

# Codage vidéo basé objet à granularité fine

Dubhe CHAVIRA-MARTÍNEZ, Stéphane PATEUX

IRISA/INRIA

Campus de Beaulieu, 35042 Rennes, France

dchavira@irisa.fr, spateux@irisa.fr

**Résumé** – Dans cet article, nous présentons un schéma de codage vidéo scalable basé texture pour des fonctionnalités de robustesse. Les différentes images sont codées à l’aide d’un codage de type analyse-synthèse avec une couche de raffinement scalable. La robustesse du schéma est accrue grâce à la possibilité de décoder indépendamment les différentes images de la vidéo.

**Abstract** – In this paper, we present a scalable texture-based video coding scheme for robust functionalities. To this purpose, the different images are coded using an analysis-synthesis coding scheme with a progressive refinement layer. The robustness is increased thanks to the ability to decode independently the different images of the video.

## 1 Introduction

Les premières normes de compression vidéo tel que MPEG-1, MPEG-2 et H.263 sont consacrées à des applications de stockage et de transmission d’information. H263+ ainsi que la norme MPEG-4 considèrent en plus la transmission d’informations multimédias sur des supports plus hétérogènes et à qualité de service variable. Un défaut intrinsèque des schémas de codage vidéo classiques (MPEG-x et H26x) est l’utilisation d’un schéma en boucle fermée (cf. figure 1 a). Ainsi si l’on considère des flux progressifs à différents niveaux de qualité, MPEG-1, MPEG-2 et H.263 offrent une scalabilité et une robustesse limitées. Toute information perdue empêche le décodage des informations suivantes dans le flux vidéo. Ainsi, si une couche haute de qualité est détériorée, il faudra attendre la prochaine référence intra pour se raccrocher à la couche de haute qualité. MPEG-4 FGS a alors été développé afin de pallier à ce défaut en assurant la transmission d’une couche de base minimale et en utilisant un raffinement codé en mode progressif pour chaque image en couche haute (cf. Fig. 1 b). Toutefois, lié à une couche basse de trop faible qualité, MPEG4-FGS souffre d’une dégradation notable en performance de compression.

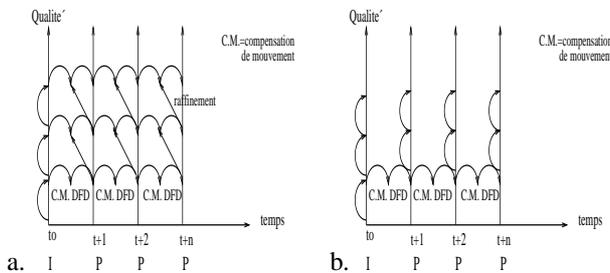


FIG. 1: Schéma de codage prédictif classique (a), schéma de codage MPEG-4 FGS (b)

Par ailleurs, MPEG-4 propose en plus la notion de codage orienté objet. Le codage orienté objet offre des fonctionnalités de manipulation sémantique si les objets définis s’y prêtent

(par exemple présentateur météo que l’on peut déplacer sur une carte de prévision en fond). Mais le codage objet peut également permettre de gagner en scalabilité (en allouant plus ou moins de débit sur les différents objets), voire de gagner en efficacité de compression [1, 5].

Nous proposons alors un schéma de codage basé texture afin d’obtenir un schéma scalable et intrinsèquement robuste. Ce schéma repose sur une approche de type analyse-synthèse. Dans un premier temps, une base de texture est créée pour les différents objets de la scène vidéo considérée. Par la suite, ces textures sont replaquées à l’aide de maillages actifs sur les différentes images afin de proposer une reconstruction pour une couche de bas niveau [6]. Enfin, des raffinements sont codés de façon progressive pour assurer un débit/qualité variable. La figure 2 présente une vue synoptique du schéma proposé.

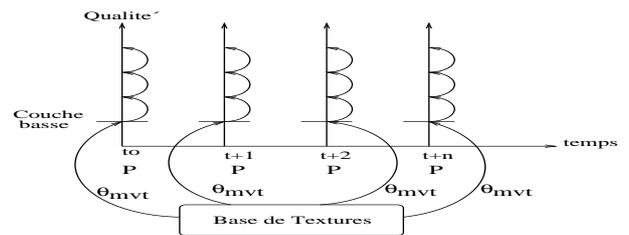


FIG. 2: Principe du schéma de codage basé-texture

## 2 Schéma de codage basé-texture

Le schéma de codage basé texture proposé se déroule en deux phases. Dans la première phase d’analyse, une base de textures est constituée via la définition d’une segmentation en objet et l’extraction d’une texture moyenne pour chaque objet. Dans la seconde phase de codage, les différentes textures sont codées, puis pour chaque image du GOP considéré, des erreurs résiduelles de prédiction sont établies et codées de façon progressive. La figure 3 montre le schéma proposé, où  $CObj_{ref}$

représente la carte d'objets de référence de l'image de référence  $I_{ref}$ .

