés sélectivement pour permettre de reconnaître le contenu 3D (chiffrement transparent), il est normal que notre approche soit sensible à des attaques telles que le lissage ou la reconstruction. Ainsi, pour certaines valeurs du niveau de dégradation $\mathcal{D}$, un lissage laplacien avec un facteur de déformation $\lambda = 0.3$ et 10 itérations permettent de récupérer un objet 3D suffisamment proche de l'objet 3D original.

Comme illustré en figure 3.8, le lissage laplacien transforme l'objet 3D chiffré en un objet 3D sensiblement similaire visuellement à l'original. Cependant, nous remarquons des pertes d'information dans des zones très distinctes de l'objet 3D, particulièrement dans les basses fréquences. Avec la distance de Hausdorff, l'objet 3D lissé a une valeur autour de 1.550×10^{-2}, tandis que l'objet 3D chiffré est aux alentours de 1.689×10^{-2}. Même si l'objet 3D se rapproche visuellement de l'original, il en reste néanmoins très différent statistiquement. De plus, ce genre d'attaques devient très rapidement inefficace à partir d'un niveau de dégradation plus important.

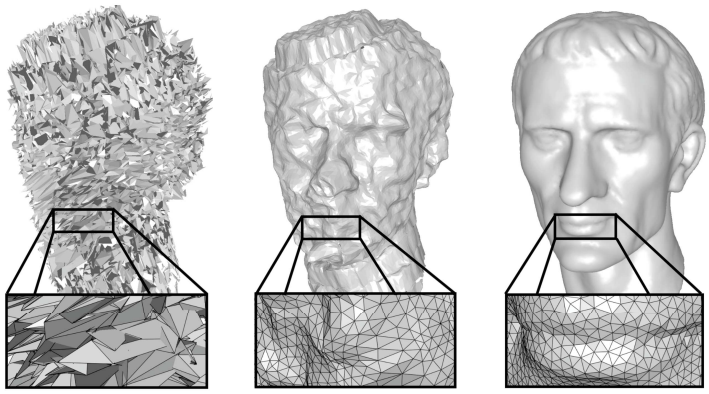
La figure 3.9 représente un exemple d'une attaque par lissage où le résultat obtenu est, pour le SVH, très différent de l'objet 3D original. La valeur de la distance de Hausdorff pour l'objet 3D lissé est de 8.416×10^{-2}, tandis que pour l'objet 3D chiffré elle est de 1.689×10^{-2}. Comme attendu, l'objet 3D lissé s'éloigne de l'objet 3D original comme indiqué par l'augmentation de la distance de Hausdorff.

Comme illustré en figure 3.10, le choix du niveau de dégradation $\mathcal{D}$ nous permet de sélectionner le niveau de confidentialité désiré, allant de la confidentialité visuelle au chiffrement transparent en passant par le chiffrement suffisant.



(a) (b) (c)

FIGURE 3.8 – Attaque par lissage laplacien : a) Objet 3D chiffré avec $\mathcal{D} =< 18, 1 >$, b) Résultat après lissage ($\lambda = 0.3$ et 100 itérations), c) Objet 3D original.



(a) (b) (c)

FIGURE 3.9 – Attaque par lissage laplacien : a) Objet 3D chiffré avec $\mathcal{D} =< 21, 1 >$, b) Résultat après lissage ($\lambda = 0.3$ et 100 itérations), c) Objet 3D original.

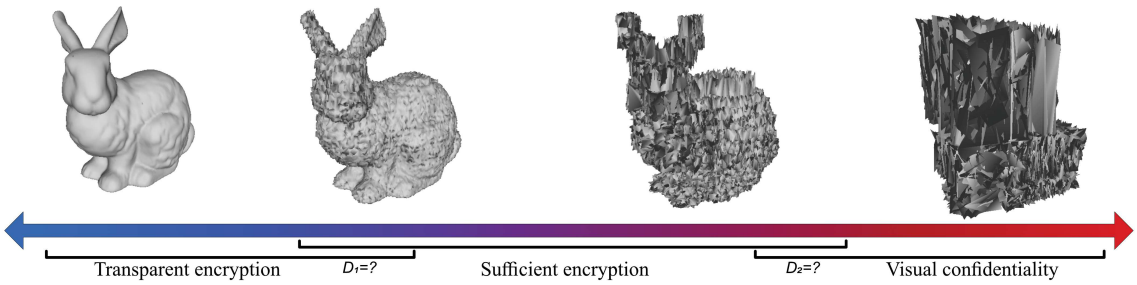


FIGURE 3.10 – L'objet 3D *Bunny* selon différents niveaux de dégradation en fonction de la confidentialité voulue.

3.3.4 Comparaison avec des méthodes de l'état de l'art

Le tableau 3.2 résume les points de comparaison entre notre méthode proposée et celles de l'état de l'art en matière de chiffrement 3D.

TABLEAU 3.2 – Comparaison de notre méthode de chiffrement sélectif 3D avec les travaux précédents.

Méthodes / Propriétés	Cho *et al.* [36]	Gschwandtner et Uhl [55]	*Coordinate Shuffling* [50]	*Dithering* [50]	*Fragment Scaling* [50]	**Méthode proposée**
Contenu protégé	Connectivité	Connectivité & Géométrie	Géométrie	Géométrie	Géométrie	**Géométrie**
Boîte englobante	Préservée	Cachée	Préservée	Préservée	Préservée	**Cachée**
Complexité	Modérée	Élevée	Faible	Faible	Modérée	**Faible**
Chiffrement transparent	✔	✔	✘	✔	✔	✔
Chiffrement suffisant	✘	✘	✘	✔	✔	✔
Confidentialité visuelle	✘	✘	✔	✔	✘	✔

Notre méthode propose de chiffrer un intervalle de bits dans la représentation binaire des coordonnées de chaque sommet afin d'en protéger le contenu. Cette approche permet aux utilisateurs de choisir le niveau de confidentialité en fonction des cas d'utilisation. Nous ne nous sommes pas intéressés à la protection de la connectivité contrairement à certains auteurs [36, 55], car nous considérons que la connectivité seule d'un objet 3D ne permet pas de le reconnaître visuellement, seul le genre topologique de l'objet 3D peut être déduit. C'est pourquoi nous visons à protéger uniquement la géométrie plutôt que la connectivité, car comme expliqué en section 1.2 du chapitre 1, la connectivité d'un objet 3D peut être reconstruite si le nuage de points approxime uniformément la surface. Ainsi, la méthode de Cho *et al.* [36] chiffrant les indices des triangles peut voir son contenu chiffré reconstruit. Contrairement à Éluard *et al.* [50] dont les méthodes préservent la boîte englobante de l'objet 3D, nous cachons cette dernière à l'intérieur des bornes de l'erreur relative liée à la représentation des valeurs flottantes. Ainsi, la modification de la boîte englobante n'est pas perceptible pour un faible niveau de dégradation. Cependant, si la confidentialité visuelle est requise pour protéger le contenu 3D, alors la boîte englobante de l'objet 3D augmente, évitant de révéler les dimensions de l'objet 3D secret.

3.4 Conclusion

Dans ce chapitre, nous nous sommes intéressés à développer une méthode de protection des objets 3D, plus particulièrement de chiffrement sélectif 3D. Nous avons ciblé en priorité la géométrie des objets 3D, car il s'agit de l'information principale les représentant. De ce fait notre approche pourrait être étendue à des nuages de points 3D tels que les données acquises par LIDAR par exemple. Ainsi, nous avons proposé une nouvelle méthode de chiffrement sélectif préservant le format et protégeant sélectivement un objet 3D en chiffrant une sélection de la représentation binaire des valeurs flottantes des coordonnées des sommets. À partir d'une clé secrète $\mathcal{K}$ et un niveau de dégradation $\mathcal{D}$ choisis par l'utilisateur, la méthode contrôle les distorsions géométriques générées par le chiffrement de l'objet 3D. Notons que les objets 3D chiffrés sélectivement peuvent être affichés dans des scènes 3D, car leur structure interne est préservée du fait que nous respectons le format original.

Dans des travaux futurs, nous souhaitons étendre l'application de notre méthode de chiffrement par l'ajout de fonctionnalités de déchiffrement de qualité variable en fonction de la clé secrète insérée. Par exemple, pour un objet 3D protégé avec un chiffrement fort nous souhaitons disposer de plusieurs clés permettant d'accéder à l'objet 3D selon différents niveaux de qualité. Ainsi, une clé secrète permettrait d'accéder à l'objet 3D en haute qualité, une autre autoriserait à accéder au contenu selon un niveau de chiffrement suffisant ou bien transparent.

Une autre possibilité envisageable est de chiffrer différentes parties de l'objet 3D selon différents niveaux de dégradation. Ainsi, les utilisateurs laisseront certaines parties de l'objet 3D avec un chiffrement transparent, tandis que d'autres seront chiffrées avec un niveau de confidentialité plus élevé. Les utilisateurs pourront également utiliser des approches de segmentation pour délimiter les zones à chiffrer de manière automatique.

Dans le chapitre 6, nous montrons comment évaluer la confidentialité des objets 3D chiffrés avec notre méthode.

Ces travaux ont fait l'objet de deux publications dans des conférences, la première méthode dans la conférence nationale CORESA 2017 [12] et la seconde dans un *workshop* de la conférence internationale IEEE ICME 2018 [13].

Chapitre 4

Partage d'objet 3D secret

Sommaire

4.1 Introduction

Dans ce chapitre, nous proposons une nouvelle méthode de partage d'image secrète se basant sur les travaux de Blakley [20] ainsi qu'une nouvelle méthode de partage sélective d'objet 3D secret fournissant des *shares* respectant le format initial des objets 3D.

Pour notre première contribution, nous nous sommes inspirés des travaux effectués par Blakley [20] sur le partage de secret, afin de proposer une nouvelle méthode de partage d'image secrète (2, n) où les équations des hyperplans sont représentées sur un seul octet pour chaque pixel d'une image en niveau de gris. De cette manière, la méthode proposée permet de partager individuellement les pixels et évite que la taille des *shares* augmente. Concernant la nouvelle méthode de partage d'objet 3D que nous proposons, celle-ci est une méthode de partage d'objet 3D secret préservant le format et fournissant donc en sortie des objets 3D. Les *shares* sont des objets 3D dits partagés représentant l'objet 3D secret en basse qualité. Pour réaliser cela, nous avons repris nos travaux sur le chiffrement sélectif présenté au chapitre 3 afin de proposer une nouvelle fonctionnalité dans le domaine du partage d'objet 3D secret permettant de contrôler la confidentialité visuelle des objets 3D partagés selon un niveau de dégradation variant en fonction des besoins des utilisateurs.

Nous présentons notre contribution de partage d'image secrète dans la section 4.2. En section 4.3, nous présentons notre contribution de partage sélectif d'objet 3D préservant le format. Enfin, la section 4.4 conclut ce chapitre et présente quelques pistes futures de recherche.

4.2 Partage d'image secrète selon la méthode de Blakley

Dans cette section, nous présentons une méthode de partage d'image secrète utilisant la méthode de partage de secret de Blakley [20]. L'idée principale est de partager individuellement les pixels de l'image secrète par des équations d'hyperplan de dimension 2 et d'encoder ces équations sur les bits d'un pixel afin que les *shares* générées aient la même taille que l'image secrète. La principale difficulté réside dans la représentation des équations de droite sur un seul octet. Ainsi, nous proposons une représentation d'équations de droite tenant sur un octet et permettant de répartir uniformément des droites distinctes. La notion d'uniformité désigne le fait que pour tout niveau de gris, il existe au moins n droites permettant de représenter un niveau de gris. En section 4.2.1, la méthode de partage d'image secrète est décrite avec les spécifications nécessaires. Tandis que dans la section 4.2.2, les résultats expérimentaux sont évalués d'un point de vue statistique. Enfin en section 4.2.3, nous concluons sur nos travaux portant sur le partage d'image secrète et ses futures améliorations.

4.2.1 Méthode de partage d'image secrète

Dans cette section, nous présentons notre méthode partage d'image secrète basée sur la méthode de Blakley [20]. Cette méthode permet de partager individuellement les pixels d'une image et permet de reconstruire l'image secrète lorsque au moins 2 utilisateurs combinent leurs *shares*, ici des images 2D. Avec notre approche, le nombre maximum d'utilisateurs participant est compris entre 2 et 7. Le processus de partage de notre méthode illustré en figure 4.1, nécessite en entrée une image secrète I et un paramètre n correspondant au nombre de *shares* à générer en sortie. La méthode est divisée en trois étapes, à savoir la phase de diffusion, le partage des pixels et la permutation par flot.

Contrairement à la méthode de Thien et Lin [129], la méthode proposée ne nécessite pas de clé secrète supplémentaire.

La phase de diffusion permet, en tant que prétraitement, de transformer l'image afin d'augmenter la robustesse de notre méthode aux attaques basées sur des analyses des histogrammes des *shares*. Durant la deuxième étape, les pixels sont partagés individuellement par la méthode proposée par Blakley [20] avec les seuils $(k = 2, n)$ pour générer des équations de droites. En sortie de l'étape de partage, un codage binaire spécifique est utilisé pour représenter les équations de droites générées sur un seul octet selon un dictionnaire de codes. Enfin, la dernière étape, qui concerne la permutation par flot, intervertit itérativement les valeurs représentant les équations afin d'uniformiser les histogrammes des *shares* en fonction de la permutation précédente et des informations de l'image secrète.

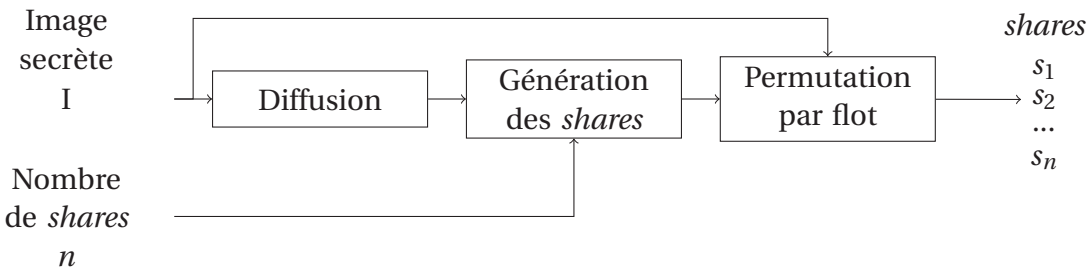


FIGURE 4.1 – Processus de partage d'une image secrète selon la méthode proposée.

Phase de diffusion

Cette étape de prétraitement consiste à diffuser l'information au sein de l'image secrète afin d'augmenter la robustesse de notre méthode face à des attaques basées sur l'analyse des histogrammes des *shares* générées lorsque l'image secrète contient des zones uniformes. Nous proposons que la phase de diffusion prenne la forme d'une transformation réversible du gradient :

$$p_{\mathrm{I_D}}(i) = (p_{\mathrm{I}}(i) + i) \bmod 256, \tag{4.1}$$

où $p_{\mathrm{I}}(i)$ est la valeur du niveau de gris du $i^{\text{ème}}$ pixel de l'image secrète et $p_{\mathrm{I_D}}(i)$ la valeur du pixel après diffusion.

Génération des images *shares*

L'étape de partage de notre méthode consiste à partager le niveau de gris de chaque pixel de l'image modifiée $\mathrm{I_D}$ en n équations de droites. Avant cela, pour chaque niveau de gris possible dans une image, ici des valeurs représentées sur un octet (8 bits) entre 0 et 255, nous assignons des coordonnées (x, y) à chacune de ces valeurs de niveau de gris où $x, y \in [\![-8\,;\,8[\![$ telles que :

$$\begin{cases} x = (p_{\mathrm{I_D}}(i) \bmod 16) - 8, \forall i \in [\![0\,;\,256[\![, \\ y = \lfloor \frac{p_{\mathrm{I_D}}(i)}{16} \rfloor - 8. \end{cases} \tag{4.2}$$

Ainsi, dans le plan discret $x, y \in [\![-8\,;\,8[\![$ chaque coordonnée représente une valeur possible de niveau de gris de l'image secrète. Ces coordonnées sont utilisées comme des

points secrets de la méthode de Blakley [20], c'est-à-dire comme des points d'intersection des droites distribuées aux utilisateurs.

Chaque utilisateur reçoit alors les coefficients des équations représentant ces droites. Nous proposons de coder ces équations sur un seul octet de manière à partager indépendamment chaque pixel de l'image secrète et de préserver le format et la taille de l'image secrète au sein des *shares*. Les équations utilisées dépendent de 3 variables, à savoir un coefficient a, un décalage l correspondant à une translation sur l'axe x appliqué à l'équation et un coefficient b. La figure 4.2 illustre la distribution des bits pour coder les coefficients d'équations de droites sur un octet. Ainsi, les coefficients a et l sont codés ensemble sur 4 bits et le coefficient b sur 4 bits.

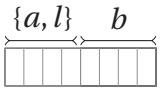


FIGURE 4.2 – Représentation du codage d'une équation de droite pour un pixel partagé sur un octet.

Pour représenter uniformément le même nombre d'équations pour toutes les coordonnées comprises dans le plan $[\![-8\ ;\ 8[\![\times[\![-8\ ;\ 8[\![$, nous utilisons un ensemble d'équations définies par :

$$y = a \times x - \mathrm{sgn}(a) \times b - a \times (b + l), \tag{4.3}$$

où la fonction $\mathrm{sgn}(a)$ est définie :

$$\forall x \in \mathbb{R},\ \mathrm{sgn}(x) = \begin{cases} -1 & \text{si } x < 0, \\ 1 & \text{sinon.} \end{cases} \tag{4.4}$$

Dans l'exemple illustré en figure 4.3, notre méthode partage le niveau de gris du pixel $p_{i_D}(i) = 189$ et à partir de l'équation (4.2), convertit le niveau de gris en coordonnées $(x = 5, y = 3)$. Les droites en bleu $(\mathrm{L}_0, \mathrm{L}_1, \mathrm{L}_2, \mathrm{L}_3)$ représentent des exemples de droites passant par le point (5,3) dont les équations de ces droites codent la valeur de niveau de gris 189. Les droites en rouge représentent les deux axes nommés Axis_1 and Axis_0 et ont respectivement pour les équations : $y = x$ et $y = -x$. Au lieu d'utiliser les axes x et y, nous construisons un ensemble d'équations de droites en utilisant comme repère les axes Axis_1 and Axis_0 pour simplifier la représentation des équations dans notre méthode. Chaque équation doit avoir un point d'intersection avec l'un des axes de ce nouveau repère.

Dans la figure 4.3, la droite notée L_1 et l'axe Axis_0 s'intersectent au point (1,-1). Ce point d'intersection noté $\mathrm{P} = (x_p, y_p)$ possède la propriété suivante : pour toutes les équations définissant des droites intersectant l'axe Axis_1 alors $\exists x \in [\![-8\ ;\ 8[\![$ tel que $\mathrm{P} = (x, x)$; tandis que pour l'axe Axis_0, $\exists x \in [\![-8\ ;\ 8[\![$ tel que $\mathrm{P} = (x, -x)$. Ces équations peuvent donc être écrites de la manière suivante :

$$\begin{cases} y = a \times x + \mathrm{B}, \\ y_p = a \times x_p + \mathrm{B}, \end{cases} \Leftrightarrow \begin{cases} y = a \times x + (y_p - a \times x_p), \\ \mathrm{B} = y_p - a \times x_p. \end{cases} \tag{4.5}$$

Or, comme expliqué précédemment le point P est le point d'intersection entre les axes Axis_0 ou Axis_1 et la droite d'équation (4.3). C'est pourquoi, nous pouvons déduire la valeur de y_p du point P tel que $y_p = -x_p$ ou $y_p = x_p$ en fonction de l'axe Axis_0 ou Axis_1, respectivement. Si le point d'intersection se trouve sur l'axe Axis_0, cela signifie que la droite possède un coefficient a positif et négatif lorsque le point d'intersection se trouve sur l'axe

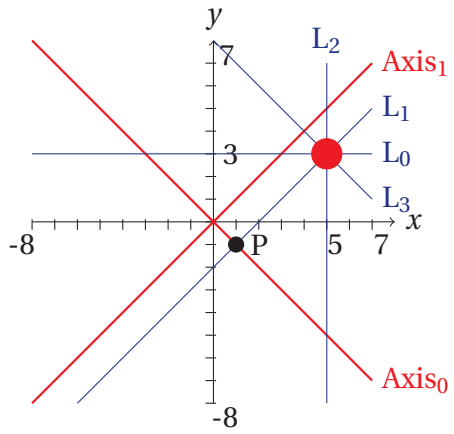


FIGURE 4.3 – Exemple d'équations de droites (L_0, L_1, L_2, L_3) utilisées pour partager le point (5,3) et P le point d'intersection de L_1 avec l'axe $Axis_0$.

$Axis_1$. La relation entre le point P et le signe du coefficient a de l'équation de la droite est égale à :

$$y_p = -sgn(a) \times x_p. \tag{4.6}$$

De cette manière, pour représenter une équation nous avons besoin de sauvegarder la coordonnée x_p, que nous notons dorénavant $b = x_p$ avec le coefficient a correspondant à la pente de la droite :

$$y = a \times x - sgn(a) \times b - a \times b. \tag{4.7}$$

A l'exception d'un cas spécial (noté "0∗") codant une équation de droite verticale ($\forall y \in [\![-8\ ;\ 8[\![, x = b$), toutes les équations de droites utilisent comme coefficient a des valeurs issues de l'ensemble :

$$\mathbb{C}_f = \mathbb{Z} \cup \{\forall i \in \mathbb{Z}^*, \frac{1}{i}\}. \tag{4.8}$$

Cependant, l'ensemble des droites générées à partir des coefficients $\mathbb{C}_f$ n'est pas suffisant pour représenter uniformément toutes les coordonnées possibles dans $[\![-8\ ;\ 8[\![\times [\![-8\ ;\ 8[\![$.

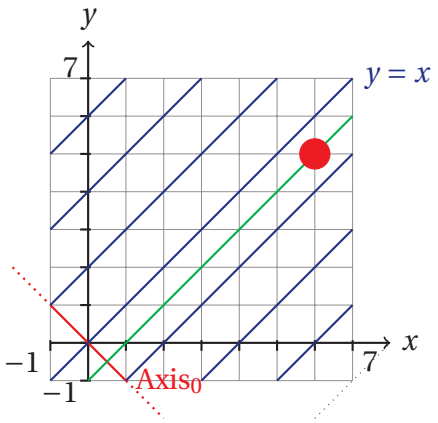


FIGURE 4.4 – Exemple d'équations de droites générées par notre méthode : les droites en bleu ont comme coefficient $a = 1$. La droite en vert correspond à la translation de la droite en bleu d'équation $y = x$ selon l'axe x par un décalage $l = 1$.

La figure 4.4 illustre un exemple avec des droites d'équation $a = 1$ et $\forall b \in [\![-8\ ;\ 8[\![$. Dans ce cas, le point en rouge de coordonnées (6, 5) par exemple n'est pas accessible par les droites en bleu correspondant aux équations générées. Pour permettre d'atteindre ces

valeurs, notre méthode translate les droites selon l'axe x par un certain décalage noté l. Par exemple, dans la figure 4.4, la droite en vert représente la droite en bleu $y = x$ translaté de 1 selon l'axe x et donnant l'équation :

$$y = x + b - (b - 1) = x - 1. \tag{4.9}$$

Pour un coefficient $a = 2$, nous avons besoin d'utiliser un décalage de 1 et de 2 pour que les équations représentent l'ensemble des valeurs de l'espace discret $[\![-8\,;\,8[\![\times[\![-8\,;\,8[\![$. Le nombre de décalages nécessaires à un ensemble de droites ayant le même coefficient a afin d'atteindre toutes les coordonnées possibles de manière uniforme est déterminé par $\sigma(a)$:

$$\forall u \in \mathbb{C}_f,\ \sigma(u) = \begin{cases} u & \text{Si } u \in \mathbb{Z} \\ \frac{1}{u} & \text{sinon} \end{cases}. \tag{4.10}$$

Le résultat $\sigma(a)$ donne ainsi le nombre de décalages nécessaires pour représenter toutes les valeurs dans l'espace discret $[\![-8\,;\,8[\![\times[\![-8\,;\,8[\![$ en utilisant la valeur de coefficient a. Toutes les équations peuvent être représentées par l'ensemble d'équations θ :

$$\theta = \{\forall x, \forall y \in [\![-8\,;\,8[\![, \forall a \in \mathbb{C}_f, \exists b \in [\![-8\,;\,8[\![, \tag{4.11}$$

$$\exists l \in [\![0\,;\,\sigma(a)]\!] \mid y = a \times x - \mathrm{sgn}(a) \times b - a \times (b + l)\}. \tag{4.12}$$

Comme expliqué précédemment, ces équations sont représentées sur un octet avec 4 bits alloués pour le coefficient b représentant les 16 valeurs de l'intervalle $[\![-8\,;\,8[\![$. Les 4 bits restants servent d'indice dans le dictionnaire de codes (*codebook*) indiquant les valeurs pour le coefficient a et le décalage l comme résumé dans le tableau 4.1.

a	l
0	0
1	0
1	1
1/2	0
1/2	1
1/2	2
2	0
2	1

a	l
2	2
0*	0
-1	0
-1	1
-1/2	0
-1/2	1
-1/2	2
-2	0

TABLEAU 4.1 – Dictionnaire des coefficients a et décalage l offrant un nombre maximum de 256 équations de droite distinctes (0* signifiant une droite verticale).

Pour éviter l'effet d'expansion de la taille des *shares*, nous utilisons seulement les 256 équations définies par l'équation (4.3) et le dictionnaire de codes du tableau 4.1. Cette restriction réduit le nombre de droites utilisables et contraint notre méthode à un nombre maximum de 7 *shares* à générer.

Permutation par flot

Après l'étape de génération des *shares*, une permutation par flot est appliquée sur les équations générées et codées pour chaque pixel de l'image secrète. Cette étape supplémentaire assure une meilleure robustesse statistique face à l'analyse d'histogramme comme pour l'étape de diffusion présentée précédemment. Chaque fois qu'un pixel est

traité, une table de permutation notée T_i est calculée pour changer la valeur des équations codées sur un octet notée $E_j(i)$ où i est l'indice du $i^{\text{ème}}$ pixel et $j \in [\![0\,;\,n[\![$ l'indice de la *share* telle que :

$$T_i = \text{Shuffle}(T_{i-1}, p_I(i-1)), \tag{4.13}$$

où la fonction Shuffle() est l'opération de permutation de la table T_{i-1} par le niveau de gris du $(i-1)^{\text{ème}}$ pixel de l'image secrète I.

Chaque fois que la méthode partage un niveau de gris d'un pixel de l'image secrète, celle-ci modifie la table de permutation précédente à l'aide d'une fonction pseudo-aléatoire utilisant la valeur du niveau de gris du pixel précédent $p_I(i-1)$ de l'image secrète I comme clé secrète. Cette approche modifie dynamiquement la table de permutation pour assurer une utilisation de toutes les valeurs de niveau de gris possibles et elle se trouve être similaire au mode de chiffrement par bloc nommé *Output FeedBack* (OFB) présenté en section 1.4.2 du chapitre 1. Pour le premier pixel partagé, la table de permutation T_0 est initialisée selon un vecteur d'initialisation.

A chaque pixel de chaque *share* est assignée une équation de droite $E_j(i)$ représentée selon le codage binaire proposé en figure 4.2. Les valeurs utilisées pour représenter ces équations sont utilisées comme indices dans la table de permutation T_i et la valeur associée dans cette table devient la valeur du niveau de gris du pixel dans la *share* s_j telle que :

$$s_j(i) = T_i[E_j(i)]. \tag{4.14}$$

Reconstruction de l'image secrète

La reconstruction de l'image secrète consiste, à partir d'au moins 2 *shares*, à calculer les points d'intersection des équations provenant de la même position de pixel de chaque *share*. Ce processus est assez similaire à celui de partage et fonctionne suivant ces 9 étapes :

Etape 1 : Récupération des pixels de chaque *share* à la même position ;

Etape 2 : Permutation des valeurs de pixels selon la table de permutation T_i ;

Etape 3 : Conversion de la valeur de chaque pixel en équations de droite ;

Etape 4 : Identification du point d'intersection (x, y) en résolvant le système linéaire formé par les équations de l'étape 2 ;

Etape 5 : Récupération de la valeur de $p_{i_D}(i)$ selon les coordonnées (x, y) du point d'intersection ;

Etape 6 : Utilisation de la transformation inverse de la phase de diffusion sur la valeur obtenu à l'étape 5 ;

Etape 7 : Conservation du résultat de l'étape 6 comme pixel de l'image secrète reconstruite ;

Etape 8 : Mise-à-jour de la table de permutation T_{i+1} selon la valeur de pixel $p(i)$ comme clé secrète ;

Etape 9 : Répétition des étapes 1-8 pour tous les pixels restant.

4.2.2 Résultats expérimentaux

Dans cette section, nous présentons des résultats expérimentaux de notre méthode de partage d'image secrète. La figure 4.5 illustre l'application de notre méthode en utilisant

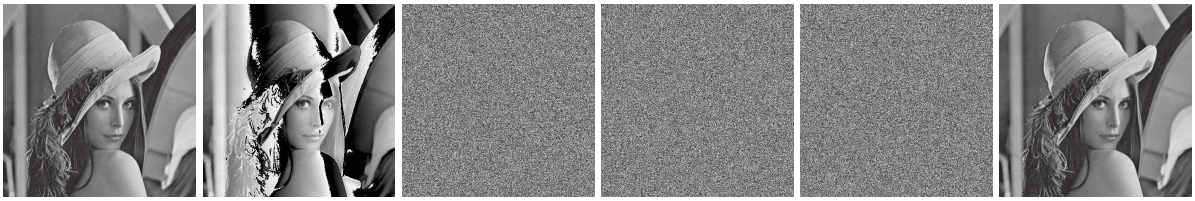
les paramètres ($k = 2, n = 3$) sur l'image *Lena*. L'utilisation des paramètres ($k = 2, n = 3$) signifie que la méthode génère trois *shares* et l'image secrète peut être reconstruite en regroupant au moins deux *shares*. La figure 4.5.a représente l'image secrète originale en niveau de gris. La figure 4.5.b correspond à l'image intermédiaire après l'étape de diffusion de la figure 4.5.a. Les figures 4.5.c-e illustrent les trois *shares* générées dans lesquelles aucune information de l'image secrète originale n'est révélée. Tandis que la figure 4.5.f représente l'image secrète reconstruite. Ces deux images sont parfaitement identiques.

Analyse statistique

Avec à notre méthode, l'image secrète est reconstruite sans perte d'information comme indiqué dans le tableau 4.2 avec une valeur de PSNR tendant vers l'infini. Les valeurs de corrélation horizontale et verticale sont très proches de zéro pour les *shares*. De plus, l'entropie au sein des *shares* atteint sa valeur maximale, à savoir 8 bits par pixel.

	PSNR (en dB)	Entropie (en bpp)	Corr. H.	Corr. V.
Share	9.04	7,99	0.00396	−0.00968
Image secrète reconstruite	∞	7,40	0.96616	0.98579

TABLEAU 4.2 – Analyse statistique d'une *share* et de l'image secrète reconstruite : PSNR (en dB), entropie (en bits par pixel), corrélations horizontale et verticale.



(a) (b) (c) (d) (e) (f)

FIGURE 4.5 – Résultats du processus de partage et de reconstruction par notre méthode : a) L'image secrète originale (512 × 512 pixels), b) L'image après diffusion à partir de (a), c-e) Les 3 *shares*, f) L'image secrète reconstruite sans perte à partir de n'importe quel groupe de 2 *shares*.

Ces résultats expérimentaux illustrent trois propriétés, à savoir les *shares* générées par notre méthode apparaissent naturellement comme des images de bruit aléatoire. En effet, les équations codées ne révèlent rien de l'histogramme de l'image secrète comme illustré en figure 4.6 grâce à l'étape de diffusion et la permutation par flot maintenant l'entropie à sa valeur maximale. Enfin, chaque pixel est partagé indépendemment, ce qui augmente la sécurité de notre méthode. Dans le but de valider nos résultats, nous appliquons notre méthode aux 10.000 images de la base de données BOWS-2[1] en utilisant les paramètres ($k = 2, n = 7$) et les résultats sont présentés dans le tableau 4.3. Le tableau 4.3 présente que

	PSNR (en dB)	Entropie (en bpp)	Corr. H.	Corr. V.
BOWS-2	8.143 (±.897)	7.997 (±.094)	0.00018 (±.0141)	−0.00026 (±.0141)

TABLEAU 4.3 – Moyennes et écart-types du PSNR (en dB), de l'entropie (en bits par pixel), et des corrélations horizontale and verticale pour les *shares* générées à partir des images issues de la base de données BOWS-2.

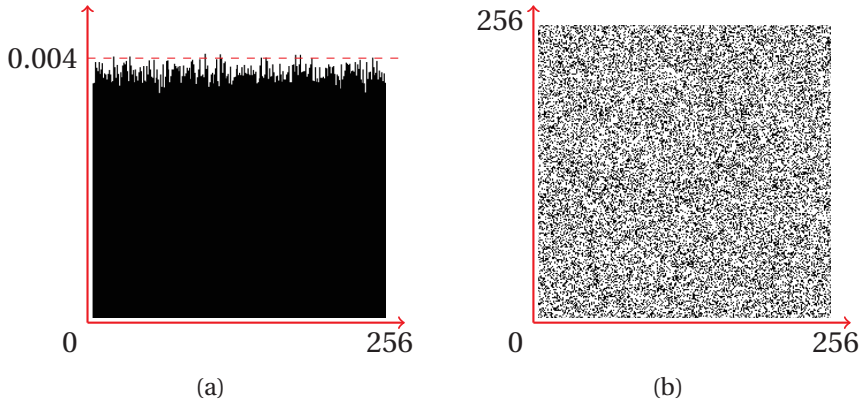


FIGURE 4.6 – Analyse statistique de la *share* illustrée figure 4.5.c : a) Densité de probabilité des valeurs des niveaux de gris, b) Corrélation horizontale.

les *shares* générées ont une valeur de PSNR inférieure à 10 dB, les corrélations horizontale et verticale sont proche de 0 et l'entropie est très proche de 8 bits par pixel.

Analyse de la sécurité

Un adversaire ne peut pas aisément reconstruire le contenu de l'image secrète à partir d'une seule *share*. La probabilité de reconstruire le secret à partir d'une *share* générée par notre méthode est de $(\frac{1}{7})^{2^{18}}$ du fait du nombre maximum d'équations possibles. Comme il s'agit d'une très faible probabilité, les attaquants pourraient essayer d'obtenir l'image originale à partir d'une *share* par force brute. Cependant, du fait des étapes de diffusion et de permutation par flot, les attaquants ont besoin de trouver la table de permutation pour chaque pixel pour reconstruire l'image secrète. Cela signifie qu'ils ont besoin de trouver le bon arrangement de 256 valeurs parmi 256 à chaque étape, ceci augmentant alors la probabilité de récupérer l'image secrète à $\frac{1}{(A_{256}^{256}\times 7)^{2^{18}}}$. Ainsi, l'approche consistant à construire une fausse *share* devient aussi difficile qu'une recherche par force brute de l'image secrète. Dans ce cas, la probabilité de deviner l'image secrète devient égale à $(\frac{1}{256})^{2^{18}}$ comme pour la méthode de partage de Shamir utilisant un corps Galois GF(2^8) [116]. En comparaison, la méthode de Thien et Lin [128], sans l'étape de permutation, a une probabilité de $(\frac{1}{251})^{2^{17}}$ à cause de la taille réduite des *shares* générées [128].

Comparaison avec l'état de l'art

Le tableau 4.4 compare notre méthode avec les méthodes de partage d'image secrète de l'état de l'art [32, 91, 116, 128, 129, 133, 135, 149]. Les *shares* générées par notre méthode préservent la taille de l'image secrète et offrent une reconstruction sans perte de l'image secrète. De plus, notre méthode propose une plus grande robustesse face aux attaques par brute force. Pour rappel, la propriété "*shares* significatives" définit les méthodes où les *shares* générés sont visuellement similaires à des images naturelles et non à des images de bruit.

1. BOWS2 Website : http ://bows2.ec-lille.fr/

Partage d'image	Taille des *shares*	Reconstruction sans perte	*Shares* significatives	Probabilité d'attaque
Shamir [116]	1	×	×	$\frac{1}{251^{2^{18}}}$
Shamir avec GF(256) [116]	1	✓	×	$\frac{1}{256^{(2^{18}}}$
[128, 129, 149]	1/2	×/✓	×/✓	$\frac{1}{251^{2^{17}}}$
Lin et Tsai [91]	4	×	✓	$\frac{1}{251^{2^{18}}}$
Tso [135]	1	×/✓	×	$\frac{1}{51^{2^{16}}}$
Tsai et Chen [133]	2	✓	×	$\frac{1}{251^{2^{18}}}$
Chen et Fu [32]	1	✓	×	$\frac{1}{64^{2^{17}}}$
Méthode proposée	1	✓	×	$\frac{1}{256^{2^{18}}}$

TABLEAU 4.4 – Comparaison de notre méthode avec des méthodes de partage d'image secrète de l'état de l'art pour $(2, n)$.

4.2.3 Conclusion

Dans cette partie, nous avons proposé une méthode de partage d'image secrète préservant le format et permettant à au moins 2 utilisateurs parmi n de pouvoir reconstruire une image secrète. Les *shares* sont ici des images pouvant être affichées ressemblant à du bruit aléatoire. Ainsi, nous avons proposé une application de la méthode de partage de secret de Blakley [20] pour les images, où les valeurs de niveau de gris des pixels sont partagées de manière indépendante. Ces valeurs sont transformées en points 2D correspondant à des points d'intersection entre des droites distribuées aux utilisateurs. Les équations de ces droites sont alors codées sur un seul octet afin de représenter un niveau de gris pour chaque pixel des *shares*. A l'aide du codage proposé, nous sommes capables de représenter un grand ensemble de droites distinctes et uniformément réparties afin de pouvoir coder au moins 7 droites pour chaque point 2D représentant un niveau de gris. Notre méthode conserve la taille de l'image secrète au sein de ses *shares* et peut être adaptée à un partage d'images couleurs. Grâce aux différentes étapes englobant le processus de partage, nous assurons une meilleure robustesse face aux attaques et notamment celles basées sur une analyse statistique des *shares*. Il est possible d'augmenter le nombre d'utilisateurs total en augmentant le nombre de bits pour représenter une équation de droite, cependant la méthode provoquera une augmentation de la taille des *shares* et le format ne sera plus préservé. Dans le chapitre 5, nous nous concentrons sur l'utilisation du partage hiérarchique d'image secrète à des fins de protection de la vie privée sur les réseaux sociaux.

4.3 Partage d'objet 3D secret

Dans cette section, nous présentons une nouvelle approche de partage d'objet 3D secret pouvant utiliser soit la méthode de partage de secret de Blakley [20], soit celle de Shamir [116]. L'idée principale est de reprendre l'approche de chiffrement sélectif 3D proposée dans le chapitre 3, où notre méthode protège la géométrie de l'objet 3D en chiffrant sélectivement des bits de la représentation binaire des coordonnées de tous les sommets. Au lieu de chiffrer les bits de la représentation binaire des coordonnées, nous proposons de les partager selon une méthode de partage de secret et d'utiliser en tant que *shares*

les objets 3D représentant l'objet 3D original en basse qualité en fonction du niveau de confidentialité souhaité. En section 4.3.1, la méthode de partage d'objets 3D sélective est décrite avec les spécifications nécessaires. Tandis que dans la section 4.3.2, des résultats expérimentaux sont évalués d'un point de vue statistique. Enfin en section 4.3.3, nous concluons sur notre méthode de partage d'objet 3D secret sélectif et ses possibles futures améliorations.

4.3.1 Méthode

Dans cette section, notre méthode de partage sélectif d'objet 3D secret préservant le format d'entrée est présentée (*Format-Compliant Selective Secret 3D Object Sharing Scheme* ou *FCSS3DOSS*). Cette méthode propose de prendre le contrôle des distorsions géométriques induites par le partage des bits représentant les coordonnées des sommets d'un objet 3D en fonction d'un niveau de dégradation désiré. Les bits sélectionnés sont substitués dans les objets 3D partagés par des mots binaires construits durant l'étape de partage selon la méthode de partage choisie (Shamir [116] ou Blakley [20]). Pour rappel, un objet 3D peut être représenté sous la forme d'un maillage $\mathcal{M} = (\mathcal{V}, \mathcal{F})$, où $\mathcal{V}$ est l'ensemble des sommets et $\mathcal{F}$ est l'ensemble des faces. Le nombre de sommets est noté V et le nombre de faces F.

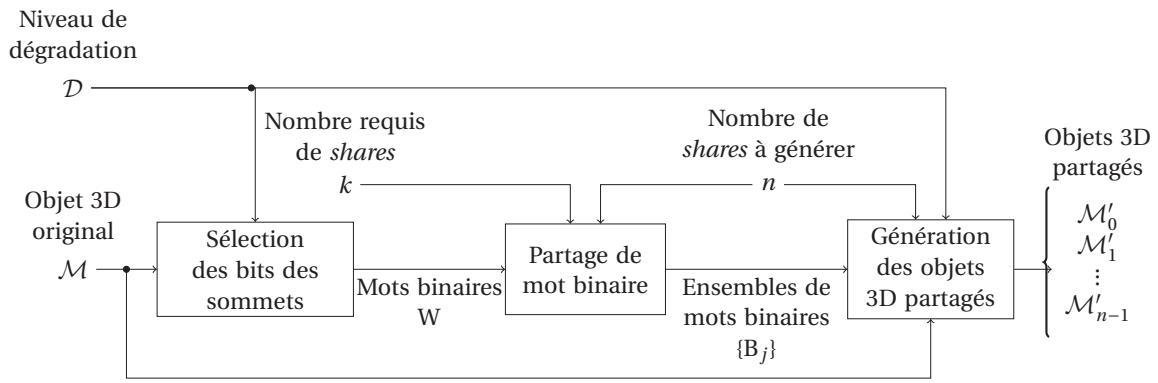


FIGURE 4.7 – Processus de partage sélectif d'objet 3D secret préservant le format.

Comme illustré en figure 4.7, en complément de l'objet 3D à partager noté $\mathcal{M}$, la méthode nécessite trois paramètres en entrée, à savoir k le nombre d'utilisateurs requis pour reconstruire l'objet 3D secret, n le nombre total d'utilisateurs et par extension, le nombre d'objets 3D partagés à générer et $\mathcal{D}$ le niveau de dégradation. La méthode est composée de trois étapes principales, à savoir la sélection des bits des sommets, le partage du mot binaire formé et la génération des objets 3D partagés.

Premièrement, nous présentons la sélection des bits des sommets à partager puis à substituer en fonction du niveau de dégradation $\mathcal{D}$. Le processus de partage utilise les bits sélectionnés comme un mot binaire à partager selon la méthode de partage de secret utilisée (Shamir [116] ou Blakley [20]). Les bits sélectionnés de la géométrie de l'objet 3D sont ensuite substitués par les mots binaires calculés à l'étape de partage dans le but de générer les n objets 3D partagés en sortie. Nous terminons alors sur la méthode de reconstruction de l'objet 3D secret grâce aux objets 3D partagés précédemment générés.

Sélection des bits des sommets

Tout comme dans l'approche de chiffrement sélectif 3D présentée au chapitre 3, notre méthode partage une sélection de bits de la représentation binaire des coordonnées des

sommets de l'objet 3D secret. Ainsi, cette approche nous assure que notre méthode préserve le format de l'objet 3D et permet à partir du niveau de dégradation désiré $\mathcal{D}$ de déterminer quel intervalle de bits doit être sélectionné pour les étapes de partage et de substitution.

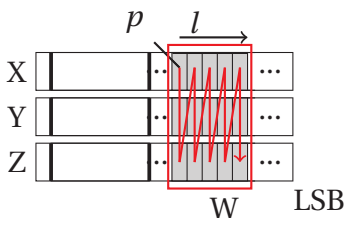


FIGURE 4.8 – Exemple de bits sélectionnés depuis la représentation binaire des coordonnées. Ces bits sont utilisés pour partager un sommet en fonction du niveau de dégradation $\mathcal{D} =< p, l >$ et la stratégie de synchronisation utilisée pour extraire les bits.

La figure 4.8 illustre comment le niveau de dégradation $\mathcal{D}$ est utilisé pour déterminer la sélection qui peut varier en fonction des deux paramètres :

— $p \in [\![0\,;\,22]\!]$, la position du premier bit sélectionné ;

— $l \in [\![1\,;\,p+1]\!]$, la longueur de l'intervalle sélectionné.

La séquence sélectionnée de bits est extraite comme un mot binaire noté W pour chaque sommet de l'objet 3D avec $|\mathrm{W}| = 3 \times l$. La sélection commence par le bit le plus significatif de la coordonnée X à la position p définie par le niveau de dégradation $\mathcal{D}$ jusqu'au dernier bit significatif de la coordonnée Z défini par l'intervalle l tout en passant par les bits de la coordonnée Y. Pour extraire un mot binaire des coordonnées pour chaque sommet, les bits sont lus en entrelaçant les bits des coordonnées comme illustré en figure 4.8. Cette stratégie permet de rassembler les bits de même poids des trois coordonnées dans les bits les plus significatifs de W. Formellement, si b_t^{X}, b_t^{Y} et b_t^{Z} sont respectivement les $t^{\text{ème}}$ bits des coordonnées X, Y et Z où $t \in [\![p-(l-1)\,;\,p]\!]$, alors :

$$\mathrm{W} = (b_p^{\mathrm{X}}, b_p^{\mathrm{Y}}, b_p^{\mathrm{Z}}, ..., b_{p-(l-1)}^{\mathrm{X}}, b_{p-(l-1)}^{\mathrm{Y}}, b_{p-(l-1)}^{\mathrm{Z}}). \tag{4.15}$$

Partage de mot binaire

Durant cette étape, à partir du mot binaire W pour chaque sommet, la méthode génère n mots binaires B_j, où $j \in [\![0\,;\,n[\![$, tel que $|\mathrm{B}_j| = |\mathrm{W}|$ avec $|.|$ le nombre de bits du mot binaire. L'objectif de ce processus est de créer les n mots binaires qui vont se substituer aux bits sélectionnés dans l'objet 3D secret dans les n objets 3D partagés. En réunissant ces mots binaires, il est alors possible de calculer le mot binaire W et de reconstruire les bits manquants au sein des coordonnées. Nous proposons l'utilisation de deux méthodes de partage, à savoir la méthode de Shamir [116] et la méthode de Blakley [20].

