Tatouage numérique de modèles de diffusion latente

Pierre FERNANDEZ[1,2], Guillaume COUAIRON[1,3], Hervé JÉGOU[1], Matthijs DOUZE[1], Teddy FURON[2]

[1]Meta AI, FAIR [2]Centre Inria de l'Université de Rennes [3]Sorbonne Université

Correspondance: Pierre Fernandez <pfz@meta.com>

Résumé – Cet article présente une stratégie de tatouage pour les modèles de diffusion latente. Le but est de cacher une marque invisible dans les images dès leur création pour faciliter leur détection. Notre approche ajuste finement le générateur afin que les images générées aient une signature binaire invisible, détectable par un un réseau extracteur de tatouage auto-supervisé. Un test statistique détermine si l'image a été produite par le modèle génératif. Ceci permet, par exemple, de détecter avec un rappel de 90% et un taux de faux positifs inférieur à 10^{-6}, l'origine d'une image générée puis recadrée pour ne conserver que 10% du contenu.

Abstract – This paper introduces a watermarking strategy for latent diffusion models. The goal is for all generated images to conceal an invisible watermark allowing for future detection. Our approach fine-tunes the generator so that the generated images have a specific binary signature, detectable by a self-supervised watermarking extractor. A statistical test determines if the image has been produced by the generative model. This allows, for example, to detect with a recall of 90% and a false positive rate lower than 10^{-6}, the origin of an image generated and then cropped to keep only 10% of the content.

1 Introduction

Les progrès récents en génération d'images et en traitement du langage naturel ont facilité la création et la manipulation d'images de manière photoréaliste. Par exemple, DALL·E 2 [1] ou Stable Diffusion [2] génèrent des images qui sont parfois indiscernables de véritables œuvres d'art, et s'imposent comme des outils de création et d'édition pour les artistes et le grand public.

Ceci est un grand pas en avant pour l'IA générative, et soulève de nouvelles questions éthiques. En effet, la sophistication des modèles est telle qu'il sera bientôt impossible de distinguer leurs produits d'images réelles, rendant difficile leur retrait de certaines plateformes et leur mise en conformité avec des normes éthiques. L'absence de traçabilité ouvre la porte à de nouvelles menaces telles que les *deep fakes* et l'usurpation d'identité ou de droits d'auteur [3]. Les méthodes de détection passives n'ont pas aujourd'hui une performance suffisante contre ces menaces. Par exemple, les meilleurs algorithmes de détection présentés par Corvi *et al* [4] ne dépassent pas 50% de rappel pour un taux de faux positifs à 10^{-3}.

Des méthodes de tatouage numérique peuvent être appliquées après à la génération d'images. L'idée est d'intégrer un message dans l'image de manière invisible, pour ensuite être extrait et servir l'identification. Toutefois, si le modèle fuit ou est diffusé, le tatouage post-génération est facile à supprimer. Stable Diffusion en est un exemple, et commenter une seule ligne du code source suffit à supprimer le tatouage. Notre méthode, *Stable Signature*, incorpore le tatouage dans la génération elle-même, sans aucune modification de l'architecture du modèle génératif. Ceci permet de tatouer toutes les images générées, même si le modèle est libre et rend le procédé immédiat.

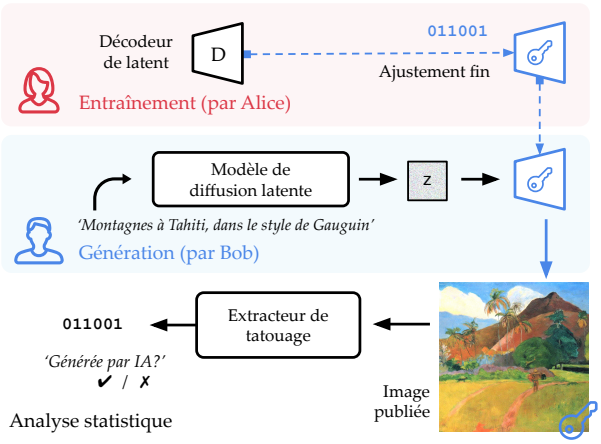

FIGURE 1 – Le décodeur de latents peut être ajusté pour intégrer préemptivement une signature dans toutes les générations.

