Étude sur l'inversion de StyleGAN dans un contexte de détection de d'hypertrucages

Matthieu DELMAS[1,2] Amine KACETE[2] Stephane PAQUELET[2] Simon LEGLAIVE[1] Renaud SEGUIER[1]

[1]CentraleSupélec, IETR (UMR CNRS 6164)

[2]IRT b<>com

Résumé – Depuis plusieurs années les méthodes de trucage vidéo à base d'intelligence artificielle (hypertrucages) se multiplient. Des modèles d'apprentissage automatique pour la détection existent, mais ils ne fonctionnent de manière optimale que dans un contexte donné, sur des méthodes de falsification précises avec un accès à la bonne base de données d'entraînement. Pour palier ces problèmes, nous proposons de projeter les images suspectes dans un espace de plus faible dimension pour faciliter la mise au point de modèles de détection. Concrètement, nous comparons différentes méthodes d'inversion d'images dans l'espace latent de StyleGAN dans le contexte de la détection d'hypertrucages et nous montrons lesquelles sont les plus performantes en terme de précision ou de temps d'inversion sur la base « DeepFakeDetectionChallenge (DFDC) preview ».

Abstract – For the last years, the forgery of videos using artificial intelligence (the creation of deepfakes) has been on the rise. Some detection models exist, but they require a specific context to function optimally. In order to progress towards more stable and easier-to-train detection methods, we work on reducing the dimensionality of the data while keeping as most information as possible before training our classification models. In particular, we compare different StyleGAN inversion methods in the context of deepfake detection. We show how such methods (and which ones) can compare to or even outperform state-of-the-art models on the DFDC preview database.

1 Introduction & Contexte

La contrefaçon de vidéos par intelligence artificielle, la création de ce qu'on appelle des hypertrucages, ou *deepfakes*, est de plus en plus facile depuis quelques années grâce à la progression fulgurante des modèles génératifs, et à la facilité de propager du contenu fallacieux via les média. Bien que ces techniques offrent de nouvelles perspectives dans différents domaines (par exemple, pour les effets spéciaux et la génération de données), sa démocratisation comporte des risques. Il est maintenant facile pour des utilisateurs malveillants de créer des vidéos truquées à des fins d'usurpation d'identité de personnalités politiques, de discrédit de sources fiables ou de chantage. Leur qualité a continuellement progressé depuis leur apparition en 2017 et il est maintenant difficile à l'œil nu de les distinguer de vidéos authentiques, disposer d'un modèle de détection entraîné spécifiquement devient nécessaire. La détection d'hypertrucages consiste à décider de l'authenticité ou non d'une vidéo comprenant un visage. Dernièrement, ces traitements étaient réalisés au moyen de réseaux de neurones à convolution (CNNs) comme MesoNet[2] qui détectent alors des artefacts de manipulation dans l'image. Mais l'entraînement de ces réseaux s'avère relativement coûteux en ressources de calcul et en temps d'entraînement. Pour palier ce problème, nous étudions ici une méthode de classification d'images de vidéos basée dans l'espace latent d'un modèle génératif (plus précisément StyleGAN [7]) où les données sont représentées en plus faible dimension.

Nous proposons de comparer différentes méthodes d'inversion d'image parmi les plus récentes de l'état de l'art dans un contexte de détection d'hypertrucages. Pour cela, nous inversons la base DeepFakeDetectionChallenge preview (DFDC) [5] dans l'espace latent de StyleGAN selon différents procédés. Nous entraînons ensuite des modèles de classification binaire sur les codes obtenus pour distinguer ceux provenant des images authentiques des autres.

Les CNN obtiennent maintenant de bons résultats dans la détection d'hypertrucages, lorsque les bases sont connues à l'avance et que l'on dispose d'une grande quantité de données d'entraînement [10]. Le réseau MésoNet par exemple [2], est conçu de manière à repérer des défauts de l'image à l'échelle mésoscopique : à mi-chemin entre le pixel et l'information de haut niveau, ce qui lui permet d'atteindre de bonnes performances tout en étant relativement léger. Cependant, cela signifie qu'avec l'amélioration des processus de contrefaçon, le coût de la mise en place de tels détecteurs augmente. Les méthodes de détection ont un temps de retard sur les nouvelles méthodes de falsification : il est nécessaire pour chaque nouvelle falsification de constituer une base de données adaptée suffisamment diverse et importante. C'est précisément l'un des désavantages de recourir à des réseaux s'appliquant directement au niveau pixel : la nécessité d'avoir accès à un grand nombre d'exemples d'entraînement. Pour modéliser une frontière dans l'espace des images, le fléau de la dimension implique la nécessité de disposer d'un très grand nombre de points de mesure. Nous proposons donc de travailler sur un espace à plus faible dimension pour entraîner les modèles de détection. De plus, nous escomptons qu'utiliser un tel espace de représentation permettra d'ignorer des détails dans l'espace image qui seraient superflus ou pourraient tromper le modèle, et garder les détails de plus haute importance (conserver l'information sur l'expression du sujet plutôt que la couleur du ciel par exemple).

