

---

# vidéo sur des réseaux de capteurs sans fils

**David Faura<sup>2</sup>, Khalil Hachicha<sup>1</sup>, Olivier Romain<sup>1</sup>,  
Patrick Garda<sup>1</sup>**

1. *Laboratoire LIP6 (CNRS)*

*Département SOC-Groupe SYEL*

*BC 167, 4 place Jussieu*

*F-75252 Paris, France*

*{khalil.hachicha,olivier.romain,patrick.garda}@upmc.fr*

2. *Thales Avionics, ACS/DTEA*

*18 avenue Maréchal Juin*

*F-92366 Meudon-la-Forêt Cedex, France*

*david.faura@fr.thalesgroup.com*

---

*RÉSUMÉ. Dans cet article, nous présentons une technique de construction de cartes binaires de mouvements d'images couleurs qui améliore significativement la qualité statistique de Motion-JPEG2000 de 10 dB. Cet algorithme est basé sur le masquage d'une image de différence obtenue par un processus markovien. Les premiers résultats montrent que l'on peut transmettre des séquences vidéo sur un réseau de transmission sans fil bas débit de type WPAN – IEEE802.15.4 avec une bonne qualité visuelle.*

*ABSTRACT. We developed an algorithm based on Markov model and MJPEG2000 video codec. The proposed compression algorithm uses a simpler model to detect the motion and allows a direct access to the motion binary map for high level processing (Intrusion detection, Tracking ...). Furthermore, this algorithm is suitable for an implementation in a wireless sensor network dedicated to video monitoring.*

*MOTS-CLÉS : chaînes de Markov, réseaux de capteurs sans fils, filtrage spatio-temporel, codage vidéo.*

*KEYWORDS: MMJPEG2000, Markov model, video coding, wireless sensor network, Maximum A Posteriori criterion.*

---

DOI:10.3166/TS.27.541-562 © 2010 Lavoisier, Paris

## Extended Abstract

We developed an original video compression algorithm named MMJPEG2000 which increases the quality of the third part of the JPEG2000 standard. To enhance the compression ratio, we divided the images flow into images of reference and images of difference. Next, we apply a masking technique to the images of difference. The mask is a motion binary map obtained from a motion detection algorithm. The analysis of motion is approached mathematically through the extraction of motion information from a sequence of images by means of specific data processing algorithms. Many algorithms of detection, estimation and interpretation of motion were developed with various parameters models. We reworked and developed the Markov model using the potential functions foreseen by the motion detection combining the spatial and the temporal information. The purpose of this technique is to localize moving and static areas in a dynamic scene. Then we attribute to each site  $s(x,y)$  one of the two labels: 1 if  $s$  belongs to a moving area and 0 if  $s$  belongs to the static background. The most probable configuration is found by using the Maximum A Posteriori criterion (MAP). The algorithm Principle is composed of two distinct steps: the first is a preprocessing phase through which we compute the absolute value of the difference matrix between the current frame and the reference frame. Next, we binarize the matrix result by setting a threshold and we determine the variance. The second step is the implementation of the Iterated Conditional Mode algorithm (ICM), which updates the binary state of the pixels of difference (moving or not) and is made site by site in a way that every change in state is taken immediately into account in the relaxation of the neighboring site. For each image site, we calculate the local energy relative to both the immobile and the mobile state. After, we allocate the state which minimizes the energy to the site being treated. Leaving the Iterated Conditional Mode algorithm, we achieve an image of minimal energy which represents the binary motion map. The product of this binary motion map and the differences between the images generate a filtered image of difference.

In order to compute the energy, one must know the state of the pixels belonging to a neighborhood defined by eight spatial neighbors and two temporal neighbors. The energy expression is the sum of two terms: The energy associated with the data ( $Ud$ ) and the energy associated with the model  $Um$ . The energy associated with the model consists of the spatial energy that is supposed to model the consistency and the compactness of a moving object and the temporal energy which represents the variation of the intensity function when the frame changes. The setting of the parameters value is based on empirical observations: good agreement between contours of masks and actual moving objects, contextual homogeneity of detected masks and insensitivity to acquisition noise. To evaluate the algorithm performances, we computed the PSNR of the reconstructed frame and also the execution speed of the algorithm for Qcif 4:4:4 sequences. We compressed the test sequences with MJPEG2000 and MMJPEG2000 to show the improvement. Due to the fact that the algorithm will be implemented on a wireless sensor network, we chose different test sequences extracted from video monitoring and TV news. The sequences were compressed with Kakadu JPEG2000 compressor. We used the 9/7 wavelet kernel, the `no_weight` option that optimizes the images PSNR, and 8x8 block sizes for the arithmetic compression. The motion detection settings are a single ICM iteration and an entropic thresholding. All the tests were made with a key frame compression ratio

of 16. We varied the difference frame compression ratio from 32 to 96. The results show that the masking technique reduces the image noise and creates an image containing relevant information before sending it to the JPEG2000 coder. This algorithm has a lower complexity than the block matching method used in the MPEG compression standard. The improvement varies between 4 and 10 dB. The PSNR is higher than 39 dB for the sequences with low motion. For the strong compression ratio ( $> 80$ ), the results could be discussed because of the presence of local defaults. These defaults are more visible in sequences with strong motion and they appear for a lower compression ratio ( $> 32$ ). For slow motion sequence, we can decrease the data rates down to 200 kbits/s (24 i/s, 4:4:4) with a good visual quality (PSNR 40 dB, Claire sequence). On the other hand, if the motion in the sequence is significant, then the data rate of 235 kbit/s is roughly the lower limit in order to prevent the defaults. We note that the MMJPEG2000 compression algorithm performs real time processing for Qcif sequences without any hardware optimization. It takes 21,34 ms to compress a Qcif frame.

In conclusion, the algorithm we proposed is able to auto-adapt to the sequence context modifications and if the motion in the sequence is not too significant, MMJPEG2000 appreciably improves the visual quality of the MJPEG2000 standard while reducing considerably the data rate. Despite losing the MJPEG2000 propriety to compress every frame individually, the proposed compression algorithm uses a simpler model to detect the motion and allows a direct access to the motion binary map for high level processing (Intrusion detection, Tracking ...). Furthermore, this algorithm is suitable for an implementation in a limited resources wireless sensor network dedicated to video monitoring.

## 1. Introduction

Les réseaux de capteurs vidéo sans fil sont constitués de nombreux systèmes électroniques autonomes possédant de faibles capacités de traitement et de transmission appelées « balises » (Romain *et al.*, 2004). Ces balises acquièrent des données de leur environnement et envoient le flux vidéo à travers un réseau de transmission sans fil bas débit de type WPAN comme Zigbee. La transmission d'un flux vidéo couleur au format QCIF n'est pas compatible avec une transmission temps réel sur un réseau bas débit de type WPAN possédant un débit maximum théorique de 250 kb/s. Un algorithme de compression est nécessaire pour réduire la quantité de données à transmettre avec un taux de compression supérieure à 84. Compte tenu des faibles capacités de traitement, de mémorisation des balises vidéo et des conditions d'utilisation, l'algorithme de compression devra être de faible complexité et atteindre de forts taux de compression. Il est donc exclu d'utiliser des méthodes de compression de type MPEGx, H26X (Akyildiz *et al.*, 2007) (Saponara *et al.*, 2003), très gourmandes en consommation d'énergie et en temps de calcul. Une méthode de compression image par image basée sur JPEG2000 semble plus adéquate pour une solution embarquée. En effet, JPEG2000 est de plus faible complexité que les algorithmes de type MPEG et il est exempt d'artefacts de types blocs. Egalement, JPEG2000 possède intrinsèquement tous les atouts pour une transmission sur les réseaux câblés et sans fil grâce à l'utilisation du concept de l'EBCOT (Embedded Block Coding with Optimal Truncated point) (Taubman, 1999). Enfin, la norme définit Motion JPEG2000 (MJPEG2000) (Group, 2000) pour la compression de