Avec la méthode de Shamir : avec l'approche proposée par Yang *et al.* [154] pour la reconstruction sans perte pour le partage d'image secrète, il a été démontré que la méthode de Shamir peut être utilisée avec un corps de Galois noté $\mathrm{GF}(2^m)$, où m est le nombre de bits du secret S à partager. Ainsi, notre méthode opère sur un corps de Galois $\mathrm{GF}(2^m)$ où $m = 3 \times l$. De cette manière, nous pouvons augmenter le nombre d'utilisateurs recevant une *share* ainsi que la sécurité de notre méthode en fonction du niveau de dégradation $\mathcal{D}$ désiré. Plus le paramètre l du niveau de dégradation $\mathcal{D}$ est élevé, plus la méthode sera

sécurisée. Nous notons x_j, où $j \in [\![0\,;\,n[\![$, la valeur assignée à chaque utilisateur et leur objet 3D partagé respectif :

$$\begin{cases} x_j \in \mathrm{GF}(2^m), \\ x_j \neq 0, \\ \forall u, w \in [\![0\,;\,n[\![,\ u \neq w \Leftrightarrow x_u \neq x_w. \end{cases} \tag{4.16}$$

La même valeur de x_j est utilisée pour tous les sommets de l'objet 3D et doit être transmise avec l'objet 3D partagé sur un canal privé dans le but de pouvoir reconstruire l'objet 3D secret. Un ensemble de coefficients noté $\mathrm{A} = \{a_i | a_i \in \mathrm{GF}(2^m)$ et $i \in [\![1\,;\,k[\![\}$ est généré de manière pseudo-aléatoire pour chaque sommet. La valeur de W est assigné au coefficient a_0. Enfin, la méthode calcule les n valeurs du polynôme $\mathrm{B}_j = f(x_j)$ selon l'équation (1.34) en utilisant l'ensemble A en tant que coefficients du polynôme pour le sommet v :

$$\mathrm{B}_j = f(x_j) = \sum_{i=0}^{k-1} a_i \times x_j^i. \tag{4.17}$$

Grâce à l'approche de Shamir, il est donc possible d'avoir un seuil k flexible tel que :

$$2 \leq k \leq n. \tag{4.18}$$

Avec la méthode de Blakley : contrairement à la méthode de Shamir, se déroulant en deux dimensions, la méthode Blakley [20] est basée sur une géométrie hyperplanaire. Ainsi, la dimension des hyperplans utilisés dépend de k qui est le nombre requis d'utilisateurs et de *share* pour pouvoir reconstruire le secret. De plus, la méthode considère le secret comme un point en k-dimensions pour chaque sommet de l'objet 3D. Pour réaliser ceci, le mot binaire W est divisé uniformément en blocs x_i, où $i \in [\![0\,;\,k-1[\![$, afin de former le point S de k-dimensions tel que :

$$\begin{cases} \mathrm{S} = (x_0, x_1, \ldots, x_{k-1}), \\ \mathrm{W} = \bigcup_{i=0}^{k-1} x_i, \\ |\mathrm{W}| = \sum_{i=0}^{k-1} |x_i|. \end{cases} \tag{4.19}$$

Notre méthode génère n hyperplans en k-dimensions tels que le point secret S représente l'intersection de tous ces hyperplans. Un ensemble de coefficients $\mathrm{A} = \{a_i | a_i \in \mathbb{N}^*$ et $i \in [\![0\,;\,k[\![\}$ est généré pseudo-aléatoirement pour chaque hyperplan avec les contraintes suivantes :

— La contrainte de **résolution** force les n hyperplans en k-dimensions générés pour chaque sommet à être distincts et permet la résolution du système linéaire pour les $\binom{n}{k}$ combinaisons possibles pour reconstruire l'objet 3D secret. La vérification de telles combinaisons est très lourde en temps de calcul. Ainsi, nous imposons trois conditions pour s'assurer que la contrainte soit respectée :

  — $a_{k-1} = 1$, de cette manière les n équations d'hyperplans en k-dimensions ne sont pas des combinaisons linéaires entre elles. Les équations d'hyperplan peuvent alors se noter :

$$x_{k-1} = -b + \sum_{i=0}^{k-2} a_j \times x_i; \tag{4.20}$$

— $\forall u, v \in [\![0\,;\, n[\![\ u \neq v \Rightarrow h_u \neq h_v$, les équations d'hyperplan h_u et h_v sont **uniques** (de part leurs coefficients a_j) ;

— $\forall i \in [\![0\,;\, k[\![, a_i \neq 0$.

Avec ces trois conditions, la méthode s'assure de fournir des combinaisons d'hyperplans permettant de reconstruire les informations manquantes.

— Une autre contrainte est celle de l'espace, cette dernière limite l'intervalle de valeurs disponibles pour chaque coefficient a_i choisi pseudo-aléatoirement. L'espace définit la capacité de stockage des coefficients de l'hyperplan et atteint un maximum de 69 bits par sommet si nous utilisons l'ensemble des bits de la mantisse (3×23). Le second paramètre contraignant l'espace est le niveau de dégradation désiré $\mathcal{D}$. Il est alors nécessaire de contraindre la taille en bits des coefficients a_i et b :

$$|b| + \sum_{i=0}^{k-2} |a_i| = |\mathrm{W}|, \text{ avec } |\mathrm{W}| = 3 \times l. \tag{4.21}$$

Comme les coefficients a_i et x_i sont considérés comme des entiers, il est nécessaire que le coefficient b puisse représenter la somme définie dans l'équation (1.36). Une fois que les coefficients a_i sont pseudo-aléatoirement générés selon les contraintes précédentes, la méthode résout alors l'équation de l'hyperplan pour déterminer la valeur du coefficient b et répète cette opération n fois afin de générer l'ensemble d'hyperplans pour chaque sommet. Une fois que les coefficients a_i et b sont fixés, alors la méthode concatène les coefficients pour former les mots binaires B_j.

Génération d'objets 3D partagés

Les mots binaires B_j générés précédemment par une des deux méthodes de partage de secret respectent la condition $|\mathrm{B}_j| = |\mathrm{W}|$. Cette condition est respectée par la méthode de Shamir [116] par l'utilisation d'un corps Galois $\mathrm{GF}(2^{|\mathrm{W}|})$ et par la méthode de Blakley avec l'équation (4.21).

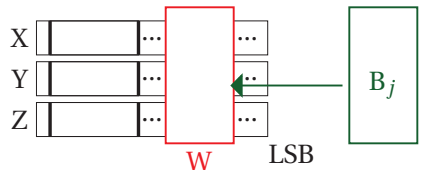


FIGURE 4.9 – Substitution des bits de W par B_j (où $j \in [\![0\,;\, n[\![$).

La figure 4.9 illustre comment notre méthode substitue W par B_j dans les coordonnées de chaque sommet de l'objet 3D secret. Cette substitution permet de générer n objets 3D appelés objets 3D partagés et notés $\mathcal{M}'_j$. Les coordonnées générées par la substitution de W par les mots binaires B_j les rendent différentes des coordonnées de l'objet 3D original. La substitution entraîne alors des modifications de l'apparence géométrique de l'objet 3D du fait de l'apparition de distorsions géométriques dans les objets 3D partagés.

Reconstruction d'objet 3D secret

Dans cette partie nous présentons le processus de reconstruction de l'objet 3D qui s'appuie sur celui de partage.

La figure 4.10 illustre la phase de reconstruction à partir de k objets 3D partagés parmi l'ensemble des n objets 3D générés précédemment. La méthode proposée extrait les mots

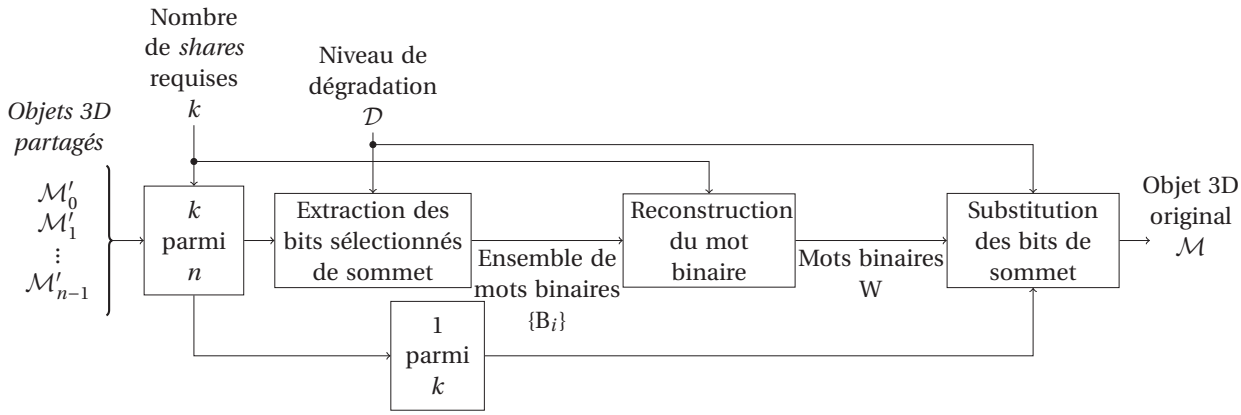


FIGURE 4.10 – Processus de reconstruction.

binaires $\{B_i\}$, où B_i est le mot binaire pour le sommet courant traité de l'objet 3D partagé du $i^{\text{ème}}$ utilisateur parmi les k présents lors du processus de reconstruction. L'ensemble des mots binaires est alors transféré à la méthode de reconstruction utilisée :

— Pour l'approche de Shamir, une interpolation de Lagrange (cf. Eq. (1.35)) est utilisée pour chaque sommet avec l'ensemble des points $\{(x_i, B_i)\}$. Pour chaque sommet, quand le nombre d'utilisateurs atteint ou dépasse le seuil k, l'interpolation retourne le mot binaire W qui est exactement le même que celui extrait de l'objet 3D original.

— Pour l'approche de Blakley, la méthode de reconstruction récupère un ensemble de coefficients représentant l'équation d'hyperplan de dimension k pour chaque sommet. La méthode rassemble les hyperplans associés au même sommet dans un ensemble et essaye de résoudre le système linéaire formé par les équations de ces hyperplans afin de récupérer le point secret S pour chaque sommet. Si les hyperplans appartiennent à l'ensemble autorisé des hyperplans et précédemment générés, alors les bits des coordonnées de S sont concaténés pour créer le mot binaire W.

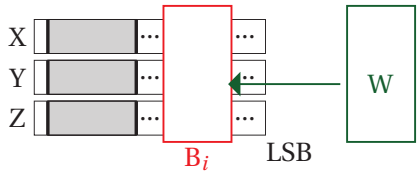


FIGURE 4.11 – Substitution des bits de B_i par W de l'objet 3D partagé $\mathcal{M}'_i$.

La figure 4.11 illustre comment notre méthode de reconstruction utilise un objet 3D partagé $\mathcal{M}'_i$ comme hôte et substitue les bits sélectionnés pour chaque sommet sommets par ceux du mot binaire reconstruit W à ce sommet. Cette opération est réitérée pour chaque sommet de l'objet 3D. Enfin, l'objet 3D reconstruit est identique à l'objet 3D secret sans perte d'informations.

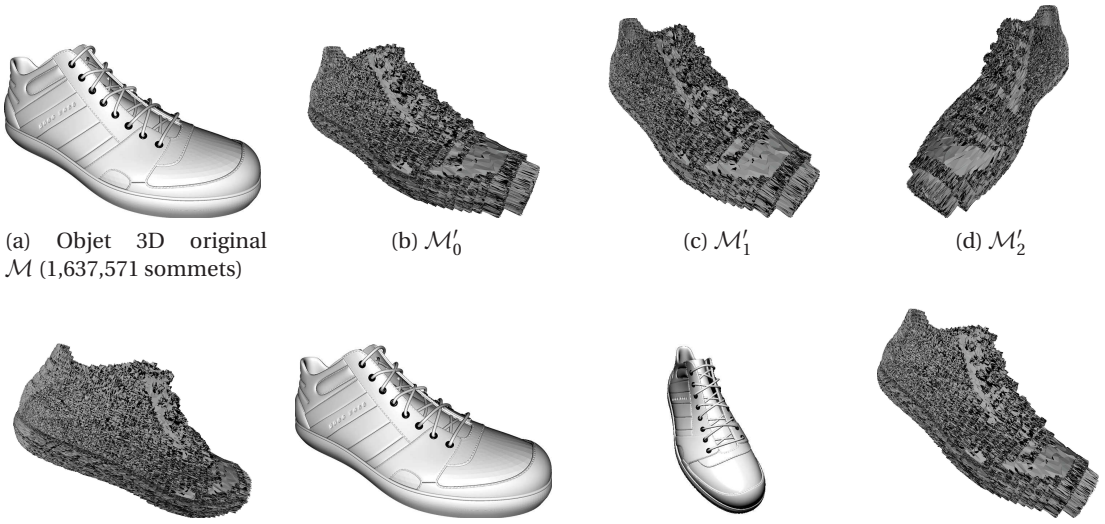
4.3.2 Résultats expérimentaux

Dans cette section, nous présentons des résultats expérimentaux obtenus avec notre méthode de partage sélectif d'objet 3D préservant le format original. Tout d'abord, nous présentons un exemple d'application de notre approche avec les deux méthodes de partage de secret issues de l'état de l'art. Nous analysons les objets 3D partagés à l'aide de

différentes métriques et étudions les effets du niveau de dégradation $\mathcal{D}$ et du partage des bits des coordonnées des sommets. Nous détaillons ensuite le calcul du nombre maximum d'objets 3D partagés pouvant être générés en fonction du niveau de dégradation $\mathcal{D}$ et de la méthode de partage de secret choisie. Nous évaluons alors la sécurité de notre méthode et sa robustesse contre des attaques se basant sur la reconstruction du contenu chiffré. Enfin, nous comparons notre méthode à celles de l'état de l'art du partage d'objet 3D présentées en section 2.3 du chapitre 2.

Application

Deux expérimentations sont présentées utilisant les paramètres ($k = 3$, $n = 4$) et $\mathcal{D} =< 18, 19 >$ avec la méthode de Shamir [116] et celle de Blakley [20]. La figure 4.12 illustre



(a) Objet 3D original $\mathcal{M}$ (1,637,571 sommets) (b) $\mathcal{M}'_0$ (c) $\mathcal{M}'_1$ (d) $\mathcal{M}'_2$

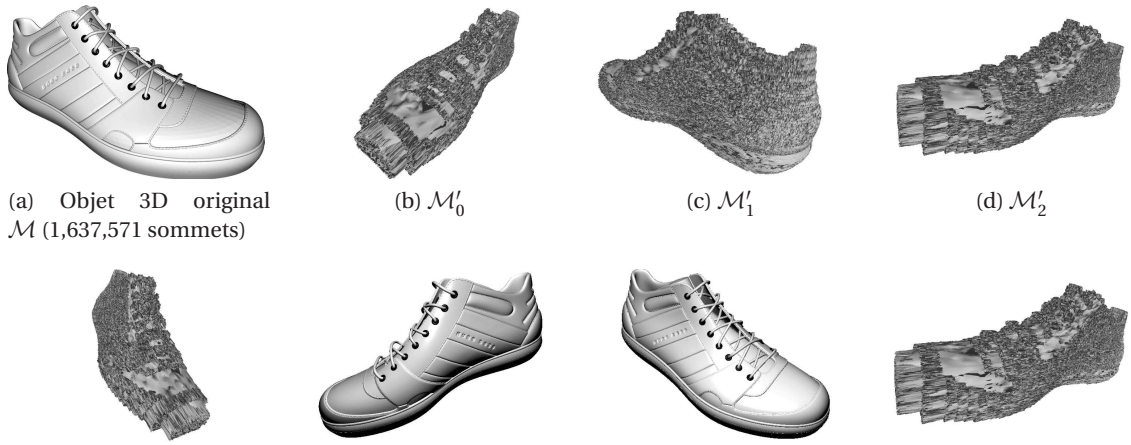
(e) $\mathcal{M}'_3$ (f) Reconstruction avec $(\mathcal{M}'_0, \mathcal{M}'_1, \mathcal{M}'_2)$ (g) Reconstruction avec $(\mathcal{M}'_1, \mathcal{M}_2, \mathcal{M}'_3)$ (h) Reconstruction avec $(\mathcal{M}'_1, \mathcal{M}'_2)$

FIGURE 4.12 – Résultats du processus de partage et de reconstruction de l'objet 3D $\mathcal{M}$ avec les paramètres $k = 3$, $n = 4$ et $\mathcal{D} =< 18, 19 >$ en utilisant la méthode de Shamir.

l'application de notre approche en utilisant la méthode de Shamir sur un objet 3D représentant une chaussure [1], noté $\mathcal{M}$, avec 1.6 million de sommets et 3 millions de triangles (figure 4.12.a). Les objets 3D représentés en figures 4.12.b-e sont les 4 objets 3D partagés générés en utilisant le processus de partage. Ces objets 3D ont le même nombre de sommets que l'objet 3D original, leur géométrie est dégradée, mais ils restent tout de même utilisables au sein d'environnements 3D. Les figures 4.12.f et 4.12.g montrent qu'à partir de n'importe quel groupe d'au moins 3 objets 3D partagés, il est possible de reconstruire l'objet 3D secret à l'identique. La figure 4.12.h représente un objet 3D reconstruit avec seulement 2 objets 3D partagés au lieu de 3. Dans ce cas, nous constatons que l'objet 3D reconstruit est autant dégradé que les objets 3D partagés.

Quand nous comparons les résultats utilisant la méthode de Shamir à ceux utilisant la méthode de Blakley, illustrés en figure 4.13, même si les dégradations géométriques ne sont pas exactement les mêmes, elles offrent un résultat très similaire en termes de dégradation pour le SVH.

1. Fourni par STRATEGIES (https://www.romans-cad.com/)



(a) Objet 3D original $\mathcal{M}$ (1,637,571 sommets) (b) $\mathcal{M}'_0$ (c) $\mathcal{M}'_1$ (d) $\mathcal{M}'_2$

(e) $\mathcal{M}'_3$ (f) Reconstruction avec $(\mathcal{M}'_0, \mathcal{M}'_1, \mathcal{M}'_2)$ (g) Reconstruction avec $(\mathcal{M}'_1, \mathcal{M}'_2, \mathcal{M}'_3)$ (h) Reconstruction avec $(\mathcal{M}'_0, \mathcal{M}'_1)$

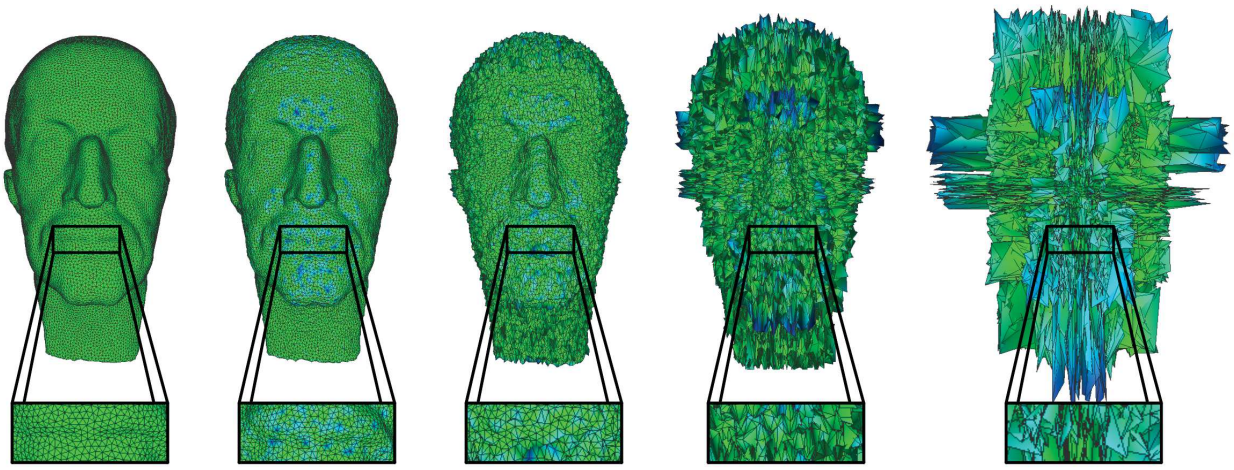
FIGURE 4.13 – Résultats du processus de partage et de reconstruction de l'objet 3D $\mathcal{M}$ avec les paramètres $k = 3$, $n = 4$ et $\mathcal{D} =< 18, 19 >$ en utilisant la méthode de Blakley.

Mesure de la dégradation visuelle

Nous avons comparé les objets 3D partagés générés par notre méthode avec l'objet 3D original à l'aide des métriques suivantes présentées en section 1.2.2 du chapitre 1 :

— La distance de Hausdorff (HD) ;

— La racine carrée de l'erreur quadratique moyenne (RMSE).



(a) Objet 3D original (b) $\mathcal{D} =< 16, 17 >$ (c) $\mathcal{D} =< 18, 19 >$ (d) $\mathcal{D} =< 20, 21 >$ (e) $\mathcal{D} =< 22, 23 >$

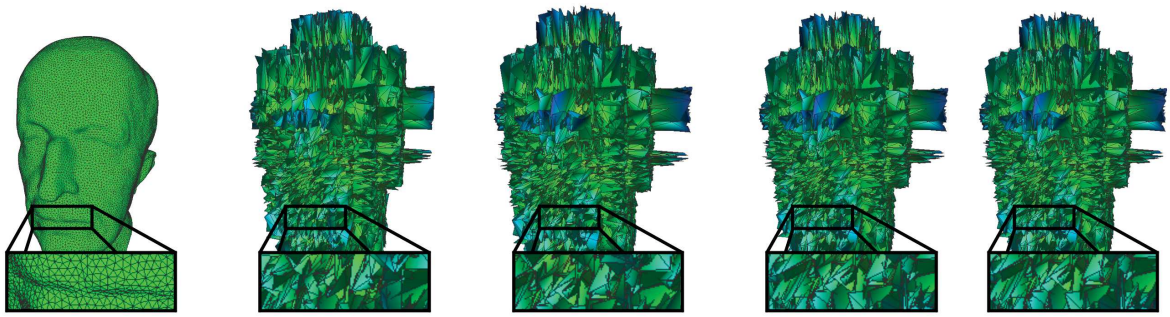
FIGURE 4.14 – Dégradation visuelle des objets 3D partagés en fonction du niveau de dégradation $\mathcal{D}$ désiré, illustrée par des cartes de distance entre sommets des objets 3D partagés et les objets 3D originaux utilisant la métrique RMSE (avec l'approche basée Shamir).

La figure 4.14 illustre avec des couleurs la distance entre les sommets de l'objet 3D original $\mathcal{M}$ avec leur sommet correspondant dans les objets 3D partagés générés à différents niveaux de dégradation. Plus un sommet est bleu, plus il est éloigné du sommet correspondant de l'objet 3D original. Nous observons que la position des premiers bits sélectionnés dans la mantisse influence grandement les distorsions géométriques comme

illustré en figure 4.14. Le tableau 4.5 récapitule les résultats des métriques HD et RMSE pour chaque niveau de dégradation expérimenté. Ce tableau révèle une augmentation de la distance des sommets testés des objets 3D partagés à leur correspondant dans l'objet 3D original quand $\mathcal{D}$ augmente.

TABLEAU 4.5 – Résultats des métriques HD et RMSE entre l'objet 3D secret et les objets 3D partagés selon différents niveaux de dégradations (avec l'approche basée Shamir).

$\mathcal{D}$	$< 16,17 >$	$< 18,19 >$	$< 20,21 >$	$< 22,23 >$
HD	0.00789055	0.0350072	0.126543	0.613757
RMSE	0.00298951	0.0119936	0.048057	0.199993



(a) Objet 3D original (b) $l = 1$ (c) $l = 8$ (d) $l = 15$ (e) $l = 22$

FIGURE 4.15 – Dégradation visuelle des objets 3D partagés en fonction du niveau de dégradation $\mathcal{D}$ désiré avec le paramètre fixé $p = 21$ et le paramètre l variant entre 1, 8, 15 et 22, illustrée par des cartes de distance entre sommets des objets 3D partagés et les objets 3D originaux utilisant la métrique RMSE (avec l'approche basée Shamir).

La figure 4.15 présente des objets 3D partagés par notre méthode utilisant l'approche de Shamir quand la valeur du paramètre p est fixée à 21 et la valeur du paramètre l est fixée à 1, 8, 15 ou 22. Nous observons que quelle que soit la valeur du paramètre l les objets 3D partagés sont visuellement similaires pour le SVH. Nous pouvons en conclure que la valeur du paramètre l du niveau de dégradation a donc un impact limité sur les distorsions géométriques.

TABLEAU 4.6 – Résultats des métriques HD et RMSE entre l'objet 3D secret et les objets 3D partagés pour le niveau de dégradation où $p = 21$ et différentes valeurs pour le paramètre l (avec l'approche basée Shamir).

$\mathcal{D}$	$< 21,1 >$	$< 21,8 >$	$< 21,15 >$	$< 21,22 >$
HD	0.262084	0.29119	0.290968	0.290968
RMSE	0.095659	0.09903	0.099027	0.099027

Le tableau 4.6 résume les valeurs des métriques HD et RMSE pour l'expérimentation présentée en figure 4.15. Nous observons que les valeurs de ces métriques sont proches entre les objets 3D partagés avec $l \in \{8, 15, 22\}$. Seul l'objet 3D partagé avec $l = 1$ possède de plus petites valeurs pour les métriques car la méthode proposée ne partage seulement que 3 bits par coordonnée. Ainsi, les résultats des métriques pour $l = 1$ correspondent aux distorsions géométriques minimales induites dans les objets 3D partagés pour ces niveaux de dégradation.

Même si la valeur du paramètre p est la plus importante à cause de son contrôle sur les distorsions géométriques, la valeur du paramètre l joue tout de même un rôle pour améliorer la sécurité de notre méthode. En effet, ce paramètre l contrôle le nombre de bits à utiliser pour stocker les données de partage et par extension, le nombre de bits à protéger par notre méthode.

Nombre maximum d'objets 3D partagés

Dans cette partie, nous présentons combien d'utilisateurs peuvent recevoir d'objets 3D partagés. Le nombre maximum d'objets 3D partagés n_{max} varie en fonction de la méthode de partage de secret choisie et du niveau de dégradation désiré $\mathcal{D}$. Pour la méthode de Shamir, n_{max} dépend du nombre d'éléments constituant le corps de Galois. En fonction du niveau de dégradation $\mathcal{D}$ et en particulier du paramètre ℓ, nous pouvons calculer ce nombre :

$$n_{max} = |\mathrm{GF}(2^m)| - 1 = |\mathrm{GF}(2^{3\times\ell})| - 1 = 2^{(3\times\ell)} - 1. \tag{4.22}$$

Par exemple, si $\mathcal{D} =< p, l = 3 >$, alors le corps de Galois utilisé $\mathrm{GF}(2^{(3\times l)})$ a $2^9 = 512$ éléments. De ce fait la méthode peut générer jusqu'à 511 objets 3D partagés.

Contrairement à la méthode de Shamir, où n_{max} dépend du nombre de bits à partager qui est déduit du paramètre l associé au niveau de dégradation $\mathcal{D}$, la méthode de Blakley requiert de stocker un ensemble de coefficients $\{\{a_i | i \in \{0, ..., k-2\}\}, b\}$ correspondant aux coefficients d'une équation d'un hyperplan de dimension k. Cela rend la méthode de Blakley moins efficace que celle de Shamir en termes de gestion d'espace. Il est possible d'estimer le nombre maximum d'objets 3D partagés pouvant être généré :

$$\forall i \in \{0, ..., k-2\}, |a_i| = [\frac{3 \times l}{2 \times (k+i)}], \tag{4.23}$$

$$n_{max} = \prod_{i=0}^{k-2} \mathrm{C}_1^{2^{|a_i|}} = \prod_{i=0}^{k-2} 2^{|a_i|}, \tag{4.24}$$

où $[x]$ est l'entier le plus proche de x.

Par exemple, si $\mathcal{D} =< p, l = 3 >$, alors 9 bits du sommet sont utilisés pour stocker les coefficients de l'hyperplan de dimension k associé au sommet. Comme la valeur de k définit également la dimension de l'hyperplan, elle détermine la manière dont les bits sélectionnés sont répartis uniformément pour les variables x_i et les coefficients a_i et b. La taille en bits des coefficients a_i peut être estimé selon l'équation (4.23).

TABLEAU 4.7 – Quantité maximale d'objets 3D partagés utilisant la méthode de Blakley pour le niveau de dégradation $\mathcal{D} =< p, l = 3 >$ en fonction de k.

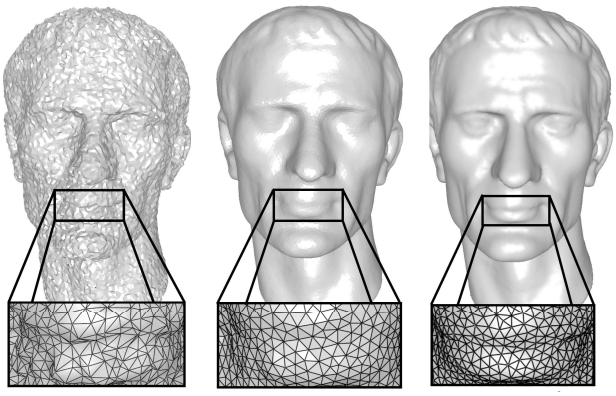
k	2	3	4	5	6
$(\lvert x_0\rvert, ..., \lvert x_{k-1}\rvert)$	(5, 4)	(3, 3, 3)	(3, 2, 2, 2)	(2, 2, 2, 2, 1)	(2, 2, 2, 1, 1, 1)
$(\lvert a_0\rvert, ..., \lvert a_{k-2}\rvert)$	(2)	(2, 1)	(1, 1, 1)	(1, 1, 1, 1)	(1, 1, 1, 1, 0)
$\lvert b\rvert$	7	6	6	5	5
n_{max}	4	8	8	16	16

Le tableau 4.7 résume la taille en bits des différents coefficients et variables en fonction de k pour le niveau de dégradation fixé $\mathcal{D} =< p, l = 3 >$. Lorsque la taille d'un coefficient a_i devient nulle, alors sa valeur est égale à 1. Nous remarquons également que le coefficient b nécessite beaucoup de bits pour représenter la somme de l'équation (1.36). L'utilisation d'une plus grande fenêtre glissante sur les coordonnées des sommets tout en

augmentant la valeur du paramètre l, permet d'utiliser davantage de bits pour les coefficients a_i et b. Dans ce cas, la méthode proposée peut produire plus d'objets 3D partagés.

Analyse de la sensibilité aux attaques

Le niveau de dégradation $\mathcal{D}$ induit directement le niveau de sécurité de l'objet 3D secret en définissant le nombre de bits sélectionnés dans chaque sommet de l'objet 3D secret pendant le processus de partage. Cependant, comme décrit dans [112], les méthodes de chiffrement partiel de données sont sensibles aux attaques cherchant à récupérer le contenu plutôt que de deviner la clé secrète. Un adversaire est en mesure d'utiliser des informations claires pour créer des attaques appropriées afin de reconstruire suffisamment le contenu. Puisque notre méthode proposée sélectionne des bits à partir des coordonnées de sommets, elle conserve alors en clair une partie des sommets de l'objet 3D d'origine. Des techniques de traitement des objets 3D, telles que des méthodes de filtrage [58, 127] ou même des approches par reconstruction telles que l'algorithme de *Marching Cubes* [93], peuvent contribuer à améliorer la qualité d'un objet 3D partagé. Néanmoins, même si un objet 3D lissé révèle davantage son contenu secret, la haute qualité de l'objet 3D est toujours protégée. Comme illustré sur la figure 4.16, la conservation de certaines propriétés de l'objet 3D peut permettre à des adversaires de concevoir des attaques pour reconstruire les bits les plus significatifs de l'objet 3D, mais la haute qualité de l'objet 3D ne peut pas être reconstruite (protection transparente).

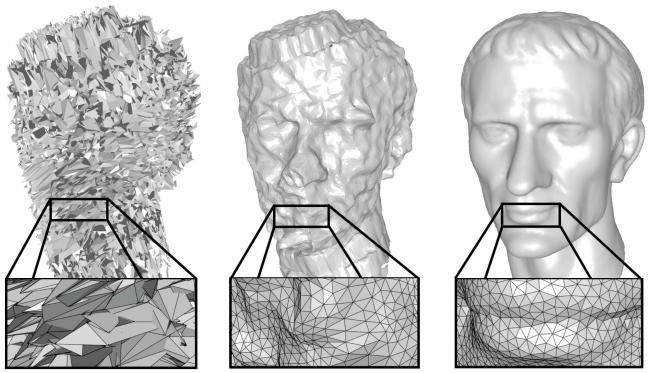


(a) Objet 3D partagé (b) Objet 3D partagé et lissé (c) Objet 3D original

FIGURE 4.16 – Attaque par filtrage laplacien sur un objet 3D partagé généré avec un niveau de dégradation $\mathcal{D} =< 18, 1 >$.

De plus, comme illustré sur la figure 4.17, une valeur plus élevée du niveau de dégradation $\mathcal{D}$ est capable de produire un objet 3D partagé plus sécurisé contre les attaques par traitement d'objet 3D. Ainsi, dans ce cas, le contenu de haute et de basse qualité de l'objet 3D secret (protection transparente et suffisante), est également protégé.

Le tableau 4.8 présente les résultats des métriques HD et RMSE pour les objets 3D partagés et lissés illustrés en figures 4.16 et 4.17. Nous pouvons observer que l'objet 3D lissé obtient des résultats inférieurs à ceux de l'objet 3D partagé de la figure 4.16. Bien que pour l'objet 3D partagé lissé le niveau de dégradation est $\mathcal{D} =< 21, 1 >$, nous remarquons que les résultats des métriques sont inférieurs à ceux de l'objet 3D partagé, mais pas aussi faibles que lorsque $\mathcal{D} =< 18, 1 >$. Les distorsions géométriques sont plus efficaces avec ce second niveau de dégradation, ce qui empêche la récupération d'un objet 3D de qualité bien supérieure.



(a) Objet 3D partagé (b) Objet 3D partagé et lissé (c) Objet 3D original

FIGURE 4.17 – Attaque par filtrage laplacien sur un objet 3D partagé généré avec un niveau de dégradation $\mathcal{D} =< 21, 1 >$.

TABLEAU 4.8 – Résultats des métriques HD et RMSE pour les objets 3D partagés et les objets 3D partagés lissés des figures 4.16 et 4.17.

Niveau de dégradation	Métrique	Objet 3D partagé	Objet 3D lissé
$\mathcal{D} =< 18, 1 >$	HD	0.0316368	0.018657
	RMSE	0.0116293	0.003460
$\mathcal{D} =< 21, 1 >$	HD	0.262084	0.191365
	RMSE	0.095659	0.049094

Un adversaire peut également utiliser une attaque par force brute pour essayer de reconstruire l'objet 3D secret. Une telle attaque requiert le bon ensemble de bits pour chaque sommet de l'objet 3D, ce qui correspond à la recherche de la bonne combinaison parmi $2^{3\times l\times V}$. Cependant, un adversaire peut décider de ne rechercher que les bits sélectionnés les plus significatifs de chaque coordonnée pour tous les sommets afin de réduire le niveau de dégradation de l'objet 3D partagé. Néanmoins, pour établir si les modifications apportées aux bits les plus significatifs sont pertinentes, un adversaire doit trouver un moyen d'évaluer si les modifications révèlent ou non le contenu secret. Dans les exemples précédents illustrés en figures 4.16 et 4.17, nous utilisons les métriques HD et RMSE, qui sont toutes les deux des métriques avec référence complète. Cela signifie que nous avons besoin de l'objet 3D original pour faire une comparaison, contrairement à un adversaire qui n'a pas accès à l'objet 3D original. En fonction du niveau de dégradation $\mathcal{D}$ appliqué lors de la phase de partage, un adversaire ne peut donc comparer ses résultats qu'avec une version lissée de l'objet 3D partagé qui, comme indiqué précédemment, ne présente aucune caractéristique de haute qualité.

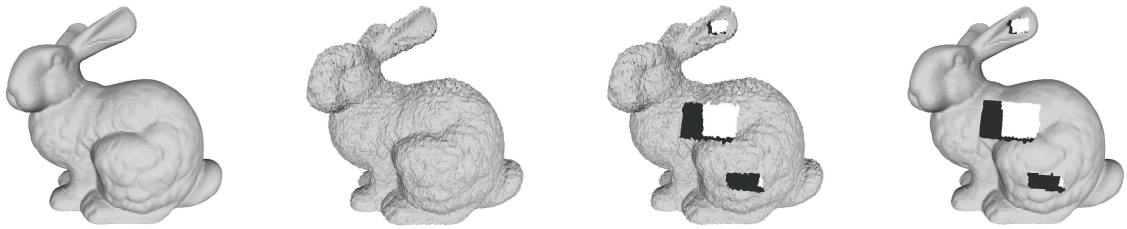
Comparaison avec l'état de l'art

Dans cette partie, Nous comparons les propriétés de notre méthode à celles de l'état de l'art actuel pour le partage d'objet 3D secret. Le tableau 4.9 présente un grand nombre de propriétés proposées dans l'état de l'art présenté en section 2.3 du chapitre 2. Nous y retrouvons les méthodes de Elsheh et Hamza [49], Anbarasi et Mala [4], Tsai [134] et Lee *et al.* [85]. Comme indiqué dans le tableau 4.9, l'un des avantages majeurs de notre méthode est qu'elle préserve le format des objets 3D. Grâce à cette propriété un objet 3D partagé distribué à un utilisateur peut toujours être visualisé en tant qu'objet 3D comme dans [85, 134]. De plus, notre approche propose une nouvelle fonctionnalité, à savoir

TABLEAU 4.9 – Comparaison de notre méthode avec les méthodes de l'état de l'art en matière de partage d'objet 3D secret [4, 49, 85, 134].

Propriétés & Paramètres	Elsheh et Hamza [49]	Anbarasi et Mala [4]	Tsai [134]	Lee *et al.* [85]	**Méthode proposée**
Méthode de partage	Blakley ou Thien & Lin	Shamir	Shamir	Reed-Solomon Codes	**Shamir ou Blakley**
Données partagées	Géométrie, Connectivité	Géométrie, Connectivité	Géométrie	Objets 3D décimés	**Géométrie**
Compression (Avant partage)	Sans perte (Huffman + ZLIB)	Sans perte (Huffman + ZLIB)	Subdivision spatiale	Décimation, EdgeBreaker, LZMA	**Aucune**
k	3	$[\![2\,;\,n[\![$	$[\![2\,;\,n[\![$	$[\![2\,;\,n[\![$	$[\![2\,;\,n[\![$ **ou** $f(n,\mathrm{D})$
n	P (premier)	P (premier)	255	Nombre d'objets 3D dans le groupe hôte	$2^{3\times l}-1$ **ou** $\prod_{j=0}^{k-2} 2^{\lvert a_j \rvert}$
Multiple	Non	Oui	Non	Oui	**Non**
Shares significatives	Non	Non	Oui	Oui	**Oui**
Préservation du format	Non	Non	Oui	Oui	**Oui**
Sélective	Non	Non	Non	Non	**Oui**
Taille des *shares*	Identique ou $\frac{1}{k}$	Identique	Grande	Grande	**Identique**
Sortie	n fichiers binaires	n fichiers binaires	n objets 3D stéganographiés	n objets 3D stéganographiés	**n objets 3D géométriquement distordus**

qu'elle peut protéger et partager l'objet 3D de manière sélective, en produisant un objet 3D partagé déformé avec un certain niveau de dégradation. Cela permet aux utilisateurs d'avoir un aperçu en basse qualité de l'objet 3D secret. À notre connaissance, aucune méthode de l'état de l'art actuel ne propose ce type de propriété. Les objets 3D partagés ont la même taille que l'objet 3D secret et ont tous le même niveau de dégradation. Du fait que les sommets de l'objet 3D secret sont partagés indépendamment, le calcul des mots binaires, peut donc être parallélisé. Il en est de même pour le processus de reconstruction. De plus, étant donné que les sommets sont partagés indépendamment, si les objets 3D partagés ont les mêmes sommets retirés ou perdus, notre méthode permet de reconstruire le contenu secret restant à condition d'avoir préservé la bonne correspondance des sommets entre eux, comme illustré en figure 4.18.



(a) Objet 3D original (b) Objet 3D partagé (c) Objet 3D partagé avec des trous (d) Objet 3D reconstruit

FIGURE 4.18 – Attaque par retrait de sommets. Les mêmes sommets ont été supprimés dans chaque objet 3D partagé.

4.3.3 Conclusion

Dans cette partie, nous avons proposé une méthode efficace de partage sélectif d'objet 3D secret préservant le format de l'objet 3D afin de permettre la visualisation des *shares* ayant la forme d'objets 3D. Nous avons ainsi présenté comment les méthodes de Shamir [116] et de Blakley [20] pouvaient être intégrées dans notre approche afin de préserver le format, protéger et partager un objet 3D. Pour ce faire, notre méthode distribue à n utilisateurs des objets 3D partagés dans lesquels une partie des bits des sommets de l'objet 3D secret sont sélectionnés puis substitués par ceux générés par la méthode de partage de secret choisie. Les sommets sont partagés indépendamment et les objets 3D partagés conservent la même taille que l'objet 3D d'origine. De ce fait, les objets 3D partagés générés par le processus de partage peuvent être visualisés dans des scènes 3D permettant un travail collaboratif. En fonction du niveau de dégradation souhaité, tous les objets 3D partagés sont plus ou moins fortement dégradés et présentent tous le même niveau de distorsions géométriques. De plus, ce niveau de dégradation peut augmenter la complexité de calcul en fonction de la méthode de partage de secret utilisée. Il permet donc aux producteurs de contenu de sélectionner le niveau de dégradation en fonction de leurs besoins en termes de confidentialité visuelle pour leurs objets 3D partagés. De cette manière, ils peuvent choisir des valeurs qui apportent une confidentialité visuelle totale ou un chiffrement transparent en passant par un chiffrement suffisant pour leurs objets 3D partagés, cela de la même manière que notre contribution sur le chiffrement sélectif 3D présenté dans le chapitre 3. Nous avons également comparé notre méthode avec des méthodes de partage d'objet 3D secret de l'état de l'art et souligné les avantages de notre méthode en termes d'efficacité. Grâce à l'utilisation des méthodes de Shamir ou de

Blakley, notre méthode peut intégrer de nouvelles propriétés et fonctionnalités apportées au cours des dernières décennies dans la communauté de partage de secret et de partage d'image secrète. Dans le chapitre 5, nous nous concentrons sur l'amélioration de notre méthode en intégrant des aspects hiérarchiques au partage d'objet 3D afin de rendre certains utilisateurs plus indispensable ou plus efficace dans la reconstruction de l'objet 3D secret. Dans le chapitre 6, nous évaluons la confidentialité des objets 3D chiffrés sélectivement en s'appuyant sur une campagne d'évaluations de plus de 50 observateurs.

4.4 Conclusion

Dans ce chapitre, nous avons présenté des nouvelles méthodes de partage de secret multimédia, et en particulier une méthode de partage d'image secrète ainsi qu'une méthode de partage d'objet 3D secret. Dans les deux approches, nous appliquons un partage où les éléments sont partagés de manière indépendante afin de permettre leur reconstruction. Ainsi, pour les images 2D, nous partageons les niveaux de gris des pixels, et les sommets pour les objet 3D.

Dans la première méthode, nous avons proposé une nouvelle approche pour le partage d'image secrète utilisant l'approche proposée par Blakley [20]. Notre approche permet de partager une image secrète entre 2 et 7 utilisateurs. Dès que 2 utilisateurs mettent en commun leurs *shares*, alors l'image secrète est reconstruite sans perte. Notre méthode code les équations de droites sur un seul octet de manière à avoir des *shares* ayant la même taille que l'image secrète et à représenter toutes les valeurs de niveau de gris uniformément. Nous avons appliqué notre méthode sur une base de 10.000 images et nous avons analysé statistiquement les *shares* générées par notre méthode. Il est aisé d'adapter notre méthode aux images en couleur en interprétant chaque canal couleur comme une image en niveaux de gris. Dans des travaux futurs, nous souhaitons augmenter le nombre d'utilisateurs nécessaire à la reconstruction sans augmenter la taille des *shares*, mais aussi inclure des fonctionnalités hiérarchiques où les utilisateurs possèdent des droits d'accès différents.

Dans notre seconde étude, nous avons proposé une méthode de partage sélectif d'objet 3D secret où chaque utilisateur reçoit un objet 3D partagé représentant l'objet 3D secret en basse qualité selon un niveau de dégradation choisi. Nous présentons notre méthode selon deux approches de partage de secret, à savoir celle de Shamir [116] et celle de Blakley [20]. Notre approche de partage s'inspire de nos travaux en chiffrement sélectif présenté dans le chapitre 3 pour permettre un partage préservant le format des objets 3D. Nous analysons statistiquement nos objets 3D partagés à l'aide de métriques objectives et nous discutons de la sécurité de notre méthode. Nous comparons notre méthode aux méthodes de l'état de l'art sur le partage d'objet 3D secret actuel. A ce jour, à notre connaissance, il s'agit de la seule méthode de partage sélectif d'objet 3D secret. Nous souhaitons continuer nos travaux sur le partage sélectif d'objet 3D secret en ajoutant de nouvelles fonctionnalités comme la reconstruction progressive, où en fonction du nombre d'utilisateurs présents à la reconstruction, l'objet 3D secret est plus ou moins reconstruit en haute qualité. Et enfin nous souhaitons inclure des fonctionnalités hiérarchiques établissant des droits d'accès différents aux objets 3D permettant de reconstruire plus ou moins rapidement l'objet 3D secret. Ainsi, dans le chapitre 5 nous proposons une amélioration de notre méthode de partage sélectif d'objet 3D secret intégrant les aspects hiérarchiques décrits en section 2.4 du chapitre 2.