Nous nous concentrons sur les modèles de diffusion latente (LDM) [2] qui sont ceux employés par Stable Diffusion. Dans les LDM, le processus de diffusion intervient dans l'espace latent d'un auto-encodeur variationnel : la diffusion du vecteur latent est guidée par un texte (ou d'autres entrées comme une carte sémantique) et les images sont générées par un décodeur après quelques itérations. Stable Signature opère en ré-entraînant ce décodeur par rétropropagation, en minimisant deux fonctions de perte : une d'image perceptuelle et une de tatouage. Celle associée au tatouage est obtenue en extrayant le message d'images générées et en comparant le résultat avec une signature binaire fixée au préalable. L'extracteur de tatouage neuronal est appris avec une version adaptée de la méthode HiDDeN [5]. Raffiner ce décodeur ne modifie pas le processus de diffusion et est compatible avec de nombreuses méthodes génératives.

Nous évaluons Stable Signature sur différentes tâches génératives en situations réelles, où les images peuvent être éditées. Nous détectons par exemple 90% des images générées avec le modèle génératif, même si elles sont recadrées à 10% de leur taille originelle, tout en ne signalant qu'un faux positif toutes les 10^6 images. Nous nous assurons aussi que la génération n'est pas affectée et que les images générées sont perceptiblement indiscernables de celles produites par le modèle originel.

2 Méthode

Tout d'abord, nous créons un extracteur de tatouage $\mathcal{W}$. Ensuite, nous affinons le décodeur $\mathcal{D}$ du LDM, de sorte que les images générées aient une signature donnée à travers $\mathcal{W}$ (la signature est un message de k-bits).

Pré-entraînement de l'extracteur. Nous utilisons HiDDeN [5], une méthode qui optimise conjointement les paramètres d'un réseau encodeur $\mathcal{W}_E$ et d'un réseau extracteur $\mathcal{W}$ pour intégrer des messages de k bits dans les images, de manière robuste aux transformations appliquées pendant l'entraînement.

Formellement, $\mathcal{W}_E$ prend en entrée une image $x_o \in \mathbb{R}^{W\times H\times 3}$ et un message m tiré uniformément sur $\{0,1\}^k$. $\mathcal{W}_E$ produit une image résiduelle δ de la même taille que x_o, multipliée par un facteur α pour produire une image tatouée $x_w = x_o + \alpha\delta$. À chaque étape d'optimisation, une transformation d'image T est échantillonnée à partir d'un ensemble $\mathcal{T}$ qui comprend des opérations courantes de traitement d'image telles que le recadrage et la compression JPEG [1]. Un message à décision douce est extrait de l'image transformée : $m' = \mathcal{W}(T(x_w))$ (au moment de l'inférence, le message décodé est donné par les signes des composantes de m'). La *perte de tatouage* est l'entropie croisée binaire (BCE) entre m et la sigmoïde $\sigma(m')$:

$$\mathcal{L}_m = -\sum_{i=1}^{k} m_i \cdot \log \sigma(m'_i) + (1-m_i)\cdot \log(1-\sigma(m'_i)).$$

Les architectures de réseau restent simples pour faciliter l'ajustement fin du décodeur dans la deuxième phase, et sont identiques à HiDDeN. Toutefois, puisque $\mathcal{W}_E$ n'est pas utile dans l'étape qui suit, sa qualité perceptuelle n'est pas aussi importante. La distorsion n'est limitée que par une fonction $\tanh$ sur la sortie de $\mathcal{W}_E$ et par le facteur d'échelle α (contrairement à HiDDeN qui utilise un réseau antagoniste). De plus, nous observons que les bits de sortie de $\mathcal{W}$ pour les images naturelles sont corrélés et fortement biaisés, ce qui viole les hypothèses de la Sec. 4. Par conséquent, nous appliquons une transformation de blanchiment par ACP à la fin de l'entraînement.

1. La transformation doit être différentiable dans l'espace des pixels. Pour la compression JPEG, la rétropropagation est faite à travers l'identité à la place [6].

