



---

UNIVERSITÉ PARIS 8 - VINCENNES À SAINT-DENIS

**Master Informatique des Systèmes Embarqués**

**Mémoire de projet tuteuré**

**Fakhri YAHIAOUI - Roman BOURSIER**

Date de soutenance : le 09/06/2020

Tuteur – Université : Farès BELHADJ



# Résumé

A faire en dernier ...



# Remerciements

Idem ...



# Table des matières

|  |            |
|--|------------|
| <b>Résumé</b>  | <b>i</b>   |
| <b>Remerciements</b>   | <b>iii</b> |
| <b>Introduction</b>  | <b>1</b>   |
| 0.1 Contexte . . . . .   | 1          |
| 0.2 Problématique . . . . .                                      | 1          |
| <b>1 Etat de l'art</b>   | <b>5</b>   |
| 1.1 Généralité sur les GANs . . . . .                            | 5          |
| 1.2 Génération d'image conditionnels . . . . .                   | 6          |
| 1.2.1 pix2pix . . . . .  | 6          |
| 1.2.2 GauGan . . . . .   | 7          |
| 1.2.3 Le transfert de style neuronal . . . . .                   | 7          |
| 1.2.4 Cycle Gan . . . . .  | 8          |
| 1.2.5 Learning to Sketch . . . . .                               | 8          |
| 1.2.6 Photo-Sketching :Inferring Contour Drawings from Images    | 9          |
| 1.2.7 Datasets disponibles . . . . .                             | 9          |
| <b>2 Propositions de solutions</b>                               | <b>11</b>  |
| 2.1 Création du dataset en semi automatique . . . . .            | 11         |
| 2.2 Photo-Sketching : Inferring Contour Drawings from Images . . | 11         |
| 2.3 Abstraire une image via les CNN . . . . .                    | 11         |
| <b>3 Conclusion et Perspectives</b>                              | <b>13</b>  |



# Introduction

## 0.1 Contexte

Dans le cadre de notre projet de fin d'étude, nous souhaitons utiliser un modèle d'apprentissage automatique du type Deep Learning, afin de produire un moteur de rendu capable d'adopter une stylisation « type » telle que la peinture chinoise. Dans un premier temps, il s'agira de proposer un modèle d'abstraction des peintures sélectionnées comme base d'apprentissage et d'utiliser le couple « peinture originale » / « abstraction » pour l'entraînement. Par la suite, un moteur de rendu d'abstractions sera connecté au réseau profond qui produira une peinture sur la base de l'abstraction.

Le modèle généré devra d'une part adopter la stylisation retenue mais aussi interpréter l'abstraction d'origine.

## 0.2 Problématique

De nombreux travaux autour de la "translation d'image à image" [IZZE16] permettent déjà d'obtenir de bons résultats.

La figure n°1 montre qu'il est possible de générer un paysage à partir de quelques lignes. Cependant, nous sommes confronté à un problème avec la figure n°2.

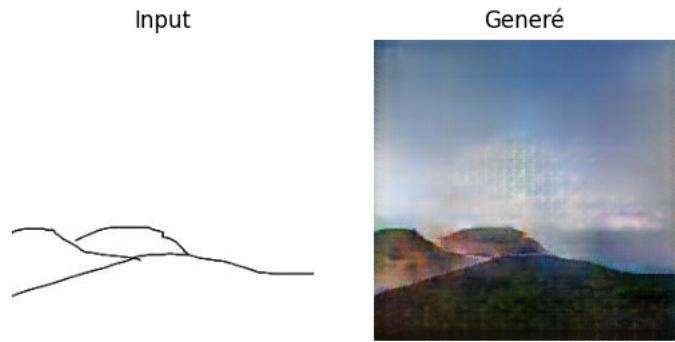


FIGURE 1 – Nous avons testé pix2pix sur un dataset composé de photos de paysages que nous avons labellisées en appliquant un filtre canny. Le résultat présenté montre le rendu une fois l'apprentissage terminé, avec un dessin fait à main levée en entrée

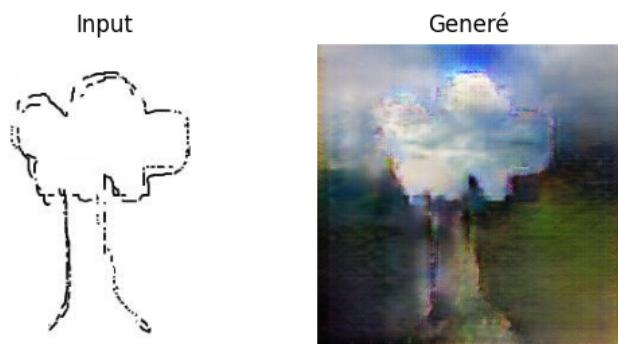


FIGURE 2 – Utilisation de pix2pix avec une abstraction d'arbre

Comme évoqué dans [SPS<sup>+</sup>18] ce type de modèle se basent sur une corrélation d'une image à l'autre, et relève d'un apprentissage supervisé. Cependant la distance entre une abstraction et une peinture est dans notre cas trop grande, ce qui nous indique peut-être que ce problème relève d'une part de "non supervisé".

