

# **Développement d'un modèle de transfert de style**

**Saïd ZELLAGUI**

## **Formation / Ecole**

Ingénieur Machine Learning / OpenClassrooms

## **Numéro du projet / Date de soutenance**

Projet 8 / 11-02-2022



# Kaggle competition

**Rapport détaillé du projet n°8**

**Date de la rédaction : février 2022**

## Table des matières

Synthèse de travail.....	3
Résumé .....	3
1. Introduction .....	4
2. Présentation de la compétition.....	4
3. Données .....	4
4. Outils .....	5
5. Présentation du modèle utilisé.....	5
6. Notre architecture personnalisée .....	5
7. Métrique d'évaluation .....	7
8. Résultats et discussion.....	7
9. Conclusion.....	8
Références .....	9

## Synthèse de travail

### Résumé

Ce travail a pour but de transférer les caractéristiques d'une image à une autre. Autrement dit, prendre une image d'un domaine source, puis la transformer en une image du domaine cible sans nécessairement avoir un mappage un à un entre les images de l'entrée au domaine cible dans l'ensemble d'entraînement.

La génération de photographies à partir de peintures, comme son nom l'indique, est la synthèse d'images photoréalistes données à une peinture, généralement par un artiste célèbre ou une scène célèbre. Dans ce challenge, des algorithmes basés sur les réseaux de neurones ont été utilisés afin de répondre à l'objectif principal qui consiste à créer un GAN qui génère 7 000 à 10 000 images de style Monet à partir d'images réelles.

Nous avons donc utilisé le CycleGAN qui est, une approche de formation de modèles de traduction d'image à l'aide de l'architecture de modèle de réseau antagoniste génératif, ou GAN. Notre choix s'est orienté vers cette approche du fait que les images fournies ne sont pas appariées. La puissance des cycleGANs réside dans sa capacité d'apprendre à capturer les caractéristiques spéciales d'une collection d'images et déterminer comment ces caractéristiques pourraient être traduites dans l'autre collection d'images, le tout en l'absence d'exemples de formation appariés.

Dans ce qui suit, la démarche du travail sera détaillée. Il s'agit des étapes réalisées pour arriver à la création des images de Monet à partir d'images réelles.

## 1. Introduction

Kaggle est une plate-forme pour apprendre, s'inspirer et partager des idées avec des spécialistes en data science. Il existe de nombreux ensemble de données qui peuvent être utilisées pour contribuer à des projets afin de faire progresser leurs compétences. Les utilisateurs peuvent également participer à des compétitions pour résoudre des problèmes réels de science des données. Ceux-ci donnent aux utilisateurs la possibilité de mettre leurs compétences à l'épreuve et d'apprendre de la communauté de plus de 3 millions scientifiques des données. Certaines des compétitions incluent également des prix en argent, qui peuvent atteindre 150 000 \$.

L'aspect communautaire permet de partager des travaux sous forme de notebook pour obtenir des retours mais aussi pour voir la manière dont les autres membres travaillent sur certains problèmes.

## 2. Présentation de la compétition

La compétition intitulée « *I'm Something of a Painter Myself* » a été choisi pour valider le projet 8. Cette compétition rentre dans le contexte de traduction d'image à image. Cela consiste à générer une nouvelle version synthétique d'une image donnée avec une modification spécifique, telle que la traduction d'un paysage d'été en hiver. Dans ce challenge, le but est d'ajouter le style Monet aux photos réelles fournies. C'est une compétition du type « ongoing ».

## 3. Données

Les données pour ce challenge ont été récupérées à partir de la plateforme Kaggle : <https://www.kaggle.com/c/gan-getting-started>. Le dataset comprend quatre répertoires : `monet_tfrec`, `photo_tfrec`, `monet_jpg` et `photo_jpg`.