Ajustement fin du modèle génératif. Tout d'abord, la signature $m = (m_1, \ldots, m_k) \in \{0,1\}^k$ est fixée. L'ajustement fin de $\mathcal{D}$ en $\mathcal{D}_m$ s'inspire de l'entraînement original de l'auto-encodeur dans LDM [2].

Une image $x \in \mathbb{R}^{H\times W\times 3}$ est introduite dans l'encodeur du LDM $\mathcal{E}$ et produit une carte d'activation $z = \mathcal{E}(x) \in \mathbb{R}^{h\times w\times c}$, sous-échantillonnée par un facteur de puissance de deux $f = H/h = W/w$. Le décodeur reconstruit une image $x' = \mathcal{D}_m(z)$ et l'extracteur récupère $m' = \mathcal{W}(x')$. Comme pour le pré-entraînement, la *perte de message* est la BCE entre m' et le m originel : $\mathcal{L}_m = \mathrm{BCE}(\sigma(m'), m)$.

En outre, le décodeur originel $\mathcal{D}$ reconstruit l'image sans tatouage : $x'_o = \mathcal{D}(z)$. La *perte d'image* $\mathcal{L}_\mathrm{i}$ entre x' et x'_o contrôle la distorsion ajoutée par le tatouage. Nous utilisons la perte perceptuelle Watson-VGG introduite par Czolbe *et al* [7]. Elle modélise la perception humaine et tient compte du masquage de luminance et de contraste.

Les poids de $\mathcal{D}_m$ sont optimisés en quelques étapes de rétropropagation afin de minimiser :

$$\mathcal{L} = \mathcal{L}_\mathrm{m} + \lambda_\mathrm{i}\,\mathcal{L}_\mathrm{i}. \tag{1}$$

Ceci est effectué sur 100 itérations avec l'optimiseur AdamW [8] et un lot de taille 4. Cet ajustement fin ne voit *que 400 images* et ne prend *qu'une minute sur un GPU*.

3 Expériences & Résultats

Tâches et métriques d'évaluation. Notre méthode n'implique que le décodeur du LDM et est donc compatible avec de nombreuses tâches génératives. Nous évaluons la génération texte-image et l'édition sur l'ensemble de validation de COCO et la super-résolution sur ImageNet. Dans toutes ces évaluations, les images générées ont une résolution de 512×512 pixels, et la signature binaire est de taille $k = 48$ bits.

Nous évaluons la dégradation de l'image à l'aide du rapport signal/bruit maximal (PSNR) : $\mathrm{PSNR}(x,x') = -10\cdot\log_{10}(\mathrm{MSE}(x,x'))$, pour $x, x' \in [0,1]^{c\times h\times w}$, ainsi que le score de similarité structurelle (SSIM) [9]. Ils comparent les images générées avec et sans tatouage. D'autre part, nous évaluons la diversité et la qualité des images générées avec la Fréchet Inception Distance (FID) [10]. Le taux d'erreur par bits (BER), *i.e* pourcentage de bits correctement décodés, évalue la robustesse du tatouage aux transformations suivantes : recadrage important (10% de l'image restante), changement de luminosité (facteur 2), combinaison d'un recadrage 50%, d'un changement de luminosité 1.5 et d'une compression JPEG 80.

Qualité de la génération d'images. La Figure 2 montre la manière dont la génération est affectée par la modification du décodeur. La différence entre images est difficile à percevoir. Ceci est surprenant pour un PSNR aussi faible, d'autant plus que l'incorporation du tatouage n'est pas limitée par un système visuel humain comme dans les

