
Transfert de Style d'image basé sur des Réseaux de Neurones Convolutifs

Ariane Alix et Marie Heurtevent

Département Mathématiques

École Normale Supérieure Paris-Saclay

94230 Cachan, France

Abstract

Le transfert de style est un développement récent dans les applications des réseaux de neurones profonds en imagerie. Le facteur limitant des approches précédentes était l'impossibilité de séparer explicitement le contenu et le style d'une image. A l'aide de réseaux de neurones convolutifs optimisés pour la reconnaissance d'objets, il est finalement possible de distinguer l'information de haut niveau dans une image, le contenu, de l'information bas niveau de l'image c'est-à-dire le style.

Ce papier introduit la première implémentation du transfert de style neuronal, réalisée par Leon Gatys et al. en 2015 [1]. Leur algorithme permet ainsi de générer de nouvelles images combinant le contenu d'une image, et le style d'une autre, par exemple une photo et une peinture.

Introduction

Depuis les années 1990, les théories artistiques derrière les œuvres d'art attirent l'attention non seulement des artistes mais aussi de nombreux chercheurs en informatique et vision artificielle [4]. Depuis, de nombreuses techniques cherchant à transformer automatiquement une image naturelle en oeuvre d'art ont été développées. C'est aujourd'hui un domaine établi dans la communauté de l'infographie. Jusqu'à 2015 cependant les algorithmes de stylisation n'étaient conçus que pour des styles artistiques particuliers et se basaient uniquement sur des extraits d'image de bas niveau qui ne réussissaient pas toujours à capturer la structure réelle de l'image de manière performante.

L'idée principale derrière le transfert de style de Gatys est de conserver la structure de l'image utilisée comme contenu, tout en appliquant la texture (lignes, formes, couleurs) de l'image de style. Le but ici est donc de transférer la texture d'une image vers une autre, en contrignant les déformations de façon à conserver le contenu sémantique de l'image d'origine (c'est-à-dire sans déformer les objets).

Séparer le contenu du style dans les photos est encore un problème difficile. Cependant, suite aux avancées récentes des réseaux de neurones convolutifs profonds pour l'imagerie, il est maintenant possible d'extraire les informations sémantiques des images à différents niveaux. De nouvelles tâches de traitement de l'image sont aujourd'hui possibles, comme la détection et classification de texture et de style. Leon Gatys utilise ainsi ces réseaux convolutifs profonds pour pouvoir traiter séparément le style artistique et le contenu d'une image. Cela lui a permis de créer une nouvelle méthode de transfert de style, basée sur la synthèse de texture classique mais ajoutant des contraintes sur les représentations caractéristiques extraites des réseaux de neurones convolutifs.

Un point intéressant de son modèle est que le contenu comme la texture d'une image peuvent être extraits d'un même réseau convolutif. Il suffit donc d'un seul réseau et d'une optimisation sur ses paramètres pour effectuer le transfert de style.

1 État de l'art

1.1 Représentation de contenu et style avant Gatys

Pour résoudre les problèmes et limitations du transfert de texture présentés au-dessus, il est nécessaire de trouver une façon de modéliser indépendamment le contenu sémantique d'une image et le style dans lequel on l'observe.

En 2000, J. B. Tenenbaum and W. T. Freeman réussissent à séparer le style du contenu dans des images de certains visages dont ils ont fait varier l'éclairage (qui est donc le style) [5]. En 2014, D. P. Kingma et al. séparent le contenu du style des caractères manuscrits pour pouvoir mieux les identifier [6]. Cependant, les applications restent simples et ne s'étendent pas à tout type d'images.



Handwriting styles for MNIST obtained by fixing the class label and varying the 2D latent variable

Figure 1: Extrait du travail de D. P. Kingma sur le style des chiffres manuscrits

1.2 Synthèse de texture

Il existe deux types d'approches principales pour la synthèse de texture: les méthodes non-paramétriques et paramétriques. Les premières se basent sur un ré-échantillonnage de pixels ou de patchs de l'image de texture et peuvent produire rapidement des textures de bonne qualité (voir [7, 8, 9]). Cependant, elles ne permettent pas de définir un modèle explicite des textures. Les modèles paramétriques en revanche s'attachent à définir un tel modèle en observant différentes mesures statistiques de l'image. Une texture est alors définie de manière unique par les résultats de ces mesures et toute image qui produit les mêmes résultats devrait être perçue comme la même texture.

