Insertion de données cachées pour une reconstruction sans perte de Régions d'Intérêts dans des vidéos H.264

Peter MEUEL[1] Marc CHAUMONT[1] William PUECH[1]

[1] Laboratoire LIRMM, UMR CNRS 5506

Université Montpellier II
161 rue Ada, 34392 Montpellier cedex 5

{peter.meuel, marc.chaumont, william.puech}@lirmm.fr

Résumé

Nous proposons dans cet article une méthode de protection de Régions d'Intérêt (RI) par insertion de données cachées. Notre méthode comporte une phase de détection automatique des RI (dans le cas présent, les visages), de masquage de ces RI afin d'empêcher une visualisation non-autorisée et enfin, une insertion de données afin de reconstruire la RI sans perte (PSNR infini). La méthode couvre également la phase de décodage de ces vidéos améliorées avec l'extraction du message, la reconstitution de la RI originale et enfin, sa fusion avec la vidéo décodée. Les atouts majeurs de notre méthode sont le faible surpoids entraîné par l'insertion comparé au gain visuel reconstruit dans la région d'intérêt, la protection de la région d'intérêt et la compatibilité avec le standard H.264 utilisé.

1 Introduction

Depuis quelques années la vidéo-surveillance connaît une croissance exponentielle. Le faible coût et la haute fiabilité des systèmes de caméras font de la vidéo-surveillance un choix de prédilection pour la surveillance de lieux sensibles ou à forte fréquentation. Néanmoins, cette situation nouvelle créée deux problèmes majeurs : celui du respect de la vie privée et celui de la transmission de l'information. Le problème le plus important est celui du respect de la vie privée. L'usage intensif de caméras de vidéo-surveillance dans les villes créée un maillage dense de caméras (Londres compterait plus de 400 000 caméras [1]). Un pirate se créant un accès à toutes les caméras de surveillance d'une ville peut ainsi suivre un individu lambda de son foyer à son lieu de travail ou autre. Ainsi, afin de préserver la vie privée de chacun, il est besoin d'une méthode interdisant l'identification immédiate des personnes filmées sans autorisation adéquate (clé de chiffrement, système de permission...).

Rodrigues *et al.* [2] ont proposé une méthode de chiffrement partiel des coefficients DCT sur des séquences d'images JPEG pour une problématique similaire. Leur méthode offre toutes les caractéristiques recherchés mais sur des séquences d'images JPEG, non transposable directement au format H.264 considéré. Dufaux et Ebrahimi [3] ont proposé une méthode de scrambling de la RI. En considérant la taille moyenne relativement faible des RI (quelques macroblocks au plus), les attaques par force brute rendent cette méthode peu robuste. Plusieurs autres travaux sur la sécurisation de RI [4] ne considèrent que des méthodes de protection destructives : il n'est pas possible de récupérer l'information de la RI. Ce défaut est disqualifiant pour une méthode de protection de séquences de vidéo-surveillance puisqu'une vidéo ne permettant pas d'identifier les protagonistes de la scène est nulle d'intérêt. La solution consistant à stocker deux versions de la même séquence, une protégée, diffusée, et une claire stockée, n'est également pas envisageable pour des raisons de coûts et de problèmes de gestions évidents.

Chen *et al.* [5] ont travaillé sur le codage par RI en utilisant une méthode de détection par couleur. La détection des visages se fait par un calcul colorimétrique basé sur les travaux de [6] et appliquent en pré-traitement un filtre passe-bas afin de réduire l'information de haute fréquence en dehors de la RI afin de minimiser au maximum la taille de codage de cette non-RI. Le processus d'allocation est basé sur le modèle de contrôle de débit TMN8 inclus dans le codeur H.263+. A l'instar de notre méthode, les travaux de Chen délivrent des vidéos compatibles avec le format initial (ici H.263+) mais le codage ne permet pas de protection de la RI, qui reste directement visible par tous.

Agrafiotis *et al.* [7] ont également travaillé sur ce problème mais avec le format H.264. Dans leur méthode, la définition de la RI est faite par l'utilisateur. L'idée est globalement la même que celle développée par Chen [5] mais le modèle de contrôle du débit se fait ici par le choix de différents Paramètre de quantification (QP) avant le codage (et non durant comme on pourrait s'y attendre). Afin d'affiner leur résultats, Agrafiotis *et al.* utilisent deux niveaux de contrôle du débit : au niveau du GOP, bien sûr, mais également au niveau de la frame. Tout comme la méthode précédente, on ne trouve aucun moyen de protection de la RI.

