Détection a contrario de la double compression vidéo et application préliminaire à la détection de deepfakes

Yanhao LI Marina GARDELLA Quentin BAMMEY Tina NIKOUKHAH
Jean-Michel MOREL Miguel COLOM Rafael GROMPONE VON GIOI

Université Paris-Saclay, ENS Paris-Saclay, CNRS, Centre Borelli

Résumé – La détection de double compression vidéo peut fournir des indices importants pour récupérer l'historique d'édition et de partage d'une vidéo. En effet, pour manipuler une vidéo, il faut d'abord la décompresser, puis effectuer les éditions désirées et enfin la ré-encoder. Dans cet article, nous proposons une méthode pour détecter la double compression vidéo dans le codec H.264. La méthode proposée détecte la périodicité temporelle des résidus de *frame* causée par le GoP fixe dans la première compression et valide les détections en utilisant un cadre *a contrario* pour contrôler le nombre de fausses alarmes. Les expériences montrent la supériorité de notre méthode par rapport à l'état de l'art sans aucun réglage de seuil et son application préliminaire à la détection de *deepfakes*. Voir `https://github.com/li-yanhao/gop_detection` pour le code.

Abstract – Video double compression detection can provide important clues to recover the video editing and sharing history. Indeed, to manipulate a video, one must first decompress it, then perform the desired editions and finally re-encode it. In this article, we propose a method to detect video double compression in H.264 codec. The proposed method detects the temporal periodicity of *frame* residuals caused by the fixed GoP in the first compression and validates the detections using *a contrario* framework to control the number of false alarms. The experiments show the superiority of our method over the state of the art without any threshold setting and its preliminary application to deepfake detection. Code is available at `https://github.com/li-yanhao/gop_detection`.

1 Introduction

Le développement de logiciels de post-production vidéo a rendu facile et courante l'édition de vidéos, que ce soit à but esthétique où à des fins malveillantes, d'où l'importance de pouvoir évaluer l'authenticité et l'integrité des vidéos[1]. À cette fin, il est utile d'analyser les artefacts liés à la compression des images. En plus des statistiques spatiales de l'analyse d'image [2]-[4], le Group of Pictures (GoP) des vidéos recèle de précieuses statistiques temporelles [5]-[8].

La structure GoP définit différents types de *frames* et leur ordre dans une vidéo, et joue un rôle crucial dans la compression vidéo [9]. Les *frames* dites *intra images* (*I-frames*) sont codées indépendamment des autres ; les *images prédites* (*P-frames*) ne codent que les changements par rapport à la *frame* précédente ; enfin, les *images prédites bidirectionnelles* (*B-frames*) codent les changements par rapport à leur *frame* précédente et suivante. Ces *frames* ont des propriétés différentes : les *I-frames* sont les moins compressibles et indépendantes de leurs voisines, tandis que les *B-* et *P-frames* sont plus compressibles et dépendent également des *frames* voisines. L'analyse de la structure GoP d'une vidéo peut ainsi aider à repérer des traces d'une compression précédente d'une vidéo.

Plusieurs travaux précédents se concentrent sur la détection de la double compression vidéo par analyse de la périodicité laissée par le GoP. Vázquez-Padín et al. [5] analysent la variation temporelle des macroblocs intra-codés et des macroblocs sautés dans les *P-frames*. Chen et al. [6] incorporent des caractéristiques de distribution des résidus de prédiction. Yao et al. [7] utilisent les caractéristiques périodiques de la chaîne de bits de données. Enfin, Yao et al. [8] révèlent les artefacts dans la séquence de comptage des octets de *frame* pour effectuer des détections en cas de GoP adaptatif.

Nous proposons de détecter si une vidéo a été compressée plusieurs fois, en analysant les anomalies dans les artefacts issus de la structure GoP de la première compression. La taille fixe de la structure GoP est utilisée dans le profil de base H.264, et est également un cas courant lorsque aucun changement de scène ne se produit [8], comme l'arrière-plan des vidéos deepfakes de remplacement de visages.

