

# Segmentation d'images couleurs pour la compression de séquences vidéo par l'algorithme Mask Motion JPEG2000

D.Faura P.Garda

LISIF, Université Pierre et Marie Curie 4, place Jussieu 75005 Paris  
(David.Faura, Patrick.Garda)@lis.jussieu.fr

## Résumé

Nous avons proposé un algorithme original de compression de séquences d'image basé sur une détection de mouvement par champs de Markov et une compression des images suivant la norme JPEG2000 [1]. Dans ce contexte, nous avons mis au point un algorithme de segmentation d'image couleur et réalisé un démonstrateur de codec combinant cet algorithme de segmentation et un codec JPEG2000. Dans cet article nous décrivons les travaux effectués sur la segmentation d'images couleurs et nous présentons les résultats obtenus pour la compression de séquences vidéo.

## Mots clefs

Segmentation vidéo, Compression, JPEG2000.

## 1 Introduction

La norme qui va dominer le domaine de la compression vidéo est sans aucun doute le MPEG-4 qui vise un large panel d'application. Cependant MPEG-4 est toujours basé sur la compensation de mouvement et sur la DCT. Ce sont deux techniques efficaces mais de forte complexité.

La norme JPEG2000 est une alternative à MPEG-4, pour les solutions embarquées, de plus faible complexité et exempte d'artefact de type bloc.

De plus JPEG2000 est compatible avec MPEG-4 et surtout elle possède intrinsèquement tous les atouts pour la transmission sur les réseaux câblés et sans fils grâce à l'utilisation du concept de l'EBCOT[2].

Dans cette article nous étudions une méthode de compression basée sur JPEG2000.

Dans la deuxième partie de l'article nous proposons un algorithme de segmentation d'images couleurs basé sur une opération de masquage avec une carte de mouvement obtenue par un processus Markovien dont le principe est décrit dans la troisième partie. La dernière partie de cet article décrit comment nous avons intégré notre algorithme dans un codec JPEG2000. En présentant les résultats obtenus, nous verrons que l'algorithme de segmentation améliore sensiblement les performances du Motion JPEG2000.

## 2 Principe de la génération des différences masquées

Suite à de nombreux travaux sur la détection de mouvement à base de champs Markoviens, nous avons développé un algorithme original de segmentation du mouvement sur des images couleurs.

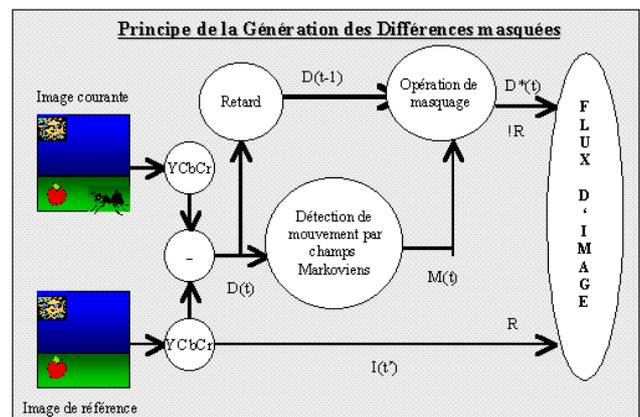


Figure 1- Principe de l'algorithme

Dans un premier temps, la séquence d'image est convertie dans le format de couleur YCbCr. Les images de différence sont utilisées pour construire trois cartes binaires de mouvement régularisées par un algorithme de champs Markoviens. Ces cartes binaires servent ensuite à masquer les images de différences. Cette technique nous permet de débiter les images et de reconstruire certaines zones du mouvement masquées par l'effet de recouvrement.

### 2.1 Transformation de couleurs

La transformation de couleur décorelle les plans RVB qui sont fortement redondants, le plus souvent on utilise un espace Luminance - Chrominance. On pourrait imaginer d'effectuer la détection de mouvement uniquement sur la composante de luminance, La majeure partie de l'information de mouvement est contenue dans ce plan, mais des artefacts gênant apparaissent comme des disparitions partielles d'objets en mouvement. En travaillant sur les 3 plans la reconstitution du mouvement est parfaite. Mais le temps de calcul est multiplié par 3.

Les transformations utilisées sont celles de la norme JPEG2000.

La transformation irréversible doit être utilisée selon la norme [3] uniquement lors d'une compression avec perte. Elle permet d'obtenir de meilleurs taux de compression que la transformation RCT mais elle est beaucoup plus lente, d'un facteur 3.

