

Compte Rendu TD2 : GAN Lab et StyleGAN3

Louis Thin

7 décembre 2025

1 Introduction

Ce travail porte sur deux approches permettant d'étudier les modèles GAN (Generative Adversarial Networks) :

- **GAN Lab**, un outil interactif facilitant la compréhension du fonctionnement interne d'un GAN ;
- **StyleGAN3**, un modèle avancé de génération d'images de haute qualité.

L'idée générale du TD était d'obtenir à la fois une compréhension théorique et empirique du fonctionnement d'un GAN. GAN Lab fournit une plateforme visuelle simple où l'on peut observer l'évolution des réseaux et les interactions entre générateur et discriminateur. À l'inverse, StyleGAN3 représente l'état de l'art actuel : son utilisation permet d'expérimenter la complexité pratique qu'impliquent les modèles modernes, notamment en termes de ressources matérielles.

2 GAN Lab

2.1 Compréhension du modèle

L'utilisation de GAN Lab permet d'étudier le fonctionnement d'un GAN dans un environnement réduit où les données sont représentées par des nuages de points en 2D. Ce cadre simplifié permet de rendre visibles les forces et faiblesses de l'apprentissage adversarial, ce qui est très difficile à percevoir dans des générateurs d'images de grande taille. Grâce à cette interface, on peut suivre en temps réel les ajustements du générateur et la réaction du discriminateur, ce qui rend le processus d'entraînement beaucoup plus intuitif.

2.1.1 Architecture d'un GAN

Pour bien comprendre ce que montre GAN Lab, il est essentiel de rappeler la structure d'un GAN. Un GAN est composé de deux réseaux de neurones :

- **Générateur** : transforme un vecteur de bruit en échantillons synthétiques ;
- **Discriminateur** : tente de distinguer les données réelles des données générées.

Ces deux réseaux sont engagés dans un jeu à somme nulle : lorsque le générateur s'améliore, le discriminateur doit s'adapter pour continuer à distinguer vrai du faux. Cette dynamique produit une forme d'équilibre qui, si elle est atteinte, correspond à un générateur capable de reproduire fidèlement la distribution cible.

2.1.2 Visualisations dans GAN Lab

GAN Lab propose plusieurs visualisations destinées à rendre cet apprentissage transparent :

1. **Model Overview Graph** : permet de visualiser la structure du GAN et le flux des données entre ses composants ;
2. **Layered Distributions View** : superposition graphique des distributions réelle, générée et évaluée par le discriminateur.

Ces éléments permettent de comprendre non seulement l'état actuel du modèle, mais aussi la manière dont les distributions évoluent au cours du temps.

2.1.3 Éléments visuels clés

L'outil propose aussi plusieurs représentations plus détaillées :

- **Manifold** : montre comment les points issus du bruit sont transformés ;
- **Heatmap du discriminateur** : fournit une vision spatiale de la frontière de décision du réseau ;
- **Gradients** : indiquent dans quelle direction les échantillons devraient se déplacer pour améliorer l'apprentissage.

Ces visualisations rendent explicite l'effet des mises à jour et aident à comprendre pourquoi un GAN peut parfois osciller ou s'effondrer (mode collapse).

2.2 Étude de différents hyperparamètres

L'expérimentation dans GAN Lab consiste à modifier des hyperparamètres clés pour observer comment ils influencent la stabilité et la convergence du modèle. Dans un GAN, l'équilibre entre les deux réseaux est fragile, et la moindre variation peut entraîner des comportements très différents.

2.2.1 Taux d'apprentissage du Générateur

Valeurs testées : 0.0001, 0.001, 0.01

Les résultats montrent que :

- **0.0001** : progression très lente vers la distribution cible ;
- **0.001** : apprentissage équilibré et relativement stable ;
- **0.01** : apprentissage rapide mais instable, pouvant conduire au *mode collapse*.

Ces observations illustrent l'importance du taux d'apprentissage : si le générateur progresse trop vite, il ne tient plus compte du retour fourni par le discriminateur.

