Limitations de la crypto-compression de vidéos HD

Erwan Reinders[1,2] Pauline Puteaux[3] Samuel Brau[2] William Puech[1]

[1] LIRMM, Univ. Montpellier, CNRS, Montpellier, France
[2] DroneGeofencing, Nîmes, France
[3] Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France

Résumé

Dans les scénarios de vidéo surveillance, il est essentiel de sécuriser le contenu visuel pendant la transmission et le stockage. Comme ces séquences vidéo peuvent représenter de gros volumes de données, il est également nécessaire de les compresser. De plus, les capteurs vidéo évoluent constamment, offrant des vidéos à des résolutions de plus en plus élevées. Le codec H.264 est une norme de compression vidéo largement utilisée dans la vidéo surveillance, même aujourd'hui. Dans cet article, nous proposons une analyse des différentes techniques de crypto-compression basées sur le codec H.264, compte tenu de cette évolution de la résolution vidéo. Nous montrons qu'il est quantitativement difficile de trouver des différences entre les vidéos crypto-compressées à basse résolution et à haute résolution, alors que visuellement, une plus grande partie du contenu original est reconnaissable à haute résolution.

Mots clefs

Sécurité multimédia, analyse de la crypto-compression vidéo, évolution de la résolution, sécurité visuelle.

1 Introduction

Les capteurs vidéo des drones modernes fournissent des vidéos en haute définition. Pour en optimiser la transmission, une étape de compression est nécessaire. Dans le but d'être conforme aux normes internationales de compression, la norme d'encodage vidéo H.264 reste le format le plus utilisé pour ce type d'application [1, 2]. Selon la nature de ce que le drone survole, il est essentiel de sécuriser cette transmission vidéo par du chiffrement. Le but du chiffrement vidéo est de s'éloigner du contenu vidéo original, de sorte qu'aucune information visuelle ne puisse en être déduite, tandis que le but de la compression vidéo est de réduire la quantité de données utilisées pour décrire le contenu visuel. Lorsqu'un contenu visuel doit être compressé et chiffré, le chiffrement peut être effectué avant, pendant ou après la compression. En 2018, Chuman *et al.* ont proposé une méthode ETC (Encryption Then Compression) pour chiffrer une image avant de la transmettre par un canal de communication qui effectue une compression JPEG (comme les réseaux sociaux) [3]. Pour réaliser le chiffrement, l'image est découpée en bloc de 8×8 pixels. Ces blocs sont ensuite mélangés entre eux et transformés (rotation, inversion).
Lorsque le chiffrement est effectué après la compression, le contenu compressé est considéré comme un flux binaire. Les méthodes de chiffrement standard, telles que AES par exemple, peuvent être appliquées à l'ensemble de ce flux binaire issu de l'étape de compression [4]. Dans ce cas, le chiffrement modifie la sémantique et la syntaxe des données compressées, qui ne peuvent plus être décompressées.

Enfin, la compression et le chiffrement peuvent être réalisés conjointement, dans ce que l'on appelle les méthodes de crypto-compression. En 2011, Shahid *et al.* ont proposé deux nouvelles méthodes de crypto-compression basées sur AES (en mode CFB) et sur les deux différents codeurs entropiques de la norme H.264, appelées SE-CAVLC et SE-CABAC [5]. En 2013, Dubois *et al.* ont présenté une nouvelle méthode de crypto-compression, basée sur la méthode SE-CAVLC. Cette méthode permet d'ajuster le nombre de coefficients à chiffrer dans une frame, pour la même sécurité visuelle [6]. Pour ce faire, une nouvelle mesure, appelée TSSIM, est proposée. Cette métrique mesure le SSIM [7] de la différence absolue entre une frame originale et la frame précédente, avec la différence absolue entre ces deux mêmes frames mais après chiffrement. Plus la valeur du TSSIM est petite, plus la sécurité visuelle est grande.

Parallèlement à cela, les capteurs vidéo ont considérablement évolué, permettant de produire des vidéos à plus haute résolution. La résolution vidéo est passée de formats QCIF (176×144 pixels) ou CIF (352×288 pixels) à des formats HD (720p : 1280×720 pixels, 1080p : 1920×1080 pixels), voire UHD comme la 4K (2160×3840 pixels).

