

Segmentation au sens du mouvement de maillages dynamiques

Khaled MAMOU, Titus ZAHARIA et Françoise PRÊTEUX

Groupe des Ecoles des Télécommunications, Institut National des Télécommunications, Département ARTEMIS

9 rue Charles Fourier, 91011 Evry Cedex

{Khaled.Mamou, Titus.Zaharia, Francoise.Preteux}@int-evry.fr

Résumé – Cet article décrit une nouvelle méthode de segmentation au sens du mouvement de maillages 3D dynamiques. L'approche repose sur un principe de décimation hiérarchique sous contrainte d'erreur quadratique moyenne de compensation de mouvement. La segmentation obtenue est exploitée dans l'étape de compensation de mouvement par le codeur d'animation 3D FAMC (*Frame-based Animated Mesh Compression*) développé par les auteurs et récemment promu (Janvier 2007) dans le standard MPEG-4 en tant qu'amendement 2 de la partie 16 – AFX (*Animation Framework eXtension*). Les résultats expérimentaux montrent des gains en débits significatifs (de 20% à 80%) de l'approche FAMC par rapport aux techniques de l'état de l'art.

Abstract – We introduce a novel motion-based segmentation technique for dynamic 3D meshes. The key contribution is a hierarchical segmentation procedure integrating motion compensation quadratic error constraint. The obtained segmentation is exploited within the skinning-based motion compensation strategy of the 3D animation encoder FAMC (*Frame-based Animated Mesh Compression*), developed by the authors and recently promoted (January 2007) within the MPEG-4 standard as amendment 2 of part 16 – AFX (*Animation Framework eXtension*). The experimental results show that the FAMC technique provides significant gains in terms of bitrates (20% -80%) with respect to the state of the art.

1. Introduction

Les contenus dynamiques 3D s'imposent aujourd'hui dans le monde du multimédia au travers d'applications phares dans le monde des jeux vidéos, des films d'animation ou encore de la création d'effets spéciaux en cinématographie.

Quelle que soit la technique utilisée pour les créer (modèles physiques, modèles de peau, méthodes de capture de mouvement...), ces contenus sont représentés le plus souvent comme des séquences de maillages 3D de topologie (connexité) fixe et de géométrie variable. Cela conduit à des représentations extrêmement lourdes (dues aux cadences vidéos habituelles de 25 à 30 objets 3D par seconde) en termes de stockage, transmission et visualisation. Il est alors nécessaire d'élaborer des techniques de représentation efficaces et adaptées à ces contenus 3D dynamiques. Dans ce contexte, la modélisation par modèle de peau (*skinning*) joue un rôle clé. Compacte et intuitive, elle rend possibles des applications multiples, comme la compression/transmission, le rendu en temps-réel ou encore la réutilisation et le clonage d'animations 3D. Un modèle de peau est défini comme une partition du maillage associée, via des coefficients d'influence, à un modèle d'animation (le plus souvent décrit par des transformations affines 3D). L'exemple classique concerne les modèles d'avatars virtuels, définis comme des objets 3D où les

segments de peau sont associés à des os dans une structure de squelette qui contrôle l'animation.

Dans cet article, nous proposons une nouvelle méthode de segmentation au sens du mouvement de maillages 3D dynamiques, spécifiquement conçue et adaptée pour la création de modèles de peau à partir de séquences d'animation arbitraires.

Après une revue de l'état de l'art (paragraphe 2), l'approche proposée est décrite en détails dans le paragraphe 3. Son application à des objectifs de compression est présentée, discutée et validée expérimentalement au paragraphe 4. Enfin le paragraphe 5 conclut l'article et ouvre des perspectives de recherches futures.

2. Etat de l'art

La segmentation au sens du mouvement a tout d'abord été considérée dans le cadre des méthodes de compression d'animation 3D, où un modèle de peau est utilisé pour l'étape de compensation de mouvement.

L'approche heuristique introduite par Lengyel [1] sélectionne au hasard 10% des triangles du maillage et classe le reste des sommets par rapport à leur mouvement. Son inconvénient majeur est lié aux sur-segmentations qui en résultent et qui pénalisent fortement son utilisation à des objectifs de compression 3D. Dans [2], les auteurs s'inspirent de l'approche de Lengyel pour modéliser le

mouvement des sommets uniquement par des transformées rigides, en introduisant une nouvelle procédure de segmentation fondée sur une minimisation au sens des moindres carrés. Toutefois, cette méthode montre ses limites dans le cas de mouvements non-rigides. Mamou *et al.* [3] proposent une autre extension de l'approche de Lengyel mettant en œuvre une segmentation fondée sur l'algorithme des *k*-moyennes (*k-means*). L'inconvénient principal de la méthode concerne le nombre de clusters qui doit être spécifié par l'utilisateur. Dans [4], un modèle de *skinning* est calculé afin d'accélérer le rendu et la détection de collisions. L'approche proposée exploite l'algorithme *mean-shift* pour détecter les parties quasi-rigides du maillage animé. Si la technique est efficace dans le cas des mouvements rigides, elle reste non-adaptée à des déformations élastiques.