séquences vidéo. La plus simple opération en compression vidéo pour diminuer les débits correspond à coder uniquement les différences entre une image à un instant donné et une image de référence. Cette technique a l'inconvénient d'être sensible au bruit et de diminuer l'efficacité de la compression. Tout en gardant cette simplicité de codage, nous avons proposé un algorithme de compression basé sur le masquage des images de différence appelé Mask Motion JPEG2000 (Faura *et al.*, 2006 ; Faura *et al.*, 2004 ; Faura *et al.*, 2003). Les masques sont obtenus par une détection des zones de mouvements basée sur les champs de Markov avec une prise de vue basée sur une caméra fixe. Ils permettent de diminuer fortement le bruit, tout en simplifiant les images envoyées au codeur JPEG2000. Nous commençons cet article par une description de l'état de l'art sur l'amélioration de la qualité d'image d'un flux MJPEG2000. Ensuite, nous exposons les principes de l'algorithme de compression MMJPEG2000. Nous présentons une évaluation des performances de l'algorithme et nous comparons ces résultats à ceux obtenus par l'algorithme MJPEG2000 afin de quantifier les améliorations obtenues en tenant compte de différents critères de qualités. Nous introduisons également des méthodes de seuillage et changement d'images de références automatiques, basées sur une analyse de la distribution des intensités des pixels, permettant d'avoir des résultats nettement supérieures à ceux obtenus dans nos précédents travaux.

## 2. États de l'art sur l'amélioration de la qualité d'image d'un flux MJPEG2000

La norme JPEG2000 est basée sur deux grands principes : la transformée en ondelettes et l'EBCOT (Embedded Block Coding with Optimized Truncation). Elle possède de meilleures performances du point de vue rapport qualité/débit que la norme JPEG et possède plus de fonctionnalités que les autres normes de codage d'images fixes (Santa-Cruz *et al.*, 2000) (Skodras *et al.*, 2001). Son intérêt est de pouvoir utiliser les modes de compression sans et avec perte tout en utilisant la même chaîne algorithmique (Adams, 2001). L'algorithme de compression considère que chaque composante de l'image est divisée en tuiles rectangulaires traitées de manière indépendante. La première étape est la soustraction d'un offset à chaque coefficient de la tuile, pour que sa dynamique soit centrée sur 0 (DC shift). Après cette modification, une transformée de couleurs sans perte RCT (Reversible Color Transform) ou ICT (Irreversible Color Transform) est effectuée. Une étape de quantification à pas unitaire ou scalaire (Deadzone) est réalisée suivant le mode de compression choisi. La valeur de cette quantification peut être modifiée grâce à l'utilisation d'une région d'intérêt (ROI, Region Of Interest). C'est une région de l'image qui est codée avec plus grande précision au détriment des autres zones de l'image qui sont alors compressées à un taux inférieur et donc dégradées (Subedar *et al.*, 2004 ; Wang *et al.*, 2002). Les tuiles sont ensuite décomposées en blocs. Le codage des blocs s'effectue plan de bits par plan de bits par un codeur arithmétique adaptatif (MQ-codeur). À la fin du codage, si le débit ciblé n'est pas atteint, un algorithme de post-compression permet de tronquer le flux compressé. Pour finir, ce flux est encapsulé et organisé dans un des cinq modes de transmission de données prévus par la norme. Le flux vidéo MJPEG2000 est une simple juxtaposition d'images compressées par l'algorithme JPEG2000. Malgré ses performances au niveau qualité – débit, plusieurs auteurs ont cherché à modifier et améliorer son flux d'images. Deux approches sont possibles : la première est décrite dans (Totozafiny *et*

*al.*, 2005) pour des applications de vidéo surveillance. Le flux d'images est décomposé en images de référence et en images contenant les objets appartenant aux régions d'intérêts. Les images de référence sont mises à jour (suivant un découpage par zone) par une méthode adaptative basée sur des gaussiennes et dans (Totozafiny, 2007) par l'application de logique floue. La ROI dans (Luthon *et al.*, 2004a) est obtenue dynamiquement grâce à une détection des zones de mouvement et est codée par la méthode « maxshift » de JPEG2000. Il faut noter que, dans (Luthon *et al.*, 2004a), la transformée de couleurs de la norme a été modifiée par une transformée de couleurs logarithmique dénommée LUX qui améliore le rendu des couleurs des images. Une approche similaire est présentée dans (Luthon *et al.*, 1993) où l'obtention de la ROI et de l'image de fond se fait grâce à un algorithme de segmentation statistique basé sur des gaussiennes. Nous présentons dans cet article une autre approche. En effet, le flux d'image est séparé en images de références et en images de différences obtenues par masquage avec une carte binaire du mouvement. Cette méthode dérive de Frantz Lohier (Lohier, 2000) où il a été montré que la technique de masquage couplée avec un codeur MJPEG améliore fortement les performances du codeur. Dans (Lohier *et al.*, 2000), la technique a été utilisée avec un codeur basé sur les ondelettes (ADevice, 2004), montrant ainsi la possibilité d'étendre la technique de masquage à un codeur de type JPEG2000. Le tableau 1 résume les résultats de la communauté en la matière.

Les méthodes décrites dans cette section sont essentiellement basées sur la détermination de la ROI, l'application des coefficients de pondération psycho-visuelle au niveau des ondelettes et l'amélioration des points de troncature du flux compressé. Malgré les améliorations qu'elles apportent, elles restent faibles par rapport à l'utilisation des méthodes qui modifient le flux d'images et en particulier la méthode différentielle (tableau 2). Nous proposons ainsi dans cet article une nouvelle approche basée sur la méthode différentielle couplée à des algorithmes markoviens permettant d'améliorer significativement les performances du MJPEG2000.

Tableau 1. Amélioration au niveau codeur et post-filtrage

Méthodes	Améliorations	Auteurs
Filtrage des artefacts	PSNR + 0.2 à 0.5 dB	(Bourdon <i>et al.</i> , 2003)
Pondération psycho visuelle	PSNR + 0.2 à 0.5 dB	(Varma <i>et al.</i> , 2004)
détection des zones de mouvement	Débit – 10 %, qualité visuelle identique	(Kim <i>et al.</i> , 2006)
Post-compression	Compromis vitesse, mémoire, PSNR	(Yeung, 2005)

Tableau 2. Amélioration par modification du flux d'image

Méthodes	Améliorations	Auteurs
ROI JPEG2000	1 CIF/s GSM. CIF à 660 kbps, PSNR + 4 dB	(Luthon <i>et al.</i> , 2004a)
Différentielle masquage	Diminution du débit de JPEG de 15 % à 35 %	(Lohier, 2000)

### 3. Principe de l'algorithme MMJPEG2000

Pour diminuer le débit, le flux vidéo est décomposé en images de référence et de différence : la différence entre l'image de référence et l'image courante est uniquement codée. L'idée directrice de l'algorithme est d'envoyer au compresseur une image simplifiée en considérant les petites variations de luminance et de chrominance entre deux images consécutives négligeables. Les images simplifiées sont obtenues à partir d'images de différence masquées par une carte binaire du mouvement. La carte binaire du mouvement est obtenue par l'utilisation d'un algorithme de détection des zones de mouvement suivi par un algorithme de régularisation de l'image binaire. Ces opérations permettent d'éliminer le bruit impulsif et les faibles zones de variation du mouvement. Grâce à cette technique nous obtenons des images de différence simplifiées où l'information utile est contenue uniquement dans la carte binaire du mouvement. En effet, seuls les pixels qui sont affectés par le mouvement sont envoyés au codeur alors que les autres pixels sont forcés à une valeur nulle (figure 1).