Ces travaux ont fait l'objet de trois publications. Tout d'abord, la méthode de partage d'image secrète dans la conférence internationale IEEE MMSP en 2017 [11]. Enfin,

la méthode de partage d'objet 3D sélective secret dans la revue internationale IEEE Transactions on Multimedia en 2019 [16], ainsi que dans une conférence internationale IEEE ICASSP en 2019 [15].

Chapitre 5

Partage hiérarchique d'objet 3D secret

Sommaire

5.1 Introduction

Dans ce chapitre, nous présentons des nouvelles méthodes de partage hiérarchique d'image ou d'objet 3D secret. Dans une première partie de ce chapitre, nous proposons une nouvelle application des méthodes de partage hiérarchique de secret et d'image secrète pour résoudre un problème majeur et récent lié à la protection de la vie privée sur les réseaux sociaux présentés par Such et Criado [124]. Dans une deuxième partie de ce chapitre, nous présentons aussi une nouvelle méthode de partage hiérarchique et sélectif d'objet 3D secret préservant le format et proposant une hiérarchie où les utilisateurs ont des droits plus ou moins importants sur le contenu 3D en fonction de leur niveau dans cette même hiérarchie.

Pour notre première contribution, nous nous sommes inspirés des travaux effectués par Belenkiy [7] sur le partage hiérarchique de secret afin de proposer un nouveau processus de protection de la vie privée pour les réseaux sociaux en proposant une méthode de partage hiérarchique de régions d'intérêt au sein d'une image. Afin de protéger l'identité de personnes, notre méthode propose de considérer les visages comme des régions d'intérêt à protéger. Concernant la nouvelle méthode de partage d'objet 3D que nous proposons, celle-ci est une méthode de partage hiérarchique et sélectif d'objet 3D secret préservant le format et proposant de hiérarchiser les utilisateurs en fonction de leur droit d'accès au contenu 3D. Pour réaliser cela, nous avons repris nos travaux sur le partage sélectif d'objet 3D secret présenté en section 4.3 chapitre 4 afin de proposer des propriétés hiérarchiques dans le domaine du partage d'objet 3D secret et répondre à des cas d'utilisation où les utilisateurs n'ont pas les mêmes droits d'accès au contenu 3D.

Nous présentons notre contribution de partage de régions d'intérêt secrètes au sein d'images dans la section 5.2. En section 5.3, nous présentons notre contribution de partage hiérarchique et sélectif d'objet 3D préservant le format et proposant une hiérarchie d'accès prioritaire au contenu en fonction du niveau des utilisateurs présents lors de la reconstruction. Enfin, la section 5.4 conclut le chapitre et donne quelques pistes futures de recherche.

5.2 Partage hiérarchique de régions d'intérêt au sein d'images issues de réseaux sociaux

Dans cette section, à l'aide de méthodes de partage hiérarchique nous nous intéressons à répondre à une problématique récente concernant le respect de la vie privée. En effet, la sécurité multimédia est devenue un problème majeur ces dernières années. Avec la croissance exponentielle d'Internet, de plus en plus de données multimédia, images et vidéos, sont transmises sur les réseaux et stockées en ligne. Les réseaux sociaux sont déterminants dans la croissance des données multimédia avec le nombre de membres croissant de ces plateformes. Ainsi, en 2019 plus de 3,484 milliards d'utilisateurs sont actifs sur les réseaux sociaux au niveau mondial [60]. Les données transitant sur ces réseaux sont généralement personnelles et générées massivement par les utilisateurs eux-mêmes ce qui pose des problèmes sévères de sécurité et en particulier de respect de la vie privée. En effet, la protection de la vie privée d'autrui est un problème majeur lorsque des données partagées sur les réseaux impliquent plusieurs utilisateurs comme illustré en figure 5.1.

Dans l'exemple en figure 5.1, Dave prend une photographie avec ses amis et publie l'image sur sa page personnelle sur un réseau social. Selon le niveau de confidentialité

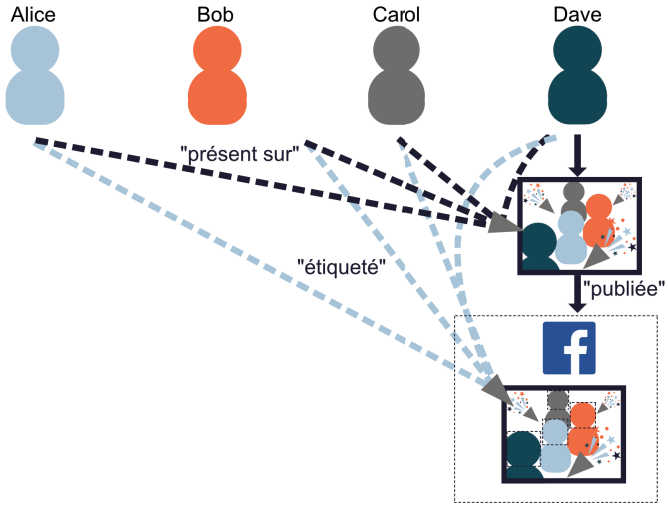


FIGURE 5.1 – Exemple de conflit de droit à la vie privée pluripartite où Dave diffuse une photographie représentant ses amis (Alice, Bob, Carol) sans leur consentement sur les réseaux sociaux.

des données que Dave a défini, un grand nombre de personnes peut accéder à cette image sur le réseau social. Cependant, ses amis présents aussi sur l'image ne font pas partie de la procédure de publication. En effet, ils ne sont pas forcément consultés avant la publication de l'image, qui contient des informations à leur sujet [124]. Dans ce contexte, il est nécessaire de proposer une solution efficace dans le but de gérer ces conflits pluripartites de droit à l'image (*Multiparty privacy conflicts*).

Nous constatons que les problèmes de conflits pluripartites de droit à l'image pour les réseaux sociaux reposent principalement sur la visibilité des visages des personnes au sein d'une image 2D. Ce sont ces types d'images qu'il est nécessaire de protéger tant que les utilisateurs ne donnent pas leur consentement. Il faut tout de même avoir la possibilité de révéler le contenu de manière partielle pour ceux qui ont autorisé l'affichage de leur visage. Les méthodes de partage d'image secrète peuvent être utilisées pour répondre à ce problème. Ainsi, comme présenté en section 2.2 du chapitre 2, ces méthodes permettent de partager une image entre n utilisateurs en distribuant des *shares* qui sont des images. Chaque *share* est personnelle et unique et permet la reconstruction de l'image secrète dès qu'au moins k parmi n sont rassemblées pour le processus de reconstruction, où k est le seuil requis d'utilisateurs présents déterminant un niveau de "confiance" entre les utilisateurs.

Nous proposons dans cette section de mettre en place un protocole basé sur le partage hiérarchique d'image secrète pour publier une image représentant plusieurs utilisateurs tout en préservant leur droit à la vie privée. Les utilisateurs, par exemple de notre figure 5.1, décident ensemble de définir un niveau de confiance au sein de leur groupe. Par le biais du réseau social, ils décident d'autoriser la publication de leur visage dans l'image si au moins k personnes parmi les n présentes dans l'image donnent leur accord. Quand Dave veut publier une photo de son groupe d'amis, chacun d'entre eux est questionné pour savoir s'il accepte la divulgation de son identité sur cette image. Si un utilisateur donne son consentement, en utilisant sa *share* et celle publique, son visage est alors reconstruit et accessible sur le réseau social. De plus, tant que $k' < k$ utilisateurs

acceptent de divulguer leur identité, seuls leurs visages sont révélés. Enfin, si k parmi n utilisateurs ont donné leur consentement, alors le contenu intégral de l'image est publié en clair. Sinon, le droit à la vie privée est respecté et leur identité reste protégée.

Dans un premier temps, nous décrivons en section 5.2.1 notre méthode de partage hiérarchique permettant de protéger la vie privée d'utilisateurs, dans ce cas leur visage sur l'image selon une structure d'accès multi-niveaux. En section 5.2.2, nous présentons des résultats expérimentaux de notre méthode. Enfin, en section 5.2.3, nous concluons sur les avantages de l'utilisation du partage hiérarchique de régions d'intérêt pour la protection de la vie privée des utilisateurs de réseaux sociaux.

5.2.1 Méthode

Notre méthode a pour but de protéger la vie privée d'utilisateurs de réseaux sociaux en proposant un mécanisme de protection de leur vie privée basé sur le partage de secret hiérarchique. Ainsi, lors de la publication d'une photographie où le visage d'un utilisateur est présent, la méthode protège le visage afin d'éviter que d'autres utilisateurs puissent y accéder tant que le consentement de l'utilisateur identifié n'a pas été donné.

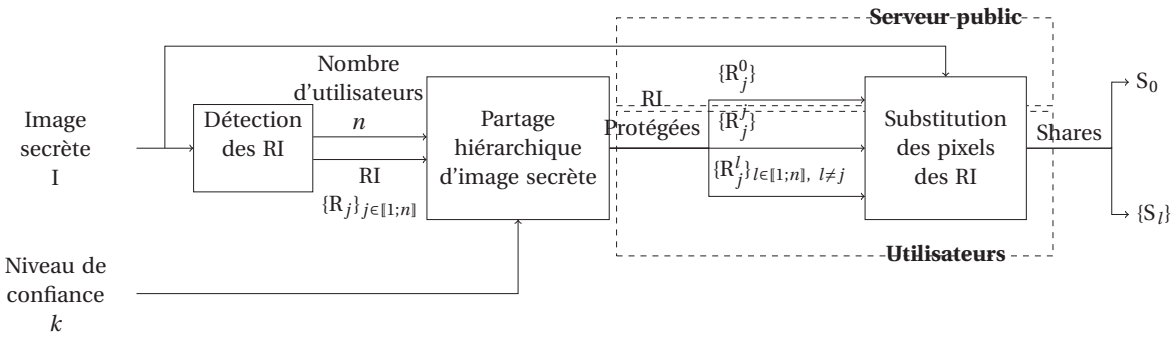


FIGURE 5.2 – Processus de partage des régions d'intérêt d'une image secrète.

Comme illustré en figure 5.2, notre méthode se divise en trois étapes principales. Tout d'abord, la détection des visages pour identifier les utilisateurs. L'étape suivante est le partage hiérarchique des régions d'intérêt (RI) correspondant aux visages des utilisateurs. Enfin, la substitution des RI, où les RI sont substituées par celles fournies en sortie du partage de secret et permettant de générer des *shares* sous la forme d'image pour chaque utilisateur ainsi qu'une *share* dite publique. Tout d'abord, l'image originale est une image représentant un groupe de n utilisateurs ; ainsi, en utilisant un algorithme de détection de visages, n RI notées R_j avec $j \in [\![1\ ;\ n]\!]$ sont identifiées et associées à chaque utilisateur. Dans ce manuscrit, nous ne nous attardons pas sur la détection de visages qui est un domaine de recherche hautement étudié [59, 86, 139] et utilisons directement des outils disponibles [23]. Les coordonnées de ces RI sont ensuite utilisées pour déterminer quelles zones de l'image secrète doivent être protégées par un partage hiérarchique d'image secrète. Les n utilisateurs définissent un niveau de confiance mutuelle k au sein de leur groupe. Ils choisissent un seuil tel que $1 \leq k \leq n$, indiquant le nombre minimum d'utilisateurs nécessaires pour reconstruire toutes les RI. Cela signifie que le contenu entier devient accessible dès qu'au moins k utilisateurs autorisent la reconstruction complète de l'image secrète. Notons que le reste de l'image, considéré comme l'arrière-plan de l'image, reste préservé en clair.

Partage hiérarchique des régions d'intérêt

Pour protéger les n RI, nous proposons d'utiliser la méthode de partage de secret multi-niveaux disjonctive proposée par Belenkiy [7] et présentée en section 2.4.4 du chapitre 2 en l'adaptant pour le partage d'image secrète. Dans le but d'éviter de la perte d'information, nous réalisons les opérations de partage de secret sur le corps de Galois GF(2^8) comme suggéré dans les travaux de Yang *et al.* [152]. Nous notons u_j où $j \in [\![1\ ;\ n]\!]$ la valeur (identifiant) qui est assignée à chaque utilisateur et la valeur u_0 désignant l'identifiant du serveur stockant la *share* publique :

$$\begin{aligned} &u_j \in \mathrm{GF}(2^8), \\ &u_j \neq 0, \\ &\forall j, l \in [\![0\ ;\ n]\!], j \neq l \Leftrightarrow u_j \neq u_l. \end{aligned} \tag{5.1}$$

Chaque RI R_j est partagée en $n+1$ *RI protégées* R_j^l, avec $l \in [\![0\ ;\ n]\!]$. Chaque *share* S_l a la même taille que l'image originale et la *share* S_0 correspond à la *share* publique. La *share* publique S_0 est composée d'un ensemble de pixels partagés (avec un niveau de confiance de 2) pour chacune des n RI $\{\mathrm{R}_j^0\}_{j\in[\![1\ ;\ n]\!]}$, ainsi que des pixels ne se trouvant pas dans les RI et restant en clair. Les autre *shares* $\{\mathrm{S}_l\}_{l\neq 0}$, contiennent aussi des pixels partagés (avec un niveau de confiance de 2) dans la RI R_j^j associée à l'utilisateur avec l'identifiant u_j, des pixels partagés (avec un niveau de confiance de $k+1$) pour les autres $n-1$ RI $\{\mathrm{R}_j^l\}_{j\in[\![1;n]\!],\ l\neq j}$ restantes et les pixels de l'arrière-plan de l'image originale.

Formellement, selon les notations présentées en section 2.4.4 du chapitre 2, chaque RI R_j va être partagée dans une hiérarchie de Belenkiy [7] comportant deux niveaux (L = 2) et où les seuils $\mathbf{k} = (k_0 = 2, k_1 = (k+1))$. Pour le groupe d'utilisateurs de niveau U_0, le serveur public (u_0) reçoit la *share* S_0 et l'utilisateur avec l'identifiant u_j reçoit la *share* S_j. Ces derniers peuvent reconstruire R_j ensemble, car $k_0 = 2$. Tandis que les autres utilisateurs appartiennent au groupe d'utilisateurs de niveau U_1 pour R_j, ces derniers avec l'aide du serveur public ne peuvent reconstruire le secret dès lorsque $k_1 = k+1$ utilisateurs sont consentants.

Pour obtenir les *shares*, chaque composante RVB de chaque RI R_j, où $j \in [\![1\ ;\ n]\!]$, est traitée séparément. Pour chaque composante, la valeur du canal pour un pixel est codée sur 8 bits. Les valeurs des pixels sont récupérées séquentiellement et interprétées comme des valeurs secrètes à partager. Une séquence aléatoire de valeurs $a_0, a_1, \ldots a_{k-1}$ est générée et la valeur de a_k est égale à la valeur du pixel. Ainsi, ces valeurs sont utilisées comme définie par le polynôme h_ℓ de l'équation (2.13) en section 2.4.4 du chapitre 2 :

$$h_\ell(u) = f^\Delta(u) = \frac{d^\Delta f}{du^\Delta}(u), \tag{5.2}$$

$$h_0(u) = f^{k+1-2} = f^{k-1}(u), \tag{5.3}$$

$$h_1(u) = f(u), \tag{5.4}$$

où $f^{(k-1)}$ est la $(k-1)^{\text{ème}}$ dérivation de la fonction $f(.)$

En utilisant ce polynôme, la valeur partagée associée à un pixel provenant de la RI R_j est calculée telle que :

— Dans la *share* publique S_0, le niveau de la hiérarchie est égale à 0 et il est associé au seuil d'accès $k_0 = 2$ dans le but de permettre la reconstruction de R_j possible dès lors que l'utilisateur avec l'identifiant u_j donne son consentement. La valeur partagée est égale à $h_0(u_0)$. Notons que cette valeur partagée est aussi utilisée dans la reconstruction de R_j, même si l'utilisateur avec l'identifiant u_j ne donne pas son

accord et seulement si le seuil de confiance de k utilisateurs est atteint lors du processus de reconstruction.

— L'utilisateur avec l'identifiant u_j possède le niveau 0 dans la hiérarchie, le plus élevé dans la reconstruction de R_j. Ainsi, l'utilisateur est capable de reconstruire sa région d'intérêt en utilisant sa *share* et celle publique S_0. La valeur partagée est égale à $h_0(u_j)$.

— Les utilisateurs avec un identifiant u_l, où $l \neq j$ et $l \neq 0$, ont le niveau 1 dans la hiérarchie, qui est associé au seuil $k_1 = k+1$. Ainsi, chaque valeur partagée dans R_j pour ces utilisateurs est égale à $h_1(u_l)$. Dans ce cas, la RI R_j ne peut être reconstruite seulement si au moins k utilisateurs donnent leur consentement à la reconstruction complète de l'image secrète.

Reconstruction des régions d'intérêt

Durant la phase de reconstruction, il existe deux scénarios possibles comme illustré dans la figure 5.3. Si le nombre d'utilisateurs participant à la reconstruction, noté k', est inférieur au niveau de confiance k, alors les k' *shares* et la *share* publique S_0 sont utilisées pour reconstruire les k' RI associés aux k' utilisateurs.

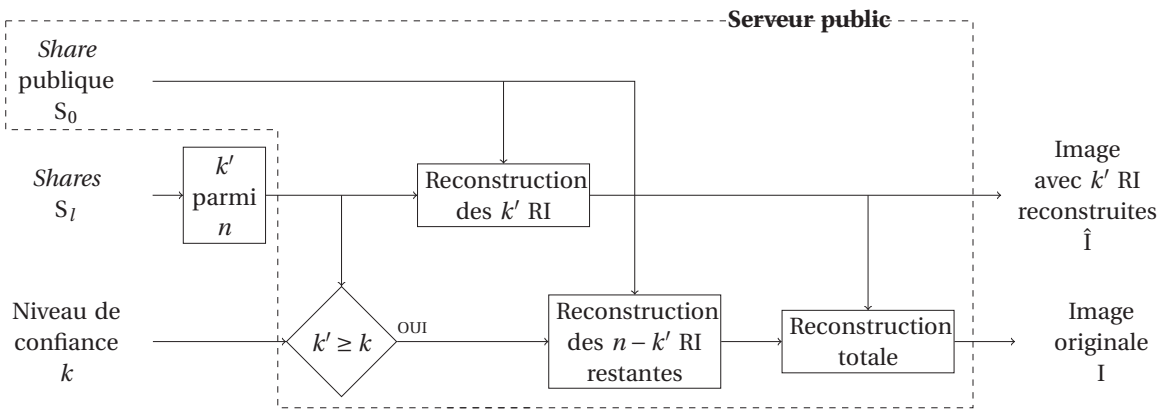


FIGURE 5.3 – Processus de reconstruction des régions d'intérêt d'une image secrète.

Pour chaque RI R_j et chaque composante couleur, les valeurs de pixels partagées peuvent être récupérées en utilisant une interpolation de Lagrange comme présentée en section 1.5.1 du chapitre 1. En effet, l'utilisateur avec l'identifiant u_j et le serveur public (avec l'identifiant u_0) peuvent construire ensemble un système linéaire déterminant la valeur des variables a_{k-1} et $a_k = s$ et par extension la valeur d'un pixel dans la région R_j. Ainsi, ces deux derniers sont du même niveau dans la hiérarchie de Belenkiy [7]. Notons que les $n-k'$ RI des utilisateurs qui ne participent pas à la reconstruction restent protégées.

Dans le second scénario, si le nombre d'utilisateurs k' est égal ou supérieur au niveau de confiance k, alors pour des raisons d'efficacité les k' RI des k' utilisateurs sont reconstruites comme pour le précédent scénario et les $n-k'$ RI restantes associées aux utilisateurs ne participant pas à la reconstruction sont aussi reconstruites comme illustré en figure 5.3. En effet, pour reconstruire les valeurs des pixels des $n-k'$ RI restantes, notre méthode utilise les valeurs des pixels des RI protégées dans les k' *shares* pour réaliser un système linéaire similaire à celui présenté à l'équation (2.15) pour la méthode de Belenkiy [7] en 2.4.4 du chapitre 2 permettant de déterminer la valeur du coefficient a_k correspondant à la valeur du pixel dans la RI courante.

Après avoir reconstruit les n RI $\{R_j\}$, où $j \in [\![1\ ;\ n]\!]$, nous substituons les valeurs de pixels de l'image publique des RI protégées par ceux des RI reconstruites pour créer l'image reconstruite complète qui est exactement identique à l'image originale.

5.2.2 Résultats expérimentaux

Dans la figure 5.4, nous présentons un exemple du processus complet de notre méthode avec les paramètres $k = 5$ et $n = 8$. Dans ce cas, huit utilisateurs ont été identifiés après la phase de détection de visage ($n = 8$) et les utilisateurs ont accepté de permettre la reconstruction de l'image originale lorsqu'au moins cinq d'entre eux donnent leur accord ($k = 5$). La figure 5.4.a illustre les RI identifiées. Dans notre approche, une *share* S_0, nommée *share* publique, est publiée sur les réseaux sociaux, comme illustré figure 5.4.b. De plus chaque utilisateur avec l'identifiant u_j, où $j \in [\![1; n]\!]$, reçoit une *share* privée S_j où toutes les RI sont protégées. Notons qu'en utilisant cette *share* privée S_j et la *share* publique S_0, alors chaque utilisateur peut reconstruire sa RI R_j correspondant à son propre visage. Par exemple, la figure 5.4.c montre une image partiellement reconstruite en utilisant la *share* S_2 associée à l'utilisateur avec l'identifiant u_2 et la *share* publique S_0. Seule la RI R_2 est en clair, tandis que toutes les autres RI restent protégées. Du moment que le nombre k' d'utilisateurs participant à la reconstruction est strictement inférieur au niveau de confiance k défini préalablement par les utilisateurs, l'image originale ne peut pas être reconstruite dans son intégralité. Par exemple, $k' = 3$ *shares* S_1, S_2 et S_4 (associées aux identifiants u_1, u_2 et u_4) et la *share* publique S_0 sont utilisées pour reconstruire l'image en figure 5.4.d. Ainsi, seul l'arrière-plan et les RI R_1, R_2 et R_4 des utilisateurs associés dans la reconstruction sont visibles en clair. Pour les utilisateurs n'ayant pas participé à la reconstruction, leur droit à l'image et à la vie privée reste respecté et leur identité demeure protégée. Dès lors que $k' \geq k$ *shares* sont rassemblées pour la reconstruction, par exemple $k' = 5$ comme en figure 5.4.e, alors le nombre d'utilisateurs est suffisant pour reconstruire l'image originale entièrement. Les k' *shares* S_1, S_3, S_5, S_7 et S_8 associées aux utilisateurs avec les identifiants u_1, u_3, u_5, u_7 et u_8 sont utilisées avec la *share* publique S_0 pour révéler les RI R_1, R_3, R_5, R_7 et R_8. De plus, les $n - k' = 3$ RI restantes R_2, R_4 et R_6 sont reconstruites en utilisant les k' *shares* et la *share* publique ensemble, car le seuil de confiance est atteint. Dans ce cas, l'image originale est reconstruite dans son intégralité et sans perte.

Le tableau 5.1 résume les résultats des différentes métriques, à savoir le PSNR (*Peak-Signal-Noise-Ratio*), le SSIM (*Structural SIMilarity*) et l'entropie pour chaque *share* S_l générée et pour chaque RI protégée R_j^l. Nous observons que pour les RI protégées, les valeurs de PSNR sont en dessous de 10 dB, les valeurs de SSIM sont presque à zéro et l'entropie est quasiment à 8 bits par pixel. Nous pouvons en conclure que statistiquement les RI sont protégées dans chaque *share*.

La figure 5.5 illustre l'histogramme des niveaux de gris de la RI R_1 de l'image originale I et dans la version protégée associée R_1^0 de la *share* S_0. Nous observons que la distribution des niveaux de gris à l'intérieur de la RI protégée R_1^0 est quasi-uniforme, signifiant que la méthode proposée est efficace pour rendre les valeurs confidentielles dans chaque *share* générée.

5.2.3 Conclusion

Dans cette partie, nous avons proposé une approche efficace pour assurer le respect du droit à la vie privée sur des images publiées sur des réseaux sociaux en se basant

FIGURE 5.4 – Illustration de la méthode proposée avec les paramètres $k = 5$, $n = 8$: a) Image originale après détection des RI associées aux huit utilisateurs (visages, en rouge), b) *Share* publique S_0 publiée sur les réseaux sociaux, c) Image partiellement reconstruite, en utilisant la *share* S_2 associée à l'identifiant u_2 et la *share* publique S_0, d) Image partiellement reconstruite, après avoir rassemblé les *shares* de $k' = 3$ utilisateurs (S_1, S_2 et S_4, associées à u_1, u_2 et u_4) et la *share* publique S_0, e) Image reconstruite dans son intégralité et parfaitement, après avoir rassemblé les *shares* de $k' = 5$ utilisateurs (S_1, S_3, S_5, S_7 et S_8, associées à u_1, u_3, u_5, u_7 et u_8) et la *share* publique S_0.

TABLEAU 5.1 – Résultats pour le PSNR, le SSIM et l'entropie des *shares* S_l provenant de la figure 5.4 pour chaque RI protégée par notre méthode, où bpp signifie bits par pixel.

	PSNR (dB)	SSIM	Entropie (bpp)
I	∞	1.0	7.489
R_1^l	7.463	0.0083	7.925
R_2^l	7.569	0.0070	7.945
R_3^l	8.044	0.0076	7.941
R_4^l	8.029	0.0018	7.935
R_5^l	9.445	0.0104	7.944
R_6^l	7.665	0.0073	7.949
R_7^l	8.380	0.0072	7.943
R_8^l	7.956	0.0078	7.956
Moyenne	8.069	0.0072	7.942

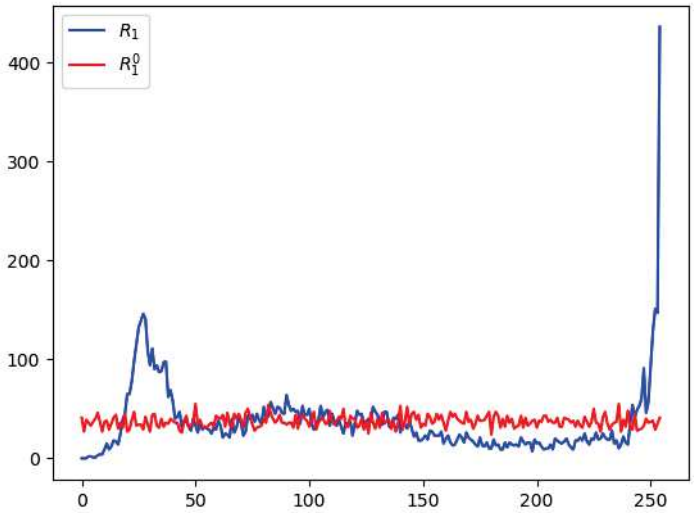
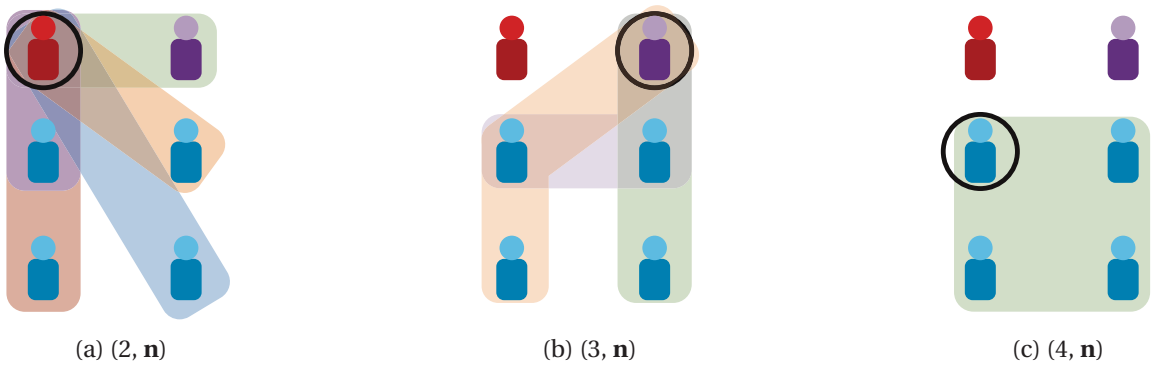


FIGURE 5.5 – Histogramme de la RI R_1 en clair et de la RI protégée R_1^0 dans la *share* S_0.

sur une méthode de partage hiérarchique des régions d'intérêt (RI) d'une image secrète. Notre approche permet de répondre à la problématique des conflits pluripartites de droit à l'image, où des utilisateurs ne souhaitent pas que leur identité soit divulguée au sein d'images diffusées par d'autres personnes sans leur consentement. Ainsi, nous avons proposé une application de la méthode de partage de secret de Belenkiy [7] pour les images. Les utilisateurs présents sont détectés sur l'image traitée par détection et reconnaissance faciale et des régions d'intérêt leur sont attribuées. Par accord mutuel, les n utilisateurs choisissent d'autoriser la visualisation en clair de toutes les régions d'intérêts lorsqu'au moins k d'entre eux autorisent la reconstruction. Si le seuil n'est pas atteint, alors seulement les régions d'intérêt associées aux utilisateurs participant à la reconstruction sont révélées à l'aide de la *share* publique S_0. Les résultats expérimentaux présentés valident notre l'efficacité de notre approche pour des cas pratiques d'utilisation. De plus, notre méthode peut être étendue à des utilisations plus traditionnelles des conflits de vie privée pluripartite, où tous les utilisateurs doivent donner leur consentement pour autoriser la reconstruction de l'image secrète. Dans ce, cas le seuil de confiance k est égal à n durant la phase de partage.

5.3 Partage hiérarchique et sélectif d'objet 3D secret

Dans cette section, nous présentons notre nouvelle approche de partage hiérarchique et sélectif d'objet 3D secret s'inspirant de nos travaux présentés en section 4.3 du chapitre 4 et des travaux proposés par Tassa [126] et Belenkiy [7] sur l'établissement de hiérarchie entre les utilisateurs présentés en section 2.4 du chapitre 2. Tout comme nos travaux précédents, nous reprenons notre approche de chiffrement sélectif 3D proposée au chapitre 3 et nous partageons par la méthode de Shamir [116] une sélection de bits au sein de la géométrie de l'objet 3D secret. Nous utilisons les approches de Tassa [126] et Belenkiy [7] pour intégrer un système de hiérarchie entre les utilisateurs. Nous proposons tout particulièrement un nouveau type de hiérarchie où les utilisateurs de niveaux supérieurs, par leur présence lors de la reconstruction, permettent de récupérer l'objet 3D secret avec des utilisateurs de n'importe quel niveau. Ainsi, contrairement à l'approche proposé par Belenkiy [7] notre méthode ne nécessite pas que tous les utilisateurs soient du même niveau pour reconstruire le contenu secret.



(a) (2, **n**) (b) (3, **n**) (c) (4, **n**)

FIGURE 5.6 – Groupes d'utilisateurs possibles autorisés à reconstruire l'objet 3D secret avec notre hiérarchie proposée, avec L = 3, où $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 4)$. Groupes autorisés à reconstruire à partir de : a) 2 utilisateurs, b) 3 utilisateurs, c) 4 utilisateurs.

Comme illustré en figure 5.6, nous reprenons l'exemple de Shamir présentant une hié-

rarchisation des utilisateurs avec le cas d'un président (en rouge), d'un vice-président (en violet) et de cadres (en bleu) [116], où L = 3 et $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 4)$. Ces utilisateurs sont classés en trois niveaux. Ainsi, lorsque le président (utilisateur de niveau 0) est inclus dans la reconstruction, alors il peut reconstruire l'objet 3D secret avec seulement un utilisateur de n'importe quel niveau. Tandis que les vice-présidents (utilisateurs de niveau 1) doivent être au moins trois et les cadres (utilisateurs de niveau 2) ont besoin d'être au moins quatre pour reconstruire l'objet 3D secret. La figure 5.6 illustre certains groupes d'utilisateurs autorisés à reconstruire l'objet 3D secret, avec le président, le vice-président et les cadres.

En section 5.3.1, la méthode de partage hiérarchique et sélectif d'objets 3D est décrite avec le nouveau type de hiérarchie proposé. Tandis que dans la section 5.3.2, les résultats expérimentaux sont évalués d'un point de vue statistique. Enfin en section 5.3.3, nous concluons sur notre méthode de partage hiérarchique et sélectif d'objet 3D secret et ses possibles futures améliorations.

5.3.1 Méthode

Dans cette section, nous présentons notre méthode de partage hiérarchique à accès prioritaire et sélectif d'objet 3D secret préservant le format 3D nommé *Priority Access Hierarchical Format-Compliant* $(\mathbb{L}, \boldsymbol{k}, \boldsymbol{n}, \mathcal{D})$ *Selective Secret 3D Object Sharing* (PAHFCSS3DOS) implémentant la hiérarchie à accès prioritaire (*Priority Access Hierarchy* ou PAH). Notre approche consiste à partager un objet 3D secret parmi un ensemble d'utilisateurs en protégeant de manière sélective les bits de la géométrie. La particularité de cette méthode face à celle proposée en section 4.3 du chapitre 4 est de rendre plus accessible la reconstruction de l'objet 3D secret pour certains groupes d'utilisateurs en fonction de leur niveau au sein de la hiérarchie. Ainsi, la spécificité de la PAH est d'autoriser la reconstruction de l'objet 3D secret avec moins d'utilisateurs que classiquement nécessaire, si les utilisateurs participant au processus sont placés en haut de la hiérarchie.

Tout d'abord, nous présentons la hiérarchie PAH. Cette structure permet aux utilisateurs de niveaux supérieurs de reconstruire l'objet 3D en priorité. Nous détaillons ensuite le processus complet de notre méthode de partage d'objet 3D. Enfin, nous présentons la méthode de reconstruction de l'objet 3D secret, selon les différentes configurations de groupe d'utilisateurs de niveaux différents.

Une nouvelle hiérarchie : Hiérarchie à accès prioritaire (PAH)

Avec ce nouveau type de hiérarchie, les utilisateurs de niveaux supérieurs sont capables de reconstruire l'objet 3D secret selon le seuil associé à leur groupe d'utilisateurs de même niveau comme pour la méthode proposée par Belenkiy [7] présentée en section 2.4.4 du chapitre 2. Cette PAH propose aussi que ces utilisateurs de niveaux supérieurs, selon le seuil qui leur est associé, puissent aussi reconstruire avec des utilisateurs de niveaux inférieurs également. Dans la hiérarchie proposée par Belenkiy, quand k_ℓ, le nombre d'utilisateurs nécessaires au niveau ℓ pour reconstruire l'objet 3D, n'est pas atteint, alors les utilisateurs de niveau ℓ sont utilisés pour la reconstruction au niveau $(\ell+1)$ avec le seuil $k_{\ell+1}$. Notre PAH permet aux utilisateurs de reconstruire l'objet 3D secret avec k_ℓ utilisateurs, quand au moins un d'entre eux est un utilisateur de niveau ℓ. Formellement, soit Γ l'ensemble de groupes autorisés à reconstruire l'objet 3D secret et U l'ensemble d'utilisateurs tous niveaux confondus, tels que :

$$\begin{cases} \mathrm{G} \subseteq \mathrm{U}, \\ \Gamma = \{\mathrm{G}, \exists\, u \in \mathrm{G} \text{ tel que } |\mathrm{G}| \geq k_{\mathbb{L}(u)}\}. \end{cases} \tag{5.5}$$

Pour obtenir ce type de hiérarchie, nous utilisons l'approche de Belenkiy [7] présentée en section 2.4.4 du chapitre 2. Mais nous gardons les coefficients a_r, où $r \in [\![0\,;\, k_{\mathrm{L}-1} - 1[\![$, qui sont utilisés durant le partage et les distribuons aux utilisateurs en fonction de la présence de ces coefficients dans leur polynôme $h_{\mathbb{L}(u)}$. Formellement, nous notons A_ℓ l'ensemble des coefficients distribués pour chaque niveau ℓ tel que :

$$\mathrm{A}_\ell = \{a_t | \forall\, t \in [\![0\,;\, k_{\mathrm{L}-1} - k_\ell[\![\}. \tag{5.6}$$

Par exemple, pour une hiérarchie à trois niveaux avec les paramètres $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 4)$, chaque utilisateur reçoit un identifiant unique, une *share* et un ensemble de coefficients :

— Les utilisateurs de niveau 0 reçoivent leur u, $h_0(u)$ et les coefficients $\mathrm{A}_0 = \{a_0, a_1\}$;

— Les utilisateurs de niveau 1 reçoivent leur u, $h_1(u)$ et les coefficients $\mathrm{A}_1 = \{a_0\}$;

— Les utilisateurs de niveau 2 reçoivent leur u, $h_2(u)$ (aucun coefficient n'est distribué).

Ces coefficients sont utilisés pour réduire le nombre d'inconnues au sein du système linéaire utilisé pour la reconstruction des données.

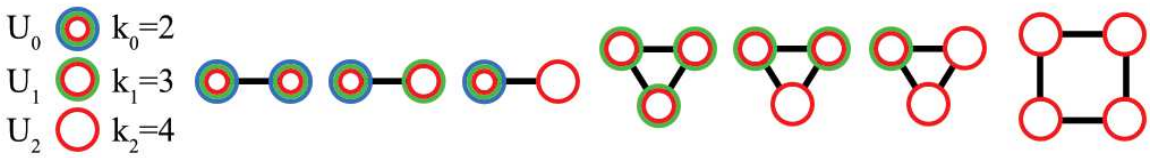


FIGURE 5.7 – Liste de groupes permettant de reconstruire le secret partagé avec la PAH en utilisant les paramètres $\mathrm{L} = 3$ et $\mathbf{k} = (2,\ 3,\ 4)$.

Comme illustré en figure 5.7, pour une hiérarchie à trois niveaux avec les paramètres $\mathbf{k} = (k_0 = 2,\ k_1 = 3,\ k_2 = 4)$, l'objet 3D secret peut être reconstruit avec seulement deux utilisateurs si au moins un d'entre eux est de niveau 0, avec seulement trois utilisateurs si au moins un d'entre eux est de niveau 1 et quatre utilisateurs sont nécessaire lorsque ces derniers sont tous de niveau 2.

Partage hiérarchique à accès prioritaire et sélectif d'objet 3D secret

Comme illustré en figure 5.8, en plus de l'objet 3D à partager, notre méthode requiert cinq paramètres en entrée, à savoir L le nombre de niveaux dans la hiérarchie, $\mathbf{k} = \{k_\ell\}$ un vecteur représentant le nombre minimum d'utilisateurs nécessaire à la reconstruction pour chaque niveau dans la hiérarchie avec $\ell \in [\![0\,;\, \mathrm{L}[\![$, $\mathbf{n} = \{n_\ell\}$ un vecteur représentant le nombre d'objets 3D partagés à générer pour chaque niveau dans la hiérarchie où le nombre total est noté $|\mathbf{n}| = \sum_{\ell=0}^{\mathrm{L}-1} n_\ell$, $\mathcal{D}$ le niveau de dégradation désiré des objets 3D partagés à générer et K_r un ensemble de clés secrètes à utiliser pour initialiser différents générateurs de nombres pseudo-aléatoires (GNPA) dans le but de produire les coefficients des polynômes utilisés durant le processus de partage de chaque sommet où $r \in [\![0\,;\, k_{\mathrm{L}-1} - 1[\![$. La méthode est divisée en quatre étapes principales, à savoir la sélection des bits des sommets, la génération des coefficients des polynômes, le partage des mots binaires extraits

et la génération des objets 3D partagés.

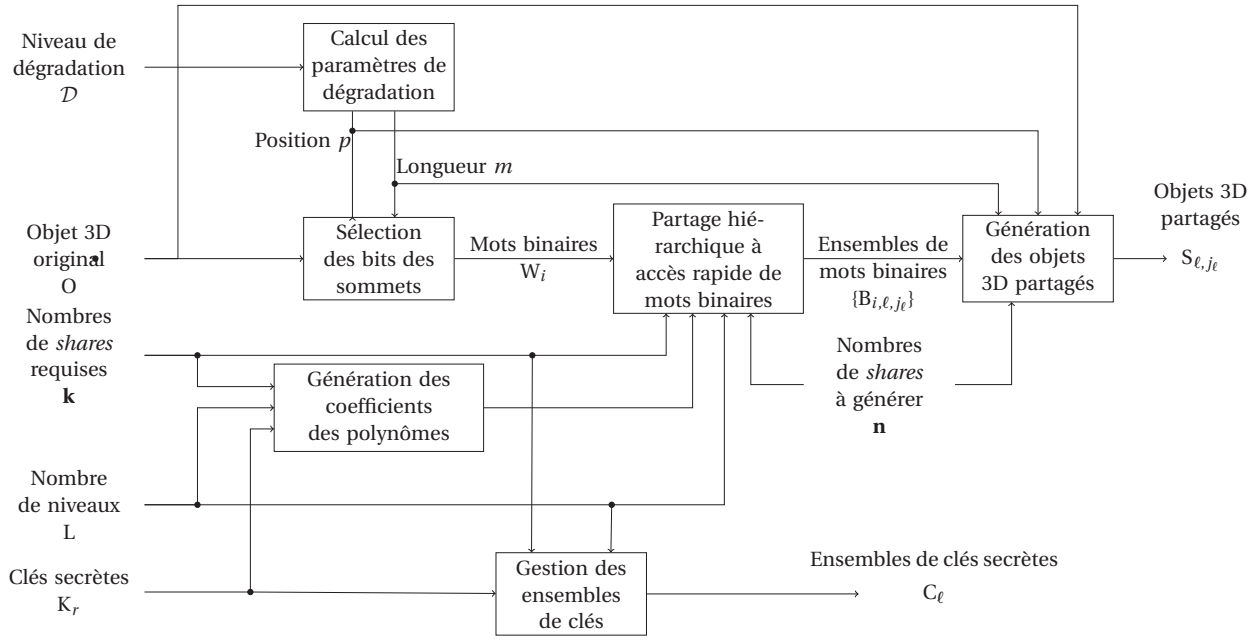


FIGURE 5.8 – Processus de partage de notre méthode PAHFCSS3DOS.

Sélection des bits des sommets Dans le but de partager un objet 3D tout en préservant le format, notre méthode applique une méthode de partage de secret sur les bits représentant les coordonnées des sommets de l'objet 3D. Le niveau de dégradation désiré $\mathcal{D}$ détermine quel intervalle de bits est sélectionné pour les étapes de partage et de substitution. Les étapes de sélection des bits des sommets et de calcul des paramètres de dégradation sont basées sur les travaux présentés en section 4.3.1 du chapitre 4. Le niveau de dégradation $\mathcal{D}$ peut varier en fonction des deux paramètres que sont $p \in [\![0\ ;\ 22]\!]$, la position du premier bit sélectionné et $l \in [\![1\ ;\ p+1]\!]$, la longueur de l'intervalle de bits à partager. La séquence de bits sélectionnés est récupérée sous la forme d'un mot binaire noté W_i pour chaque sommet de l'objet 3D secret, où $|W_i| = 3 \times l$ et $i \in [\![0\ ;\ |V|[\![$.

Génération de coefficients de polynôme À cause de la structure de PAH, la méthode doit stocker les coefficients A_ℓ décrits précédents avec le résultat du polynôme utilisé h_ℓ en fonction du niveau de l'utilisateur. De plus, dans le cas de notre méthode, tous les sommets sont partagés indépendant ce qui requiert de stocker $|V|$ ensembles de coefficients additionnels. Ainsi, pour éviter un agrandissement de la taille de la *share*, stocker ces coefficients additionnels et préserver le format de l'objet 3D, nous proposons de générer les coefficients pseudo-aléatoirement et de transmettre seulement les clés secrètes utilisées en fonction du niveau des utilisateurs dans la hiérarchie. Normalement, un GNPA génère tous les coefficients des polynômes utilisés pour partager des secrets et la clé secrète ayant servi à initialiser le générateur n'est pas conservée. Nous proposons d'avoir plusieurs GNPA initialisant les coefficients de même degré dans tous les polynômes utilisés pour partager les sommets. Ainsi, notre méthode utilise $(k_{L-1} - 2)$ clés secrètes distinctes K_r pour initialiser les $(k_{L-1} - 2)$ GNPA notés R_r. Ces GNPA R_r génèrent les coefficients $a_{i,r}$ pour les polynômes de degré $(k_{L-1} - 1)$ pour tous les sommets de l'objet 3D. De cette manière, au lieu de stocker les coefficients avec chaque mot binaire distribué en fonction du

niveau d'un utilisateur $\mathbb{L}(u_j)$, nous transmettons seulement, avec un objet 3D partagé, à chaque utilisateur un ensemble de clés permettant de générer les coefficients requis pour la reconstruction dans le cadre de la PAH présentée en section 5.3.1. Par exemple, pour une hiérarchie à trois niveaux avec les paramètres $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 4)$, 3 GNPA, notés R_r, sont utilisés pour générer les coefficients $a_{i,r}$ avec $i \in [\![0\,;\ |\mathrm{V}|[\![$ et $r \in [\![0\,;\ k_{\mathrm{L}-1} - 1[\![$, comme illustré dans le tableau 5.2.

TABLEAU 5.2 – Exemple de génération des coefficients des polynômes $a_{i,r}$ avec les GNPA R_r avec les paramètres $\mathrm{L} = 3$ et $\mathbf{k} = (2,3,4)$.