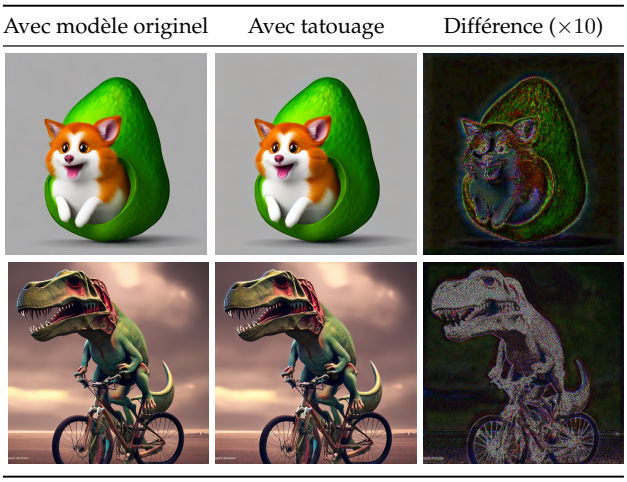

FIGURE 2 – Images générées à partir de texte par les modèles génératifs avec ou sans tatouage. Le PSNR est de 35.4 dB dans la première ligne et de 28.6 dB dans la seconde. Les images générées ont un aspect naturel car les zones modifiées sont situées là où l'œil n'est pas sensible.

techniques de tatouage professionnelles. Le décodeur du LDM a en effet appris à ajouter le signal du tatouage uniquement sur les zones texturées auxquelles l'œil humain n'est pas sensible, tandis que les arrière-plans uniformes restent intacts.

La Table 1 présente une évaluation quantitative sur 5k images générées pour les différentes tâches. Nous indiquons la FID, ainsi que le PSNR et le SSIM moyens qui sont calculés entre les images générées par les décodeurs tatoué et originel. Les résultats montrent que, quelle que soit la tâche, le tatouage a un très faible impact sur la FID de la génération. Le PSNR moyen est d'environ 30 dB et le SSIM d'environ 0.9. Ces valeurs sont un peu faibles du point de vue tatouage car nous ne les optimisons pas explicitement. En effet, dans un scénario réel, on ne disposerait que de la version tatouée de l'image. Par conséquent, nous n'avons pas besoin d'être aussi proches que possible de l'image originale, mais voulons seulement générer des images sans artefacts, et qu'il soit difficile de savoir si un tatouage est présent ou non.

Robustesse du tatouage. Nous évaluons la robustesse du tatouage à différentes transformations. Pour chaque tâche, nous générons 10×1k images avec 10 modèles marqués avec différents messages binaires, et nous indiquons le BER moyen en Table 1.

Nous constatons que le tatouage est robuste pour plusieurs tâches et transformations, et le BER du décodage est toujours supérieure à 0.9. En outre, le décodage n'est pas parfait même sans édition, principalement car certaines images sont plus difficiles à tatouer (par exemple celles qui sont très uniformes, comme en arrière-plan de Fig. 2) et pour lesquelles les BER sont plus faibles.

Il convient de noter que la robustesse ne provient que de l'extracteur de tatouage, puisqu'aucune transformation n'est vue au cours de la phase de d'ajustement fin. Si le pipeline d'intégration du tatouage est appris pour être robuste vis-à-vis d'une augmentation (grâce à $\mathcal{T}$), le LDM apprendra à produire des tatouages qui sont robustes vis-à-vis de cette augmentation lors de l'ajustement fin.

Comparaison avec le tatouage post-hoc. Une autre façon de tatouer les images générées consiste à les traiter après la génération (post-hoc). Cette méthode peut être plus simple, mais moins sûre et moins efficace. Nous comparons notre méthode à DCT-DWT [11] qui encode l'information dans les fréquences de l'image (utilisée dans le code de Stable Diffusion), une approche itérative [12], et un encodeur/extracteur comme HiDDeN [5]. Nous utilisons notre implémentation pour les deux dernières méthodes, et en particulier nous utilisons un masquage perceptuel [13] pour s'assurer de leur qualité visuelle.

La Table 1 compare la qualité de génération et la robustesse sur 5k images générées. Dans l'ensemble, Stable Signature obtient des résultats comparables en terme de robustesse. Les performances de HiDDeN sont un peu plus élevées, mais ses bits de sortie ne sont pas i.i.d., ce qui signifie qu'il ne peut pas être utilisé avec les mêmes garanties. Nous observons également que la génération post-hoc donne de moins bons résultats qualitatifs et les images ont tendance à présenter des artefacts. Cela peut s'expliquer par le fait que Stable Signature est intégrée dans le processus de génération de haute qualité avec de l'auto-encodeur du LDM, qui est capable de modifier les images de manière plus subtile.