#### 3.1. L'algorithme de détection des zones de mouvement markovien

##### 3.1.1. Notations

- $E$  : carte binaire de mouvement.
- $\hat{E}$  : carte binaire de l'image d'origine.
- $s$  : site de coordonnée  $(x,y)$ .
- $O_t = \{O_t(s), s \in E\}$  la valeur absolue de l'image de différence.
- $I_t = \{I_t(s), s \in E\}$  : image courante.
- $e(s)$  : la valeur d'un site après binarisation.
- $U_m$  et  $U_a$  : énergies associées aux modèles et données.

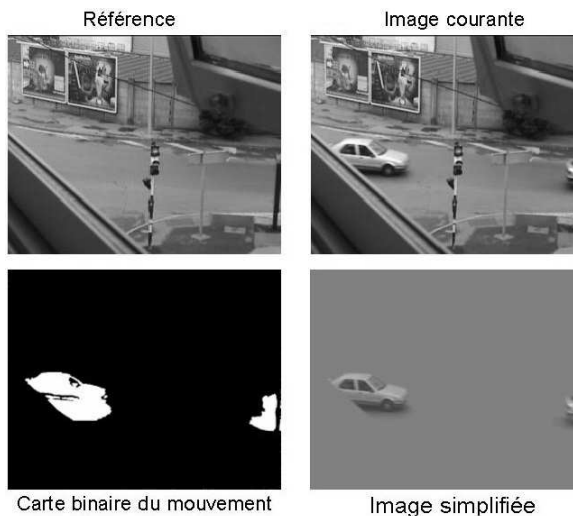


Figure 1. Exemple de traitement

3.1.2. Schéma bloc de la détection des zones de mouvement markovienne

Nous avons développé un algorithme de détection des zones de mouvement basé sur les champs de Markov de (Luthon *et al.*, 1993) (Caplier, 1995). La figure 2 résume les différentes étapes nécessaires pour obtenir la carte binaire du mouvement utilisée lors de l'opération de masquage.

Cette carte est obtenue à partir de la minimisation de l'énergie de la carte binaire  $\hat{E}(t)$  bruitée. Pour cela, le modèle de détection des zones de mouvement utilise une image de différences absolues  $O_t$  et deux cartes binaires du mouvement  $E(t-1)$  et  $\hat{E}(t)$ . On applique cet algorithme indépendamment sur les trois plans et de manière identique. Les hypothèses de départ sont une illumination avec une faible variation lente et progressive et une caméra fixe. Dans ces conditions, l'information de mouvement est directement liée à la variation de la fonction intensité.

3.1.3. Création de la carte binaire de mouvement

La première étape de l'algorithme est l'obtention d'une image de différence qui sera le résultat d'une soustraction entre l'image courante  $I_t(s)$  et une image de référence,  $ref(s)$ , et ceci dans un souci de reconstruction parfaite de la scène. En appliquant une opération de valeur absolue sur cette image de différence, on obtient l'image des observations  $O_t(s)$ . En chaque site  $s = (x,y)$ , la valeur de l'observation  $O_t(s)$  est :

$$O_t(s) = |I_t(s) - ref(s)| \tag{1}$$

En chaque site  $s = (x,y)$ , la valeur  $sig_t(s)$  du signe de l'observation  $O_t(s)$  est :

$$sig_t(s) = \begin{cases} 1 & \text{si } I_t(s) - ref(s) < 0 \\ 0 & \text{sinon} \end{cases} \tag{2}$$

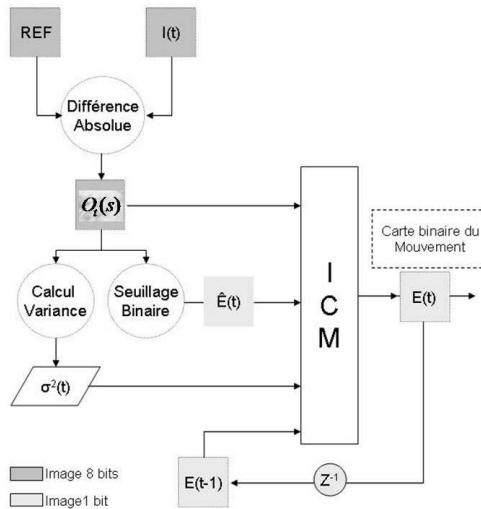


Figure 2. Détection des zones de mouvement markovienne

La carte des changements temporels est obtenue par seuillage de l'image des observations. La carte binaire du mouvement  $\hat{E}(t)$  est alors obtenue. Le seuil est déterminé par une méthode adaptative basée sur une estimation de l'écart type du bruit de fond (voir section 3.2).

$$\begin{cases} O_t(s) > \text{seuil} & \text{alors } e(s) = 1 \\ \text{sinon} & e(s) = 0 \end{cases} \quad [3]$$

#### 3.1.4. Le modèle et minimisation d'énergie

Le calcul de l'énergie nécessite la connaissance des états des pixels voisins appartenant au voisinage spatio-temporel composé des 8 plus proches voisins spatiaux de la carte binaire du présent et du plus proche voisin temporel appartenant à la carte binaire régularisée précédente. Le modèle d'énergie que nous utilisons est composé d'un terme représentant l'énergie potentielle et d'un terme représentant l'énergie attaché aux données. Il est de la forme :

$$U(o(s), e(s)) = U_m(e(s)) + U_a(o(s), e(s)) \quad [4]$$

– L'énergie du modèle  $U_m$  exprime la cohérence et la compacité d'un objet en mouvement. Nous avons modifié la forme de l'énergie de modèle pour prendre en considération la valeur du pixel au site courant. Cette modification favorise la cohérence spatiale (Riquebourg, 1997). L'énergie de modèle sera de la forme :

$$U_m(e(s)) = V_s(e(s)) + \sum_{c(s,r)} V(e(s), e(r)) \quad [5]$$

Les fonctions potentielles  $V$  s'expriment en fonction d'un paramètre  $\beta$  qui caractérise le poids de chaque site. Dans le modèle de (Caplier, 1995), les coefficients  $\beta$  associés aux cliques temporelles sont différents des  $\beta$  spatiaux pour favoriser l'élimination de la traînée de l'objet. Comme nous devons reconstruire une scène, la conservation de l'ancienne position du mobile est nécessaire, nous prenons donc tous les beta avec un poids identique. La valeur de ce paramètre sera prise égale à 20 (Caplier, 1995). Les potentiels  $V_c$  prennent les valeurs suivantes :

$$V_c(e_s, e_r) = \begin{cases} -\beta_r & \text{si } e_s = e_r \\ +\beta_r & \text{si } e_s \neq e_r \end{cases} \quad [6]$$

– L'énergie attachée aux données  $U_a$  empêche le modèle de trop dériver par rapport à l'initialisation. Elle est définie de la manière suivante :

$$U_a(o_s, e_s) = \frac{1}{2\sigma^2} [o_s(s, t) - \psi(e_s, t)]^2 \quad [7]$$

ou

$$\psi(e(s, t)) = \begin{cases} 0 & \text{si } e(s) = 0 \\ \alpha & \text{si } e(s) = 1 \end{cases} \quad [8]$$

La variance  $\sigma^2$  est calculée pour chaque image sur les observations  $o_s(t)$ . Une valeur empirique de  $\alpha = 20$  est généralement utilisée. Des techniques d'actualisation



du paramètre peuvent être employées. Pour minimiser l'énergie, un algorithme de relaxation déterministe est utilisé MPC (minimisation des probabilités conditionnelles : équivalent à un ICM avec 1 itération). La technique de balayage de l'image utilisée est ligne par ligne site récursive. À la suite de ce traitement une image d'énergie minimale est obtenue représentant la carte binaire du mouvement  $E(t)$ .

