**Projet Graffiti**

**Par Thomas Bek**

Le projet Graffiti s’inscrit dans le cadre du projet libre de cinquième année d’EPITECH.

Le but de Graffiti est de permettre, par l’utilisation de réseaux de neurones convolutionnels (CNNs), la classification automatique d’œuvres d’art picturales digitalisées par style artistique.

Ce projet a été inspiré par les travaux de Gatys et Al., rapportés dans A Neural Algorithm of Artistic Style.

Le présent document a pour but de constituer un rapport de l’avancement global du projet Graffiti, des problèmes et interrogations soulevées pendant son développement et des résultats obtenus.

Au-delà de sa finalité technique, le projet Graffiti constitue d’abord pour son auteur un outil d’auto-formation au sujet du Machine Learning, et plus particulièrement à la compréhension, la conception et l’utilisation de réseaux de neurones convolutionnels.

**1/- Graffiti**

* 1. **Introduction**

La définition de ce qui constitue une œuvre d’art est disputée, et a changé à travers les siècles.

A travers plusieurs siècles d’histoire de l’art, la définition de ce qui constitue ou non une œuvre d’art a changé, et reste constante évolution. On peut s’accorder à dire que dans le cas des beaux-arts, les artistes parviennent à créer des expériences visuelles uniques, qui appellent à nos émotions, notre intuition, notre intelligence ou encore nos souvenirs… Dans le cas particulier de la peinture, l’expérience visuelle qui nous est donnée par l’artiste peut être décomposée en deux parties : d’une part le « sujet », le contenu visuel de l’image, constitué d’objets plus ou moins reconnaissables comme un personnage, un arbre, le ciel… D’autre part l’interprétation de ce sujet par l’artiste, son « style » qui lui est propre. C’est à travers ce lien complexe entre le sujet et le style qu’un artiste parvient à créer une expérience visuelle unique [1].

L’analyse et la comparaison de travaux artistiques sont le sujet de recherches et conversations depuis que nos ancêtres peignaient les murs de leurs cavernes, mais ce n’est que depuis peu que ces domaines ont été ouverts à l’analyse par des machines, par le biais des avancées récentes en intelligence artificielle. Ce qui constitue le style artistique peut paraître un concept nébuleux, voire subjectif, et le réduire à un jeu mathématique peut sembler difficile sinon impossible, ou même presque insultant. Cela n’a cependant pas empêché les esprits les plus pragmatiques de notre histoire d’essayer. Le nombre d’or par exemple, défini par Euclide, fût tenu pendant longtemps comme un critère de beauté absolue et se retrouve dans grand nombre d’œuvres à travers l’histoire de l’art.

Les avancées récentes en intelligence artificielle nous permettent d’approcher le style artistique de manière systématique, par un procédé qui mimique la façon dont notre propre cerveau traite les images.

* 1. **Cadre du projet**

L’objet du projet Graffiti est de définir une approche automatisée de l’analyse de peintures, d’en extraire l’information liée au style artistique et de permettre la comparaison de différentes œuvres de manière concrète.

Tout d’abord, il sera nécessaire de séparer le sujet et le style d’une œuvre picturale digitalisée. Un portrait, qu’il soit peint par Van Gogh ou Rembrandt reste un portrait, bien que le résultat final soit si différent (cf. Fig. 1).

Des travaux existants [1, 4] ont montré l’utilité de Réseaux de Neurones Convolutionnels (CNN) pour ce type de problématiques. Ils nous permettent de créer une représentation stratifiée d’une image donnée, chaque couche contenant des informations liées au sujet de l’image. La corrélation entre ces couches peut être comprise comme la définition du style de l’image.

Dans un second temps, il sera nécessaire de comparer l’information *a priori* abstraite liée à différentes œuvres de manière concrète. Les styles artistiques de Picasso et Delacroix, figures de proue du Cubisme et du Romantisme, sont évidemment très différentes au premier coup d’œil (cf. Fig. 2), comment quantifier ces différences ? Il faudra concevoir une approche généralisée à la comparaison de styles artistiques. Dans ce but, le système Graffiti devra pouvoir prévoir avec un intervalle de confiance raisonnable l’artiste, le courant artistique ou l’époque d’une œuvre picturale digitalisée. Des approches statistiques ont été notamment utilisées dans la détection de faux artistiques [15]. L’utilisation de méthodes similaires semble être une approche valable dans notre cas.