Leon Gatys publie en 2015 un papier pour synthétiser les textures se basant sur des réseaux de neurones [3]. Sa méthode est paramétrique: les mesures de l'image utilisées sont les réponses de l'image aux différents filtres des couches du réseau. Il est alors possible de synthétiser une texture similaire à celle de l'image en entrée en modifiant une image de bruit blanc aléatoire pour que ses réponses aux différents filtres du réseau soient similaires à celles de l'image d'origine. La Figure 3 illustre ce principe.

1.3 Inversion de représentation d'image

L'idée principale est de reconstruire une image à partir de sa représentation encodée telle que les résultats en sortie des filtres d'un réseau de neurones (features). Gatys se base sur une méthode publiée en 2014 par Aravindh Mahendran et Andrea Vedaldi ([11]), qui permet de reconstruire des images correspondant à chaque couche d'un CNN.

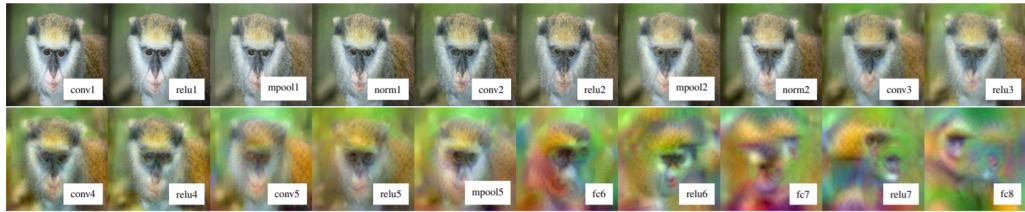


Figure 2: Reconstruction d'image depuis les réponses à chaque couche d'un CNN (voir [11])

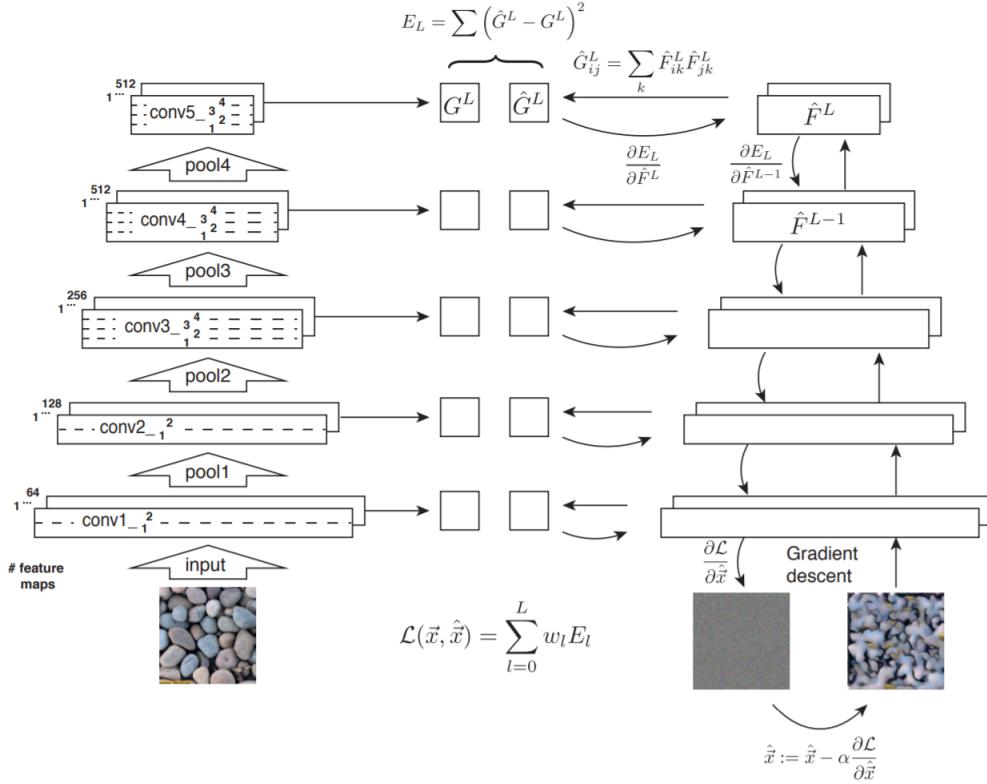


Figure 1: Synthesis method. Texture analysis (left). The original texture is passed through the CNN and the Gram matrices G_l on the feature responses of a number of layers are computed. Texture synthesis (right). A white noise image \hat{x} is passed through the CNN and a loss function E_l is computed on every layer included in the texture model. The total loss function \mathcal{L} is a weighted sum of the contributions E_l from each layer. Using gradient descent on the total loss with respect to the pixel values, a new image is found that produces the same Gram matrices \hat{G}_l as the original texture.