### 3.1.5. Influence du nombre d'itérations

Le respect de contraintes temps réel des systèmes numériques n'est pas compatible avec l'attente de la convergence de l'algorithme de régularisation. Un petit nombre d'itérations est utilisé pour résoudre ce problème. Son influence sur la qualité des images masquées a été évaluée en mesurant le PSNR des images reconstruites et l'entropie des images de différence. La figure 3 présente l'impact de l'utilisation de 1, 2, 3 et 4 itérations sur le PSNR. Nous remarquons que l'utilisation d'une seule itération d'ICM (Iterative Conditional Mode) pour la régularisation de la carte binaire du mouvement dégradera faiblement les performances du masquage. En conséquence nous avons opté dans la suite pour l'utilisation d'une seule itération pour l'algorithme de masquage.

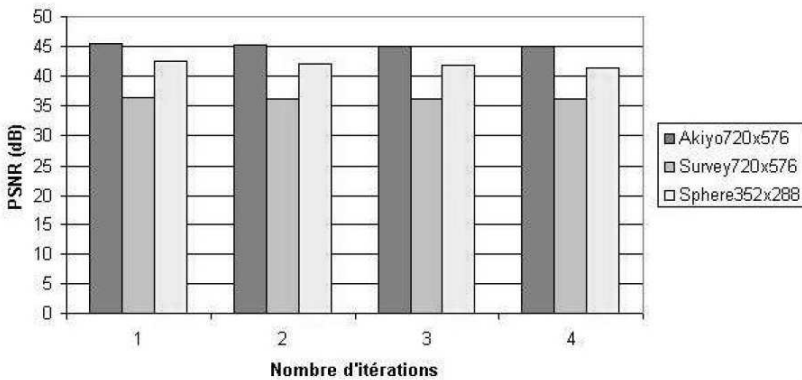


Figure 3. Influence du nombre d'itération sur le PSNR

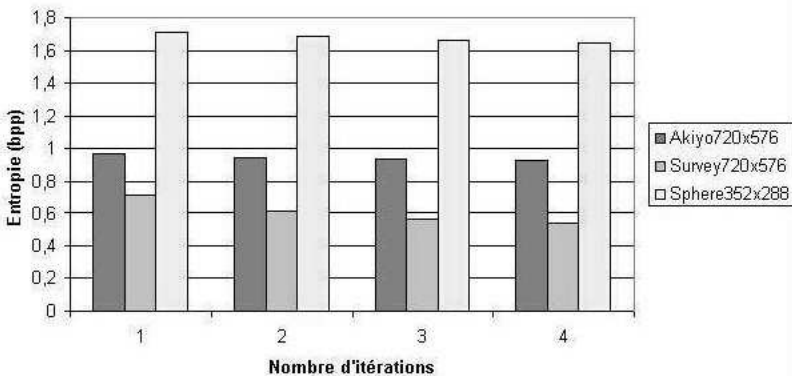


Figure 4. Influence du nombre d'itérations sur l'entropie  $L$  l'algorithme

L'algorithme de masquage fait chuter l'entropie des séquences vidéo si le mouvement de la scène n'est pas trop important : le nombre d'itérations contribue au filtrage de l'image. Ce phénomène peut être observé sur la figure 4. Nous constatons une diminution de l'entropie avec le nombre croissant d'itérations tout en étant du même ordre de grandeur. On remarque aussi que l'entropie de la séquence Survey décroît plus fortement que pour la séquence Akiyo, ce qui veut dire que cette séquence est plus bruitée.

### 3.2. Seuillage

La qualité de la carte binaire du mouvement final dépend du résultat du seuillage de l'image de différence. La binarisation permet de séparer les objets en mouvement du bruit de fond. Le bruit de fond correspond aux sites où il n'y a pas de mouvement et où les variations d'intensité sont uniquement dues au bruit du système d'acquisition. Pour la suite des développements, nous considérons que les distributions des objets en mouvement et du bruit de fond sont gaussiennes. L'analyse de la distribution de l'intensité des pixels de l'image est l'une des méthodes qui permet la séparation des classes. Plusieurs méthodes de seuillage utilisant l'histogramme des images sont présentées et étudiées dans (Sahoo *et al.*, 1988). Dans cette section, une méthode de seuillage adaptatif basée sur l'histogramme de l'image des observations est proposée. Elle est de faible complexité et répond plus à nos besoins. Ces conditions de validité seront les mêmes que celles du seuillage entropique présenté dans (Luthon *et al.*, 2004b) :

- Caméra fixe.
- Illumination avec une faible variation lente et progressive.
- Le bruit est additif gaussien, non corrélé, de faible intensité par rapport au signal.

– Le mouvement n'est pas prédominant par rapport au champ de la caméra.

Sous ces hypothèses :

- Les pixels appartenant aux objets en mouvement occupent une faible partie de l'image.
- Les pixels appartenant au bruit de fond sont proéminents.

Avec ces hypothèses, l'image de différences comprend une forte population de pixels proches de la valeur nulle. La figure 5 est un exemple d'histogramme d'image de différences d'une séquence de vidéo surveillance.

Un histogramme qui respecte ces hypothèses peut se représenter comme une courbe bimodale. Le premier lobe représente le bruit de fond (à éliminer). Ses caractéristiques sont de posséder une hauteur très supérieure au second et une population nombreuse de faible intensité, proche de la valeur nulle. Le second lobe représente les objets en mouvement. La figure 6 illustre la méthode proposée.

En nous basant sur la méthode de (Rosin, 2001) correspondant à une analyse graphique de l'histogramme pour extraire le premier lobe, nous proposons une estimation de la moyenne  $\mu$  et de l'écart type du fond  $\sigma_b$  à partir de l'histogramme

de l'image. Le premier lobe de cet histogramme est en fonction de l'intensité  $x$  exprimé de la manière suivante :

$$B(x) = \frac{1}{\sigma_b \sqrt{2\pi}} \exp^{-\frac{1}{2} \left(\frac{x-\mu}{\sigma_b}\right)^2} \quad [9]$$

Dans les conditions exposées ci-dessus la valeur moyenne correspond au maximum  $H$  de la gaussienne et se trouve proche de la valeur nulle.  $H$  correspond aussi au maximum de l'histogramme de l'image. Si on se place à la valeur moyenne  $x = \mu$  alors :

$$B(\mu) = \frac{1}{\sigma_b \sqrt{2\pi}} = \max(B(x)) = H \quad [10]$$

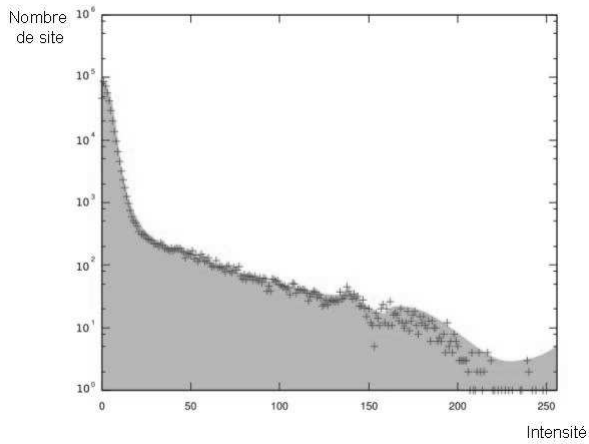


Figure 5. Histogramme des différences

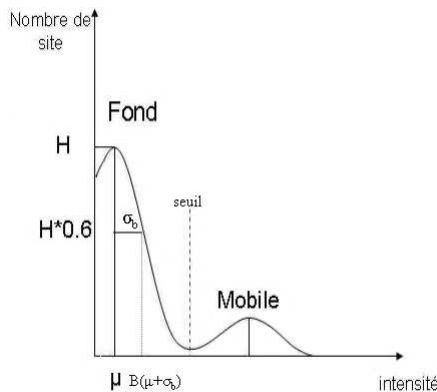


Figure 6. Elimination du fond gaussien

Nous obtenons une relation entre la position sur la gaussienne et la valeur maximum  $H$  :

$$B(x) = H \cdot \exp^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma_b}\right)^2} \quad [11]$$

