

Technologies MPEG pour la Compression de Maillages 3D Statiques et Dynamiques

Khaled Mammou¹, Titus Zharia² et Françoise Prêteux²

¹ École Nationale des Sciences de l'Informatique, Université de La Manouba,
Laboratoire CRISTAL, Groupe de Recherche Images et Formes de Tunisie (GRIFT),
Campus Universitaire de la Manouba 2010, Tunisie.
kmamou@gmail.com

² Institut TELECOM / TELECOM SudParis,
Département ARTEMIS
9, Rue Charles Fourier, 91011 Evry Cedex - FRANCE
Titus.Zaharia@it-sudparis.eu

Résumé Cet article esquisse une vue d'ensemble des différentes technologies de compression de maillages 3D statiques et dynamiques développés dans le cadre de la norme ISO/MPEG (*Moving Picture Experts Group*). En partant du codeur de maillages statiques 3DMC (*3D Mesh Coding*) normalisé en 1998, et en passant par les technologies IC (*Interpolator Compression*), FAMC (*Frame-based Animated Mesh Compression*) et SC3DMC (*Scalable Complexity 3D Mesh Coding*), parues entre 2003 et 2010, les principes de ces approches sont présentés et leurs performances sont objectivement comparées et discutées.

Mots clés Compression, MPEG, maillage 3D, animation, FAMC, IC, 3DMC, TFAN, SC3DMC.

Abstract This article presents a detailed overview of the different ISO/MPEG (*Moving Picture Experts Group*) technologies dedicated to the compression of static and dynamic 3D meshes. The principles and the performances of the MPEG technologies 3DMC (*3D Mesh Coding*), IC (*Interpolator Compression*), FAMC (*Frame-based Animated Mesh Compression*) and SC3DMC (*Scalable Complexity 3D Mesh Coding*), published between 1998 and 2010, are presented and their performances objectively evaluated and discussed.

Key words Compression, MPEG, 3D mesh, animation, FAMC, IC, 3DMC, TFAN, SC3DMC.

1 Introduction

Aujourd'hui, les applications multimédias professionnelles ou grand public mettent de plus en plus en scène des contenus 3D aussi bien statiques que dynamiques, dans des contextes industriels divers. La Conception Assistée par Ordinateur (CAO) par l'industrie automobile ou aéronautique, les nouveaux services de télémédecine, les industries du jeu vidéo, des films d'animation 3D, des effets spéciaux ou

33 encore des dessins animés sont quelques exemples représentatifs de domaines où la modélisation 3D
34 joue un rôle incontournable.

35 Selon l'application considérée et les techniques utilisées pour les générer, ces contenus 3D sont
36 stockés sous divers formats, normalisés ou propriétaires. Pour des raisons d'interopérabilité
37 multiplateformes, de généralité ou encore de protection de la propriété intellectuelle, ces contenus sont
38 le plus souvent représentés sous forme de maillages 3D, statiques ou dynamiques.

39 L'inconvénient majeur de cette représentation est en revanche lié aux coûts importants de
40 stockage et de transmission. D'où la nécessité de mettre en œuvre des outils de compression efficaces
41 optimisés pour ce type de contenus. Cet article présente les différentes technologies actuellement
42 offertes par la norme ISO/MPEG (*Moving Picture Experts Group*) pour la compression de maillages
43 3D statiques et dynamiques. Le reste de l'article est structuré comme suit. La Section 2 se focalisera
44 sur les approches de compression de maillages 3D statiques. La section 3 sera dédiée à la compression
45 d'animations 3D. La section 4 conclut l'article en esquissant les perspectives de nouveaux
46 développements.

47 2 Technologies MPEG pour la Compression de Maillages 3D Statiques

48 Les maillages 3D sont aujourd'hui représentés sous une multitude de formats normalisés (*e.g.*,
49 VRML¹, MPEG [2] et COLLADA²) ou propriétaires (*e.g.*, 3DS et Maya³ développés par la société
50 Autodesk et BLEND de Blender⁴). Quel que soit le format considéré, la représentation commune
51 adoptée est celle par triangles indexés, dont le principe consiste à stocker la géométrie du maillage
52 sous la forme d'une matrice G de dimension $V \times 3$ (où V est le nombre de sommets), où chaque ligne
53 correspond aux coordonnées cartésiennes d'un sommet. La connectivité du maillage est, quant à elle,
54 spécifiée par une matrice d'entiers, notée Γ , de dimension $F \times 3$ (où F est le nombre de triangles).
55 Chaque ligne de Γ décrit les indexe des trois sommets qui composent un triangle du maillage.

56 En supposant que les coordonnées et les indexes sont codées sur 32 bits et que le maillage est
57 manifold, sans bords et de genre zéro, la quantité d'information $Q(M)$ nécessaire pour décrire
58 complètement le maillage M est de donnée par [1]:

$$59 \quad Q(M) = 288 \times V \text{ bits.} \quad (1)$$

60 Ce coût exorbitant est dû à la redondance inhérente à la représentation par triangles indexés. En
61 effet, dans la matrice Γ , chaque sommet apparaît autant de fois que le nombre de triangles dans
62 lesquels il intervient (*i.e.* en moyenne 6). De plus, la quantification sur 32 bits de la géométrie offre
63 une précision supérieure aux besoins applicatifs des données 3D. Afin d'éliminer cette redondance,
64 une littérature extrêmement riche a été dédiée à la compression de maillages 3D (*cf.* [3] pour une
65 étude détaillée). Dans cette section nous nous focaliserons uniquement sur les deux codeurs MPEG
66 3DMC [4] et MPEG SC3DMC [5] normalisées respectivement en 1998 et 2008 et qui de façon
67 générique à tout maillage 3D. Les codeurs WSS (*Wavelets Subdivision Surfaces*) et MeshGrid qui
68 s'appliquent uniquement qu'à des classe particulière de représentation de surfaces 3D (*e.g.* surfaces
69 de subdivision ou surfaces implicites) ne seront pas considérés. Le lecteur peut se référer à [17] et [18]
70 pour une description détaillées de ces techniques.