Contrairement aux méthodes de la littérature, la nouvelle technique de segmentation hiérarchique de maillages dynamiques 3D proposée ici permet de gérer efficacement des animations à la fois articulées et élastiques. Détaillons-en à présent les principales étapes.

3. Approche proposée

Soit $(M^t)_{t \in \{1, \dots, F\}}$ une séquence de maillages 3D (F étant le nombre de trames), de connexité constante et de géométrie variable dans le temps, notée $(G^t)_{t \in \{1, \dots, F\}}$. La géométrie du maillage $G_t = (\mathcal{X}_t^v)_{v \in \{1, \dots, V\}}$ à l'instant t est représentée par le vecteur des coordonnées homogènes $\mathcal{X}_t^v = (x_t^v, y_t^v, z_t^v, 1)$ de tous les sommets de M^t (V étant le nombre de sommets).

Segmenter la séquence $(M^t)^t$ au sens du mouvement revient alors à déterminer une K -partition $\Pi = (\pi_k)_{k \in \{1, \dots, K\}}$ des sommets de $(M^t)^t$ telle que le mouvement de chaque partie π_k soit fidèlement représenté par une transformée affine 3D. Le critère de fidélité de la représentation considéré est l'erreur quadratique moyenne de compensation de mouvement, notée $E(\Pi)$ et définie par :

$$E(\Pi) = \sum_{t=1}^F \sum_{v=1}^V \left\| \mathcal{X}_t^v - A_t^{k(v)} \mathcal{X}_1^v \right\|^2, \quad (1)$$

où :

- $k(v)$ est l'index de la partie à laquelle le sommet v appartient (*i.e.* $v \in \pi_{k(v)}$), et
- $A_t^{k(v)}$ représente la transformée affine 3D associée à la partie $\pi_{k(v)}$ à la trame t , définie par :

$$A_t^{k(v)} = \arg \min_A \sum_{p \in \pi_{k(v)}} \left\| A \mathcal{X}_1^p - \mathcal{X}_t^p \right\|^2. \quad (2)$$

L'objectif est de déterminer, pour une erreur fixée E_0 , une partition Π^* ayant le nombre de *patches* minimal K et vérifiant la contrainte $E(\Pi^*) \leq E_0$.

L'algorithme proposé dans cet article est fondé sur une stratégie de décimation privilégiant la simplification de sommets ayant le même mouvement affine. Le principe consiste à appliquer successivement des opérations de décimation topologique de type *half-edge collapse* [15] (Figure 1). Une opération de *half-edge collapse* appliquée à une arête (v, w) , notée $hecol(v, w)$, fusionne les deux sommets v et w en un seul sommet. Le sommet w est alors éliminé et toutes ses arêtes incidentes sont connectées à v . A chaque étape du processus de simplification, pour tout sommet v , la liste de ses ancêtres, notée v^* , est stockée et définie récursivement par :

1. A l'initialisation, v^* est vide.
2. A chaque $hecol(v, w)$, mise à jour de v^* comme suit : $v^* \leftarrow v^* \cup w^* \cup \{w\}$.

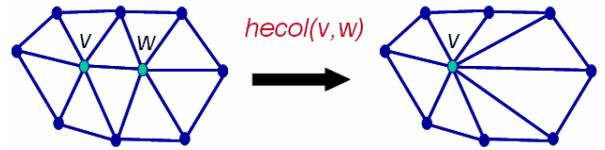


FIG. 1 : Opération *half-edge collapse* : l'arête (v, w) est contractée et les deux sommets v et w fusionnés.

