

République algérienne démocratique et populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université du 8 mai 1945-Guelma-
Faculté des mathématiques, de l'informatique et des sciences de la matière
Département d'informatique



Mémoire de fin d'étude Master

Filière : Informatique

Option:

Science et technologie de l'information et de la communication.

Thème

Génération d'image faciale
synthétique à base de points
caractéristiques.

Présenté par : MAIZI Khawla

<u>Membre de jury</u>	<u>Qualité</u>
Dr. FERKOUS Chokri	Président
Mme. BORDJIBA Yamina	Encadrante
Pr. KOUAHLA Mohammed Nadjib	Examineur

Juin 2023

Résumé

Les avatars et les dessins animés sont largement utilisés dans divers domaines des technologies émergentes, de la réalité augmentée aux robots sociaux. Ces avatars aident à percevoir l'information de manière naturelle et jouent un rôle clé dans la transmission de divers signaux émotionnels.

L'objectif de notre projet est de concevoir un système intelligent capable de transférer les caractéristiques d'un visage réel à un visage de dessin animé, permettant à cet avatar d'être utilisé dans des contextes variés.

Notre approche est basée sur les récentes avancées en matière d'apprentissage profond, en particulier les réseaux génératifs adversaires et les autoencodeurs. Le système proposé commence par extraire des points caractéristiques de chaque image, qui servent ensuite d'entrée à l'autoencodeur. Le module d'extraction des points caractéristiques est en fait le codeur de l'autoencodeur, qui a été entraîné sur la base de données FER-G-DB et a atteint une précision de 78%.

La sortie de ce module est ensuite utilisée comme entrée du module de génération de visages de dessins animés. Pour réaliser cette génération, nous avons utilisé le générateur pré-entraîné StyleGAN3, via l'apprentissage par transfert sur la base de données FER-G-DB.

L'utilisation d'autoencodeurs et du générateur StyleGAN3 représente une approche intéressante pour atteindre l'objectif de ce projet, et les résultats obtenus jusqu'à présent sont encourageants, avec un potentiel d'amélioration significatif grâce à un entraînement plus avancé.

Mots-clés : génération d'image, autoencodeur, StyleGAN3, vecteur latent, Cartoon(dessin animé), avatar.

Abstract

Avatars and cartoons are widely used in various areas of emerging technology, from augmented reality to social robots. These avatars help to perceive information in a natural way and play a key role in transmitting various emotional signals.

The aim of our project is to design an intelligent system capable of transferring the characteristics of a real face to a cartoon face, enabling this avatar to be used in a variety of contexts.

Our approach is based on recent advances in deep learning, in particular adversarial generative networks and autoencoders. The proposed system starts by extracting minutiae from each image, which are then used as input to the autoencoder. The minutiae extraction module is in fact the encoder of the autoencoder, which has been trained on the FER-G-DB database and has achieved 78% accuracy.

The output of this module is then used as input for the cartoon generation module. To carry out this generation, we used the StyleGAN3 generator, previously trained by knowledge transfer from the FER-G-DB database.

The use of autoencoders and the StyleGAN3 generator represents an interesting approach to achieving the objective of this project, and the results obtained so far are encouraging, with significant potential for improvement through more advanced training.

Keywords : image generation, autoencoder, StyleGAN3, latent vector, cartoon, avatar.

ملخص:

تُستخدم الصور الرمزية والرسوم المتحركة على نطاق واسع في مختلف مجالات التكنولوجيا الناشئة، من الواقع المعزز إلى الروبوتات الاجتماعية. تساعد هذه الصور الرمزية في إدراك المعلومات بطريقة طبيعية وتلعب دورًا رئيسيًا في نقل الإشارات العاطفية المختلفة.

الهدف من مشروعنا هو تصميم نظام ذكي قادر على نقل خصائص الوجه الحقيقي إلى وجه كرتوني، مما يتيح استخدام هذه الصورة الرمزية في سياقات متنوعة.

يعتمد نهجنا على التطورات الحديثة في التعلم العميق، ولا سيما الشبكات التوليدية العنصرية وأجهزة التشفير التلقائي. يبدأ النظام المقترح باستخراج التفاصيل الدقيقة من كل صورة، والتي يتم استخدامها بعد ذلك كمدخل إلى وحدة التشفير التلقائي. وحدة استخراج التفاصيل الدقيقة هي في الواقع المشفر لجهاز التشفير التلقائي، والذي تم تدريبه على قاعدة بيانات FERG-DB وحقق دقة تصل إلى 78٪.

ثم يتم استخدام مخرجات هذه الوحدة كمدخلات لوحدة توليد الرسوم المتحركة. لتنفيذ هذا التوليد، استخدمنا مولد StyleGAN3 ، الذي تم تدريبه مسبقًا عن طريق نقل المعرفة من قاعدة بيانات FERG-DB.

يمثل استخدام أجهزة التشفير التلقائي ومولد StyleGAN3 أسلوبًا مثيرًا للاهتمام لتحقيق هدف هذا المشروع، والنتائج التي تم الحصول عليها حتى الآن مشجعة، مع إمكانات كبيرة للتحسين من خلال المزيد من التدريب المتقدم.

الكلمات الرئيسية: توليد الصور، التشفير التلقائي، StyleGAN3، ناقل كامن، رسوم متحركة، أفاتار.

Remerciements

Je tiens à exprimer mes sincères remerciements à toutes les personnes qui ont contribué à la réalisation de ma mémoire de fin d'études.

Tout d'abord, je tiens à exprimer ma sincère gratitude envers Madame BORD-JIBA Yamina, mon encadrante, pour ses précieux conseils et son soutien tout au long de cette période de travail. Sa guidance précieuse m'a permis de progresser et d'atteindre mes objectifs.

Je souhaite également exprimer ma gratitude envers les membres de mon jury, pour avoir consacré leur temps et leur expertise à évaluer mon travail. Leurs remarques et suggestions constructives ont grandement contribué à l'amélioration de ma recherche.

Je tiens à remercier mes enseignants et mes professeurs qui m'ont transmis leurs connaissances et leur passion pour mon domaine d'étude. Leurs enseignements ont été une source d'inspiration et ont nourri ma réflexion tout au long de mon parcours académique.

Un grand merci à mes amis, ma très chère amie "Ines" et ma famille pour leur soutien inconditionnel et leurs encouragements tout au long de cette aventure. Leur présence et leurs mots d'encouragement ont été d'une importance capitale pour moi.

Enfin, je tiens à exprimer ma reconnaissance envers toutes les personnes qui ont participé de près ou de loin à la réalisation de ce projet. Leur contribution a été précieuse et a contribué à faire de cette expérience une réussite.