Il s'avère également que le standard H.264 définit un moyen d'encodage par RI en permettant d'attribuer

différents QP à différentes portions (*slices*) de l'image. Cette méthode ne permet cependant de restreindre la visualisation de la RI.
Le deuxième problème est celui de la transmission du flux vidéo. Si la compression vidéo est indispensable pour atteindre des débits raisonnables, de l'ordre du Mb/s au plus, le cas de la vidéo-surveillance se distingue avec des situations dans lesquelles les débits de transmission sont encore plus faible que la normale, de l'ordre de la centaine de kilo-octet par seconde, la vidéo-surveillance d'un bus par exemple. Les deux principales solutions sont évidemment l'amélioration du codec vidéo ou l'augmentation du débit entre la caméra et le lieu de visualisation. Néanmoins, une autre possibilité est possible : optimiser le flux en attribuant plus de bits d'informations aux RI lors de la compression. Ainsi, nous pouvons fortement comprimer l'image tout en gardant une bonne qualité visuelle sur la RI (les visages des personnes filmées).
L'insertion de données cachées est un sujet de recherche particulièrement actif principalement pour des applications de tatouage. Sur ce sujet, [8] constitue un état de l'art de l'insertion de données dans des documents audiovisuels numériques et [9] propose un tour d'horizon des méthodes de tatouages de vidéos.
Notre méthode suit ce raisonnement en "offrant" un plus grand espace de codage à la RI par l'insertion d'information dans l'espace de codage des régions non RI.

2 La méthode proposée

Dans cette section, nous décrivons la méthode proposée. Nous présentons d'abord le fonctionnement global de notre méthode puis nous détaillerons chaque partie séparément.

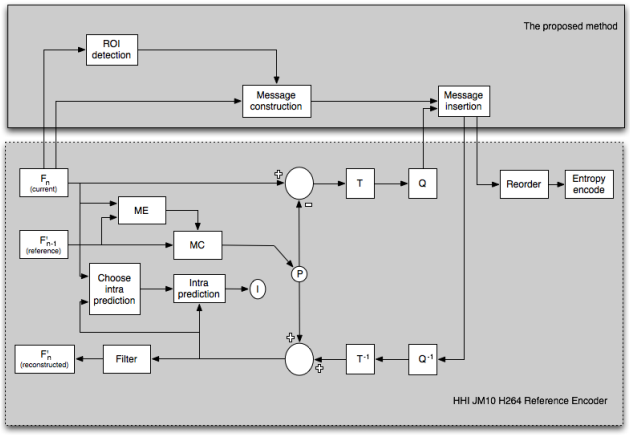

FIG. 1 – *Schéma général de la méthode*

La première étape de notre méthode est la détection de la RI. Notre méthode supporte la détection et la protection de plusieurs RI, moyennant une capacité d'insertion adéquate, mais nous considérons le cas d'une unique RI pour simplifier les explications de notre méthode. Une fois la RI détectée, nous construisons notre message. Ce message est constitué des valeurs de luminance de la RI. Ces valeurs sont codées sans perte premièrement par un codage prédictif puis par un codeur entropique. Nous obtenons ainsi en sortie une séquence w de bits que nous insérons dans les deux Bit de Poids Faible (LSB) des coefficients DCT quantifiés non nuls.
Au décodage, le processus inverse est effectué, à savoir : extraction, décodage entropique puis prédictif, réorganisation puis insertion des valeurs dans la vidéo décodée.

2.1 Détection de la RI

Nous détaillons maintenant la première étape de notre méthode : la détection de la RI, qui se ramène, dans le cas de la vidéo-surveillance, à la détection de visages. Le modèle de notre approche, une détection par colorimétrie, a déjà été étudiée dans [6] et nous nous basé sur la méthode utilisé dans [2]. La première étape de la détection de visage est de marquer comme *pixels de peau* tous les pixels satisfaisant la formule suivante :

$$\sqrt{(P_u - \mathrm{ref_u})^2 + (P_v - ref_v)^2} < d \qquad (1)$$

P_u et P_v sont respectivement la composante U (Cr) et la composante V (Cb) du pixel en cours et d est le seuil de détection en dessous duquel un pixel est considéré comme *pixel de peau*. Cette méthode se base sur la constatation que les "couleurs" de peau ne diffèrent en réalité que par leur luminosité. De façon empirique, nous avons fixé les valeurs de P_u, P_v et d respectivement à 110, 150 et 10 (expérimentations faites dans [2]).
Cette méthode de détection présente deux avantages majeurs : elle est rapide et robuste : tous les pixels représentant une peau sont effectivement détectés comme tels. Malheureusement, cette méthode n'est pas fiable et des artefacts apparaissent : des pixels marqués comme pixels de peau n'ont s'en pas réellement. Afin de résoudre ce problème, nous effectuons une série d'ouvertures et de fermetures successives (opérations morphologiques) afin d'éliminer les petits groupes isolés de pixels.
Au final, nous obtenons un groupe de pixel marqués que nous redéfinissons en une zone rectangulaire, qui constituera notre RI.