Figure 3: Génération de texture par Leon Gatys (voir [3])

1.4 Transfert de style par réseaux de neurones

La performance et supériorité des réseaux de neurones pour la classification des images ayant été démontrée depuis 2012 avec ImageNet [2], Leon Gatys, Alexander Ecker et Matthias Bethge eurent l'idée en 2015 de les utiliser pour générer des images artistiques. Jusqu'à cette date, les réseaux de neurones n'avaient jamais été utilisés pour générer des images en prenant en compte le style et l'art.

Concrètement, Gatys combine sa génération de texture neuronale [3] avec une méthode d'inversion de représentation de l'image [11], pour générer une nouvelle image de même texture que l'image de style et de même contenu sémantique que l'image de contenu.

2 Représentation du style et du contenu d'une image grâce aux réseaux profonds

2.1 Représentations dans les couches du réseau

Lorsque l'on entraîne des réseaux neuronaux convolutifs pour la reconnaissance visuelle d'objets, ils développent une représentation hiérarchisée de l'image, où le contenu est décrit de plus en plus précisément lorsque l'on s'enfonce dans les couches. Concrètement, l'image en entrée est transformée via de nombreux filtres (et peut être approximativement reconstituée) qui sont de plus en plus sensibles

au contenu réel de l'image mais moins au style: on obtient une représentation de ces caractéristiques mais pas de leur apparence précise.

Ainsi, les premières couches du réseau de neurones sont proches de l'image d'origine et contiennent les pixels exacts de l'image d'entrée; tandis que les couches plus profondes renferment les caractéristiques du contenu: la structure des objets, leur arrangement etc.

2.2 Fonctions de perte de contenu

Lorsque Gatys cherche à reconstruire le contenu d'une image depuis sa représentation neuronale avec la méthode de [11], il définit une fonction de perte permettant d'évaluer à quel point le contenu de l'image générée est proche de celui de l'image d'origine. Pour cela, il étudie la différence de réponse de l'image d'origine et de l'image générée à chaque couche l du réseau.

Pour l une couche du réseau, c l'image de contenu d'origine avec $(C_{ij}^l)_{ij}$ sa représentation à la couche l , et x avec $(X_{ij}^l)_{ij}$ l'image générée, la perte entre les deux images à la couche l est donnée par:

$$\mathcal{L}_{\text{contenu}}(c, x, l) = \frac{1}{2} \sum_{ij} (C_{ij}^l - X_{ij}^l)^2$$

2.3 Fonctions de perte de style

Un espace de fonctions est défini pour expliciter les informations de texture, comme dans [3]. Cet espace est constitué des corrélations entre les différentes réponses des filtres sur toute l'étendue spatiale des réponses caractéristiques (feature maps) à chaque couche du réseau. Concrètement, on définit des matrices représentant l'information de texture pour une image x à chaque couche l , que l'on calcule avec une matrice de Gram:

$$G_{x,ij}^l = \sum_k X_{ik}^l X_{kj}^l$$

On peut finalement définir la fonction de perte de style entre l'image de style s et l'image générée x au niveau de la couche l (qui est de dimension $N_l \times M_l$):

$$\mathcal{L}_{\text{style}}(s, x, l) = \frac{1}{4N_l^2 M_l^2} \sum_{ij} (G_{s,ij}^l - G_{x,ij}^l)^2$$

3 Principe de l'algorithme

L'algorithme final vise à générer une nouvelle image x dont le contenu est proche de l'image de contenu c , et le style proche de l'image de style s . Cela se conceptualise par la minimisation d'une fonction de perte totale:

$$\mathcal{L}_{\text{total}}(x, c, s) = \alpha \sum_l w_l^c \mathcal{L}_{\text{contenu}}(c, x, l) + \beta \sum_l w_l^s \mathcal{L}_{\text{style}}(s, x, l)$$

Où les paramètres α , β , $(w_l^c)_l$ et $(w_l^s)_l$ sont à choisir. Les $(w_l^c)_l$ déterminent à quelles couches du réseau on accorde le plus d'importance dans la génération de la nouvelle image par rapport au respect du contenu, $(w_l^s)_l$ par rapport au respect du style. α , β sont des facteurs de poids tels que pour un grand α par rapport à β , l'image générée sera proche de l'image de contenu avec peu de stylisation, et inversement. On fait face ici à un compromis entre représentation du contenu et du style, ce que nous illustrerons dans les résultats de la section suivante.