- **monet\_jpg** : 300 peintures de Monet format 256x256 au format jpeg
- **monet\_tfrec** : 300 peintures de Monet format 256x256 au format tfrecord
- **photo\_jpg** : 7028 photos de taille 256x256 au format jpeg
- **photo\_tfrec** : 7028 photos de taille 256x256 au format tfrecord

Les répertoires `monet_tfrec` et `monet_jpg` contiennent les mêmes images de peinture, et les répertoires `photo_tfrec` et `photo_jpg` contiennent les mêmes photos. Nous allons donc utiliser

les deux répertoires `monet_jpg` et `photo_jpg` présentant les images sous format jpeg. Le dataset utilisé pour l'entraînement comprend 300 images de taille 256x256 en format jpeg dans le répertoire `monet_jpg`.

#### 4. Outils

La méthode utilisée ici est une adaptation et amélioration du tutoriel réalisée par Amy Jang en 2021 (<https://www.kaggle.com/amyjang/monet-cyclegan-tutorial>). Nous avons utilisé une architecture CycleGAN pour ajouter le style Monet aux photos. L'implémentation du réseau de générateurs est basée sur l'approche décrite pour le transfert de style par Justin Johnson dans l'article de 2016 intitulé "[Perceptual Losses for Real-Time Style Transfer and Super-Resolution](#)".

Les analyses ont été faites sur l'outil Google Collaboratory. Ce choix a été fait pour pouvoir bénéficier d'un GPU lors de la phase de modélisation à l'aide de réseau de neurones. Un seul notebook Jupyter "`Projet_8_kaggle_competition_openclassroom.ipynb`" a été produit à l'issue de ce travail. Pour réaliser cette étude, le langage de programmation utilisé est Python.

#### 5. Présentation du modèle utilisé

La formation d'un modèle pour la traduction d'image à image nécessite généralement un grand ensemble de données d'exemples appariés. Ces ensembles de données peuvent être difficiles et coûteux à préparer, voire impossibles dans certains cas, comme les photographies de peintures d'artistes décédés depuis longtemps.

Le CycleGAN est une technique qui implique la formation automatique de modèles de traduction d'image à image sans exemples appariés. Les modèles sont formés de manière non supervisée à l'aide d'une collection d'images des domaines source et cible qui n'ont pas besoin d'être liées de quelque manière que ce soit.

#### 6. Notre architecture personnalisée

Nous avons deux collections de photographies non appariées, ce qui signifie qu'il s'agit de photos d'endroits différents à des moments différents ; nous n'avons pas exactement les mêmes scènes en style de Monet et en réel.

- **Collection 1** : photos réelles.
- **Collection 2** : photos de Monet.

Nous avons donc développé une architecture de deux GAN, et chaque GAN a un modèle discriminateur et un modèle générateur, ce qui signifie qu'il y a quatre modèles au total dans l'architecture. Le premier GAN génère des photos style Monet à partir de photos réelles prises par une caméra, et le deuxième GAN génère des photos réelles à partir de photos style Monet.

- **GAN 1** : traduit les photos réelles (collection 1) en photos de Monet (collection 2).
- **GAN 2** : traduit les photos de Monet (collection 2) en photos réelles (collection 1).

Chaque GAN a un modèle de générateur conditionnel qui synthétise une image à partir d'une image d'entrée. Et chaque GAN a un modèle discriminateur pour prédire la probabilité que l'image générée provienne de la collection d'images cible. Les modèles de discriminateur et de générateur pour un GAN sont formés sous une perte contradictoire normale comme un modèle GAN standard.

Nous pouvons résumer les modèles générateur et discriminateur du GAN 1 comme suit :

**Modèle de générateur 1 :**

- **Input** : prend des photos réelles (collection 1).
- **Output** : génère des photos de Monet (collection 2).

**Modèle de discriminateur 1 :**

- **Input** : prend des photos de Monet de la collection 2 et de la sortie du générateur modèle 1
- **Output** : calcul la probabilité que l'image provienne de la collection 2.

De même, nous pouvons résumer les modèles générateur et discriminateur du GAN 2 comme suit :

**Générateur Modèle 2 :**

- **Input** : prend des photos de Monet (collection 2).
- **Output** : génère des photos réelles (collection 1).