2.2.2 Taux d'apprentissage du Discriminateur

Valeurs testées : 0.0001, 0.001, 0.01

De manière similaire, des modifications du taux du discriminateur entraînent :

- **0.0001** : un discriminateur trop lent ;
- **0.001** : une progression équilibrée ;
- **0.01** : un réseau trop performant qui empêche le générateur de s'améliorer.

Un discriminateur trop puissant ne fournit plus un gradient exploitable : il rejette immédiatement les échantillons artificiels sans nuance.

2.2.3 Nombre d’itérations du Discriminateur

Tester 1, 2, 3 ou 5 itérations du discriminateur par mise à jour du générateur permet d’observer l’importance du rythme d’entraînement :

- **1** : le discriminateur n’a pas assez de temps pour se calibrer ;
- **2–3** : équilibre satisfaisant entre les deux réseaux ;
- **5** : le générateur est écrasé par un discriminateur trop performant.

Ce paramètre est souvent négligé, mais en pratique il est crucial pour éviter les dérives du modèle.

2.2.4 Dimension du bruit

L’espace latent joue également un rôle essentiel. Les dimensions testées (10 à 200) montrent que :

- **10–20** : diversité limitée ;
- **50–100** : bonne couverture de la distribution réelle ;
- **200+** : complexité accrue sans gain significatif.

Un espace latent trop petit limite la variété des échantillons générés, tandis qu’un espace trop grand rend l’apprentissage plus difficile.

2.2.5 Distribution de bruit

Les tests entre bruit uniforme et gaussien confirment que la distribution gaussienne se prête mieux aux interpolations et à la géométrie naturelle de l’espace latent utilisé par la plupart des GAN modernes.

3 StyleGAN3

3.1 Instanciation du modèle

La seconde partie du TD consiste à manipuler StyleGAN3, un modèle très avancé capable de générer des visages photoréalistes. Pour cela, plusieurs étapes sont nécessaires :

1. téléchargement du modèle ;
2. chargement du générateur `G_ema` ;
3. utilisation du GPU quand c’est possible ;
4. mise en mode évaluation.

Chaque étape est assez lourde en termes de ressources, ce qui met en évidence la différence entre l’approche pédagogique de GAN Lab et l’utilisation concrète d’un GAN moderne.

3.2 Inférences

3.2.1 Inference 1 — Réussite

La première inférence avec la seed 931185 montre que le modèle est correctement chargé et utilisable.



FIGURE 1 – Image générée par StyleGAN3 (seed = 931185)

L'image générée est d'une qualité remarquable, témoignant de la puissance du modèle.

3.2.2 Erreurs lors des inférences suivantes

Cependant, dès la deuxième inférence, une erreur critique apparaît :

```
DefaultCPUAllocator: not enough memory: you tried to allocate  
2990304576 bytes.
```

Cette erreur empêche toute génération supplémentaire.

3.2.3 Analyse des causes probables

Plusieurs hypothèses peuvent expliquer cette panne :

1. **RAM insuffisante** : exécuter StyleGAN3 en 1024×1024 sur CPU nécessite plusieurs gigaoctets de mémoire ;
2. **Absence de GPU** : l'intégralité du calcul repose sur la mémoire centrale ;
3. **Mémoire non libérée** entre deux inférences ;
4. **Taille du modèle** particulièrement élevée ;
5. **Couches convolutionnelles coûteuses** en haute résolution.

Cela illustre clairement que les GAN modernes sont très gourmands en ressources, bien plus que les exemples manipulés dans GAN Lab.

3.2.4 Solutions envisageables

Pour contourner le problème, plusieurs pistes sont possibles :

- utiliser une carte graphique avec plus de VRAM ;
- réduire la résolution à 512×512 ;
- exécuter une seule inférence par lancement du script ;
- nettoyer explicitement la mémoire entre deux inférences ;
- utiliser une machine dotée de davantage de RAM.

3.3 Analyse des résultats

La première génération démontre la qualité exceptionnelle du modèle, mais les limites matérielles rencontrées rappellent que StyleGAN3 vise des environnements GPU haut de gamme. Le contraste avec GAN Lab est particulièrement instructif : l'un est conçu pour l'apprentissage interactif, tandis que l'autre incarne le niveau de complexité requis pour la production d'images réalistes.