¹ www.web3d.org/x3d/specifications/vrml/

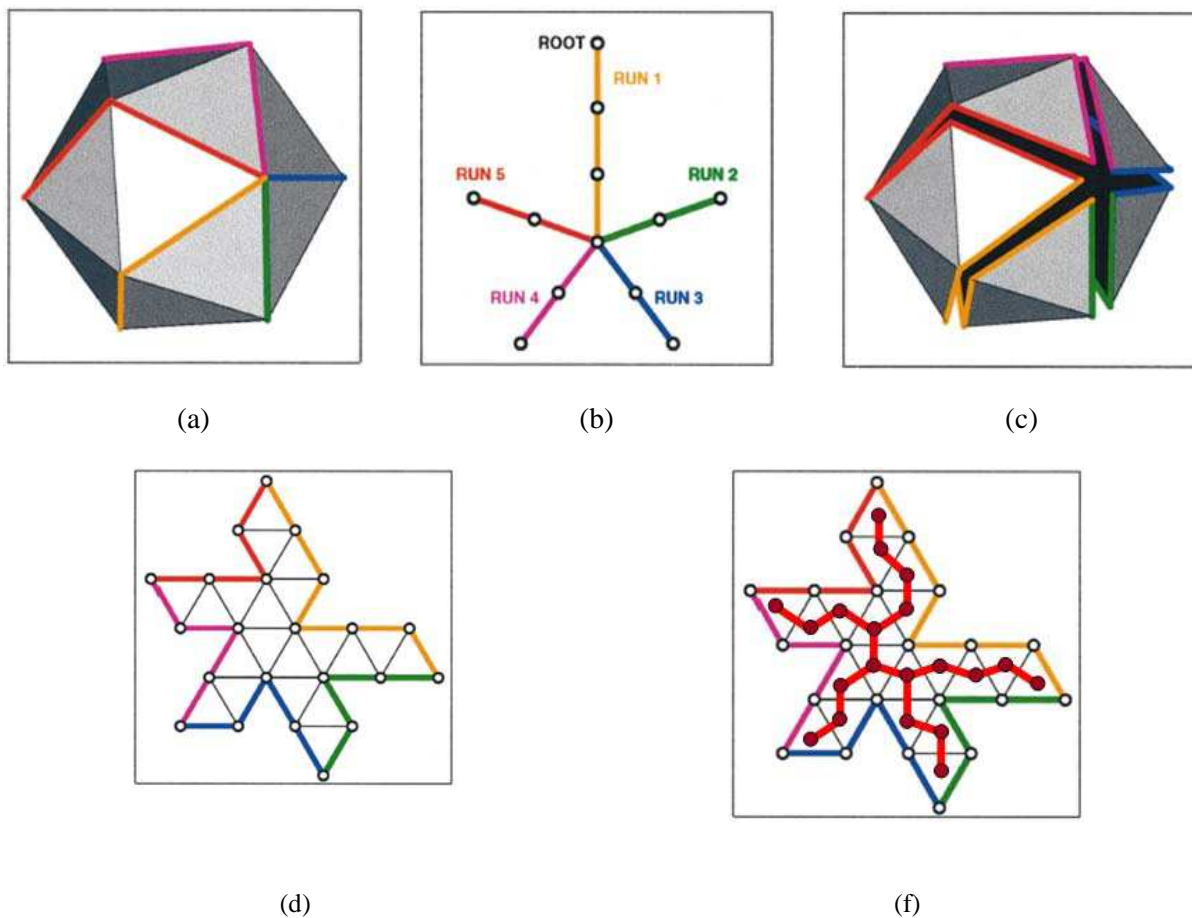
² www.khronos.org/files/collada_spec_1_4.pdf

³ www.autodesk.com

⁴ www.blender.org

71 **2.1 Codage par arbres couvrants**

72 Les méthodes de codage par arbres couvrants (*spanning trees*) s'appuient sur une observation
 73 fondamentale de Turan [6], qui a établi qu'un graphe planaire peut être codé avec un nombre constant
 74 de bits par sommet, en utilisant deux arbres couvrants : un arbre de sommets et un arbre des triangles.
 75 En partant de ce résultat théorique, Taubin et Rossignac proposent une méthode de codage de la
 76 connectivité appelée "*topological surgery*" [4]. Cette technique rebaptisée 3DMC (*3D Mesh Coding*)
 77 a été adoptée en 1998 par la norme *MPEG-4 Visual*.



78 **Figure 1** : Exemple de codage par arbres couvrants : (a) maillage original, (b) arbre couvrant des sommets, (c)
 79 découpage du maillage selon l'arbre couvrant des sommets, (d) polygone planaire simplement connexe suite à
 80 (c) et (e) arbre couvrant des triangles (Source : [4]).

81 L'idée du codeur 3DMC consiste à découper le maillage selon un ensemble d'arêtes pour le
 82 convertir en un polygone planaire simplement connexe. Les sommets selon lesquels le découpage a
 83 été opéré (*i.e.*, sommets dupliqués lors du découpage) définissent l'arbre couvrant des sommets.
 84 L'arbre couvrant des triangles est obtenu en assignant à chaque triangle du polygone planaire un
 85 sommet et à chaque couple de triangles voisins (*i.e.*, partageant une arête commune) une arête. Les
 86 auteurs montrent qu'avec cette nouvelle représentation il est possible de retrouver exactement la
 87 connectivité initiale moyennant un changement d'ordre des sommets et des triangles du maillage. Les

88 deux arbres sont enfin codés en utilisant une technique RLE (*Run Length Encoding*). La Figure 1
89 illustre les principales étapes de codage par arbres couvrants pour un exemple simple de maillage 3D.

90 L'information de géométrie est codée en appliquant une quantification uniforme aux
91 coordonnées du maillage, suivie d'une prédiction linéaire selon la règle du « parallélogramme » [8] et
92 un codage entropique [9] des résidus. En pratique, le codeur 3DMC code la connectivité du maillage
93 en utilisant 2.5-7 bits par sommet (bps). L'information de géométrie est compressée sur 15-20 bps.

94 Cet algorithme n'est en revanche pas directement applicable aux maillages de type non-
95 manifold. Dans ce dernier cas, les maillages doivent tout d'abord être décomposés en plusieurs
96 composantes connexes de type manifold [7], ce qui implique un découpage complémentaire du
97 maillage qui diminue significativement l'efficacité globale du codage [1]. Notons également que le
98 codeur 3DMC requiert que la totalité de l'information décrivant les deux arbres binaires soit chargée
99 en mémoire, ce qui le rend inadapté à une implantation matérielle ou sur terminaux à faible
100 performances de stockage et de calcul.