Fig

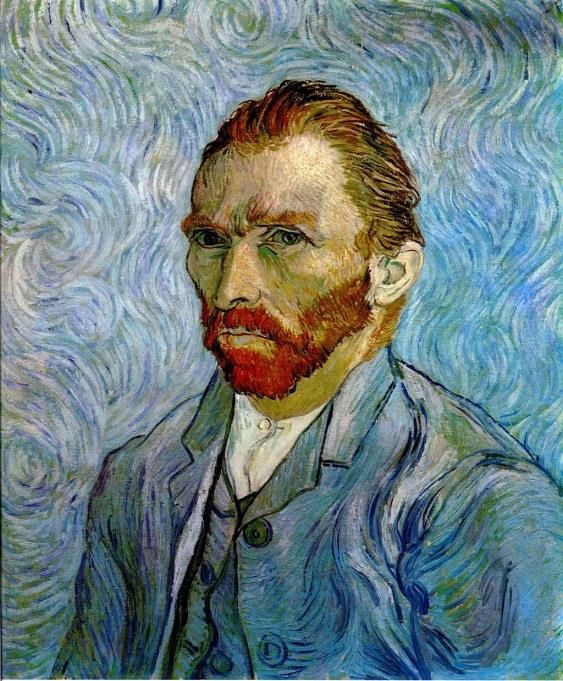
. 1:

*Auto-portrait*, Vincent Van Gogh, 1889 (gauche);

*Auto-portrait*, Rembrandt, 1652 (droite)

Fig. 2:

Nu assis, Mademoiselle Rose, Eugene Delacroix, 1820-1824 (gauche) ; Nu assis, Pablo Picasso, 1908 (droite)



* 1. **Revue de littérature**

L’analyse du style d’images artistiques a été le sujet de nombreux articles scientifiques ces dernières années.

La vision par ordinateur, l’analyse graphique de peintures et dessins, la science de l’art, la génération non-photoréaliste (Non Photorealistic Rendering), la stylisation d’images ou de vidéo sont autant de domaines de recherches très actifs [7]. De nombreux travaux se concentrent notamment sur la reproduction de styles artistiques divers sur des images non-photoréalistes, avec des applications directes en design graphique, dans le montage en post-production pour le cinéma, dans les moteurs graphiques de jeux… [6] De nombreuses approches sont suivies [5], en particulier le Machine Learning (ML) [8]. Le style d’une œuvre picturale digitalisée a été extrait avec succès et appliqué à une image cible [1], and plus récemment à une vidéo [3]. Le développement des réseaux de neurones profonds (Deep Neural Networks), domaine de recherche activement développé a vu de nombreuses applications récentes dans le domaine de la vision par ordinateur ou la reconnaissance faciale par exemple.

Les applications liées à l’art de l’intelligence artificielle sont un champ de recherche nouveau et grandissant. Google a lancé de nombreux projets sur le sujet, notamment le Google Art Project, and utilise actuellement sa technologie Deep Dream pour des recherches sur la création d’œuvres artistiques originales par l’intelligence artificielle. La détection de faux est une autre application clé des recherches courantes en science de l’art. Par exemple, des méthodes statistiques ont permis récemment la détection de faux pour des dessins [15, 17]. Dans une étude récente, la technique a permis de distinguer huit dessins de l’artiste du XVIe siècle Pieter Bruehgel l’Ancien de cinq imitations qui lui étaient faussement attribuées jusqu’à il y a une dizaine d’années environ [16].

De plus, de par le fait que les réseaux de neurones sont inspirés par notre compréhension actuelle de l’architecture et du fonctionnement biologique de notre cerveau, il a été montré qu’ils peuvent nous fournir une perspective intéressante sur la façon dont nous traitons l’information ou apprenons à exécuter de nouvelles tâches [23].

Le Deep Learning and les technologies de réseaux de neurones sont largement disponibles pour la recherche, notamment par le biais de l’ouverture d’outils et de frameworks tels que Caffe [18] ou Torch [20].