Dans cet article, nous proposons d'analyser l'évolution de la sécurité visuelle des méthodes de crypto-compression vidéo H.264 appliquées à des vidéos de résolutions de plus en plus élevées. Dans la section 2, nous analysons et dessinons les limites du chiffrement vidéo par rapport à l'évolution de la qualité vidéo. Nous présentons ensuite dans la section 3 une analyse expérimentale de la méthode de crypto-compression de Shahid *et al.* [5]. Enfin, nous concluons en section 4.

2 Analyse théorique et limitation des méthodes de chiffrement vidéo

Dans la section 2.1 nous présentons l'évolution de la résolution vidéo, tandis que dans la section 2.2 nous donnons un aperçu des éléments syntaxiques produits par le codeur H.264. Enfin, dans la section 2.3, nous détaillons la méthode de l'état de l'art sur laquelle nous nous concentrons dans nos expériences.

2.1 Évolution de la résolution des vidéos

Comme illustré figure 1, la résolution des données vidéos a considérablement augmentée depuis la norme CIF (Common Intermediate Format), qui offrait une résolution de 352×288 pixels, ou même QCIF, avec une résolution de 176×144 pixels, vers des normes beaucoup plus élevées, telles que la Full HD (1920×1080 pixels), la 4K (3840×2160 pixels), et même la 8K (7680×4320 pixels). Le format CIF était largement utilisé dans les premiers systèmes de vidéo surveillance et vidéo conférence.

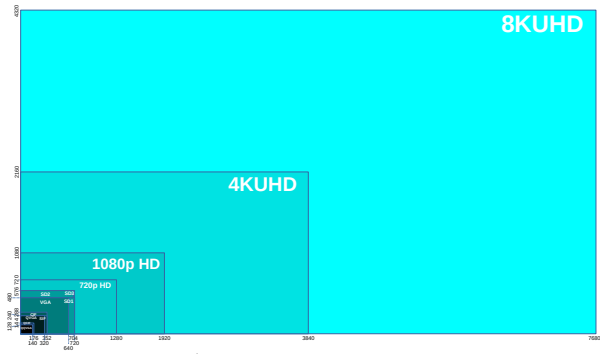


FIGURE 1 – Évolution de la résolution des vidéos.

Dans le cas des vidéos de drones, plus la qualité de la vidéo est bonne et plus celle-ci peut être utilisée ultérieurement (reconnaissance de personnes, suivi des cibles). Cependant, quelle que soit la résolution de la vidéo d'entrée, l'encodeur vidéo H.264 ne traite que les blocs de 16×16 pixels, appelés des macroblocks (MB). Cela signifie que pour une même vidéo enregistrée à différentes résolutions, un MB, en termes de pixels, contient plus d'informations visuelles globales et moins de détails dans une vidéo à basse résolution que dans une vidéo à haute résolution.

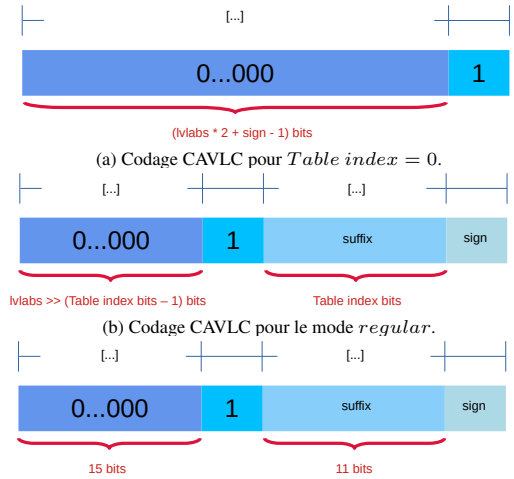
(a) Frame QCIF. (b) Frame HD 1080p.

FIGURE 2 – Contenu d'un MB de 16×16 pixels selon la résolution vidéo. La même frame en résolution : a) QCIF, le MB contient une grande partie de la tête, b) HD 1080p, le MB contient seulement un oeil de la même tête.