2 Analyse de la double compression

Nous supposons une taille GoP constante pendant la compression et commençons par étudier la structure GoP basée uniquement sur les images I et P (voir la figure 1), comme considéré dans [5]-[8]. Chaque GoP commence par une image I, suivie d'images P. Une image P encodée de manière unique ne code que la différence par rapport à son image de référence, elle-même encodée de manière unique, sous la forme d'une résiduelle de prédiction, qui est ensuite stockée en version quantifiée. La quantification du résiduel de prédiction en perte de qualité corrèle une image P avec l'image précédente.

Après la deuxième compression, une *P-frame* est soit P-P, soit I-P, selon si elle était déjà une *P-frame* à la première compression ou était une *I-frame* déplacée en *P-frame* lors de la seconde compression. Contrairement à une *P-P-frame*, une *I-P-frame* n'est pas corrélée à sa *frame* de référence *avant* la deuxième compression, les résidus *après* la deuxième compression ont tendance à être beaucoup plus élevés dans les *I-P-frames* que dans les *P-P-frames*. Ce phénomène est représenté dans la figure 1, où des pics périodiques anormaux apparaissent après la deuxième compression. De plus, étant donné que la taille de GoP est constante et que les *I-frames* sont également espacées dans la première compression, les *I-P-frames* dans la deuxième compression sont également espacées de manière égale et forment des pics de résidus périodiques avec une période égale à la taille de GoP primaire. Nous pou-

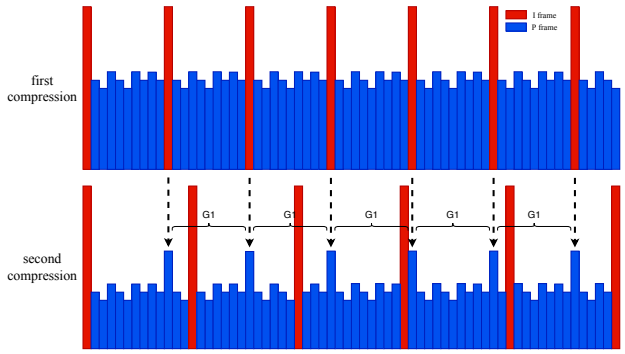

FIGURE 1 : Un diagramme des artéfacts dans les résidus de prédiction des *P-frames* dans une vidéo recompressée (pics anormaux périodiques). G1 est la taille de GoP de la 1ère compression.

vons ainsi détecter la double compression en trouvant une séquence de pics de résidus périodiques dans les *P-frames*.

3 Méthode Proposée

Notre méthode *a contrario* consiste à détecter les pics résiduels périodiques dans les images P d'une vidéo pour décider si la vidéo a été recompressée. Soit $R_t \in \mathbb{R}^{H\times W}$ le résidu de prédiction dans l'espace de luminance d'une image P au temps t ; nous utilisons $r_t = \frac{1}{HW}\sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|R_t(i,j)|$ comme résidu de l'image.

3.1 Un cadre de détection *a contrario*

Le cadre de détection *a contrario* [10] est basé sur le *principe de non-accidentalité* qui affirme qu'une structure est pertinente lorsqu'une grande déviation par rapport à son espérance est observée. L'idée principale est de contrôler le nombre de faux positifs d'un événement sous l'hypothèse nulle H_0. Soit $\mathbf{r} := \{r_t$: la t-ème *frame* n'est pas une *I-frame*$\}$ une variable aléatoire multivariée représentant une séquence de résidus de prédiction des *P-frames*. L'hypothèse nulle H_0 est que la vidéo n'a pas été recompressée, ainsi sous H_0 il n'y a pas de pics résiduels périodiques dans les *P-frames* ; les *P-frames* proches les unes des autres devraient avoir des résidus similaires. S'il y avait une double compression, il devrait y avoir une séquence périodique de résidus de *P-frames* $S(p,b,\mathbf{r}) := \{r_b, r_{b+p}, r_{b+2p}, ...\} \cap \mathbf{r}$ commençant à r_b avec une période p qui ont plus de chances d'avoir des valeurs plus grandes que leurs voisins. Étant donné une séquence candidate $S(p,b,\mathbf{r})$, nous comptons ses éléments supérieurs à leurs d résiduels de *P-frames* voisins de chaque côté :

$$k(p,b,\mathbf{r}) := \sum_{r\in S(p,b,\mathbf{r})} \mathbb{1}_{\left\{r \geq \max B_d(r)\right\}}, \tag{1}$$

où $B_d(r) \subset \mathbf{r}$ est l'ensemble des résiduels d'un voisinage de r comprenant d *P-frames* avant et d *P-frames* après dans $\mathbf{r}$.