Le processus de décimation est guidé par une stratégie visant à fusionner les sommets avec des mouvements affines proches. Le critère de fusion est lié au coût $C(v, w)$ de fusion de l'arête (v, w) défini par :

$$C(v, w) = \sum_{t=1}^F \left(\sum_{p \in V(v, w)} \left\| A_t^{v, w} \mathcal{X}_1^p - \mathcal{X}_t^p \right\|^2 \right), \quad (3)$$

avec $V(v, w) = v^* \cup w^* \cup \{v, w\}$, et

$$A_t^{v, w} = \arg \min_A \sum_{p \in V(v, w)} \left\| A \mathcal{X}_1^p - \mathcal{X}_t^p \right\|^2. \quad (4)$$

A chaque étape du processus de décimation, l'opération *hecol* est appliquée à l'arête de plus faible coût. Une partition $\Pi^l = (\pi_k^l)_{k \in \{1, \dots, K\}}$ est alors obtenue par :

$$\forall k \in \{1, \dots, K\}, \pi_k^l = \{p_k^l\} \cup p_k^{l*}. \quad (5)$$

où $\{p_k^l\}$ représente l'ensemble des sommets du maillage obtenu à l'étape l du processus de simplification.

Le processus est itéré tant que $E(\Pi^l)$ reste inférieure à la valeur prédéfinie E^0 .

Notons que l'approche de segmentation hiérarchique décrite ci-dessus traite chaque Composante Connexe (CC) du modèle séparément des autres ce qui peut conduire à des sur-segmentations dans le cas de modèles animés à multiples CCs et ayant des mouvements affines proches (*i.e.*, chaque CC est considérée comme un cluster à part).

Afin de résoudre ce problème de sur-segmentation, un processus de simplification itérative qui élimine à chaque étape la partie induisant une erreur minimale, est appliqué. L'erreur minimale induite par la suppression d'une partie k est calculée pour la partition obtenue en assignant tous les sommets de k aux autres parties et en appliquant ensuite une procédure de raffinement qui vise à optimiser la

partition sans changer le nombre de parties. Le principe consiste à alterner itérativement une phase d'estimation des transformées affines et une phase de mise à jour de la partition. La phase de calcul des transformées affines associe à chaque partie π_k la transformée affine A_i^k qui décrit le mieux son mouvement conformément à l'équation (2). La phase de mise à jour associe chaque sommet v à la partie k_0 qui décrit le mieux son mouvement,

$k_0 = \arg \min_{k \in \{1, \dots, K\}} \sum_{i=1}^F \|A_i^k \chi_i^v - \chi_i^v\|^2$. Ce processus est itéré

jusqu'à convergence (pas de changement de la partition) ou lorsqu'un nombre fixé d'itération est atteint.

La figure 2 présente les résultats de segmentation obtenus pour les maillages dynamiques « Dance », « Snake », « Chicken », « Horse_gallop » et « Raptor ». Notons que, dans le cas des personnages articulés, les différentes parties du corps sont bien déterminées. Pour l'animation élastique « Snake », dix parties seulement suffisent pour représenter le mouvement.

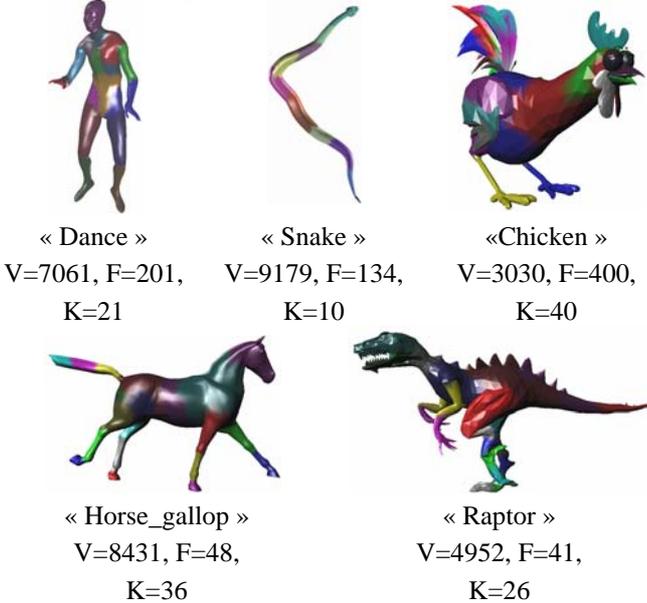


FIG. 2 : Segmentation de maillages dynamiques de mouvements, formes et complexités variés.

4. Application à la compression

Disposer d'une partition adéquate permet de construire directement un modèle de peau, qui peut être exploité dans le cadre des applications de compression de séquences d'animation.

Détaillons tout d'abord le modèle de peau associé à la partition précédemment déterminée.

4.1 Compression par modèle de peau

Le principe est d'exprimer le mouvement d'un sommet sous forme d'une combinaison linéaire pondérée de mouvements des *patches* du modèle. La position prédite $\hat{\chi}_t^v$ d'un sommet $v \in \{1, \dots, V\}$ à un instant $t \in \{1, \dots, F\}$ est ainsi donnée par :

$$\hat{\chi}_t^v = \sum_{k=1}^K \omega_k^v A_i^k \chi_i^v, \quad (6)$$

où ω_k^v est un coefficient réel qui contrôle l'influence du *patch* k sur le mouvement du sommet v .