La figure 2 illustre un exemple du contenu d'un MB en fonction de la résolution vidéo. Nous constatons que pour la même vidéo, un MB QCIF (figure 2a) couvre plus de contenu qu'un MB 1080p (figure 2b), même si les deux MB contiennent le même nombre de pixels (16×16).

2.2 Éléments syntaxiques dans H.264

L'encodeur vidéo H.264 est composé de différentes étapes, telles que la prédiction, la transformation, la quantification et le codage. Pour obtenir un flux binaire compressé, trois redondances sont exploitées : les redondances spatiales (au sein d'une même frame), temporelles (entre deux frames différentes) et statistiques (codage entropique avec CAVLC ou CABAC).



(a) Codage CAVLC pour $Table\ index = 0$.

(b) Codage CAVLC pour le mode $regular$.

(c) Codage CAVLC pour le mode $escape$.

FIGURE 3 – Schémas du codage des NZ dans CAVLC.

Le codage CAVLC est une méthode de codage adaptative, variable en taille et à tables fixes. Ces tables VLC sont définies arbitrairement dans le codeur et leur choix dépend de seuils fixes. Pour chaque sous-MB de 4×4 pixels d'un MB à coder, les coefficients prédits/transformés/quantifiés sont d'abords ré-ordonnées, de sorte à regrouper les coefficients de même fréquence entre eux. On peut alors dissocier deux grands types de coefficients à coder : les coefficients nuls (codés par RLE) et les coefficients non nuls (ou NZ). Tous les NZ consécutifs égaux à ± 1 en fin d'ordre, au maximum des trois derniers, sont considérés comme des $trailing - ones$, et seul leur signe et leur nombre sont codés. La figure 3 illustre le codage des coefficients NZ restant, avec $lvlabs$ la valeur absolue du coefficient à coder par CAVLC, $sign$ pour le signe de ce coefficient (0 pour positif, 1 pour négatif), et $Table\ index$ faisant référence à la table de longueur fixe utilisée. $Table\ index$ est réévalué pour chaque nouveau NZ à coder dans le sous-MB, et qui n'est pas un $trailing\text{-}one$. La figure 3a montre le processus de codage du coefficient pour $Table\ index = 0$ (le premier coefficient du sous-MB à coder, sous certaines conditions). La figure 3b montre le processus de codage pour toutes les autres valeurs de $Table\ index$, et la figure 3c montre le processus de codage lorsque la valeur du coefficient à coder est supérieure à $15 \ll (Table\ index - 1)$, à l'exception $Table\ index = 0$, où le seuil est de 8.

2.3 Analyse de la résolution sur une méthode spécifique de crypto-compression

En 2011, Shahid *et al.* ont proposé deux méthodes de crypto-compression pour H.264 AVC, appelées SE-CAVLC et SE-CABAC, pour les deux codeurs présents dans H.264 : CAVLC et CABAC [5]. Dans la méthode SE-CAVLC, les coefficients codés dans la sortie du codeur CAVLC sont chiffrés. Pour ce faire, comme l'illustre la figure 4, l'encodeur vidéo prédit d'abord les pixels d'un

MB, par rapport à la même image ou aux voisins temporels (images précédentes ou suivantes), puis transforme uniquement le résultat de cette prédiction à l'aide d'une DCT 4×4 entière, quantifie ces pixels prédits transformés, et les réordonne afin de regrouper les coefficients à coder par fréquences.

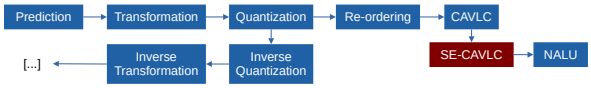


FIGURE 4 – Schéma général de la méthode SE-CAVLC [5].