**Modèle de discriminateur 2 :**

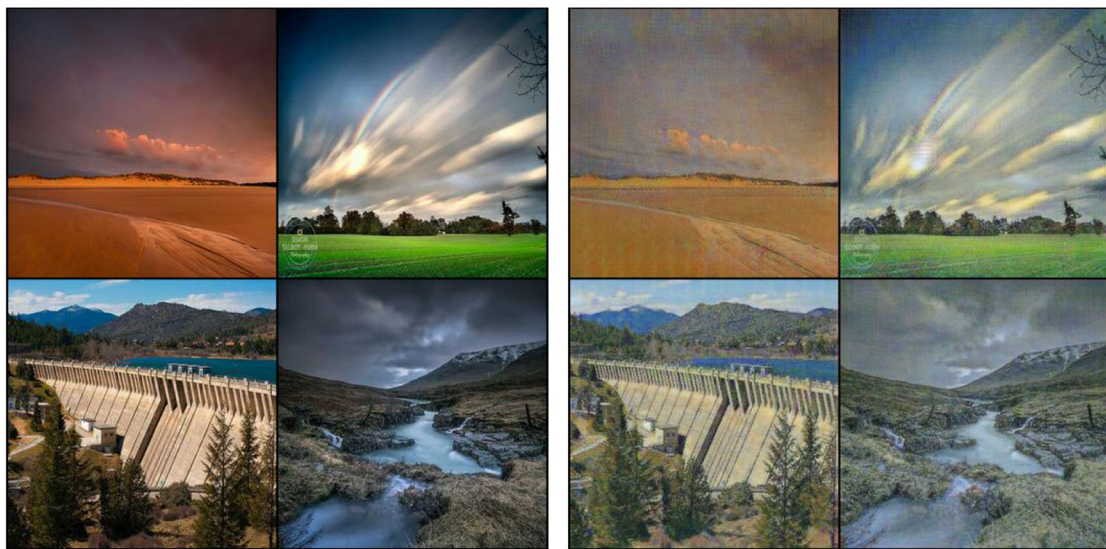
- **Input** : prend des photos réelles de la collection 1 et de la sortie du modèle de générateur 2.
- **Output** : calcul la probabilité que l'image provienne de la collection 1.

## 7. Métrique d'évaluation

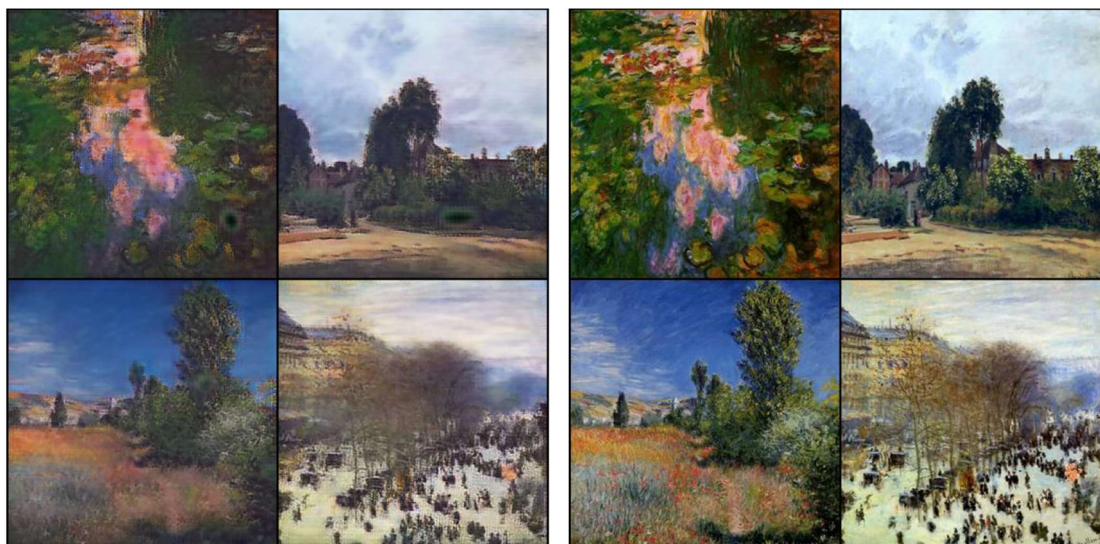
Les réseaux antagonistes génératifs n'ont pas de fonction objective, ce qui rend difficile la comparaison des performances de différents modèles. Les modèles de générateur GAN sont évalués en fonction de la qualité des images générées, souvent dans le contexte du domaine du problème cible. Dans notre cas, il s'agit d'une évaluation manuelle des images synthétisées par un modèle de générateur. Il s'agit d'utiliser le modèle générateur pour créer un lot d'images de synthèse, puis d'évaluer la qualité et la diversité des images par rapport au domaine cible. C'est là que les juges humains sont invités à classer ou à comparer des exemples d'images réelles et générées du domaine.