Pour avoir une estimation de l'écart type, il suffit de calculer  $B(x)$  pour la valeur  $x = \mu + \sigma_b$ , on obtient alors :

$$B(\mu + \sigma_b) = H \cdot \exp^{-\frac{1}{2}} \simeq H \cdot 0.6 \quad [12]$$

Il existe donc une relation entre l'écart type et la hauteur maximale. Si on se place à une hauteur de gaussienne égale à  $P(x) = H \times 0.6$ , la valeur correspondante de  $x$  donne directement une estimation de  $\mu + \sigma_b$ . La moyenne  $\mu$  et l'écart type  $\sigma_b$  peuvent donc être obtenus par un balayage séquentiel de l'histogramme. La valeur de l'écart type sera approximée en se plaçant à une hauteur de  $H \times 0.5$ . Un décalage vers la droite des bits de la valeur de la hauteur de la gaussienne donne une bonne estimation de l'écart type :  $1.1774 * \sigma_b$ . À partir de  $\mu$  et de  $\sigma_b$ , on définit le seuil de binarisation comme étant de la forme :

$$seuil = \mu + k \cdot \sigma_b \quad [13]$$

Le paramètre  $k$  permet d'ajuster le taux d'élimination de la gaussienne de fond. La figure 7 montre la variation automatique de la valeur du seuil en fonction de  $k$ .

### 3.3. Mise à jour de l'image de référence

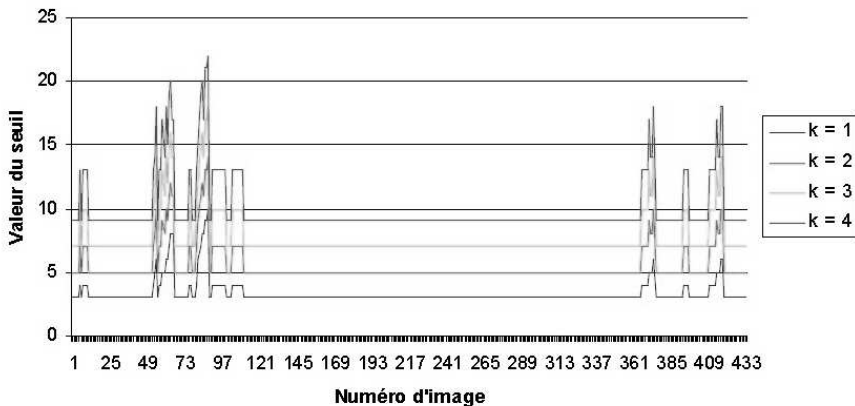


Figure 7. Seuillage en fonction du paramètre  $k$ , séquence Akyio

Dans la littérature, il existe plusieurs stratégies qui permettent la mise à jour de l'image de référence (Vannoorenberghe *et al.*, 1998) (Totozafiny, 2007). Les méthodes liées à la détection des zones de mouvement vont mettre à jour l'image de référence à chaque nouvelle image s'il y a eu un changement dans l'image de

référence. Cette approche donne une détection des zones de mouvement plus précise mais a comme contrepartie d'augmenter le nombre d'images de référence transmises. Pour actualiser l'image de référence, nous avons utilisé une technique d'indexation vidéo. Un des champs de cette discipline est de classifier les effets de transition dans les séquences vidéo (Ahanger *et al.*, 1996) (Oliveira *et al.*, 1997). Les effets de transition que nous voulons détecter sont les changements brutaux de contexte et les changements graduels. Comme nous ne pouvons pas stocker plusieurs images dans les balises, nous ne pouvons pas utiliser les méthodes de prospection temporelle qui sont utilisées dans ce domaine pour la classification. Nous avons opté pour une méthode différentielle. L'algorithme de mise à jour des images de référence est basé sur un détecteur de changement de plan différentiel et sur un détecteur de changement de plan progressif (Ahanger *et al.*, 1996). Pour cela, il nous faut définir deux seuils, soit  $\tau_1$  le seuil haut et  $\tau_2$  le seuil bas, avec  $\tau_1 \gg \tau_2$ , et les comparer au pourcentage  $\tau$  des pixels affectés par le mouvement. Ce pourcentage  $\tau$  est obtenu de la manière suivante : soit  $P(x,y)$  la valeur du pixel de la carte binaire du mouvement  $E(t)$  à la position  $(x,y)$ ,  $l$  et  $c$  étant respectivement le nombre de ligne et de colonne de l'image alors :

$$\tau = \frac{\sum_{x=0}^{l-1} \sum_{y=0}^{c-1} P(x,y)}{l \times c} \times 100 \quad [14]$$

Dans notre approche,  $\tau_1$  et  $\tau_2$  sont choisies de manière empirique à des valeurs égales à 10 % et 40 %. Nous avons considéré trois configurations pour les valeurs de  $\tau$  (tableau 3) :

Tableau 3. Mise à jour de l'image de référence

	Pourcentage $\tau$	Mouvement	Action
1)	$\tau > \tau_1$	Important	Une image de référence est immédiatement émise
2)	$\tau_1 > \tau > \tau_2$	intermédiaire	Si 12 images successives, mettre à jour l'image de référence.
3)	$\tau < \tau_2$	Faible	Il n'y a pas besoin de mettre à jour l'image de référence.

### 3.4. Initialisation de la carte des changements temporels

La carte des changements temporels est la carte binaire du mouvement qui doit être minimisée. Son initialisation (initialisation des étiquettes) est une opération cruciale. Elle permet d'obtenir une carte binaire du mouvement finale possédant une énergie proche du minimum global d'énergie. Elle peut être obtenue par seuillage de l'image des observations. Nous avons utilisé la méthode la plus simple en fixant un seuil de binarisation à une valeur constante. Ensuite, ce seuil est comparé aux valeurs de chaque pixel de l'image (Lohier *et al.*, 2000 ; Caplier, 1995 ; Lacassagne, 2000). Nous notons que l'initialisation pourra être obtenue en effectuant un test de vraisemblance pour chaque point de l'image sur un voisinage de dimension  $3 \times 3$  ou  $5 \times 5$ , si le test est positif alors  $\hat{O}_t = 1$  sinon 0. L'extension de l'algorithme à la détection multi-étiquettes utilise une méthode similaire. Dans le cas où deux images des observations sont utilisées, l'initialisation du champ d'étiquettes est réalisée en

effectuant un OU logique entre les deux cartes binaires obtenues après la binarisation des deux images des observations.

### 3.5. Stratégie de mise à jour des sites

La visite des sites est faite par un parcours simple de l'image qui respecte l'ordonnancement des pixels en mémoire. Pour accélérer la convergence vers un minimum local de l'énergie, nous prenons en compte directement les modifications des étiquettes précédentes dans le calcul de l'énergie du site  $s$  (figure 8).