Dans son papier, Gatys effectue toutes ses générations d'images avec des poids de contenu de $\frac{1}{5}$ pour les couches "conv1_1", "conv2_1", "conv3_1", "conv4_1" et "conv5_1", et un poids de style de 1 pour la couche "conv4_2". L'optimisation se fait ensuite via un algorithme de LBFGS.

4 Résultats

4.1 Compromis entre contenu et style

Nous avons testé l'algorithme avec les mêmes paramètres de poids que Gatys (voir au-dessus), mais en faisant varier les paramètres α et β de la fonction de perte totale. On remarque que l'image en haut à droite est la plus similaire à l'image de contenu d'origine, puisque la perte de contenu est plus importante dans le calcul. A l'inverse, l'image en bas à droite est plus modifiée, et certains éléments de l'image se retrouvent ainsi déformés pour faire correspondre le style.



Figure 4: Exemple de résultats pour différents ratios de $\frac{\alpha}{\beta}$

4.2 Comparaison avec un algorithme de transfert de couleurs

Comme l'explique Gatys, le transfert de style peut être considéré comme un problème de transfert de texture dans lequel une contrainte sur le contenu sémantique de l'image cible est ajoutée. Ainsi, si le style de l'image se résume plus à sa palette de couleurs qu'à sa texture, une simple égalisation d'histogrammes, telle que vue en cours, devrait suffire. Ceci devrait donc être particulièrement vrai si nous faisons un transfert de style entre deux photos.