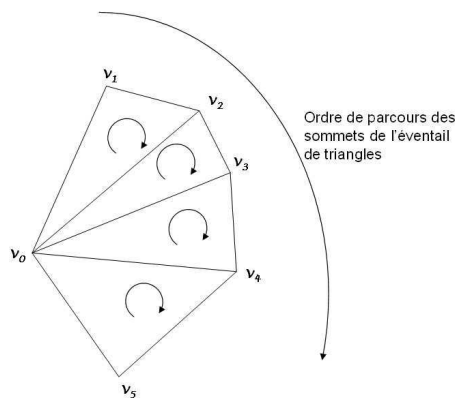
101 2.2 Codage par éventails de triangles

102 L'approche de compression de maillage 3D TFAN, adopté en 2010 par la norme MPEG SC3DMC,
103 repose sur une décomposition des triangles du maillage à coder en un ensemble d'éventails de
104 triangles. Concept central de la représentation proposée, un éventail de triangles est défini comme suit.

105 Un éventail de triangles (*Triangle Fan* - TF) TF de degré d est un ensemble ordonné de d triangles
106 définis par une séquence ordonnée $(t_j)_j$ de $(d + 2)$ sommets $(v_0, v_1, v_2, \dots, v_{d+1})$ tels que :

$$107 \quad \forall j \in \{0, \dots, d-1\}, t_j = \{v_0, v_{j+1}, v_{j+2}\}. \quad (2)$$

108 La Figure 2 illustre un exemple d'éventail de triangles de degré 4.



109

110 **Figure 2 :** Eventail de triangles de degré 4 décrit par les sommets $(v_0, v_1, v_2, v_3, v_4, v_5)$.

111 Par définition, les triangles d'un éventail satisfont les propriétés suivantes :

- 112 • (P1) : les deux triangles successifs d'un éventail sont adjacents, au sens où ils partagent une
113 arête commune,
- 114 • (P2) : les triangles d'un éventail ont la même orientation,
- 115 • (P3) : tous les triangles d'un éventail partagent un même sommet v_0 appelé sommet central.

116 Notons également que l'orientation des triangles implique un unique ordre de parcours des
117 sommets de l'éventail. L'éventail TF est alors complètement déterminé par la suite ordonnée de ses
118 sommets $(v_0, v_1, v_2, \dots, v_{d+1})$, énumérés à partir du sommet central v_0 .

119 Le schéma global de fonctionnement de l'approche de codage TFAN peut être décrit comme suit.
 120 Au départ, les ensembles des sommets et des triangles du maillage sont considérés comme non-visités
 121 (ou non-parcourus). Le codeur TFAN exploite une structure de file \mathcal{F} pour stocker les sommets du
 122 maillage. Au départ la file \mathcal{F} est alimentée par le premier sommet non-visité.

123 A chaque itération, un sommet est extrait de \mathcal{F} . Soit v_j le sommet extrait à l'itération j . Le
 124 sommet v_j est marqué alors comme visité, son nouvel ordre de parcours est stocké dans $O(v_j)$ et les
 125 triangles non-visités qui lui sont incidents sont décomposés en un ensemble d'éventails.

126 Notons que plusieurs stratégies sont possibles pour décomposer les triangles incidents à v_j en une
 127 séquence d'éventails. Dans [5], l'approche itérative décrite ci-dessous a été adoptée.

128 A chaque étape, un éventail $TF_n(j)$ est créé en partant du triangle t_0 ayant le minimum de triangles
 129 voisins (*i.e.*, triangles qui partagent une arête avec t_0) non-visités et de même orientation que le
 130 triangle t_0 . Le triangle t_0 est ensuite ajouté à $TF_n(j)$ et marqué comme visité. Si t_0 a des voisins non-
 131 visités de même orientation alors un voisin t_1 est choisi au hasard, ajouté à $TF_n(j)$ et marqué comme
 132 visité. La même opération est ensuite appliquée à t_1 . Ce processus est réitéré jusqu'à épuisement des
 133 triangles voisins de même orientation et non visités. S'il reste des triangles incidents à v_j , encore non
 134 visités, alors un nouvel éventail est construit en partant encore une fois du triangle qui a un nombre
 135 minimal de voisins non visités et de même orientation.

136 Soient :

- 137 – $(TF_n(j))_{n \in \{1, \dots, N(j)\}}$ l'ensemble des $N(j)$ éventails de triangles associés à v_j ,
- 138 – $\{v_j, w_j^n(1), w_j^n(2), \dots, w_j^n(d(j, n)+1)\}$ les sommets ordonnés de l'éventail $TF_n(j)$ et $d(j, n)$ son
 139 degré.

140 Notons par $L(j)$ la liste des sommets partageant avec v_j au moins un triangle visité du maillage et
 141 ayant un ordre de parcours supérieur à celui de v_j . La liste $L(j)$ est triée selon un ordre croissant en
 142 considérant la relation d'ordre suivante :

$$143 \quad \forall \{w_1, w_2\} \in L(j), \quad w_1 < w_2 \Leftrightarrow O(w_1) < O(w_2). \quad (3)$$

144 A l'exception du sommet v_j , tous les sommets de l'éventail $TF_n(j)$ sont traités itérativement en
 145 respectant l'ordre de parcours défini par $TF_n(j)$. Ici, à chaque sommet $w_j^n(k)$ est associée une valeur
 146 binaire $s_j^n(k)$ indiquant s'il a été visité ou non. Toutes les valeurs $s_j^n(k)$ relatives à un même éventail
 147 sont insérées successivement dans une file $S(j, n)$.

148 Si le sommet $w_j^n(k)$ n'a pas été visité (*i.e.*, $s_j^n(k) = 0$), alors il est marqué comme visité, ajouté à
 149 la structure de file \mathcal{F} et inséré à la fin de la liste $L(j)$. Son ordre de parcours est également stocké dans
 150 $O(w_j^n(k))$. Si au contraire le sommet $w_j^n(k)$ a déjà été visité (*i.e.*, $s_j^n(k) = 1$), deux cas peuvent être
 151 distingués. Si $w_j^n(k)$ appartient à $L(j)$ alors l'index $\mu_j^n(k)$ de $w_j^n(k)$ dans $L(j)$ est stocké dans une file
 152 $I(j, n)$. Sinon, la valeur négative $\mu_j^n(k) = O(v_j) - O(w_j^n(k))$ est stockée dans $I(j, n)$.

153 Rappelons que par définition de la liste $L(j)$, tous ses sommets ont un ordre de parcours (*i.e.*,
 154 nouvel index assigné) supérieur à celui de v_j . Par conséquent, la quantité $\mu_j^n(k)$ est toujours négative.
 155 Cette propriété sera exploitée par le décodeur afin de déduire s'il doit rechercher l'index du sommet
 156 $w_j^n(k)$ dans la liste $L(j)$ ou non (*cf.* Paragraphe 2.3).