Si suffisamment de pics sont présents dans la séquence $S(p,b,\mathbf{r})$, la séquence est considérée significative. Un seuil $\kappa(p,b)$, dépendant des paramètres de la séquence, doit ainsi être fixé pour valider la séquence sous $k(p,b,\mathbf{r}) \geq \kappa(p,b)$. Soit C l'ensemble candidat contenant toutes les paires possibles de (p,b), le nombre total de détections est donné par :

$$D(\mathbf{r}) := \sum_{(p,b)\in C} \mathbb{1}_{\{k(p,b,\mathbf{r})\geq\kappa(p,b)\}}. \tag{2}$$

κ est défini afin que le nombre attendu de détections $D(\mathbf{r})$ sous H_0 soit inférieur à un seuil ϵ :

$$\mathbb{E}\left[D(\mathbf{r})\right] = \sum_{(p,b)\in C} \mathbb{P}\Big(k(p,b,\mathbf{r}) \geq \kappa(p,b)\Big) < \epsilon. \tag{3}$$

La valeur de κ est choisie en appliquant la correction de Bonferroni [11]. Nous regroupons tous les (p,b) possibles dans C par période et divisons ϵ en parts égales entre eux, puis en parties égales entre tous les décalages possibles pour chaque période, qui, pour un p donné, sont $b \in [0, p-1]$. Comme nous avons besoin de d résidus voisins des deux côtés d'un résidu testé, la distance entre deux résidus testés ne doit pas être inférieure à $2d+1$. La période maximale doit être inférieure à la moitié de la longueur de la séquence. Étant donné une vidéo de n images, les périodes possibles sont donc $p \in \left[2d+1, \lfloor\frac{n-1}{2}\rfloor\right]$. Avec cette partition, un couple (p,b) est attribué à $\kappa(p,b)$ choisi comme

$$\kappa(p,b) = min\left\{\eta,\ \mathbb{P}\Big(k\left(p,b,\mathbf{r}\right) \geq \eta\Big) < \frac{\epsilon}{N(p,b)}\right\}, \tag{4}$$

où $N(p,b)$ est le nombre de périodes possibles multiplié par le nombre de décalages possibles pour p, à savoir $N(p,b) = (\lfloor\frac{n-1}{2}\rfloor - 2d)p$. En procédant ainsi, l'équation 3 est satisfaite.

Considérons maintenant une séquence observée de *P-frames* résiduels X comme une réalisation de $\mathbf{r}$. Un candidat (p,b) nous donne le nombre de pics $K = k(p,b,X)$. Au lieu de comparer directement $\kappa(p,b)$ et K, le Nombre de Fausses Alarmes (NFA) de ce candidat est calculé

$$\text{NFA}(p,b,X) := N(p,b)\cdot\mathbb{P}\big(k\left(p,b,\mathbf{r}\right) \geq K\big), \tag{5}$$

et le candidat est validé si $\text{NFA}(p,b,X) < \epsilon$.

Enfin, la probabilité sous H_0 qu'un résidu testé soit un pic plus grand que tous ses $2d$ voisins est :

$$\rho = \mathbb{P}\Big(r \geq \max B_d(r)\Big) = \frac{1}{2d+1}. \tag{6}$$

Étant donné que chaque observation de pic résiduel utilise des voisinages disjoints pour la validation de crête, deux résidus testés étant indépendants, le nombre d'éléments de crête dans une séquence $S(p,b,X)$ suit une distribution binomiale $k(p,b,\mathbf{r}) \sim \mathcal{B}\big(\#S(p,b,\mathbf{r}),\ \frac{1}{2d+1}\big)$ et le NFA est donné par :

$$\begin{aligned}\text{NFA}(p,b,X) &= N(p,b)\cdot\mathbb{P}\left(k(p,b,\mathbf{r}) \geq K\right)\\ &= \left(\left\lfloor\frac{n-1}{2}\right\rfloor - 2d\right)p\mathbf{B}\left(K,\ \#S(p,b,\mathbf{r}),\ \frac{1}{2d+1}\right)\end{aligned} \tag{7}$$

où n est le nombre de *frames* de la vidéo, d est la portée de chaque voisinage de test, $\#S(p,b,\mathbf{r})$ est la longueur de la séquence périodique testée, K est le nombre observé de résidus de crête et $\mathbf{B}$ est la queue de la loi binomiale : $\mathbf{B}(l,m,\rho) = \sum_{i=l}^{m}\binom{m}{i}\rho^i(1-\rho)^{m-i}$.