Dans SE-CAVLC, l'espace de chiffrement est défini comme les valeurs qui ont la même longueur de code par rapport à la table VLC utilisée ($Table\ index$) pour le codage de ce coefficient. Pour ce faire, les coefficients dont $Table\ index = 0$ ne sont pas chiffrés (figure 3a), car seul le suffixe des coefficients codés est chiffré (figure 3b et 3c). Cela permet de maintenir la conformité du format, car sinon la table VLC à sélectionner pour encoder les coefficients suivants pourrait être différente, ce qui introduirait des erreurs de lecture potentielles lors du décodage. Comme c'est la valeur absolue du coefficient qui est prise en compte lors du choix de la table VLC, le signe du coefficient codé est également chiffré. Enfin, le signe des $trailling\text{-}ones$ est chiffré, ce qui n'interfère pas avec le décodage du flux binaire.

3 Analyse expérimentale

FIGURE 5 – Frame $\#45$ de la vidéo $Office$.

La frame brute $\#45$ (figure 5) compressée avec $QP = 18$ produit, pour les résolutions QCIF (176×144), CIF (352×288), HD 720p (1280×720) et HD 1080p (1280×1080) les images illustrées figure 6a, figure 6c, figure 6e et figure 6g respectivement. Dans la figure 6b, la frame $\#45$ a été crypto-compressée en résolution QCIF, tandis que dans la figure 6d la frame $\#45$ a été crypto-compressée en résolution CIF. En augmentant la résolution, dans la figure 6f et figure 6h, nous pouvons voir la même image crypto-compressée pareillement pour des résolution HD 720p et HD 1080p, respectivement. Nous observons, à partir de la même méthode de crypto-compression, que les résultats sont visuellement différents en fonction de la résolution.

La figure 7 illustre le même processus avec $QP = 36$. Nous pouvons observer que plus la quantification est importante et plus l'impact de la crypto-compression sur la sécurité visuelle est faible, et plus la résolution de la vidéo crypto-compressée est élevée et plus il est difficile de maintenir un niveau suffisant de sécurité visuelle. En réalisant une analyse sur les 100 premières frames de la vidéo $Office$ (tableau 1), Nous obtenons un PSNR moyen

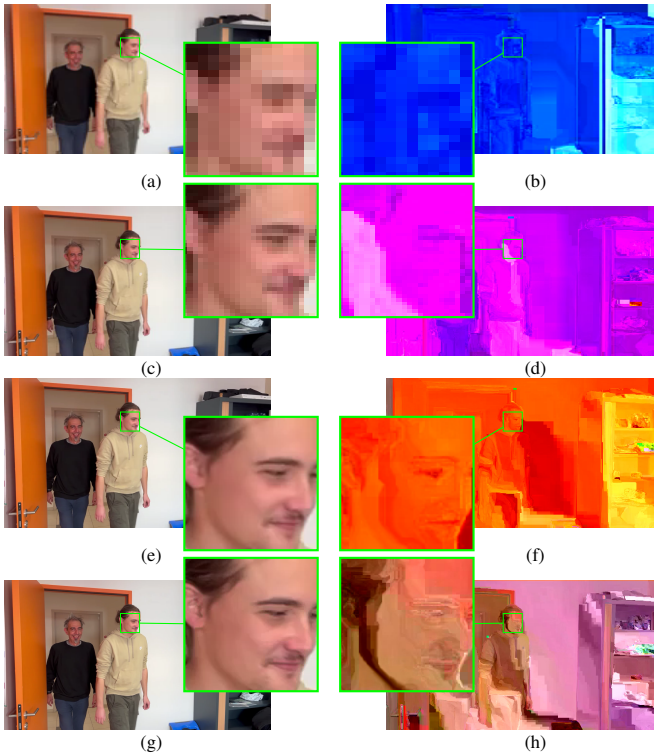


FIGURE 6 – Résultats de la compression et crypto-compression (à différentes résolutions) sur la frame $\#45$ de la vidéo Office avec $QP = 18$: frames compressées en 1^{ere} colonne, et frames crypto-compressées en 2^{eme}. a) et b) En résolution QCIF (176×144), c) et d) En CIF (352×288), e) et f) En HD 720p (1280×720) et g) et h) En HD 1080p (1920×1080).