**1.4 A Neural Algorithm of Artistic Style**

Gathys et al. présentent dans [1] une méthode permettant de transférer le style artistique d’une image source à une image cible. Ils utilisent pour ce faire un réseau de neurones convolutionnels, et parviennent notamment à donner à une photo de Turbingen en Allemagne le style caractéristique de la *Nuit Etoilée* de Van Gogh.

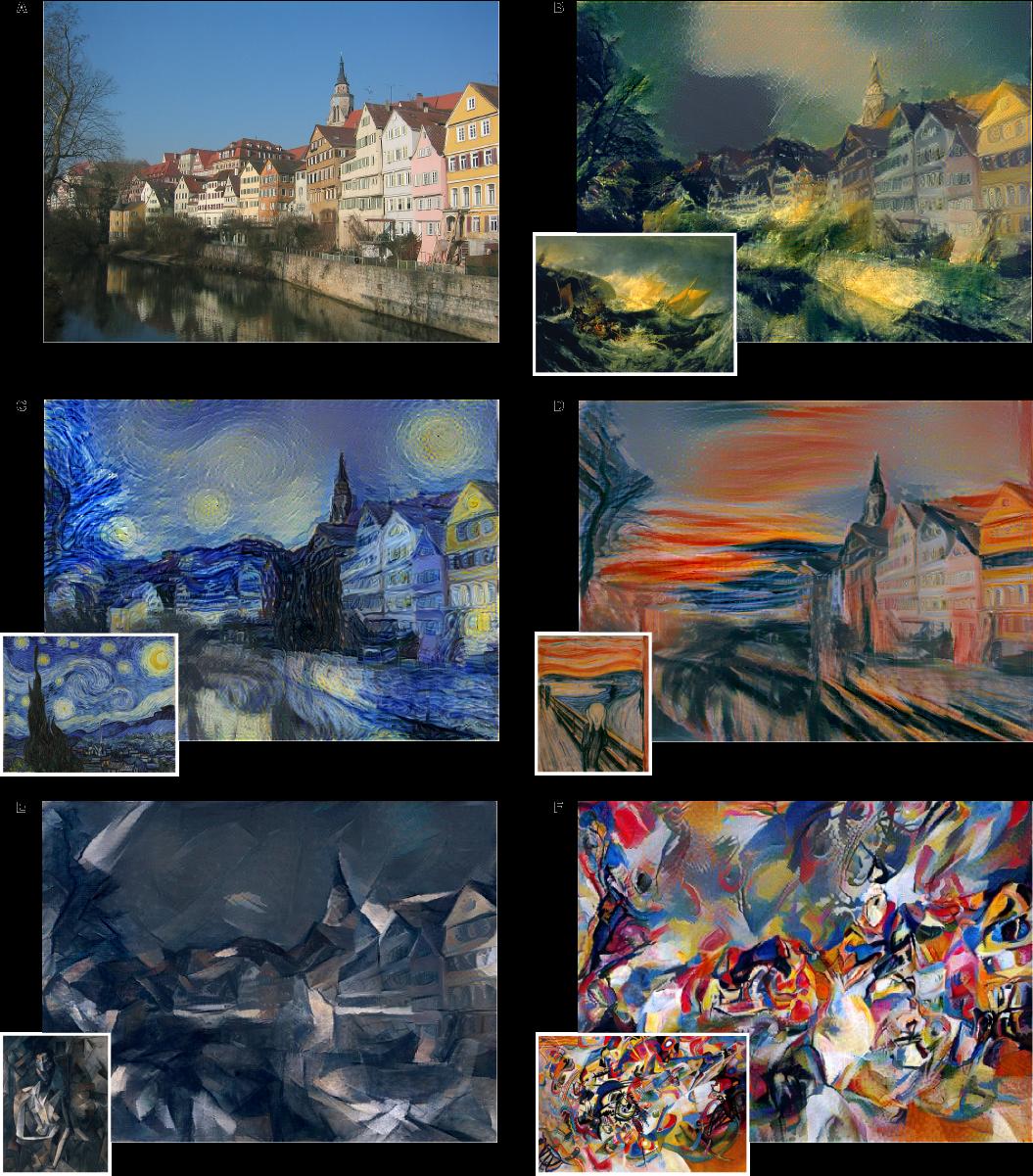
Les auteurs partent de l’observation que le style artistique d’une peinture peut être comprise comme la corrélation entre différents éléments d’une image.