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.144 \\ -0.16875 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & -0.08131 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

## 2.2 Images de Différence

Le détecteur de mouvement utilisé consiste en une soustraction entre une image de Référence  $I_r(t')$  et une image courante  $I(t)$ .

$$t' = \left\lfloor \frac{t}{R} \right\rfloor$$

$$D(t) = I_r(t') - I(t)$$

Le résultat de la différence  $D(t)$  est bufférisé pour obtenir l'image de différence  $D(t-1)$  et sert d'entrée à l'algorithme Markovien, en prenant sa valeur absolue ( $o(t)$  est l'image des observations) :

$$o(t) = |I_r(t') - I(t)|$$

## 2.3 Masquage

L'image  $D(t)$  est très sensible aux variations d'intensité. Pour régulariser et éliminer le bruit impulsif, on utilise un masquage entre une carte binaire du mouvement obtenue par un processus Markovien  $M(t)$  et l'image de différence  $D(t-1)$ . Soit  $D^*(t)$  l'image de différence régularisées et  $\Theta$  l'opérateur de masquage.

$$D^*(t) = D(t-1) \Theta M(t)$$

## 2.4 Structure du flux

Le flux d'image généré est constitué d'images de référence et d'images de différence régularisées, dans l'algorithme décrit dans [4] la fréquence de rafraîchissement de l'image de référence  $R$  est de 6 images de différence pour une image de référence.

## 2.5 La reconstruction du flux

Pour reconstruire une image  $I(t)$ , on ajoute l'image de différence masquée courante  $D^*(t)$  à l'image de référence  $I_r(t')$ . Cette technique de reconstruction est certainement une des plus rapide. La seule opération effectuée pour reconstruire l'image est une addition pixel à pixel.

$$I(t) = D^*(t) + I_r(t')$$

## 2.6 Évaluation de l'algorithme

Pour évaluer les performances de l'algorithme, nous avons mesuré l'entropie, le PSNR et la vitesse de segmentation.

### 2.6.1 L'entropie

L'entropie est une mesure de l'information dans une image. Elle se mesure en Bit par pixel Sa formule est donnée par :

$$H(x) = - \sum_{x=0}^{x=n} P(x) \log P(x)$$

$P(x)$  étant la probabilité du symbole  $x$ ,  $x$  étant l'intensité d'un pixel.

Plus l'entropie est faible et plus les images sont simplifiées et nécessitent moins de bits pour leur codage.

On a mesuré l'entropie en trois points de l'algorithme, pour voir si les étapes successives produisent une diminution de l'entropie, au niveau de la transformée de couleur YCbCr, de la soustraction et après le masquage. Nous présentons les résultats obtenus pour la séquence Claire :

Entropie	C 1	C 2	C 3
RVB	4,01	4,2	4.4
YCbCr	4,28	2,81	3,38
Soustraction	1,97	1,38	1,73
Masquage	0,55	0,08	0,14

Table 1- Entropie du Masquage

On voit très bien d'après ces résultats l'intérêt de la transformée de couleur et du masquage, Ce sont des méthodes efficaces pour réduire l'information contenue dans une séquence d'images.

### 2.6.2 Le PSNR

Pour évaluer la qualité de la reconstruction, nous utilisons comme critère le PSNR, critère très utilisé en compression mais ne donnant qu'une mesure statistique des dégradations, de la manière suivante :

$$PSNR = 10 \log \frac{255^2}{MSE(Y)}$$

$Y$  étant la luminance définie précédemment.

Bien que visuellement sans défaut à la cadence vidéo, la mesure du PSNR, sans compression, sera la borne supérieure des mesures de qualité ultérieures pour les images reconstruites.

Pour la séquence Claire, le PSNR maximum est de 44 dB pour les images reconstruites.

### 2.6.3 Vitesse de segmentation

Les vitesses de segmentation des images ont été mesurées sur un Athlon de fréquence 1,5 GHZ avec 256 Mo de RAM, sous Windows 98. L'algorithme de segmentation a

été écrit en C++, sans optimisation. Nous avons effectué des mesures sur des séquences couleurs, de tailles différentes, sans sous échantillonnage des chrominances.

Nom	Taille	Vitesse en images par seconde
Akiyo	172x148	62,5
Claire	360x288	14,3
Chapeau	720x576	3,6

Table 2- Vitesse de Segmentation

### 3 L'algorithme de détection de mouvement Markovien

Les hypothèses de départ sont une illumination quasi constante et une caméra fixe. Dans ces conditions, l'information de mouvement est directement liée à la variation de la fonction intensité. On applique l'algorithme de détection de mouvement indépendamment sur les trois plans et de manière identique.

#### 3.1 La localité

Les images sont modélisées suivant la théorie mathématique des champs de Markov. L'intérêt de cette modélisation est la propriété de localité. Elle nous permet de dire que le choix de l'état d'un pixel ne dépend que des états des pixels appartenant à son voisinage.