3.2 Adaptation aux vidéos avec *B-frames*

La méthode ci-dessus ne fonctionne que sur des vidéos recompressées ne contenant que des images I et P, où l'augmentation anormale des résidus se produit strictement sur des images I-P périodiques. Cependant, lorsqu'on considère des vidéos avec des *B-frames*, une *I-frame* peut également être relocalisée en image B lors de la seconde compression. L'augmentation périodique des résidus se produit ainsi à la fois dans les images I-P et I-B. Néanmoins, nous ne pouvons pas comparer directement les résidus d'une *P-frame* et d'une *B-frame* en raison des taux de compression différents. Heureusement, l'augmentation anormale des résidus dans une image I-B peut également être trouvée dans son *P-frame* suivante (voir figure 2). Cela est dû

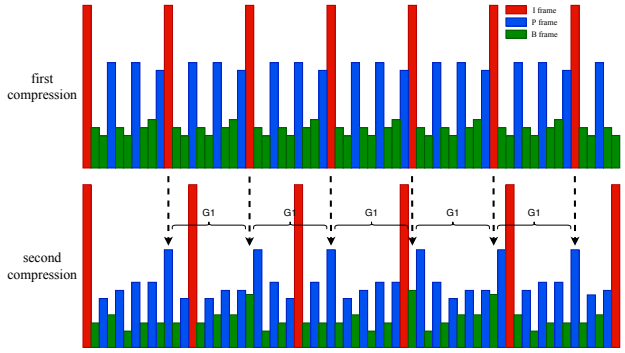

FIGURE 2 : Un diagramme des artefacts dans les résidus de prédiction des *P-frames* dans une vidéo compressée avec des *B-frames*, présentant des pics anormaux quasi périodiques. Aux 2ème, 4ème et 5ème flèches, l'augmentation anormale des résidus se produit dans la *P-frame* suivante d'une *I-B-frame*.

au fait que lors de la seconde compression, la *P-frame* juste après une image I-B utilise une image de référence qui n'appartenait pas au même groupe d'images que lui-même dans la première compression, ce qui entraîne des résidus de prédiction plus élevés par rapport à d'autres *P-frames* qui reposent sur des images de référence du même groupe d'images.

Cette observation nous amène à adapter la méthode aux vidéos avec des images B : avec la même définition de $\mathbf{r}$, pour chaque séquence candidate $S(p, b, \mathbf{r}) = r_b, r_{b+p}, r_{b+2p}, ... \cap \mathbf{r}$, au lieu de la tester directement, nous testons $\tilde{S}(p, b, \mathbf{r}) := \tilde{r}_b, \tilde{r}_{b+p}, \tilde{r}_{b+2p}, ... \cap \mathbf{r}$ où l'opérateur $r_i \mapsto \tilde{r}_i$ associe chaque résidu au résidu de son immédiat *P-frame* suivant, y compris lui-même. Les $\tilde{r}_i$ seront supprimés s'il n'y a pas de *P-frame* suivant pour r_i, ou s'il y a une image I entre r_i et $\tilde{r}_i$ qui couvre l'augmentation anormale du résidu.