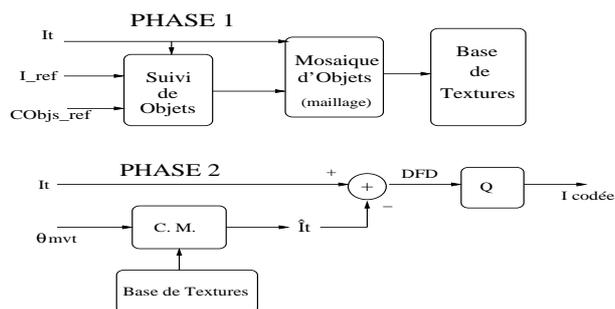


FIG. 3: Schéma de codage basé-texture

## 2.1 Phase 1 : extraction de la base de textures

Les approches de codage orienté-objet nécessitent une segmentation de la scène, segmentation idéalement qui se voudrait être aussi fidèle que possible aux objets physiques présents. Différents algorithmes de segmentation ont alors été présentés dans la littérature afin d'essayer d'obtenir un tel résultat ; la plupart de ces algorithmes reposant sur une segmentation initiale qui est suivie par la suite. Nous pouvons alors classer les algorithmes de suivi en deux grands groupes : les techniques de suivi par projection avant [3, 8] (prédiction de la segmentation par projection avant, puis raffinement) et les techniques de suivi par projection arrière [2, 4] (segmentation de l'image courante en régions, puis classification des régions par projection arrière sur la carte de segmentation objet de l'image précédente).

Dans le schéma proposé, la segmentation utilisée est une segmentation pseudo-automatique. Une segmentation initiale est définie sur la première image du GOP puis est suivie par un algorithme de type « projection arrière ». Cet algorithme exploite notamment des critères statistiques spatio-temporels permettant de modéliser l'évolution des objets et d'obtenir un suivi performant de multiples objets déformables [7]. Une fois cette segmentation obtenue, une mosaïque des différents objets est créée en utilisant la méthode proposée dans [6]. Pour chaque objet, les différents VOPs sont replaqués sur une image de référence (la première du GOP) afin de constituer une mosaïque de la texture de l'objet concerné. Ces mosaïques des différents objets donnent naissance à la « base de textures » voir figure 3.

## 2.2 Phase 2 : codeur

La couche basse de notre schéma de codage (figure 2) est reconstruite via l'utilisation d'une technique de prédiction par compensation en mouvement d'une texture de référence (généralement localisée sur la première image du groupe d'images considéré) vers une image quelconque. Du fait que le mouvement peut être complexe, nous utilisons alors une technique de compensation en mouvement par maillages actifs couplée à l'utilisation d'une segmentation en objet [6]. Les maillages actifs permettent d'avoir une bonne modélisation du mouvement assurant une bonne qualité de prédiction en couche basse. Par ailleurs, l'utilisation de la segmentation objet permet de limiter les dégénérescences du maillage et de pouvoir gérer les zones

d'occultations liées aux mouvement relatif entre les différents objets présents dans la scène.

Le codeur code alors les différentes mosaïques de la « base de textures » en mode intra, puis calcule l'erreur résiduelle de prédiction sur chaque image. Ces erreurs résiduelles sont par la suite codées progressivement. Dans cette étude, des techniques de codage par DCT et JPEG2000 sont utilisées. Ces diverses informations sont par la suite organisées en deux flux : un flux de base contenant la définition des différentes mosaïques/textures, et un flux complémentaire définissant pour chaque image, la carte de segmentation des objets, le mouvement des différents objets, et les informations définissant l'erreur résiduelle.