Table des matières

Liste des figures	4
Liste des tableaux	6
Introduction générale	7
1 Réseau adversarial génératif (GAN)	10
1.1 Introduction	10
1.2 Apprentissage Automatique	10
1.2.1 Définition	10
1.2.2 Type d'apprentissage	11
1.3 Apprentissage en profondeur	12
1.4 Les Réseau de neurones	13
1.4.1 Le perceptron multicouche	14
1.4.2 Apprendre avec la rétro-propagation :	15
1.4.3 Les Architecture de DL	17
1.5 Réseaux de neurones non supervisés :	22
1.5.1 Réseau adversarial génératif (GAN)	22
1.5.2 Les Types des GANs :	24
.....	24
1.5.3 Les domaines d'application des GANs	24
1.5.4 Auto-Encoder (AE) :	25
1.6 StyleGan	27
1.6.1 Les Variantes de Stylegan	27
1.7 Conclusion	31
2 AvatarGénération de visages de dessins animés : Avatar ou cartoon.	32
2.1 Introduction	32
2.2 Définition	32

2.3	Type d'avatar	33
2.3.1	Avatar tridimensionnelle	33
2.3.2	Animal Avatars :	33
2.3.3	Avatar de jeu :	34
2.3.4	Avatar humain :	34
2.3.5	Cartoon Avatar :	35
2.4	Domaines d'utilisation des avatars :	36
2.4.1	Dans la réalité virtuelle :	36
2.4.2	Les jeux Video :	36
2.4.3	Réseaux sociaux :	37
2.4.4	Les systèmes d'apprentissage en ligne	37
2.5	Les bases de données d'avatars :	37
2.5.1	FERG (Facial Expression Research Group Database) :	37
2.5.2	MetFaces dataset :	38
2.5.3	Disney Dataset	39
2.5.4	CartoonSet Dataset	39
2.6	Travaux connexes	40
2.6.1	Article 1 : "3D Cartoon Face Generation with Controllable Ex- pressions from a Single GAN Image[46] "	40
2.6.2	Article 2 : "XGAN : Unsupervised Image-to-Image Translation for Many-to-Many Mappings[34] "	40
2.6.3	Article 3 : "Cartoon Face to Human Face Translation using Contour Loss based CycleGANs[42]"	41
2.6.4	Article 4 : "Resolution Dependent GAN Interpolation for Control- lable Image Synthesis Between Domains [31]"	41
2.6.5	Article 5 : "Landmark-guided deformation transfer of template facial expressions for automatic generation of avatar blend- shapes [30]"	42
2.6.6	Article 6 : " normalsize STYLEGAN2 for cartoon face genera- tion [6]"	42
2.6.7	Article 7 : "Automatic Unpaired Human Character Facial Re- targeting Based on Deep learning Approach [19]"	43
2.6.8	Article 8 : "Normalized Avatar Synthesis Using StyleGAN and Perceptual Refinement [22]"	43
2.7	Conclusion	44
3	Conception	45
3.1	Introduction	45

3.2	Objectifs	45
3.3	Architecture du système proposé	46
3.4	Detection de visage et Extraction les points caractéristiques	47
3.5	Module de mappage(Mapping module)	49
3.6	Le module de génération d'image(Synthesis module)	51
3.7	Conclusion	52
4	Implémentation	53
4.1	Introduction	53
4.2	Environnement de développement	53
4.2.1	Google Colaboratory	53
4.3	Langage de programmation et bibliothèques utilisées	54
4.3.1	Python	54
4.4	Les bibliothèques utilisées	54
4.4.1	TensorFlow	54
4.4.2	Keras	55
4.4.3	Numpy	55
4.4.4	Matplotlib	55
4.4.5	Gradio	56
4.5	Implémentation et apprentissage	56
4.5.1	Bases de données utilisées	56
4.5.2	Implémentation de l'autoencodeur	57
4.5.3	Implémentation du générateur	57
4.6	Résultats et discussions	58
4.7	Quelques résultats	60
4.8	Conclusion	61
	Conclusion générale	62
	Bibliographie	64

Table des figures

1.1	Flux d'apprentissage automatique[8].	11
1.2	La relation entre l'intelligence artificielle, le ML et le deep Learning. .	12
1.3	un neurone biologique.	13
1.4	un neurone artificiel.[45]	14
1.5	L'architecture d'un perceptron multicouche.[45]	14
1.6	L'algorithme de Descente de gradient .[33]	16
1.7	Les techniques de DL.	17
1.8	Schéma des réseaux de neurones convolutifs	18
1.9	Une opération de convolution.[9]	19
1.10	Une opération de Max-pooling.[9]	20
1.11	Tracé de la fonction d'activation de ReLU.	21
1.12	Exemple du Dropout[43].	22
1.13	Les type des Gans[13].	22
1.14	Exemples de visages humains générés à partir de la structure propo- sée[13].	23
1.15	Architecture classique d'un GAN.	24
1.16	La structure d'un auto-encodeur.[9]	25
1.17	L'architecture du Stylegan.[24]	28
1.18	L'architecture du Stylegan2[24].	29
1.19	L'architecture du Stylegan2-Ada.[16]	30
1.20	L'architecture du Stylegan3[25].	31
2.1	3D avatar.	33
2.2	Animal Avatar.	34
2.3	Avatar Human[49].	35
2.4	Avatar Cartoon[42].	36
2.5	Quatre échantillons de sujets présentant sept expressions (neutre, co- lère, dégoût, peur, bonheur, tristesse et surprise)[5].	38
2.6	Exemple de la base MetFaces.	38

2.7	Exemples de Disney Dataset [31].	39
2.8	Exemple de la base CartoonSet	40
3.1	L'objectif de l'application	46
3.2	Architecture générale de l'application	47
3.3	Les 68 points caractéristiques détecter par Dlib[39].	48
3.4	la première étape du notre processus.	48
3.5	Le module de mappage	49
3.6	Architecture détaillée de l'autoencoder	50
3.7	la troisième étape du notre processus	51
3.8	L'architecture du Stylegan3[25].	52
4.1	Interface de Colab	54
4.2	Nombre d'images par émotions de caractère aia de la base de données FERG	56
4.3	Les courbes de précision et de perte de l'expérimentation 1	59
4.4	Les courbes de précision et de perte de l'expérimentation 2	59
4.5	Les courbes de précision et de perte de l'expérimentation 5	59
4.6	Quelques exemples de génération de cartoon.	60

Liste des tableaux

4.1	Les paramètres d'apprentissage de l'autoencodeur	57
4.2	Les Résultats d'évaluation réalisées	58

Introduction générale

La génération d'images faciales synthétiques est un domaine de recherche en pleine expansion. Dans ce domaine, la traduction d'image à image, qui vise à apprendre la correspondance entre une image d'entrée et une image de sortie, a fait l'objet d'études récentes qui ont connu un succès remarquable. Ces techniques trouvent leur utilité dans divers domaines, tels que les jeux vidéo, la réalité virtuelle et la médecine.

De nos jours, les avatars jouent un rôle crucial dans la société. Ils permettent aux gens de marquer leur identité en ligne et de communiquer avec d'autres utilisateurs sur les plateformes numériques, ce qui ne doit pas être sous-estimé.

Les progrès de l'intelligence artificielle ont été considérables ces dernières années, offrant aux chercheurs et aux développeurs des modèles sophistiqués pour la génération d'images de haute qualité. Ces progrès ont conduit à l'émergence d'outils puissants, notamment les générateurs adversaires génératifs (GAN), qui produisent des images convaincantes et très réalistes. En particulier, StyleGAN est l'un des modèles les plus répandus dans le domaine de la génération d'images de visages, atteignant ainsi des résultats de pointe en termes de fidélité et de qualité visuelles. En même temps, ce modèle possède des propriétés de démêlage de l'espace latent du StyleGAN, ce qui permet de contrôler divers facteurs de variation dans l'image du visage à l'aide des variables latentes.

Ce mémoire se focalise sur la problématique du transfert d'une image de visage vers un dessin animé (cartoon), en préservant les caractéristiques essentielles du visage original. Notre objectif principal consiste à concevoir et implémenter une application de génération de dessins animés en utilisant des modèles d'apprentissage profond. Nous combinons un auto-encodeur pour l'extraction des caractéristiques du visage d'entrée, puis le générateur du StyleGAN3 pour la création du visage du cartoon. Pour entraîner nos modèles, nous utilisons l'ensemble de données FER-G-DB

et évaluons leur performance en utilisant les ensembles de données CK+ et celeba.