2.2 Construction du message

La construction du message s'effectue en trois étapes. La première est la réorganisation des valeurs de la RI précédemment détectée en un vecteur v de n_v bits. Pour des raisons de facilité, nous ajoutons un en-tête contenant les informations de la RI au début de v.
La seconde étape est un codage prédictif c_p du vecteur v en un vecteur v' : $v' = c_p(v)$. Enfin, la dernière étape est un codage entropique c_e du vecteur v' en un vecteur w qui sera le vecteur servant à l'insertion : $w = c_e(v')$.
Une fois le vecteur w obtenu, nous lui appliquons un chiffrement afin de maximiser la sécurisation du message. Nous utilisons l'algorithme AES par flots afin d'obtenir un

message chiffré de la même longueur que le message clair et ce afin de ne pas augmenter la distorsion due à l'insertion.

2.3 Insertion du message

La méthode d'insertion est une substitution des deux LSB des coefficients DCT quantifiés non nuls par deux bits du message calculé. Nous avons choisi l'insertion sur les LSB pour sa haute capacité malgré la faible robustesse et le fort bruit engendré par cette technique.
Le fait d'insérer notre message sur les deux LSB de chaque coefficient non nul génère une grande distorsion, qui, bien qu'elle soit habituellement gênante, ne nous affecte pas puisque qu'elle ne gêne pas la visualisation de la scène et que notre méthode permet une reconstruction de la RI.
Considérons le cas d'une insertion de la séquence 00 dans un octet $0000\ 00xx$. Une simple substitution nous donnerait l'octet 0000 0000 qui serait alors considéré comme nul et dont l'information ne serait pas extraite au décodage. Afin de remédier à ce problème, nous marquons un octet $o = xxxx\ xxxx$ en o_m en utilisant la formule d'insertion suivante :

$$o_m = \begin{cases} 4 \times \text{signof(o) si } 1 \leq |\text{o}| \leq 3 \\ xxxx\ xx00 \text{ sinon} \end{cases} \quad (2)$$

2.4 Protection de la RI

Nous détaillons maintenant les mesures prises pour sécuriser la RI. Nous commençons par supprimer toute information des macroblocks de la RI afin d'empêcher toute attaque. Nous effectuons cette suppression en mettant tous les coefficients des macroblocks de la RI à zéro et en indiquant au codeur H.264 de coder la RI en mode Skip, c'est-à-dire qu'aucune information n'est transmise et que le codeur reconstruira la RI en utilisant l'information insérée par notre méthode.
Ensuite, nous modifions l'encodeur H.264 pour qu'il n'utilise aucun block de la RI pour des prédictions d'autres macroblocks. En effet, dans le cas de prédictions intra, les valeurs des MBS voisins sont utilisés afin prédire le contenu d'un MB. Si à l'encodage, les valeurs de la RI étaient utilisées avant masquage pour le calcul de prédictions des MBs voisins, cela constituerait une porte d'accès à l'information de la RI. Ainsi, nous interdisons à l'encodeur d'utiliser les MBs de la RI pour la prédiction de tout autre MB.

2.5 Reconstruction de la RI

La reconstruction de la RI s'effectue en deux temps. La première étape consiste en l'extraction et le décodage du message inséré par notre méthode. Ceci se fait facilement en testant la parité des deux LSB. La seconde étape est la substitution de la RI décodée par le message extrait dans la vidéo décodée, puisque le message inséré est constitué des valeurs d'origine de la RI.
Afin d'obtenir un PSNR infini avec la vidéo originale, lorsque nous reconstruisons la RI dans la vidéo décodée, nous désactivons le Deblocking Filter (DF). En effet, le DF, une des nombreuses améliorations d'H.264, est une opération qui a pour but de lisser les bords des MBs dans le but d'éviter l'effet de crénelage dû à la DCT. Si son intérêt est appréciable dans la majorité des cas, son application nous empêchait d'obtenir un PSNR infini sur la RI, nous avons ainsi décidé de le désactiver.