Nous avons demandé à un plusieurs personnes de dessiner schématiquement un paysage composé de montagnes et d'arbres éléments courants dans une peinture de paysages chinoise :



Nous pouvons dire que ce type de dessins constitue des "abstractions", que nous souhaitons convertir en peinture chinoises. En tant qu'humain nous percevons les objets de manière abstraite et sémantique.

Nous pouvons déjà noter qu'il n'existe pas une seule façon de représenter un paysage mais plusieurs. Selon le dessinateur, les dessins possèdent des niveaux "d'informations" plus ou moins élevé, c'est à dire plus ou moins "abstraits". La figure n°1 nous indique des arbres, mais nous ne savons rien de la forme potentiel que ceux-ci pourraient avoir (bien que nous pouvons dire qu'il s'agit plutôt d'un sapin que d'un pin). Tandis que sur la figure n°2 nous avons déjà plus de détails.



# Chapitre 1

## Etat de l'art

Nous présentons dans un premier temps l'architecture GANs, qui est au cœur de notre problématique. Nous présenterons également différents modèles existants sur lesquels nous avons testé nos données.

Enfin, nous évoquerons, de manière plus succincte, des recherches liées à notre sujet mais n'utilisant pas les réseaux de neurones afin de pouvoir identifier les avantages et inconvénients des deux mondes.

### 1.1 Généralité sur les GANs

Les réseaux adverses génératifs sont une classe d'algorithmes d'apprentissage non-supervisé. Ces algorithmes ont été introduits dès 2014 par Goodfellow [GPAM<sup>+</sup>14] et permettent de générer des images avec un fort degré de réalisme. La structure des GANs se compose de deux parties, le générateur et le discriminateur. Le générateur génère des données et les soumet au discriminateur dont le but est d'évaluer le degré de crédibilité de la donnée.

$G(z, 01)$  se charge de "mapper" le bruit  $z$  vers l'espace désiré  $D(x, 02)$  retourne et la probabilité dans l'intervalle  $[0, 1]$  que  $x$  vient du dataset original. Enfin  $O_i$  représente les poids définis par chacun des modèles.

Les poids sont donc mis à jours pour :

1 - Maximiser la probabilité que les données  $x$ , soient classifiées comme appartenant au dataset d'origine. Soit maximiser la fonction de perte  $D(x)$

2 - Minimiser la probabilité que de fausses images appartiennent au dataset d'origine. Soit minimiser la fonction de perte  $D(G(z))$

Pour résumer c'est une sorte de jeu, dans lequel le générateur essaie de maximiser la probabilité que sa sortie soit reconnue comme réel, tandis que le discriminateur essaie de minimiser la même valeur.

Mathématiquement La fonction de perte peut-être écrite ainsi :

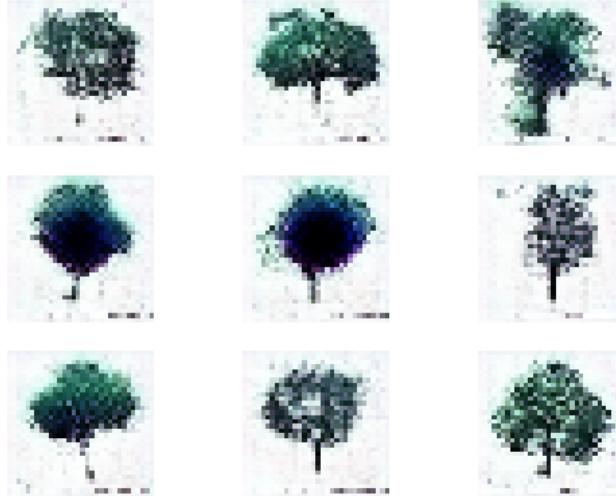


FIGURE 1.1 – Test de génération d’arbres utilisant l’architecture DCGAN

Les Gans conditionnels que nous abordons plus en détails plus loin, apprennent à partir d’une observation  $x$  et d’un vecteur de bruit aléatoire  $z$ , vers  $y :G : x, z \beta y$ .

Il existe aujourd’hui une très grande variété de travaux de recherches implémentants les Gans (CGAN, Bayesian GAN, GYCLE GAN ...). Nous présentons ici des solutions existantes permettant de répondre partiellement à notre problématique.