$i^{\text{ème}}$ sommet	R_0	R_1	R_2
0	$a_{0,0}$	$a_{0,1}$	$a_{0,2}$
1	$a_{1,0}$	$a_{1,1}$	$a_{1,2}$
...	...	...	...
$\lvert\mathrm{V}\rvert - 1$	$a_{\lvert\mathrm{V}\rvert-1,0}$	$a_{\lvert\mathrm{V}\rvert-1,1}$	$a_{\lvert\mathrm{V}\rvert-1,2}$

Partage hiérarchique à accès prioritaire de mot binaire Durant cette étape, à partir du mot binaire W_i extrait pour chaque sommet, la méthode génère $|\mathbf{n}|$ mots binaires $\mathrm{B}_{i,\ell,j_\ell}$, où $j_\ell \in [\![0\,;\ n_\ell[\![$ tel que $|\mathrm{B}_{i,\ell,j_\ell}| = |\mathrm{W}_i|$. Notre approche réalise les opérations de partage en utilisant un corps de Galois $\mathrm{GF}(2^{|\mathrm{W}_i|})$, où $|\mathrm{W}| = 3 \times l$ pour s'assurer de la condition $|\mathrm{B}_{i,\ell,j_\ell}| = |\mathrm{W}_i|$. Cela augmente le nombre maximum d'utilisateurs et la sécurité de notre méthode selon la sélection de bits définie par le niveau de dégradation $\mathcal{D}$ désiré. Chaque utilisateur reçoit un unique identifiant u_j et un niveau dans la hiérarchie $\mathbb{L}(u_j)$ avec $j \in [\![0\,;\ |\mathbf{n}|[\![$ tel que :

$$\begin{cases} u_j \in \mathrm{GF}(2^{|\mathrm{W}_i|}), \\ u_j \neq 0, \\ \mathbb{L}(u_j) = \ell \in [\![0\,;\ \mathrm{L}[\![, \\ \forall j, j' \in [\![0\,;\ |\mathbf{n}|[\![,\ j \neq j' \Rightarrow u_j \neq u_{j'}. \end{cases} \tag{5.7}$$

Pour chaque sommet, W_i est partagé en utilisant l'identifiant u_j de chaque utilisateur, le niveau de l'utilisateur $\mathbb{L}(u_j)$ et les coefficients $\{a_{i,r}\}$ tels que $a_{i,r} \in \mathrm{GF}(2^{|\mathrm{W}_i|})$, où $a_{i,r}$ est généré pseudo-aléatoirement utilisant un GNPA avec la clé secrète K_r. Pour notre HAR présentée en section 5.3.1, le coefficient $a_{k_{\mathrm{L}-1}}$ est assigné à la valeur de W_i comme pour la hiérarchie de Belenkiy. Une fois que les coefficients sont initialisés, $|\mathbf{n}|$ mots binaires $\mathrm{B}_{i,\ell,j_\ell}$, sont générés en utilisant le polynôme déterminé pour le niveau d'utilisateur $\mathbb{L}(u_j)$.

Génération des objets 3D partagés Les mots binaires $\mathrm{B}_{i,\ell,j_\ell}$ générés précédemment respectent la condition $|\mathrm{B}_{i,\ell,j_\ell}| = |\mathrm{W}_i|$ grâce à l'utilisation d'un corps de Galois $\mathrm{GF}(2^{|\mathrm{W}_i|})$. Notre méthode réutilise la même stratégie de synchronisation employée à l'étape de sélection des bits des sommets pour substituer W_i par $\mathrm{B}_{i,\ell,j_\ell}$ au sein des coordonnées du $i^{\text{ème}}$ sommet de l'objet 3D secret. Ce processus permet de générer $|\mathbf{n}|$ objets 3D appelés objets 3D partagés et que nous notons S_{ℓ,j_ℓ}. Ces objets 3D partagés sont différents de l'objet 3D secret par la présence de distorsions géométriques. Les coordonnées générées par la substitution de W_i par les mots binaires $\mathrm{B}_{i,\ell,j_\ell}$ sont transformées et deviennent différentes de celles des sommets de l'objet 3D original.

Gestion des ensembles de clés Comme expliqué précédemment, nous utilisons des clés secrètes K_r, où $r \in [\![0\,;\, k_{L-1}[\![$ pour générer les coefficients des polynômes utilisés pour partager les mots binaires W_i. Pour éviter de stocker les coefficients des polynômes, nous avons besoin de distribuer ces clés secrètes en fonction du niveau de chaque utilisateur dans la hiérarchie. Ainsi, notre méthode calcule les ensembles de clés secrètes C_ℓ qui sont transmis de manière sécurisée à tous les utilisateurs selon leur niveau dans la hiérarchie, tel que :

$$C_\ell = \{K_t | \forall\, t \in [\![0\,;\, k_{L-1} - k_\ell[\![\}. \tag{5.8}$$

Reconstruction de l'objet 3D secret

La figure 5.9 illustre le processus de reconstruction d'un objet 3D secret à partir des objets 3D partagés précédemment générés. Pour la PAH proposée en section 5.3.1, cer-

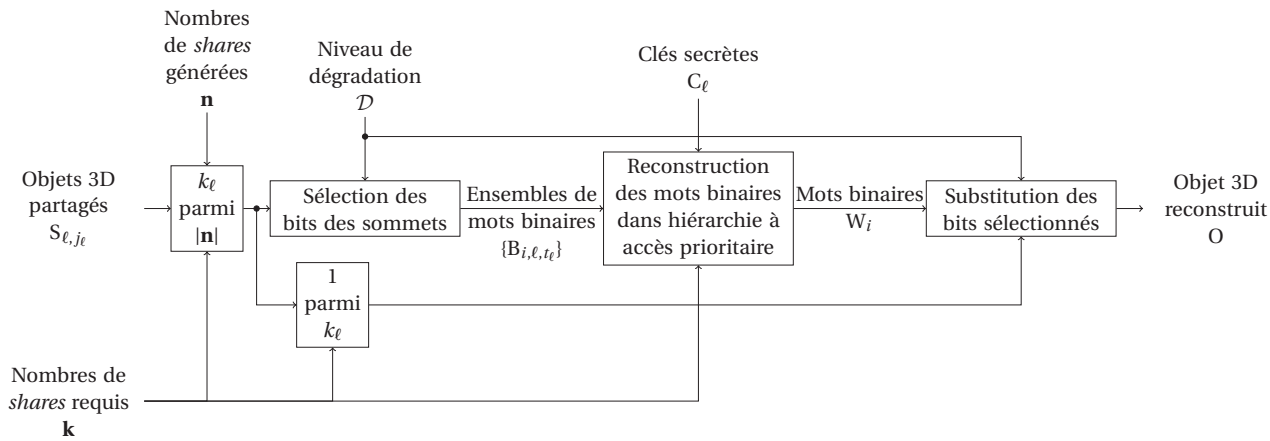


FIGURE 5.9 – Processus de reconstruction de l'objet 3D secret.

tains utilisateurs ou groupes d'utilisateurs, en fonction de leur niveau, peuvent reconstruire l'objet 3D secret plus avec moins d'utilisateurs requis. En utilisant l'exemple précédent avec les paramètres $\mathbf{k} = (k_0 = 2, k_1 = 3, k_2 = 4)$, un utilisateur u_{j_0} de niveau 0 possède comme informations son identifiant u_{j_0}, sa *share* $h_0(u_{j_0})$ et l'ensemble de clés $C_0 = \{K_0, K_1\}$ correspondant au niveau 0 pour reconstruire la valeur des coefficients $a_{i,0}$ et $a_{i,1}$. La phase de reconstruction de l'objet 3D secret est similaire à ce que propose Belenkiy pour son approche. Si des utilisateurs avec des niveaux inférieurs à celui de l'utilisateur u_{j_0} sont présent, alors les coefficients sont utilisés pour éliminer des inconnues au sein du système linéaire permettant la reconstruction d'un mot binaire $\hat{W}_i$. Par exemple pour un utilisateur de niveau 1 :

$$\begin{bmatrix} 0 & 2 & 6u_{j_0} \\ 1 & 2u_{i_1} & 3u_{j_1}^2 \end{bmatrix} \times \begin{bmatrix} a_{i,1}^* \\ a_{i,2} \\ a_{i,3} \end{bmatrix} = \begin{bmatrix} h_0(u_{j_0}) \\ h_1(u_{j_1}) \end{bmatrix}, \tag{5.9}$$

ce qui correspond à :

$$\begin{bmatrix} 2 & 6u_{j_0} \\ 2u_{j_1} & 3u_{j_1}^2 \end{bmatrix} \times \begin{bmatrix} a_{i,2} \\ a_{i,3} \end{bmatrix} = \begin{bmatrix} h_0(u_{j_0}) \\ h_1(u_{j_1}) - a_{i,1} \end{bmatrix}. \tag{5.10}$$

Si le groupe d'utilisateurs présent à la reconstruction de l'objet 3D secret est autorisé, alors le mot binaire reconstruit $\hat{W}_i$ est égal à W_i extrait durant le processus de partage.

Dans ce cas, la méthode utilise un des objets 3D partagés comme hôte et substitue les bits sélectionnés par le mot binaire $\hat{W}_i$ reconstruit pour chaque sommet. Enfin, l'objet 3D reconstruit est identique à l'objet 3D secret sans distorsion ou perte d'information.

5.3.2 Résultats expérimentaux

Dans cette partie, nous présentons des résultats expérimentaux de notre méthode. Tout d'abord, en section 5.3.2, nous présentons un exemple détaillé et nous réalisons des analyses statistiques en observant les objets 3D partagés et reconstruit par notre méthode pour un objet 3D secret. Cette analyse est aussi réalisée sur une base de données d'objets 3D. En section 5.3.2, nous analysons la sécurité de notre méthode face à différentes attaques. En section 5.3.2, nous comparons notre méthode intégrant notre hiérarchie à accès prioritaire aux deux autres hiérarchies présentées en section 2.4.4 du chapitre 2.

Exemple détaillé

Dans cette section, nous décrivons un exemple complet de partage hiérarchique et sélectif d'un objet 3D selon notre méthode avec la hiérarchie à accès prioritaire décrite en section 5.3.1. Dans la figure 5.10, nous présentons l'objet 3D original utilisé, nommé *Lion Head*, comportant 40 000 sommets et 78.000 faces.

FIGURE 5.10 – Objet 3D original *Lion Head* (40 000 sommets et 78 000 faces).

À partir de l'objet 3D original, 15 objets 3D partagés sont générés par notre méthode avec les paramètres suivants : $L = 3$, $\mathbf{k} = (2, 3, 4)$, $\mathbf{n} = (5, 5, 5)$ et $\mathcal{D} =< 18, 3 >$.

Les objets 3D représentés dans la figure 5.11 sont générés avec notre méthode. Dans cette même figure, les objets 3D S_{00} à S_{04} représentent les objets 3D partagés avec les utilisateurs de niveau 0, les objets 3D partagés S_{10} à S_{14} avec les utilisateurs de niveau 1 et les objets 3D partagés S_{20} à S_{24} avec les utilisateurs de niveau 2. Nous observons que tous les objets 3D partagés sont visuellement très similaires. En effet, les distorsions géométriques induites par la méthode, fournissent un rendu similaire pour le SVH et cela quelque soit le niveau dans la hiérarchie de l'objet 3D partagé.

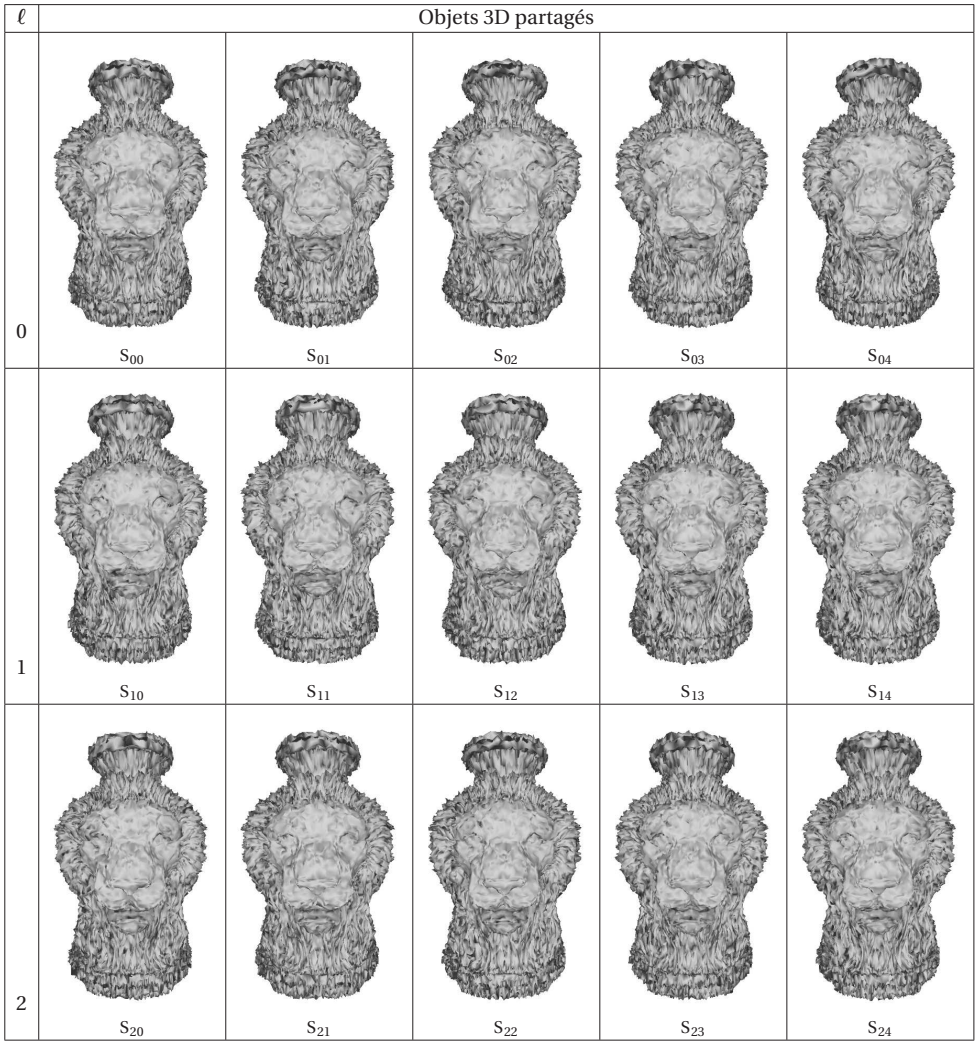


FIGURE 5.11 – Objets 3D partagés, où ℓ est le niveau de l'objet 3D partagé, générés par notre méthode avec les paramètres L = 3, $\mathbf{k} = (2,3,4)$, $\mathbf{n} = (5,5,5)$ et $\mathcal{D} =< 18,3 >$.

Pour la reconstruction de l'objet 3D secret, le processus décrit en section 5.3.1 est appliqué en sélectionnant plusieurs configurations d'objets 3D partagés de groupes de 2, 3 ou 4. La figure 5.12 illustre les résultats du processus de reconstruction en fonction des groupes d'objets 3D partagés formés. Nous notons que pour les groupes de seulement deux objets 3D partagés, il est nécessaire d'avoir au moins un objet 3D partagé appartenant au niveau 0 afin que l'objet 3D secret puisse être reconstruit. Avec trois objets 3D partagés, la reconstruction requiert la présence d'au moins un objet 3D partagé de niveau 0 ou 1. En effet, si un groupe n'est composé uniquement d'objets 3D partagés de niveau 3, alors il n'est pas possible de reconstruire l'objet 3D secret. Seul le cas où nous rassemblons au moins quatre objets 3D partagés, alors la reconstruction est toujours possible, et ce pour n'importe quel niveau d'objets 3D partagés. Avec le groupe formé des objets 3D partagés S_{10} et S_{11} (deux objets de niveau 1), comme ce dernier ne contient pas d'objets 3D partagés de niveau 0, la reconstruction de l'objet 3D secret n'est pas possible. La même chose se produit pour le groupe formé des objets 3D partagés S_{10} et S_{20} (un objet de niveau 1 et un autre de niveau 2). Enfin, le troisième groupe d'objets 3D partagés S_{20}, S_{21} et S_{22} ne permet pas la reconstruction de l'objet 3D secret, car pour cet exemple, avec seulement des objets 3D partagés de niveau 2, au moins $k_2 = 4$ utilisateurs participant à la reconstruction sont nécessaires.

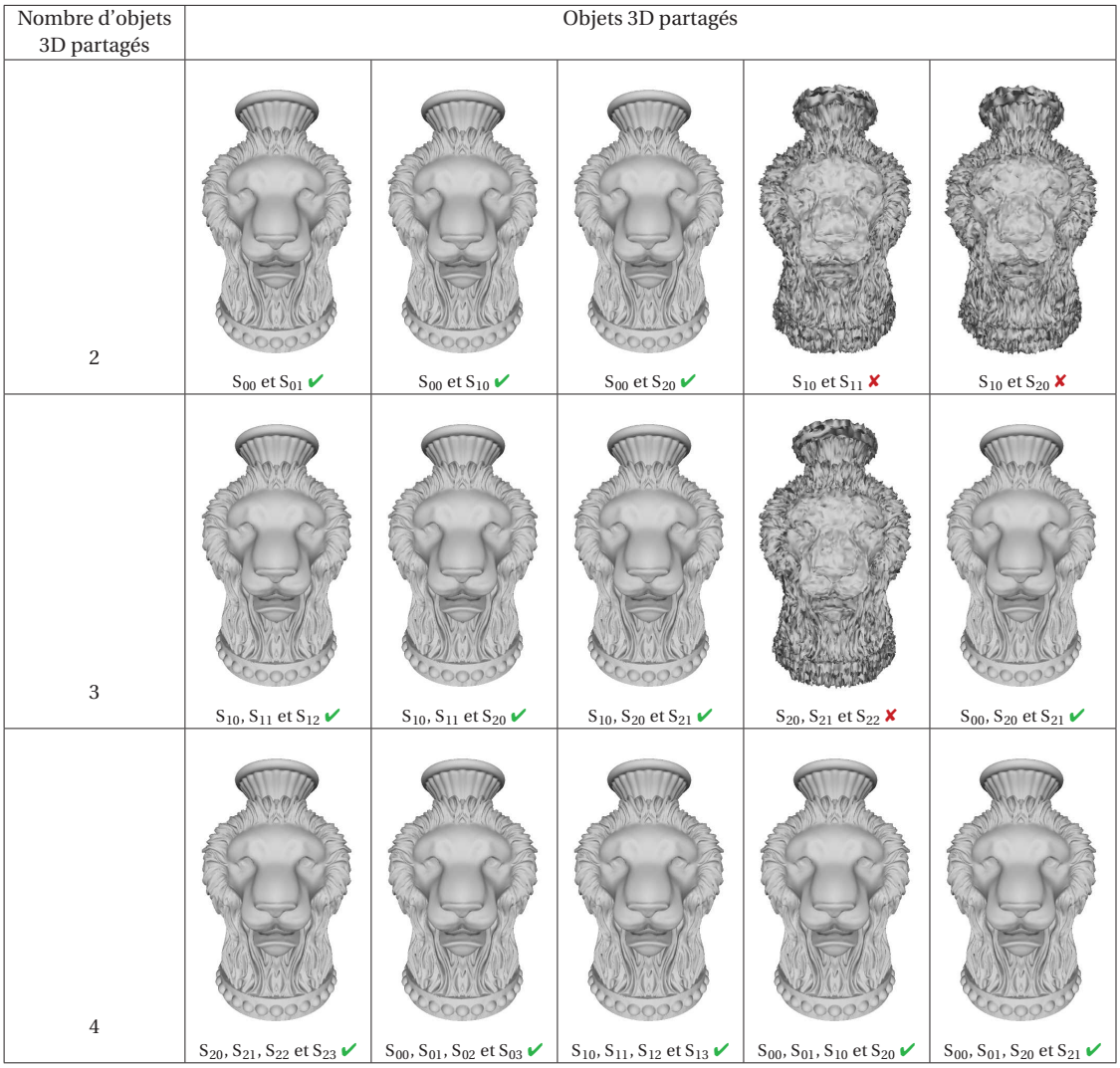


FIGURE 5.12 – Résultats de reconstruction de l'objet 3D secret à partir de 2, 3 ou 4 objets 3D partagés appartenant à des niveaux différents dans la hiérarchie à accès rapide.

Nous comparons la dégradation visuelle des objets 3D partagés générés par notre méthode à l'objet 3D secret selon les métriques HD et RMSE présentées en section 1.2.2 du chapitre 1. Le tableau 5.3 résume les résultats obtenus pour les objets 3D partagés pour chaque niveau dans la hiérarchie proposée illustrée en figure 5.11. Nous remarquons que

TABLEAU 5.3 – Résultats des métriques RMSE et HD pour les objets 3D partagés de la figure 5.11 avec le niveau de dégradation $\mathcal{D} =< 18,3 >$.

Métrique	S_{00}	S_{10}	S_{20}
RMSE (10^{-2})	3.856	3.783	3.877
HD (10^{-1})	2.226	2.469	2.461

les résultats présentés dans le tableau 5.3 sont uniformes pour les trois niveau de la hiérarchie et le niveau de dégradation fixé $\mathcal{D} =< 18,3 >$. Ainsi, les distorsions géométriques produites pour chacun des trois niveaux de la hiérarchie, fournissent des résultats similaires selon les deux métriques.

Pour valider ces mesures, nous testons sur les 400 objets 3D de la base de données *Princeton Mesh Segmentation* [33] contenant des objets 3D normalisés. Nous partageons chaque objet 3D selon différents niveaux de dégradation et calculons la moyenne et l'écart-type de chaque métrique sur tous les objets 3D partagés pour un même niveau de dégradation. La figure 5.13 présente les valeurs moyennes et écarts-types des métriques RMSE

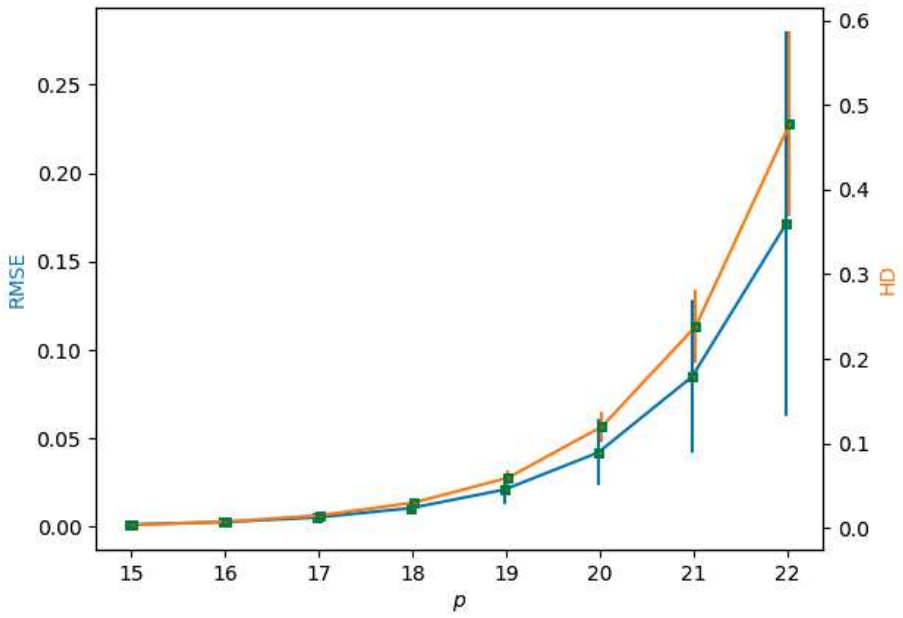


FIGURE 5.13 – Moyennes et écart-types des valeurs de RMSE et de HD pour la base de données d'objets 3D *Princeton Mesh Segmentation* [33] (400 objets 3D) selon le niveau de dégradation désiré $\mathcal{D} =< p, p+1 >$.

et HD des objets 3D partagés en fonction du niveau de dégradation $\mathcal{D}$. Notons que plus le niveau de dégradation est élevé, plus les distorsions géométriques sont présentes et fortes dans les objets 3D partagés. Lorsque p est inférieur à 18, les métriques ont des valeurs très faibles. La distance de Hausdorff révèle la même dynamique que pour le RMSE à un facteur d'échelle près. Nous pouvons en déduire que notre méthode dégrade les objets 3D

de cette base de données de manière uniforme et cela quelque soit le niveau de l'objet 3D partagé.

Analyse de la sécurité

Dans cette section, nous analysons la sensibilité de notre méthode face à des attaques en fonction du niveau de dégradation choisi. En effet, comme notre méthode modifie une séquence de bits sélectionnés dans la représentation binaire des coordonnées et plus particulièrement de la mantisse pour chaque sommet, notre méthode peut être sensible à des attaques cherchant à reconstruire le contenu protégé au lieu de la clé secrète [112]. Comme précédemment décrit, le niveau de dégradation $\mathcal{D}$ détermine directement la force des distorsions géométriques induites dans les objets 3D partagés durant le processus de partage. Un adversaire qui intercepte un de ces objets 3D partagés est laissé avec des informations en clair et des informations protégées par le partage. L'attaquant peut essayer d'améliorer son objet 3D pour reconstruire une meilleure version de l'objet 3D partagé.

Nous rappelons qu'un adversaire avec $(k_\ell - 1)$ objets 3D partagés ne peut pas reconstruire l'objet 3D secret, où ℓ est le niveau le plus élevé du groupe d'objets 3D partagés. Les informations en clair contenues dans les objets 3D sont identiques et seuls les bits sélectionnés sont différents. Ces derniers sont protégés par la méthode de partage de secret de Shamir [116] présentée en section 1.5.1 du chapitre 1. Par conséquent, l'entropie est maximale par le fait que nous utilisons des coefficients générés pseudo-aléatoirement.

En utilisant une attaque par force brute, un adversaire peut réduire les distorsions géométriques en cherchant la valeur des bits protégés de chaque sommet. Comme expliqué dans l'analyse de la sécurité pour notre méthode présentée en section 4.3 du chapitre 4, cela consiste à trouver la bonne combinaison parmi $2^{3\times r\times|\mathrm{V}|}$ possibilités, où $|\mathrm{V}|$ est le nombre de sommets. Cette attaque peut être simplifiée en cherchant itérativement à trouver les premiers bits protégés de la mantisse, ce qui revient à une recherche de la solution parmi les $2^{3\times|\mathrm{V}|}$ possibilités.

Cependant, contrairement à une attaque par brute force trop consommatrice de ressources, il est possible d'utiliser une attaque nommée attaque par *zéro-bit*, ou tous les bits protégés sont assignés à zéro. Beaucoup plus rapide à mettre en place, cette attaque rend possible la reconstruction d'objets 3D de meilleure qualité en fonction de niveau de dégradation. Notons à nouveau que notre méthode sélective préserve naturellement une partie de la géométrie de l'objet 3D, ce qui conserve certaines propriétés sur le contenu de l'objet 3D secret. Par conséquent, des approches par traitement d'objet 3D peuvent modifier la forme de l'objet 3D, par exemple des algorithmes de filtrage. Un adversaire peut attaquer des objets 3D partagés interceptés en essayant d'en améliorer la qualité avec de tels traitements comme les lissages laplaciens [58] ou de Taubin [127] comme illustré en figure 5.14.

La figure 5.14 illustre ces deux types d'attaques possibles (cf. figures 5.14.a,e) sur deux objets 3D partagés selon deux niveaux de dégradation $\mathcal{D} =< 18,8 >$ et $\mathcal{D} =< 22,8 >$. Nous remarquons que dans la figure 5.14.b pour $\mathcal{D} =< 18,8 >$ que l'attaque par *zéro-bit* nous permet de récupérer une version similaire de l'objet 3D secret, mais qui reste néanmoins visuellement éloignée de l'objet 3D secret original. De plus, l'objet 3D en figure 5.14.b contient des artéfacts de discrétisation comme des auto-intersections, des triangles en chevauchement et ne possèdent pas de caractéristiques de haute qualité pour ce niveau de dégradation. Quand nous utilisons une attaque par traitement avec un lissage laplacien sur l'objet 3D partagé illustré en figure 5.14.c, nous observons que le traitement donne une surface de meilleure qualité comparée à celle de l'objet 3D partagé. Cepen-

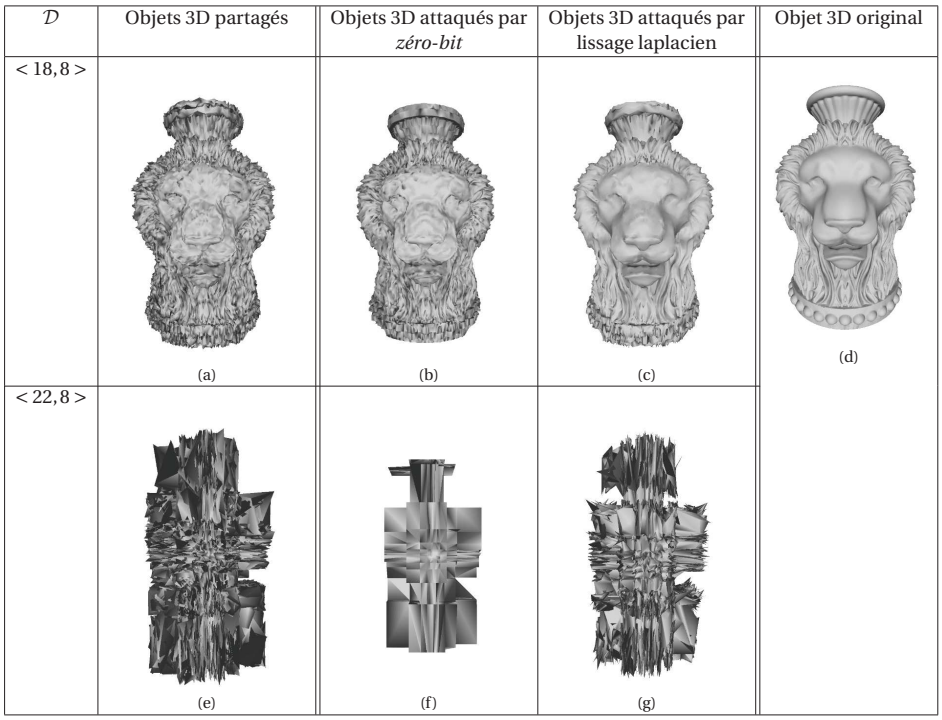


FIGURE 5.14 – Différentes attaques sur des objets 3D partagés en fonction du niveau de dégradation $\mathcal{D}$.

dant les détails de haute qualité restent protégés.

Avec une plus grande valeur de niveau de dégradation, ces attaques ne permettent pas à un utilisateur de récupérer un objet 3D de meilleure qualité, au contraire elles augmentent les distorsions géométriques. Par exemple, pour l'attaque par *zéro-bit* sur l'objet 3D partagé avec le niveau de dégradation $\mathcal{D} =< 22,8 >$ comme illustré en figure 5.14.e, l'objet 3D récupéré après l'attaque illustrée en figure 5.14.f, révèle absolument rien de l'objet 3D secret. La confidentialité visuelle de l'objet 3D secret est assurée et les objets 3D partagés à ce niveau sont beaucoup plus résistants à ces attaques. De même, pour l'attaque par lissage laplacien illustrée en figure 5.14.g, nous remarquons que l'objet 3D obtenu après l'attaque ne fournit aucune information sur le contenu secret. Ainsi, comme pour les attaques par *zéro-bit*, les attaques par traitement ne permettent pas de récupérer un objet 3D de qualité suffisante pour un certain intervalle du niveau de dégradation. Les approches par reconstruction donnent des résultats moins efficaces que celles par lissage, car les normales sont déterminées en fonction de l'orientation des faces et des sommets voisins. Cela rend ainsi l'utilisation des méthodes comme l'algorithme des *Marching Cubes* [93] ou la méthode de Poisson [77] inutiles.

Le tableau 5.4 résume les résultats des différentes métriques utilisées pour l'analyse des objets 3D partagés attaqués selon les deux types d'attaques précédentes. Nous remarquons que les résultats des métriques pour les objets 3D attaqués sont plus faibles que pour les objets 3D partagés, mais visuellement les résultats varient grandement en fonction du niveau de dégradation, comme illustré en figure 5.14.

Comparaison avec deux autres hiérarchies

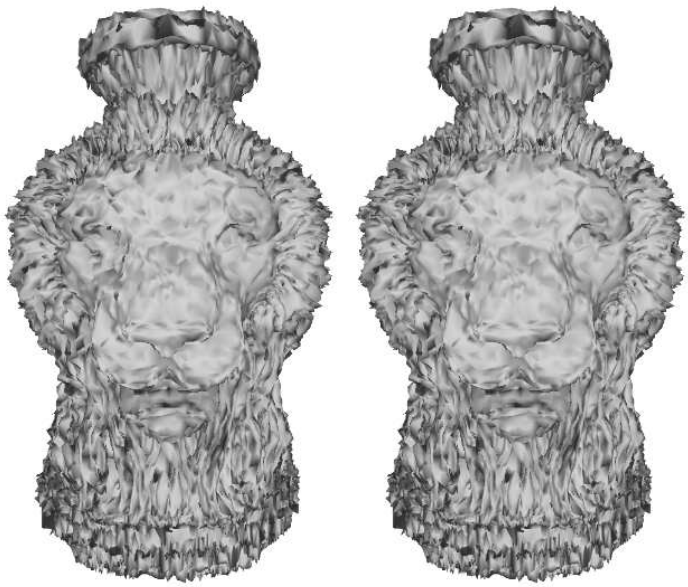
Dans cette section, nous substituons dans notre méthode, la hiérarchie à accès prioritaire PAH par deux autres hiérarchies proposées par Tassa [126] et par Belenkiy [7] précédemment décrites dans la section 2.4.4 du chapitre 2. Pour réaliser cette comparaison,

TABLEAU 5.4 – Résultats des métriques pour les objets 3D partagés et les objets 3D attaqués par *zéro-bit* ou par lissage laplacien.

Métriques	$\mathcal{D}$	Objets 3D partagés	Objets 3D attaqués par *zéro-bit*	Objets 3D lissés
RMSE ($10^{-}2$)	$< 18,8 >$	3.843	2.865	2.404
	$< 22,8 >$	85.07	62.34	81.57
Hausdorff ($10^{-}1$)	$< 18,8 >$	2.264	2.348	2.034
	$< 22,8 >$	43.64	16.83	39.61

nous remplaçons les conditions de partage selon la hiérarchie à accès prioritaire par les conditions des deux autres hiérarchies. Ainsi, la valeur du mot binaire W_i extrait pour chaque sommet est assignée à un coefficient spécifique du polynôme de partage. Pour la hiérarchie de Tassa [126], c'est le coefficient a_0 qui est assigné à la valeur de W_i. Tandis que pour la hiérarchie de Belenkiy [7], le coefficient $a_{k_{L-1}}$ est assigné à la valeur de W_i. Dans les deux cas, aucune clé secrète n'est distribuée aux utilisateurs.

Pour le processus de reconstruction de l'objet 3D secret, la principale différence est le paramètre **k**, correspondant au vecteur de nombres représentant le nombre minimum d'utilisateurs requis par niveau. Pour l'approche de Tassa [126], il est nécessaire d'appliquer une interpolation afin de résoudre le système linéaire formé de k_{L-1} inconnues comme pour l'équation (2.12). La valeur du coefficient a_0 est récupérée comme mot binaire W_i.



(a) Hiérarchie de Tassa (b) Hiérarchie de Belenkiy

FIGURE 5.15 – Objets 3D partagés de niveau 1 par notre méthode avec la hiérarchie de Tassa [126] et la hiérarchie de Belenkiy [7] avec les paramètres $L = 3$, $\mathbf{k} = (2,3,4)$, $\mathbf{n} = (5,5,5)$ et $\mathcal{D} =< 18,3 >$.

La figure 5.15.a illustre un exemple d'un objet 3D partagé de niveau 1 qui a été distribué aux utilisateurs selon la hiérarchie de Tassa avec les mêmes paramètres utilisés pour l'expérimentation avec la PAH : $L = 3$, $\mathbf{k} = (2,3,4)$, $\mathbf{n} = (5,5,5)$ et $\mathcal{D} =< 18,3 >$. Comme pour la hiérarchie proposée, nous observons que les objets 3D partagés, sans regard à leur ni-

veau dans la hiérarchie, ont le même niveau de distorsions géométriques induites par le processus de partage.

Nombre d'objets 3D partagés	Objets 3D partagés				
2	S_{00} et S_{01} ✘	S_{00} et S_{10} ✘	S_{00} et S_{20} ✘	S_{10} et S_{11} ✘	S_{10} et S_{20} ✘
3	S_{10}, S_{11} et S_{12} ✘	S_{10}, S_{11} et S_{20} ✘	S_{10}, S_{20} et S_{21} ✘	S_{20}, S_{21} et S_{22} ✘	S_{00}, S_{20} et S_{21} ✘
4	S_{20}, S_{21}, S_{22} et S_{23} ✘	S_{00}, S_{01}, S_{02} et S_{03} ✔	S_{10}, S_{11}, S_{12} et S_{13} ✘	S_{00}, S_{01}, S_{10} et S_{20} ✔	S_{00}, S_{01}, S_{20} et S_{21} ✘

FIGURE 5.16 – Reconstruction de l'objet 3D secret utilisant différents groupes de k_ℓ utilisateurs pour la hiérarchie de Tassa [126].

Reconstruire l'objet 3D secret est seulement possible lorsque les conditions de seuil sont atteintes, comme dans la figure 5.16. Dans ce cas, l'objet 3D secret peut être reconstruit quand au moins deux utilisateurs de niveau 0, trois utilisateurs de niveau 1 et quatre utilisateurs de niveau 2 participent à la reconstruction. Comme il s'agit d'une méthode de partage hiérarchie multi-niveaux, les utilisateurs de niveaux supérieurs peuvent être utilisés pour atteindre les seuils des niveaux inférieurs. Ainsi, un utilisateur de niveau 0 compte comme un utilisateur de niveaux 1 et 2, tandis qu'un utilisateur de niveau 1 compte aussi comme un utilisateur de niveau 2. Les conditions de reconstruction de la hiérarchie de Tassa sont donc plus contraignantes que la hiérarchie proposée.

Pour la hiérarchie de Belenkiy, une interpolation de Hermite-Birkhoff est aussi nécessaire, mais dans ce cas le système linéaire à résoudre peut être plus petit en taille en fonction des utilisateurs du groupe de reconstruction. La figure 5.15.b présente un exemple d'objet 3D partagé de niveau 1 selon la hiérarchie de Belenkiy avec les mêmes paramètres utilisés pour la hiérarchie proposée : $L = 3$, $\mathbf{k} = (2,3,4)$, $\mathbf{n} = (5,5,5)$ et $\mathcal{D} =< 18,3 >$. Comme pour la hiérarchie proposée et la hiérarchie de Tassa, notons que les distorsions géométriques apparues à cause du processus de partage sont visuellement similaires pour tous

les objets 3D générés. Comme décrit précédemment en section 2.4.4 du chapitre 2, la valeur du coefficient $a_{(k_{L-1}-1)}$ correspond à celle du mot binaire W_i. Une fois le mot binaire W_i extrait, nous réalisons le même processus de substitution pour la reconstruction de chaque sommet de l'objet 3D secret comme présenté en section 5.3.1. Le tableau 5.17 présente les combinaisons possibles de groupes de reconstruction de l'objet 3D secret à partir d'objets 3D partagés selon la hiérarchie de Belenkiy. Dès que k_ℓ utilisateurs du même niveau ℓ sont présents, alors la reconstruction de l'objet 3D secret devient possible. Sinon les utilisateurs de niveaux supérieurs participent à la reconstruction au niveau $(\ell+1)$. Dans cet exemple, la reconstruction devient accessible, par exemple, quand deux utilisateurs de niveau 0 participent à la reconstruction. Les conditions de reconstruction de la hiérarchie de Belenkiy sont donc plus contraignantes que celle de la hiérarchie proposée.

Nombre d'objets 3D partagés	Objets 3D partagés				
2	 S_{00} et S_{01} ✔	 S_{00} et S_{10} ✘	 S_{00} et S_{20} ✘	 S_{10} et S_{11} ✘	 S_{10} et S_{20} ✘
3	 S_{10}, S_{11} et S_{12} ✔	 S_{10}, S_{11} et S_{20} ✘	 S_{10}, S_{20} et S_{21} ✘	 S_{20}, S_{21} et S_{22} ✘	 S_{00}, S_{20} et S_{21} ✘
4	 S_{20}, S_{21}, S_{22} et S_{23} ✔	 S_{00}, S_{01}, S_{02} et S_{03} ✔	 S_{10}, S_{11}, S_{12} et S_{13} ✔	 S_{00}, S_{01}, S_{10} et S_{20} ✔	 S_{00}, S_{01}, S_{20} et S_{21} ✔

FIGURE 5.17 – Reconstruction de l'objet 3D secret utilisant différents groupes de k_ℓ utilisateurs pour la hiérarchie de Belenkiy [7].

Le tableau 5.5 résume les résultats des métriques sur les objets 3D partagés en fonction de la hiérarchie utilisée, où la figure 5.15 illustre des exemples d'objets 3D partagés. Nous remarquons que pour le même niveau de dégradation $\mathcal{D}$, les résultats entre les différentes hiérarchies sont très similaires. Cela nous assure que ces trois hiérarchies maintiennent le même niveau de sécurité pour le même niveau de dégradation. Notons que le nombre de combinaisons permettant la reconstruction de l'objet 3D secret est plus élevé pour notre PAH que pour les autres. Notre hiérarchie à accès prioritaire permet de

construire plus aisément que la hiérarchie de Tassa ou celle de Belenkiy pour reconstruire l'objet 3D secret.

TABLEAU 5.5 – Résultats des métriques RMSE et HD pour les objets 3D partagés selon les différentes hiérarchies pour le niveau de dégradation $\mathcal{D} =< 18,3 >$.

		Objets 3D partagés		
Métriques	Hiérarchie	S_{00}	S_{10}	S_{20}
RMSE (10^{-2})	Tassa	3.682	3.798	3.684
	Belenkiy	3.795	3.843	3.821
	PAH	3.856	3.785	3.877
HD (10^{-1})	Tassa	2.583	2.357	2.201
	Belenkiy	2.502	2.457	2.621
	PAH	2.226	2.469	2.461

5.3.3 Conclusion

Dans cette partie, nous avons proposé une méthode partage hiérarchique et sélectif d'objet 3D secret préservant le format, offrant une hiérarchie à accès prioritaire et protégeant visuellement l'objet 3D. L'approche proposée distribue des objets 3D partagés à $|\mathbf{n}|$ utilisateurs possédant des droits d'accès différents à l'objet 3D secret selon la hiérarchie à accès prioritaire. Cette hiérarchie permet de reconstruire l'objet 3D secret avec moins d'utilisateurs requis selon le niveau le plus élevé des utilisateurs participant à la reconstruction. L'objet 3D secret peut être reconstruit sans perte en rassemblant un nombre spécifique d'objets 3D partagés noté k_ℓ selon le niveau ℓ qui est le plus élevé parmi les utilisateurs participant à la reconstruction. Notre méthode permet aux utilisateurs de choisir le niveau de dégradation pour les objets 3D partagés pour visuellement protéger le contenu de l'objet 3D secret. Nous avons analysé les résultats obtenus avec notre méthode et nous avons montré les avantages de notre hiérarchie à accès prioritaire pour reconstruire plus rapidement l'objet 3D secret par rapport à deux hiérarchies de l'état de l'art dans un contexte de partage hiérarchie d'objet 3D.

5.4 Conclusion

Dans ce chapitre, nous avons présenté des nouvelles méthodes de partage hiérarchique de secret multimédia et en particulier une méthode de partage hiérarchique des régions d'intérêts secrètes au sein d'une image issue des réseaux sociaux, ainsi qu'une méthode de partage hiérarchique et sélectif d'objet 3D secret offrant une hiérarchie à accès prioritaire.

Avec l'utilisation des aspects hiérarchiques présentés en section 2.4 du chapitre 2, nous avons pu construire un nouveau protocole pour la gestion des conflits pluripartites de vie privée. Ainsi, grâce à notre méthode de partage hiérarchique des régions des régions d'intérêts secrètes au sein d'une image, nous proposons un outil permettant la diffusion d'une photographie représentant plusieurs personnes sans que leur identité soit dévoilée sans leur accord. À l'aide d'un algorithme de détection et de reconnaissance des visages, des individus sont identifiés. Préalablement, les utilisateurs définissent un seuil de confiance k déterminant le nombre minimum de personnes nécessaires pour que l'image protégée soit entièrement visible sur le réseau social. Si ce seuil n'est pas franchi,

alors seules les personnes ayant donné leur accord seront affichées sur le réseau social. Suite à la détection, notre méthode récupère des régions d'intérêt délimitées pour chaque personne identifiée qui sont ensuite partagées à l'aide de la méthode de Belenkiy [7] afin de protéger les pixels de ces régions d'intérêt. Chaque utilisateur reçoit une *share* permettant de reconstruire en priorité la région d'intérêt correspondant à son visage. Les résultats expérimentaux démontrent la faisabilité de notre méthode et sa possible application au sein des réseaux sociaux comme outil de gestion des conflits pluripartites de vie privée. Pour l'instant, les régions d'intérêt correspondent à une boîte englobant la zone correspondant au visage. Dans des travaux futurs, nous pourrions nous intéresser à sélectionner seulement les pixels correspondant au visage de la personne détectée, ou bien alors améliorer le rendu en appliquant un chiffrement sélectif au niveau des pixels des régions d'intérêt au lieu d'avoir un aspect semblable à du bruit. Ainsi, les régions d'intérêt pourraient être protégées selon des niveaux de confidentialité différents tels que du chiffrement transparent ou bien du chiffrement suffisant selon les besoins des utilisateurs.