## 8. Résultats et discussion

Nous avons exécuté le modèle pour l'ensemble de données dans le répertoire photo\_jpg, mais en raison du manque de ressources, nous avons simplement exécuté le modèle pendant 20 époques et obtenait les résultats suivants.



(a) Réelles  $\rightarrow$  Monet



(b) Monet → réelles

**Figure 1 :** (a) images réelles convertis en image de Monet (b) images de Monet convertis en image réelles

Dans l'ensemble, les résultats produits par CycleGAN sont très bons et la qualité d'image traduite se rapproche de l'image originale (figure 1). Il est très difficile de distinguer l'image synthétisée de la vraie. Ces résultats montrent la capacité des CycleGAN par rapport à d'autres modèles d'apprentissage entièrement supervisé. Le CycleGAN a largement dépassé les autres techniques de traduction d'images non supervisées disponibles.

## 9. Conclusion

Le transfert de style fait référence à l'apprentissage du style artistique d'un domaine, souvent des peintures, et à l'application du style artistique à un autre domaine, comme les photographies. Le CycleGAN est démontré en appliquant le style artistique de Monet à des photographies de paysages, ainsi que l'inverse.

Le but de ce projet était d'utiliser une approche de modélisation non supervisée afin de traduire des images. Les données collectées sur le style artistique de Monet nous permettent de faire une extraction de caractéristiques afin de les transférer à des images réelles. On dispose de 300 images de style Monet de taille 256x256x3 de différentes scènes, ce qui était suffisamment large pour l'entraînement de notre modèle.

Les résultats prédits ont montré un bon accord avec les images originaux. Ces résultats indiquent que le modèle développé est capable de transférer les caractéristiques d'une image à une autre avec une grande précision.



## Perspectives et limites

Lors d'utilisations de CycleGAN, il est important d'être conscient de ses forces et de ses limites. Cela fonctionne bien sur les tâches qui impliquent des changements de couleur ou de texture, comme les traductions de photos de jour en nuit, ou les tâches de photos à peinture comme le transfert de style de collection. Cependant, les tâches qui nécessitent des modifications géométriques substantielles de l'image, telles que les traductions de chat à chien, échouent généralement.

Dans le cas des GANs, la notion éthique est fondamentale et faite beaucoup débat. Les GAN offrent des perspectives intéressantes pour plusieurs domaines, notamment en analyse d'images et en montage vidéo. Néanmoins, ils pourraient contribuer à créer et répandre des fake news. Évidemment ces inconvénients et les dangers qui en découlent ne doivent pas être une barrière au progrès technologique. Mais ils demandent une meilleure régulation, notamment grâce à des outils de traçabilité des données et des algorithmes utilisés.

Un CycleGAN a un flux de données plus complexe car il comporte deux paires générateur-discriminateur. L'accès massif à la mémoire externe entraîne également une longue latence pour l'apprentissage et l'inférence, par conséquent l'équipement utilisé pour le calcul doit être adapté.

## Références

- [1] Zhu, Jun-Yan, Taesung Park, Phillip Isola and Alexei A. Efros. "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks." 2017 IEEE International Conference on Computer Vision (ICCV) (2017): 2242-2251.
- [2] Salimans, Tim, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford and Xi Chen. "Improved Techniques for Training GANs." NIPS (2016).
- [3] Borji, Ali. "Pros and Cons of GAN Evaluation Measures." ArXiv abs/1802.03446 (2019): n. pag.
- [4] Heusel, Martin, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler and Sepp Hochreiter. "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium." NIPS (2017).