4 Cas Pratique : Détection

Alice partage un modèle génératif avec Bob, qui souhaite générer des images. Elle intègre une signature binaire de k bits avec Stable Signature. L'extracteur de tatouage décode ensuite les messages des images qu'il reçoit et signale si le message est proche de la signature d'Alice. Une application est de bloquer les images générées par IA sur une plateforme de partage de contenu.

Test de détection statistique. Soit $m \in \{0,1\}^k$ la signature d'Alice. Nous extrayons le message m' d'une image x et le comparons à m. Comme dans des travaux précédents [16], le test de détection repose sur le nombre de bits correspondants $M(m,m')$: si

$$M(m,m') \geq \tau \quad \text{où} \quad -\tau \in \{0,\dots,k\}, \tag{2}$$

l'image est signalée. Cela tient compte du fait que l'extracteur de tatouage n'est pas parfait et donne un niveau de confiance au signalement.

Formellement, nous testons l'hypothèse statistique H_1 : "x a été généré par le modèle d'Alice" contre l'hypothèse nulle H_0 : "x n'a pas été généré par le modèle d'Alice". Sous H_0 (c'est-à-dire pour les images naturelles), nous supposons que les bits $m'_1,\dots,m'_k$ sont i.i.d. et suivent une loi de Bernoulli de paramètre $0,5$. $M(m,m')$ suit alors

			PSNR / SSIM ↑	FID ↓	BER ↑ sur : Iden.	Reca.	Lumi.	Comb.
Méthodes	Dct-Dwt [11]	0.14 (s/img)	39.5 / 0.97	19.5 (−0.4)	0.86	0.52	0.51	0.51
	SSL Watermark [12]	0.45 (s/img)	31.1 / 0.86	20.6 (+0.7)	1.00	0.73	0.93	0.66
	HiDDeN [5]	0.11 (s/img)	32.0 / 0.88	19.7 (−0.2)	0.99	0.97	0.99	0.98
	Stable Signature	-	30.0 / 0.89	19.6 (−0.3)	0.99	0.95	0.97	0.92
Tâches	Édition	DiffEdit [14]	31.2 / 0.92	15.0 (−0.3)	0.99	0.95	0.98	0.94
	Inpainting	Glide [15]	31.1 / 0.91	16.8 (+0.6)	0.99	0.97	0.98	0.93
	Super-Résolution	LDM [2]	34.0 / 0.94	11.6 (+0.0)	0.98	0.93	0.96	0.92

Table 1 – Qualité de la génération et comparaison avec le tatouage post-hoc sur des images 512×512 et des signatures 48-bit. Le PSNR et le SSIM sont calculés entre les images des générateurs originel et tatoué. Pour la FID, nous montrons en (orange) la différence par rapport à l'originel. Le tatouage post-hoc est évalué sur des images générées pour la tâche texte-image.

une distribution binomiale de paramètres $(k, 0.5)$. Sous H_1, $M(m, m')$ est susceptible d'être plus grand que $0.5k$. Par conséquent, si $M(m, m')$ est grand, la p-valeur (probabilité d'obtenir une valeur plus grande sous H_0) est faible et nous concluons que le modèle d'Alice a généré l'image ; une image x est donc signalée si $M(m, m') > \tau$. Le taux de faux positifs (TFP) est la probabilité que $M(m, m')$ prenne une valeur supérieure au seuil τ. Il s'obtient à partir de la fonction de répartition de la distribution binomiale, et une écriture peut être obtenue avec la fonction Bêta incomplète régularisée $I_x(a; b)$:

$$\begin{aligned} \text{TFP}(\tau) &= \mathbb{P}\left(M(m, m') > \tau | H_0\right) = \frac{1}{2^k} \sum_{i=\tau+1}^{k} \binom{k}{i} \\ &= I_{1/2}(\tau + 1, k - \tau). \end{aligned} \tag{3}$$