2 Méthode

Nous proposons une comparaison de différentes méthodes d'inversion d'images extraits d'hypertrucages dans l'espace latent de StyleGAN2, dans le but de mettre en place de bout en bout un processus de détection d'hypertrucages. Après avoir inversé une image suspecte dans l'espace latent de StyleGAN, nous entraînons un modèle de détection à discerner les codes latents provenant d'images trafiquées des authentiques. Cette section décrit d'abord l'architecture de StyleGAN, avant de poursuivre sur les différentes étapes du processus : l'inversion des images en général et les méthodes utilisées en particulier puis l'entraînement des modèles de détections sur des codes latents issus de l'inversion de la base DFDC.

2.1 StyleGAN

Le réseau génératif StyleGAN [7] a l'avantage, en plus de produire des images réalistes à haute résolution, de travailler avec un espace latent intermédiaire. Plutôt que de générer directement une image à partir d'un échantillon d'une variable aléatoire vectorielle, cet échantillon est d'abord transformé en un code latent, ou vecteur de style, qui sera ensuite réinjecté à différentes étapes du processus de génération. L'architecture du réseau générateur StyleGAN peut être résumée ainsi, avec z un vecteur gaussien, x une image, et w le vecteur de style correspondant :

$$\begin{aligned} &z, w \in \mathbb{R}^{512}, x \in \mathbb{R}^{1024\times1024\times3} \\ &M : z \to w, M(z) = w \\ &S : w \to x, S(w) = x \\ &G : z \to x, G(z) = S \circ M(z) \end{aligned} \tag{1}$$

M, S sont respectivement un réseau de neurones de transformation de la variable aléatoire z en le vecteur de style w, et un réseau de synthèse, produisant l'image finale à partir de w. L'espace image du réseau M, noté W a été étudié dans la littérature comme variété satisfaisante pour l'espace des images de visages [7] [14]. Des directions dans cet espaces ont également été identifiées pour modifier artificiellement des images réelles en les *éditant* [11] [6] (par exemple agir sur l'âge ou l'émotion du sujet représenté). De plus, des études précédentes ont aussi montré l'avantage d'utiliser des espaces de faible dimension, notamment d'Auto-encodeurs variationnels [3] [8] dans la détection d'hypertrucages. L'espace latent de StyleGAN en particulier a été prouvé plus efficace dans ce contexte qu'une analyse en composantes principales[4]. Cependant, ces derniers travaux présentent l'inconvénient d'utiliser une méthode d'inversion relativement lente et ancienne, il est proposé ici de vérifier si des méthodes plus récentes peuvent améliorer les résultats ou la rapidité du processus.

2.2 Principe de l'inversion d'une image dans l'espace latent de StyleGAN

Mathématiquement, inverser une image dans l'espace latent de StyleGAN peut se traduire par résoudre l'équation 2 avec σ une mesure de similarité.

$$w^* = \underset{w}{\operatorname{argmin}}\, \sigma(x, S(w)) \tag{2}$$

Algorithme 1 : Optimisation dans l'espace latent de StyleGAN [7]

Données : $x \in \mathbb{R}^{1024\times1024\times3}, (\epsilon, \eta) \in \mathbb{R}^2$
$S : \mathbb{R}^{512} \to \mathbb{R}^{1024\times1024\times3}$
Résultat : $w^*, S(w^*) = \hat{x}$
$w \leftarrow w_0$;
tant que $\sigma(S(w), x) > \epsilon$;
faire
  $L = L(S(w), x)$;
  $w \leftarrow w - \eta\nabla L$
fin

Comme le réseau générateur de StyleGAN est hautement complexe, il n'est pas trivial de faire correspondre une image donnée à son code latent. Tout un pan de la littérature est consacré à cette tâche[14]. Les approches proposées pour l'accomplir se distinguent principalement sur deux points : le processus pour obtenir le code latent et dans quel espace exactement l'image est inversée.