Notre mémoire est structuré en quatre chapitres qui sont présentés brièvement comme suit :

Chapitre 1 : Réseau adversarial génératif (GAN)

Dans ce chapitre, nous aborderons différentes notions liées au deep learning et aux réseaux génératifs adversaires, en mettant l'accent sur les principaux types de modèles qui ont été largement étudiés en littérature.

Chapitre 2 : Génération de visages de dessins animés : Avatar ou cartoon.

Dans ce chapitre, une attention particulière est accordée aux différentes catégories d'avatars et à leurs domaines d'application, ainsi qu'à la présentation des bases de données d'avatars disponibles actuellement. En outre, quelques études récentes axées sur la création d'avatars sont présentées pour illustrer les progrès réalisés dans ce domaine.

Chapitre 3 : Conception

Ce chapitre offre un aperçu approfondi de la conception de notre projet, en détaillant l'architecture et la structure globale de notre projet, tout en mettant l'accent sur les choix effectués tout au long du processus de développement. Nous décrivons également les différentes fonctionnalités des composants clés de notre application, en expliquant comment ils interagissent les uns avec les autres pour atteindre les objectifs fixés.

Chapitre 4 : Implémentation

Ce chapitre d'implémentation fournit une vue détaillée de la mise en œuvre de notre projet. Il décrit tout d'abord l'environnement de développement ainsi que les langages et les bibliothèques utilisés. Ensuite, les étapes de la mise en œuvre de chaque composant du système proposé sont détaillées. Les tests et validations effectués pour s'assurer du bon fonctionnement de l'application sont également abordés. Enfin, les résultats obtenus sont analysés et les performances du système sont éva-

luées par rapport aux objectifs initialement fixés. nous avons présenté et interprété les différents résultats obtenus.

Pour conclure ce mémoire, nous présentons un récapitulatif global des résultats obtenus et des principales contributions de notre étude. En outre, nous formulons des suggestions et des recommandations pour les travaux futurs.

Chapitre 1

Réseau adversarial génératif (GAN)

1.1 Introduction

L'apprentissage en profondeur (DL) est une branche de l'apprentissage automatique (ML) et de l'intelligence artificielle (IA) qui est a connu un regain d'intérêt sans précédent en raison d'avancées technologiques majeures, notamment dans le domaine de l'apprentissage automatique, qui étend les capacités des ordinateurs et améliore leurs performances dans de nombreux domaines (traitement du langage, compréhension de la parole, reconnaissance d'images, robots, etc.). Ces avancées ouvrent de grandes perspectives en matière d'innovation technologique et d'automatisation de l'environnement de travail[51].

1.2 Apprentissage Automatique

1.2.1 Définition

L'apprentissage automatique, également connu sous le nom de "Machine Learning", est un domaine de recherche en intelligence artificielle qui vise à extraire des connaissances pertinentes à partir de grandes quantités de données. Cette approche repose sur l'utilisation d'algorithmes d'apprentissage automatique qui permettent de modéliser le processus d'apprentissage. Ces modèles peuvent être utilisés pour prédire la classe ou le groupe auquel appartiendront de nouvelles instances de données. En somme, l'objectif de l'apprentissage automatique est de fournir des connaissances utiles à partir de données en utilisant des algorithmes sophistiqués.[9] comme le montre dans la figure 1.1 .

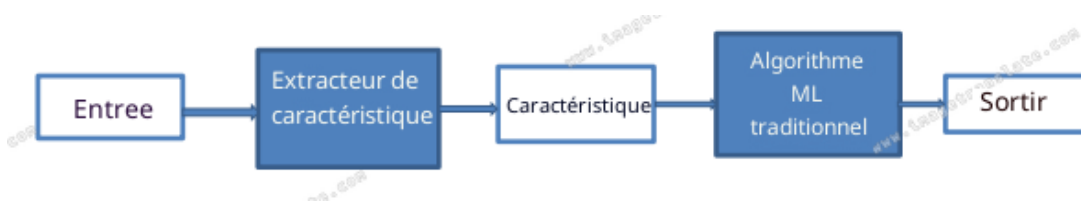


FIGURE 1.1 – Flux d'apprentissage automatique[8].

1.2.2 Type d'apprentissage

1. Apprentissage supervisé : L'objectif de l'apprentissage supervisé est d'utiliser que des données étiquetées pour apprendre. Il existe principalement deux types d'algorithmes d'apprentissage supervisé : les algorithmes de régression, qui peuvent convertir l'espace d'entrée en un domaine de valeurs réelles (par exemple, la régression bayésienne et la régression par vecteur de support), et les algorithmes de classification, qui peuvent convertir l'espace d'entrée en classes prédéfinies (comme les réseaux neuronaux artificiels, les arbres de décision, les forêts d'arbre et les machines à vecteur de support SVM)[48].

2. Apprentissage non supervisé : nous avons défini D des exemples de formation non étiquetés.

- Pas de variable cible
- Les connexions se font uniquement entre les valeurs des variables des exemples
- Absence de tuteur et d'instructeur

L'objectif est d'identifier la similarité ou la distance entre les objets pour construire des cluster

3. Apprentissage semi-supervisé : L'apprentissage semi-supervisé est une solution alternative pour pallier le manque de données annotées. Cette méthode d'entraînement permet de combiner des données étiquetées et non étiquetées pour entraîner des modèles. Cette particularité en fait une option intéressante pour les cas où les données classifiées sont limitées, pouvant servir d'alternative aux méthodes d'apprentissage supervisé.[9]

4. Apprentissage par renforcement : Il est utilisé pour résoudre des problèmes de prise de décision séquentielle dans des environnements incertains. Ce processus implique un agent, également appelé sujet, acteur ou décideur, et un environnement ou un système. L'agent doit prendre une série de décisions, chaque décision ayant un impact sur l'état de l'environnement qui évolue dans le temps de manière aléatoire. Les décisions prises par l'agent consistent en des choix d'actions qui dépendent de l'état actuel de l'environnement. Par conséquent, chaque décision prise par l'agent

entraîne un changement dans l'état de l'environnement, qui à son tour influence les décisions suivantes. La dynamique temporelle de ces événements qui caractérisent l'interaction entre l'agent et l'environnement est cruciale dans le processus d'apprentissage par renforcement.

1.3 Apprentissage en profondeur

L'apprentissage profond est un domaine de recherche récent en apprentissage automatique qui vise à rapprocher la discipline de son but ultime : la création d'une intelligence artificielle. Il se caractérise par l'utilisation d'algorithmes inspirés du fonctionnement et de la structure du cerveau humain pour apprendre à modéliser des relations complexes entre des données à plusieurs niveaux de représentation. En résumé, l'apprentissage profond se concentre sur le développement d'algorithmes sophistiqués capables d'apprendre de manière autonome à partir de données complexes et qui pourraient contribuer de manière significative à la création d'une intelligence artificielle puissante[9]