Ils utilisent le modèle VGG-19, auquel ils fournissent d’abord l’image cible. Ils conservent l’activation du dernier niveau de convolution du modèle comme représentation du contenu de l’image.

Ils fournissent ensuite au modèle l’image source. Pour un niveau de convolution donnée, ils calculent une matrice de Gram. La somme des matrices de Gram obtenues est considérée comme la représentation du style de l’image.

Le transfert du style de l’image source vers l’image cible est compris comme un problème d’optimisation.

Le modèle reçoit en entrée une texture de bruit blanc qu’il est chargé de « peindre » avec à la fois le contenu de l’image cible et le style de l’image source. La méthode consiste à entrainer le CNN à minimiser une fonction, consistant en la somme de la perte de signal de style et de la perte de signal de contenu sur l’image en cours de peinture.



**1.5 Principe de Graffiti**

L’idée derrière Graffiti est simple. Le projet se décomposera en deux modules. D’une part, le module d’extraction du style artistique, d’autre part, un module prédisant le style d’une peinture.

Image X

Style de Image X

Image 1

Image 2

Image n

Style 1

Style 2

Style n

Prédiction du courant de X

Extraction du Style

Extraction du Style

Classification

En comprenant le style artistique de la même façon que Gethys et al., il doit être possible de l’extraire d’une image, avant de le comparer au style d’autres œuvres.

Il devrait être possible d’entraîner un second réseau de neurones à prédire le courant auquel appartiennent les données de style qui lui sont présentées.

Eventuellement, il devrait être possible de prévoir d’autres attributs, comme l’artiste, l’époque…

**2/- Compte-rendu du projet**

**Semaine 1 : Travaux préliminaires**

**Objectifs :**

Etablir une revue de la littérature sur les sujets connexes au projet Graffiti : réseaux de neurones convolutionnels, applications de l’intelligence artificielle aux beaux-arts, reconnaissance d’image etc…

Rechercher des exemples d’applications existantes proches du projet Graffiti.

Acquérir une compréhension de base des principes des réseaux de neurones convolutionnels.

**Semaine 2 : Extraction du style**

**Objectifs :**

Réaliser une implémentation de la méthode décrite par Gethys et al.

Acquérir une expérience pratique de l’utilisation des CNNs.

**Résultats :**

Implémentation réalisée à l’aide du framework TensorFlow.

**Observations :**

Après une recherche préliminaire sur les différentes méthodes possibles pour classifier ultérieurement les données de style, j’ai décidé d’utiliser le modèle Inception, développé par Google, qui semblait pouvoir répondre à mes besoins. L’interface avec le framework TensorFlow était librement disponible et facile à réutiliser. Pour pouvoir assurer plus facilement la compatibilité entre mes deux modules, j’ai choisi de me tourner entièrement vers l’utilisation de TensorFlow, plutôt que d’utiliser Caffe (qui était utilisé dans l’article original)

L’implémentation en tant que telle ne m’a pas posé trop de problèmes, grâce à la documentation claire de TensorFlow et aux nombreux exemples existants.

L’entraînement donne de bons résultats au-delà de 1000 itérations.

Cependant, le processus est très lent sur ma configuration. Il faut généralement plus de 30 minutes pour peindre une image de l’ordre de 800x800

Les poids et formules proposées par Gethys et al. Fonctionnent bien, mais je regrette que cette contrainte de temps m’empêche de pouvoir vraiment expérimenter avec des poids différents ou de nombreux styles.

**Semaines 3 et 4 : Extraction du style**

**Objectifs :**

Adapter le script réalisé à la problématique de Graffiti, et ne conserver que les données liées au style.

Formatter les données de façon à pouvoir les utiliser par la suite, pour la classification.

**Résultats :**

Données de style récupérées sous la forme d’une image.

**Observations :**

Je pensais au départ récupérer les données de style directement sous la forme d’un tenseur et adapter l’entrée du classifier pour les recevoir.