La première catégorie de processus mise au point se base sur une approche par optimisation, où une première version du code latent est initialisée, puis améliorée par descente de gradient pour minimiser une ou des distances (souvent une norme euclidienne et une distance perceptive telle que LPIPS[15]) dans l'espace image. Un algorithme d'optimisation tel que celui proposé par *Image2StyleGAN*[1] pour obtenir le code latent w depuis une image réelle est décrit Algorithme 1. La notation avec accent circonflexe $\hat{x}$ représente une approximation de l'image x. Dans *Image2StyleGAN*, une somme pondérée d'une distance perceptive et de la norme euclidienne est utilisée pour la fonction de coût L .

Une alternative consiste à entraîner un réseau (typiquement un CNN) encodeur qui permet d'obtenir directement un code pour chaque image. La première méthode a l'avantage de ne pas nécessiter préalablement l'entraînement supervisé d'un réseau, mais souffre d'une lenteur d'exécution pour chaque inversion (typiquement de l'ordre de quelques minutes). La seconde, au contraire, permet d'obtenir des codes latents à une vitesse de plusieurs images par seconde, mais peut souffrir d'une moins bonne qualité de reconstruction sur l'image finale (l'optimisation peut être poussée jusqu'à obtenir une image quasiment identique). La Figure 1 résume les deux procédés principaux.

Des approches hybrides existent, où un encodeur effectue une première inversion, qui servira d'initialisation à un processus d'optimisation, elles permettent un compromis entre temps de calcul et qualité d'inversion.

D'un autre côté, des alternatives à l'espace latent W ont été proposées par la communauté. Dans l'article original de StyleGAN, les vecteurs $w \in W$ sont injectés à l'identique à 18 endroits dans le processus de génération. Des études, débutant avec Abdal *et al.* [1] ont montré qu'inverser le StyleGAN en 18 vecteurs différents (dans un espace appelé $W+$), selon où ils sont injectés, résulte en une reconstruction d'image plus fidèle, au détriment de la facilité d'édition des codes obtenus (il est instinctivement plus difficile de trouver des changements sémantiques significatifs dans l'espace image si l'espace latent est de dimension 512×18 plutôt que 512).

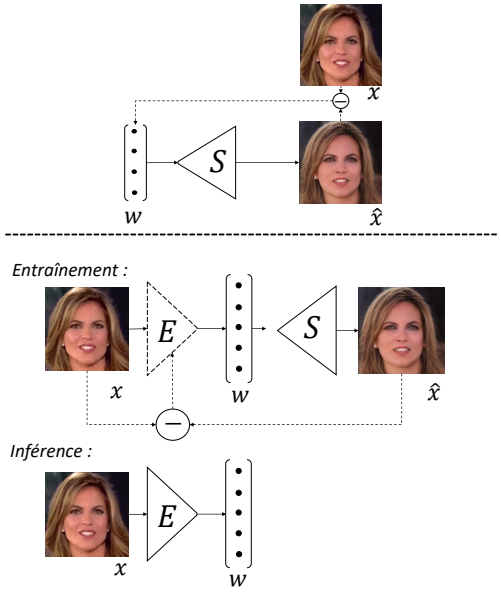

FIGURE 1 : Les deux méthodes principales de l'inversion de StyleGAN, les flèches en pointillés désignent respectivement le procédé d'optimisation et l'apprentissage de l'encodeur. Le réseau de synthèse de StyleGAN est représenté par le bloc S. En haut, la méthode par optimisation fournit un code w optimisé en minimisant une différence dans l'espace image jusqu'à obtention d'un résultat satisfaisant. En bas, un réseau encodeur E est entraîné à fournir directement à partir de l'image un code qui permet de reconstruire l'image. Le processus d'inférence est représenté simplement en dessous.

2.3 Méthodes d'inversions considérées

Nous comparons quatre approches différentes pour résoudre l'équation 2 et obtenir un code latent w depuis une image x :

- IDInvert [16] propose une méthode hybride avec un Encodeur E puis optimisation, où les auteurs profitent d'une sémantique apprise du domaine initial (*i.e.* les images de visages) pour améliorer la qualité d'inversion en assurant que les codes latents obtenus soient au plus proche possible dans la distribution habituelle de StyleGAN. Pour cela, ils minimisent (Équation 3) une somme pondérée distances dans le domaine image, dans un espace $W+$ à 14 canaux et dans les activations d'un réseau VGG $F(x)$. Ils résolvent le problème décrit en équation 3, avec $\lambda_{vgg}, \lambda_{dom}$ deux réels.