Dans notre seconde contribution, nous avons étendu les fonctionnalités de notre méthode proposée en section 4.3 du chapitre 4 en intégrant des aspects hiérarchiques sans pour autant perdre l'approche de chiffrement sélectif. Ainsi, chaque utilisateur reçoit un objet 3D partagé avec un niveau particulier dans la hiérarchie et selon un niveau de dégradation choisi. Les objets 3D partagés correspondent à l'objet 3D secret déformé géométriquement selon le niveau de dégradation choisi. Grâce à la hiérarchie à accès prioritaire, les utilisateurs de niveaux supérieurs peuvent reconstruire l'objet 3D secret en haute qualité avec n'importe quel utilisateur de n'importe quel niveau. Pour se faire, ils leur faut rassembler k_ℓ autres objets 3D partagés de n'importe quel autre niveau. Du moment où il existe un utilisateur de niveau ℓ dans le groupe de reconstruction, alors ce groupe doit au moins contenir k_ℓ autres utilisateurs pour réussir la reconstruction. Dans nos expérimentations, nous avons évalué le rendu visuel des objets 3D ainsi que le résultat de certaines métriques avec référence et nous avons comparé notre méthode et notre méthode avec d'autres hiérarchies de l'état de l'art. Afin d'accélérer la reconstruction de l'objet 3D secret selon la présence de certains utilisateurs, nous avons distribué des clés secrètes aux utilisateurs en fonction de leur niveau dans la hiérarchie. Ces clés permettent une reconstruction plus prioritaire, car elles initialisent des générateurs de nombres pseudo-aléatoires utilisés pour construire les polynômes de partage pour chaque sommet. Ainsi, les utilisateurs de niveaux supérieurs possèdent un ensemble de clés pour réaliser cette accélération. Seule une de ces clés n'est pas distribuée afin d'assurer la sécurité de la méthode. Dans des travaux futurs, nous souhaitons nous intéresser à ce que les utilisateurs ne reçoivent qu'une seule et unique clé pour simplifier la méthode. Pour cela, nous avons besoin d'avoir une relation entre les clés où dans le cas un utilisateur possède la première clé K_0, il peut déduire la clé K_1, K_2, …, $\mathrm{K}_{\mathrm{L}-1}$, mais lorsqu'il possède la clé K_i, il ne peut que déduire les clés suivantes K_{i+1}, K_{i+2}, jusqu'à $\mathrm{K}_{\mathrm{L}-1}$. Un des moyens pour réaliser une telle approche est d'utiliser des fonctions de dérivation de clé (ou *key derivation function*) comme la fonction *Password-Based Key Derivation Function 2* (PBKDF2) [76]. De cette manière, à partir d'une seule clé secrète, il est alors possible d'obtenir une suite de clés dérivées. Par extension, en distribuant la première clé utilisée nous pouvons permettre à ce dernier de reconstruire l'objet 3D secret seul. Ainsi, nous pourrions proposer une méthode hybride de chiffrement sélectif 3D et de partage hiérarchique d'objet 3D secret.

Ces travaux ont fait l'objet de trois publications dont une soumise. Tout d'abord, la méthode de partage de régions d'intérêts secrètes au sein d'une image issue des réseaux sociaux a fait l'objet d'une publication dans la conférence nationale CORESA en 2018 [14] puis dans la conférence internationale IEEE ICIP en 2019 [17]. Enfin, la méthode de par-

tage hiérarchique et sélectif d'objet 3D a fait l'objet d'une soumission auprès du journal international IEEE Transactions on Information Forensics and Security en 2019 [19].

Chapitre 6

Analyse de la confidentialité des objets 3D

Sommaire

6.1 Introduction

Comme nous l'avons indiqué dans l'état de l'art sur la sécurité multimédia du chapitre 1, les objets 3D peuvent être protégés soit par insertion de données cachées soit par chiffrement sélectif.

Pour l'insertion de données cachées, il existe un domaine de recherche important qui consiste à détecter la présence de messages cachés au sein des supports, à savoir la stéganalyse [79, 140]. Depuis quelques années, la stéganalyse s'intéresse à détecter la présence de messages cachés au sein des objets 3D [87–89, 158]. Ainsi, il est normal de voir apparaître de nouvelles méthodes d'insertion essayant d'être imperceptibles face aux dernières approches de stéganalyse [159]. Dans cette partie, nous présentons notre première contribution réalisée en collaboration avec l'université de York dans le cadre d'un séjour de Zhenyu Li au LIRMM où, à partir de la méthode proposée par Itier et Puech [64], nous avons proposé de rendre les messages cachés les plus imperceptibles possible.

Pour le chiffrement sélectif d'objets 3D comme nous l'avons présenté par notre méthode de chiffrement sélectif 3D en section 1.4 du chapitre 3, ou par nos méthodes de partage sélectif d'objet 3D secret des chapitres 4 et 5, nous avons pu observer que les métriques dites objectives permettaient de détecter la présence de distorsions géométriques au sein de l'objet 3D. Cependant, ces métriques se révèlent incapables de définir les paramètres de dégradation nécessaires à l'obtention d'un chiffrement transparent, suffisant ou confidentiel. De plus, en fonction des caractéristiques de l'objet 3D, telles que la forme ou la densité de points, les paramètres utilisés lors du chiffrement sélectif ne donnent pas forcément les mêmes résultats visuels. De ce fait, les métriques objectives ne reflètent ni l'état de l'objet 3D ni la présence de distorsions géométriques sauf lorsque ces dernières sont extrêmes. Dans cette partie, nous souhaitons étudier les objets 3D chiffrés sélectivement en se basant sur des évaluations subjectives, non pas de la qualité, mais de la confidentialité des objets 3D chiffrés. Ainsi, nous proposons une base de données d'objets 3D sélectivement chiffrés et des résultats suite à des évaluations subjectives dans le but de construire une métrique de confidentialité.

En section 6.2, nous présentons notre contribution sur l'amélioration des méthodes d'insertion de données cachées face à la stéganalyse 3D. Dans la section 6.3, nous présentons notre base de données d'objets 3D sélectivement chiffrés dans le but de réaliser des évaluations subjectives de la confidentialité et nous présentons les résultats de ces dernières. En section 6.4, nous concluons nos travaux sur des pistes futures dans le domaine de la stéganalyse 3D et de l'évaluation subjective de la confidentialité des objets 3D.

6.2 Résistance à la stéganalyse des méthodes d'insertion de données cachées hautes capacités

L'insertion de données cachées 3D (IDC) est utilisée, comme présenté dans le chapitre 1, dans le but d'insérer ou cacher des informations au sein des objets 3D sans provoquer de modifications visibles et détectables statistiquement.

En section 6.2.1, nous présentons un état de l'art sur la stéganalyse, l'art de détecter la présence de messages cachés dans le domaine 3D. Nous détaillons en section 6.2.2 la méthode proposée par Itier et Puech [64] et des améliorations proposées pour augmenter la résistance de la méthode à la stéganalyse. Enfin, en section 6.2.4 nous concluons sur ces travaux réalisés en collaboration avec l'université de York.

6.2.1 Stéganalyse 3D

La première analyse d'objets 3D marqués a été proposée en 2014 par Yang et Ivrissimtzis [158]. Ces derniers proposent d'analyser les caractéristiques (*features*) des objets 3D pour entraîner un classifieur à distinguer les objets 3D sans marque de ceux marqués. Pour cela, Yang et Ivrissimtzis ont proposé d'extraire des caractéristiques à partir des coordonnées cartésiennes et laplaciennes, des angles dièdres et normales aux triangles de l'objet 3D testé et normalisé avec le même objet 3D ayant été lissé par un filtre laplacien. Li et Bors [88] ont amélioré cette approche en utilisant un ensemble de caractéristiques locales. Ces deux approches utilisent des statistiques de caractéristiques locales 3D comme descripteurs en entrée d'algorithmes et de méthodes basés sur de l'apprentissage supervisé afin d'entraîner le classifieur détectant la présence de messages cachés ou non au sein d'objets 3D. Le principal problème en stéganalyse est de trouver les caractéristiques statistiques permettant de détecter les distorsions engendrées par l'IDC.

Li et Bors ont proposé un vecteur de 52 caractéristiques [88]. Avant l'extraction de ces caractéristiques, un filtrage laplacien est appliqué à l'objet 3D donnant un objet étalonné pour le système de stéganalyse. Les caractéristiques sont ensuite extraites de l'objet 3D original ainsi que de l'objet 3D lissé. Les quatre premiers moments, à savoir la moyenne, la variance, le coefficient d'asymétrie (*skewness*) et le *kurtosis* sont calculés pour l'objet 3D original ainsi que pour l'objet 3D filtré. Leurs différences sont alors utilisées comme données d'entrée au système de stéganalyse. L'IDC modifiant légèrement la surface de l'objet 3D, les moments statistiques des caractéristiques 3D locales sont adaptés pour la stéganalyse. Les caractéristiques 3D utilisées pour la stéganalyse dans [158] incluent la position et la norme des sommets dans les systèmes de coordonnées cartésiennes ou laplaciennes [157], les normales aux faces, les angles dièdres entre des triangles voisins, les normales aux sommets, la courbure gaussienne et le taux de courbure.

6.2.2 Amélioration de la résistance à la stéganalyse

Dans cette section, nous cherchons à améliorer la résistance à la stéganalyse de la méthode d'IDC haute capacité proposée par Itier et Puech [64], qui est basée sur une quantification du chemin hamiltonien (ou *Hamiltonian Path Quantization*). Pour réaliser cela, nous étudions trois aspects de la méthode, à savoir le paramètre d'intervalle Δ, une sélection différente des sous-intervalles s et une différente manière d'insérer à l'aide du système de coordonnées sphériques (SCS) avec les coordonnées (r, θ, ϕ), où r est la coordonnée radiale, θ et ϕ sont les coordonnées angulaires. Nous analysons le déplacement de chaque sommet v_i de l'objet 3D. Comme l'insertion est réalisée dans le SCS, nous considérons le déplacement du sommet selon la coordonnée radiale. Nous supposons que les sous-intervalles sont distribués de manière uniforme et que le déplacement du sommet dans la coordonnée radiale est défini par :

$$D_p = \sum_{j=1}^{s} \sum_{k=1}^{s} P_j Q_k |j-k| \frac{\Delta}{s}, \tag{6.1}$$

où P_j est la probabilité que le sommet v_i soit positionné dans le $j^{\text{ème}}$ sous-intervalle et $\frac{\Delta}{s}$ est la longueur de chaque sous-intervalle. Nous supposons que le sommet v_i se trouve dans le $k^{\text{ème}}$ sous-intervalle tel que $P_j = \frac{1}{s}$, $j = 1, 2, \ldots, s$. Si les symboles dans le message sont uniformément distribués, alors la coordonnée du sommet modifié v_i se trouve aléatoirement dans l'intervalle et $Q_k = \frac{1}{s}$, $k = 1, 2, \ldots, s$.

insi, l'équation (6.1) se simplifie telle que :

$$D_p = \frac{\Delta}{3}(1 - \frac{1}{s^2}). \tag{6.2}$$

Les équations calculant les déplacements du sommet selon les deux coordonnées angulaires du SCS sont similaires à l'équation (6.2). À partir de l'équation (6.2), il est alors possible d'en déduire que si le paramètre d'intervalle Δ est plus faible, alors le déplacement du sommet est tout autant réduit. En même temps, quand le nombre de sous-intervalles s augmente, ce qui accroît la capacité d'insertion, alors le déplacement du sommet augmentera également. Cependant, l'influence du nombre de sous-intervalles sur le déplacement est très faible, car $\frac{dD_p}{ds} = \frac{2\Delta}{3s^3}$ est très petit quand s est large.

À partir de cette analyse, il est intéressant de réduire le paramètre d'intervalle Δ le plus possible afin de limiter les distorsions géométriques sur les caractéristiques 3D utilisées par le système de stéganalyse. Nous analysons également l'influence de ces changements sur les caractéristiques utilisées pour la stéganalyse 3D.

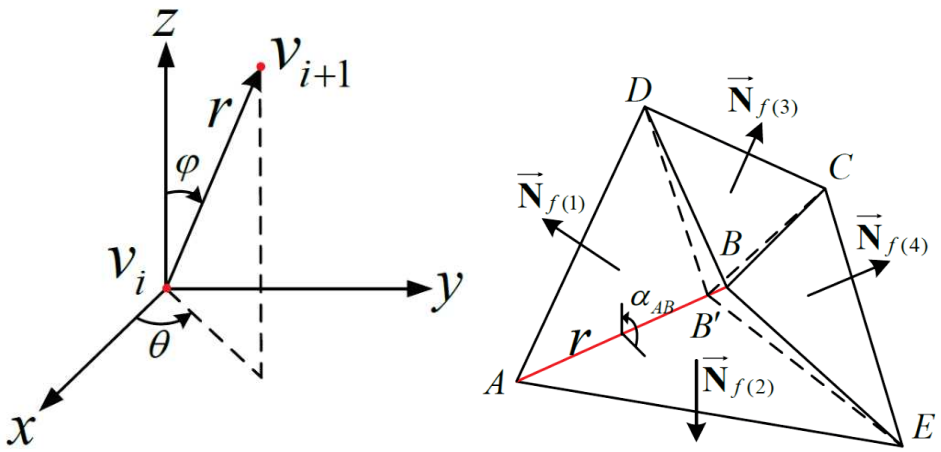


FIGURE 6.1 – Représentation des modifications liées à l'insertion dans le SCS : a) Exemple d'insertion sur le vecteur d'arête v_i et v_{i+1} selon le SCS, b) Les modifications engendrées par le HPQ uniquement selon la coordonnée radiale de l'arête d'insertion.

Comme illustré en figure 6.1.a, la version originale de l'IDC haute capacité par HPQ proposée par Itier et Puech [64] insère le message secret dans les trois coordonnées sphériques (r, θ, ϕ) calculées à partir de chaque paire de sommets successifs formant le chemin hamiltonien. Cependant, les modifications sur les deux coordonnées angulaires θ et ϕ peuvent influencer de manière plus significative les caractéristiques utilisées lors de la stéganalyse que la coordonnée radiale r. La figure 6.1.b illustre les modifications produites par l'insertion en utilisant seulement la coordonnée radiale pour insérer un octet du message sur l'arête de vecteur $\vec{AB}$ entre les sommets A et B. La nouvelle position du sommet B, B' est toujours dans la même direction que le vecteur $\vec{AB}$. Les normales aux faces $\triangle ABD$ et $\triangle ABE$, notées $\vec{N}_{f(1)}$ et $\vec{N}_{f(2)}$, ne sont pas affectées par le déplacement du sommet B. L'angle dièdre entre les faces $\triangle ABD$ et $\triangle ABE$ illustrées en figure 6.1, est égal à :

$$\alpha_{AB} = arccos \frac{\vec{N}_{f(1)} \cdot \vec{N}_{f(2)}}{|\vec{N}_{f(1)}||\vec{N}_{f(2)}|}. \tag{6.3}$$

Notons que l'angle dièdre α_{AB} n'est pas influencé par la modification du sommet B. Cependant, le déplacement du sommet B change la direction des normales aux faces

$\triangle CDB$ et $\triangle CBE$ notées $\vec{N}_{f(3)}$ et $\vec{N}_{f(4)}$, sauf quand toutes les faces sont sur le même plan. En pratique, de nombreux objets 3D générés par CAO sont constitués localement de zones plates. De plus, pour les objets 3D ayant une résolution élevée, les régions lisses sont localement plates. Dans ces cas, les arêtes d'insertion souffrent de légères modifications se propageant depuis le déplacement de l'arête d'insertion précédente.

Dans le cas contraire, où la méthode insère uniquement sur n'importe laquelle des coordonnées angulaires du vecteur $\vec{AB}$, les normales aux faces et les angles dièdres tels que $\vec{N}_{f(1)}$, $\vec{N}_{f(2)}$ et α_{AB} sont tous directement affectés. Si l'insertion est appliquée en utilisant les trois coordonnées du SCS, comme proposé en [64], l'espace possible des déplacements des sommets est bien plus large. Cela résulte en des distorsions significatives pour les caractéristiques 3D utilisées pour la stéganalyse et de ce fait améliore la détection des objets 3D marqués.

Selon cette analyse, la résistance à la stéganalyse de la méthode d'IDC 3D est améliorée seulement lorsque l'insertion est réalisée sur la coordonnée radiale de l'arête d'insertion dans le SCS. Cependant,les arêtes pour l'insertion sont choisies lors de la construction du chemin hamiltonien, il est probable que certaines d'entre elles n'existent pas dans le maillage original. Néanmoins, les arêtes du chemin hamiltonien recouvrent fortement les arêtes du maillage original. Ainsi, les distorsions générées par l'insertion sont moins significatives.

6.2.3 Résultats expérimentaux

Dans cette section, nous testons notre amélioration de la résistance à la stéganalyse pour la méthode proposée par Itier et Puech [64]. Dans nos expérimentations, nous utilisons 354 objets 3D hôtes provenant de la base de données du projet *Princeton Mesh Segmentation* [33]. Nous utilisons différentes valeurs pour les paramètres de la méthode d'IDC [64], où nous insérons les informations dans différentes coordonnées du SCS pour obtenir des objets 3D marqués. Le système de stéganalyse utilise l'ensemble de caractéristiques LFS52 (*52-dimensional Local Feature Set*) proposé par Li et Bors [88]. Pour chaque paramètre de l'algorithme d'insertion, un système de stéganalyse est entraîné sur plus de 260 paires d'objets 3D hôtes et d'objets 3D marqués correspondants, puis celui-ci est alors testé sur 94 paires de test. Nous utilisons l'ensemble de *Fisher Linear Discriminate* (FLD) [159] pour entraîner le système de stéganalyse, qui est la méthode la plus utilisée dans le domaine de la stéganalyse.

Afin d'observer l'influence du paramètre d'intervalle Δ sur la résistance à la stéganalyse de la méthode d'insertion [64], nous fixons les valeurs de Δ à $\{10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}\}$. Les domaines d'insertion sont les trois coordonnées du SCS. Le taux de charge utile est de 24 bits par sommet (bps), consistant en 8 bits par coordonnée du SCS, ce qui signifie que le nombre de sous-intervalles est de $s = 2^8$.

En figure 6.2, nous présentons les résultats finaux des taux d'erreurs à la détection des objets 3D marqués selon la méthode de Itier et Puech [64] en fonction du paramètre d'intervalle Δ. Nous remarquons qu'une petite valeur de Δ tend à augmenter le taux d'erreur de détection, cela signifie une résistance plus grande à la stéganalyse. Dans le cas de $\Delta = 10^{-4}$, l'intervalle devient trop large pour trouver suffisamment de positions de sommet acceptable pour insérer, ainsi la charge utile est inférieure à 24 bps, ce qui explique aussi pourquoi le taux d'erreur de détection augmente.

Ainsi, dans le tableau 6.1, nous résumons les résultats des erreurs de détection en fonction des domaines d'insertion, où HPQ représente la méthode originale [64] insérant sur les trois coordonnées du SCS avec une charge utile de 24 bps, HPQ-PA représente

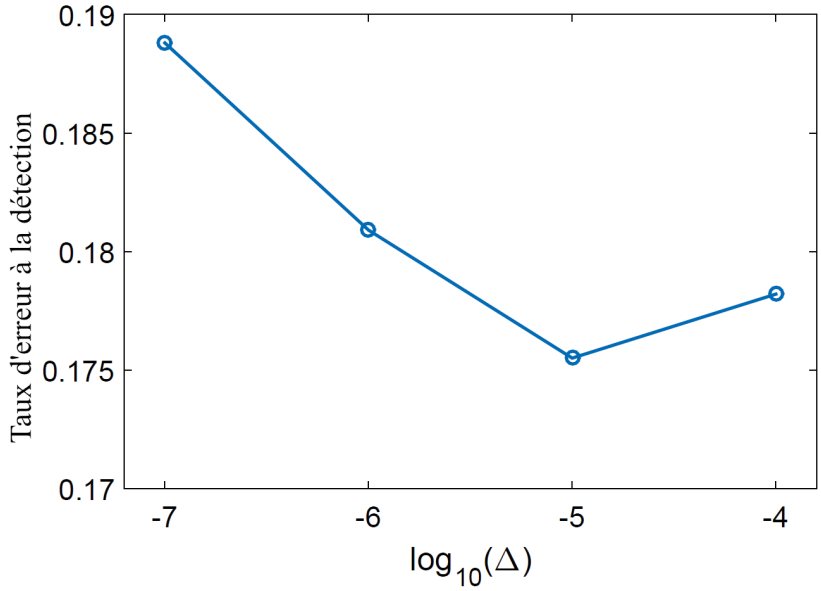


FIGURE 6.2 – Taux d'erreurs à la détection d'objets 3D marqués selon la méthode d'IDC haute capacité proposée pr Itier et Puech [64] en fonction du paramètre d'intervalle Δ.

	HPQ(24 bps)	HPQ-PA (8 bps)	HPQ-R (8 bps)	HPQ-R (24 bps)
LFS52	0.1888 (±.0253)	0.2553 (±.0242)	0.3112 (±.0259)	0.3085 (±.0246)
Angle dièdre	0.1866 (±.0174)	0.3431 (±.0222)	0.4400 (±.0197)	0.4441 (±.0156)
Laplacien	0.3545 (±.0168)	0.3750 (±.0227)	0.3800 (±.0276)	0.3803 (±.0197)
Courbure	0.3563 (±.0232)	0.3963 (±.0190)	0.4033 (±.0253)	0.4069 (±.0212)

TABLEAU 6.1 – Résultats médians et écarts-types des taux d'erreur de détection pour la stéganalyse d'objets 3D marqués selon la méthode d'IDC haute-capacité HPQ [64] et ses variantes, pour $\Delta = 10^{-7}$.

la variante insérant seulement dans la coordonnée angulaire (ϕ) du SCS avec une charge utile de 8 bps, HPQ-R représente la variante insérant seulement dans la coordonnée radiale du SCS. Pour HPQ-R, nous considérons deux niveaux de charge utile pour cette variante, à savoir 8 bps et 24 bps correspondant à deux tailles de sous-intervalles $s = 2^8$ et $s = 2^{24}$, respectivement. Dans toutes les méthodes évaluées, le paramètre d'intervalle Δ est égal à 10^{-7}.

Durant les tests, nous considérons les caractéristiques provenant de l'ensemble LFS52 afin d'observer l'influence de l'insertion dans les différentes coordonnées du SCS sur les caractéristiques utilisées pour la stéganalyse. Dans le tableau 6.1, l'angle dièdre correspond au sous-ensemble de 4 caractéristiques représentant les moments (moyenne, variance, le coefficient d'asymétrie et le *kurtosis*) calculés à partir des angles dièdres des objets 3D. Le champ "Laplacien" représente le sous-ensemble de 16 caractéristiques représentant les 4 moments calculés à partir des coordonnées des sommets et de leur normale dans le système de coordonnées laplacien. Le champ "Courbure" correspond au sous-ensemble de 8 caractéristiques représentant les informations de courbure au niveau des sommets [158]. Après comparaison des résultats de HPQ, HPQ-PA et HPQ-R, nous remarquons que l'insertion dans la coordonnée radiale du SCS améliore grandement la résistance à la stéganalyse de la méthode proposée par Itier et Puech [64]. En effet, le taux d'erreur à la détection augmente passant ainsi de 18.88% à 31.12% lors de l'utilisation de l'ensemble de caractéristiques LFS52. Les caractéristiques basées sur l'angle dièdre sont les plus efficaces pour détecter l'utilisation de la méthode HPQ (18.66%), mais elles deviennent inefficaces dès qu'il s'agit de détecter les insertions à base de HPQ-R (44.40%). Cela signifie que les angles dièdres des objets 3D sont mieux préservés lorsque l'insertion se passe uniquement sur la coordonnée radiale. De plus, même si nous triplons la charge utile pour HPQ-R, sa résistance à la stéganalyse baisse seulement de 0.8% par rapport à l'insertion avec une charge utile de 8 bps pour l'ensemble de caractéristiques LFS52.

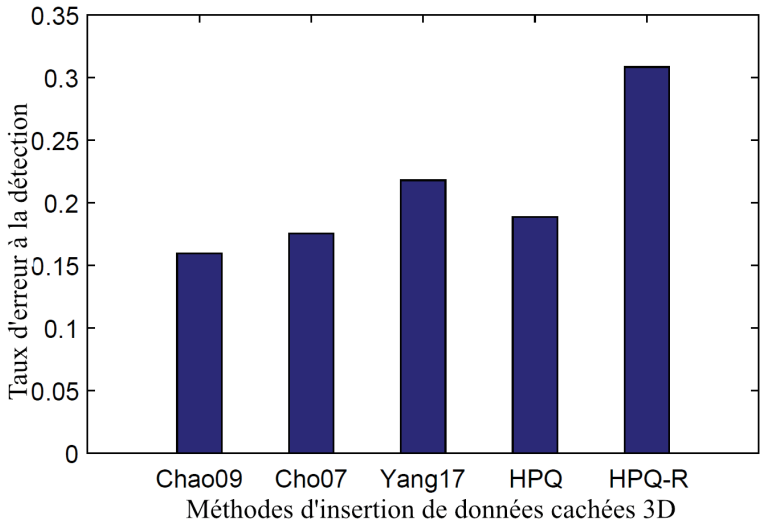


FIGURE 6.3 – Taux d'erreurs de détection pour la stéganalyse de méthodes d'insertion de données cachées 3D de l'état de l'art, en utilisant l'ensemble de caractéristiques LFS52 [89].

Nous comparons la méthode proposée HPQ-R à d'autres méthodes d'IDC de l'état de l'art sur leur résistance à la stéganalyse. Ainsi nous comparons avec les approches proposées par Chao *et al.* [31], Cho *et al.* [35], Yang *et al.* [159] et la méthode originale de Itier et Puech [64] notées Chao09, Cho07, Yang17 et HPQ, respectivement. Pour Chao *et al.* [31], nous considérons un nombre de couches d'insertion de 10 et un nombre d'intervalles de

10^4 de manière à ce que la charge utile soit de 10 bits par sommet environ. Pour la méthode de Cho *et al.* [35], nous insérons au maximum 64 bits et la force du tatouage est de 0.04. Nous assignons le paramètre K dans la méthode Yang17 [155] à 128 tandis que la limite supérieure de l'algorithme pour la charge utile est de $\lceil (K-2)/2 \rceil$ bits. Comme illustré en figure 6.3, les taux d'erreurs à la détection des objets 3D marqués selon les différentes méthodes varient entre 0.15 et 0.3 lors de l'utilisation de l'ensemble LFS52 pour la stéganalyse. Nous remarquons que l'approche HPQ-R proposée possède la plus forte résistance à la stéganalyse face à des systèmes entraînés avec l'ensemble de caractéristiques LFS52. Ainsi, notre amélioration provoque une réduction du taux de détection des objets 3D marqués pour les systèmes de stéganalyse de 9 à 15%.

6.2.4 Conclusion

Dans cette section, nous nous sommes concentrés sur l'amélioration de la méthode d'insertion de données cachées haute-capacité proposée par Itier et Puech [64] réalisant une quantification d'un chemin hamiltonien construit sur le nuage de points, face à des systèmes de stéganalyse cherchant à détecter la présence de messages cachés au sein des objets 3D. Nous analysons l'influence du paramètre d'intervalle Δ et du nombre de sous-intervalles sur le déplacement des sommets servant à coder le message lors de la construction du chemin hamiltonien. Nous observons également que l'insertion de données uniquement sur la coordonnée radiale selon le système de coordonnées sphériques génère moins de distorsions sur les caractéristiques utilisées lors de la stéganalyse. Au contraire, l'insertion de données sur les coordonnées angulaires est plus facilement détectable. Les résultats expérimentaux présentent différents tests où l'insertion se déroule sur les trois coordonnées sphériques ou uniquement sur la coordonnée radiale ou une coordonnée angulaire. Des systèmes de stéganalyse sont ensuite entraînés pour reconnaître les objets 3D marqués avec l'ensemble de caractéristiques LFS52. Les expérimentations confirment nos analyses sur l'influence de la coordonnée radiale et les coordonnées angulaires. Dans des travaux futurs, nous souhaitons améliorer encore plus la résistance à la stéganalyse de nos objets 3D marqués en optimisant l'insertion de données cachées en rajoutant des heuristiques minimisant les modifications détectées par les caractéristiques utilisées par le système de stéganalyse.

6.3 Évaluation subjective de la confidentialité des objets 3D

Dans cette section, nous nous intéressons à l'évaluation de la confidentialité des objets 3D sélectivement chiffrés. En effet, nous avons pu constater dans le chapitre 3 pour le chiffrement sélectif 3D, dans la section 4.3 du chapitre 4 pour le partage sélectif d'objets 3D et dans la section 5.3 du chapitre 5, que nos méthodes permettent de protéger les objets 3D secrets selon différents niveaux de confidentialité. Ainsi, à l'aide d'un niveau de dégradation, nous pouvons assurer la confidentialité visuelle d'un objet 3D en protégeant sa forme et son contenu, ou alors assurer une protection suffisante de manière à laisser la forme de l'objet 3D reconnaissable, mais pas son contenu. Et enfin, nos méthodes peuvent permettre aux utilisateurs de reconnaître le contenu et la forme des objets 3D tout en protégeant la haute qualité de ces objets (protection transparente). Nous avons également pu constater que les métriques dites objectives permettaient à partir d'un certain niveau de dégradation de détecter la présence des distorsions géométriques, mais ne permettaient pas de manière isolée de distinguer un niveau de confidentialité spécifique.

En effet, la plupart des métriques présentées en section 1.2.2 du chapitre 1, ont pour but d'analyser la qualité des objets 3D, mais pas leur confidentialité. Ainsi, dès que du bruit est ajouté à la position des sommets, il est très simple pour ces métriques de détecter la présence de ces distorsions géométriques. Dans ces travaux, nous cherchons à établir une métrique de confidentialité pour les objets 3D sélectivement chiffrés permettant de différencier les niveaux de confidentialité présentés en section 1.4.3 du chapitre 1, à savoir le chiffrement transparent, suffisant ou la confidentialité visuelle.

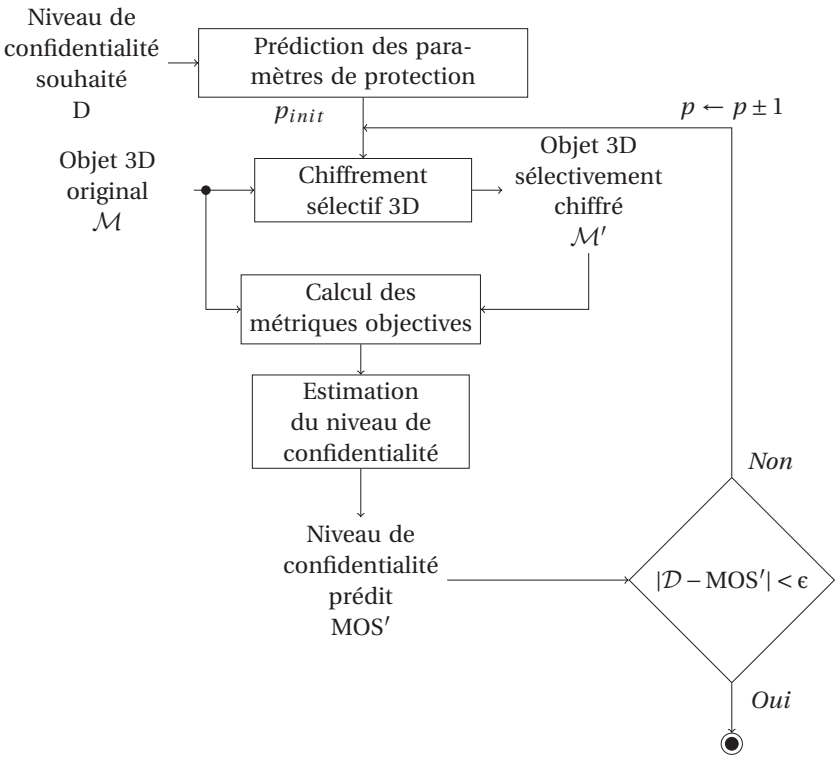


FIGURE 6.4 – Processus de chiffrement sélectif 3D itératif avec estimation des paramètres de chiffrement et prédiction du niveau de confidentialité.

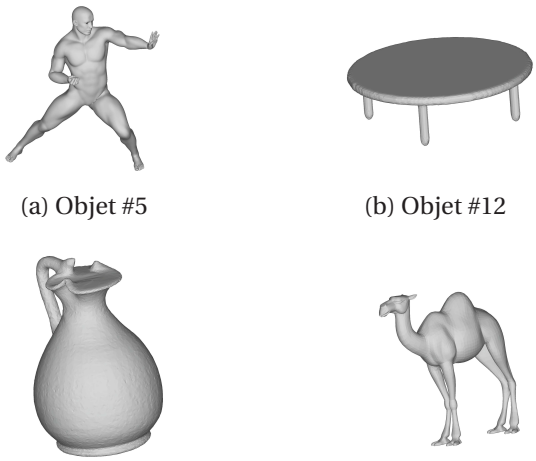
La première utilisation souhaitée de cette métrique consiste à prédire, à la sortie d'une méthode de chiffrement sélectif 3D, si l'objet 3D sélectivement chiffré possède bien le niveau de confidentialité souhaité comme illustré en figure 6.4. Ainsi, nous souhaitons étudier la performance des métriques avec référence de l'état de l'art pour ce travail de prédiction du niveau de confidentialité des objets 3D sélectivement chiffrés. La seconde utilisation, illustrée en figure 6.4 également, consiste à automatiser le choix des paramètres donnés en entrée de notre méthode de chiffrement sélectif 3D proposée dans le chapitre 3 en fonction des besoins utilisateurs représentés sous la forme d'un niveau de confidentialité. Pour cela, nous utilisons des approches basées par apprentissage sur les données récoltées lors de nos évaluations subjectives.

En section 6.3.1, nous présentons la base de données d'objets 3D sélectivement chiffrés que nous avons construite pour réaliser les évaluations subjectives et ainsi collecter des réponses d'observateurs face à des objets 3D. Nous détaillons ensuite le protocole d'évaluation et ses spécificités dans la section 6.3.2. En section 6.3.3, nous analysons statistiquement les résultats des évaluations subjectives et essayons de corréler ces résultats à différentes métriques 3D selon différentes approches. Enfin, nous concluons cette partie sur nos travaux d'analyse de la confidentialité des objets 3D sélectivement chiffrés en section 6.3.4.

6.3.1 Construction de la base de données d'objets 3D sélectivement chiffrés

Dans le domaine de l'informatique graphique, il n'existe pas, à notre connaissance, de base de données d'évaluations subjectives portant sur la confidentialité d'objets 3D sélectivement chiffrés. Les bases de données d'objets 3D ont été principalement construites pour l'étude de la qualité [56] dans des contextes de traitement, de compression ou bien pour des applications de segmentation d'objets 3D [33, 82, 103]. Ainsi, ces base de données ont été construites pour étudier un spectre plus restreint des objets 3D où la qualité est un facteur essentiel, alors que dans ces travaux nous cherchons à étudier la confidentialité des objets 3D.

La base de données que nous proposons est construite à partir d'objets 3D provenant des bases de données existantes, à savoir celle du *Princeton Mesh Segmentation Project* [33], du challenge *SHREC* pour la segmentation 3D de 2012 [82] et de 2014 [103] ou encore de la base de données *Thingi10k* [163]. À partir de toutes ces bases de données, nous sélectionnons 50 objets 3D. 4 sont présentés en figure 6.5, qui vont servir à construire notre base de données, nommée *Visual Confidentiality Dataset* (VCD).



(a) Objet #5 (b) Objet #12

(c) Objet #24 (d) Objet #32

FIGURE 6.5 – Exemples d'objets 3D de référence de la base de données *Visual Confidentiality Dataset* (VCD).

Comme illustré en figure 6.5, certains objets 3D de référence peuvent provenir de CAO (cf. figure 6.5.a) ou bien de systèmes d'acquisition (cf. figure 6.5.b). Enfin, certains de ces objets 3D représentent des entités vivantes, dites organiques, tandis que le complément représente des objets inertes (casque, statue, vase, *etc.*). En faisant varier les origines des objets 3D de référence ainsi que le contenu représenté, nous souhaitons pouvoir évaluer une grande variété d'objets 3D. Tout cela dans le but d'étudier les effets du chiffrement sélectif 3D et voir où ce dernier est le plus efficace pour protéger le contenu, mais aussi le plus adapté en fonction de la densité de points, de la taille de l'objet ou bien de ses courbures locales pouvant influencer sa forme chiffrée.

La figure 6.6 représente la distribution des objets en fonction de leur origine et de leur type. Parmi ces 50 objets 3D de référence, 31 d'entre eux sont des objets 3D provenant de numérisation à l'aide de système d'acquisition contre seulement 19 objets 3D produits à l'aide d'outils de CAO. Parmi ces objets 3D, 21 d'entre eux représentent des entités vivantes que nous notons organiques tandis que les 29 autres représentent des objets

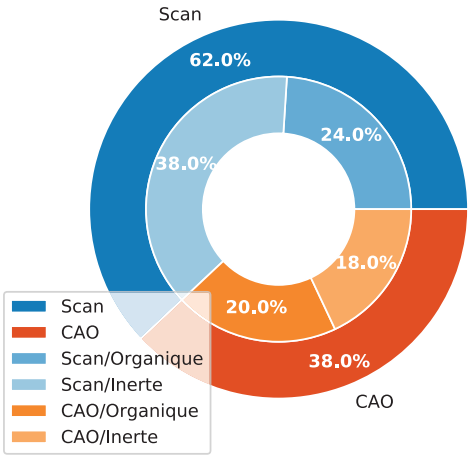


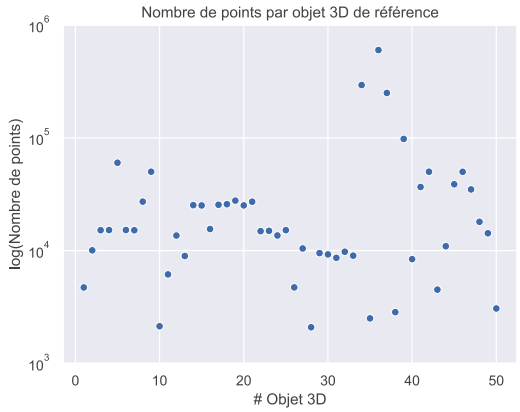
FIGURE 6.6 – Distribution des objets 3D de référence de la base de données *Visual Confidentiality Dataset* (VCD) en fonction de l'origine (CAO ou Scan) et du type des données (Organique ou Inerte).

inertes. Nous notons que les objets 3D de référence se distinguent par certaines propriétés à savoir le nombre de points, le nombre de faces ainsi que la longueur moyenne d'une arête et d'autres caractéristiques comme illustré en figure 6.7.

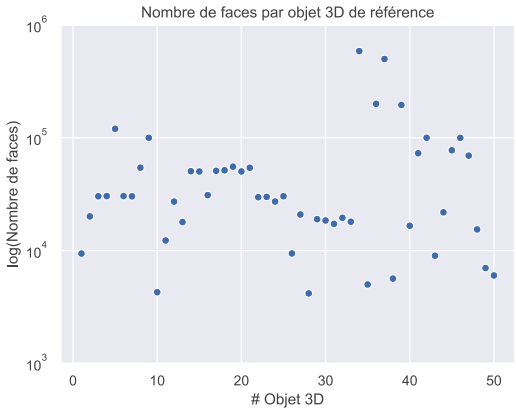
Nous proposons de chiffrer ces objets 3D de référence en utilisant la méthode de chiffrement sélectif présentée dans le chapitre 3. Nous chiffrons les 50 objets 3D en utilisant les niveaux de dégradation $\mathcal{D} =< p, 9 >$, où $p \in [\![13\,;\,22]\!]$ comme illustré en figure 6.8 pour l'objet #18. Pour chaque chiffrement, nous changeons la clé secrète $\mathcal{K}$ pour éviter d'avoir un biais. Au final, en comptant les objets 3D de référence, nous obtenons notre base de données VCD qui est composée de 550 objets 3D dont 500 sont chiffrés sélectivement.

6.3.2 Protocole d'évaluation

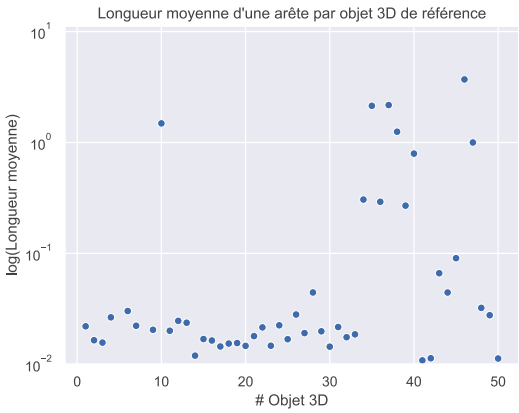
Dans cette partie, nous détaillons le protocole d'évaluation. Généralement, les images numériques sont sujettes à de grandes variétés de distorsions liées à l'acquisition, aux traitements, à la compression, au stockage, à la transmission, à la reproduction ou à l'insertion de données cachées. Il en résulte alors une dégradation de la qualité visuelle. Le même constat peut être fait pour les objets 3D. Au départ, les évaluations subjectives sont généralement employées pour évaluer la performance des métriques objectives. Avec l'arrivée des méthodes d'apprentissage automatique (ou *Machine Learning*), certaines métriques sont construites à partir des données produites par les évaluations [83]. Les métriques subjectives se basent sur des évaluations de la qualité perçue par les observateurs. Les évaluations subjectives consistent donc à faire analyser par des observateurs (experts ou non-experts) des images, des vidéos, ou bien des objets 3D dans le but de noter la qualité selon une échelle de qualité prédéfinie. Cette échelle peut être construite en utilisant des catégories (excellente, bonne, pauvre, ou mauvaise) ou bien avec un score allant de 1 (la plus mauvaise qualité) à 10 (la meilleure qualité).



(a) Nombre de points

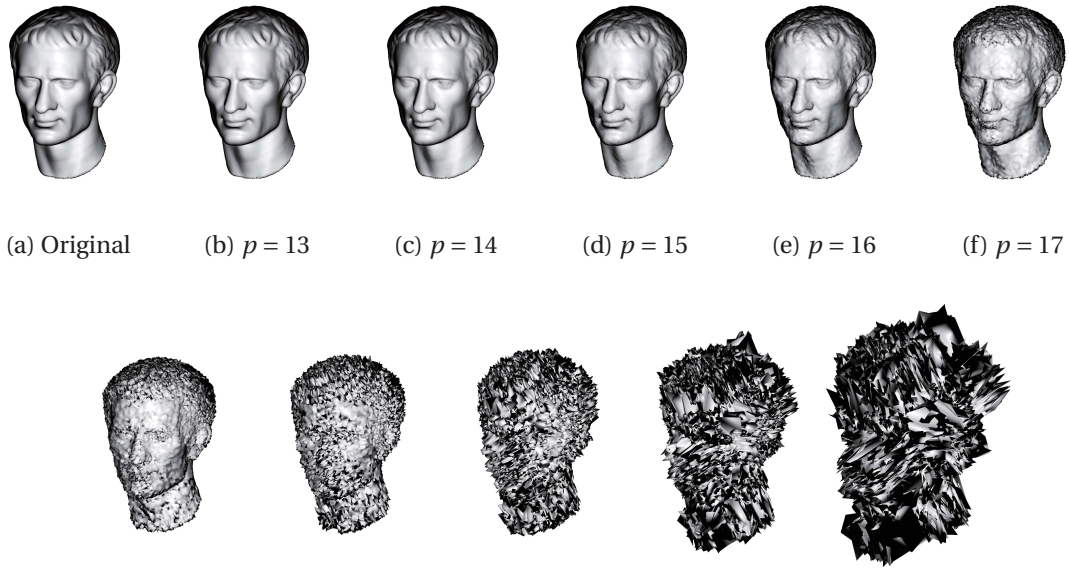


(b) Nombre de faces



(c) Longueur moyenne d'une arête

FIGURE 6.7 – Caractéristiques des objets 3D de référence de la base VCD.



(a) Original (b) $p = 13$ (c) $p = 14$ (d) $p = 15$ (e) $p = 16$ (f) $p = 17$

(g) $p = 18$ (h) $p = 19$ (i) $p = 20$ (j) $p = 21$ (k) $p = 22$

FIGURE 6.8 – Objets 3D représentant le modèle #18 de la base VCD, selon les différents niveaux $\mathcal{D} =< p, 9 >$, où $p \in [\![13 ; 22]\!]$.

Système d'évaluation

La mesure la plus commune pour la création de ces métriques est le score d'opinion (*Opinion Score* ou OS) qui varie généralement entre 1 et 5.

OS	Signification	Forme	Contenu	Qualité
1	Confidentialité visuelle	Confidentielle	Confidentiel	Médiocre
2	Protection suffisante	Accessible	Confidentiel	Médiocre
3	Protection transparente	Accessible	Accessible	Basse
4	Objet 3D bruité	Accessible	Accessible	Moyenne
5	Objet 3D sans distorsion	Accessible	Accessible	Haute

TABLEAU 6.2 – Valeurs et significations des scores d'opinion dans le cadre de l'évaluation de la confidentialité.

Le tableau 6.2 présente l'échelle que nous avons définie pour l'étude de la confidentialité des objets 3D pour nos évaluations subjectives. Ainsi, les valeurs de 1, 2 et 3 correspondent aux différents niveaux de chiffrement définis par A. Pommer [1] dans le cadre de scénarios pour le chiffrement sélectif de données visuelles, à savoir la confidentialité visuelle, le chiffrement suffisant et le chiffrement transparent. Ainsi, nous avons associé un OS de valeur 1 à la confidentialité visuelle là où nos méthodes de chiffrement sélectif d'objets 3D génèrent des objets 3D dont la forme et le contenu sont protégés. Le OS de valeur 2 est associé au chiffrement suffisant permettant de reconnaître uniquement la forme de l'objet 3D, mais pas son contenu. Enfin un OS de valeur 3 correspond au chiffrement transparent permettant de reconnaître la forme et le contenu, mais la haute qualité est protégée. Un OS de valeur 4 permet aux observateurs de différencier les objets 3D seulement bruités des objets 3D qui sont de bonne qualité avec un OS de valeur 5. La valeur de OS 5 correspond à des objets 3D n'ayant visiblement aucun défaut et par conséquent

à des objets 3D de bonne qualité et non chiffrés.