## 1.2 Génération d’image conditionnels

Grâce aux conditionnals GANs il est aujourd’hui possibles de générer des images réalistes basées sur des labels de classes, des textes ou des images.

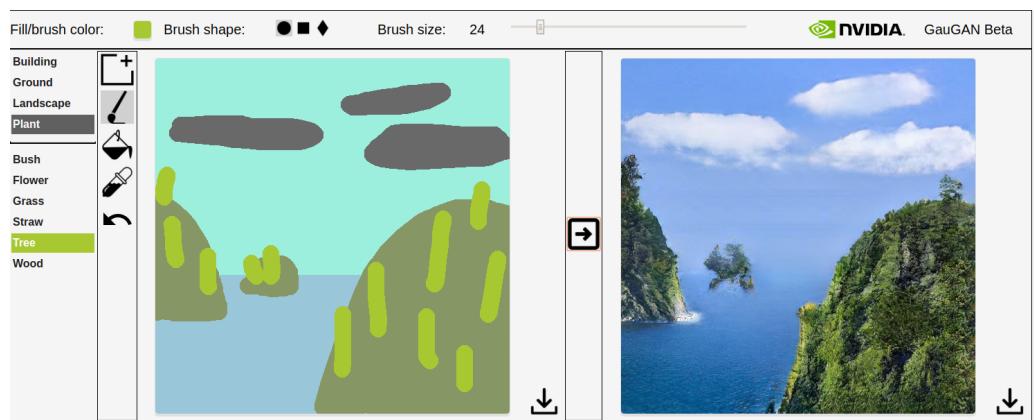
### 1.2.1 pix2pix

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou et Alexei A. Efros proposent en 2014 une nouvelle approche intitulée Image-to-Image Translation with Conditional Adversarial Networks.[GEB15] Leur contribution principale fut de montrer en quoi les cGans peuvent permettre de résoudre efficacement les problèmes de translation d’image et ainsi de proposer un framework pouvant être appliquéd à n’importe quel domaine.

### 1.2.2 GauGAN

Nvidia [PLWZ19] présente en 2019 une forme spécifique de génération d'image basé sur des "semantic segmentation mask".

Selon les auteurs, les couches de normalizations ont tendance à faire perdre l'information contenu dans les masques sémantiques d'entrées car ils ne dépendent pas de données externe. Ces derniers introduisent donc la notion de "normalization conditionnel" ou "adaptative", permettant notamment de prendre en compte les informations spatiales.



### 1.2.3 Le transfert de style neuronal

Cette technique fut décrite par Leon A. Gatys dans son article "A Neural Algorithm of Artistic Style". [GEB15]

Celle-ci utilise trois images, la première est l'image d'entrée (bruit), la seconde représente le style de référence (comme une peinture par exemple), la dernière correspond à l'image que l'on souhaite transformer.

Le principe consiste à définir deux fonctions de distance. La première décrit comment le contenu de deux images sont différentes, et la seconde représente la différence de style. On obtient alors deux images, une pour le style souhaité, une pour le contenu.

L'objectif est de transformer l'image d'entrée en minimisant la distance avec l'image de contenu et avec l'image de style. On obtiens alors par rétro-propagation, une image qui correspond au contenu de l'image d'origine et au style souhaitée.



Cela nécessite la création d'un réseau convolutionnel, en effet le modèle doit être en mesure de capturer les invariances et de définir les caractéristiques de l'images (chats vs chien) afin d'en obtenir une compréhension générale.

L'avantage de cette technique est quelle nécessite pas de dataset à proprement parlé, seulement deux images en guise d'input

#### 1.2.4 Cycle Gan

#### 1.2.5 Learning to Sketch

Ce travail [SPS<sup>+</sup>18] est important pour nous car il présente une méthode permettant de transformer une photos en croquis, en essayant d'imiter la façon de faire d'un humain. Leur model est capable de réaliser le croquis séquentiellement, c'est à dire ligne par ligne.

Les auteurs proposent de résoudre le problème "des styles subjectifs et variés de dessin humain" en utilisant un modèle hybride supervisé et non-supervisé. L'objectif étant de palier au "signal de supervision faible et bruyant" induit par l'écart important entre un croquis sa photo correspondante.