Résultats. Nous intégrons dans le décodeur du LDM une signature aléatoire m, générons 1k images et utilisons le test (2). Nous nous intéressons au compromis entre TFP et taux de vrais positifs (TVP), *i.e* la probabilité de détecter une image générée, tout en faisant varier $\tau \in \{0, ..., 48\}$. Par exemple, pour $\tau = 0$, nous signalons toutes les images donc TFP $= 1$, et TVP $= 1$. Le TFP est déduit de l'éq. (3), car il est trop petit pour être mesuré sur des problèmes de taille raisonnable (approximation validée expérimentalement). L'expérience est répétée 10 fois et nous présentons la moyenne des résultats.

La Figure 3 montre les courbes TVP/TFP pour différentes transformations. Lorsque les images générées ne sont pas modifiées, Stable Signature détecte 99% d'entre elles, quand 1 image naturelle sur 10^9 est signalée. Pour le même FPR $= 10^{-9}$, nous détectons 65% pour la combinaison de recadrage, luminosité et compression. À titre de comparaison, nous présentons également les résultats d'une méthode passive [4].

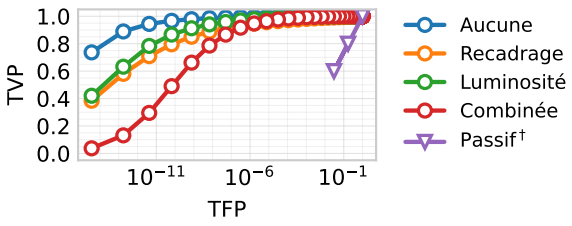

FIGURE 3 – **Résultats de la détection**. Courbe des taux Vrais Positifs / Faux Positifs de la détection sous différentes transformations. Passif† indique une détection passive (sans tatouage) [4].

5 Conclusion

Les modèles de génération d'images ont déjà un impact important sur la société. Avec ce travail, nous avons mis en lumière l'utilité d'utiliser du tatouage au lieu de s'appuyer sur des méthodes de détection passives. Nous espérons que cela encourage l'emploi d'approches similaires avant de rendre ces modèles accessibles au public.

Références

[1] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv :2204.06125*, 2022.

[2] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *CVPR*, 2022.

[3] E. Denton, "Ethical considerations of generative ai," in *AI for Content Creation Workshop CVPR*. IEEE, 2021.

[4] R. Corvi, D. Cozzolino, G. Zingarini, G. Poggi, K. Nagano, and L. Verdoliva, "On the detection of synthetic images generated by diffusion models," *arXiv preprint arXiv :2211.00680*, 2022.

[5] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "Hidden : Hiding data with deep networks," in *ECCV*, 2018.

[6] C. Zhang, A. Karjauv, P. Benz, and I. S. Kweon, "Towards robust deep hiding under non-differentiable distortions for practical blind watermarking," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021.

[7] S. Czolbe, O. Krause, I. Cox, and C. Igel, "A loss function for generative neural networks based on watson's perceptual model," *NeurIPS*, 2020.

[8] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*, 2018.

[9] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment : from error visibility to structural similarity," *IEEE transactions on image processing*, no. 4, 2004.

[10] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *NeurIPS*, 2017.

[11] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital watermarking and steganography*. Morgan kaufmann, 2007.

[12] P. Fernandez, A. Sablayrolles, T. Furon, H. Jégou, and M. Douze, "Watermarking images in self-supervised latent spaces," in *ICASSP*. IEEE, 2022.

[13] P. Fernandez, M. Douze, H. Jégou, and T. Furon, "Active image indexing," in *ICLR*, 2023.

[14] G. Couairon, J. Verbeek, H. Schwenk, and M. Cord, "Diffedit : Diffusion-based semantic image editing with mask guidance," *ICLR*, 2023.

[15] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "Glide : Towards photorealistic image generation and editing with text-guided diffusion models," *arXiv preprint arXiv :2112.10741*, 2021.

[16] D. Lin, B. Tondi, B. Li, and M. Barni, "Cycleganwm : A cyclegan watermarking method for ownership verification," *arXiv preprint arXiv :2211.13737*, 2022.