157 Une fois tous les sommets de $TF_n(j)$ traités, ses triangles sont marqués comme visités et le
 158 processus réitéré jusqu'à ce que tous les sommets du maillage soient visités. Si la file \mathcal{F} devient vide
 159 et que tous les sommets ne sont pas encore visités, alors le premier sommet non visité est ajouté à \mathcal{F} ,
 160 puis le processus est réitéré.

161 Afin de compresser la représentation TFAN, il est nécessaire de coder de façon compacte pour
 162 chaque éventail $TF_n(j)$, son degré $d(j, n)$ ainsi que les deux listes $S(j, n)$ et $I(j, n)$. Bien naturellement, il
 163 est tout à fait possible de coder directement chacune de ces informations en exploitant un codeur

164 arithmétique. Toutefois, afin d'optimiser encore plus la méthode de codage TFAN, nous proposons de
 165 distinguer les neuf configurations qui sont en pratique les plus fréquentes et de leur associer des
 166 représentations plus compactes. Pour rester générique et prendre en compte tous les autres cas, une
 167 dixième configuration est introduite. Elle correspond au cas où l'ensemble des informations $d(j, n)$,
 168 $S(j, n)$ et $I(j, n)$ est codé.

169 Soit $C(j, n)$ le numéro de la configuration associée à l'éventail $TF_n(j)$. Le Tableau 1 résume les
 170 dix configurations retenues par l'approche TFAN.

171 Les configurations déterminées sont finalement codées à l'aide du codeur arithmétique adapté
 172 [9] afin de tenir compte de leurs propriétés statistiques.

173 **Tableau 2 :** Les dix configurations TFAN (X représente une valeur arbitraire).

Configuration	Information initiale	Information codée	Configuration	Information initiale	Information codée
$C(j, n) = 1$	$d(j, n), S(j, n) = \{1, 0, 0, \dots, 0, 0, 1\}$ et $\mathcal{I}(j, n) = \{1, 2\}$	$C(j, n) = 1$ et $d(j, n)$	$C(j, n) = 6$	$d(j, n), S(j, n) = \{1, 0, 0, \dots, 0, 0, 0\}$ et $\mathcal{I}(j, n) = \{2\}$	$C(j, n) = 6$ et $d(j, n)$
$C(j, n) = 2$	$d(j, n), S(j, n) = \{1, X, X, \dots, X, X, 1\}$ et $\mathcal{I}(j, n) = \{1, X, X, X, 2\}$	$C(j, n) = 2, d(j, n),$ $S'(j, n) = \{X, X, \dots, X, X\}$ et $\mathcal{I}'(j, n) = \{X, X, X\}$	$C(j, n) = 7$	$d(j, n), S(j, n) = \{0, 0, 0, \dots, 0, 0, 0\}$ et $\mathcal{I}(j, n) = \{\}$	$C(j, n) = 7$ et $d(j, n)$
$C(j, n) = 3$	$d(j, n), S(j, n) = \{0, 0, 0, \dots, 0, 0, 1\}$ et $\mathcal{I}(j, n) = \{1\}$	$C(j, n) = 3$ et $d(j, n)$	$C(j, n) = 8$	$d(j, n), S(j, n) = \{1, 0, 0, \dots, 0, 0, 1\}$ et $\mathcal{I}(j, n) = \{2, 1\}$	$C(j, n) = 8$ et $d(j, n)$
$C(j, n) = 4$	$d(j, n), S(j, n) = \{0, 0, 0, \dots, 0, 0, 1\}$ et $\mathcal{I}(j, n) = \{2\}$	$C(j, n) = 4$ et $d(j, n)$	$C(j, n) = 9$	$d(j, n), S(j, n) = \{1, X, X, \dots, X, X, 1\}$ et $\mathcal{I}(j, n) = \{2, X, X, X, 1\}$	$C(j, n) = 9, d(j, n),$ $S'(j, n) = \{X, X, \dots, X, X\}$ et $\mathcal{I}'(j, n) = \{X, X, X\}$
$C(j, n) = 5$	$d(j, n), S(j, n) = \{1, 0, 0, \dots, 0, 0, 0\}$ et $\mathcal{I}(j, n) = \{1\}$	$C(j, n) = 5$ et $d(j, n)$	$C(j, n) = 10$	$d(j, n), S(j, n) = \{X, X, \dots, X, X, X\}$ et $\mathcal{I}(j, n) = \{X, X, X, X, X\}$	$C(j, n) = 10, d(j, n),$ $S'(j, n) = \{X, X, \dots, X, X, X\}$ et $\mathcal{I}'(j, n) = \{X, X, X, X, X\}$

174
175

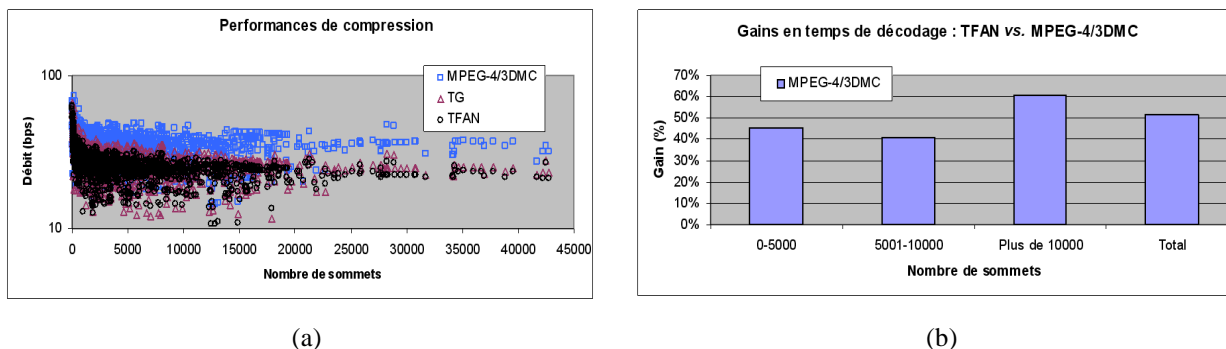
176 La décomposition en éventails de triangles est également exploitée afin d'assurer une prédiction
 177 efficace et de faible coût de calcul des positions et des attributs associés aux sommets du maillage.
 178 Comme dans [8], la règle de prédiction du " parallélogramme " est exploitée en l'appliquant aux
 179 sommets de chaque éventail de la représentation TFAN considérée. Les erreurs résiduelles de
 180 prédiction sont codées en appliquant le codeur arithmétique [9].