Le vecteur optimal des poids $\omega^v = (\omega_k^v)_{k \in \{1, \dots, K\}}$ est exprimé par :

$$\omega^v = \arg \min_{\alpha \in \mathbb{R}^K} \sum_{i=1}^F \left\| \sum_{k=1}^K \alpha_j A_i^k \chi_i^v - \chi_i^v \right\|. \quad (7)$$

En pratique, il est raisonnable de considérer qu'un cluster influence uniquement ses propres sommets et ceux appartenant à un cluster voisin. Soient $k(v)$ le cluster contenant le sommet v et $\theta(v)$ l'ensemble des clusters voisins incluant $k(v)$. L'équation (7) est résolue sous les contraintes suivantes :

$$\forall k \notin \theta(v), b_k = 0. \quad (8)$$

L'équation (7) avec les contraintes (8) conduit à un problème de minimisation au sens des moindres carrés, résolu par une méthode de pseudo-inverse [14].

Les erreurs de prédiction après compensation du mouvement par modèle de peau, notées $e_t^v = (e_t^{v,x}, e_t^{v,y}, e_t^{v,z})$ sont définies par :

$$\forall v \in \{1, \dots, V\}, e_t^v = \chi_t^v - \hat{\chi}_t^v. \quad (9)$$

Pour chaque sommet v du maillage dynamique, les erreurs de prédiction associées sont considérées comme trois séquences temporelles $(e_t^{v,x})_t$, $(e_t^{v,y})_t$ et $(e_t^{v,z})_t$ ($t \in \{1, \dots, F\}$). Les spectres $(s_p^{v,x})_p$, $(s_p^{v,y})_p$ et $(s_p^{v,z})_p$ de chaque séquence sont alors déterminés en appliquant une Transformée en Cosinus Discrète (TCD) monodimensionnelle temporelle.

Cette procédure est appliquée à l'ensemble des sommets du maillage. Les coefficients spectraux obtenus sont enfin multiplexés dans un seul vecteur S de dimension $3 \times V \times F$, défini comme suit :

$$S = \prod_{p \in \{1, \dots, F\}} \prod_{v \in \{1, \dots, V\}} (s_p^{v,x}, s_p^{v,y}, s_p^{v,z}), \quad (10)$$

où \prod désigne l'opérateur de concaténation.

Le vecteur S ainsi obtenu est finalement codé à l'aide du codeur arithmétique CABAC [13], comme décrit dans [5].

4.2 Résultats expérimentaux

Les expérimentations ont été conduites sur la base de test MPEG-4 AFX [5], composée d'un ensemble de 30 modèles animés de géométries, complexités (400 à 42000 sommets et de 21 à 784 trames) et mouvements différents (rigides, rigides par morceaux et élastiques).

La figure 3 présente les courbes débit/distorsion obtenues pour notre codeur *skinning hiérarchique* et les méthodes *skinning K-means* [3], *Dynapack* [6], *MCGV* [7], *RT* [2], *GV* [8], *AFX-IC* [9] et *D3DMC* [12] pour les séquences « Chicken » et « Dance ». Ici, les débits sont

exprimés en bits par sommet par trame (bpst). Les distorsions sont en terme d'erreur RMSE définie dans [10].

Les résultats de la Figure 3 montrent que l'approche de codage proposée offre des gains allant de 20% à 80% par rapport aux techniques de l'état de l'art. Ce bon comportement se maintient sur l'ensemble de la base, avec en gain de 60% en moyenne par rapport à la technique IC (*Interpolation Compression*) précédemment retenue dans le standard MPEG-4 AFX.

En raison des performances objectivement établies [5], la méthode de compression par modèle de peau a été retenue en Janvier 2007 dans le standard MPEG-4 AFX en tant qu'amendement 2 à la partie 16 et sous le nom de FAMC (*Frame-based Animated Mesh Compression*) [11].