3 Résultats

Les résultats ont été obtenus sur une vidéo faite pour l'expérimentation car les vidéos de référence ne satisfont pas la situation traditionnelle de la vidéo-surveillance, à savoir : une caméra fixe filmant des personnes se déplaçant à travers la scène.
La Figure 3 montre les différentes étapes de notre méthode. Pour chaque étape, nous présentants trois instantanées de la vidéo : la 70^e, la 110^e ainsi que la 125^e image. La première ligne représente la vidéo non compressé. La seconde ligne montre l'étape de détection de la RI. La troisième ligne est la vidéo simplement encodé au format H.264. La quatrième ligne correspond à la vidéo encodée via notre méthode mais sans reconstruction de la RI et la cinquième ligne correspond à la vidéo encodée avec notre méthode et avec la reconstruction de la ROI.
Les tables suivantes présentent les résultats numériques de notre méthode. Chaque table présente le PSNR global moyen et le PSNR global de la RI pour la vidéo H.264 normale, la vidéo H.264 marquée et la vidéo H.264 marquée et reconstruite.
Parce que la modification des coefficients DCT est importante, le PSNR de notre vidéo modifiée est inférieur à celui de la vidéo normal. Mais lorsqu'on mesure le PSNR de la RI, notre méthode montre toute son efficacité par un gain net du PSNR.
Notre méthode ne donne pas un PSNR infini sur l'ensemble des images d'une vidéo pour des raisons de manque de capacité. Malgré tout, avec une capacité insuffisante, le gain reste sensible. La prochaine étape sera de rendre le message scalable (par une DCT par exemple) afin d'optimiser le gain lorsque l'espace d'insertion est insuffisant.

Video	PSNR_{All}	PSNR_{ROI}	Insertion
Normal H.264	48.037	53.776	100%
Rec. ROI	47.714	60.560	

FIG. 3 – *Bitrate : 3828 kbits/s*

A un débit de 3828 kbits/s, nous obtenons plus de 20% de frames avec un PSNR infini, c'est-à-dire une reconstruction sans perte de la RI. Sur les autres frames, nous obtenons un PSNR moyen de 60.650 dB.

4 Conclusion

Dans cet article, nous avons présenté une méthode permettant la protection de RI dans une vidéo dans le cadre de

a_1 : frame 75 a_2 : frame 90 a_3 : frame 105

Vidéo H.264 Standard

b_1 : frame 75 b_2 : frame 90 b_3 : frame 105

Vidéo H.264 compressée/décompressée avec notre méthode, sans reconstruction de la RI

c_1 : frame 75 c_2 : frame 90 c_3 : frame 105

Vidéo H.264 compressée/décompressée avec notre méthode, avec reconstruction de la RI

FIG. 2 – *Comparaison visuelle des différents flux*

Video	$\mathrm{PSNR_{All}}$	$\mathrm{PSNR_{ROI}}$	Insertion
Normal H.264	44.582	49.424	100%
Rec. ROI	44.097	56.173	

FIG. 4 – *Bitrate : 2797 kbits/s*

Video	$\mathrm{PSNR_{All}}$	$\mathrm{PSNR_{ROI}}$	Insertion
Normal H.264	40.247	45.466	100%
Rec. ROI	39.569	48.221	

FIG. 5 – *Bitrate : 1882 kbits/s*

la vidéo-surveillance. Notre méthode détecte de façon automatique les visages filmés et attribue, via une méthode d'insertion, un plus grand nombre de bits de codage. De plus, grâce à l'insertion de données cachées, l'information supplémentaire concernant le visage n'est pas directement accessible et on peut la protéger avec une méthode de chiffrement.

Références

[1] C. Norris M. McCahill. On the threshold to urban panopticon ? analysing the employement of cctv in european cities and assesing its social and political impacts. *5th Framework Programme of the European Commission*, Juin 2002.

[2] J. M. Rodrigues, W. Puech, P. Meuel, J.C. Bajard, et M. Chaumont. Face protection by fast selective encryption in a video. *IET THE CRIME AND SECURITY Conference*, pages 420–425, 2006.

[3] F. Dufaux et T. Ebrahimi. Smart video surveillance system preserving privacy. *Proceedings of SPIE*, 5685 :54–63, 2005.

[4] M. Boyle, C. Edwards, et S. Greenberg. The effects of filtered video on awareness and privacy. *Proceedings of the 2000 ACM Conference on Computer supported Cooperative Work*, pages 1–10, 2000.

[5] M.-J. Chen, M.-C. Chi, C.-T. Hsu, et J.-W. Chen. Roi video coding based on h.263+ with robust skin-color detection technique. *IEEE Transactions on Consumers Electronics*, 49 :724–730, 2003.

[6] D. Chai et K. N. Ngan. Face segmentation using skin-color map in videophone applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 9 :551–564, 1999.

[7] D. Agrafiotis, D. R. Bull, N. Canagarajah, et N. Kamnoonwatana. Multiple priority region of interest coding with h.264. *IEEE International Conference on Image Processing*, pages 53–56, 2006.

[8] J. Bloom I. Cox, M. Miller. *Digital Watermarking*. 2001.

[9] G. Doerr et J.-L. Dugelay. A guide tour of video watermarking. *Signal Processing : Image Communication*, 18 :263–282, 2003.