## 2.3 Décodeur

Au niveau du décodeur (figure 4), l'information de définition des différentes textures est tout d'abord reçue via un canal à qualité de service garantie. Ensuite, les informations propres à chaque image sont reçues et décodées. Le flux ainsi généré permet d'avoir une grande souplesse et robustesse. Le serveur, voire même les noeuds du réseaux, peuvent ainsi adapter dynamiquement le débit envoyé afin de limiter la saturation et les pertes engendrées. Différentes stratégies peuvent alors être définies afin de privilégier la fluidité de la vidéo, ou la qualité des images en diminuant la fréquence temporelle.

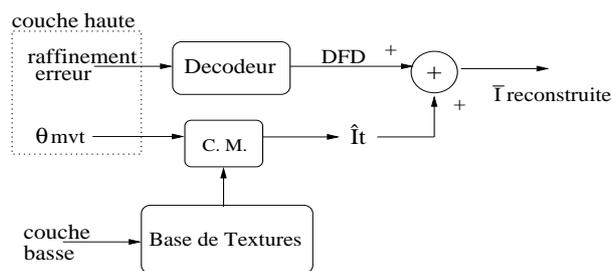


FIG. 4: Schéma du décodeur basé-texture

## 3 Résultats

Nous avons testé le schéma de codage proposé sur plusieurs séquences. Les séquences utilisées présentent divers objets avec des mouvements indépendants entre eux, ainsi qu'un mouvement de caméra.

Les figures suivantes donnent les résultats de compression à différents débits pour notre schéma pour différentes séquences. Nous présentons ci-dessous les résultats obtenus sur la séquence Foreman QCIF. La figure 5 présente les courbes de performance du codeur. La figure 6, montre les différents objets utilisés pour le codage, ainsi que les images reconstruites pour un pas de quantification (QP) de 15 (figure 7 a) et pour un QP de 30 (figure 7 b) lors d'un codage par DCT. La figure 8 illustre un exemple de scalabilité objet où le coût de codage est le même que la figure 7 b mais la qualité visuelle est meilleure.

Les résultats obtenus sur la séquence Coastguard QCIF sont présentés dans les figures 9, 10 ; les résultats obtenus sur la séquence Table Tennis sont présentées dans les figures 11, 12. Nous pouvons observer sur ces résultats que pour un coût de codage de 0.4 bits/pixel, le schéma proposé offre une bonne

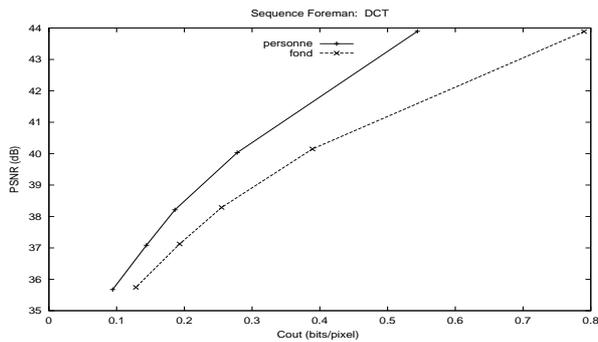


FIG. 5: Mesures du PSNR en fonction du coût de codage de la séquence *Foreman*

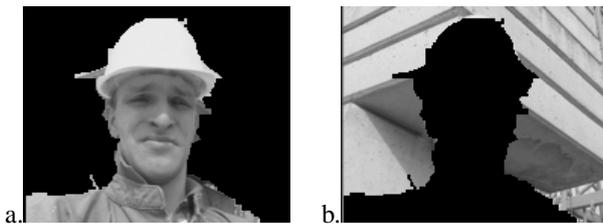


FIG. 6: Objets codé pour la séquence *Foreman* : personne (a), fond (b)

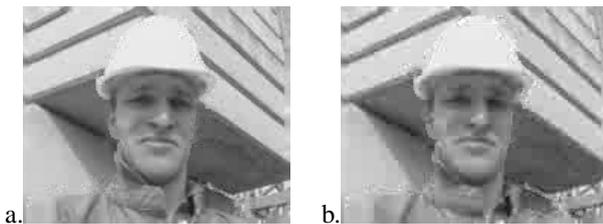


FIG. 7: Séquence *Foreman* reconstruite : QP=15, coût=0.4 bits/pixel (a) ; QP=30, coût=0.2 bits/pixel (b)

performance pour toutes les séquences traitées.

Les résultats suivants montrent les courbes de performance du codeur proposé en utilisant une technique de codage JPEG-2000. La figure 13 présente la courbe de performance de la séquence *Foreman*, la figure 14 celle de la séquence *Coastguard*, figure 15, la séquence *Table Tennis*.