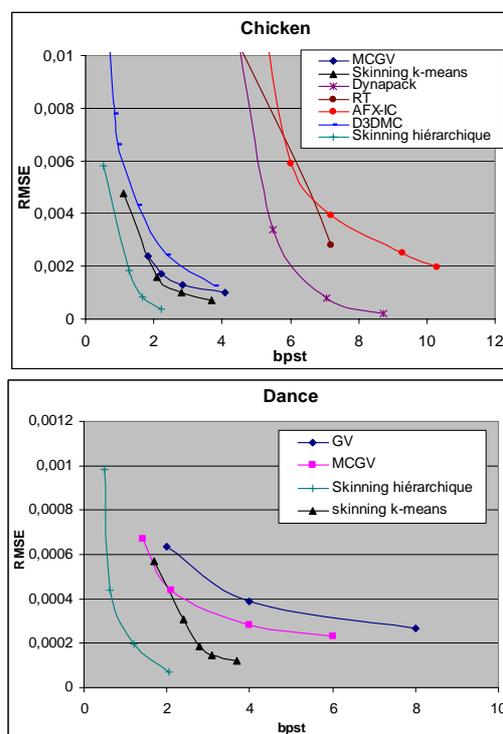


FIG. 3 : Courbes débit/distorsion : approches *skinning hiérarchique*, *skinning K-means*, MCGV, RT, GV, D3DMC, Dynapack et AFX-IC.

5. Conclusion et perspectives

Dans cet article, nous avons présenté une nouvelle technique de segmentation hiérarchique de maillages dynamiques au sens du mouvement. Le résultat de segmentation est exploité dans l'étape de compensation de mouvement par modèle de peau du codeur FAMC développée par les auteurs et promu dans le standard MPEG-4 en tant que amendement 2 de la partie 16 – AFX (*Animation Framework eXtension*). Les résultats expérimentaux montrent des gains en débits significatifs (de 20% à 80%) par rapport à l'état de l'art.

Les perspectives de recherche concernent principalement l'étude théorique de la vitesse de

convergence ainsi que l'optimisation de la complexité de calcul de l'approche de segmentation proposée.

Références

- [1] LENGYEL J., «Compression of time-dependent geometry», *ACM Symposium on Interactive 3D Graphics*, p. 89-96, 1999.
- [2] COLLINS G., HILTON A., «A rigid transform basis for animation compression and level of detail», *ACM Symposium on Interactive 3D Graphics*, p. 21–29, 2005.
- [3] MAMOU K., ZAHARIA T., PRETEUX F., «A skinning approach for dynamic 3D mesh compression», *Computer Animation and Virtual Worlds*, 17(3-4), p. 337-346, 2006.
- [4] DOUG L. J., CHRISTOPHER D. T., «Skinning mesh animations», *ACM Tran. on Graphics*, 24(3), 2005.
- [5] MAMOU K., ZAHARIA T., PRETEUX F., «Results of evaluation experiment EE1 on static and animated 3D mesh coding: skinning-based dynamic mesh compression», *ISO/IEC JTC1 SC29/WG11 M13888*, Hangzhou, 2006.
- [6] IBARRIA L., ROSSIGNAC J., «Dynapack: space-time compression of the 3D animations of triangle meshes with fixed connectivity», *Proceeding Eurographics'03*, p.126-135, 2003.
- [7] MAMOU K., ZAHARIA T., PRETEUX F., «Multi-chart geometry video: a compact representation for 3D animations», *Symposium on 3D Data Processing, Visualization and Transmission*, 2006.
- [8] BRICENO H. M., SANDER P. V., MCMILLAN L., GORTLER S., HOPPE H., «Geometry videos: a new representation for 3D animations», *ACM Siggraph Symposium on Computer Animation*, p. 136-146, 2003.
- [9] JANG E. S., KIM J. D. K., JUNG S. Y., HAN M. J., WOO S. O., LEE S. J., «Interpolator data compression for MPEG-4 animation», *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, p. 989-1008, 2004.
- [10] ASPERT N., SANTA-CRUZ D., EBRAHIMI T., «MESH: Measuring Errors between Surfaces using the Hausdorff distance», *IEEE International Conference in Multimedia and Expo (ICME)*, p. 705-708, 2002.
- [11] PREDA M., ZAHARIA T. (Editors), WD 1.0 of ISO/IEC 14496-16:2006/AMD2 (Frame-based Animated Mesh Compression), *ISO/IEC JTC1 SC29/WG11 N8732*, 2007.
- [12] MULLER K., SMOLIC A., KAUTZNER M., EISERT P., WIEGAND T., «Predictive compression of dynamic 3D meshes», *IEEE International Conference on Image Processing*, p. 621-624, 2005.
- [13] MARPE D., SCHWARZ H., WIEGAND T., «Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard», *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7), pp. 620- 636, 2003.
- [14] PRESS W. H., TEUKOLSKY S. A., VETTERLING W. T., FLANNERY B. P., «Numerical recipes in C: the art of scientific computing», *Cambridge Press*, 2^{ème} édition, 1992.
- [15] HOPPE H., «Progressive meshes», *Computer Graphics*, Vol. 30(Annual Conference Series), pp. 99-108, 1996.