#### 3.2 Le voisinage spatio-temporel

Le voisinage C choisi est composé d'une clique spatio-temporelle de 8 voisins spatiaux, et de deux voisins temporels (un pour le passé et un pour le futur). Ce voisinage sera utilisé pour le calcul de l'énergie locale.

#### 3.3 Le modèle d'énergie

Le calcul de l'énergie nécessite la connaissance des états des pixels voisins appartenant au voisinage spatio-temporel. Il est composé d'une énergie attachée au modèle et d'une énergie associée aux données. Ce modèle d'énergie a été introduit par le LIS-Grenoble [6] et est de la forme :

$$u(o_s, e_s) = u_m(e_s) + u_a(o_s, e_s)$$

L'énergie d'attache au modèle  $u_m$  exprime la cohérence et la compacité d'un objet en mouvement est de la forme :

$$u_m(e_s) = \sum_{c \in C} V_c(e_s, e_r)$$

Où c est une clique spatio-temporelle d'ordre 2 appartenant au voisinage C défini en 3.2,  $e_r$  correspond à une réalisation du voisinage C et  $e_s$  du site courant.

Les potentiels  $V_c$  prennent les valeurs suivantes :

$$V_c(e_s, e_r) = \begin{cases} -\beta_r & \text{si } e_s = e_r \\ +\beta_r & \text{si } e_s \neq e_r \end{cases}$$

$\beta_r$  dépend du type de voisinage, il vaudra  $\beta_s$  si  $e_r$  appartient au voisinage spatial,  $\beta_p$  pour le voisinage temporel du passé et  $\beta_f$  pour le voisinage temporel du futur. Typiquement on prend les valeurs suivantes :

$$\beta_p = 10, \beta_s = 20, \beta_f = 30$$

L'énergie d'attache aux données empêche le modèle de trop dériver par rapport à l'initialisation, elle est définie de la manière suivante :

$$u_a(o_s, e_s) = \frac{1}{2\sigma^2} [o_s(s, t) - \Psi(e_s, t)]^2$$

$$\text{où } \psi(e(s, t)) = \begin{cases} 0 & \text{si } e(s) = 0 \\ \alpha & \text{si } e(s) = 1 \end{cases}$$

La variance  $\sigma^2$  est calculée pour chaque image sur les observations  $o_s(t)$ . On prend en général  $\alpha = 20$ , mais des techniques d'actualisations du paramètre  $\alpha$  peuvent être utilisées [6].

#### 3.4 Minimisation de l'énergie

Pour minimiser l'énergie un algorithme de relaxation déterministe l'ICM (Iterated Conditional Mode) est utilisé. L'algorithme converge en quelques itérations vers le premier minimum local, nous l'utilisons avec deux itérations, voire une itération pour le temps réel, suivant la taille de l'image. La technique de balayage de l'image utilisée est ligne par ligne site récursive. A la suite de ce traitement nous obtenons une image d'énergie minimale représentant la carte binaire du mouvement.

## 4 Le Codec Mask Motion JPEG2000

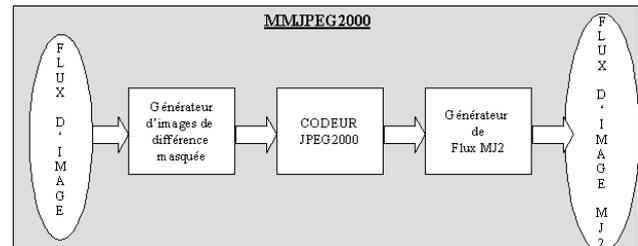


Figure 2 – MMJPEG2000

#### 4.1 Description du système.

Le système de compression est composé de trois modules distincts, comme sur la figure 2.

Le premier de ces modules est le générateur des images de différence masquées vu précédemment. Pour le coupler avec le codeur, une transformation inverse couleurs (ICT) est ajoutée pour que le codeur ait en entrée des images RVB. Le codeur utilisé est Kadaku [7], il suit la norme JPEG2000, pour nos mesures nous utilisons uniquement les paramètres par défaut. Le dernier module est un module permettant de concaténer le flux compressé des images de la séquence et d'ajouter les entêtes de Motion

JPEG2000 [2], dont le flux est compatible avec le flux MPEG-4.