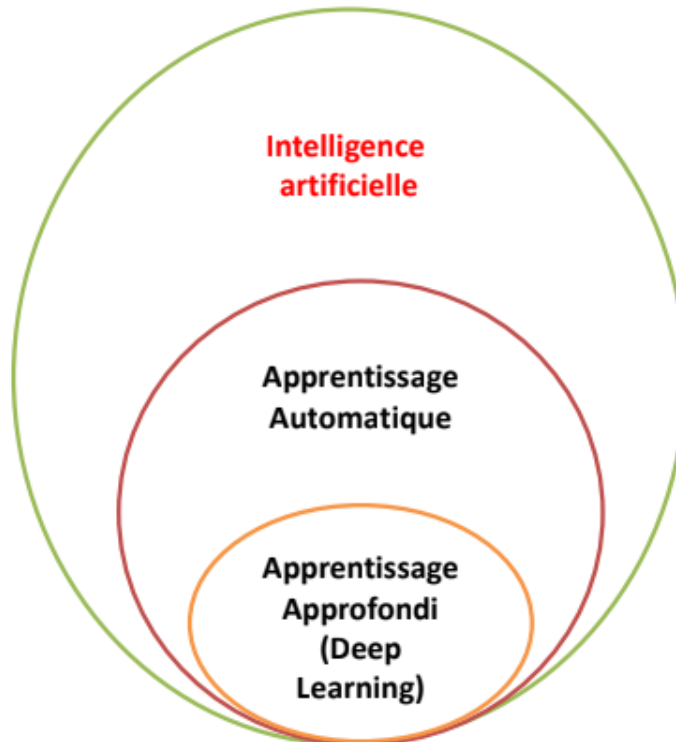


FIGURE 1.2 – La relation entre l'intelligence artificielle, le ML et le deep Learning.

1.4 Les Réseau de neurones

Les réseaux neuronaux artificiels sont directement inspirés des réseaux neuronaux biologiques du cerveau humain, qui comportent de nombreux neurones biologiques hautement connectés. Ces neurones communiquent entre eux en transmettant des signaux par l'intermédiaire de synapses. Ces systèmes biologiques sont extrêmement complexes par rapport aux systèmes artificiels, en raison du grand nombre de neurones inter connectés qui ne peuvent pas être traités par une machine.[15] ,La structure d'un neurone biologique est décrite dans la figure1.3

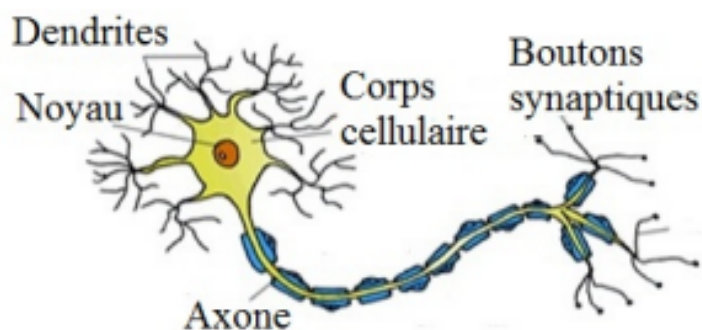


FIGURE 1.3 – un neurone biologique.

Les neurones artificiels ont la capacité d'être utilisé pour une grande variété de problèmes. Si une fonction échelon est choisie comme fonction d'activation, Les neurones peuvent être utilisés pour la classification binaire, Ces types de neurones utilisés pour la classification binaire, sont appelés perceptron.[45]

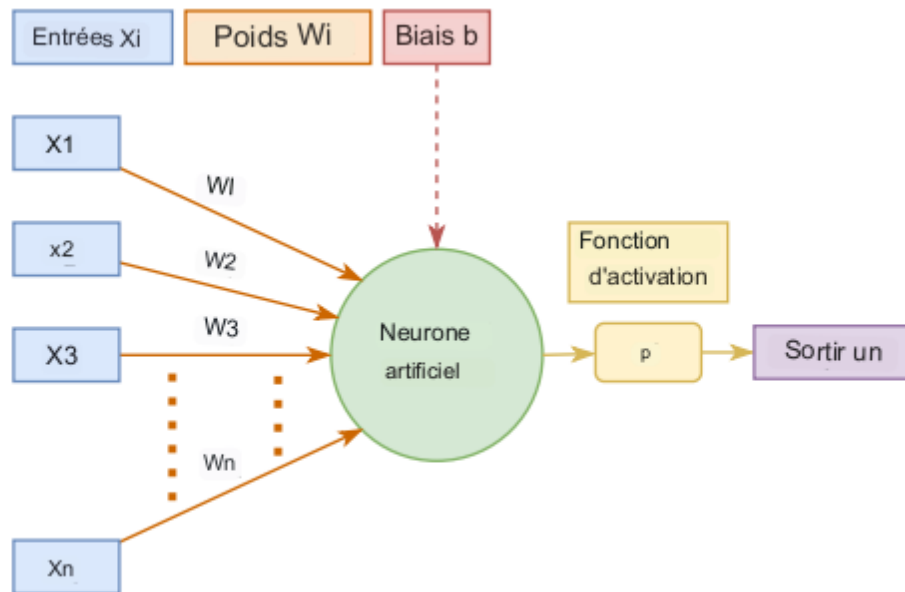


FIGURE 1.4 – un neurone artificiel.[45]

1.4.1 Le perceptron multicouche

Un perceptron multicouche est un réseau de neurones, connu sous le nom de MLP (Multilayers) composé de plusieurs couches : une couche d'entrée, des couches cachées et une couche de sortie.)

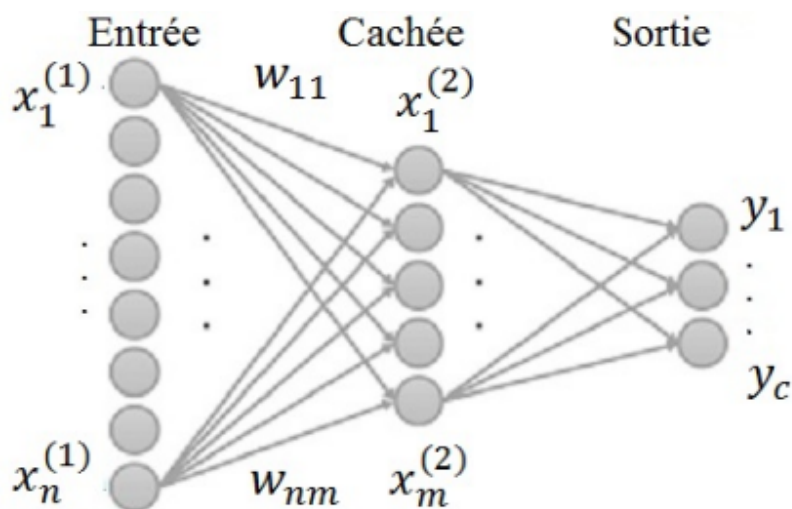


FIGURE 1.5 – L'architecture d'un perceptron multicouche.[45]

La figure 1.5 illustre le processus d'entrée des données dans un perceptron multicouche (MLP). Chaque neurone de la première couche reçoit des données brutes en

entrée, tandis que les couches suivantes reçoivent en entrée les valeurs d'activation de la couche précédente, représentées sous la forme d'un vecteur. La sortie du MLP est constituée des valeurs d'activation de chaque neurone de la dernière couche, également appelée couche de sortie. Dans un MLP, l'activation de chaque couche est calculée en fonction du vecteur d'entrée et est utilisée dans la rétropropagation pour analyser la méthode d'apprentissage pour les problèmes non linéaires.[15] L'adaptation d'un MLP à des poids croissants w_{ij} est réalisée en minimisant une fonction de coût C . [9]

1.4.2 Apprendre avec la rétro-propagation :

Les modèles d'apprentissage en profondeur impliquant plusieurs couches de projection non linéaires sont difficiles à entraîner des modèles d'apprentissage profond repose toujours sur l'algorithme de rétropropagation.