4 Résultats Expérimentaux

4.1 Détection de double compression vidéo

Pour comparer la méthode proposée avec [5]-[7], nous avons d'abord sélectionné 19 séquences YUV non compressées[1]. Chaque vidéo a été divisée en groupes de 400 images maximum. En suivant les paramètres de [6] et [7] et en utilisant le logiciel *ffmpeg* avec l'encodeur *libx264*, la première compression a été effectuée avec différentes constantes de débits $B1 \in \{300, 700, 1100\}$ kbps et des tailles GoP $G1 \in \{10, 15, 30, 40\}$, tandis que pour la deuxième compression, les débits étaient $B2 \in \{300, 700, 1100\}$ kbps et les tailles GoP étaient $G2 \in \{9, 16, 33, 50\}$. Plus le débit est élevé, moins la compression est forte. Au total, l'ensemble de données construit compte 228 vidéos compressées une fois et 2736 vidéos compressées deux fois. Étant donné que les méthodes comparées ne fonctionnent que sur des vidéos sans images B, les vidéos ont d'abord été compressées uniquement avec des images I et P pour la comparaison. De plus, les méthodes comparées ne détectent que les vidéos recompressées sans aucun décalage temporel et ne cherchent que des signaux périodiques commençant à la première image. Par conséquent, nous imposons de ne détecter que des séquences périodiques à décalage nul ($b = 0$). Le nombre de voisins de chaque côté pour valider une résiduelle maximale a été fixé à $d = 3$.

Les courbes Précision-Rappel (PR) sont calculées dans la figure 3 sur un nombre égal de vidéos compressées une et deux

[1] *akiyo*, *bridge_close*, *bridge_far*, *city*, *crew*, *deadline*, *flower_garden*, *football(a)*, *foreman*, *galleon*, *harbour*, *ice*, *highway*, *mad900*, *mthr_dotr*, *paris*, *students*, *soccer* et *sign_irene* provenant de https://media.xiph.org/video/derf/

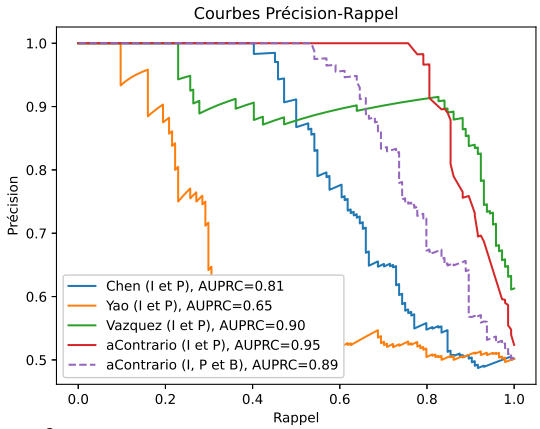

FIGURE 3 : Les courbes Précision-Rappel et aire sous la courbe sur l'ensemble des vidéos testées. Les courbes pleines correspondent aux vidéos encodées avec uniquement des images I et P, celle en pointillés est associée aux mêmes vidéos encodées avec des images I, P et B pour notre méthode.

fois. Notre méthode obtient la meilleure AUPRC (aire sous la courbe), mais aussi le meilleur rappel à précision parfaite, point important en criminalistique où le risque de faux positif doit être évité.

Nous avons aussi approfondi la comparaison avec l'ajustement des seuils. 12 des séquences brutes (1872 vidéos) ont été sélectionnées comme ensemble d'entraînement pour trouver les paramètres empiriques de [5]-[7], validés sur les 7 autres séquences (1092 vidéos). Le seuil de chaque méthode est choisi pour fixer la précision sur l'ensemble d'entraînement à 95%. Pour notre méthode, le seuil ϵ représente la limite supérieure du nombre attendu de faux positifs, qui ne doit pas être supérieur à 1. Par conséquent, nous avons fixé ϵ à 1 et 0,1. Les résultats sur l'ensemble de test sont indiqués dans le tableau 1. Un choix simple et raisonnable de ϵ suffit pour que notre méthode surpasse les autres pour toutes les mesures. Cela est dû au fait que le choix de ϵ utilisé ne dépend pas des caractéristiques des vidéos, tandis que les méthodes comparées nécessitent des paramètres empiriques adaptés à chaque vidéo. Notre méthode présente donc une meilleure généralisabilité, et dépend de plus très peu du choix précis du seuil.

Enfin, pour évaluer les performances de la méthode proposée sur des vidéos contenant des *B-frames*, les mêmes séquences ont été compressées avec les mêmes paramètres, à l'exception de l'utilisation de *B-frames*. Ensuite, la détection a été effectuée en utilisant l'adaptation décrite dans la section 3.2. La courbe PR obtenue est montrée par la courbe en pointillé dans la figure 3. Comme on peut le voir, la présence de *B-frames* rend la détection de la double compression plus difficile, mais à 100% de précision, notre méthode est encore capable de récupérer plus de 50% des vidéos recompressées.