entre la vidéo originale et les versions crypto-compressées ($QP = 18$) de $8,985\ dB$ pour la vidéo QCIF, $9,577\ dB$ pour la vidéo CIF, $9,442\ dB$ pour la vidéo HD 720p, et $9,753\ dB$ pour la vidéo HD 1080p. Nous ne constatons donc pas d'augmentation réelle du PSNR à mesure que la résolution augmente, alors qu'entre la vidéo originale et les versions compressées, l'augmentation est plus significative, avec $46,123\ dB$ pour la vidéo QCIF, $47,861\ dB$ pour la vidéo CIF, $49,622\ dB$ pour la vidéo HD 720p, et $50,444\ dB$ pour la vidéo HD 1080p. La même analyse est effectuée pour $QP = 36$. Nous pouvons constater qu'entre la vidéo originale et les versions crypto-compressées, les valeurs de PSNR et de SSIM sont similaires pour toutes les résolutions vidéo.

Dans le tableau 2, nous présentons les résultats des métriques UACI et NPCR, entre la vidéo $Office$ originale et celle crypto-compressée, pour les 100 premières frames. Nous pouvons constater que les valeurs de ces métriques sont similaires entre les résolutions, et entre les QPs.

Dans la figure 8, l'algorithme de détection des contours de Canny est appliqué à la composante Y de la frame $\#45$ de la vidéo $Office$ ($\sigma = 0, s_{min} = 25, s_{max} = 50$). Sur les images compressées de la première colonne ($QP = 18$), nous observons que les principales informations visuelles sont accentuées par le filtre de Canny : les deux protagonistes entrant dans le bureau, le cadre de la porte et l'étagère sur la droite. Ces informations sont accentuées en

FIGURE 7 – Résultats de la compression et crypto-compression (à différentes résolutions) sur la frame #45 de la vidéo Office avec $QP = 36$: frames compressées en 1^{ere} colonne, et frames crypto-compressées en 2^{eme}. a) et b) En résolution QCIF (176×144), c) et d) En CIF (352×288), e) et f) En HD 720p (1280×720) et g) et h) En HD 1080p (1920×1080).

QP	Métriques		QCIF	CIF	HD 720p	HD 1080p
18	PSNR (dB)	$OvsC$	46, 123	47, 861	49, 622	50, 444
		$OvsCC$	8, 985	9, 577	9, 442	9, 753
	SSIM	$OvsC$	1	1	1	0, 999
		$OvsCC$	0, 052	0, 057	0, 099	0, 115
36	PSNR (dB)	$OvsC$	32, 571	35, 521	38, 255	39, 599
		$OvsCC$	11, 376	12, 508	11, 915	11, 646
	SSIM	$OvsC$	0, 995	0, 996	0, 999	0, 999
		$OvsCC$	0, 266	0, 295	0, 338	0, 326

TABLEAU 1 – PSNR et SSIM entre la vidéo $Office$ originale et les versions compressées ($OvsC$) et crypto-compressées ($OvsCC$) à des résolutions différentes (100 premières frames).

QP	Métriques	QCIF	CIF	HD 720p	HD 1080p
18	UACI	0, 303	0, 285	0, 281	0, 271
	NPCR	0, 995	0, 994	0, 994	0, 993
36	UACI	0, 223	0, 199	0, 209	0, 217
	NPCR	0, 994	0, 993	0, 993	0, 993

TABLEAU 2 – UACI et NPCR entre la video $Office$ originale et les versions crypto-compressées à des résolutions différentes (100^{ere} frames).

basse résolution, comme QCIF et CIF (1^{ere} et 2^{eme} lignes respectivement), et en haute résolution, comme HD 720p et HD 1080p (3^{eme} et 4^{eme} lignes respectivement). Lorsqu'une étape de crypto-compression est effectuée (dernière