$$w^* = \underset{w}{\operatorname{argmin}}(||x - S(w)|| + \lambda_{vgg}||F(x) - F(S(w))||_2 + \lambda_{dom}||w - E(S(w))||_2) \tag{3}$$

- Psp [9] introduit 18 modules « map2style », notés β, indépendants pour convertir les représentations intermédiaires d'un CNN à connexions résiduelles préalablement entraîné (ResNet) R en un vecteur de style dans $W+$ utilisable par StyleGAN, de composantes $\forall i \in [\![1\ ;\ 18]\!], w_i = \beta_i(R(x))$.
- E2Style [13] propose d'abord d'estimer un premier code latent $w_0 = E_0(x)$, qui peut être séquentiellement raffiné en $w_t = E_t(x, S(w_{t-1})) + w_{t-1}$. Les encodeurs E_t sont entraînés séparément afin de profiter des avantages des approches hybrides encodeur-optimisation sans avoir à passer par un processus de descente de gradient en inférence.

Méthode	Entraînement	Validation	Test
A	33329	4717	9341
B	2076	303	685
Authentique	27276	3947	7895

TABLE 1 : Nombre et répartition des exemples pour chaque méthode de manipulation et pour les vidéos originales

- HFGI [12] est une des approches les plus récentes utilisées à l'heure actuelle. Après une estimation du code latent $w_0 = E_0(x), \hat{x}_0 = S(w_0)$ de l'image à inverser, un second encodeur E_c va transformer l'erreur de reconstruction $\Delta = x - \hat{x}_0$ en un deuxième code latent. Ces deux codes sont ensuite fusionnés par un module D qui les intègre à StyleGAN pour obtenir le code final $\hat{x} = S(D(w_0, E_c(\Delta)))$.

Toutes ces méthodes inversent les images dans l'espace $W+$ (ou une variante à 14 dimensions pour la méthode IDInvert). Pour la méthode E2Style, nous n'utilisons que le prmier encodeur E_0. Nous nous sommes concentrés sur des méthodes à base de réseau encodeur ou hybrides. En effet dans le contexte de détection d'hypertrucages, les méthodes à base d'optimisation prendraient trop de temps pour être utilisables, à la fois pour constituer une base d'entraînement, ou en inférence, pour prendre une décision sur une nouvelle vidéo.

2.4 Entraînement du modèle de détection

Nous considérons la base de données DFDC « preview »[5], qui comporte au total 5214 vidéos d'acteurs consentants, séparée en 1084 vidéos authentiques et leurs versions modifiées selon deux différentes méthodes de génération de deepfake, nommées A et B. Nous avons séparé 70% des vidéos en exemples d'entraînement, 20% de tests et 10% de validation. Pour chaque vidéo, une quarantaine d'images, échantillonnées à environ 3 images par seconde, ont été inversées pour travailler sur une quantité raisonnable d'images. Finalement, le nombre d'exemples utilisé est décrit dans la Table 1.

A partir des images extraites des vidéos, une extraction et un alignement des visages présents est effectuée selon le protocole FFHQ[7] avant d'appliquer les différentes méthodes d'inversion. Pour chaque base de code ainsi obtenue, nous entraînons un modèle d'apprentissage automatique relativement simple (cinq couche de neurones linéaires, activés par une fonction ReLu) à minimiser l'entropie croisée binaire pour détecter les codes latents provenant des images manipulées.

3 Résultats

Pour chaque méthode d'inversion, nous avons entraîné une instance du modèle neuronal à discerner les codes latents provenant d'images trafiquées ou authentiques. Les précisions obtenues par ces méthodes et leur rapidité sont présentées Table 2. A titre de comparaison, nous incluons les résultats de la discrimination obtenus par les auteurs de LatentForensics [4]. Une inversion préalable à la détection dans l'espace latent de StyleGAN permet bien d'obtenir des précisions de détection d'hypertrucages supérieures aux méthodes de l'état de l'art. L'inversion E2Style est la plus adaptée au problème pour cette base de données si la précision de détection est prioritaire.

Inversion	Précision (%)	Débit (s/image)	PSNR⋆
IDInvert [16]	94*	86.9	**22.3**
Psp [9]	96.31	0.0492	20.7
HFGI [12]	96.43	**0.0396**	-
E2Style [13]	**98.46**	0.0618	21.3

TABLE 2 : Part de codes d'image correctement classifiées, la meilleure méthode est en gras. Les mesures de vitesse d'inversion ont été réalisées sur une NVIDIA QUADRO RTX 3000
* Résultat tiré de l'étude LatentForensics [4]
⋆ Résultats tirés de l'étude E2Style [13]

Les méthodes Psp et HFGI permettent quant à elles d'obtenir des performances satisfaisantes plus rapidement. Même si la méthode hybride IDInvert atteint la meilleure reconstruction (PSNR), elle semble moins adaptée dans ce contexte. Les méthodes à base d'encodeurs (Psp, HFGI et E2Style) sont donc bien envisageables pour effectuer de la détection d'hypertrucages dans un contexte réel.