### 3.6. Masquage

L'image  $O(t)$  étant une image de différence, par conséquent elle est très sensible aux variations d'intensité. L'opération de masquage est nécessaire pour régulariser et éliminer le bruit impulsionnel. Cette opération joue un rôle de filtrage et seules les variations d'intensité jugées pertinentes seront conservées tandis que les autres seront forcées à la valeur nulle. L'opération de masquage est obtenue en effectuant un ET binaire entre la carte binaire du mouvement  $E(t)$  et l'image de différence absolue  $O(t)$ . Dans le même temps, une réduction et un décalage de la dynamique sont effectués de manière à ce que l'image de différence masquée  $D * t$  possède la même dynamique que l'image d'origine. La valeur  $d_t(s)$  de la différence masquée est :

$$d_t(s) = 128 + (1 - 2 * sig_t(s) * \left( \frac{e_t(s) \cdot o_t(s)}{2} \right)) \quad [15]$$

Cette opération force à zéro les pixels de  $O(t)$  qui ne sont pas détectés en mouvement dans la carte binaire du mouvement  $E(t)$ . Plus le nombre de pixels immobiles est important, plus l'opération de masquage simplifie les images de différence absolue.

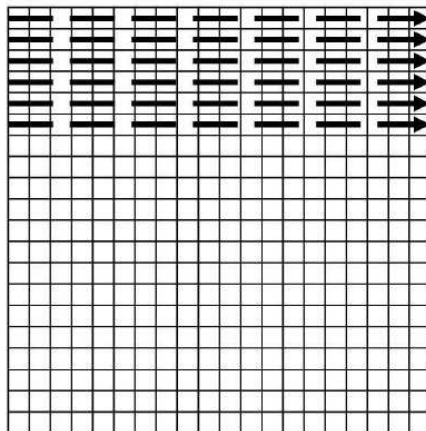


Figure 8. Sens du parcours de l'image

## 4. Évaluation de l'algorithme

La base de test est composée de 14 séquences vidéo couleur possédant des contextes très différents avec un nombre d'images variant de 100 à 1500. Nous présentons ici 2 séquences : Akyio et Survey au format QCIF. Nous avons réalisé une version logicielle de l'algorithme de masquage des images de différence. Nous avons retenu 2 méthodes pour évaluer la qualité :

- Le PSNR (Peak Signal to Noise Ratio). Il donne une mesure statistique des dégradations.
- Estimation avec une métrique de qualité perceptuelle (méthode à double stimulus (ITU-Group, 2004)) : on compare la séquence originale aux séquences dégradées. L'observateur doit alors attribuer une note à l'image dégradée selon une échelle prédéfinie, mais ces tests ont le désavantage d'être coûteux, longs et fastidieux à mettre en œuvre. Les métriques perceptuelles représentent une alternative aux tests subjectifs. Elles exploitent les caractéristiques du SVH pour améliorer la corrélation entre la note qu'elles fournissent et celles données par un ensemble d'observateurs (Tiffany *et al.*, 2001) (Tantaoui *et al.*, 2003). L'évaluation perceptuelle est réalisée avec le logiciel VQM (Video Quality Metric) (Pinson *et al.*, 2004).

### 4.1. Comparaison entre MJPEG2000 et MMJPEG2000

Le système de compression que nous utilisons pour effectuer nos tests est composé de trois modules distincts. Le premier de ces modules est le générateur d'images de différence masquées vu précédemment. Le codeur utilisé est Kakadu (Taubman *et al.*, 2002), il suit la norme JPEG2000. Pour nos mesures nous utilisons 4 niveaux de décomposition en ondelette et l'option « no\_weights » est activée, car nous avons constaté que la compression est de meilleure qualité d'un point de vue statistique. 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. La séparation du flux vidéo en images de références et de différences nous permet de compresser plus fortement les images de différences par rapport aux images de référence. Nous avons fixé le taux de compression des images de différences de manière à avoir un bon rapport qualité/débit. Nous avons créé des séquences avec des taux de compression pour l'image de références de 16 et pour 6 valeurs de compression, variant de 32 à 256, pour les images de différence.

#### 4.1.1. Comparaison statistique

La figure 9 représente l'évolution du PSNR en fonction du débit de la séquence Survey. Elle permet d'effectuer une comparaison statistique entre MMJPEG200 et MJPEG2000. Nous constatons que MMJPEG2000 améliore la qualité statistique de MJPEG2000. En fonction du débit, le gain obtenu varie entre 4 dB et 10 dB.

#### 4.1.2. Comparaison objective avec une métrique de qualité perceptuelle

La figure 10 représente l'évolution du MOS en fonction du débit de la séquence Akiyo. Elle permette d'effectuer une comparaison entre MMJPEG200 et MJPEG2000 en estimant la qualité perçue. Globalement, nous observons bien l'amélioration notée lors de la comparaison statistique sauf pour les faibles taux de compression de l'image de différence.

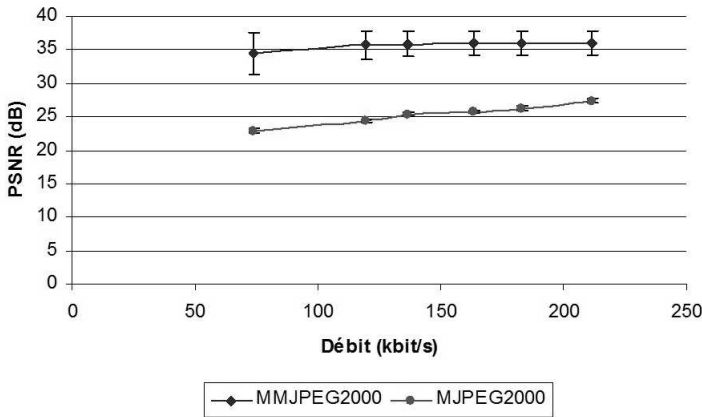


Figure 9. Comparaison statistique MMJPEG2000 vs MJPEG2000 : séquence Survey

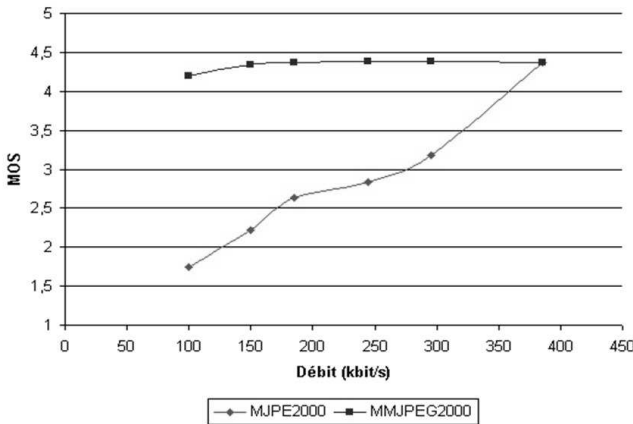


Figure 10. Estimation de la qualité perçue, séquence Akiyo

#### 4.1.3. Comparaison visuelle

La figure 11 montre une partie de la séquence Survey. Elle permet de comparer la qualité visuelle des images compressées par MJPEG2000 et MMJPEG2000 par rapport à la séquence originale. Les séquences sont compressées de manière à obtenir un débit de 120 kbit/s pour la séquence Survey.