Après de nombreux essais je décide d’adopter une approche différente. Il s’avère trop complexe de rendre compatibles la sortie de mon extracteur de style et l’entrée d’inception. Les données de style comprennent au total quatre tenseurs, chacun étant associé à un poids. Ma première approche consistait à ajouter deux niveaux de pooling supplémentaires à l’entrée d’Inception, mais sans succès.

Je décide d’enregistrer directement les données de style sous la forme d’une image, que je pourrais donner directement en entrée à mon classifier. Pour ce faire, j’adapte mon extracteur pour qu’il transfère le style de mon image source sur une texture de bruit blanc. Ainsi, et en augmentant le poids des niveaux les plus fins, j’obtiens une texture relativement uniforme correspondant au style de mon image source.

**Semaine 5 : Classification**

**Objectifs :**

Bâtir une base de données de peintures et les soumettre au module d’extraction

Entrainer Inception3 par transfert à identifier de nouvelles classes, correspondant à des mouvements artistiques

Obtenir des séries de résultats expérimentaux

**Résultats :**

Traiter suffisamment d’images prend énormément de temps.

Je n’ai pu entraîner Inception3 que sur des sets de données extrêmement réduits. Malheureusement la qualité des images de style était pauvre (200 itérations). Les résultats obtenus sont peu concluants, les styles sont trop facilement confondus, et la prédiction semble dépendre beaucoup trop des couleurs.

Par curiosité, j’ai testé le modèle sur les peintures non traitées.

On obtient à peu de choses près le même intervalle de confiance autour de 76% !

Après réflexion, utiliser Inception ne pouvait pas être une bonne idée. L’objectif même d’Inception est d’identifier des éléments à différentes échelles du contenu. Pour Graffiti, il faudrait utiliser un modèle adapté à l’identification de textures…

**Semaine 6 : Rédaction et préparation**

**Objectifs :**

Rédaction du rapport de projet

Préparation de la soutenance finale

**Conclusions :**

A l’issue de ce projet, je pense avoir pu observer que les matrices de Gram ne suffisent pas décrire le style d’une image, au sens du style artistique. Il me semble qu’il doit être possible cependant d’établir un modèle spécialisé dans ce but. Peut-être serait-il possible d’obtenir de meilleurs résultats à l’aide d’un modèle adapté aux textures pour commencer.

Malgré les résultats expérimentaux limités que j’ai pu obtenir, j’ai appris beaucoup au sujet des réseaux de neurones convolutionnels, et du machine learning plus largement. Je maîtrise dorénavant les concepts de base des CNNs et j’ai pu acquérir une expérience pratique du développement avec TensorFlow.

**Références**

1. Gatys, L.A., Ecker, A.S., Bethge, M.: *A neural algorithm of artistic style.*

CoRR abs/1508.06576 (2015)

[http://arxiv.org/abs/1508.06576](http://arxiv.org/abs/1508.06576%20)

1. Nikulin, Y., Novak, R.: *Exploring the neural algorithm of artistic style.*

CoRR abs/1602.07188 (2016)

[http://arxiv.org/abs/1602.07188](http://arxiv.org/abs/1602.07188%20)

1. Ruder, M., Dosovitskiy, A., Brox, T.: *Artistic style transfer for videos.*

arXiv:1604.08610 (2016)

[http://arxiv.org/abs/1311.3715](http://arxiv.org/abs/1311.3715%20)

1. Karayev, S. et al.: *Recognizing image style.*

arXiv preprint arXiv:1311.3715 (2013)

[http://arxiv.org/abs/1311.3715](http://arxiv.org/abs/1311.3715%20)

1. Stork, D. G.: *Computer vision and computer graphics analysis of paintings and drawings: An introduction to the literature.*

In Computer Analysis of Images and Patterns, Springer (2009) [http://www.diatrope.com/stork/CAIPPlenary.pdf?origin=publication\_detail](http://www.diatrope.com/stork/CAIPPlenary.pdf?origin=publication_detail%20)

1. Hertzmann, A.: *Non-Photorealistic Rendering and the Science of Art.*

In Proc. NPAR (2010)