4 Conclusion et travaux futurs

Nous avons comparé différentes manières d'implémenter un processus de détection de deepfakes dans l'espace latent de StyleGAN, et montré comment ce paradigme permettait de mettre en place des modèles qui atteignent des fortes précisions de détection tout en gardant un temps de traitement raisonnable. Nous comptons poursuivre dans cette direction, il serait notamment intéressant d'étudier si passer par ces espaces de plus faible dimension permet aux modèles de mieux généraliser à de nouvelles méthodes de manipulation.

Références

[1] Rameen ABDAL, Yipeng QIN et Peter WONKA : Image2stylegan : How to embed images into the stylegan latent space ? *In Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019.

[2] Darius AFCHAR, Vincent NOZICK, Junichi YAMAGISHI et Isao ECHIZEN : Mesonet : a compact facial video forgery detection network. *In 2018 IEEE international workshop on information forensics and security (WIFS)*, pages 1–7. IEEE, 2018.

[3] Davide COZZOLINO, Justus THIES, Andreas RÖSSLER, Christian RIESS, Matthias NIESSNER et Luisa VERDOLIVA : Forensictransfer : Weakly-supervised domain adaptation for forgery detection. *arXiv preprint arXiv :1812.02510*, 2018.

[4] Matthieu DELMAS, Amine KACETE, Stephane PAQUELET, Simon LEGLAIVE et Renaud SEGUIER : Latentforensics : Towards lighter deepfake detection in the stylegan latent space, 2023.

[5] Brian DOLHANSKY, Russ HOWES, Ben PFLAUM, Nicole BARAM et Cristian Canton FERRER : The deepfake detection challenge (dfdc) preview dataset. *arXiv preprint arXiv :1910.08854*, 2019.

[6] Erik HÄRKÖNEN, Aaron HERTZMANN, Jaakko LEHTINEN et Sylvain PARIS : Ganspace : Discovering interpretable gan controls. *Advances in Neural Information Processing Systems*, 33:9841–9850, 2020.

[7] Tero KARRAS, Samuli LAINE et Timo AILA : A style-based generator architecture for generative adversarial networks. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[8] Hasam KHALID et Simon S WOO : Oc-fakedect : Classifying deepfakes using one-class variational autoencoder. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 656–657, 2020.

[9] Elad RICHARDSON, Yuval ALALUF, Or PATASHNIK, Yotam NITZAN, Yaniv AZAR, Stav SHAPIRO et Daniel COHEN-OR : Encoding in style : a stylegan encoder for image-to-image translation. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021.

[10] Andreas ROSSLER, Davide COZZOLINO, Luisa VERDOLIVA, Christian RIESS, Justus THIES et Matthias NIESSNER : Faceforensics++ : Learning to detect manipulated facial images. *In Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1–11, 2019.

[11] Yujun SHEN, Jinjin GU, Xiaoou TANG et Bolei ZHOU : Interpreting the latent space of gans for semantic face editing. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9243–9252, 2020.

[12] Tengfei WANG, Yong ZHANG, Yanbo FAN, Jue WANG et Qifeng CHEN : High-fidelity gan inversion for image attribute editing. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11379–11388, 2022.

[13] Tianyi WEI, Dongdong CHEN, Wenbo ZHOU, Jing LIAO, Weiming ZHANG, Lu YUAN, Gang HUA et Nenghai YU : E2style : Improve the efficiency and effectiveness of stylegan inversion. *IEEE Transactions on Image Processing*, 31:3267–3280, 2022.

[14] Weihao XIA, Yulun ZHANG, Yujiu YANG, Jing-Hao XUE, Bolei ZHOU et Ming-Hsuan YANG : Gan inversion : A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[15] Richard ZHANG, Phillip ISOLA, Alexei A EFROS, Eli SHECHTMAN et Oliver WANG : The unreasonable effectiveness of deep features as a perceptual metric. *In CVPR*, 2018.

[16] Jiapeng ZHU, Yujun SHEN, Deli ZHAO et Bolei ZHOU : In-domain gan inversion for real image editing. *In European conference on computer vision*, pages 592–608. Springer, 2020.