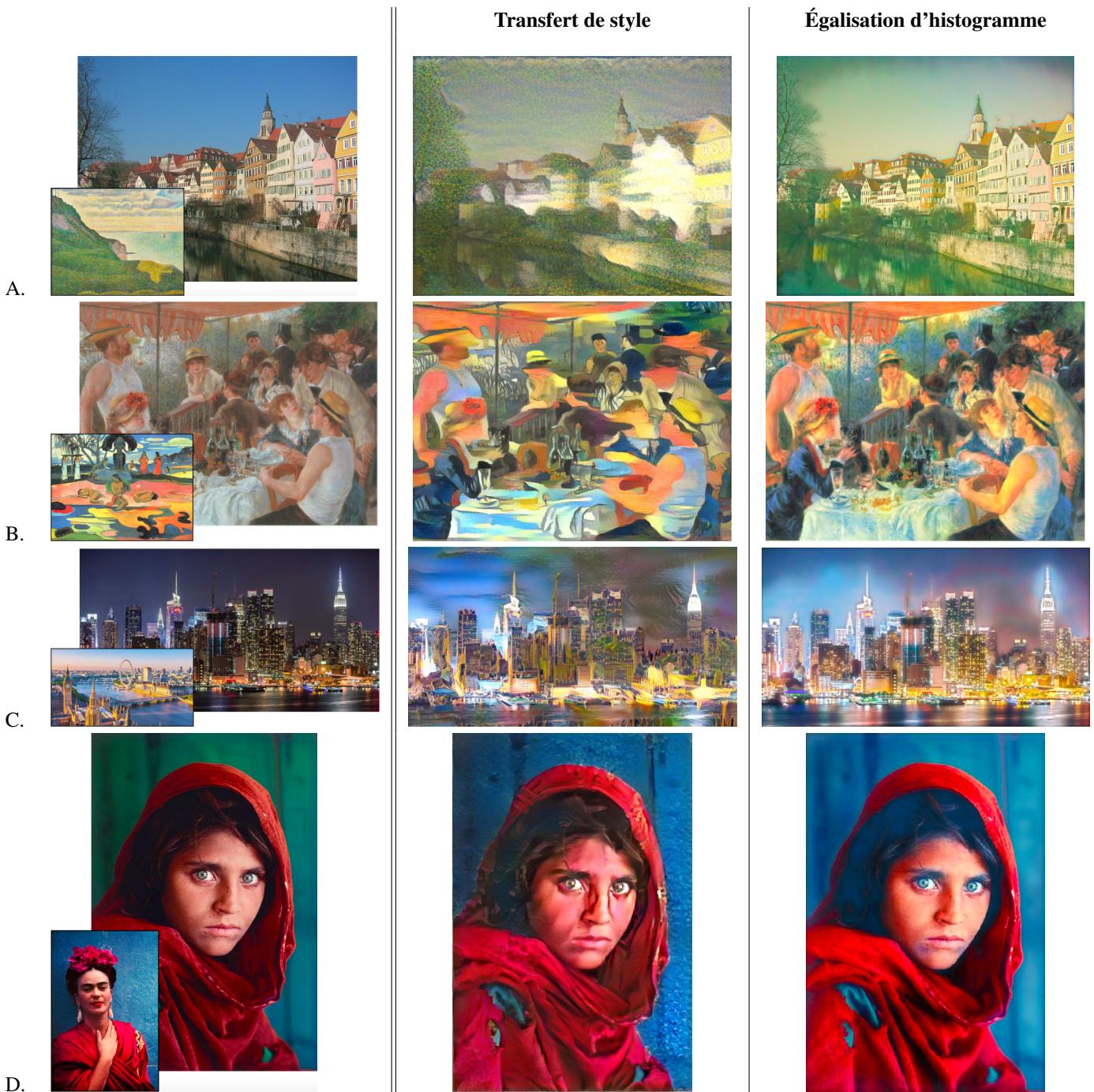


Table 1: **A.** Photo de Tübingen dans le style de "Paysage marin à Port-en-Bessin" de Georges Seurat. **B.** "Le Déjeuner des canotiers" de Auguste Renoir dans le style de "Mahana No Atua" de Paul Gauguin. **C.** Photo de New York de nuit dans le style d'une photo de Londres de jour. **D.** Photo de Sharbat Gula dans le style d'une photo de Frida Kahlo.

Conformément à notre intuition, nous pouvons voir que sur les deux premiers exemples, les performances de l'égalisation d'histogrammes sont nettement inférieures à celle de l'algorithme de Gatys. Les performances des deux algorithmes sur les deux jeux de photos **C.** et **D.** sont elles, discutables. Nous avons d'abord repris un exemple cité dans l'article avec des photos trouvées en ligne : le transfert du style d'une photo de Londres de jour sur une photo de New York de nuit. Ce problème est plus difficile qu'il en a l'air. En effet, la photo de New York de nuit comporte des zones très sombres

et des zones très claires. De plus, les nombreuses lumières font apparaître dans la nuit des reflets très forts dans l'eau et des cônes de lumière, ressemblant à des halos, autour des points lumineux les plus importants.

L'égalisation des histogrammes transfère bien les couleurs, cependant les halos de lumière restent très fortement visibles et nuisent à l'impression de lumière naturelle qui serait typique d'une photo prise en pleine journée. De plus, comme l'égalisation d'histogrammes ne se base absolument pas sur une compréhension du contenu sémantique, les régions les plus sombres de l'image originale de New York de nuit restent les plus sombres dans l'image générée. Ainsi, si sur la photo de Londres, les régions sombres correspondent à la végétation et les bâtiments sont, eux, bien éclairés, ceux-ci restent très sombres sur l'image en sortie.

Malgré tout, la photo résultante de l'égalisation des histogrammes semble plus crédible et ressemble bien plus à une photo que celle générée par le modèle de Gatys. En effet, celui-ci essaie de retrouver la texture présente dans le ciel de jour dans celui de nuit et non seulement renforce la présence des halos, mais crée encore plus de textures. Le tout laisse donner un air très artificiel et peu crédible à la photo.

De la même manière, le modèle de Gatys crée une texture dans une image qui n'en a pas sur l'exemple des photos de Sharbat Gula et Frida Kahlo. Si ceci pourrait être intéressant sur le mur derrière Sharbat, et pourrait ainsi donner l'impression que les deux femmes ont été photographiées au même endroit, l'ajout de texture sur le visage de la jeune femme nuit à l'image. L'image générée fait apparaître ce qui ressemble à des tâches de naissance, qui n'ont pas lieu d'être.

Par contre, l'algorithme d'égalisation d'histogrammes donne un résultat extrêmement satisfaisant. Celui-ci modifie les couleurs mais ne crée pas de textures qui semblent artificielles. De plus, la nouvelle palette de couleurs donne une impression de flou, comme sur la photo de Frida. Le résultant en est excellent.