En rétropropagation, les variables du modèle sont mises à jour de manière itérative jusqu'à convergence avec l'algorithme d'optimisation basé sur la descente de gradient. En plus de l'algorithme classique de descente de gradient Traditionnellement, de nombreuses variantes de descente de gradient ont également été proposées ces dernières années Améliorer les performances académiques, y compris Momentum, Adagrad, Adam, Gadam, etc.,[50]

1. Méthode de la rétro-propagation :La rétropropagation est une méthode d'analyse utilisée dans les réseaux neuronaux artificiels pour évaluer l'erreur de chaque neurone après le traitement d'un lot de données. Elle est souvent utilisée dans les algorithmes d'optimisation par descente de gradient pour ajuster les poids des neurones en calculant le gradient de la fonction de perte. Cette technique est parfois appelée propagation inverse de l'erreur parce que l'erreur est calculée à la sortie et propagée à travers les couches du réseau. Elle nécessite également des sorties connues et souhaitées pour chaque valeur d'entrée, et est donc considérée comme une méthode d'apprentissage supervisée..[33]

L'algorithme peut être décrit comme suit :

Entrée :

- Ensemble de données d'entraînement
- Architecture du réseau de neurones (nombre de couches, nombre de neurones par couche)
- Fonction de perte (par exemple, erreur quadratique moyenne) ●Taux d'apprentissage

● Nombre d'époques (nombre de fois que l'algorithme parcourt l'ensemble de données d'entraînement)

1. Initialiser les poids du réseau de manière aléatoire
2. Répéter le nombre d'époques : Pour chaque exemple d'entraînement :
 - i. Effectuer une propagation avant pour calculer les sorties de chaque couche
 - ii. Calculer la dérivée partielle de la fonction de perte par rapport à chaque poids dans le réseau en utilisant la règle de la chaîne
 - iii. Mettre à jour les poids en utilisant la formule :
 $\text{poids} = \text{poids} - \text{taux d'apprentissage} * \text{dérivée partielle de la fonction de perte par rapport au poids}$
3. Retourner les poids ajustés

Descente de gradient : L'algorithme de rétropropagation utilise la descente de gradient, une méthode d'optimisation itérative du premier ordre couramment utilisée dans l'apprentissage automatique. Cette méthode permet de trouver le minimum d'une fonction en utilisant des étapes proportionnelles à la valeur négative du gradient de la fonction au point actuel. Elle est utilisée pour éviter les minimum locaux dans les fonctions optimisées.[33]

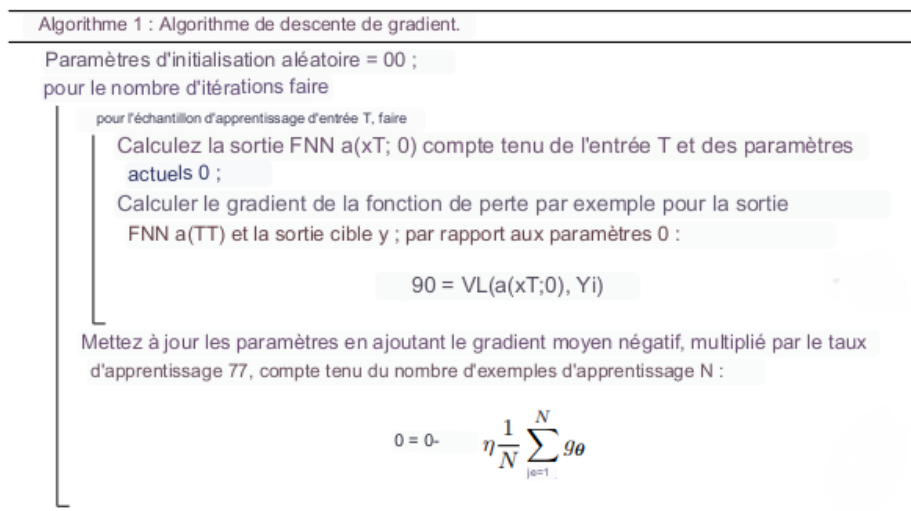


FIGURE 1.6 – L'algorithme de Descente de gradient .[33]

2. Les techniques d'optimisation : Le MLP repose principalement sur la méthode de descente de gradient, qui comprend diverses variantes telles que :

- Descente de gradient stochastique (SGD)
- Descente de gradient à mini-lots (BGD)

La descente avec inertie

Descente de gradient accélérée de Nesterov (NAG)[9]

1.4.3 Les Architecture de DL

Dans cette section, nous examinons différents types de réseaux profonds, qui impliquent généralement plusieurs couches d'étapes de traitement de l'information apprises dans une structure hiérarchique. Les techniques d'apprentissage en profondeur peuvent être regroupées en trois grandes catégories : (i) les réseaux en profondeur pour l'apprentissage supervisé ou discriminatif; (ii) les réseaux en profondeur pour l'apprentissage non supervisé ou génératif; et (iii) les réseaux en profondeur pour l'apprentissage hybride qui combine ces deux types d'apprentissage et d'autres éléments pertinents, comme l'illustre la figure 1.7.[36]

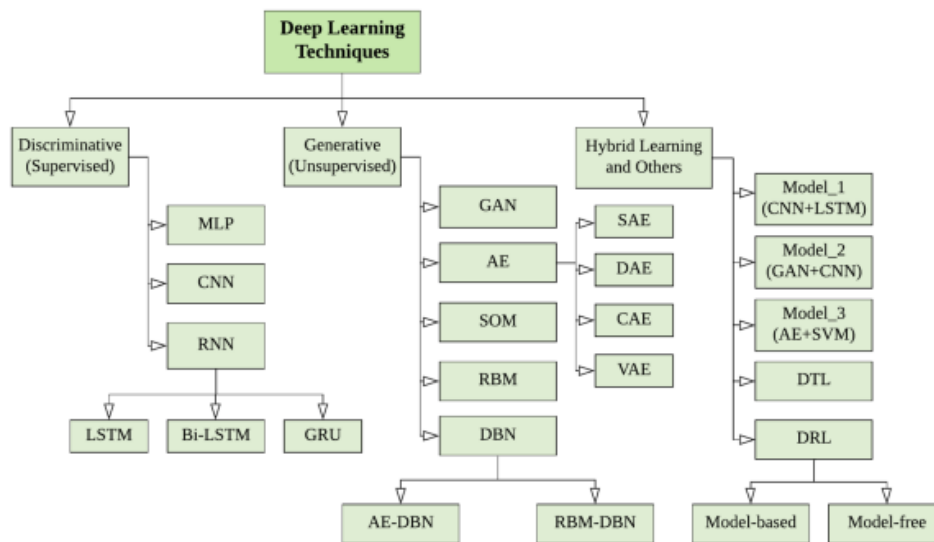


FIGURE 1.7 – Les techniques de DL.