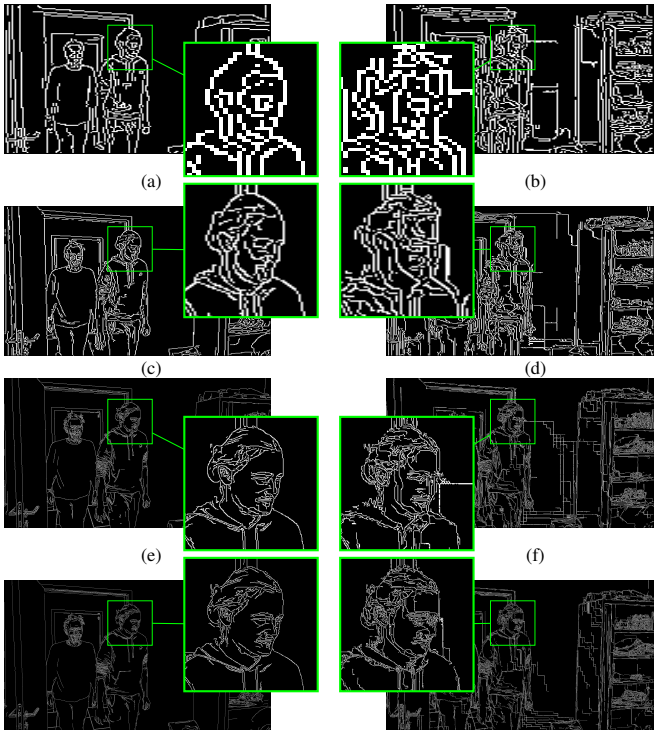


FIGURE 8 – Détection des contours de Canny sur la frame #45 de la vidéo $Office$: frames compressées en 1^{ere} colonne, frames crypto-compressées en 2^{eme}. La 1^{ere} ligne sont des frames en résolution QCIF (176×144), la 2^{eme} en CIF (352×288), la 3^{eme} en HD 720p (1280×720) et la 4^{eme} en HD 1080p (1920×1080).

colonne avec le même $QP = 18$), le résultat est conforme aux analyses précédentes. Sans connaître la vidéo originale, il est difficile de comprendre en basse résolution la construction visuelle de la scène vidéo crypto-compressée. Cependant, plus la résolution est élevée, plus les informations visuelles crypto-compressées sont compréhensibles. Ce phénomène est directement lié à la nature des données chiffrées lors de l'étape de compression.

4 Conclusion

Dans cet article, nous proposons une analyse des méthodes de crypto-compression H.264 sur des vidéos à haute résolution, et démontrons les limites de ces méthodes en termes de sécurité visuelle. Ces limites proviennent principalement de la taille limitée d'un MB dans l'encodeur vidéo. Si, d'un point de vue statistique, une vidéo crypto-compressée semble sûre, quelle que soit sa résolution, ce n'est pas le cas visuellement. Dans cette optique, il serait intéressant de calculer une métrique permettant de mieux mesurer la sécurité visuelle, autrement qu'au moyen d'une métrique appliquée image par image. Cette mesure peut être basée sur une analyse préalable de l'image, telle que la détection des contours. En outre, une seule image vidéo peut parfois suffire au système visuel humain pour comprendre l'ensemble du contenu d'une vidéo crypto-compressée. Il est donc essentiel de prendre en compte l'aspect de l'intégration temporelle afin de sécuriser une vidéo.

Références

[1] ITU Telecom. Advanced video coding for generic audiovisual services. *ITU-T Recommendation H. 264*, 2003.

[2] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, et Ajay Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7) :560–576, 2003.

[3] Tatsuya Chuman, Warit Sirichotedumrong, et Hitoshi Kiya. Encryption-then-compression systems using grayscale-based image encryption for JPEG images. *IEEE Transactions on Information Forensics and security*, 14(6) :1515–1525, 2018.

[4] Vincent Rijmen et Joan Daemen. Advanced encryption standard. *Proceedings of federal information processing standards publications, national institute of standards and technology*, 19 :22, 2001.

[5] Zafar Shahid, Marc Chaumont, et William Puech. Fast protection of H.264/AVC by selective encryption of CAVLC and CABAC for I and P frames. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(5) :565–576, 2011.

[6] Loïc Dubois, William Puech, et Jacques Blanc-Talon. Confidentiality metrics and smart selective encryption for HD H.264/AVC videos. Dans *21st European Signal Processing Conference (EUSIPCO 2013)*, pages 1–5. IEEE, 2013.

[7] Zhou Wang, A.C. Bovik, H.R. Sheikh, et E.P. Simoncelli. Image quality assessment : from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4) :600–612, 2004.