Mode de *stimulus*

Parmi les différents protocoles d'évaluation subjective de la qualité il existe 4 modes pré-dominants, à savoir le *single-stimulus*, le *double-stimulus*, le *forced-choice pairwise comparison* et *similarity judgments*. Chacun de ces modes possède ses avantages ainsi que ses défauts [56]. Cependant, dans le cadre de notre évaluation subjective, il apparaît très clair que nous devons employer le protocole *single-stimulus*. En effet, étant à la recherche d'une métrique cherchant à quantifier la confidentialité d'un objet 3D sélectivement chiffré, l'utilisation de protocoles utilisant deux objets 3D (l'objet 3D original et celui chiffré, par exemple) fournit des informations sur la forme et le contenu de l'objet 3D dont nous cherchons à analyser la confidentialité. Ainsi, les autres modes de *stimulus* cherchant à confronter deux objets 3D de qualité différente ne nous apporteraient aucune information sur la confidentialité.

De par la nature des données, dans notre cas des objets 3D, nous permettons aux observateurs d'interagir avec ces objets 3D en autorisant les déplacements de caméra (translation, rotations, zooms) afin qu'ils puissent observer la forme de l'objet 3D sous tous ses angles.

Environnement d'évaluation

Les évaluations subjectives avec un protocole standardisé fournissent des résultats corrects et universels. Cependant, elles nécessitent un environnement contrôlé et de nombreuses limitations à cause du jugement humain pouvant varier de manière significative en fonction des conditions externes et des individus. C'est pourquoi nos évaluations sont réalisées dans une pièce spécialisée à huis clos pour garder le contrôle sur des éléments essentiels comme la lumière, la résolution de l'écran et la distance à l'écran.

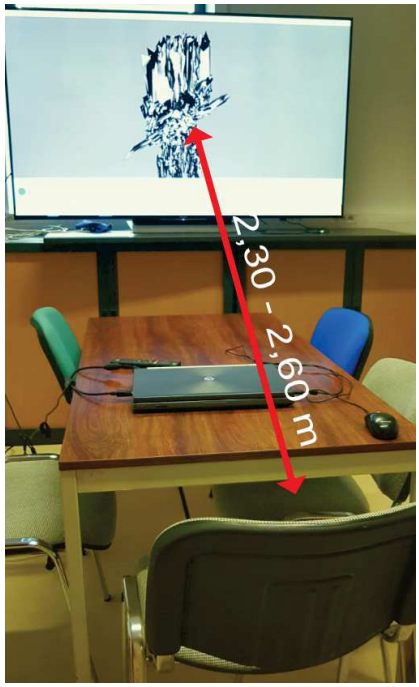


FIGURE 6.9 – Salle d'évaluation subjective de la confidentialité des objets 3D sélectivement chiffrés.

Comme illustré en figure 6.9, les observateurs sont positionnés en face d'un écran professionnel LCD *Sony FW-75XD8501* 4K Ultra HD de 190.5cm, basé sur une technologie LED avec une résolution de 3840x2160 pixels et avec une luminosité d'environ 450cd/m^2. Les observateurs se trouvent à une distance comprise entre 2.30-2.60 mètres.

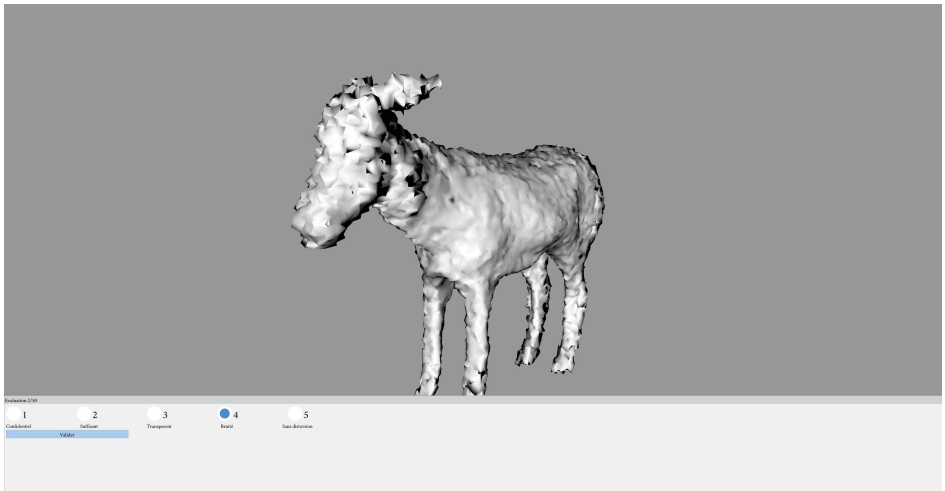


FIGURE 6.10 – EVA3D : Outil d'évaluation subjective de la confidentialité des objets 3D sélectivement chiffrés, l'observateur sélectionne une valeur de score d'opinion pour l'objet 3D sélectivement chiffré.

Les objets 3D sont affichés sur un fond gris uniforme, sans texture, en utilisant un *shader* basé sur le modèle de Phong [102] avec un matériaux gris clair virant vers le blanc dans les zones spéculaires comme illustré en figure 6.10.

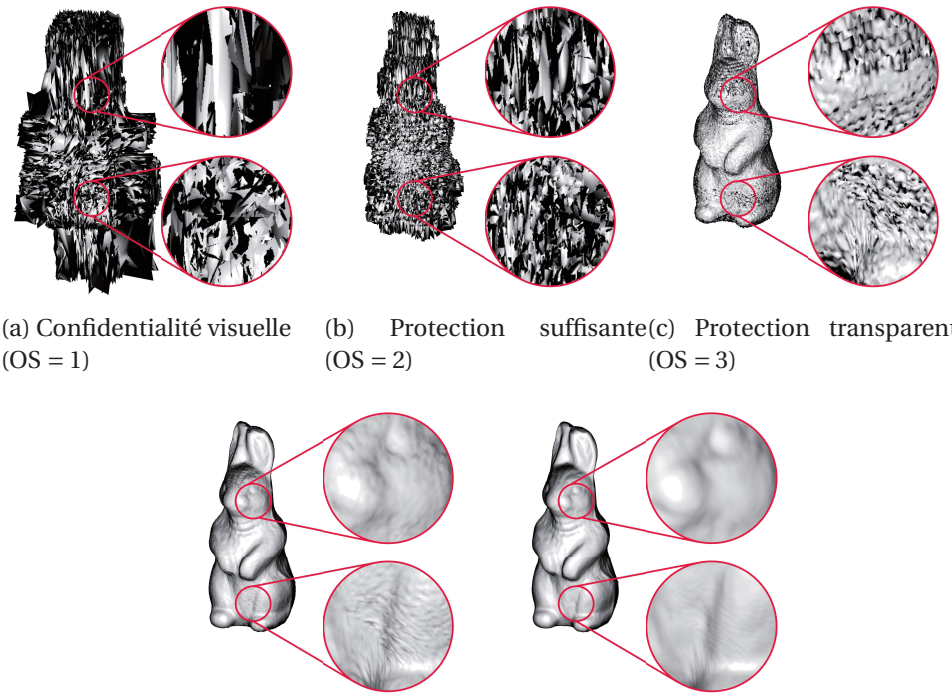
Procédure d'évaluations

Comme décrit en section 6.3.1, nous avons 50 objets 3D de référence et pour chacun de ces objets, 10 variations supplémentaires en fonction des paramètres de chiffrement ($13 \leq p \leq 22$). Nous décidons de faire évaluer à chaque observateur 50 objets 3D distincts. Pour éviter que les observateurs apprennent à reconnaître les objets 3D, nous décidons de ne leur montrer qu'une seule version de chacun des 50 objets 3D de référence. Ainsi pour chaque objet 3D de base, nous tirons aléatoirement une variation. Avant la phase d'évaluation, les observateurs sont initiés au problème de la confidentialité des objets 3D.

Comme illustré en figure 6.11, les observateurs sont initiés à la problématique en présentant un objet 3D sélectivement chiffré à différents niveaux de dégradation à l'aide de l'outil EVA3D. L'ordre dans lequel les objets 3D sélectivement chiffrés sont présentés est important, car il sert à montrer l'évolution de la forme de l'objet 3D vers quelque chose de reconnaissable, de même pour le contenu et comment le score d'opinion doit être assigné durant la phase d'évaluation.

6.3.3 Analyse des évaluations

Dans cette section, nous présentons les différents résultats issus de notre phase d'évaluations subjectives de la confidentialité des objets 3D sélectivement chiffrés. Tout d'abord, 54 observateurs ont participé à ces évaluations subjectives, soit 2700 valeurs de MOS assignées par les observateurs aux objets 3D de la base VCD. Dans un premier temps, nous détaillons des informations sur les observateurs anonymes (genre, âge). Nous analysons



(a) Confidentialité visuelle (OS = 1)
(b) Protection suffisante (OS = 2)
(c) Protection transparente (OS = 3)
(d) Objet 3D bruité (OS = 4)
(e) Objet 3D sans distorsion (OS = 5)

FIGURE 6.11 – Objets 3D sélectivement chiffrés pour la phase d'initiation des observateurs aux différents niveaux de confidentialité des objets 3D au sein de l'application EVA3D.

ensuite la distribution générale des scores sur la base de données et les premières déductions qui peuvent en être tirées.

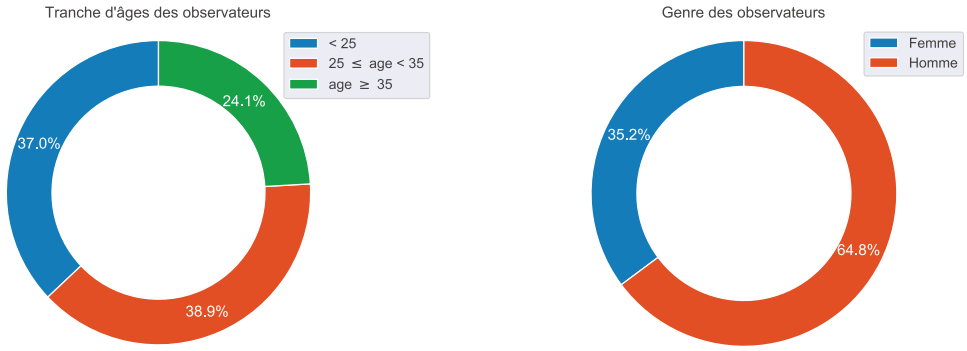
Analyse des observateurs

La population d'observateurs ayant participé à nos évaluations subjectives est variée. Elle est composée à la fois d'experts dans le domaine de l'informatique graphique, du traitement d'image ainsi que de personnes dites non-expertes.

La figure 6.12 représente la distribution des observateurs selon leur âge ou leur genre. Comme illustré en figure 6.12.a, nos observateurs sont divisés en trois tranches d'âge, à savoir les moins de 25 ans (20), les 25-35 ans (21) et les plus de 35 ans (13). Nous notons que 19 observateurs sont des femmes et 31 des hommes selon la figure 6.12.b.

Analyse des évaluations

Les 54 observateurs ont donc généré 2700 valeurs de scores d'opinion (OS) réparties sur l'ensemble des 550 objets 3D de la base de données. Sur les 2700 OS, 263 sont dédiés aux objets 3D de référence. Ces scores sur les objets 3D de référence servent principalement à identifier les objets 3D de référence ambigus, c'est-à-dire les objets ayant une apparence présentant naturellement des distorsions pour les observateurs. Cela permet ainsi d'analyser la perception de la qualité pour les objets 3D provenant de numérisation d'objets du monde réel. De plus, nous pouvons en tirer des informations supplémentaires sur le seuil de sensibilité aux distorsions des différents observateurs. Au final, 2437 OS sont dédiés à l'évaluation des objets 3D sélectivement chiffrés par notre méthode.



(a) Âge (b) Genre

FIGURE 6.12 – Distributions des observateurs par : a) Âge, b) Genre.

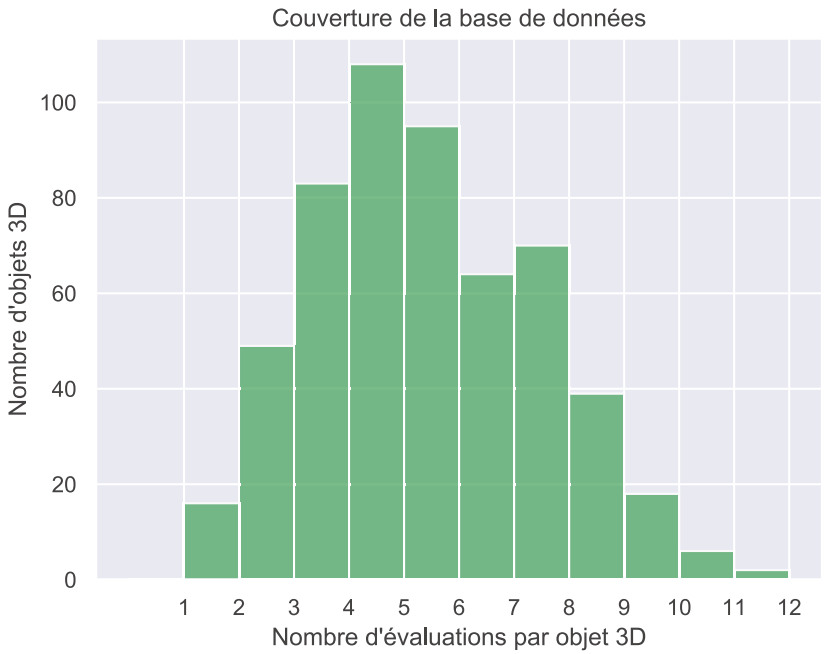


FIGURE 6.13 – Distribution du nombre d'objets 3D en fonction de leur nombre d'évaluations.

Du fait de la sélection aléatoire des objets 3D lors de la phase d'évaluation, nous avons une base avec des objets 3D ayant un nombre différent d'évaluations. Dans la figure 6.13, nous présentons la distribution du nombre d'objets 3D en fonction du nombre d'évaluations par objet 3D. Nous pouvons constater que 100% des objets 3D de la base ont été évalués au moins une fois.

Nous remarquons également que seulement 16 objets 3D se retrouvent avec une seule évaluation, 49 avec deux évaluations, 83 avec trois évaluations, 108 avec quatre évaluations, 95 avec cinq évaluations jusqu'à seulement 2 objets 3D avec onze évaluations. Le tableau 6.3 récapitule le nombre d'objets 3D évalués ayant dépassé au moins t évaluations, où $t \in [\![1 ; 11]\!]$ tels qu'illustré dans la figure 6.13. Ainsi, nous notons que 100% de la base a été évaluée au moins 1 fois, 97.1% au moins deux fois, 88.18% au moins trois fois, jusqu'à 53.4% au moins cinq fois. Ainsi, 294 objets 3D ont été évalués au moins cinq fois au sein de notre base de données et 256 objets 3D évalués moins de cinq fois. Et 2 objets 3D ont été évalués onze fois, ce qui a été le nombre maximum d'évaluations pour un objet 3D.

TABLEAU 6.3 – Nombre d'objets 3D évalués ayant au moins t fois.

Nombre d'évaluations	1	2	3	4	5	6	7	8	9	10	11
Nombre d'objets 3D	550	534	485	402	294	199	135	65	26	8	2
%	100	97.09	88.18	73.09	53.45	36.18	24.55	11.82	4.738	1.455	.3636

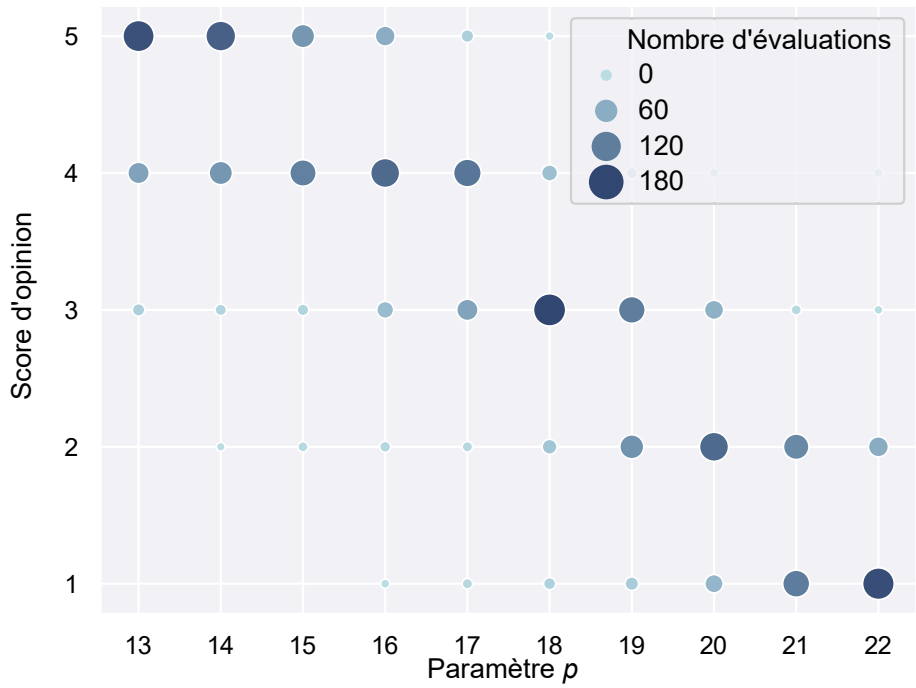


FIGURE 6.14 – Distribution des valeurs des OS en fonction du paramètre p.

Dans la figure 6.14, nous pouvons observer l'évolution des valeurs des scores d'opinion données par les observateurs durant les évaluations subjectives en fonction du paramètre p utilisé lors du chiffrement sélectif des objets 3D ($13 \leq p \leq 22$). Dans notre base

VCD, nous remarquons que les valeurs des OS attribuées varient généralement autour de trois, quatre valeurs pour chaque niveau de chiffrement p. Ainsi, nous déduisons qu'il n'existe pas de valeur spécifique de p nous permettant d'obtenir une valeur unique de OS, quelque soit l'objet 3D chiffré. Nous notons tout de même que pour $p \in \{13,\ 14\}$, les observateurs ont majoritairement privilégié un OS de 5, pour $p \in \{15,\ 16,\ 17\}$ un OS de 4, pour $p \in \{18,\ 19\}$ un OS de 3, pour $p \in \{20,\ 21\}$ un OS de 2 et enfin un OS de 1 pour $p \in \{22\}$. De plus, nous constatons que pour certaines valeurs de p, les valeurs de OS sont attribuées de manière quasi-uniforme entre deux valeurs, en particulier pour $p \in \{15,\ 19,\ 21\}$. Nous pouvons en déduire qu'il s'agit de valeurs de p pivots où un changement visuel se produit pour une grande partie des objets 3D sélectivement chiffrés pour les observateurs. Nous remarquons tout de même qu'il existe des couples de valeurs (p, OS) très rares, mais qui existent tout de même. Par exemple, la première évaluation donnant un OS de 1 apparaît pour $p = 16$, tandis que pour un OS de 5 le dernier paramètre de chiffrement p est égal à 18.

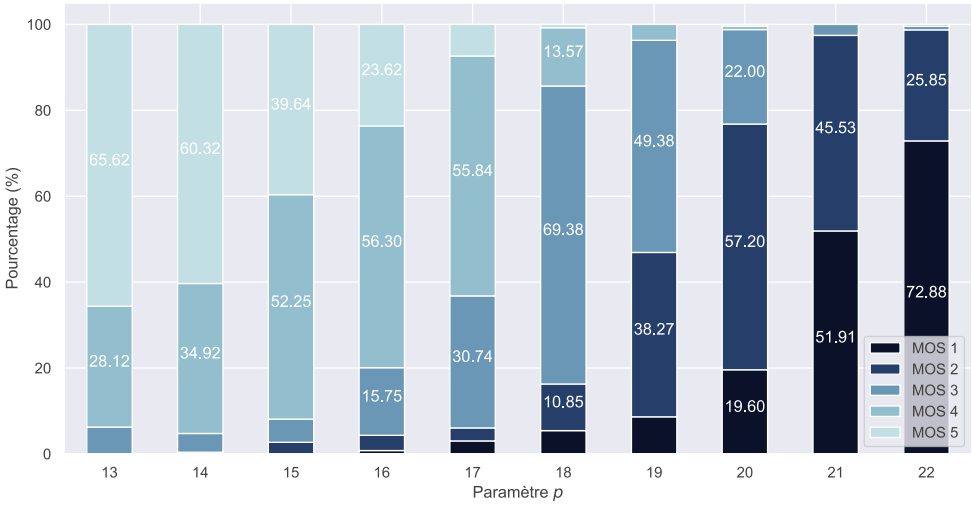


FIGURE 6.15 – Distribution des pourcentages des valeurs des OS en fonction du paramètre p.

La figure 6.15 présente la distribution des pourcentages des valeurs des OS en fonction du paramètre p. Nous constatons plus précisément que les observateurs ont très majoritairement (valeur très supérieure à 50%) donné des OS de 5 pour $p \in \{13,\ 14\}$ (65.62%, 60.32%), des OS de 4 pour $p \in \{\ 16,\ 17\}$ (56.30%, 55.84%), des OS de 3 pour $p \in \{18\}$ (69.38%), des OS de 2 pour $p \in \{20\}$ (57.20%) et des OS de 1 pour $p \in \{\ 22\}$ (72.88%). Les observateurs sont cependant plus mitigés pour $p \in \{15,\ 19\ 21\}$. En effet, si nous regardons plus en détail pour $p = 15$, bien que les observateurs ont voté pour un OS de 4 à 52.25%, il y a tout de même 39.64% qui ont choisi un score de 5. Nous retrouvons une situation similaire pour $p = 19$ où aucune valeur de OS ne dépasse les 50%, avec 49.38% pour un OS de 3, 38.27% pour un OS de 2 et 12.35% pour les autres valeurs de OS. Enfin, pour $p = 21$, un OS de 1 est de 51.91% contre 45.53% avec un OS de 2. À partir de ces résultats, nous pouvons commencer à entrevoir des intervalles précis représentant les différents niveaux de confidentialité. Ainsi, les observateurs considèrent les objets sélectivement chiffrés avec un paramètre p égal à 18 ou 19 comme étant une protection transparente, à 19 ou 20 comme étant une protection suffisante et enfin à 21 ou 22 comme une confidentialité visuelle. Malgré un chiffrement avec p égal à 13 ou 14, les observateurs considèrent majoritairement que les objets 3D n'ont pas de distorsions géométriques visibles et que

celles-ci n'apparaissent qu'à partir de $p = 15$.

(a) MOS (b) Med-OS

FIGURE 6.16 – Distribution des valeurs moyennes (MOS) et médianes (Med-OS) des scores d'opinions pour chaque objet 3D chiffré en fonction du paramètre p.

La figure 6.16.a représente les valeurs moyennes des OS calculées pour chaque objet 3D chiffré de la base VCD à partir des valeurs de OS fournies par les observateurs. Nous notons ces valeurs moyennes des scores d'opinion : *Mean Opinion Score* (MOS). Nous constatons en figure 6.16.a que, malgré les MOS donnés majoritairement par les observateurs pour chaque valeur du paramètre de chiffrement p présent dans la figure 6.15, certains objets 3D sont considérés fortement chiffrés malgré une valeur de p faible. Ou au contraire, des objets 3D sont considérés avec un chiffrement transparent ou suffisant malgré une valeur de p élevée.

Dans la figure 6.16.b, nous présentons les valeurs médianes des OS pour chaque objet 3D chiffré en fonction du paramètre p. Nous notons ces valeurs médianes des scores d'opinion : *Median Opinion Score* (Med-OS). Ainsi, nous pouvons constater des résultats très similaires à ceux présents sur la figure 6.16.b.

Prédiction des paramètres de protection à partir d'un niveau de confidentialité (D) **souhaité**

Pour pouvoir prédire le paramètre de chiffrement p, nous calculons des corrélations entre les valeurs de p et les valeurs des OS des observations. Un coefficient de corrélation est une mesure statistique décrivant la relation linéaire entre deux variables. Ces valeurs de corrélation sont comprises entre -1 et $+1$. Une valeur de corrélation proche de $+1$ indique que les deux variables ont une très grande relation linéaire positive, tandis qu'un coefficient de corrélation proche de -1 montre que les deux variables ont une très grande relation linéaire négative. Un coefficient de corrélation proche de 0 indique que les veux variables sont très indépendantes et qu'il n'existe aucune relation entre les deux.

Le tableau 6.4 présente les coefficients de corrélation calculés entre le paramètre p et les valeurs des OS des observations. Nous avons choisi d'utiliser les données des évaluations de trois manières différentes, à savoir en utilisant directement toutes les valeurs de OS données par les observateurs, en utilisant les valeurs médianes des OS (Med-OS) pour chaque objet 3D et enfin en utilisant les valeurs moyennes des OS (MOS) pour chaque objet 3D. La première approche utilise l'ensemble des OS donnés (2437 évaluations) ce qui permet de calculer une valeur au plus proche de la réalité. Les deux autres approches

Valeurs	Coefficients de corrélation	Paramètre de chiffrement p
OS	Pearson	-0.861
	Spearman	-0.863
Med-OS	Pearson	-0.889
	Spearman	-0.892
MOS	Pearson	-0.906
	Spearman	-0.904

TABLEAU 6.4 – Coefficients de corrélation entre le paramètre p et les valeurs obtenues par les évaluations subjectives (OS) sur la base VCD.

(valeurs médianes et moyennes) utilisent, respectivement les Med-OS et les MOS attribués aux 500 objets 3D sélectivement chiffrés. Nous constatons qu'il existe une relation forte entre le paramètre p et les valeurs des OS des observateurs. Grâce à cette analyse, nous remarquons que le paramètre p est fortement corrélé aux valeurs des OS des observateurs. De ce fait, nous pouvons construire un modèle permettant de prédire la valeur de p en fonction d'un niveau de confidentialité souhaité. Pour cela, nous décidons d'appliquer une régression polynomiale afin de construire un modèle d'apprentissage statistique (ou *Statistical Learning*). Ainsi, nous séparons les données de la base VCD en deux bases d'objet 3D distincts, à savoir une base d'objets 3D pour la phase d'entraînement et une base d'objets 3D pour la phase de tests. Le but étant "d'entraîner" le modèle sur un sous-ensemble représentatif des données et de tester la validité du modèle sur le reste de la base qui n'aura jamais été observé par le modèle. Pour cela, nous utilisons 30 objets 3D de référence (et leurs 300 versions chiffrées associées) pour la phase d'entraînement et 20 objets 3D de référence (et leurs 200 versions chiffrées associées) pour les tests.

TABLEAU 6.5 – Résultats des régressions polynomiales en fonction des valeurs des OS des observateurs utilisés sur la base d'entraînement.

	Scores		
Métriques de régression	OS	Med-OS	MOS
R^2 *score*	0.7846	0.8442	**0.8651**
Explained Variance Score	0.7846	0.8442	**0.8651**
Mean Absolute Error	1.0089	0.8513	**0.7774**
Mean Squared Error	1.7807	0.2857	**1.1128**
Max Error	6.1276	5.2334	**4.7248**
Median Absolute Error	0.9268	0.7394	**0.5838**

Le tableau 6.5 présente les résultats des modèles de prédiction du paramètre p en fonction des valeurs des OS pour la base de d'entraînement. À nouveau, nous testons, avec la base d'entraînement, sur les valeurs des OS des observateurs, sur les valeurs médianes attribuées pour chaque objet 3D (Med-OS) et sur les valeurs moyennes des OS obtenues pour chaque objet 3D (MOS). Nous calculons en premier le coefficient de détermination (ou R^2 *score*) :

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{n-1} (y_i - \overline{y})^2}, \tag{6.4}$$

où y est le vecteur de scores de la vérité terrain, $\hat{y}$ le vecteur de scores prédits par le modèle

et $\overline{y}$ la moyenne des scores de y.

Le R^2 *score* donne une information sur la qualité du modèle, par exemple un modèle donnant les bonnes prédictions sans prendre en compte les données en entrée reçoit un score de 0.0. Le score peut devenir négatif si le modèle est mauvais, tandis qu'un modèle donnant de bons résultats en prenant en compte les données en entrée a un score qui tend vers 1.0.

Le score de variance expliquée est une métrique permettant d'évaluer la qualité des prédictions selon un rapport entre la différence des variances de la prédiction et de la vérité terrain :

$$EVS(y, \hat{y}) = 1 - \frac{variance(y - \hat{y})}{variance(y)}, \tag{6.5}$$

où *variance* est le carré de l'écart-type pour y et $\hat{y}$, respectivement $variance(y)$ et $variance(\hat{y})$.

Nous calculons également l'erreur absolue moyenne (ou *Mean Absolute Error*), l'erreur quadratique moyenne (ou *Mean Squarred Error*), l'erreur absolue maximale (ou *Max Error*), l'erreur absolue médiane (ou *Median Absolute Error*).

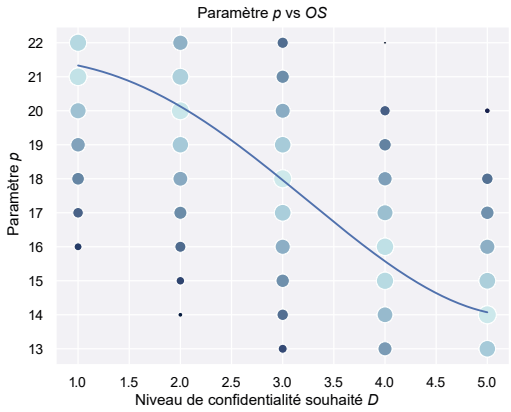
Dans le tableau 6.5, nous constatons que les meilleurs résultats sont obtenus lorsque nous entraînons avec les valeurs des MOS notre modèle pour prédire la valeur de p à partir d'un niveau de confidentialité (compris entre 1 et 5). En effet, le R^2 *score* a une valeur autour de 0.8651. De plus, nous remarquons que l'erreur absolue médiane est seulement de 0.5832 ce qui signifie que les valeurs de p prédites sont majoritairement proches de ce qui est attendu. Notons tout de même, l'erreur maximale qui est de 4.7248 pour les valeurs moyennes contre 6.1276 en utilisant les valeurs brutes. Cela signifie qu'il y a une grande diversité dans les objets 3D de la base.

TABLEAU 6.6 – Résultats des régressions polynomiales en fonction des valeurs des OS des observateurs sur la base de test.

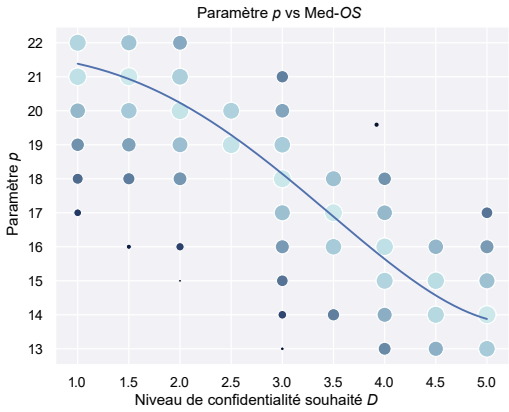
	Scores		
Métriques de régression	OS	Med-OS	MOS
R^2 *score*	0.6728	0.7293	**0.7443**
Explained Variance Score	0.6792	0.7383	**0.7508**
Mean Absolute Error	1.2379	1.1304	**1.0745**
Mean Squared Error	2.6915	2.2336	**2.1094**
Max Error	6.4270	5.2334	**5.2846**
Median Absolute Error	1.0393	0.8766	**0.7303**

Le tableau 6.6 présente les résultats des modèles de prédiction du paramètre p en fonction des valeurs des OS pour la base de test. Nous observons une baisse des scores pendant la phase de test. En effet, les meilleurs résultats obtenus, avec les valeurs moyennes, sont de 0.7443 contre 0.865 pendant la phase d'entraînement pour le score R^2 et le score de variance expliquée est d'environ 0.7508 contre 0.865 pendant la phase d'entraînement. Ces résultats montrent la robustesse du modèle utilisant les valeurs des MOS pour l'apprentissage. Ainsi, face aux données de la base de test, les modèles basés sur les valeurs des OS ou les valeurs des Med-OS, baissent respectivement de 0.7846 à 0.6728 et de 0.8442 à 0.7293.

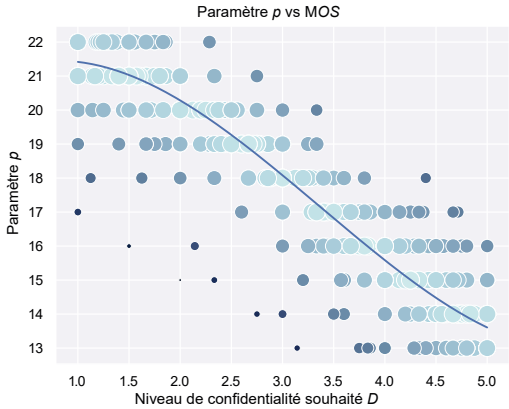
Dans la figure 6.17, nous présentons les régressions polynomiales pour la prédiction de p en fonction du niveau de confidentialité souhaité et des valeurs des MOS des observateurs utilisées pour générer le modèle, à savoir avec les valeurs brutes des MOS obtenus lors des évaluations subjectives en figure 6.17.a, avec les valeurs médianes par objet 3D



(a) OS



(b) Med-OS



(c) MOS

FIGURE 6.17 – Régressions polynomiales pour la prédiction du paramètre de chiffrement p.

figure 6.17.b et avec les valeurs moyennes par objet 3D en figure 6.17.c. La courbe bleue représente le polynôme obtenu suite à la régression polynomiale pour chaque modèle.

La taille des marqueurs et leur couleur correspondent à l'erreur de prédiction absolue (ou *Absolute Error*), ainsi plus le rond est petit et foncé, plus l'erreur est élevée. Nous pouvons constater visuellement que le modèle basé sur les scores moyens semble être celui avec le plus d'erreurs, mais en réalité il s'agit de celui qui minimise le mieux les erreurs comme nous l'avons observé dans le tableau 6.6.

TABLEAU 6.7 – Polynômes obtenus pour la prédiction de p à partir d'un niveau de confidentialité souhaité $\mathcal{D}$ après régression polynomiale des valeurs des scores d'opinion (OS, Med-OS, MOS) obtenues durant les évaluations subjectives, $p = f(\mathcal{D})$.

Scores observés	$p = f(\mathcal{D})$
MOS	$20.5429 + \begin{bmatrix}\mathcal{D} & \mathcal{D}^2 & \mathcal{D}^3\end{bmatrix} \times \begin{bmatrix}2.2295 \\ -1.4998 \\ 0.1590\end{bmatrix}$
Med-OS	$20.6683 + \begin{bmatrix}\mathcal{D} & \mathcal{D}^2 & \mathcal{D}^3\end{bmatrix} \times \begin{bmatrix}2.1686 \\ -1.4359 \\ 0.1459\end{bmatrix}$
MOS	$21.0404 + \begin{bmatrix}\mathcal{D} & \mathcal{D}^2 & \mathcal{D}^3\end{bmatrix} \times \begin{bmatrix}1.6931 \\ -1.2442 \\ 0.1212\end{bmatrix}$

Le tableau 6.7 présente les coefficients des trois polynômes obtenus par régression. Ainsi, grâce à nos évaluations subjectives nous avons pu établir un modèle pour la prédiction du paramètre de chiffrement p en fonction d'un niveau de confidentialité souhaité. Nous notons que le meilleur modèle pour la prédiction de p est obtenu avec un modèle utilisant les valeurs moyennes grâce aux scores élevés pour R^2 et la variance expliquée, mais également avec les faibles erreurs comparés aux valeurs des OS et aux valeurs des Med-OS présentées dans le tableau 6.6.

TABLEAU 6.8 – Prédictions du paramètre p en fonction du niveau de confidentialité souhaité pour l'objet 3D #18 illustré en figure 6.8, où n est le nombre d'évaluations.

		Scores observés			Paramètre p (prédit)		
p	n	OS	Med-OS	MOS	$f_{\text{brut}}(\text{D})$	$f_{\text{médian}}(\text{D})$	$f_{\text{moyen}}(\text{D})$
13	1	5	5.0	5.0	14.07	13.86	13.55
14	5	5	5.0	5.0	14.07	13.86	13.55
15	7	⟦4 ; 5⟧	5.0	4.7	[14.06; 15.64]	13.86	14.06
16	9	4	4.0	4.0	15.64	15.71	15.66
17	6	⟦3 ; 4⟧	4.0	3.6	[15.64; 18.03]	15.71	16.49
18	9	⟦2 ; 4⟧	3.0	3.2	[15.64; 20.07]	18.20	17.63
19	6	⟦2 ; 4⟧	2.5	2.6	[15.64; 20.07]	19.40	19.00
20	1	1	1.0	1.0	21.43	21.54	21.61
21	3	1	1.0	1.0	21.43	21.54	21.61
22	5	1	1.0	1.0	21.43	21.54	21.61

Le tableau 6.8 compare les résultats obtenus pour les prédictions du paramètre p avec le modèle $f(.)$ en fonction des valeurs données à l'apprentissage pour l'objet 3D #18 de la base VCD et ses dix versions chiffrés avec p allant de 13 à 22 comme illustré en figure 6.8. Nous constatons que nous arrivons à prédire un paramètre de chiffrement p relativement

proche de celui utilisé durant la construction de la base de données à partir du niveau de confidentialité souhaité. Les différentes prédictions donnent des valeurs légèrement inférieures pour p lorsque le niveau de confidentialité souhaité est supérieur ou égal à 3. À partir d'un niveau de confidentialité souhaité de 2, les paramètres prédits sont supérieurs à la valeur de chiffrement attendue. Ainsi, nos modèles arrivent donc à proposer des valeurs adaptées pour le paramètre de chiffrement, notamment lorsque le score de confidentialité souhaité est inférieur ou égal à 2. En effet, la valeur prédite de p est plus grande que la valeur attendue.

Estimation d'un score de confidentialité à partir de métriques objectives par régression polynomiale

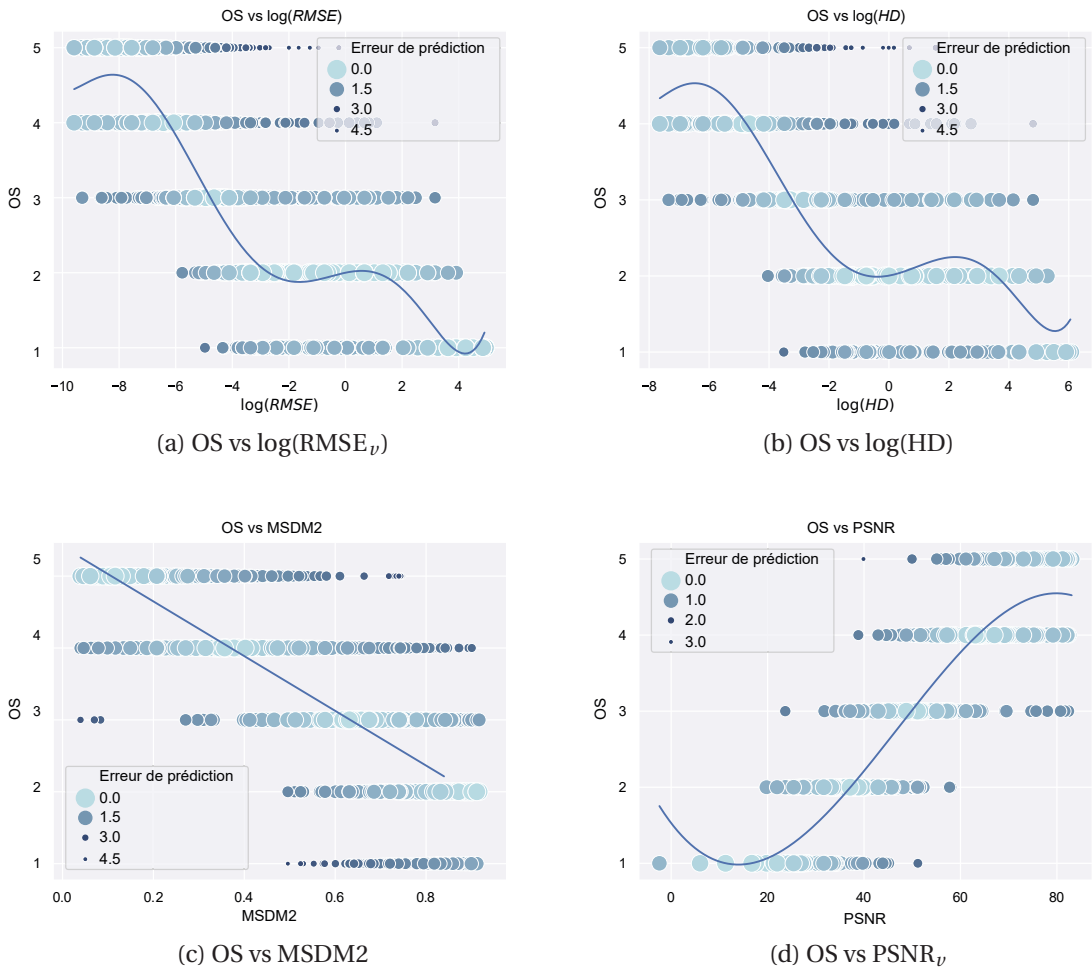
Nous venons de montrer qu'il était possible de prédire le paramètre p à partir d'un niveau de confidentialité (D) souhaité. Nous pouvons donc automatiquement proposer une valeur pour ce paramètre de chiffrement comme illustré en figure 6.4 dans la section 6.3. Toujours à partir de nos évaluations subjectives, nous pouvons construire dès à présent une métrique pour estimer le niveau de confidentialité des objets 3D sélectivement chiffrés. Chaque objet 3D de la base de données a été étudié à l'aide des métriques objectives présentées en section 1.2.2 du chapitre 1. Chaque objet a été ainsi comparé à son objet 3D de référence. De ce fait, nous analysons l'efficacité des métriques pour étudier la confidentialité visuelle des objets 3D sélectivement chiffrés.

TABLEAU 6.9 – Coefficients de corrélation entre les valeurs des OS des observateurs et les métriques objectives présentées en section 1.2.2 du chapitre 1.

		Métriques objectives				
Scores observés	Coefficients de corrélation	$RMSE_v$	HD	DAME	MSDM2	$PSNR_v$
OS	Pearson	-0.215	-0.224	0.084	-0.703	0.891
	Spearman	-0.761	-0.742	0.094	-0.699	0.896
Med-OS	Pearson	-0.238	-0.247	-0.035	-0.735	0.923
	Spearman	-0.814	-0.773	-0.116	-0.726	0.930
MOS	Pearson	-0.241	-0.250	-0.035	-0.739	0.939
	Spearman	-0.821	-0.779	-0.119	-0.735	0.942

Le tableau 6.9 présente les coefficients de corrélation calculés pour chaque métrique objective utilisée. Comme précédemment, nous avons choisi d'utiliser les résultats des évaluations de trois manières différentes, à savoir en utilisant directement toutes les valeurs brutes données par les observateurs, en utilisant les valeurs médianes pour chaque objet 3D et enfin en utilisant les valeurs moyennes pour chaque objet 3D. La première approche utilise l'ensemble des valeurs brutes (2437 évaluations) ce qui permet de calculer une valeur au plus proche de la réalité, mais avec du bruit. Les deux autres approches (médiane et moyenne) utilisent les valeurs attribuées aux 500 objets 3D sélectivement chiffrés. Nous présentons la corrélation des valeurs brutes, des valeurs médianes et des valeurs moyennes avec les métriques $RMSE_v$, HD, DAME, MSDM2 et $PSNR_v$. Nous constatons clairement dans le tableau 6.9 que la métrique $PSNR_v$ possède la plus forte corrélation avec les données quelque soit le type des valeurs utilisées. La métrique MSDM2 possède également des coefficients de corrélation intéressants. Les métriques $RMSE_v$ et HD n'ont pas un bon coefficient de corrélation de Pearson, cela signifie qu'il n'existe pas

de relation linéaire entre les scores et ces métriques. Cependant, la corrélation de Spearman donne de bons résultats (supérieurs à 0.70 en valeur absolue) du fait de la relation non-linéaire entre les scores et ces métriques. Seule la métrique DAME est totalement indépendante des scores donnés par les observateurs. Pour rappel, comme décrit en section 1.2.2 du chapitre 1, cette métrique est basée sur la moyenne des différences des angles dièdres pondérées par l'aire des triangles. Or, nous pouvons supposer que les sommets voyant leur position géométrique varier grandement, il en est de même pour l'aire des triangles formés par ces sommets. Ainsi, certains triangles deviennent très larges, tandis que d'autres se réduisent, ceci faussant ainsi les résultats donnés par cette métrique. Généralement, nous observons également que les valeurs moyennes des observations sont les plus facilement corrélées aux métriques dans le tableau 6.9, certainement à cause de l'effet de filtrage dû à la moyenne. Nous considérons dès lors qu'il est plus intéressant de s'attarder sur les métriques MSDM2 et PSNR_v, ainsi que sur les logarithmes des métriques RMSE_v et HD, soit $\log(\text{RMSE}_v)$ et log(HD).



(a) OS vs $\log(\text{RMSE}_v)$

(b) OS vs log(HD)

(c) OS vs MSDM2

(d) OS vs PSNR_v

FIGURE 6.18 – Distribution des OS obtenus par les observateurs en fonction des valeurs des métriques $\log(\text{RMSE}_v)$, log(HD), MSDM2 et PSNR_v.