	Précision	Rappel	F1
Chen [6]	0,976	0,578	0,726
Yao [7]	0,660	0,419	0,512
Vázquez-Padín [5]	0,667	0,075	0,135
Méthode proposée, $\epsilon = 1$	**1,000**	**0,969**	**0,984**
Méthode proposée, $\epsilon = 0{,}1$	1,000	0,958	0,979

TABLE 1 : Précisions, rappels et scores F1 sur l'ensemble de test. Les méthodes comparées utilisent des seuils présélectionnés dans l'ensemble d'entraînement, tandis que notre méthode n'utilise que des valeurs de ϵ choisies manuellement comme borne supérieure du nombre attendu de fausses détections.

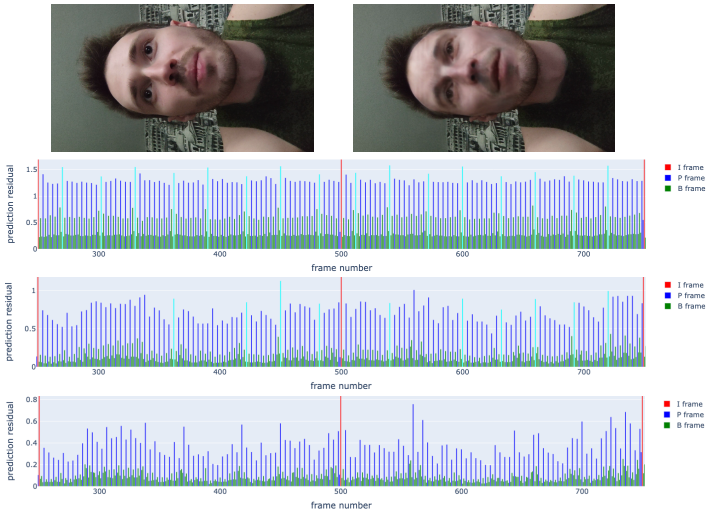

FIGURE 4 : Les pics résiduels périodiques (en cyan) sont détectés en arrière-plan (1er diagramme) et sur le visage (2e diagramme) dans une vidéo authentique recompressée (l'aperçu à gauche), alors que aucune périodicité n'est détectée sur le visage (3e diagramme) dans la même vidéo falsifiée (l'aperçu à droite).

4.2 Application locale de la méthode

La méthode proposée permet de détecter la double compression d'une vidéo dans sa globalité. Nous évaluons maintenant son applicabilité à la détection de *deepfakes*, qui nécessite une analyse locale des traces de double compression. Nous considérons donc le cas d'une vidéo compressée qui est falsifiée puis recompressée. La falsification effaçant souvent les traces de compression, on suppose que la zone falsifiée n'est compressée qu'une fois, alors que l'arrière-plan l'est deux fois.

Pour vérifier cela, nous avons fait les expériences suivantes : nous avons utilisé 8 vidéos [2] puis nous avons appliqué MobileFaceSwap [12] pour remplacer les visages originaux par 8 portraits différents, et recompressé les images truquées avec la configuration par défaut de *ffmpeg* pour avoir 8 fausses vidéos. De l'autre côté, nous avons directement recompressé les vidéos authentiques. La méthode proposée a été appliquée pour détecter des pics résiduels périodiques dans la zone du visage et en arrière-plan de ces vidéos recompressées. Les visages ont été extraits par Mediapipe [3], et les résidus sur le visage (resp. en arrière-plan) ont été calculés localement en moyennant les résidus de prédiction absolus sur le visage (resp. en arrière-plan). Les résultats sont visibles en figure 4.

Parmi les 8 vidéos falsifiées, 6 vidéos avaient des pics résiduels périodiques détectables en arrière-plan, et aucune vidéo n'en avait dans la zone du visage. Cette expérience préliminaire montre que le truquage de contenu par réseaux de neurones efface les résidus périodiques dans la zone falsifiée tandis que cette périodicité est souvent conservée dans l'arrière-plan authentique, ce qui pourrait être exploité afin de distinguer une fausse vidéo d'une vidéo authentique. Cependant, parmi les 8 vidéos authentiques recompressées, la périodicité des pics résiduels a été détectée en arrière-plan dans 6 vidéos, alors que celle n'a été détectée sur le visage que dans 2 vidéos. Cette trace périodique est donc fragile face aux mouvements importants dans la vidéo, qui sont fréquents dans la zone du visage.