#### 4.1.4. Analyse et interprétation des résultats

Pour avoir une qualité d'image acceptable, il faut que l'image de référence présente un PSNR au minimum proche de 35 dB. La qualité de l'image de référence permet d'augmenter globalement la qualité de l'image reconstruite. Toutefois, nous observons localement dans les zones affectées par le mouvement des défauts locaux de zones de flou, irisations plus ou moins gênantes qui dépendent du taux de compression des images de différences. Ces défauts sont majoritairement dus au



« flickering » introduit par JPEG2000 pour les forts taux de compression. Ils ne sont presque pas visibles, pour les séquences Akiyo et Survey, jusqu'à un taux de compression des images de différence voisin de 64, correspondant à un débit de 230kbit/s et 160kbit/s pour une résolution QCIF. Lorsque le taux de compression est dans l'intervalle [64, 256] pour les images de différence, les erreurs de reconstruction sont de plus en plus visibles. La qualité des images obtenues permet toujours l'identification des objets si le nombre de pixels appartenant à l'objet mobile est suffisant. Pour conclure, malgré les défauts locaux que nous venons de mettre en évidence pour les forts taux de compression, nous constatons que globalement les séquences compressées par MMJPEG2000 sont de meilleure qualité du point de vue statistique et visuel que celles compressées par MJPEG2000 à débit équivalent. Les mesures réalisées montrent une augmentation du PSNR qui varie entre 6 et 10 dB.

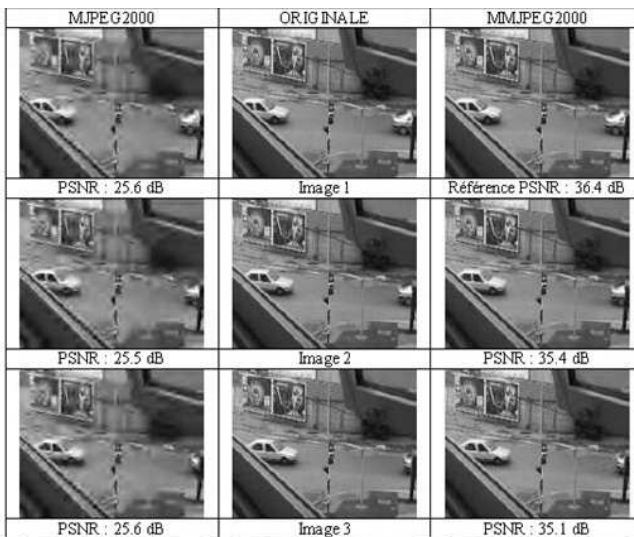


Figure 11. Comparaison visuelle : séquence survey

#### 4.1.5. Analyses et estimation de la durée de traitement de l'algorithme de masquage

L'algorithme markovien couplé à MJPEG2000 nous a permis un gain très intéressant en le comparant aux algorithmes dérivés de MPEG. Le tableau 4 donne une idée sur le gain apporté par cette technique en la comparant à une estimation de mouvement classique utilisé dans les codeurs vidéo MPEGx et H26x.

Tableau 4. Nombre d'opérations par pixel

	MPEG Motion estimation	MRF-Motion Detection	Gain
Full search	$32^2 \times 3op(+, -, abs) = 3072$	$3 \times 15op.(9+, 5LUT, 1<)=45$	x 68
Logarithmic	$(8(\log_2 15) + 1) \times 3 = 99$	$3 \times 15op.(9+, 5LUT, 1<)=45$	x2
Hierarchical L=3	$\{[2 \cdot (15/4) + 1]^2 + 720/4\} / 16\} \times 3 = 49$	$15 * \sum_{k=0, L-1} 1/2^{2k} = 20$	

L'algorithme de masquage est écrit en C++, sans optimisation et pour une seule itération de d'ICM (= MPC). Les mesures sont effectuées sur des séquences couleurs, de tailles différentes, sans sous-échantillonnage des chrominances, sans prendre en compte les accès disques pour le chargement et l'écriture des images. Pour indication, le tableau 5 contient également les mesures effectuées sur des formats CIF et CCIR601.

La génération des maques respecte la contrainte « temps réel » pour des images au format QCIF (tableau 5). Par ailleurs, l'encodeur JPEG2000 Kakadu (Taubman *et al.*, 2002) encode une image Qcif en moins de 15 ms, ce qui permet d'espérer de réaliser un codeur temps réel pour MMJPEG2000. Pour l'optimisation de l'algorithme de détection des zones de mouvement par champs markoviens, on pourra se référer à (Lacassagne, 2000) et (Lohier, 2000).

Tableau 5. Temps de traitement de l'algorithme de masquage

Opération	QCIF(ms)	CIF(ms)	CCIR601(ms)
Soustraction	0.71	2.96	12.54
Seuillage histogramme	0.40	1.99	7.87
Binarisation	1.18	4.1	20.33
Détection référence	0.09	0.58	2.75
MPC	4.73	19.13	78.86
Masquage	0.9	3.54	14.47
Total	8.00	32.3	136.78
Images par seconde	125	31	7

## 5. Conclusion

Nous avons développé et validé un algorithme de compression basé sur la norme JPEG2000 qui est adapté à une intégration dans une balise de vidéosurveillance à bas débit. Pour cela, nous avons choisi comme concept de fusionner un algorithme de détection des zones de mouvements et de compression. Nous avons inséré une opération de masquage directement dans le codeur entre la transformée de couleurs et d'ondelette. La décomposition de l'algorithme en couches fonctionnelles permettra un contrôle de la consommation de la balise. Les modules de compression et de transmission de données ne seront déclenchés que si le détecteur de mouvement constate une anomalie dans son champ de vision. En plus de cette fonction de trigger, la carte binaire du mouvement est utilisée pour effectuer une opération de masquage de l'image de différence. Sous certaines conditions, cette opération permet de simplifier les images de différence et de réduire le flux de données résultant de la compression. Ainsi, avec le même matériel, nous réalisons à la fois une opération de détection des zones de mouvements robustes et de prétraitement qui améliore la qualité de la compression vidéo. Les résultats obtenus montrent que l'algorithme

proposé améliore sensiblement la qualité de MJPEG2000. Cette amélioration pour les forts taux de compression est de l'ordre de 10 dB. Elle permet d'atteindre, pour les séquences au format QCIF, un débit de 150 kbit/s théorique tout en ayant une bonne qualité visuelle. Cette amélioration se constate, pour toutes les séquences dans les formats QCIF, CIF et CCIR, du moment que l'hypothèse d'une caméra fixe est respectée. L'algorithme proposé est capable de s'adapter aux variations de luminosité du milieu extérieur, grâce à l'introduction d'un seuillage automatique qui est fonction des caractéristiques locales de l'histogramme et d'une actualisation de l'image de référence plus souple. Les mesures montrent que le traitement est temps réel pour des images au format QCIF. Elles laissent envisager une intégration d'un codeur MMJPEG2000 dans une balise de vidéosurveillance et une transmission de la vidéo couleur à une cadence de 25 images par seconde en temps réel sur un réseau bas débit de type Zigbee.