Dans ce qui suit ,nous détaillons d'abord l'architecture d'un réseau de neurone convolutif qui constitue le modèle de base des architectures DL , Ensuite nous présentons les bases des modèles GAN et les Autoencoder

1.Les réseaux neuronal convolutifs :

Le réseau neuronal convolutif (CNN) est une variante des réseaux neuronaux artificiels (ANN) qui se distingue par sa capacité à apprendre automatiquement un grand nombre de filtres parallèles spécifiques à l'ensemble de données d'apprentissage, dans les limites du problème de modélisation prédictive spécifique, tel que la

classification d'images. Les CNN sont constitués de neurones dont les poids et les biais peuvent être appris. Chaque neurone spécifique reçoit de nombreuses entrées, en fait une somme pondérée, applique une fonction d'activation et produit une sortie. Les CNN sont composés de quatre types de couches sont :[37]

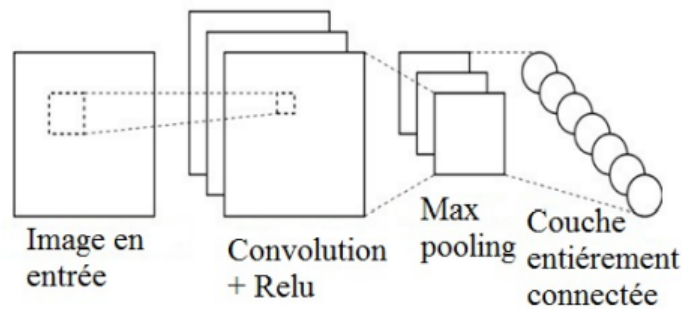


FIGURE 1.8 – Schéma des réseaux de neurones convolutifs .

1) Couche de convolution

Dans une architecture CNN, les composants les plus importants sont les couches convolutionnelles. Il consiste en un ensemble de filtres convolutifs. L'image d'entrée, représentée sous la forme d'un volume à N dimensions, est convoluée avec ces filtres pour produire une carte de caractéristiques de sortie

le noyau du filtre :est Une grille de nombres ou de valeurs discrètes décrit le kernel. Chaque valeur est appelée un poids de base. Au début du processus de formation du CNN, des nombres aléatoires sont attribués en tant que poids du kernel. Ces poids sont ensuite ajustés à chaque époque d'apprentissage ; ainsi, le kernel apprend à extraire des caractéristiques significatives.

.Convolutional Operation :L'opération de convolution est une opération clé des réseaux neuronaux convolutifs (CNN) utilisés pour le traitement des images. Cette opération consiste à appliquer un filtre ou un noyau de convolution à l'image d'entrée en effectuant une opération de produit de points entre les valeurs de chaque pixel de l'image et les valeurs du filtre. Cette opération permet de capturer des caractéristiques spécifiques de l'image, telles que les bords, les contours ou les textures. La convolution peut également être effectuée sur plusieurs canaux d'entrée pour capturer des informations supplémentaires. Elle est suivie d'une fonction d'activation non linéaire pour introduire la non-linéarité dans les données et augmenter la capacité

de représentation du réseau. La convolution est souvent utilisée en combinaison avec d'autres couches, telles que des couches de mise en commun ou de normalisation, pour construire des architectures CNN plus complexes..[4]

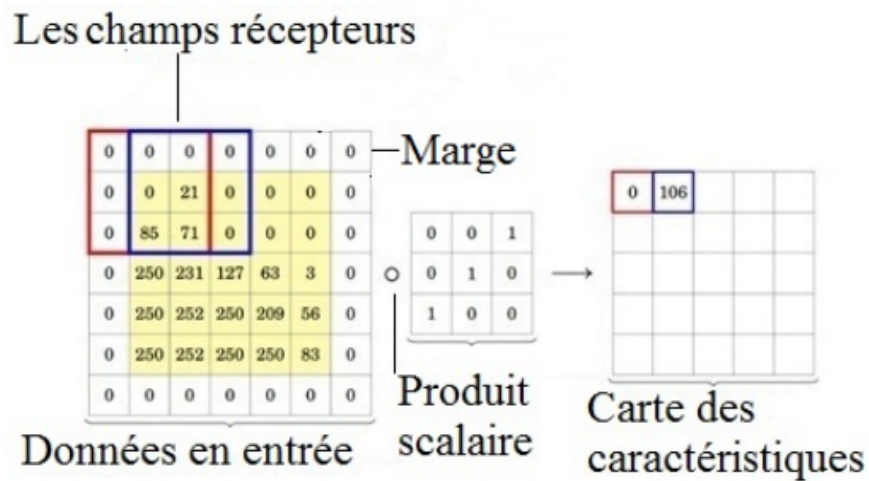


FIGURE 1.9 – Une opération de convolution.[9]

2)La couche de **pooling** est principalement utilisée pour effectuer un sous-échantillonnage des cartes d'entités générées par les opérations de convolution. En réduisant la taille des cartes d'entités, cette approche permet de créer des cartes d'entités plus petites comme le montre la figure1.10.

3)Couche entièrement connectée :Les couches entièrement connectées d'un CNN ont une structure similaire à celle d'un MLP. Leur objectif est d'apprendre des combinaisons non linéaires entre les caractéristiques extraites par les couches convolutives. Le résultat de la dernière couche convolutive est un vecteur qui représente l'entrée d'un ensemble de couches entièrement connectées. Dans la classification supervisée, la dernière couche est utilisée pour prédire les classes à l'aide de la fonction d'activation Softmax.

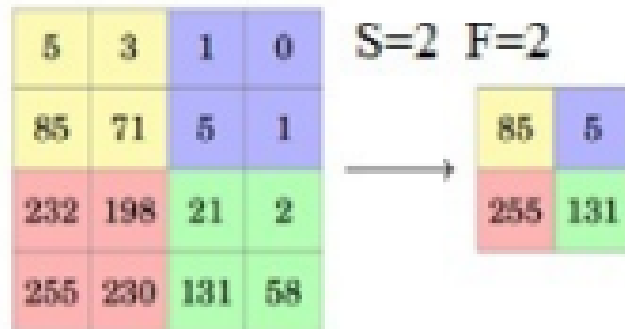


FIGURE 1.10 – Une opération de Max-pooling.[9]

ReLU : La fonction ReLU est un acronyme pour "Rectified Linear Unit" (unité linéaire rectifiée), couramment utilisée dans les réseaux neuronaux. L'avantage de l'utilisation de la fonction ReLU est que seuls certains neurones sont activés, et non tous les neurones en même temps. Ainsi, un neurone sera désactivé lorsque la sortie de la transformation linéaire est nulle. La ReLU est plus efficace que d'autres fonctions car seul un sous-ensemble de neurones est activé à la fois. Toutefois, dans certaines situations, la valeur du gradient peut être nulle, ce qui empêche de modifier les poids et les biais pendant la rétropropagation de l'apprentissage du réseau neuronal.[38], la figure 1.11 présente le graphe de la fonction d'activation Relu

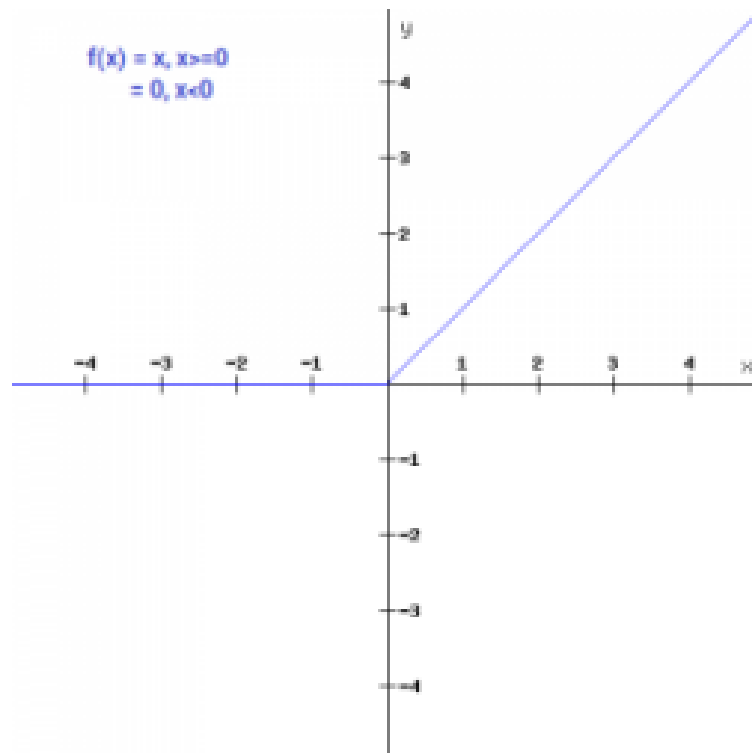


FIGURE 1.11 – Tracé de la fonction d'activation de ReLU.