La figure 6.18 représente les valeurs des métriques ($\log(\text{RMSE}_v)$, log(HD), MSDM2 et PSNR_v) en fonction de l'ensemble des valeurs des OS obtenues des observateurs lors des campagnes d'évaluations subjectives de la base VCD. Nous constatons que les métriques PSNR_v et MSDM2 semblent visuellement avoir une bonne relation polynomiale avec les évaluations subjectives. Les valeurs des métriques $\log(\text{RMSE}_v)$ et log(HD) sont bien trop

étalées pour une même valeur de OS ce qui rend difficile de trouver un polynôme adapté. Cependant, l'utilisation de cet ensemble non filtré de scores entraîne forcément la présence d'un bruit, sans parler des possibles valeurs aberrantes pouvant apparaître lors d'une mauvaise saisie de OS.

TABLEAU 6.10 – Résultats des régressions polynomiales des métriques sélectionnées pour les valeurs des OS.

(a) Phase d'entraînement

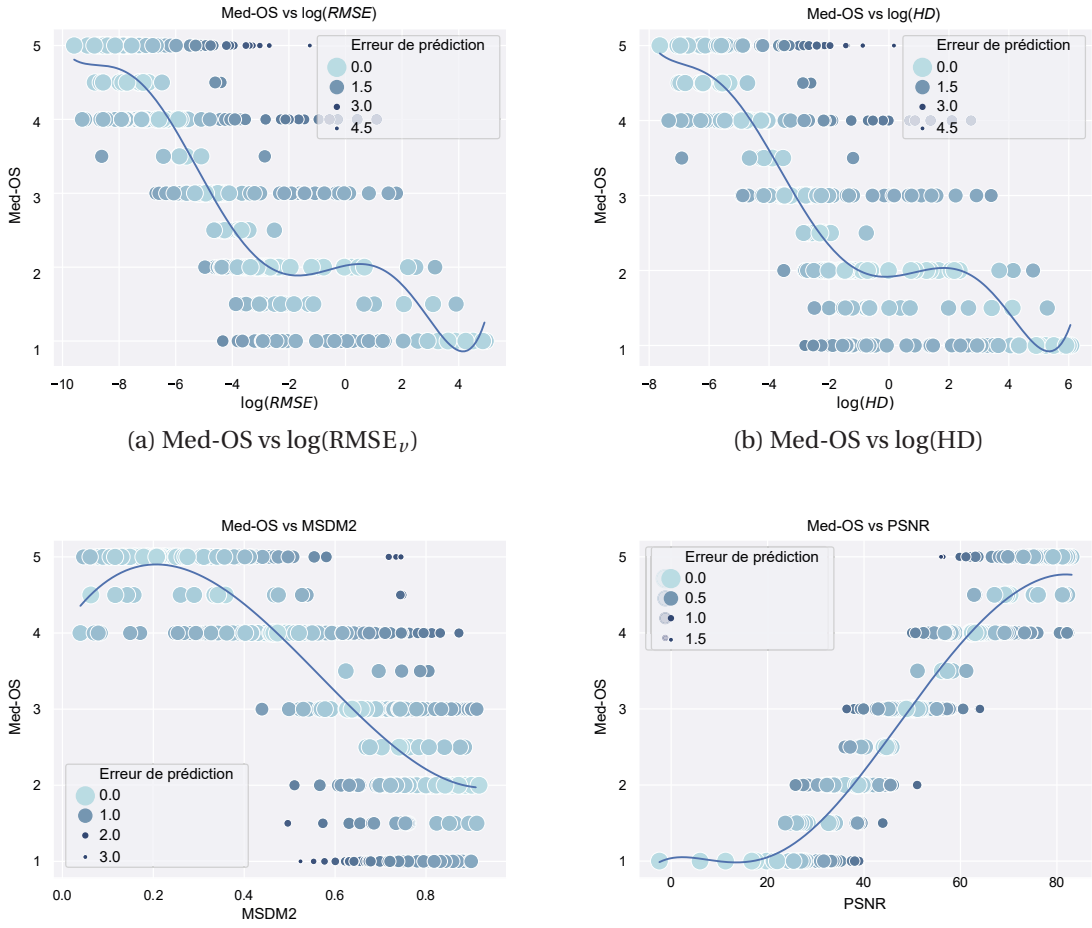
	Métriques 3D			
Métriques de régression	log(RMSE$_v$)	log(HD)	MSDM2	PSNR
R^2 *score*	0.7565	0.6853	0.5283	**0.7638**
Explained Variance Score	0.7653	0.6944	0.5283	**0.7639**
Mean Absolute Error	**0.5155**	0.5909	0.7278	0.5273
Mean Squared Error	0.4153	0.5259	0.7772	**0.3891**
Max Error	2.9112	2.8852	2.3386	**2.1092**
Median Absolute Error	**0.3980**	0.5038	0.6308	0.4880

(b) Phase de test

	Métriques 3D			
Métriques de régression	log(RMSE$_v$)	log(HD)	MSDM2	PSNR
R^2 *score*	0.3372	0.1170	0.2473	**0.7284**
Explained Variance Score	0.4048	0.2297	0.2486	**0.7284**
Mean Absolute Error	0.9509	1.0075	0.9699	**0.5735**
Mean Squared Error	1.4268	1.5638	1.3874	**0.5007**
Max Error	3.1232	3.0069	**2.5222**	2.7952
Median Absolute Error	0.9027	0.9319	0.9154	**0.4987**

Cette analyse est confirmée avec les résultats obtenus dans le tableau 6.10. L'ensemble des scores de qualité (R^2 et *Explained Variance Score*) baissent, tandis que les erreurs augmentent drastiquement. Néanmoins, il n'y a pas de modifications dans l'ordre d'importance des métriques. En effet, le PSNR$_v$ et le MSDM2 sont toujours les meilleures métriques possibles. Notons que le score R^2 des log(RMSE$_v$) et log(HD) est très faible avec respectivement 0.3372 et 0.1170 signifiant que ces métriques ne sont pas efficaces du tout pour construire notre métrique de confidentialité.

Nous montrons dans la figure 6.19 la même analyse avec les valeurs médianes associées à chaque objet 3D de la base de données. Comme pour la figure 6.18, nous constatons visuellement que la métrique PSNR$_v$ et le MSDM2 possèdent respectivement une relation linéaire et une relation polynomiale avec les données. Pour les logarithmes des métriques RMSE$_v$ et HD, nous observons qu'il est très difficile de trouver une solution qui réduise les erreurs d'estimation. Le tableau 6.11 présente les résultats des métriques de régression pour les métriques 3D et les valeurs des Med-OS des observations pour les phases d'entraînement et de test. Pour l'étape d'entraînement, nous constatons une augmentation des erreurs et une baisse des scores évaluant la qualité des estimations par rapport au tableau 6.12 Ainsi, la métrique PSNR$_v$ semble être la mesure la plus efficace pour estimer le score d'opinion et par extension le niveau de confidentialité. Tandis que la métrique MSDM2 reste celle qui minimise le mieux les erreurs de estimation. Les logarithmes des métriques RMSE$_v$ et HD sont finalement les moins efficaces pour cette tâche.



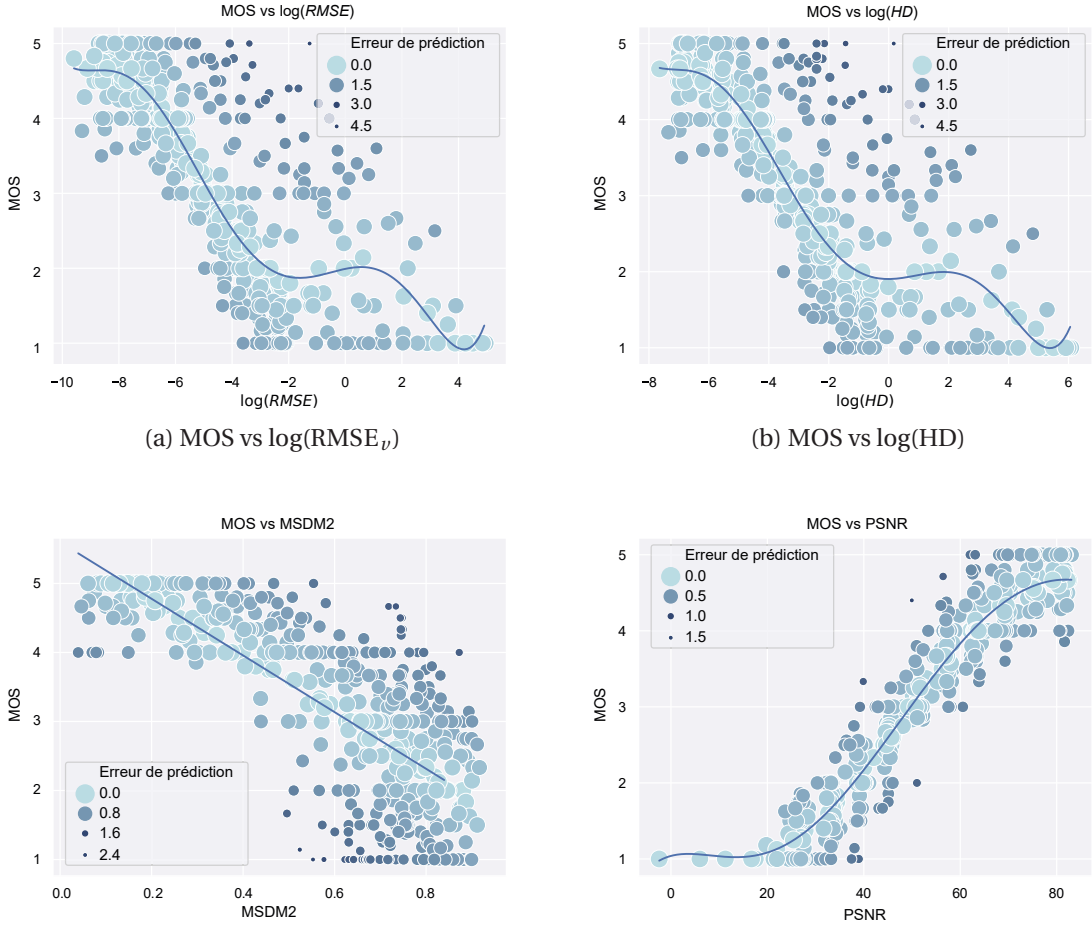
(a) Med-OS vs log(RMSE$_v$)

(b) Med-OS vs log(HD)

(c) Med-OS vs MSDM2

(d) Med-OS vs PSNR$_v$

FIGURE 6.19 – Distribution des valeurs des Med-OS pour chaque objet 3D chiffré en fonction des valeurs des métriques log(RMSE$_v$), log(HD), MSDM2 et PSNR$_v$.



(a) MOS vs log(RMSE$_v$)

(b) MOS vs log(HD)

(c) MOS vs MSDM2

(d) MOS vs PSNR$_v$

FIGURE 6.20 – Distribution des valeurs des MOS pour chaque objet 3D chiffré en fonction des valeurs des métriques log(RMSE$_v$), log(HD), MSDM2 et PSNR$_v$.

TABLEAU 6.11 – Résultats des régressions polynomiales des métriques sélectionnées pour les valeurs médianes des Med-OS.

(a) Phase d'entraînement

	Métriques 3D			
Métriques de régression	log(RMSE$_v$)	log(HD)	MSDM2	PSNR
R^2 *score*	0.8029	0.8034	0.6659	**0.8933**
Explained Variance Score	0.8125	0.8135	0.6667	**0.89335**
Mean Absolute Error	0.4418	0.4389	0.5990	**0.3500**
Mean Squared Error	0.3304	0.3295	0.5522	**0.1764**
Max Error	2.9074	2.8987	1.8892	**1.1603**
Median Absolute Error	0.3301	0.3198	0.5022	**0.3076**

(b) Phase de test

	Métriques 3D			
Métriques de régression	log(RMSE$_v$)	log(HD)	MSDM2	PSNR
R^2 *score*	0.3980	0.2839	0.3892	**0.8801**
Explained Variance Score	0.4591	0.3832	0.3892	**0.8801**
Mean Absolute Error	0.8982	0.9792	0.9005	**0.4124**
Mean Squared Error	1.2996	1.5459	1.3028	**0.2572**
Max Error	3.0991	3.0775	2.6963	**1.4555**
Median Absolute Error	0.8030	0.9162	0.8242	**0.3514**

La figure 6.20 illustre les valeurs des MOS pour chaque objet 3D de la base de données. Nous constatons visuellement que la métrique PSNR$_v$ et le MOS possèdent une relation très linéaire, alors que la métrique MSDM2 possède une relation plus polynomiale. Pour log(RMSE$_v$) et log(HD) que nous notons log(RMSE$_v$) et log(HD), nous constatons que les MOS varient de manière linéaire.

Dans le tableau 6.12, nous présentons les différents scores des métriques des régressions polynomiales entre les métriques log(RMSE$_v$), log(HD), PSNR et MSDM2 et les valeurs moyennes pour la phase d'entraînement et la phase de test. Pour la phase d'entraînement illustrée dans le tableau 6.12.a, nous remarquons que le PSNR obtient les meilleurs scores pour R^2 avec 0.9194 et pour la variance expliquée avec 0.9195. Nous constatons également que les log(RMSE$_v$) et log(HD) ont des scores R^2 aussi supérieurs à 0.80. Tandis que le MSDM2 obtient des scores corrects pour R^2 avec 0.6388 et pour la variance expliquée avec 0.6311. Cependant, comme nous pouvons le voir dans le tableau 6.12.b récapitulant les scores pour les métriques pour la phase de test, nous constatons une chute des scores pour log(RMSE$_v$), log(HD) et MSDM2 passant ainsi en dessous de 0.40 pour le score R^2. Seul le PSNR conserve un score R^2 élevé avec 0.9041. Ainsi, le PSNR est donc le plus efficace pour estimer le niveau de confidentialité d'un objet 3D sélectivement chiffré. Les résultats sont légèrement supérieurs à ceux obtenus pour les valeurs des Med-OS.

Au final, les meilleurs scores pour les métriques de régression et les plus faibles erreurs d'estimation ont été obtenus à partir des valeurs moyennes des MOS assignés à chaque objet 3D. Les métriques qui se sont révélées les plus efficaces pour estimer le niveau de confidentialité sont le PSNR$_v$ et le MSDM2 dans une moindre mesure. Étant donné l'écart entre les log RMSE$_v$ et log HD pour les phases d'entraînement et de test, il apparaît évident que ces dernières apportent peu d'information utile pour estimer le niveau de confiden-

TABLEAU 6.12 – Résultats des régressions polynomiales des métriques sélectionnées pour les valeurs moyennes des MOS.

(a) Phase d'entraînement

	Métriques 3D			
Métriques de régression	$\log(\mathrm{RMSE}_v)$	log(HD)	MSDM2	PSNR
R^2 *score*	0.8282	0.8284	0.6385	**0.9194**
Explained Variance Score	0.8382	0.8390	0.6392	**0.9195**
Mean Absolute Error	0.3850	0.3855	0.6132	**0.2795**
Mean Squared Error	0.2668	0.2665	0.5532	**0.1234**
Max Error	2.1102	2.0982	1.8354	**1.1275**
Median Absolute Error	0.3210	0.3227	0.5625	**0.2373**

(b) Phase de test

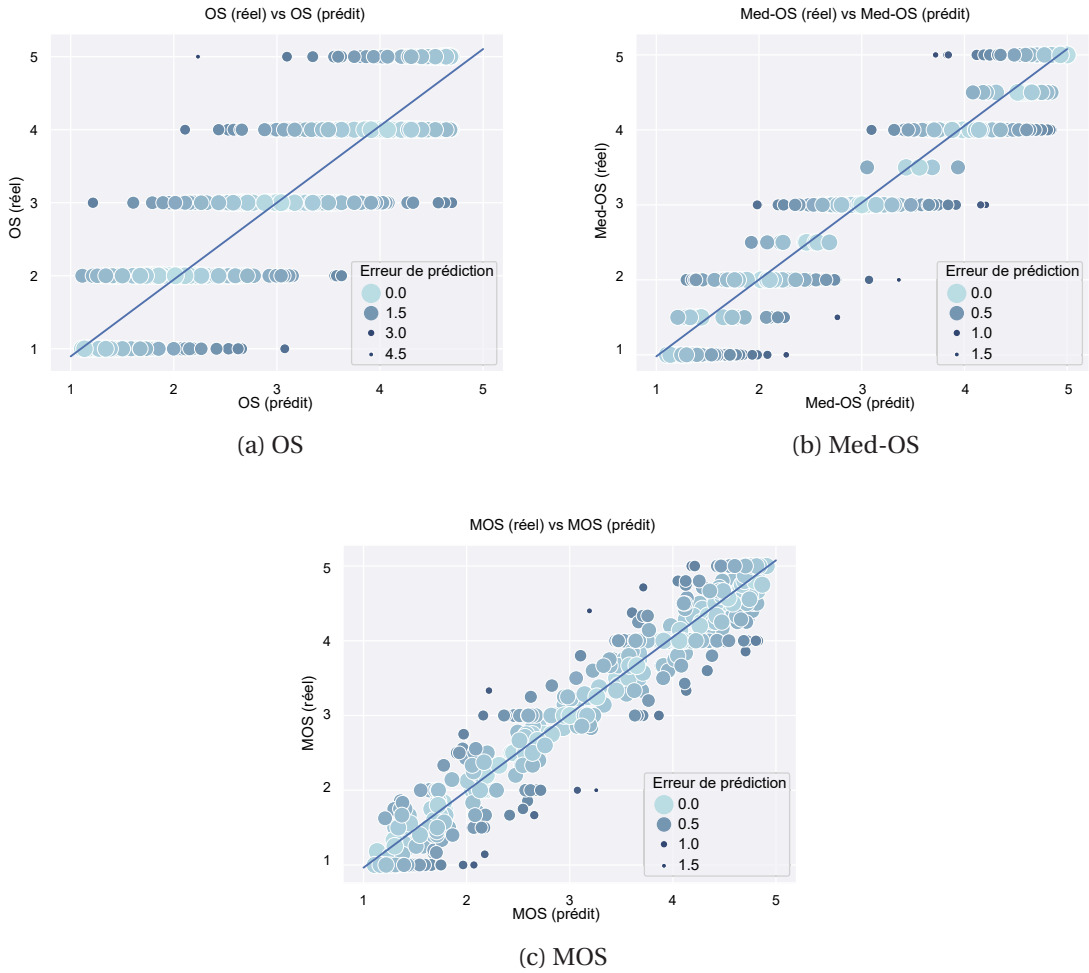
	Métriques 3D			
Métriques de régression	$\log(\mathrm{RMSE}_v)$	log(HD)	MSDM2	PSNR
R^2 *score*	0.3893	0.2720	0.3685	**0.9041**
Explained Variance Score	0.4656	0.3925	0.3767	**0.9042**
Mean Absolute Error	0.8279	0.9203	0.9111	**0.3341**
Mean Squared Error	1.1650	1.3888	1.2178	**0.1850**
Max Error	3.1190	3.0939	2.3294	**1.3767**
Median Absolute Error	0.6689	0.8035	0.8446	**0.2740**

tialité d'un objet 3D. À partir de ces résultats, nous construisons les estimations en fonction des métriques, puis nous calculons pour chaque objet 3D de la base VCD le niveau de confidentialité estimé. Une nouvelle régression polynomiale est alors entraînée à partir des valeurs de OS estimés par les métriques pour la base d'entraînement.

Dans la figure 6.21, nous montrons les résultats des estimations du niveau de confidentialité des objets 3D de la base de test (20 objets 3D de référence et leurs 200 variations) selon une régression polynomiale basée sur les estimations des métriques $\log\mathrm{RMSE}_v$ et $\log\mathrm{HD}$, PSNR_v et MSDM2. Visuellement, les niveaux de confidentialité estimés semblent bien correspondre aux valeurs de OS des observateurs. Ce modèle d'estimation, noté g_{All}, peut se formuler de la manière suivante :

$$\begin{cases} g_{\log(\mathrm{RMSE}_v)} = g_{\log(\mathrm{RMSE}_v)}(\log(\mathrm{RMSE}_v(\mathcal{M},\mathcal{M}'))), \\ g_{\log(\mathrm{HD})} = g_{\log(\mathrm{HD})}(\log(\mathrm{HD}(\mathcal{M},\mathcal{M}'))), \\ g_{\mathrm{MSDM2}} = g_{\mathrm{MSDM2}}(\mathrm{MSDM2}(\mathcal{M},\mathcal{M}')), \\ g_{\mathrm{PSNR}_v} = g_{\mathrm{PSNR}_v}(\mathrm{PSNR}_v(\mathcal{M},\mathcal{M}')), \\ g_{All}(\mathcal{M},\mathcal{M}') = \alpha + \beta \times g_{\log(\mathrm{RMSE}_v)} + \gamma \times g_{\log(\mathrm{HD})} + \delta \times g_{\mathrm{MSDM2}} + \eta \times g_{\mathrm{PSNR}_v}, \\ \alpha = -0.06441, \\ \beta = 0.1657, \\ \gamma = -0.2301, \\ \delta = 0.1147, \\ \eta = 0.9778. \end{cases} \tag{6.6}$$

La figure 6.22 présente les résultats des estimations du niveau de confidentialité des objets 3D de la base de test (20 objets 3D de référence et leurs 200 variations) selon une

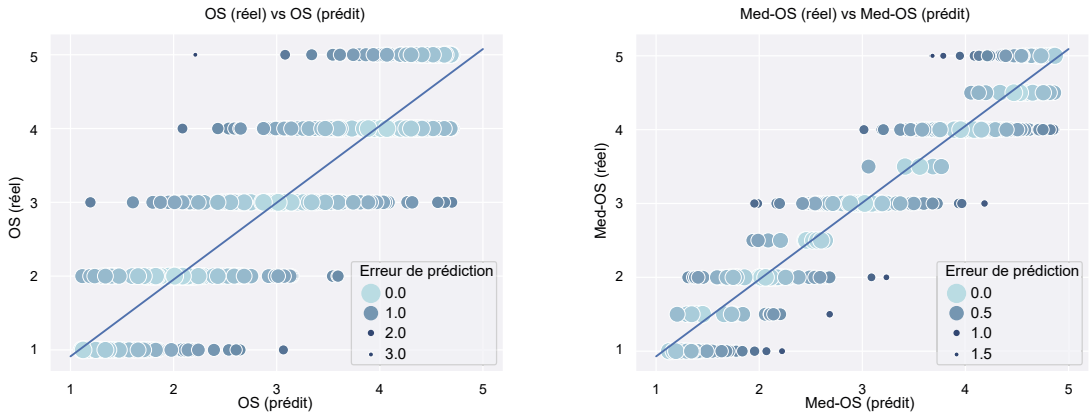


(a) OS

(b) Med-OS

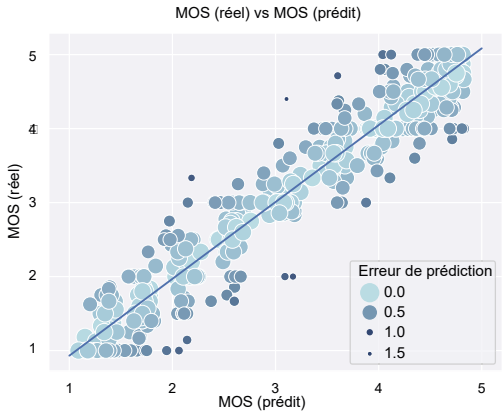
(c) MOS

FIGURE 6.21 – Estimation des niveaux de confidentialité à partir d'une régression polynomiale basée sur les estimations des métriques $\log(\mathrm{RMSE}_v)$ et $\log(\mathrm{HD})$, PSNR_v et MSDM2.



(a) OS

(b) Med-OS



(c) MOS

FIGURE 6.22 – Estimation des niveaux de confidentialité à partir d'une régression polynomiale basée sur les estimations des métriques PSNR_ν et MSDM2.

régression polynomiale basée sur les estimations des métriques PSNR_v et MSDM2. Visuellement, les niveaux de confidentialité estimés semblent bien correspondre aux valeurs de OS des observateurs. Ainsi, ce second modèle d'estimation, noté $g_{\text{PSNR}_v,\text{MSDM2}}$, réutilisant les estimations des OS donnés par les métriques PSNR_v et MSDM2 s'écrit :

$$\begin{cases} g_{\text{PSNR}_v,\text{MSDM2}}(\mathcal{M},\mathcal{M}') = \alpha + \times\beta g_{\text{MSDM2}} + \gamma \times g_{\text{PSNR}_v}, \\ \alpha = -0.1076, \\ \beta = 0.1087, \\ \gamma = 0.9298. \end{cases} \tag{6.7}$$

Statistiquement, il existe tout de même des erreurs estimations comme illustré dans le tableau 6.13.

TABLEAU 6.13 – Résultats des métriques de régression sur nos régressions polynomiales entre les valeurs de OS des observateurs et les valeurs estimées par les métriques.

(a) Phase d'entraînement

	g_{All}			$g_{\text{PSNR}_v,\text{MSDM2}}$		
Métriques de régression	OS	Med-OS	MOS	OS	Med-OS	MOS
R^2 *score*	0.7662	0.8975	**0.9224**	0.7662	0.8971	**0.9222**
Explained Variance Score	0.7662	0.8975	**0.9224**	0.7662	0.8971	**0.9222**
Mean Absolute Error	0.5228	0.3380	**0.2716**	0.5228	0.3371	**0.2714**
Mean Squared Error	0.3851	0.2928	**0.1188**	0.3852	0.1701	**0.1192**
Max Error	2.0760	1.418	**1.0740**	2.0648	1.1849	**1.0913**
Median Absolute Error	0.4744	0.4076	**0.3148**	0.4740	0.2938	**0.2193**

(b) Phase de test

	g_{All}			$g_{\text{PSNR}_v,\text{MSDM2}}$		
Métriques de régression	OS	Med-OS	MOS	OS	Med-OS	MOS
R^2 *score*	0.7267	0.8794	**0.9040**	0.7276	0.8827	**0.9046**
Explained Variance Score	0.7267	0.8866	**0.9068**	0.7277	0.8847	**0.9052**
Mean Absolute Error	0.5783	0.4148	**0.3450**	0.5228	0.4113	**0.1840**
Mean Squared Error	0.5038	0.2572	**0.1852**	0.5021	0.2502	**0.1750**
Max Error	2.7657	1.3610	**1.2570**	2.7892	1.3306	**1.2935**
Median Absolute Error	0.5010	0.3554	**0.2844**	0.5000	0.3410	**0.2600**

Dans le tableau 6.13, nous récapitulons les résultats des différentes métriques de régression pour notre fonction d'estimation g_{All} utilisant les métriques $\log(\text{RMSE}_v)$, $\log(\text{HD})$, MSDM2 et PSNR_v, ainsi qu'une fonction d'estimation utilisant uniquement le PSNR_v et le MSDM2. Tout d'abord, nous remarquons que l'estimation du niveau de confidentialité par toutes les métriques atteint un score R^2 de 0.9224 pour la phase d'entraînement contre 0.9040 pour la phase de test lors de l'utilisation des valeurs moyennes. Notons également qu'en utilisant uniquement les métriques PSNR_v et MSDM2, nous sommes capables de produire de meilleures estimations avec un score R^2 et un score de variance expliquée de 0.9046. Dans le tableau 6.13, nous analysons également les résultats obtenus en combinant les estimations des deux métriques afin de profiter de la forte corrélation linéaire du

PSNR_v avec la corrélation polynomiale du MSDM2. De très bons scores de R^2 sont obtenus en utilisant seulement ces deux métriques avec 0.9222 pour la phase d'entraînement et de 0.9046 pour la phase de test. Notons que l'erreur absolue moyenne est nettement plus faible avec cette autre fonction passant ainsi de 0.3450 avec toutes les estimations des métriques contre seulement 0.1840 pour la phase de test. Toujours dans la phase de test, l'erreur quadratique moyenne baisse également de 0.1852 à 0.1750. De même pour l'erreur médiane absolue qui passe de 0.2844 à 0.2600.

Dans cette partie, nous avons ainsi réussi à construire une métrique avec référence permettant d'estimer le niveau de confidentialité des objets 3D sélectivement chiffrés en fonction de leur objet 3D original. Nous choisissons le modèle d'estimation $g_{\text{PSNR}_v,\text{MSDM2}}$ comme base pour notre métrique en utilisant les valeurs moyennes, car il maximise le score R^2 tout en minimisant le plus d'erreurs. Nous nommons cette métrique *Visual Confidentiality Measure* (ou VCM), tel que :

$$\text{VCM}(\mathcal{M},\mathcal{M}') = \text{clip}(g_{\text{PSNR}_v,\text{MSDM2}}(\mathcal{M},\mathcal{M}'),1.0,5.0), \tag{6.8}$$

où la fonction clip(.) restreint la valeur de sortie entre les valeurs 1.0 et 5.0,

Cette métrique VCM permet, en utilisant une combinaison linéaire des métriques PSNR_v et MSDM2, d'estimer la confidentialité des objets 3D chiffrés selon des approches de chiffrement sélectif. Elle permet également d'améliorer l'approche de chiffrement sélectif proposée au chapitre 3 en estimant le niveau de confidentialité de l'objet 3D chiffré. Ainsi, elle permet de corriger le choix du paramètre de chiffrement p afin d'obtenir un chiffrement plus efficace pour le niveau de confidentialité souhaité comme présenté pour l'objet 3D #18 dans le tableau 6.14. La dernière colonne du tableau 6.14 compare les résultats obtenus en utilisant les quatre métriques. Les résultats sont, pour cet exemple, très similaires.

TABLEAU 6.14 – Estimations du niveau de confidentialité en fonction des estimations fournies à partir des métriques pour l'objet 3D #18 illustré en figure 6.8, où n est le nombre d'évaluations.

		Scores observés				
p	n	OS	Med-OS	MOS	$\text{VCM}(\mathcal{M},\mathcal{M}')$	g_{All}
13	1	5	5.0	5.0	4.684	4.676
14	5	5	5.0	5.0	4.556	4.544
15	7	$[\![4\,;\,5]\!]$	5.0	4.7	4.305	4.290
16	9	4	4.0	4.0	3.943	3.929
17	6	$[\![3\,;\,4]\!]$	4.0	3.6	3.503	3.482
18	9	$[\![2\,;\,4]\!]$	3.0	3.2	3.017	3.011
19	6	$[\![2\,;\,4]\!]$	2.5	2.6	2.527	2.547
20	1	1	1.0	1.0	2.063	2.070
21	3	1	1.0	1.0	1.654	1.661
22	5	1	1.0	1.0	1.341	1.332

6.3.4 Conclusion

Dans cette partie, nous avons créé une base de données pour l'évaluation subjective de la confidentialité des objets 3D sélectivement chiffrés. Cette base est constituée de 50 objets 3D de référence et de dix variations de chiffrement pour chacun d'entre eux selon la méthode proposée dans le chapitre 3 avec un paramètre de chiffrement $p \in [\![13\,;\,22]\!]$.

À partir des résultats de 54 observateurs, nous avons analysé les valeurs des scores d'opinion fournies par les observateurs et construit un modèle de prédiction des paramètres de chiffrement en fonction d'un niveau de confidentialité représentant le niveau de chiffrement souhaité par les utilisateurs ($p = f(\mathrm{D})$). De plus, à partir de ces mêmes résultats et à l'aide de métriques objectives, nous avons également construit une métrique de confidentialité visuelle des objets 3D en utilisant des approches d'apprentissage statistique permettant d'estimer le niveau de confidentialité appliqué à nos objets 3D sélectivement chiffrés ($\mathrm{MOS}' = \mathrm{VCM}(\mathcal{M}, \mathcal{M}')$). Dans des travaux futurs, nous souhaitons exploiter encore plus les résultats de nos évaluations subjectives en utilisant d'autres outils d'apprentissage supervisé ou non supervisé, ainsi que les caractéristiques des objets 3D. En effet, grâce à nos analyses, nous avons constaté que la représentation géométrique de l'objet 3D et sa connectivité ont un impact important sur le niveau de confidentialité des objets 3D sélectivement chiffrés. De ce fait, nous souhaitons prendre en compte les caractéristiques de ces objets 3D comme nous l'avons fait dans la section 6.2 de ce même chapitre.

6.4 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle approche d'amélioration des méthodes d'insertion de données cachées pour augmenter la résistance aux approches de stéganalyse. Dans une seconde partie de ce chapitre, nous avons développé une étude sur la confidentialité visuelle des objets 3D sélectivement chiffrés afin de concevoir une métrique de confidentialité, nommée VCM.

Dans notre première contribution, à partir des travaux de Itier et Puech [64] nous avons proposé, en collaboration avec l'université de York, une modification dans la manière d'insérer les données pour rendre plus imperceptible l'insertion face aux systèmes de stéganalyse. En effet, ces derniers se basent sur l'analyse des objets 3D marqués à partir d'un ensemble de caractéristiques 3D et leurs statistiques. Face au système de stéganalyse employé par Li [87], nous avons pu constater que la méthode proposée Itier et Puech [64] influence fortement les angles dièdres de l'objet 3D, mais qu'également le chemin hamiltonien construit pour l'insertion est principalement composé d'arêtes existantes dans le maillage. À partir de nos analyses, nous avons donc proposé de changer la méthode d'insertion dans les trois coordonnées sphériques par une insertion seulement dans la coordonnée radiale afin de réduire la modification des angles dièdres entre les triangles. Les résultats expérimentaux montrent une nette augmentation du taux d'erreur à la détection, passant de 18.88% à 31.12% en maintenant le même taux d'insertion de 24 bits par sommet. Dans des travaux futurs, nous souhaitons prendre plus en compte les caractéristiques utilisées en stéganalyse 3D pour assurer une insertion encore plus imperceptible. En effet, nous remarquons que cette problématique commence à être étudiée [162].

Dans notre deuxième contribution, nous avons proposé d'évaluer la confidentialité des objets 3D sélectivement chiffrés selon notre méthode proposée au chapitre 3 dans le but de créer une métrique de confidentialité. Les approches de chiffrement sélectif 3D produisent des objets 3D sélectivement chiffrés dans lesquels la forme ou le contenu peuvent être protégés à la visualisation par l'insertion de distorsions géométriques. Ainsi, les méthodes basées sur des approches de chiffrement sélectif comme présenté dans le chapitre 3, en section 4.3 du chapitre 4 et en section 5.3 du chapitre 5 génèrent des objets 3D pouvant avoir un niveau de chiffrement transparent, suffisant ou avec une confidentialité visuelle. Cependant, avec les métriques objectives il est difficile d'établir les seuils pivots de changement entre ces trois niveaux. De plus, selon les paramètres de chiffrement, la géométrie et la connectivité de l'objet 3D, les résultats peuvent varier signifi-

cativement. Nous avons donc construit une base de données de 550 objets 3D dont 500 sont sélectivement chiffrés et réalisé une campagne d'évaluations subjectives. À partir des résultats obtenus auprès de plus de 54 observateurs, nous avons réalisé des analyses statistiques afin de produire, par régression linéaire, un modèle d'estimation des paramètres de chiffrement en fonction du score de confidentialité souhaité pour l'objet 3D sélectivement chiffré. De plus, à partir de ces mêmes résultats nous avons construit, à l'aide d'une combinaison des métriques objectives avec référence, une nouvelle métrique de confidentialité visuelle pour les objets 3D sélectivement chiffrés. Dans des travaux futurs, nous souhaitons exploiter encore plus ces données d'évaluations subjectives en utilisant des approches basées apprentissage comme des réseaux de neurones, des réseaux de neurones convolutionnels (2D et 3D) et en tirant profit des caractéristiques inhérentes à ces objets 3D. Nous pourrions construire des métriques de confidentialité sans référence afin d'évaluer la confidentialité d'un objet 3D sélectivement chiffré sans connaissance de l'objet 3D original.

Ces travaux ont fait l'objet de deux publications dont une en cours de soumission. Tout d'abord, nos travaux sur l'amélioration de la résistance pour les méthodes d'insertion de données cachées 3D haute-capacité ont fait l'objet d'une publication dans la conférence internationale IEEE ICIP en 2017 [90]. Enfin nos travaux sur l'étude de la confidentialité visuelle des objets 3D sélectivement chiffrés sont en cours de soumission [18].

Conclusion et perspectives

Dans ce chapitre, nous présentons un bilan de nos travaux effectués durant cette thèse ainsi que des possibles pistes de recherches futures. Dans un premier temps, nous résumons le contenu du manuscrit et nous dressons un bilan sur les contributions réalisées. Enfin, dans un second temps nous présentons de nouveaux challenges à considérer ainsi que des premières idées pour y contribuer.

Conclusion

Dans cette thèse, nous avons proposé de nouvelles méthodes pour sécuriser des objets 3D. Dans l'état de l'art, nous avons présenté les différentes notions définissant les objets 3D, l'insertion de données cachées, le chiffrement sélectif et le partage de secret, et plus particulièrement le partage de secret multimédia pour les images et les objets 3D. Nous avons exploré également dans l'état de l'art des approches hiérarchiques pour le partage de secret. Nous avons ainsi réalisé un état de l'art des méthodes de référence à partir desquelles nos travaux ont été développés. Dans le cadre fixé par la société STRATEGIES, nous nous sommes intéressés au chiffrement sélectif dans le but de sécuriser les informations géométriques de l'objet 3D. Dans ce contexte, nous avons voulu, tout en préservant le format initial de l'objet 3D, autoriser la visualisation de certaines informations selon le niveau de confidentialité souhaité.

Notre première contribution a consisté à chiffrer sélectivement la géométrie de l'objet 3D et plus particulièrement une sélection de bits au sein de la représentation binaire des valeurs flottantes des coordonnées des sommets. Tout en préservant le format pour permettre la visualisation de l'objet 3D chiffré sélectivement au sein de scènes 3D, notre méthode génère des distorsions géométriques lors du chiffrement de l'objet 3D à partir d'une clé secrète $\mathcal{K}$ et un niveau de dégradation $\mathcal{D}$ choisis par l'utilisateur. Ainsi, comme nos résultats le montrent, cette solution permet de fournir plusieurs niveaux de chiffrement, à savoir la confidentialité visuelle, le chiffrement suffisant et le chiffrement transparent. La confidentialité visuelle consiste à protéger la forme et le contenu de l'objet 3D. Le chiffrement suffisant quant à lui a pour but de protéger le contenu de l'objet 3D, mais de laisser la forme reconnaissable. Enfin, le chiffrement transparent permet de rendre la forme et le contenu reconnaissables, tout en protégeant la haute qualité de l'objet 3D.

Dans notre deuxième contribution, nous nous sommes inspirés de nos travaux sur le chiffrement sélectif 3D pour développer de nouvelles approches dans le domaine du partage d'objet 3D secret. Dans un premier temps, nous avons proposé d'adapter la méthode de partage de secret de Blakley [20] pour les images en niveau de gris afin de pouvoir les partager entre 2 et 7 collaborateurs et permettre la reconstruction de l'image avec au moins 2 *shares*. Dans cette méthode, nous avons proposé un codage particulier pour les équations de droites, afin de représenter uniformément chaque niveau de gris (avec le même nombre de droites possibles par niveau de gris). Notre méthode évite l'expan-

sion de la taille des *shares* tout en conservant la taille de l'image secrète. Dans un second temps, toujours en nous inspirant de nos travaux sur le chiffrement sélectif et le partage d'image secrète, nous avons construit une nouvelle méthode de partage d'objet 3D secret. Cette méthode préserve à la fois le format d'entrée en fournissant des objets 3D comme *shares*, et propose la propriété de chiffrement sélectif dans le but de contrôler le niveau de dégradation. Au lieu de chiffrer une séquence de bits sélectionnés au sein des coordonnées des sommets de l'objet 3D secret, nous appliquons une méthode de partage de secret [20, 116] sur cette séquence et nous substituons les bits sélectionnés à ceux des séquences générées. Les objets 3D générés, nommés objets 3D partagés, sont ensuite distribués aux utilisateurs. Ces objets 3D partagés peuvent être vus comme des versions de basse qualité selon le niveau de dégradation souhaité. Lorsqu'un sous-ensemble de k parmi n utilisateurs rassemblent leurs objets 3D, alors il est possible de reconstruire l'objet 3D secret en haute qualité et sans perte. Conformément aux résultats expérimentaux que nous présentons, nous sommes capables d'offrir des objets 3D partagés selon un niveau de dégradation qui peut être contrôlé selon les cas d'utilisation. Notons que notre méthode fonctionne aussi bien avec la méthode de Shamir [116] que celle de Blakley [20]. Nous proposons ainsi, à notre connaissance, la première méthode de partage sélectif d'objet 3D.

Dans notre troisième contribution, nous étendons les travaux réalisés précédemment sur le partage d'objet 3D secret pour y ajouter des aspects hiérarchiques afin de prendre en compte le rôle des utilisateurs et leurs droits d'accès au contenu 3D. Tout d'abord, nous étudions une problématique de respect de la vie privée dans les images de réseaux sociaux. À cause du développement fulgurant des réseaux sociaux, les conflits pluripartites de vie privée deviennent de plus en plus fréquents. Nous proposons alors de répondre à cette problématique en utilisant le partage hiérarchique d'image secrète afin de protéger des régions d'intérêt au sein d'image, dans notre cas les visages des utilisateurs. Dès qu'un utilisateur donne son consentement pour la publication de son identité au sein de la photo, seuls son visage et ceux des autres utilisateurs ayant donné leur consentement sont visibles sur le média. Nos résultats montrent la faisabilité de notre méthode, et nos analyses statistiques valident la sécurité des régions d'intérêt. Enfin, nous proposons une méthode de partage sélectif et hiérarchique d'un objet 3D avec une hiérarchie dite à accès prioritaire. Ainsi, à chaque niveau de la hiérarchie est associé un seuil requis d'utilisateur. Dès lors que ce seuil est dépassé, l'objet 3D secret peut être reconstruit sans perte. L'approche proposée distribue ainsi des objets 3D partagés aux utilisateurs avec des clés secrètes en fonction du niveau dans lequel ces derniers se trouvent. Cette approche permet d'accélérer le processus de reconstruction en fonction des droits des utilisateurs. Sans ces clés, la hiérarchie fonctionne comme celle proposée par Belenkiy [7]. Les résultats présentés montrent la faisabilité de notre méthode. Nous en analysons ensuite la sécurité, notamment la résistance aux attaques selon des approches de filtrage en fonction du niveau de dégradation. Ainsi, notre méthode, pour des niveaux de dégradation élevés, se montre robuste à ces attaques et ne révèle rien de l'objet 3D secret. Tandis que pour des niveaux de dégradation plus faible, le contenu devient accessible, mais la haute qualité reste protégée.

Dans notre quatrième et dernière contribution, nous avons analysé les objets 3D produits par ces méthodes de protection. Ainsi, dans un premier temps nous avons proposé d'améliorer la robustesse face à la stéganalyse de la méthode d'insertion de données cachées haute-capacité proposée par Itier et Puech [64] réalisée durant la précédente thèse en collaboration avec la société STRATEGIES [63]. Cette méthode construit un chemin hamiltonien des points les plus proches tout en insérant le message sur les coordon-

nées sphériques entre deux sommets successifs dans le chemin. Nous avons proposé dans cette contribution d'analyser à l'aide d'un système de stéganalyse le taux de détection de présence de messages cachés au sein des objets 3D marqués par cette méthode. À partir de notre analyse, nous avons observé que la méthode d'insertion influence grandement les caractéristiques 3D des angles dièdres. Ainsi, nous avons soumis l'idée de modifier les modalités d'insertion en insérant uniquement sur la coordonnée radiale du système de coordonnées sphériques. Nos résultats expérimentaux montrent une augmentation non négligeable du taux d'erreur à la détection. Enfin, nous avons analysé les objets 3D sélectivement chiffrés. La littérature définit trois niveaux de chiffrement sélectif, à savoir la confidentialité visuelle, le chiffrement suffisant et le chiffrement transparent. Nos méthodes de chiffrement et de partage permettent de générer des objets 3D selon ces trois niveaux, cependant nous n'avions pas de mesure fiable afin d'évaluer la confidentialité de ces objets 3D. Nous avons ainsi proposé d'analyser, en construisant une base de données d'objets 3D sélectivement chiffrés, nommée *Visual Confidentiality Dataset* (VCD), d'étudier la confidentialité visuelle de ces objets 3D par une campagne d'évaluations subjectives afin d'obtenir des scores d'opinion moyenne (MOS). Grâce à ces évaluations subjectives, nous avons construit plusieurs modèles de régression polynomiale afin d'estimer les paramètres de chiffrement en fonction d'un niveau de confidentialité souhaité dans le but de simplifier les paramètres pour l'utilisateur. Enfin, nous avons construit, toujours à partir de ces résultats et des métriques objectives, une nouvelle métrique. Nous avons étudié ces résultats selon trois approches d'agrégation, à savoir les valeurs brutes où toutes les valeurs fournies par les observateurs sont utilisées, les valeurs médianes pour chaque objet 3D de la base et les valeurs moyennes pour chaque objet. Nous avons réussi à construire une métrique, avec référence et en utilisant des métriques objectives, permettant de prédire le niveau de confidentialité des objets 3D de manière précise.

En conclusion, dans ces travaux de recherche nous avons créé plusieurs méthodes afin de répondre à la problématique de sécurisation des objets 3D, cela pour divers scénarios d'utilisation. Nous avons également analysé la confidentialité des objets 3D sélectivement chiffrés. Bien que nous ayons évalué la confidentialité de ces objets 3D sélectivement chiffrés, il reste néanmoins certains points à améliorer au sein de nos méthodes et d'autres à approfondir. C'est pourquoi nous proposons des pistes afin de les améliorer.

Perspectives

Dans cette section, nous présentons des perspectives sur le chiffrement sélectif 3D, le partage d'objet 3D et l'analyse de la confidentialité visuelle qui, selon nous, pourraient être intéressantes à approfondir.

Durant ces travaux de thèse, nous nous sommes principalement concentrés sur le chiffrement sélectif 3D et le partage d'objet 3D. Nous nous sommes attardés durant la fin de la thèse sur l'analyse de la confidentialité des objets 3D sélectivement chiffrés. Contrairement à l'insertion de données cachées 3D, l'état de l'art sur le chiffrement sélectif 3D ou le partage d'objet 3D est succinct et mérite d'être approfondi. Combiner l'insertion de données cachées pour la traçabilité des objets et le chiffrement sélectif permettrait de construire de nouvelles approches homomorphiques appliquées au domaine de la sécurité multimédia. L'étude de la confidentialité est, à notre connaissance, inexistante dans le domaine. En effet, les études subjectives se concentrent principalement sur la qualité des objets 3D et très récemment sur des nuages de points.