181 La même stratégie peut être exploitée pour coder les autres attributs associés aux sommets du
 182 maillage (e.g., couleur, coordonnées de textures, normales...). Toutefois, dans le cas des normales et
 183 des couleurs, la prédiction par la règle du "parallélogramme" est remplacée par une simple prédiction
 184 delta qui donne en pratique de meilleurs résultats [4].

185 2.3 Résultats expérimentaux

186 Le corpus de test du projet SEMANTIC-3D est constitué de plus de 4000 maillages 3D de complexité
 187 et de taille variables, correspondant à divers types de pièces d'une voiture Renault Laguna. Comme
 188 détaillé dans [1], ces données présentent des spécificités topologiques (non-manifolds, multiples
 189 composantes connexes...) qui rendent les approches classiques de codage de maillages 3D soit
 190 inapplicable soit inefficaces. Afin d'évaluer les performances de compression pour des maillages avec
 191 différents nombres de sommets, les débits sont exprimés en bits par sommet (bps).

192 La Figure 3.a présente les débits obtenus suite à la compression de la totalité de la base de
 193 données CAO (i.e., plus de 4000 modèles) avec les trois techniques : TFAN, MPEG-4/3DMC et TG
 194 [9] et pour une quantification sur 12 bits.



195 **Figure 3 :** (a) Performances de compression et (b) gain en temps de décodage.

196 Notons que la technique TG ne s'applique qu'à des maillages manifolds et orientés. Un
 197 processus de conversion des maillages originaux (en maillages manifolds et orientés) [7] a été au
 198 préalable appliqué aux modèles de la base afin de pouvoir les compresser par les codeurs TG et
 199 3DMC.

200 Les gains obtenus par l'approche TFAN par rapport au codeur MPEG-4/3DMC et TG sont
 201 respectivement de l'ordre de 30% et 6%. Ces gains s'expliquent en grande partie par l'efficacité de la
 202 stratégie de codage de la connectivité par éventails de triangles qui permet de représenter de façon
 203 compacte et générique la connectivité de tout maillage triangulaire. En revanche, le codeur MPEG-
 204 4/3DMC code une information auxiliaire pour gérer les maillages non-manifold avec bords, ce qui
 205 peut être coûteux en termes de débit dans le cas de maillages CAO à multiples composantes connexes.

206 Le codeur TFAN offre des résultats de compression légèrement supérieurs à ceux de la
 207 technique TG. En effet, le fait de convertir les maillages CAO en maillages manifolds accentue leurs
 208 singularités et induit la création de nouveaux sommets (voir [1]). En moyenne, le processus de
 209 conversion augmente de 12% le nombre de sommets à coder et par conséquent dégrade les
 210 performances de codage de l'approche TG. Le codeur TFAN s'affranchit d'une telle limitation.

211 Les gains en termes de temps de décodage obtenus par l'approche TFAN par rapport à la
 212 technique MPEG-4/3DMC sont résumés Figure 3.b. Les temps de décodage de l'approche TG ne sont
 213 pas rapportés puisque nous ne disposons pas du code source du décodeur.

214 Ces résultats montrent que le codeur TFAN permet de réduire les temps de décodage de l'ordre
 215 de 50% en moyenne. Pour les maillages comportant plus de 10000 sommets ces gains peuvent
 216 atteindre 61%. Dans le cas de maillages moins denses, les gains obtenus sont de l'ordre de 41-45%. La
 217 vitesse moyenne de décodage de l'approche TFAN est de l'ordre de 499875 sommets par seconde.
 218 Ces résultats confirment la pertinence de l'approche proposée pour des applications nécessitant un
 219 décodage de basse complexité.

220 **3 Compression d'animations 3D**

221 La représentation par trames-clés consiste à stocker l'animation sous forme de séquences de maillages
 222 3D, représentant les trames-clés de l'animation. Les trames intermédiaires nécessaires pour garantir les
 223 cadences vidéo habituelles (25-30 trames/seconde) sont obtenues à l'aide de procédures classiques
 224 d'interpolation. En rendant disponible uniquement le résultat final de l'animation, la représentation par
 225 trames clés permet aux créateurs de contenus de protéger leurs modèles d'animation d'une
 226 réutilisation illicite par des tiers. De plus, cette représentation est indépendante de la technique utilisée
 227 pour générer le contenu et permet ainsi de disposer d'un format générique et multiplateforme.

228 L'inconvénient majeur de cette représentation est en revanche lié aux coûts importants de
 229 stockage et de transmission (de l'ordre de 120 Mo par minute pour un maillage de 10000 sommets).
 230 En effet, même pour de courtes séquences de quelques minutes, des milliers de modèles 3D sont
 231 nécessaires.

232 La problématique de la compression de maillages 3D animés de connectivité constante et de
 233 géométrie variable dans le temps a été considérée pour la première fois par Lengyel [10]. Elle peut
 234 être formalisée mathématiquement comme suit.

235 Soit $(M_t)_{t \in \{1, \dots, T\}}$ une séquence de maillages 3D (où T représente le nombre de trames). Sous l'hypothèse
 236 d'une connectivité fixe, notée Γ , la géométrie du maillage à l'instant t est représentée par une matrice
 237 G_t de dimension $3 \times V$ (où V est le nombre de sommets). L'objectif est alors d'élaborer des
 238 représentations compactes de la séquence géométrique $(G_t)_t$ capables de prendre en compte les
 239 corrélations spatio-temporelles du signal, tout en garantissant le support des fonctionnalités suivantes :

240 • *Streaming* : interpréter le flux binaire au fur et à mesure de sa transmission au terminal en
 241 décodant au fil de l'eau des paquets de données décrivant chacun un segment temporel de l'animation,

242 • Scalabilité spatiale : adapter (par décodage partiel du flux compressé) la résolution spatiale (le
 243 nombre de sommets/triangles) du maillage aux performances d'affichage du terminal et/ou au débit
 244 disponible,

245 • Scalabilité temporelle : décoder partiellement le flux binaire afin de générer une animation à une
 246 cadence vidéo adaptée aux performances de calcul du terminal,

247 • Scalabilité en qualité : adapter (grâce à une structuration adéquate du flux binaire) la qualité (i.e.,
 248 précision des positions des sommets) de l'animation au débit disponible,

249 • Codage des attributs : coder efficacement les attributs associés aux sommets d'un maillage
 250 dynamique (e.g., normales, couleurs, coordonnées de texture...),

251 • Codage quasi-sans pertes : assurer une borne maximale sur l'erreur induite par la compression
 252 des positions et des attributs d'un maillage dynamique.