4) **Dropout** : une technique de régularisation efficace pour les réseaux de type feed-forward neural networks (FNN). Elle consiste à former un ensemble virtuel de tous les sous-réseaux qui peuvent être construits en supprimant des unités de la couche de sortie d'un réseau de base. Cette approche réduit le surapprentissage en forçant le réseau à apprendre des représentations plus robustes des données d'entrée. En d'autres termes, Dropout empêche les neurones de s'adapter de manière trop spécifique aux données d'apprentissage, ce qui peut entraîner une baisse des performances lors de la phase de test[45]. la figure1.12 montre un exemple de cette technique

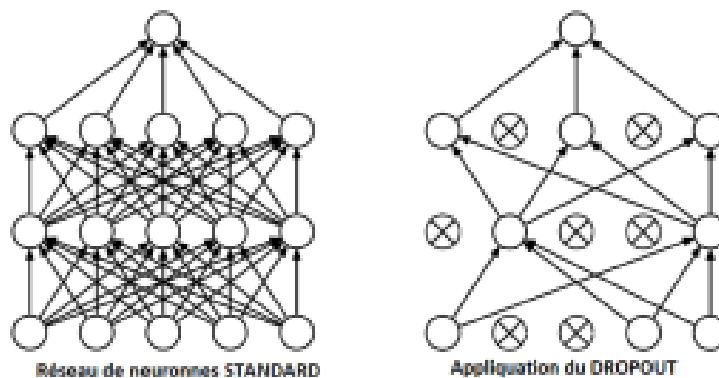


FIGURE 1.12 – Exemple du Dropout[43].

1.5 Réseaux de neurones non supervisés :

1.5.1 Réseau adversarial génératif (GAN)

un modèle de réseau génératif adversarial (GAN) a été développé en raison de sa capacité à générer des échantillons de haute qualité de visages réels. La synthèse d'images de visages à l'aide des GAN a été un succès, permettant une optimisation immédiate et des résultats réalistes. De plus, les GAN conditionnels étendus offrent la possibilité de contrôler les caractéristiques des images en manipulant des vecteurs potentiels et en décodant des représentations dimensionnelles réduites en espace latent. La figure suivante illustre les différents types de GANs[13].

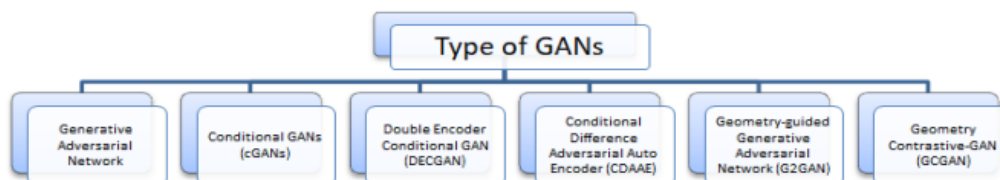


FIGURE 1.13 – Les type des Gans[13].

Présentation de GAN Les GANs [18] font partie de la famille des modèles génétiques. Ils sont capables de générer des images d'objets similaires à partir d'un ensemble de données sans qu'ils soient nécessairement identiques à ceux qui existent déjà. Dans le domaine des visages humains, l'objectif des GANs est de générer des photographies à la fois réalistes, distinctives et diverses. La figure 1.14 illustre deux exemples de résultats obtenus par (Karras et al(2019)).[13].



FIGURE 1.14 – Exemples de visages humains générés à partir de la structure proposée[13].

Les GANs sont structurés en deux parties principales : le générateur et le discriminateur :

Le Générateur : A pour objectif de produire des images de haute qualité en utilisant un bruit d'entrée (gaussien ou uniforme) et en le transformant dans un espace d'image. Pour être efficace, le générateur nécessite un espace latent avec une distribution spécifiée, connue sous le nom de distribution dormante. Son rôle est de garantir que le générateur produit les meilleures images possibles en utilisant cette distribution.

Le Discriminateur :Le discriminateur, quant à lui, apprend à distinguer les fausses images générées par le générateur des vraies données disponibles dans le jeu d'entraînement. Même si le discriminateur sert de fonction de soutien, il n'en est pas moins vital puisqu'il fournit les informations nécessaires et suffisantes au générateur.

d'un point de vue probabiliste, transfère la distribution latente dans l'espace d'arrivée et définit ainsi une mesure d'image. Les GANs utilisent ensuite cette mesure d'image pour estimer la distribution cible. Le discriminateur, quant à lui, définit une distance (ou divergence) entre les deux distributions de probabilité, la distribution cible et la distribution produite, en discriminant les vraies et fausses images. En termes d'optimisation, le générateur cherche à tromper le discriminateur tandis que le discriminateur est entraîné de manière supervisée : il accepte en entrée des images vraies et fausses et tente de les catégoriser correctement [44].

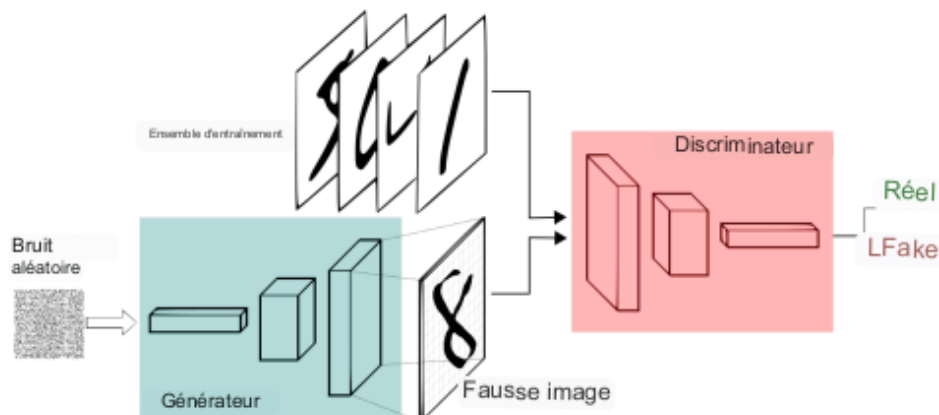


FIGURE 1.15 – Architecture classique d'un GAN.

1.5.2 Les Types des GANs :

1. Réseau adversarial génératif conditionnel(cGAN) :

Une fois que le vecteur potentiel est paramétré, l'information contenue dans l'image est structurée de manière efficace, donnant ainsi naissance au réseau adversarial génératif conditionnel (cGAN). Ce type de réseau est capable de générer des images contrôlées par des étiquettes et des caractéristiques spécifiques. En analysant les résultats de sortie, les cGANs sont également utilisés pour des applications telles que la modification faciale, le vieillissement du visage et d'autres transferts d'expressions faciales.