Crypto-compression 3D

Nous pensons que le chiffrement sélectif 3D peut être étendu à d'autres formes de représentation des objets 3D et tout particulièrement pour les formats de compression de données 3D. En effet, avec l'augmentation très importante de la taille des données 3D, mais aussi avec l'évolution des technologies, en particulier avec l'arrivée de la réalité augmentée, la réalité virtuelle et l'impression 3D, il serait intéressant d'intégrer nos approches dans ces nouveaux formats de compression, souvent open-source, en profitant du fait que les données sont quantifiées. Contrairement aux valeurs flottantes, les données issues de quantification sont contraintes dans des intervalles connus liés au volume englobant de l'objet 3D. Cela permettrait d'appliquer un chiffrement sélectif de manière uniforme sur l'ensemble de l'objet 3D préservant ainsi le volume englobant ou bien de sélectionner différents intervalles à protéger selon différents niveaux de dégradation, tout en respectant les formats.

Décodage hiérarchique basé sur un trousseau de clés

Une autre proposition que nous énonçons dans ces perspectives serait de renverser le principe de chiffrement sélectif en offrant un décodage hiérarchique comme illustré en figure 6.23.

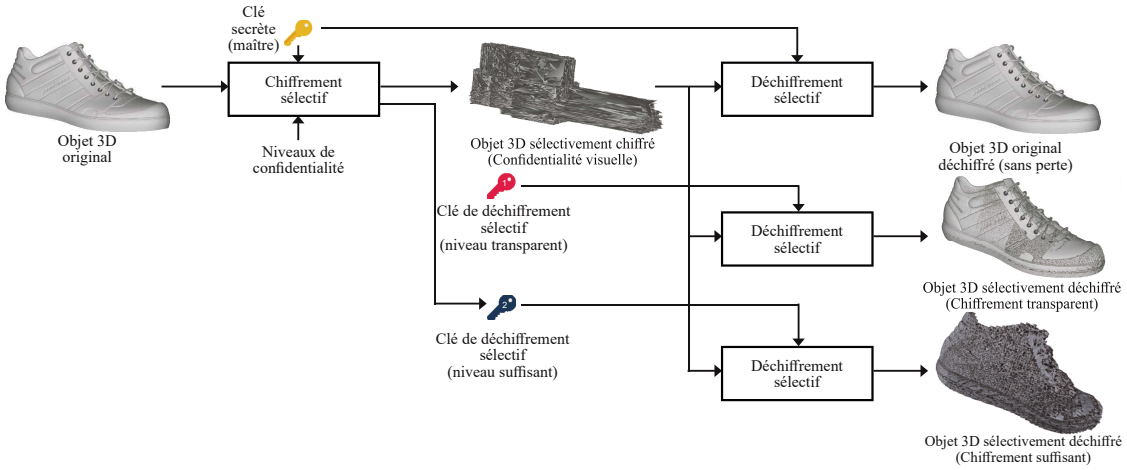


FIGURE 6.23 – Déchiffrement sélectif selon différentes clés secrètes.

L'objet 3D serait chiffré au plus fort niveau de chiffrement et, en fonction des clés secrètes générées à partir de la clé de chiffrement, le déchiffrement générerait des objets 3D à différents niveaux de qualité. Pour le partage hiérarchique d'objet 3D secret, nous pourrions simplifier notre méthode afin que les utilisateurs ne reçoivent qu'une seule et unique clé en fonction de leur niveau dans la hiérarchie à accès prioritaire au lieu d'un jeu de clés. En utilisant des fonctions de dérivation de clé nous pourrions à partir d'une clé principale dériver de nouvelles clés à distribuer aux utilisateurs. De plus, la clé principale pourrait être utilisée pour déchiffrer directement n'importe quel objet 3D partagé sans l'aide de collaborateur. Ainsi, nous pourrions proposer, à notre connaissance, la première méthode hybride de chiffrement sélectif 3D et de partage hiérarchique et progressif d'objet 3D secret. Enfin, une autre possibilité envisageable pour les approches de chiffrement sélectif serait de chiffrer différentes parties de l'objet 3D selon différents niveaux de dégradation. Ce chiffrement sélectif et localisé permettrait de protéger des parties critiques d'un objet 3D tout en laissant le reste chiffré de manière transparente ou suffisante. Pour

cela, il serait peut-être intéressant de segmenter l'objet 3D original et être capable de récupérer cette segmentation lorsque l'objet 3D est chiffré. Une solution envisageable serait d'employer des méthodes d'insertion de données cachées réversibles ou sans distorsion afin de marquer les sommets aux frontières des régions segmentées.

Métriques de confidentialité

L'évaluation de la confidentialité des objets 3D sélectivement chiffrés est,d'une certaine manière, à l'opposée de la notion d'évaluation de la qualité des objets 3D. En effet, en recherchant la confidentialité, nous souhaitons dans nos analyses associer les dégradations géométriques insérées par nos méthodes aux différents niveaux de chiffrement sélectif. Comme le domaine de l'évaluation de la confidentialité est, à notre connaissance, peu exploité, nous avons proposé de créer notre propre ensemble d'objets 3D sélectivement chiffrés et de réaliser des évaluations subjectives. Bien que les résultats obtenus soient intéressants, nous pensons que nous avons juste commencé à effleurer un nouveau domaine de recherche. Ainsi, une proposition que nous évoquons en perspective serait d'exploiter encore plus ces données d'évaluations subjectives en utilisant des approches basées apprentissage comme des réseaux de neurones, des réseaux de neurones convolutionnels (2D et 3D) tout en tirant profit des caractéristiques inhérentes à ces objets 3D.

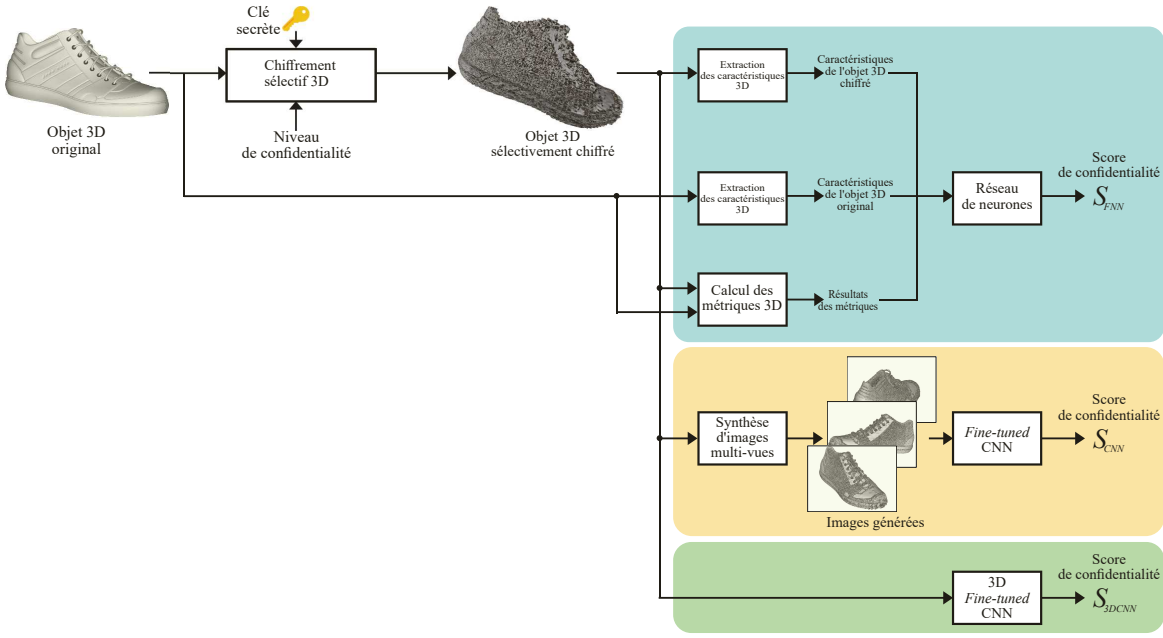


FIGURE 6.24 – Perspectives d'approches par apprentissage profond pour la réalisation de métriques pour la confidentialité des objets 3D sélectivement chiffrés.

Avec le développement des méthodes d'apprentissage profond pour l'image et les nuages de point, nous pourrions concevoir une métrique sans référence exploitant les informations visuelles de l'objet 3D sélectivement chiffré. Que cela soit sous forme d'images en utilisant plusieurs points de vue comme pour nos observateurs ou en exploitant directement le nuage de points comme illustré en figure 6.24. De plus, il serait intéressant également d'employer des réseaux de neurones spécifiquement dédiés aux objets 3D comme proposés par Qi *et al.* [106].

Liste des contributions

Conférences nationales :

— [12] S. Beugnon, W. Puech, J.-P. Pedeboy. "Chiffrement sélectif d'objet 3D". Colloque COmpression et REpréssentation des Signaux Audiovisuels (CORESA), Novembre 2017, Caen, France.

— [14] S. Beugnon, P. Puteaux, W. Puech. "Protection de la vie privée par partage hybride de photos sur les réseaux sociaux". Colloque COmpression et REprésentation des Signaux Audiovisuels (CORESA), Novembre 2018, Poitiers, France.

Conférences internationales :

— [90] Z. Li, S. Beugnon, W. Puech, A. Bors. "Rethinking the high capacity 3D steganography : Increasing its resistance to steganalysis". IEEE International Conference on Image Processing (IEEE ICIP), Septembre 2017, Beijing, China.

— [11] S. Beugnon, W. Puech, J.-P. Pedeboy. "An efficient lossless (2, n) secret image sharing based on Blakley's scheme". IEEE International Workshop on Multimedia Signal Processing (IEEE MMSP), Octobre 2017, Luton, U-K.

— [13] S. Beugnon, W. Puech, J.-P. Pedeboy. "From Visual Confidentiality To Transparent Format-Compliant Selective Encryption Of 3D Objects". IEEE International Conference on Multimedia and Expo Workshop (IEEE ICME Workshop), Juillet 2018, San-Diego, USA.

— [15] S. Beugnon, W. Puech, J.-P. Pedeboy. "A format-compliant selective secret 3d object sharing scheme based on shamir's scheme". IEEE International Conference on Acoustic Speech Signal Processing (IEEE ICASSP), Mai 2019, Brighton, U-K.

— [17] S. Beugnon, P. Puteaux, W. Puech. "Privacy protection for social media based on a hierarchical secret image sharing scheme". IEEE International Conference on Image Processing (IEEE ICIP), Septembre 2019, Taipei, Taiwan.

Revues internationales :

— [16] S. Beugnon, W. Puech, J.-P. Pedeboy. "Format-Compliant Selective Secret 3D Object Sharing Scheme". IEEE Transactions on Multimedia, 2019.

— [19] S. Beugnon, W. Puech, J.-P. Pedeboy. "A Fast Access Hierarchical Format-Compliant Selective Secret 3D Object Sharing Scheme". IEEE Transactions on Information Forensics and Security, Soumis.

— [18] S. Beugnon, N. Amalou, W. Puech, J.-P. Pedeboy. "Visual Confidentiality Measure : a new metric to predict the confidentiality of selectively encrypted 3D objects". IEEE Transactions on Information Forensics and Security, En cours de soumission.

Bibliographie

[1] A. U. A. Pommer. Application scenarios for selective encryption of visual data. In P. W. J. Dittmann, J. Fridrich, editor, *Multimedia and Security Workshop, ACM Multimedia*, pages 71–74, Juan-les-Pins, France, Dec. 2002. 27, 133

[2] P. Amat. *Digitalisation sécurisée d'objets 3D : application aux formes et aux lignes de style de chaussures. (Secure 3D objects digitization : application to shoe shapes and shoe line styles)*. PhD thesis, Université de Montpellier 2, France, 2008. 2

[3] P. Amat, W. Puech, S. Druon, et J.-P. Pedeboy. Lossless 3d steganography based on mst and connectivity modification. *Signal Processing : Image Communication*, 25 (6) :400–412, 2010. 2

[4] L. J. Anbarasi et G. A. Mala. Verifiable multi secret sharing scheme for 3d models. *Int. Arab J. Inf. Technol.*, 12(6) :708–713, 2015. 40, 43, 44, 87, 88

[5] N. Aspert, D. Santa-Cruz, et T. Ebrahimi. Mesh : Measuring errors between surfaces using the hausdorff distance. In *Proceedings of the 2002 IEEE International Conference on Multimedia and Expo, ICME 2002, Lausanne, Switzerland. August 26-29, 2002. Volume I*, volume 1, pages 705–708. IEEE, 2002. 12

[6] A. Beimel. Secret-sharing schemes : a survey. In *International Conference on Coding and Cryptology*, pages 11–46. Springer, 2011. 29, 30, 32

[7] M. Belenkiy. Disjunctive Multi-Level Secret Sharing. *IACR Cryptology ePrint Archive*, 2008 :18, 2008. 32, 33, 45, 47, 49, 50, 94, 97, 98, 102, 103, 104, 106, 107, 113, 114, 115, 116, 117, 118, 160

[8] J. C. Benaloh. Secret Sharing Homomorphisms : Keeping Shares of a Secret Secret (Extended Abstract). In *Advances in Cryptology — CRYPTO' 86*, Lecture Notes in Computer Science, pages 251–260. Springer, Berlin, Heidelberg, Aug. 1986. ISBN 978-3-540-18047-0 978-3-540-47721-1. 30, 33

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

[10] M. Berger, A. Tagliasacchi, L. M. Seversky, P. Alliez, J. A. Levine, A. Sharf, et C. T. Silva. State of the art in surface reconstruction from point clouds. In S. Lefebvre et M. Spagnuolo, editors, *Eurographics 2014 - State of the Art Reports, Strasbourg, France, April 7-11, 2014*, pages 161–185. Eurographics Association, 2014. 10

[11] S. Beugnon, W. Puech, et J. Pedeboy. An efficient lossless (2, n) secret image sharing based on blakley's scheme. In *19th IEEE International Workshop on Multimedia Signal Processing, MMSP 2017, Luton, United Kingdom, October 16-18, 2017*, pages 1–6. IEEE, 2017. 90, 165

[12] S. Beugnon, W. Puech, et J.-P. Pedeboy. Chiffrement sélectif d'objets 3d. In *Proceedings of COmpression et REprésentation des Signaux Audiovisuels*, 2017. 66, 165

[13] S. Beugnon, W. Puech, et J. Pedeboy. From visual confidentiality to transparent format-compliant selective encryption of 3d objects. In *2018 IEEE International Conference on Multimedia & Expo Workshops, ICME Workshops 2018, San Diego, CA, USA, July 23-27, 2018*, pages 1–6. IEEE Computer Society, 2018. 66, 165

[14] S. Beugnon, P. Puteaux, et W. Puech. Protection de la vie privée par partage hybride de photos sur les réseaux sociaux. In *Proceedings of COmpression et REprésentation des Signaux Audiovisuels*, 2018. 118, 165

[15] S. Beugnon, W. Puech, et J. Pedeboy. A format-compliant selective secret 3d object sharing scheme based on shamir's scheme. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*, pages 2657–2661. IEEE, 2019. 91, 165

[16] S. Beugnon, W. Puech, et J.-P. Pedeboy. Format-compliant selective secret 3d object sharing scheme. *IEEE Transactions on Multimedia*, 2019. 91, 165

[17] S. Beugnon, P. Puteaux, et W. Puech. Privacy protection for social media based on a hierarchical secret image sharing scheme. In *2019 IEEE International Conference on Image Processing, ICIP 2019, Taipei, Taiwan, September 22-25, 2019*. IEEE, 2019. 118, 165

[18] S. Beugnon, N. Amalou, W. Puech, et J.-P. Pedeboy. Visual confidentiality measure : a new metric to predict the confidentiality of selectively encrypted 3d objects. *IEEE Transactions on Information Forensics & Security*, en cours de soumission. 157, 165

[19] S. Beugnon, W. Puech, et J.-P. Pedeboy. A fast access hierarchical format-compliant selective secret 3d object sharing scheme. *IEEE Transactions on Information Forensics & Security*, soumis. 119, 165

[20] G. R. Blakley. Safeguarding cryptographic keys. In *Proceedings of the National Computer Conference*, volume 48, pages 313–317, 1979. 29, 30, 32, 33, 34, 36, 40, 43, 68, 69, 70, 76, 77, 78, 79, 80, 81, 82, 83, 85, 89, 90, 159, 160

[21] A. G. Bors. Watermarking mesh-based representations of 3-d objects using local moments. *IEEE Trans. Image Processing*, 15(3) :687–701, 2006. 19

[22] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, et B. Lévy. *Polygon Mesh Processing*. A K Peters, 2010. ISBN 978-1-56881-426-1. 10, 11

[23] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 96

[24] E. F. Brickell. Some ideal secret sharing schemes. In J. Quisquater et J. Vandewalle, editors, *Advances in Cryptology - EUROCRYPT '89, Workshop on the Theory and Application of of Cryptographic Techniques, Houthalen, Belgium, April 10-13, 1989, Proceedings*, volume 434 of *Lecture Notes in Computer Science*, pages 468–475. Springer, 1989. 32, 33

[25] E. F. Brickell et D. M. Davenport. On the classification of ideal secret sharing schemes. *J. Cryptology*, 4(2) :123–134, 1991.

[26] E. F. Brickell et D. R. Stinson. Some improved bounds on the information rate of perfect secret sharing schemes. *J. Cryptology*, 5(3) :153–166, 1992. 32, 33

[27] J. B. Campbell et R. H. Wynne. *Introduction to remote sensing*. Guilford Press, 2011. 10

[28] P. R. Cavalcanti et U. T. Mello. Three-Dimensional Constrained Delaunay Triangulation : a Minimalist Approach. In *IMR*, pages 119–129, 1999. 10

[29] F. Cayre et B. Macq. Data hiding on 3-D triangle meshes. *IEEE Transactions on signal Processing*, 51(4) :939–949, 2003. 19

[30] F. Cayre, P. Rondao-Alface, F. J. M. Schmitt, B. Macq, et H. Maître. Application of spectral decomposition to compression and watermarking of 3d triangle mesh geometry. *Sig. Proc. : Image Comm.*, 18(4) :309–319, 2003. 19

[31] M.-W. Chao, C.-h. Lin, C.-W. Yu, et T.-Y. Lee. A high capacity 3d steganography algorithm. *IEEE transactions on visualization and computer graphics*, 15(2) :274–284, 2009. 13, 19, 127

[32] C. Chen et W. Fu. A geometry-based secret image sharing approach. *J. Inf. Sci. Eng.*, 24(5) :1567–1577, 2008. 75, 76

[33] X. Chen, A. Golovinskiy, et T. Funkhouser. A Benchmark for 3d Mesh Segmentation. *ACM Transactions on Graphics (TOG)*, 28(3) :73, 2009. 61, 111, 125, 130

[34] Y. Chen, L. Chen, et S. J. Shyu. Secret image sharing with smaller shadow sizes for general access structures. *Multimedia Tools Appl.*, 75(21) :13913–13929, 2016. 39

[35] J. Cho, R. Prost, et H. Jung. An oblivious watermarking for 3-d polygonal meshes using distribution of vertex norms. *IEEE Trans. Signal Processing*, 55(1) :142–155, 2007. 127, 128

[36] M. Cho, S. Kim, M. Sung, et G. On. 3d fingerprinting and encryption principle for collaboration. In *International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution (AXMEDIS)*. IEEE, 2006. 19, 65

[37] B. Chor, S. Goldwasser, S. Micali, et B. Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults (extended abstract). In *26th Annual Symposium on Foundations of Computer Science, Portland, Oregon, USA, 21-23 October 1985*, pages 383–395. IEEE Computer Society, 1985. 33

[38] P. Cignoni, C. Rocchini, et R. Scopigno. Metro : Measuring error on simplified surfaces. In *Computer Graphics Forum*, volume 17, pages 167–174. Wiley Online Library, 1998. 12

[39] S. Cimato et C.-N. Yang. *Visual Cryptography and Secret Image Sharing (Digital Imaging and Computer Vision)*. CRC Press, Inc., Boca Raton, FL, USA, 2011. ISBN 143983721X, 9781439837214. 37, 38

[40] S. Cimato, R. D. Prisco, et A. D. Santis. Probabilistic visual cryptography schemes. *Comput. J.*, 49(1) :97–107, 2006. 38

[41] I. Cox, M. Miller, J. Bloom, J. Fridrich, et T. Kalker. *Digital watermarking and steganography*. Morgan kaufmann, 2007. 16

[42] J. Daemen et V. Rijmen. Rijndael, the advanced encryption standard. *Dr. Dobb's journal*, 26(3) :137–139, 2001. 26

[43] R. Davis. The data encryption standard in perspective. *IEEE Communications Society Magazine*, 16(6) :5–9, 1978. 23

[44] M. Desoubeaux. *Codes de traçage de traîtres pour la protection de contenus numériques*. PhD thesis, Montpellier 2, 2013. 19

[45] P. Deutsch et J.-L. Gailly. Zlib compressed data format specification version 3.3. Technical report, IETF, 1996. 42, 43

[46] W. Diffie et M. Hellman. New directions in cryptography. *IEEE transactions on Information Theory*, 22(6) :644–654, 1976. 27

[47] M. J. Dworkin. Recommendation for block cipher modes of operation : Galois/counter mode (gcm) and gmac| nist. Technical report, NIST, 2007. 24

[48] E. Elsheh et A. Hamza. Robust approaches to 3d object secret sharing. *Image Analysis and Recognition*, pages 326–335, 2010. 36, 40, 41, 42, 43, 44

[49] E. Elsheh et A. B. Hamza. Secret sharing approaches for 3d object encryption. *Expert Systems with Applications*, 38(11) :13906–13911, 2011. 40, 41, 42, 43, 44, 87, 88

[50] M. Éluard, Y. Maetz, et G. J. Doërr. Impact of geometry-preserving encryption on rendering time. In *2014 IEEE International Conference on Image Processing, ICIP 2014, Paris, France, October 27-30, 2014*, pages 4787–4791. IEEE, 2014. 28, 29, 65

[51] O. Farràs, J. Martí-Farré, et C. Padró. Ideal multipartite secret sharing schemes. *Journal of cryptology*, 25(3) :434–463, 2012. 33

[52] J. D. Foley, F. D. Van, A. Van Dam, S. K. Feiner, J. F. Hughes, J. HUGHES, et E. ANGEL. *Computer graphics : principles and practice*, volume 12110. Addison-Wesley Professional, 1996. 9

[53] J. Fridrich. Image watermarking for tamper detection. In *Proceedings of the 1998 IEEE International Conference on Image Processing, ICIP-98, Chicago, Illinois, USA, October 4-7, 1998*, pages 404–408. IEEE Computer Society, 1998. 19

[54] D. Goldberg. What every computer scientist should know about floating-point arithmetic. *ACM Computing Surveys (CSUR)*, 23(1) :5–48, 1991. 57

[55] M. Gschwandtner et A. Uhl. Protected Progressive Meshes. In Springer, editor, *Advances in Visual Computing*, pages 35–48. Springer, 2009. 28, 65

[56] J. Guo. *Contributions to objective and subjective visual quality assessment of 3d models. (Contributions à l'évaluation objective et subjective de la qualité visuelle des modèles 3D)*. PhD thesis, University of Lyon, France, 2016. 130, 134

[57] L. Harn et C. Lin. Detection and identification of cheaters in (t , n) secret sharing scheme. *Des. Codes Cryptography*, 52(1) :15–24, 2009. 33

[58] L. R. Herrmann. Laplacian-isoparametric grid generation scheme. *Journal of the Engineering Mechanics Division*, 102(5) :749–907, 1976. 86, 112

[59] E. Hjelmås et B. K. Low. Face detection : A survey. *Computer Vision and Image Understanding*, 83(3) :236–274, 2001. 96

[60] Hootsuite. The global state of digital in 2019 report. https://hootsuite.com/pages/digital-in-2019, 2019. Accessed : 2019-01-14. 94

[61] D. A. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9) :1098–1101, 1952. 42, 43

[62] IEEE. IEEE Standard for Floating-Point Arithmetic. *IEEE Std 754-2008*, pages 1–70, 2008. 57

[63] V. Itier. *Nouvelles méthodes de synchronisation de nuages de points 3D pour l'insertion de données cachées. (New 3D point cloud synchronization methods for data hiding)*. PhD thesis, University of Montpellier, France, 2015. 2, 34, 160

[64] V. Itier et W. Puech. High capacity data hiding for 3d point clouds based on static arithmetic coding. *Multimedia Tools Appl.*, 76(24) :26421–26445, 2017. 2, 19, 20, 21, 122, 123, 124, 125, 126, 127, 128, 156, 160

[65] V. Itier, W. Puech, et A. G. Bors. Cryptanalysis aspects in 3-D watermarking. In *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014.

[66] V. Itier, W. Puech, et A. G. Bors. Cryptanalysis aspects in 3-d watermarking. In *2014 IEEE International Conference on Image Processing, ICIP 2014, Paris, France, October 27-30, 2014*, pages 4772–4776. IEEE, 2014.

[67] V. Itier, W. Puech, G. Gesquière, et J. Pedeboy. Joint synchronization and high capacity data hiding for 3d meshes. In R. Sitnik et W. Puech, editors, *Three-Dimensional Image Processing, Measurement (3DIPM), and Applications 2015, San Francisco, California, USA, February 10-12, 2015*, volume 9393 of *SPIE Proceedings*, page 939305. SPIE, 2015.

[68] V. Itier, W. Puech, G. Gesquière, et J.-P. Pedeboy. Joint synchronization and high capacity data hiding for 3d meshes. In *SPIE/IS&T Electronic Imaging*, pages 939305–939305. International Society for Optics and Photonics, 2015. 20

[69] V. Itier, W. Puech, et J. Pedeboy. Highcapacity data-hiding for 3d meshes based on static arithmetic coding. In *2015 IEEE International Conference on Image Processing, ICIP 2015, Quebec City, QC, Canada, September 27-30, 2015*, pages 4575–4579. IEEE, 2015. 19

[70] V. Itier, W. Puech, et J.-P. Pedeboy. High capacity data-hiding for 3d meshes based on static arithmetic coding. In *IEEE International Conference on Image Processing (ICIP)*, pages 4575–4579. IEEE, 2015. 34

[71] V. Itier, N. Tournier, W. Puech, G. Subsol, et J. Pedeboy. Analysis of an emst-based path for 3d meshes. *Computer-Aided Design*, 64 :22–32, 2015. 2

[72] V. Itier, A. G. Bors, W. Puech, et J.-P. Pedeboy. Secure High Capacity Data Hiding for 3d Meshes. *Electronic Imaging*, 2016(21) :1–7, 2016. 20

[73] M. Ito, A. Saito, et T. Nishizeki. Secret sharing scheme realizing general access structure. *Electronics and Communications in Japan (Part III : Fundamental Electronic Science)*, 72(9) :56–64, 1989. 30

[74] O. Kafri et E. Keren. Encryption of pictures and shapes by random grids. *Opt. Lett.*, 12(6) :377–379, Jun 1987. doi: 10.1364/OL.12.000377. 38

[75] D. Kahn. *The Codebreakers : The comprehensive history of secret communication from ancient times to the internet.* Simon and Schuster, 1996. 22, 39

[76] B. Kaliski. Pkcs# 5 : Password-based cryptography specification version 2.0. Technical report, IETF, 2000. 118

[77] M. M. Kazhdan, M. Bolitho, et H. Hoppe. Poisson surface reconstruction. In A. Sheffer et K. Polthier, editors, *Proceedings of the Fourth Eurographics Symposium on Geometry Processing, Cagliari, Sardinia, Italy, June 26-28, 2006*, volume 256 of *ACM International Conference Proceeding Series*, pages 61–70. Eurographics Association, 2006. 10, 113

[78] A. Kerckhoffs. La cryptographic militaire. *Journal des sciences militaires*, pages 5–38, 1883. 22, 23

[79] J. Kodovský, J. J. Fridrich, et V. Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Trans. Information Forensics and Security*, 7(2) :432–444, 2012. 122

[80] H. Krawczyk. Secret sharing made short. In D. R. Stinson, editor, *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, volume 773 of *Lecture Notes in Computer Science*, pages 136–146. Springer, 1993. 33

[81] G. Lavoué, E. D. Gelasca, F. Dupont, A. Baskurt, et T. Ebrahimi. Perceptually driven 3d distance metrics with application to watermarking. In *Applications of Digital Image Processing XXIX*, volume 6312, page 63120L. International Society for Optics and Photonics, 2006. 14

[82] G. Lavoué, J. Vandeborre, H. Benhabiles, M. Daoudi, K. Huebner, M. Mortara, et M. Spagnuolo. Shrec'12 track : 3d mesh segmentation. In M. Spagnuolo, M. M. Bronstein, A. M. Bronstein, et A. Ferreira, editors, *Eurographics Workshop on 3D Object Retrieval 2012, Cagliari, Italy, May 13, 2012. Proceedings*, pages 93–99. Eurographics Association, 2012. 130

[83] G. Lavoué, I. Cheng, et A. Basu. Perceptual quality metrics for 3d meshes : Towards an optimal multi-attribute computational model. In *IEEE International Conference on Systems, Man, and Cybernetics, Manchester, SMC 2013, United Kingdom, October 13-16, 2013*, pages 3271–3276. IEEE, 2013. 131

[84] G. Lavoué. A multiscale metric for 3d mesh visual quality assessment. In *Computer Graphics Forum*, volume 30, pages 1427–1437. Wiley Online Library, 2011. 12, 14

[85] S.-S. Lee, Y.-J. Huang, et J.-C. Lin. Protection of 3d models using cross recovery. *Multimedia Tools and Applications*, 76(1) :243–264, Jan. 2017. ISSN 1380-7501, 1573-7721. 40, 42, 43, 44, 87, 88

[86] S. Z. Li et A. K. Jain, editors. *Handbook of Face Recognition, 2nd Edition*. Springer, 2011. ISBN 978-0-85729-931-4. 96

[87] Z. Li. *Steganalytic methods for 3D objects*. PhD thesis, University of York, UK, 2018. 122, 156

[88] Z. Li et A. G. Bors. 3d mesh steganalysis using local shape features. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, March 20-25, 2016*, pages 2144–2148. IEEE, 2016. 123, 125

[89] Z. Li et A. G. Bors. Selection of robust features for the cover source mismatch problem in 3d steganalysis. In *23rd International Conference on Pattern Recognition, ICPR 2016, Cancún, Mexico, December 4-8, 2016*, pages 4256–4261. IEEE, 2016. 122, 127

[90] Z. Li, S. Beugnon, W. Puech, et A. G. Bors. Rethinking the high capacity 3d steganography : Increasing its resistance to steganalysis. In *2017 IEEE International Conference on Image Processing, ICIP 2017, Beijing, China, September 17-20, 2017*, pages 510–414. IEEE, 2017. 157, 165

[91] C. Lin et W. Tsai. Secret image sharing with steganography and authentication. *Journal of Systems and software*, 73(3) :405–414, 2004. 39, 75, 76

[92] C. Lin, L. Harn, et D. Ye. Ideal perfect multilevel threshold secret sharing scheme. In *Proceedings of the Fifth International Conference on Information Assurance and Security, IAS 2009, Xi'An, China, 18-20 August 2009*, pages 118–121. IEEE Computer Society, 2009. 33, 39

[93] W. E. Lorensen et H. E. Cline. Marching Cubes : A High Resolution 3d Surface Construction Algorithm. *SIGGRAPH Computer Graphics*, 21(4) :163–169, 1987. ISSN 0097-8930. 86, 113

[94] M. Luo, K. Wang, A. G. Bors, et G. Lavoué. Local patch blind spectral watermarking method for 3d graphics. In *International Workshop on Digital Watermarking*, pages 211–226. Springer, 2009. 19

[95] A. Maglo, G. Lavoué, F. Dupont, et C. Hudelot. 3d mesh compression : Survey, comparisons, and emerging trends. *ACM Computing Surveys (CSUR)*, 47(3) :44, 2015. 11

[96] M. Naor et A. Shamir. Visual cryptography. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 1–12. Springer, 1994. 36, 37, 38

[97] M. Naor et A. Shamir. Visual cryptography ii : Improving the contrast via the cover base. In *International Workshop on Security Protocols*, pages 197–202. Springer, 1996. 38

[98] M. Nojoumian et D. R. Stinson. Sequential Secret Sharing as a New Hierarchical Access Structure. *IACR Cryptology ePrint Archive*, 2015 :403, 2015. 33

[99] R. Ohbuchi, S. Takahashi, T. Miyazawa, et A. Mukaiyama. Watermarking 3d polygonal meshes in the mesh spectral domain. In *Proceedings of the Graphics Interface 2001 Conference, Ottawa, Ontario, Canada, June 7-9, 2001*, pages 9–18. Canadian Human-Computer Communications Society, 2001. 19

[100] R. Ohbuchi, A. Mukaiyama, et S. Takahashi. A frequency-domain approach to watermarking 3d shapes. In *Computer graphics forum*, volume 21, pages 373–382. Wiley Online Library, 2002. 19

[101] F. A. Petitcolas, R. J. Anderson, et M. G. Kuhn. Information hiding-a survey. *Proceedings of the IEEE*, 87(7) :1062–1078, 1999. 18

[102] B. T. Phong. Illumination for computer generated pictures. *Commun. ACM*, 18(6) : 311–317, 1975. 135

[103] D. Pickup, X. Sun, P. L. Rosin, R. R. Martin, Z. Cheng, Z. Lian, M. Aono, A. B. Hamza, A. M. Bronstein, M. M. Bronstein, S. Bu, U. Castellani, S. Cheng, V. Garro, A. Giachetti, A. Godil, J. Han, H. Johan, L. Lai, B. Li, C. Li, H. Li, R. Litman, X. Liu, Z. Liu, Y. Lu, A. Tatsuma, et J. Ye. Shape retrieval of non-rigid 3d human models. In B. Bustos, H. Tabia, J. Vandeborre, et R. C. Veltkamp, editors, *Eurographics Workshop on 3D Object Retrieval, Strasbourg, France, 2014. Proceedings*, pages 101–110. Eurographics Association, 2014. 130

[104] W. Puech et J. Rodrigues. Crypto-Compression of Medical Images by Selective Encryption of DCT. In *EUSIPCO : EUropean SIgnal Processing COnference*, Antalya, Turkey, 2005. 27

[105] W. Puech, A. G. Bors, et J. M. Rodrigues. Protection of Colour Images by Selective Encryption. In C. Fernandez-Maloigne, editor, *Advanced Color Image Processing and Analysis*, pages 397–421. Springer New York, New York, NY, 2013. ISBN 978-1-4419-6190-7. 27

[106] C. R. Qi, H. Su, K. Mo, et L. J. Guibas. Pointnet : Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 77–85. IEEE Computer Society, 2017. 163

[107] I. S. Reed et G. Solomon. Polynomial codes over certain finite fields. *Journal of the society for industrial and applied mathematics*, 8(2) :300–304, 1960. 42, 43

[108] R. L. Rivest, A. Shamir, et L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2) :120–126, 1978. 27

[109] P. Rondao-Alface et B. Macq. From 3d mesh data hiding to 3d shape blind and robust watermarking : A survey. *Trans. Data Hiding and Multimedia Security*, 2 :91–115, 2007. 19

[110] J. Rossignac et A. Szymczak. Wrap&Zip decompression of the connectivity of triangle meshes compressed with Edgebreaker. *Computational Geometry*, 14(1) :119–135, Nov. 1999. ISSN 0925-7721. 43

[111] R. A. Rueppel. *Analysis and design of stream ciphers*. Springer Science & Business Media, 2012. 26

[112] A. Said. Measuring the strength of partial encryption schemes. In *IEEE International Conference on Image Processing (ICIP)*, volume 2, pages II–1126. IEEE, 2005. 63, 86, 112

[113] G. Sansoni, M. Trebeschi, et F. Docchio. State-of-The-Art and Applications of 3D Imaging Sensors in Industry, Cultural Heritage, Medicine, and Criminal Investigation. *Sensors*, 9(1) :568–601, 2009. 8, 11

[114] I. J. Schoenberg. On Hermite-Birkhoff interpolation. *Journal of Mathematical Analysis and Applications*, 16(3) :538–543, 1966. 49, 50

[115] Z. Shahid, M. Chaumont, et W. Puech. Fast Protection of H.264/AVC by Selective Encryption of CAVLC and CABAC for I and P Frames. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(5) :565–576, 2011. ISSN 1051-8215. 27

[116] A. Shamir. How to share a secret. *Communications of the ACM*, 22(11) :612–613, 1979. 29, 30, 31, 33, 34, 36, 38, 40, 41, 42, 43, 47, 48, 49, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 89, 90, 102, 103, 112, 160

[117] S. J. Shyu. Visual cryptograms of random grids for general access structures. *IEEE Trans. Circuits Syst. Video Techn.*, 23(3) :414–424, 2013. 38

[118] G. J. Simmons. The prisoners' problem and the subliminal channel. In *Advances in Cryptology*, pages 51–67. Springer, 1984. 18

[119] G. J. Simmons. How to (Really) Share a Secret. In *Advances in Cryptology — CRYPTO' 88*, Lecture Notes in Computer Science, pages 390–448. Springer, New York, NY, Aug. 1988. ISBN 978-0-387-97196-4 978-0-387-34799-8. 32, 33, 45, 46

[120] G. Singh. A study of encryption algorithms (RSA, DES, 3des and AES) for information security. *International Journal of Computer Applications*, 67(19), 2013. 26

[121] C. Stergiou, K. E. Psannis, B.-G. Kim, et B. Gupta. Secure integration of iot and cloud computing. *Future Generation Computer Systems*, 78 :964–975, 2018. 26

[122] J. Stern. *Science du secret (La)*. Odile Jacob, 1998. 22

[123] I. Stroud. *Boundary representation modelling techniques*. Springer Science & Business Media, 2006. 9

[124] J. M. Such et N. Criado. Multiparty privacy in social media. *Commun. ACM*, 61(8) : 74–81, 2018. 94, 95

[125] T. Tassa. Hierarchical threshold secret sharing. In *Theory of Cryptography Conference*, pages 473–490. Springer, 2004. 32, 33

[126] T. Tassa. Hierarchical threshold secret sharing. *Journal of cryptology*, 20(2) :237–264, 2007. 45, 47, 48, 49, 50, 102, 113, 114, 115, 117

[127] G. Taubin. Curve and surface smoothing without shrinkage. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 852–857. IEEE, 1995. 86, 112

[128] C. Thien et J. Lin. Secret image sharing. *Computers & Graphics*, 26(5) :765–770, 2002. 36, 37, 38, 39, 41, 42, 43, 69, 75, 76

[129] C. Thien et J. Lin. An image-sharing method with user-friendly shadow images. *IEEE Transactions on circuits and systems for video technology*, 13(12) :1161–1169, 2003. 39, 42, 69, 75, 76

[130] F. Torkhani. *Analyse subjective et évaluation objective de la qualité perceptuelle des maillages 3D. (Subjective and objective perceptual quality assessment of 3D meshes)*. PhD thesis, University of Grenoble, France, 2014. 12, 15

[131] N. Tournier. *Synchronisation pour l'insertion de données dans des maillages 3D. (Synchronization for data hiding in 3D meshes)*. PhD thesis, Université de Montpellier 2, France, 2014. 2

[132] C. Tsai, C. Chang, et T. Chen. Sharing multiple secrets in digital images. *Journal of Systems and Software*, 64(2) :163–170, 2002. 39

[133] M. Tsai et C. Chen. A study on secret image sharing. In *Proceedings of the 6th International Workshop on Image Media Quality and its Applications, Tokyo, Japan*. Citeseer, 2013. 39, 75, 76

[134] Y.-Y. Tsai. A Secret 3d Model Sharing Scheme with Reversible Data Hiding Based on Space Subdivision. *3D Research*, 7(1) :1, 2016. 40, 42, 43, 44, 87, 88

[135] H. Tso. Sharing secret images using Blakley's concept. *Optical Engineering*, 47(7) : 077001–077001, 2008. 75, 76

[136] M. Van Droogenbroeck et R. Benedet. Techniques for a selective encryption of uncompressed and compressed images. In *Advanced Concepts for Intelligent Vision Systems (ACIVS)*, pages 90–97, Ghent, Belgium, 2002. 27

[137] L. Vása et J. Rus. Dihedral angle mesh error : a fast perception correlated distortion measure for fixed connectivity triangle meshes. *Comput. Graph. Forum*, 31(5) : 1715–1724, 2012. 12, 14, 15

[138] G. S. Vernam. Cipher printing telegraph systems : For secret wire and radio telegraphic communications. *Journal of the AIEE*, 45(2) :109–115, 1926. 26

[139] P. A. Viola et M. J. Jones. Robust real-time face detection. In *ICCV*, page 747, 2001. 96

[140] H. Wang et S. Wang. Cyber warfare : steganography vs. steganalysis. *Commun. ACM*, 47(10) :76–82, 2004. 122

[141] K. Wang, G. Lavoue, F. Denis, et A. Baskurt. A Comprehensive Survey on Three-Dimensional Mesh Watermarking. *IEEE Transactions on Multimedia*, 10(8) :1513–1527, Dec. 2008. ISSN 1520-9210. 18, 19

[142] K. Wang, M. Luo, A. G. Bors, et F. Denis. Blind and robust mesh watermarking using manifold harmonics. In *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2009. 19

[143] K. Wang, F. Torkhani, et A. Montanvert. A fast roughness-based approach to the assessment of 3d mesh visual quality. *Computers & Graphics*, 36(7) :808–818, 2012. 12, 14

[144] R.-Z. Wang et S.-J. Shyu. Scalable secret image sharing. *Signal Processing : Image Communication*, 22(4) :363–373, 2007. 39

[145] R.-Z. Wang et C.-H. Su. Secret image sharing with smaller shadow images. *Pattern Recognition Letters*, 27(6) :551–555, 2006. 39

[146] Z. Wang, A. C. Bovik, H. R. Sheikh, et E. P. Simoncelli. Image quality assessment : from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4) :600–612, 2004. 14

[147] J. Weir et W. Yan. A comprehensive study of visual cryptography. *Trans. Data Hiding and Multimedia Security*, 5 :70–105, 2010. 38

[148] J. Wu et L. Kobbelt. Efficient spectral watermarking of large meshes with orthogonal basis functions. *The Visual Computer*, 21(8-10) :848–857, 2005. 19

[149] Y. Wu, C. Thien, et J. Lin. Sharing and hiding secret images with size constraint. *Pattern Recognition*, 37(7) :1377–1385, 2004. 75, 76

[150] T. Xu et Z. q. Cai. A Novel Semi-fragile Watermarking Algorithm for 3d Mesh Models. In *2012 International Conference on Control Engineering and Communication Technology*, Dec. 2012. 19

[151] X. Yan, S. Wang, X. Niu, et C.-N. Yang. Random grid-based visual secret sharing with multiple decryptions. *Journal of Visual Communication and Image Representation*, 26 :94–104, Jan. 2015. ISSN 1047-3203. 39

[152] C. Yang, P. Li, C. Wu, et S. Cai. Reducing shadow size in essential secret image sharing by conjunctive hierarchical approach. *Sig. Proc. : Image Comm.*, 31 :1–9, 2015. 97

[153] C.-C. Yang, T.-Y. Chang, et M.-S. Hwang. A (t,n) multi-secret sharing scheme. *Applied Mathematics and Computation*, 151(2) :483–490, Apr. 2004. ISSN 0096-3003. 38, 39

[154] C.-N. Yang, T.-S. Chen, K. H. Yu, et C.-C. Wang. Improvements of image sharing with steganography and authentication. *Journal of Systems and software*, 80(7) : 1070–1076, 2007. 39, 78

[155] C.-N. Yang, X. Wu, Y.-C. Chou, et Z. Fu. Constructions of general (k,n) reversible AMBTC-based visual cryptography with two decryption options. *Journal of Visual Communication and Image Representation*, 48 :182–194, Oct. 2017. ISSN 1047-3203. 128

[156] H. Yang, X. Sun, et G. Sun. A high-capacity image data hiding scheme using adaptive lsb substitution. *Radioengineering*, 18(4) :509–516, 2009. 19

[157] Y. Yang et I. P. Ivrissimtzis. Polygonal mesh watermarking using laplacian coordinates. *Comput. Graph. Forum*, 29(5) :1585–1593, 2010. 123

[158] Y. Yang et I. P. Ivrissimtzis. Mesh discriminative features for 3d steganalysis. *TOMCCAP*, 10(3) :27 :1–27 :13, 2014. 122, 123, 127

[159] Y. Yang, R. Pintus, H. E. Rushmeier, et I. P. Ivrissimtzis. A 3d steganalytic algorithm and steganalysis-resistant watermarking. *IEEE Trans. Vis. Comput. Graph.*, 23(2) : 1002–1013, 2017. 122, 125, 127

[160] S. Zafeiriou, A. Tefas, et I. Pitas. Blind robust watermarking schemes for copyright protection of 3d mesh objects. *IEEE Transactions on Visualization and Computer Graphics*, 11(5) :596–607, Sept. 2005. ISSN 1077-2626. 18, 19

[161] J. Zhao, J. Zhang, et R. Zhao. A practical verifiable multi-secret sharing scheme. *Computer Standards & Interfaces*, 29(1) :138–141, Jan. 2007. ISSN 0920-5489. 33

[162] H. Zhou, K. Chen, W. Zhang, Y. Yao, et N. Yu. Distortion design for secure adaptive 3-d mesh steganography. *IEEE Trans. Multimedia*, 21(6) :1384–1398, 2019. 156

[163] Q. Zhou et A. Jacobson. 2018 cover image : Thingi10k. *Comput. Graph. Forum*, 37 (1) :451–452, 2018. 130