253 L'élaboration de techniques de compression efficaces avec des fonctionnalités de transmission et
 254 d'affichage adaptées aux contenus dynamiques est aujourd'hui un enjeu majeur comme en témoigne
 255 l'important nombre de travaux de la littérature émergente consacrée à ce sujet (cf. pour un état de l'art
 256 détaillé se référer à [1]). Dans ce qui suit nous nous intéresserons particulièrement au deux codeurs
 257 normalisés IC (*Interpolation Compression*) [11] and FAMC (*Frame-based Animated 3D Mesh*
 258 *Compression*) [12] adoptés par le standard MPEG-4/AFX respectivement en 2003 et 2008.

259 3.1 Codeur AFX-IC

260 Le principe du schéma de compression IC consiste à représenter l'animation par un sous-ensemble de
 261 trames clés sous-échantillonnant la séquence $(M_t)_t$. A partir de ces trames clés, les trames
 262 intermédiaires sont déterminées automatiquement en appliquant une interpolation linéaire. Pour coder
 263 la séquence des trames clés, un mécanisme de prédiction spatio-temporelle est appliqué.

264 Différentes stratégies peuvent être adoptées pour la sélection des trames-clés, depuis le simple
 265 échantillonnage temporel uniforme jusqu'à des techniques plus sophistiquées qui minimisent un
 266 critère d'erreur globale sur l'ensemble de la séquence pour une sélection automatique et optimale.
 267 Ainsi, dans [11] les auteurs proposent-ils de partir d'un ensemble minimal de trames clés constitué de
 268 la première et de la dernière trame de la séquence et de le raffiner itérativement, en ajoutant
 269 successivement des trames clés jusqu'à obtenir un seuil d'erreur d'approximation prédéfini.

270 Par principe de causalité, les approches prédictives nécessitent de considérer un ordre de
 271 parcours des sommets du maillage, spécifiant une relation d'antériorité entre ceux-ci. Cet ordre peut

272 être défini de différentes manières, soit en considérant l'ordre naturel dans lequel les sommets sont
273 spécifiés dans le format de représentation considéré (*e.g.*, VRML), soit en fonction de l'approche de
274 compression statique de la première trame (exploitant le plus souvent les relations d'adjacence entre
275 les sommets du maillage). Quelle que soit l'approche retenue, la prédiction d'un sommet courant v est
276 effectuée à partir d'un ensemble de sommets antérieurs à v par rapport à l'ordre considéré. On parle
277 alors d'une prédiction à partir de sommets déjà codés/décodés.

278 Dans la formulation initiale introduite dans [11], les trois prédicteurs suivants sont proposés :
279 (1) un prédicteur spatial, (2) un prédicteur temporel, et (3) un prédicteur spatio-temporel.

280 Les prédicteurs spatial et temporel correspondent respectivement à une simple prédiction delta par
281 rapport à l'ordre de parcours spatial et temporel. Le prédicteur spatio-temporel combine les deux, en
282 ajoutant au prédicteur temporel un facteur différentiel de correction spatiale.

283 La techniques MPEG-4/AFX-IC présente l'avantage d'un faible coût de calcul, ce qui la rend
284 particulièrement adaptée aux applications de décodage en temps-réel. Toutefois, une règle de parcours
285 déterministe des sommets du maillage est nécessaire, ce qui présente un handicap pour des
286 fonctionnalités plus avancées comme la scalabilité spatiale ou la scalabilité en qualité.

287 3.2 Codeur FAMC

288 Le codeur FAMC combine une stratégie de compensation de mouvement par modèle de peau
289 (*skinning*), à un codage par transformée TCD/ondelettes/ACP des erreurs résiduelles de prédiction. Le
290 schéma synoptique de l'algorithme de compression FAMC est présenté Figure 4.

291 Tout d'abord, la première trame de la séquence est compressée à l'aide du codeur de maillages
292 statiques TFAN.

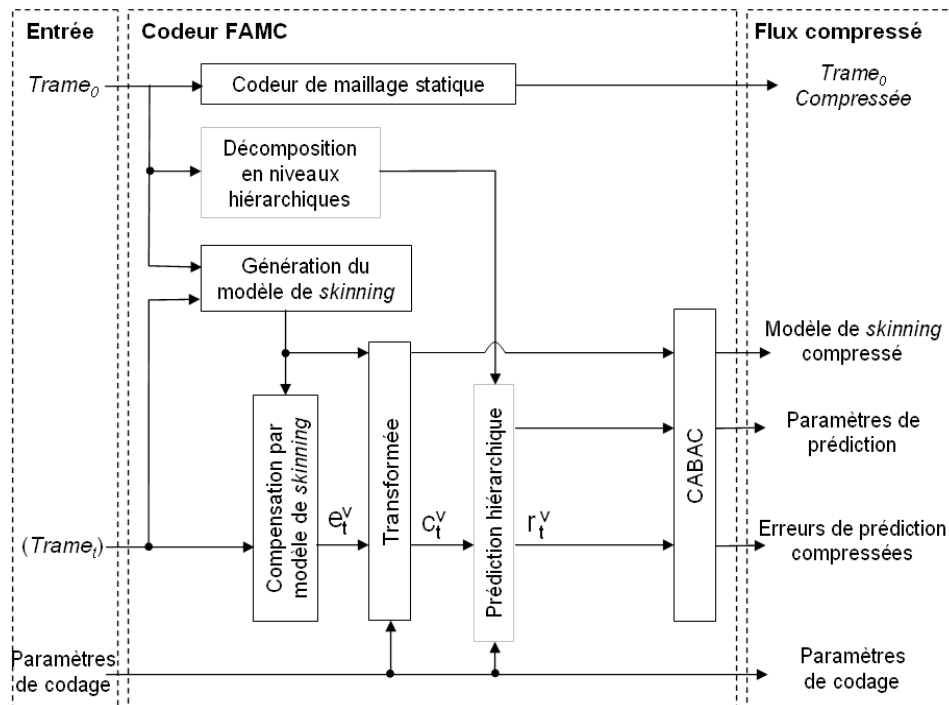
293 Le module de génération du modèle de peau (*skinning*) exploite la procédure de segmentation
294 hiérarchique présentée dans [13]. Ici, les sommets du maillage sont partitionnés en un ensemble de
295 clusters, de façon à ce que le mouvement de chacun puisse être décrit par une unique transformée
296 affine 3D. Un modèle de mouvement affine est alors associé de manière optimale à chaque cluster et
297 pour chaque trame (*cf.* [12]). Ce modèle est ensuite exploité dans l'étape de compensation de
298 mouvement, le mouvement de chaque sommet étant exprimé comme une combinaison linéaire
299 pondérée des mouvements des différentes parties (*cf.* [12]).