4.3 Comparaison avec un SinGAN

En Septembre 2019, Tamar Rott Shaham a introduit le concept de SinGAN [10], un modèle génératif inconditionnel qui s'entraîne sur une seule image et permet un grand nombre de tâches liées au traitement d'images.

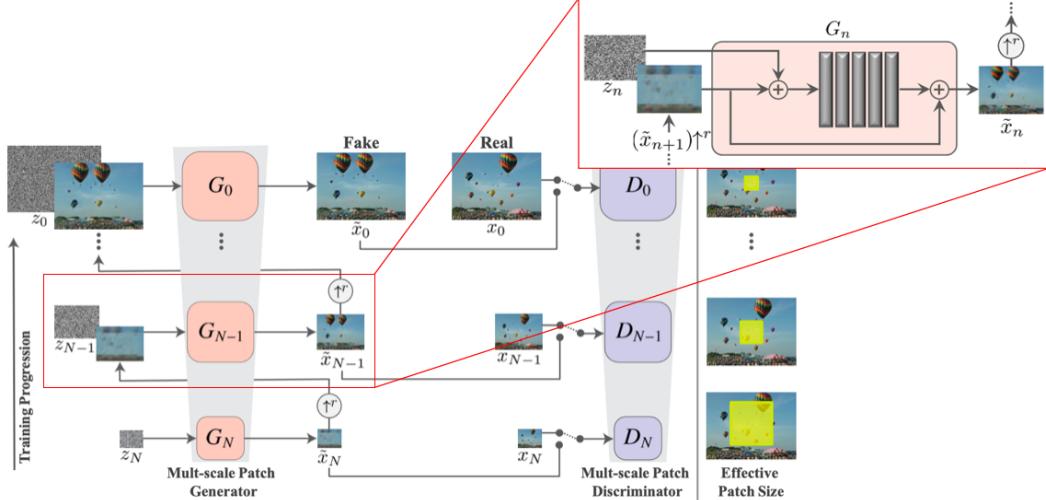


Figure 5: Pipeline multi-échelle d'un SinGAN

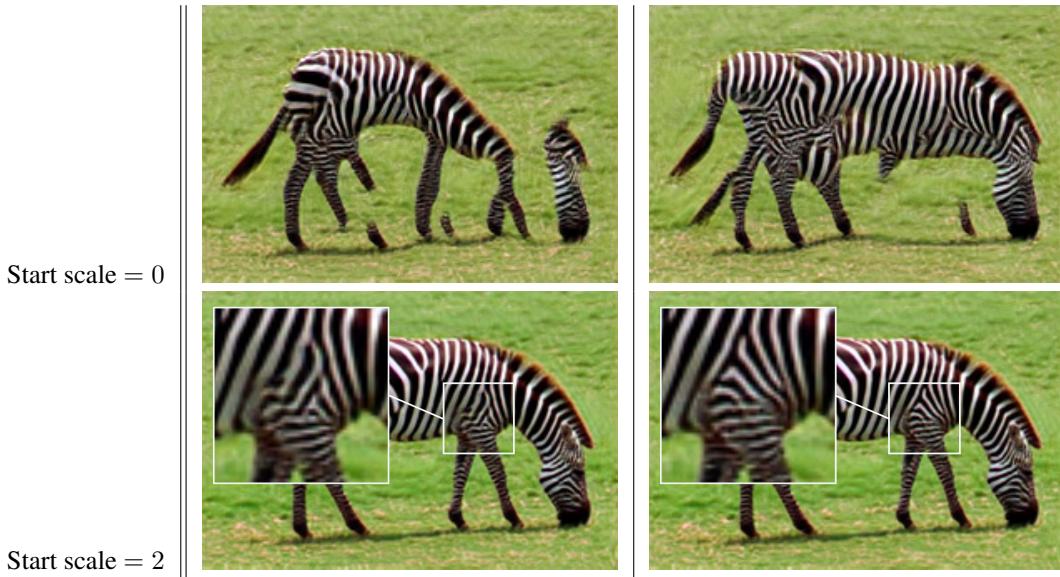
Un SinGAN est composé d'une pyramide de GANs qui travaillent, lors de l'entraînement et l'inférence, du plus grossier au plus fin. En effet, à une échelle $n \leq N$, le générateur G_n cherche à duper le discriminateur D_n en créant des images qui respectent la distribution de patchs de l'image originale x sous-échantillonnée par un facteur r^n : \tilde{x}_n .