Nous pouvons observer sur les courbes de performance de la DCT et JPEG2000 que pour un coût de codage de 0.4 bits/pixel, la mesure du PSNR de JPEG2000 des objets (de toutes les séquences traitées) est légèrement plus basse que celle-là obtenue par DCT. Toutefois, JPEG2000 offre une progressivité naturelle dans le codage de la couche de raffinement.

Par rapport aux résultats obtenus, nous pouvons dire que le schéma proposé offre une bonne performance pour toutes les séquences traitées.

## 4 Conclusion

Dans cet article, nous avons présenté un schéma de codage vidéo basé-texture scalable et robuste. Ce schéma repose sur une approche de type analyse-synthèse. La robustesse du sché-



FIG. 8: Séquence *Foreman* reconstruite au coût=0.2 bits/pixel : QP=20 pour la personne et QP=30 pour le fond

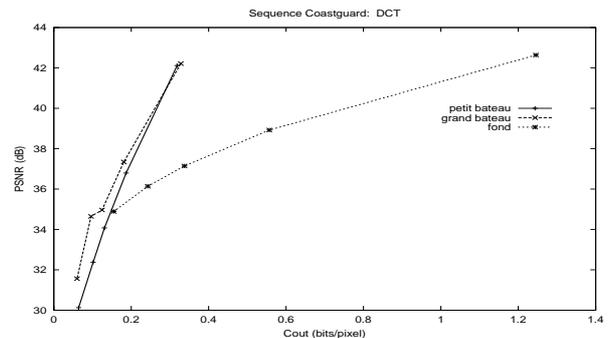


FIG. 9: Mesures du PSNR en fonction du coût de codage de la séquence *Coastguard*

ma proposé est accrue grâce à la possibilité de décoder indépendamment les différentes images de la vidéo.

Les résultats obtenus montrent que le schéma proposé offre un schéma de codage progressif avec un bon niveau de performance.

## Références

- [1] P. Salembier, F. Marqués et al. *Segmentation-based video coding system allowing the manipulation of objects* IEEE Trans. On Circuits and Systems for Video Technology, volume 7, No.1, february 1997.
- [2] C. Gu, et M. Lee. *Semantic video object tracking using region-based classification* Proc. of the IEEE International Conference on Image Processing, pp. 643-647, Chicago, october 1998.
- [3] F. Moscheni, F. Dufaux et M. Kunt. *Object tracking based on temporal and spatial information* IEEE Int. Conf. On Acoustics, Speech, and Signal Proc. Conf., volume 4, pp. 1914-1917, New York, USA, may 1996.
- [4] S. Pateux. *Tracking of video objects using a backward projection technique* Visual Conference on Image Processing, volume 4067, pp. 1107-1114, Perth, Australia, june 2000.
- [5] J. Morros et F. Marques. *Scalable segmentation-based coding of video sequences addressing content-based functionalities* ICIP, volume 2, pp. 1-4, october 1997.
- [6] S. Pateux, G. Marquant et D. Chavira-Martínez. *Object mosaicking via meshes and cracklines technique. Application to low bit-rate video coding* PCS 2001, Séoul, Corée, april 2001.

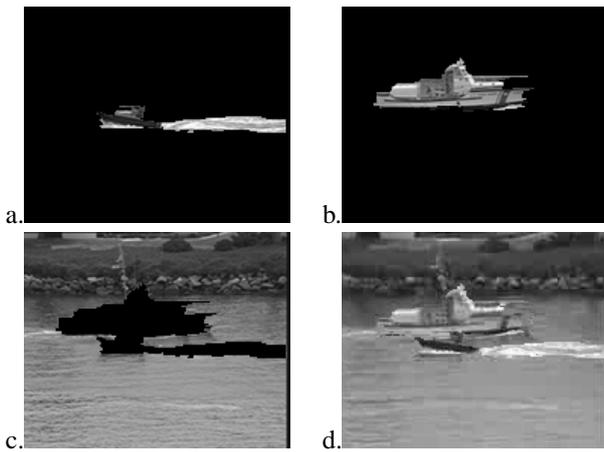


FIG. 10: Objets codé pour la séquence *Coastguard* : petit bateau (a), grand bateau (b), fond (c), séquence reconstruite , QP=20, coût=0.4 bits/pixel