300 Notons, que le mouvement affine est toujours estimé par rapport à la première trame de
301 l'animation afin de :

- 302 1. Permettre un accès aléatoire aux trames de l'animation,
- 303 2. Gérer efficacement les pertes d'information dans le contexte de transmission en
304 environnements bruités.

305 Les erreurs résiduelles de compensation de mouvement sont finalement codées à l'aide d'une
306 transformée temporelle ou spatio-temporelle. Une dernière étape de prédiction hiérarchique peut être
307 optionnellement appliquée. Celle-ci exploite une décomposition en niveaux hiérarchiques du maillage,
308 adaptée à une prédiction spatio-temporelle locale qui assure la scalabilité spatiale du flux FAMC (*cf.*
309 [14]).

310



311

312

Figure 4 : Schéma synoptique du codeur FAMC. En grisé, les modules optionnels.

313

Les transformées suivantes ont été retenues par la norme :

314

- l'identité pour permettre un codage uniquement par prédiction hiérarchique,

315

- la DCT (*Discrete Cosine Transform*) bien connue pour ses propriétés de concentration de l'énergie du signal en un nombre réduit de coefficients,

316

317

- la transformée en ondelettes (TO) implantée par schéma de lifting, pour sa capacité d'allier compacité de représentation et faible complexité de calcul, ainsi que pour le support du codage quasi-sans perte,

318

319

320

- une décomposition par ACP, pour l'optimalité de la représentation.

321

Quel que soit le type d'information (paramètres du modèle de *skinning*, coefficients spectraux...), les différents éléments de la représentation FAMC sont codés à l'aide du codeur arithmétique binaire adaptatif avec contextes - CABAC (Context-based Adaptive Binary Coding) [15].

322

323

324

Notons la grande flexibilité de l'architecture modulaire du codeur FAMC qui rend possible l'intégration de plusieurs méthodes, pouvant répondre à différentes contraintes d'application.

325

326

3.3 Résultats expérimentaux

327

Le corpus de test ainsi que les critères objectifs d'évaluation ont été fixés dans le cadre des expérimentations techniques (*Core Experiments*) [16] du standard MPEG-4 AFX. La base de test retenue est composée d'une trentaine de maillages dynamiques (Tableau 1) de différentes formes et complexités, animés par des mouvements variés, qu'ils soient articulés ou élastiques.

328

329

330

331

La mesure de distorsion considérée, introduite dans [11] et notée D_w , représente une approximation de

332

la distorsion moyenne de la séquence décodée.

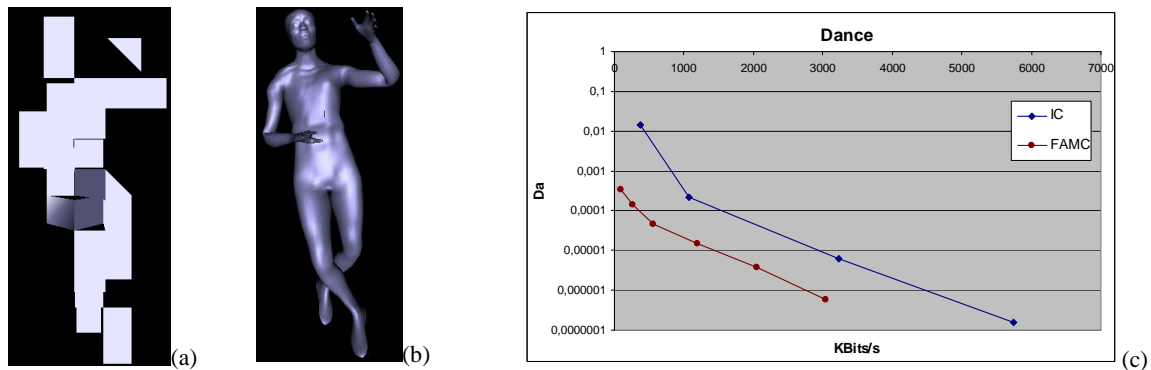
333 Le facteur de gain G de l'approche FAMC par rapport à la technique IC est exprimé
 334 par $G = \left(1 - \frac{T(FAMC)}{T(IC)}\right)$, où $T(FAMC)$ et $T(IC)$ représentent respectivement la taille (en kbits) des flux
 335 binaires obtenus par les codeurs FAMC et IC pour une même distorsion. Une étude comparative par
 336 rapport aux autres méthodes de la littérature est présentée dans [1].

337 Le tableau 2 présente les gains obtenus par l'approche FAMC par rapport au codeur IC. En moyenne,
 338 l'approche FAMC des débits 60% plus bas.

339 **Tableau 2** : Corpus de test et performances de compression FAMC vs. IC (V : nombre de sommets, T : nombre
 340 de trames et K : nombre *patches* pour le modèle de *skinning*).

Animation	V	T	K	G(%)	Animation	S	T	K	G(%)
Box	687	29	17	66	Jump	15830	222	111	72
Camel_collapse	21877	53	96	74	Mr_Gik	10589	22	22	80
Camel_gallop	21877	48	31	76	Rabbit	434	19	13	46
Chicken	3030	400	40	77	Raptor	4952	41	26	88
Cow	2904	204	30	71	Shark	826	41	7	90
Dance	7061	201	22	86	Sphere	482	24	1	2
Dish	1730	26	1	71	TableCloth	29791	31	8	-38
Dolphin	6180	101	11	91	Troll	4961	36	63	83
Dragon	3211	38	14	69	Wolf	2320	31	29	75
Eagle	636	31	16	62	Ball	5552	100	1	99
Elephant_gallop	42321	48	27	72	Flex	1743	45	15	40
Ffd	434	13	1	24	Ripple	758	24	2	36
Flag	1381	31	2	37	Snake	9179	134	36	95
Horse_collapse	8431	53	129	60	Twist	836	21	2	40
Horse_gallop	8431	48	36	64	Humanoid	7646	154	20	94