Un SinGAN permet tout d'abord de générer des nouvelles images respectant la distribution de patchs - à toute échelle - de l'image originale.



Table 2: Images générées de différentes tailles et ratios.

Un SinGAN offre la possibilité de commencer l’inférence à n’importe quelle échelle, ce qui mène à des changements plus ou moins fins.



Nous pouvons voir ici qu’en générant de nouvelles images à partir de l’échelle 0, nous obtenons des zèbres peu réalistes. Par contre, si nous ne commençons qu’à l’échelle 2, seuls les détails des rayures des zèbres vont être modifiés, et l’arrangement général des objets de l’image sera préservé.

Une fois la distribution de patchs d’une image apprise par le modèle, celui-ci sert pour différentes tâches de traitement d’images. 6

Dans le cas de la harmonisation, le SinGAN est entraîné sur l’image originale sur laquelle nous pouvons ensuite ajouter des objets d’autres images. Le modèle se chargera ensuite de fusionner de façon harmonieuse l’objet externe dans l’image originale. En étendant le masque binaire qui signale au modèle où est situé l’objet externe, à l’image entière, le concept pourrait être réutilisé pour faire du transfert de style.

Nous avons donc commencé par l’appliquer sur l’exemple des peintures de Renoir et de Gauguin, avec différentes échelles initiales d’inférence. Un SinGAN a donc été entraîné sur la peinture de Gauguin. Puis, nous avons procédé à une tâche de harmonisation en donnant en entrée au modèle entraîné la peinture de Renoir.

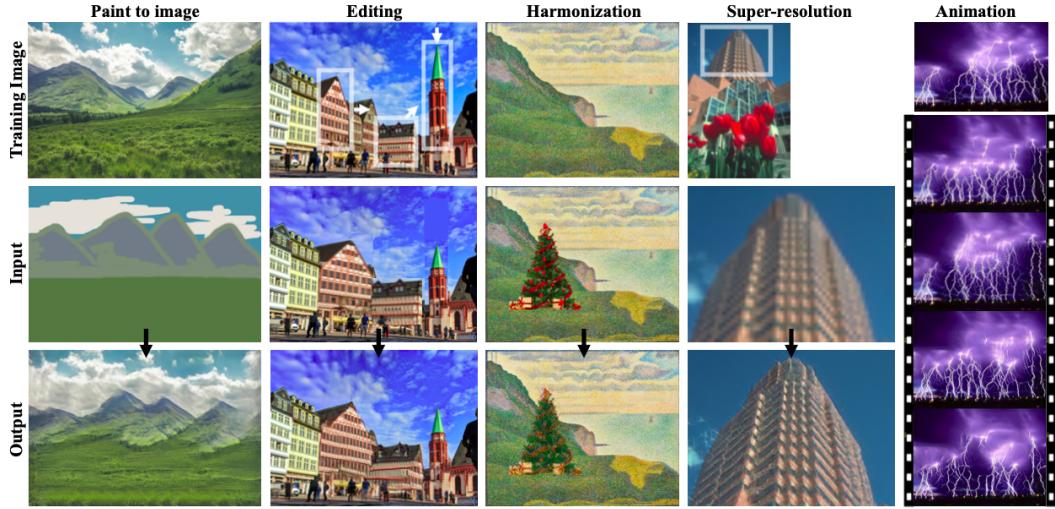


Figure 6: Utilisations d'un SinGAN



Table 3: Transfert de style utilisant un SinGAN : "Le Déjeuner des canotiers" de Auguste Renoir dans le style de "Mahana No Atua" de Paul Gauguin. En commençant de l'échelle 1 [en haut à gauche] à l'échelle 8 [en bas à droite].

Nous voyons bien, comme avec les zèbres, que plus l'échelle de départ est élevée, moins l'objet externe est modifié. Les SinGAN ne cherchent pas à comprendre le contenu sémantique de l'image, et ceci se perçoit dans les résultats: si l'échelle initiale est trop élevée, la peinture de Renoir est trop préservée. Cependant en prenant une échelle initiale plus faible, la peinture de Renoir perd vite de son sens. Regardons cependant comment les SinGANs gèrent le transfert de style sur les autres images tests.