L'architecture est décomposé en 4 sous-modèle contenant chacuns leur propres sous-réseaux d'encodeurs et de décodeur :

- Une premier "supervisé" qui traduit une photo en croquis  $D(E(photo)) \rightarrow sketch$
- Un second supervisé aussi qui renvoie un croquis au domaine photo  $(D(E(sketch))) \rightarrow photo$

- Un troisième non-supervisé pour reconstruire la photo  $D(E(photo)) \beta photo$
- Le dernier non-supervisé pour reconstruire le croquis  $D(E(sketch)) \beta sketch$

Le processus complet d'une photo vers un croquis serait donc :  $x \beta Ep(x) \beta Ds(Ep(x)) \beta Es(Ds(Ep(x)))$

### 1.2.6 Photo-Sketching :Inferring Contour Drawings from Images

"Les bords, les limites et les contours sont des sujets importants dans l'étude de la vision par ordinateur" [LLM<sup>+</sup>19]

L'article nous présente une technique pour générer un dessin de contour à partir d'une photo. Les "edges detector" traditionnels comme Canny, captent uniquement les signaux de haute fréquence dans l'image sans la comprendre.

Les auteurs ont collecté un dataset de 5000 pairs dessin/photos, créer la plateforme de crowdsourcing "Amazon Mechanical Turk". En effet aucun dataset de croquis actuellement connu ne répondait à la problématique (nombre d'éléments dans l'image, limites internes manquantes, le contenu non reconnaissable, les zones ombrées vides etc ..). Au final il s'agit d'un problème d'image-to-image translation.

Un élément intéressant de ce modèle la possibilité d'avoir plusieurs labels différents pour la même image, car comme vu en ... Le même objet peut être interprété de plusieurs manière différentes.

### 1.2.7 Datasets disponibles

Il est évident qu'il n'existe pas à l'heure actuel de dataset de paires croquis/peinture chinoise.

#### Sketch

Un des principaux problèmes de la recherche dans le domaine de la génération de croquis est le manque de dataset disponible. Aujourd'hui les plus grosses bases de données que nous avons trouvées sont TU-Berlin [EHA12], Sketchy [SBHH16] et Quickdraw. BSDS500

L'article (<https://arxiv.org/pdf/1901.00542.pdf>) est intéressant car il nous offre une classification du niveau d'abstraction des croquis. Un paramètre à prendre en compte également est le fait qu'il existe plus d'éléments dans le croquis.

Photo/croquis pair QMUL-Shoe-Chair-V2 dataset [46], qui contient plus de 8000 exemples.

### **Peinture chinoises**

"La peinture chinoise désigne toute forme de peinture originaire de Chine ou pratiquée en Chine ou par des artistes chinois hors de Chine." [] Il existe beaucoup de style courants et genres différentes (paysage, peinture narratives, faunes, flore, personnage). Nous avons aujourd'hui "scrapé" une collection de plus de 800 peintures. noté qu'un des problèmes auquel nous allons faire face et la multiplicité des homothéties (Kakemono, paysage). Comme évoqué dans [], Les RCNN n'acceptent des images de tailles différents, il faudra donc trouver un moyen de trouver un bon compromis dans le redimensionnement de notre dataset.

# Chapitre 2

## Propositions de solutions

Voici les solutions envisagées :

### 2.1 Cr eation du dataset en semi automatique

En utilisant le logiciel LabelBox, il est possible de labeliser une image, en cr eant des s emantique layout[1]. Nous avons d j  un collection de plus de 1000 images de peinture chinoises, les labelis s nous prendrait environ 2 jours hommes  raison d'une minute par image. Une fois cette tape r aliss e, nous pourrions g n rer une ou plusieurs abstractions pour chaque peintures, en utilisant le dataset "quick draw" entre autres. Outre l'inconv nient du temps pass , cette solution n'est pas g n ralisable  autre chose que notre probl matique.

### 2.2 Photo-Sketching : Inferring Contour Drawings from Images

### 2.3 Abstraire une image via les CNN

Nous savons qu'un CNN est capable de d etecter des caract ristiques d'une image

Comme le prouve le travail de <https://github.com/dribnet/perceptionengines>, il est peut- tre possible de g n rer un repr sentation abstraite en utilisant cette propri t  des CNN.



## Chapitre 3

### Conclusion et Perspectives



# Bibliographie

- [EHA12] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(4) :44 :1–44 :10, 2012.
- [GEB15] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015.
- [GPAM<sup>+</sup>14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.
- [IZZE16] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016.
- [LLM<sup>+</sup>19] Mengtian Li, Zhe Lin, Radomír Mech, Ersin Yumer, and Deva Ramanan. Photo-sketching : Inferring contour drawings from images. *CoRR*, abs/1901.00542, 2019.
- [PLWZ19] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. *CoRR*, abs/1903.07291, 2019.
- [SBHH16] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database : Learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (proceedings of SIGGRAPH)*, 2016.
- [SPS<sup>+</sup>18] Jifei Song, Kaiyue Pang, Yi-Zhe Song, Tao Xiang, and Timothy M. Hospedales. Learning to sketch with shortcut cycle consistency. *CoRR*, abs/1805.00247, 2018.