341 Dans le cas de maillages dynamiques avec déformation élastique d'amplitude importante (e.g.
 342 "Camel_collapse", "Horse_collapse" et "Troll"), des modèles de *skinning* complexes sont nécessaires pour
 343 décrire l'animation ($K=96, 129, 63, \dots$). Malgré cela, les gains obtenus par FAMC sont importants puisque
 344 de l'ordre de 70%. Le codeur FAMC se révèle toutefois moins performant sur les séquences animées avec
 345 des mouvements élastiques de faible amplitude (e.g. "Flex", "Sphere" et "Tablecloth"). Dans ce cas, le
 346 modèle de mouvement par *skinning* n'apporte pas de valeur ajoutée, la compression étant obtenue via le
 347 codage par DCT/Ondelettes. Les performances de codage varient selon les modèles et sont comprises entre
 348 -38% et 40%. La technique FAMC est extrêmement efficace dès lors qu'elle est appliquée à des maillages
 349 dynamiques articulés (e.g., « Humanoid », « Dance » et « Camel_gallop ») : les gains obtenus sont de
 350 l'ordre de 80%. Pour ce type d'animation, la procédure de compensation de mouvement par *skinning*
 351 permet de modéliser fidèlement le champ de mouvement avec un nombre réduit de *patches* ($K=20, 22,$
 352 $31, \dots$) ce qui conduit à des séquences reconstruites de haute qualité à partir de très bas débits. La figure 5,
 353 illustre, sur l'animation « Dance », la supériorité du codeur FAMC par rapport à la technique IC.
 354



355 **Figure 5** : Séquence «Dance» compressée avec IC à 255 kbits/s (a), FAMC à 95 kbits/s(b). Courbes débit
 356 distorsion FAMC vs. IC (c).

357 4 Conclusion

358 Dans cet article nous avons présenté une étude approfondie des différentes technologies MPEG pour
 359 la compression de maillages 3D statiques et dynamiques.

360 La comparaison conduite sur les corpus de test de SEMANTIC-3D de MPEG-4, établit
 361 objectivement la supériorité des codeurs TFAN et FAMC par rapport aux approches 3DMC et IC. En
 362 termes de performances débit/distorsion, le codeur TFAN offre jusqu'à 30% de gains par rapport à
 363 3DMC avec des temps de décodage deux fois moins importants. Le codeur FAMC permet de gains de
 364 l'ordre de 60% en moyenne tout offrant un ensemble de complet de fonctionnalités avancées de
 365 *streaming*, de scalabilité (spatiale, temporelle et en qualité), ou encore de codage quasi-sans pertes.

366 Dans nos futures travaux nous nous intéresserons aux travaux actuellement conduit par le
 367 groupe MPEG-4/3DG (*3D Graphics*) qui prépare la future norme de compression multi-résolution
 368 MR3DMC (*Multi-Resolution 3D Mesh Coding*) dédiée aux maillages 3D denses.

369 Références

- 370 [1] K. Mamou. Compression de maillages 3D statiques et dynamiques, *Ph.D. Dissertation Université Paris V*
 371 *- René Descartes*, France, September 2008.
- 372 [2] Information Technology – Coding of Audio-Visual Objects – Part 16: Animation Framework eXtension, ISO/IEC
 373 14496-16:2003.
- 374 [3] J. Peng, C.-S. Kim, C.-C. J. Kuo. Technologies for 3D mesh compression: A survey, *ELSEVIER Journal of Visual*
 375 *Communication and Image Representation*, 16(6), 688-733, 2005.
- 376 [4] G. Taubin and J. Rossignac. Geometric compression through topological surgery. *Transactions on*
 377 *Graphics*, 17(2): 84-115, 1998.
- 378 [5] K. Mamou, T. Zaharia, F. Prêteux. TFAN: A Low Complexity 3D Mesh Compression Algorithm. *Journal*
 379 *of Computer Animation and Virtual Worlds*, vol. 20(2-3), 343-354, 2009.
- 380 [6] G. Turan. On the succinct representations of graphs. *Discrete Applied Mathematics*, 8 :289–294, 1984.
- 381 [7] F. Lazarus and A. Guézic and G. Taubin and B. Horn. Cutting and stitching : converting sets of polygons
 382 to manifold surfaces. *IEEE Transaction on Visualization and Computer Graphics*, 7(2) :136–151, 2001.
- 383 [8] C. Touma and C. Gotsman. Triangle mesh compression. *Graphics Interface*, 25-34, 1998.
- 384 [9] A. Moffat, R. Neal, et I. Witten. Arithmetic coding revisited. *ACM Transactions on Information Systems*,
 385 16(3) :256–294, 1998.

- 386 [10] J. Lengyel. Compression of time-dependent geometry. In ACM Symposium on Interactive 3D Graphics,
387 pages 89–96, Atlanta, E tats-Unis, 1999.
- 388 [11] E. S. Jang, J. D. K. Kim, S. Y. Jung, M. J. Han, S. O. Woo, et S. J. Lee. Interpolator data compression for
389 MPEG-4 animation, *Circuits and Systems for Video Technology*, 14(7):989–1008, Juillet 2004.
- 390 [12] Khaled Mamou, Titus Zaharia, Françoise Prêteux. A skinning approach for dynamic 3D mesh
391 compression. *Computer Animation and Virtual Worlds*, Vol. 17(3-4), p. 337-346, Juillet 2006.
- 392 [13] K. Mamou, T. Zaharia, F. Prêteux. A skinning prediction scheme for dynamic 3D mesh compression.
393 *SPIE Conference on Mathematics of Data/Image Pattern Recognition, Compression, and Encryption with*
394 *Applications IX*, San Diego, CA, Vol. 6315, 1-12, 2006.
- 395 [14] K. Mamou, T. Zaharia, F. Prêteux, N. Stefanoski, J. Ostermann. Frame-based compression of animated
396 meshes in MPEG-4. *IEEE International Conference on Multimedia & Expo (ICME'2008)*, Hannover,
397 Germany, 2008.
- 398 [15] H. Kirchhoffer, D. Marpe, K. Muller, T. Wiegand. Context-adaptive binary arithmetic coding for frame-
399 based animated mesh compression. *IEEE International Conference on Multimedia and Expo*, 341-344,
400 2008.
- 401 [16] M. Preda. 3D Graphics Compression Core Experiments Description, ISO/IEC JTC 1/SC 29/WG 11
402 N8499, Hangzhou, China, October 2006.
- 403 [17] F. Moran, P. Gioia, M. Stelias, M. Bourges-Sevenier, N. Garcia. Subdivision surfaces in MPEG-4. *IEEE*
404 *Conference on Image Processing*, 5-8, 2002.
- 405 [18] I. A. Salomie , A. Munteanu , A. Gavrilescu , G. Lafruit , P. Schelkens, R. Deklerck, J. Cornelis.
406 MESHGRID - A compact, multi-scalable and animation-friendly surface representation. *IEEE Conference*
407 *on Image Processing*, 13-16, 2002.
- 408