## Bibliographie

- Adams M. (2001), The JPEG-2000 Still Image Compression Standard, Rapport de recherche n° N2412, ISO/IEC/JTC1/SC29/WG1, september.
- ADevice, [http://www.analog.com/en/prod/765\\_810\\_ADV601,00.html](http://www.analog.com/en/prod/765_810_ADV601,00.html), Technical report, Analog device, 2004.
- Ahanger G., Little T. (1996), « A survey of technologies for parsing and indexing digital video », *Journal of visual communication and image representation*, vol. 7, n° p. 28-43.
- Akyildiz I., Kaushik T., Chowdhury R. (2007), « Survey on wireless multimedia sensor networks », *The International journal of Computer and Telecommunications Networking*, vol. 51, p. 921-960.
- Bourdon P., Augereau B., Olivier C., Chatellier C. (2003), « A PDE-based method for ringing artefact removal on greyscale and color JPEG2000 images », *IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 3, p. 729-732.
- Caplier A. (1995), Détection de mouvement par approche markovienne, Thèse de doctorat, INPG Grenoble.
- Faura D., Garda P. (2003), « Segmentation d'images couleurs pour la compression de séquences vidéo par l'algorithme Mask, Motion JPEG2000 », *CORESA 2003* Lyon.
- Faura D., Garda P. (2004), « Masked Motion JPEG2000: A new video compression scheme based on JPEG2000 », *8th IEEE International Symposium on Consumer Electronics 2004*, Londres.
- Faura D., Romain O., Garda P. (2006), « MMJPEG2000: A video compression scheme based on JPEG2000 », *IEEE International Conference on Image Processing, ICIP, Atlanta (USA)*.
- Group M. C. (2000), JPEG2000 part-3, ISO/IEC 15444-3, Technical report, ISO-IEC.
- ITU-Group, ITU-R Recommendation BT.1683, Rapport, ITU, 2004.
- Kim J., Kim S., Won C. (2006), « Motion JPEG2000 Coding Scheme Based on Human Visual System for Digital Cinema », *Springer Berlin / Heidelberg Book Advances in Image and Video Technology*, vol. 4319, p. 869-877.
- L. Lacassagne (2000), Détection de mouvement et suivi d'objet temps réel, Thèse de doctorat, Thèse de Doctorat de l'Université P. et M. Curie.

- Lohier F. (2000), Méthodologie de programmation et évaluation des processeurs de traitement du signal parallèles pour le traitement d'images en temps réel, Thèse de doctorat, Thèse de doctorat l'Université P. et Marie Curie.
- Lohier F., Garda P., Lacassagne L. (2000), « Masked-Motion-JPEG2000: A new reduced complexity video sequence compression scheme based on a MRF-motion detection algorithm toward inter frame masking », *Conf. On Signal Processing Applications and Technology*.
- Luthon F., Beaumesnil B. (2004a), « Color and ROI with JPEG2000 for wireless videosurveillance », *International Conference on Image Processing*, vol. 5, p. 3205-3208.
- Luthon F., Caplier A. (1993), « Motion detection and segmentation in image sequences using Markov Random Field Modeling », *4th Eurographics Animation and Simulation Workshop*, vol., p. 265-275.
- Luthon F., Lievin M., Faux F. (2004b), « On the use of entropy power for threshold selection », *Elsevier Science, Signal processing*, vol. 84, n° 10, p. 1789-1804.
- Oliveira I., Correira N., Guimaraes N. (1997), « Image Processing Technique for Video Content Ex-traction », *Image indexing and retrieval*, vol., p. 61-70.
- Pinson M., Wolf S. (2004), « A new standardized method for objectively measuring video quality », *IEEE Transactions on Broadcasting*, vol. 50, n° 3, p. 312-322.
- Riquebourg Y. (1997), Analyse de mouvements articulés : mesure et suivi 2D ; application à la télé-surveillance, Thèse de doctorat, Thèse de doctorat université de Rennes I.
- Romain O., Granado B., Pinna A., Faura D., Hachicha K., Garda P. (2004), « MERITE A versatile platform for wireless sensor network », *IEEE International Symposium on Industrial Electronics*, Ajaccio, vol., p. 125-129.
- Rosin P. (2001), « Unimodal thresholding », *ELSEVIER Pattern Recognition*, vol. 34, p. 2083-2096.
- Sahoo P., Soltani S., Wong A., Chen Y. (1988), « A survey of thresholding techniques », *Computer Vision, Graphics and Image Processing*, vol. 41, p. 233-260.
- Santa-Cruz D., Ebrahimi T. (2000), « An analytical study of JPEG 2000 functionalities », *ICIP'00*, vol. 2, p. 49-52.
- Saponara S., Blanch C., Denolf K., Bormans J. (2003), « The JVT Advanced Video Coding Standard: Complexity and Performance Analysis on a Tool-by-Tool Basis ».
- Skodras A., Christopoulos C., Ebrahimi T. (2001), « The JPEG2000 Still Image Compression Standard », *IEEE Signal Processing Magazine*, vol. 18, p. 36-58.
- Subedar M., Karam L., Abousleman G. (2004), « JPEG2000-Based shape adaptive algorithm for the efficient coding of multiple region of interest », *ICIP'04*, vol. 2, p. 1293-1296.
- Tamtaoui A., Aboutajdine D. (2003), « Mesure de la qualité visuelle des images compressées », *Traitement du Signal*.
- Taubman D. (1999), « High performance scalable Image compression with EBCOT », *ICIP'99*, vol. 9, n° 7, p. 1158-1170.
- Taubman D., Marcellin M. (2002), *JPEG2000, Image Compression, Fundamentals, Standards and Practice*, Kluwer Academic Publishers, Boston.
- Tiffany T., Hakim S. (2001), « Apport d'un bloc de segmentation d'erreur dans l'évaluation de la qualité d'images », *GRETSI01*.

- Totozafiny T. (2007), Compression d'images couleur pour application à la télésurveillance routière par transmission vidéo à très bas débit, Thèse de doctorat, Université de Pau et des pays de l'Adour.
- Totozafiny T., Patrouix O., Luthon F., Coutelier J. (2005), « Motion reference image JPEG2000: Road surveillance application with wireless device », *Visual Communications and Image Processing VCIP'05*.
- Vannoorenberghe P., Motamed C., J. Postaire (1998), « Réactualisation d'une image de référence pour la détection du mouvement dans les scènes urbaines », *Traitement du Signal*, vol. 15, n° 2, p. 139-1484.
- Varma K., Bell A. (2004), « Improving JPEG2000's perceptual performance with weights based on both contrast sensitivity and standard deviation », *ICASSP'04*, vol. 3, p. 665-668.
- Wang Z., Banerjee S., Evans B., Bovik A. (2002), « Generalized bitplane by bitplane shift method for JPEG2000 ROI coding », *ICIP'02*, vol. 3, p. 81-84.
- Yeung Y. (2005), « Efficient Rate Control for JPEG2000 Image Coding », *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, n° 3, p. 335-344.

Article reçu le 9/06/2010  
 Accepté le 15/03/2011



*Le Dr David Faura est titulaire d'un doctorat de l'université de Pierre-et-Marie-Curie spécialité Électronique depuis 2008 sur le thème de l'étude d'une architecture électronique embarquée pour la compression vidéo basée sur JPEG2000. Depuis 2009, il est ingénieur Études Amonts à THALES Avionics. Ses travaux concernent l'étude et la mise en œuvre des futures architectures des plateformes avioniques.*



*Khalil Hachicha : né en Tunisie en 1978. Il a obtenu le diplôme de docteur en électronique en 2005. Il est actuellement enseignant chercheur à l'université de Pierre-et-Marie-Curie à Paris. Son domaine de recherche porte essentiellement sur la modélisation, l'optimisation et la conception des architectures électroniques embarquées.*



**Patrick Garda** est Professeur à l'université Pierre-et-Marie-Curie depuis 1995. Il est actuellement responsable de l'équipe « Systèmes Electroniques » du laboratoire LIP6, responsable de l'électronique au sein de l'école d'ingénieurs Polytech'Paris-UPMC et directeur du GIS francilien eSys. Ses recherches actuelles portent sur la modélisation des performances des systèmes hétérogènes (consommation et intégrité du signal) et les dispositifs médicaux.



**Olivier Romain** est ancien élève de l'ENS de Cachan. Il est actuellement maître de conférences à l'université Pierre-et-Marie-Curie. Il obtint son doctorat et son habilitation en 2001 et 2011, en électronique à l'université Pierre-et-Marie-Curie. Actuellement, ses activités de recherche sont tournées vers la modélisation et la conception de systèmes électroniques pour des applications de radio logicielle et d'implants médicaux.