## 4.2 Comparaison entre MMJPEG2000 et d'autres codecs.

L'intérêt des images de différences est de pouvoir les compresser plus fortement que les images de Référence. Nous avons créé des séquences avec des taux de compression pour l'image de référence variant de 16 à 64 et des taux de compression pour les images de différence variant de 32 à 512. On obtient des séquences de taux de compression donné, on calcule le PSNR moyen de la séquence puis on compare avec la même séquence compressée au même taux de compression avec Motion-JPEG2000. Sur la figure 3, on constate que les images compressées avec MMJPEG2000 sont de meilleure qualité du point de vue statistique et visuelle sauf pour les faibles taux de compression où ils ont des PSNR quasi identiques. Pour avoir une qualité d'image acceptable il faut que l'image de référence ait un PSNR au minimum proche de 35. La qualité de l'image de Référence permet d'augmenter globalement la qualité de l'image reconstruite, mais localement, en particulier dans les zones affectées par le mouvement, des défauts locaux, zones de flou, irisations.

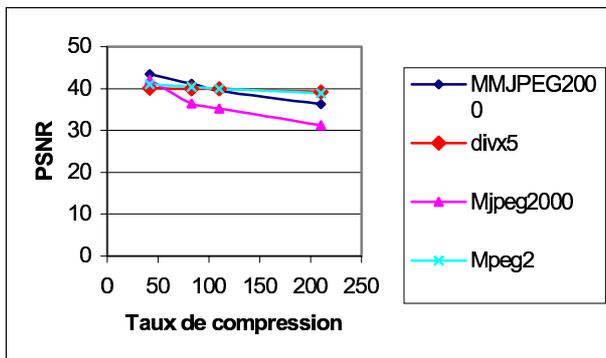


Figure 3- MJPEG2000 vs MMJPEG2000

plus ou moins gênantes sont introduits selon le taux de compression des images de différence. Ces défauts sont majoritairement dus au défaut de "flickering" introduit par JPEG2000 pour les fort taux de compression[9].

Sur la figure 3 nous faisons aussi, la comparaison de MMJPEG2000 avec deux autres codecs, un codec de type MPEG-4 : DIVX 5 et un codec MPEG-2.

Les courbes obtenues montrent la très bonne tenue de MMJPEG2000 dans la plage de taux de compression [50,210] face à ces deux codecs dominant le marché. De plus, il faut rappeler que nous n'effectuons aucun sous échantillonnage des chrominances.

## 5 Conclusion

Les résultats obtenus sur le module de segmentation couleur sont très prometteurs du point de vue de la qualité et de la vitesse de segmentation de l'image. En effet, cet algorithme présente une très faible complexité pour l'encodage et aussi pour la reconstruction du signal vidéo : une simple addition pixel à pixel suffit. La qualité visuelle de la reconstruction est très bonne, si les mouvements de la scène ne sont pas de trop grande vitesse. L'algorithme de segmentation est indépendant du codec ce qui fait qu'il peut être intégré dans n'importe quel codec et nous envisageons de l'intégrer dans un environnement MPEG-4 et comme post traitement dans une rétine CMOS.

Les résultats de compression obtenus par l'algorithme de segmentation et JPEG2000 sont très encourageants, car grâce à la carte binaire du mouvement, il serait possible de déterminer des zones d'intérêts qui seront moins compressées que les autres zones diminuant ainsi les défauts locaux dus aux mouvements. Cette fonctionnalité de régions d'intérêt est une des nombreuses fonctionnalités de JPEG2000 et la robustesse vis à vis des erreurs de transmissions font que le MMJPEG2000 peut être une alternative très intéressante pour des applications embarquées et pour la transmission vidéo sur internet.

## Référence

- [1] F.Lohier, P.Garda, L.Lacassagne, Procédé et dispositif de traitement de séquences d'images avec masquage *Brevet UPMC, Français en cours d'extension internationale, Demande FR 62060 L du 3 février 2000*
- [2] D.Taubman "High Performance Scalable Image Compression with EBCOT", *in Proc IEEE int. conférence Image Processing*, vol III, pp. 344-348,(Kobe,Japan) October 1999.
- [3] JPEG2000 part-1, ISO/IEC 15444-1:2000.
- [4] F.Lohier, "Méthodologie de programmation et évaluation des processeurs de traitement du signal parallèles pour le traitement d'images en temps réel", thèse de doctorat, UPMC, LISIF, 2000.
- [5] A.Caplier, Thèse "Modèle Markovien de détection de mouvement dans les séquences d'images". thèse de doctorat, INPG-TIRF, 1995.
- [6] L.Lacassagne ; Thèse "Détection de mouvement et suivi d'objet temps réel", UPMC, LISIF, 2000.
- [7] David S.Taubman, Michael W.Marcellin JPEG2000, Image Compression, Fundamentals, Standards and Practice. Boston : Kluwer Academic Publishers., 2002.
- [8] JPEG2000 part-3, ISO/IEC 15444-3:2000.
- [9] Tetsuro Kuge, "Wavelet Picture coding and its several problem of the application to interlace HDTV and the ultra high definition images"  
<http://www.nhk.or.jp/str1/publiclab/labnote/lab481.html>.