[http://www.dgp.toronto.edu/~hertzman/ScienceOfArt/hertzmannscienceOfArt.pdf](http://www.dgp.toronto.edu/~hertzman/ScienceOfArt/hertzmannscienceOfArt.pdf%20)

1. Rosin, P., Collomosse, J.: *Image and Video-Based Artistic Stylisation*

Springer (2013)

1. Xie, N.: *A Machine Learning Approach for Auromatic Stroke Generation in Oriental Ink Painting* (2012)
2. Litwinowicz, P.: *Processing images and video for an impressionist effect.*

In Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques. pp. 407{414. SIGGRAPH '97, ACM Press/Addison-Wesley Publishing Co. (1997)

<http://dx.doi.org/10.1145/258734.258893>

1. Tenenbaum, J. B., Freeman, W. T.: *Separating style and content with bilinear models.*

Neural computation 12, 1247–1283 (2000)

<http://www.mitpressjournals.org/doi/abs/10.1162/089976600300015349>

1. Krizhevsky, A., Sutskever, I., Hinton, G. E.: *Imagenet classification with deep convolutional neural networks.*

In Advances in neural information processing systems, 1097–1105 (2012)

[http://papers.nips.cc/paper/4824-imagenet](http://papers.nips.cc/paper/4824-imagenet%20)

1. Li, C., Wand, M.: *Combining Markov random fields and convolutional neural networks for image synthesis.*

CoRR abs/1601.04589 (2016)

[http://arxiv.org/abs/1601.04589](http://arxiv.org/abs/1601.04589%20)

1. Kyprianidis, J. E., Collomosse, J., Wang, T., Isenberg, T.: *State of the ”Art”: A Taxonomy of Artistic Stylization Techniques for Images and Video.*

Visualization and Computer Graphics, IEEE Transactions on 19, 866–885(2013)

[http://ieeexplore.ieee.org/xpls/abs\_all.jsp?arnumber=6243138](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6243138%20)

1. Simonyan, K., Zisserman, A.: *Very Deep Convolutional Networks for LargeScale Image Recognition.*

arXiv:1409.1556 (2014)

[http://arxiv.org/abs/1409.1556](http://arxiv.org/abs/1409.1556%20)

1. He, C., Fang, Y.: *Digital Image Forgery Detection Based on Extraction of GCD*

[http://en.cnki.com.cn/Article\_en/CJFDTOTAL-XTFJ201101004.htm](http://en.cnki.com.cn/Article_en/CJFDTOTAL-XTFJ201101004.htm%20)

1. Klarreich, E.: Con artist: *Scanning program can discern true art*.

News Volume 166, Issue 22, page 340, 27 November 2004. [http://onlinelibrary.wiley.com/doi/10.2307/4015852/abstract](http://onlinelibrary.wiley.com/doi/10.2307/4015852/abstract%20)

1. Farid, H., Lyu, S.: *Higher-order Wavelet Statistics and their Application to Digital Forensics*.

In IEEE Workshop on Statistical Analysis in Computer Vision (in conjunction with CVPR) (2003)

[http://www.ists.dartmouth.edu/library/35.pdf](http://www.ists.dartmouth.edu/library/35.pdf%20)

1. Jia, Y., E. Shelhamer, E.,Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: *Caffe: convolutional architecture for fast feature embedding.*

arXiv preprint arXiv:1408.5093 (2014)

<http://arxiv.org/abs/1408.5093>

1. Caffe documentation

<http://caffe.berkeleyvision.org/>

1. Torch documentation

http://torch.ch

1. Collobert, R., Kavukcuoglu, K., Farabet, C.: Torch7: *A matlab-like environment for machine learning.*

In BigLearn, NIPS Workshop (2011)

<http://ronan.collobert.com/pub/matos/2011_torch7_nipsw.pdf>

1. Seims, J.: *Putting the Artist in the Loop.*

Industrial Light & Magic (1999)

<http://dl.acm.org/citation.cfm?doid=563666.563685>

1. Yamins, D. L. K. et al.: *Performance-optimized hierarchical models predict neural responses in higher visual cortex*.

Proceedings of the National Academy of Sciences 201403112 (2014) <http://www.pnas.org/content/early/2014/05/08/1403112111>