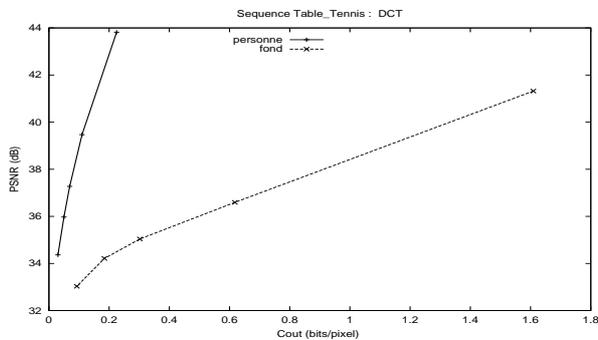


FIG. 11: Mesures du PSNR en fonction du coût de codage de la séquence *Table Tennis*

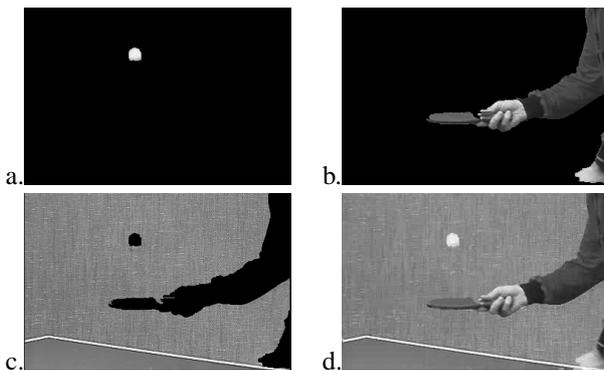


FIG. 12: Objets codé pour la séquence *Table Tennis* : balle (a), personne (b), fond (c), séquence reconstruite QP=15, coût=0.38 bits/pixel

- [7] D. Chavira-Martínez et S. Pateux. *Non-rigid object tracking using a likelihood spatio-temporal model* Visual Communications and Image Processing, Lugano, Switzerland, July 2003.
- [8] F. Marqués et C. Molina. *Object tracking for content-based functionalities* Visual Conference on Image Processing, volume 3024, No. 1, pp. 190-199, San Jose, February 1997.

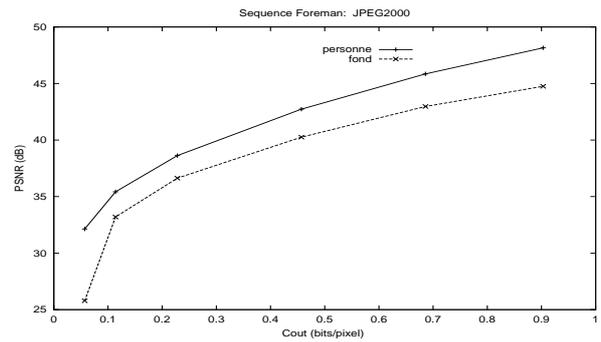


FIG. 13: Mesures du PSNR en fonction du coût de codage de la séquence *Foreman*

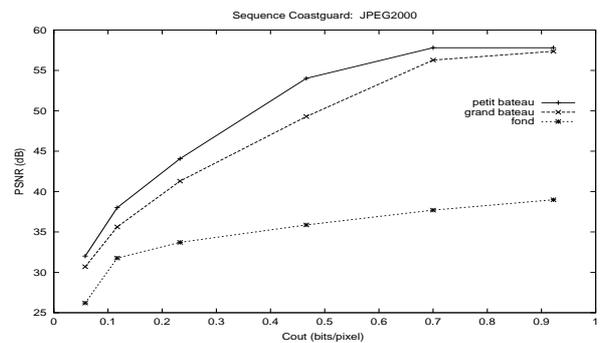


FIG. 14: Mesures du PSNR en fonction du coût de codage de la séquence *Coastguard*

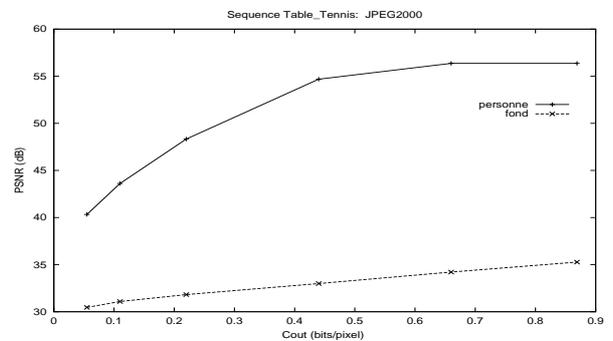


FIG. 15: Mesures du PSNR en fonction du coût de codage de la séquence *Table Tennis*