[2] téléchargées depuis https://www.kaggle.com/datasets/tapakah68/selfies-and-video-dataset-4-000-people

[3] https://pypi.org/project/mediapipe

5 Conclusion

Nous proposons une méthode *a contrario* pour détecter la double compression dans les vidéos H.264, en détectant la périodicité des résidus des *P-frames*, également adaptée aux vidéos à *B-frames*. Cette méthode surpasse l'état de l'art et ne dépend que peu d'un réglage des seuils. Une évaluation préliminaire sur des vidéos *deepfakes* a été effectuée pour montrer sa possibilité de distinguer ces vidéos truquées des authentiques.

À l'avenir, le détecteur proposé pourra être combiné avec d'autres caractéristiques distinctives au niveau de la *frame* afin d'améliorer la robustesse de la trace périodique aux recompressions différentes et aux mouvements importants dans la scène.

Remerciements Ces travaux ont reçu le financement de l'ANR (Apate, ANR-22-CE39-0016), de l'UE (Horizon Europe VERA.AI, n° 101070093), de la Région Île-de-France et d'UDOPIA (UDOPIA-ANR-20-THIA-0013). Le Centre Borelli est également membre de Université Paris Cité, SSA, et INSERM.

Références

[1] N. SHELKE et S. SINGH KASANA, "A comprehensive survey on passive techniques for digital video forgery detection", *Multimedia Tools and Applications*, t. 80, p. 1-64, fév. 2021.

[2] T. NIKOUKHAH, J. ANGER, T. EHRET, M. COLOM, J.-M. MOREL et R. G. von GIOI, "Jpeg grid detection based on the number of dct zeros and its application to automatic and localized forgery detection", in *IEEE/CVF CVPRW*, 2019.

[3] Q. WANG et R. ZHANG, "Double JPEG compression forensics based on a convolutional neural network", *EURASIP Journal on Information Security*, t. 2016, n° 1, p. 1-12, 2016.

[4] Q. BAMMEY, T. NIKOUKHAH, M. GARDELLA, R. G. von GIOI, M. COLOM et J.-M. MOREL, "Non-Semantic Evaluation of Image Forensics Tools : Methodology and Database", in *IEEE/CVF WACV*, jan. 2022, p. 3751-3760.

[5] D. VAZQUEZ-PADIN, M. FONTANI, T. BIANCHI, P. COMESAÑA, A. PIVA et M. BARNI, "Detection of video double encoding with GOP size estimation", in *WIFS*, 2012.

[6] S. CHEN, T. SUN, X. JIANG, P. HE, S. WANG et Y. Q. SHI, "Detecting double H. 264 compression based on analyzing prediction residual distribution", in *IWDW*, 2017.

[7] H. YAO, S. SONG, C. QIN, Z. TANG et X. LIU, "Detection of double-compressed H. 264/AVC video incorporating the features of the string of data bits and skip macroblocks", *Symmetry*, t. 9, n° 12, p. 313, 2017.

[8] H. YAO, R. NI et Y. ZHAO, "Double compression detection for H. 264 videos with adaptive GOP structure", *Multimedia Tools and Applications*, t. 79, p. 5789-5806, 2020.

[9] I. E. RICHARDSON, *The H. 264 advanced video compression standard*. John Wiley & Sons, 2011.

[10] A. DESOLNEUX, L. MOISAN et J.-M. MOREL, *From gestalt theory to image analysis : a probabilistic approach*. Springer Science & Business Media, 2007, t. 34.

[11] Y. HOCHBERG et A. TAMHANE, *Multiple Comparison Procedures*, English. John Wiley, 1987.

[12] Z. XU, Z. HONG, C. DING et al., "Mobilefaceswap : A lightweight framework for video face swapping", in *Proceedings of the AAAI Conference on Artificial Intelligence*, t. 36, 2022, p. 2973-2981.