Table 4: **A.** Photo de Tübingen dans le style de "Paysage marin à Port-en-Bessin" de Georges Seurat. **B.** "Le Déjeuner des canotiers" de Auguste Renoir dans le style de "Mahana No Atua" de Paul Gauguin. **C.** Photo de New York de nuit dans le style d'une photo de Londres de jour. **D.** Photo de Sharbat Gula dans le style d'une photo de Frida Kahlo.

Contrairement à l'égalisation d'histogrammes, les SinGANs se concentrent moins sur la palette de couleurs. Nous pouvons donc facilement voir, notamment sur la peinture de Renoir et sur la photo de New York et nuit, que les couleurs sont restées très semblables aux images originales, donnant un net avantage au modèle de Gatys.

Toutefois, sur la photo de Tübingen, le SinGAN a réussi à lui donner un style impressionniste, même sans le transfert de couleurs. De plus, si de loin, l'image générée par le modèle de Gatys semble

meilleur, cela pourrait être simplement parce qu'elle est visuellement plus similaire à la peinture de Seurat. Cependant, en regardant de plus près le contenu sémantique, le ciel de Tübingen a pris, avec le modèle de Gatys, les couleurs vertes très présentes sur une peinture comportant beaucoup de végétation. Avec le SinGAN, celui-ci a gardé sa couleur bleue. Même si le ciel pourrait être plus texturé, celui-ci a une couleur plus proche du bout de ciel dépassant des nuages sur la peinture de Seurat. Ainsi, en travaillant mieux l'échelle initiale d'inférence, voire en fusionnant possiblement deux images générées avec deux échelles initiales différentes, le SinGAN semblerait pouvoir produire des résultats plus cohérents du point de vue du contenu sémantique.

Finalement, le modèle de Gatys présente les mêmes défauts sur la photo de Sharbat que vus précédemment. L'image générée par le SinGAN a encore une fois une palette de couleurs plus similaire à celle de l'image initiale de Sharbat, mais un style qui semble plus proche de la photo de Frida avec un même flou artistique.

5 Extensions du transfert de style

5.1 Application aux vidéos

La difficulté supplémentaire du transfert de style pour des vidéos réside dans la cohérence temporelle. On veut en effet qu'un même objet soit représenté de la même façon dans les différentes images, même s'il se déplace. Cela ne serait pas le cas si l'on appliquait simplement un algorithme de transfert de style à chacune des images de la vidéo.

Un tel modèle a été créé en 2016 par Manuel Ruder, Alexey Dosovitskiy et Thomas Brox [12]. Leur méthode est basée sur le transfert de style d'image fixe, et ajoute deux composantes: la cohérence à court terme et la cohérence à long terme.

Conclusion

Le modèle de Gatys reste une référence absolue pour le transfert de style entre deux images. Cependant, il est important de noter que son modèle repose sur une association entre texture et contenu sémantique. Ainsi, son modèle obtiendra de moins bons résultats sur une photo ou une image comportant peu de texture. Dans ce cas là, il pourrait être plus intéressant d'utiliser un simple algorithme d'égalisation d'histogrammes qui générera de meilleurs résultats sans réseaux de neurones.

References

- [1] Leon A. Gatys et al. (2015). *Image Style Transfer Using Convolutional Neural Networks*
- [2] Krizhevsky et al. (2012). *ImageNet Classification with Deep Convolutional Neural Networks*
- [3] Leon A. Gatys et al. (2015). *Texture Synthesis Using Convolutional Neural Networks*
- [4] Yongcheng Jing et al. (2017). *Neural Style Transfer: A Review*
- [5] J. B. Tenenbaum and W. T. Freeman. (2000). *Separating style and content with bilinear models*
- [6] D. P. Kingma et al. (2014). *Semi-supervised Learning with Deep Generative Models*
- [7] A. A. Efros and W. T. Freeman. (2001). *Image quilting for texture synthesis and transfer*
- [8] V. Kwatra, A. Schodl et al. (2003). *Graphcut textures: image and video synthesis using graph cuts*
- [9] L. Wei and M. Levoy. (2000). *Fast texture synthesis using tree-structured vector quantization*
- [10] T. R. Shaham, T. Dekel, T. Michaeli (2019). *SinGAN: Learning a Generative Model from a Single Natural Image*
- [11] A. Mahendran and A. Vedaldi. (2014). *Understanding Deep Image Representations by Inverting Them*
- [12] Manuel Ruder, Alexey Dosovitskiy, Thomas Brox. (2016). *Artistic style transfer for videos*