2. Double Encoder Conditional GAN (DECGAN) :

Les GANs peuvent être réglés directement pour fournir les données les plus réalistes et les plus plausibles. Les DCGAN sont utilisés pour réguler les caractéristiques des images produites. Grâce à l'apprentissage associatif, deux codeurs échangent des caractéristiques. Les données peuvent être corrélées du domaine cible au domaine d'origine en reconstruisant sur un ensemble cohérent de qualités intrinsèques.[29]

1.5.3 Les domaines d'application des GANs

En plus d'être présents dans des domaines importants, les réseaux adversariens génératifs ont un grand potentiel :

L'analyse d'images : La littérature sur l'analyse d'images à l'aide des GAN est très variée. ont envisagé comment les GAN pourraient faciliter l'édition d'images. Il est possible de le faire revivre, de lui ajouter des lunettes, ou de modifier son

genre en se déplaçant dans certaines directions de l'espace dormant. sont maintenant capables de modifier une image en lui donnant un style photo ou tableau.

La génération de vidéos : Au-delà de l'analyse d'images, les GANs ont été appliqués avec succès dans divers domaines de recherche. En se basant sur les récents développements en analyse vidéo, notamment la convolution 3D, les GAN se sont révélés très efficaces pour produire des vidéos.

Le langage : Les GANs ne sont pas directement applicables dans le domaine du traitement du langage naturel (NLP) en raison de leur conception initiale qui implique le calcul des gradients de la sortie du générateur. Malheureusement, cette opération n'est pas réalisable dans le domaine discret, tel que le traitement du langage naturel. Par conséquent, les GANs ne sont pas directement adaptés à l'application dans le NLP.[44]

1.5.4 Auto-Encoder (AE) :

Un auto-encodeur (AE) est une approche d'apprentissage non supervisé bien connue qui utilise des réseaux neuronaux pour apprendre des représentations. Un auto-codeur comme le montre la figure 1.16 est composé de trois parties : l'encodeur, le code et le décodeur. L'encodeur comprime l'entrée et génère le code, que le décodeur utilise ensuite pour recréer l'entrée. L'auto-encodeur est largement utilisé dans de nombreuses tâches d'apprentissage non supervisé, tel que la réduction de la dimensionnalité, l'extraction de caractéristiques, le codage efficace, la modélisation générative, le débruitage, etc.

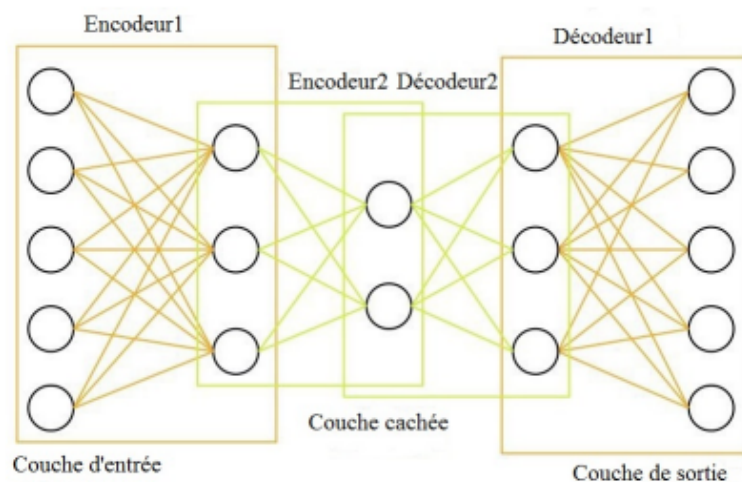


FIGURE 1.16 – La structure d'un auto-encodeur.[9]

1.5.2.1 les types de AE : Plusieurs variétés d'auto-encodeurs ont été proposées depuis leur introduction. Les plus couramment utilisés sont décrits dans ce qui suit :

Auto-codeur de débruitage (DAE) : Un auto-codeur de débruitage est une variante de l'auto-codeur de base qui vise à améliorer la représentation en modifiant le critère de reconstruction. Son objectif est de réduire le risque d'apprentissage de la fonction d'identité en minimisant l'erreur moyenne de reconstruction sur les données d'apprentissage. Concrètement, il prend en entrée des données corrompues et apprend à les nettoyer en récupérant la version originale non déformée en sortie. En informatique, les auto-codeurs de débruitage agissent comme des filtres puissants utilisés pour le prétraitement automatique des données. Par exemple, dans le cas d'une image, un auto-codeur de débruitage peut améliorer sa qualité afin d'optimiser la précision de la reconnaissance[36].

Variational Autoencoder (VAE) : La principale différence entre un auto-codeur variationnel (VAE) et l'auto-codeur classique mentionné précédemment réside dans son efficacité pour la modélisation générative. Les VAE transforment les données d'entrée en paramètres d'une distribution de probabilité, tels que la moyenne et la variance d'une distribution gaussienne, tandis que les auto-codeurs traditionnels transposent l'entrée en un vecteur latent. Dans un VAE, on suppose une distribution de probabilité sous-jacente des données sources, et les paramètres de cette distribution sont ensuite estimés. Bien que cette méthode ait été initialement développée pour l'apprentissage non supervisé, elle a été utilisée avec succès dans d'autres contextes tels que l'apprentissage supervisé et l'apprentissage semi-supervisé[36].

Autoencodeur clairsemé (SAE) : Dans le cadre de ses exigences d'apprentissage, un autoencodeur clairsemé impose une pénalité de clairsemination à la couche de codage. Bien que les SAE puissent contenir plus d'unités cachées que d'entrées, seul un nombre limité d'unités cachées peut être actif en même temps, ce qui donne un modèle clairsemé. Par conséquent, ce modèle est obligé de répondre aux propriétés statistiques uniques des données d'apprentissage tout en respectant ses limites[36].

1.6 StyleGAN

NVIDIA StyleGAN est une nouvelle technologie basée sur l'IA qui permet de produire des images photoréalistes de personnes, de paysages et d'objets avec un niveau de réalisme sans précédent. Il repose sur les avancées des réseaux adverses génératifs (GAN), une forme de réseau neuronal qui a récemment été utilisée pour générer des images et des vidéos truquées très réalistes. Le fonctionnement de StyleGAN repose sur une collaboration entre deux réseaux : un générateur qui crée la sortie souhaitée et un discriminateur qui essaie de distinguer les images générées des images réelles. Les GAN peuvent apprendre à produire des images de plus en plus réalistes en ajustant les poids de ces réseaux en fonction des informations fournies par le discriminateur[25].

1.6.1 Les Variantes de Stylegan

Stylegan StyleGAN est un modèle de réseau adversarial génératif (GAN) qui se distingue par sa capacité à générer des images de haute qualité et haute résolution. Cette architecture, initialement proposée par les chercheurs de NVIDIA en 2018, a rapidement suscité l'intérêt de la communauté scientifique en raison de ses résultats impressionnants.

L'un des principaux avantages de StyleGAN réside dans sa capacité à personnaliser de nombreuses caractéristiques des images générées, telles que la qualité, le style et même des attributs spécifiques comme la couleur des cheveux et les expressions faciales. Cela est rendu possible grâce à l'utilisation d'un espace latent, un espace vectoriel de haute dimension qui contient des informations sur les différents aspects des images créées.

Le générateur de StyleGAN utilise un réseau neuronal profond qui prend en entrée un vecteur de bruit aléatoire et génère une image de haute qualité en utilisant un ensemble de filtres appris à partir des données d'entraînement. Le modèle utilise également des "styles" pour contrôler la génération de l'image, permettant à l'utilisateur de manipuler des attributs tels que la couleur, la luminosité, la texture et la géométrie.

De plus, StyleGAN utilise la technique de normalisation adaptative des instances (AdaIN), qui permet au réseau de modifier la moyenne et l'écart-type des caractéristiques de l'image générée, offrant ainsi une plus grande flexibilité dans la création d'images variées et réalistes.

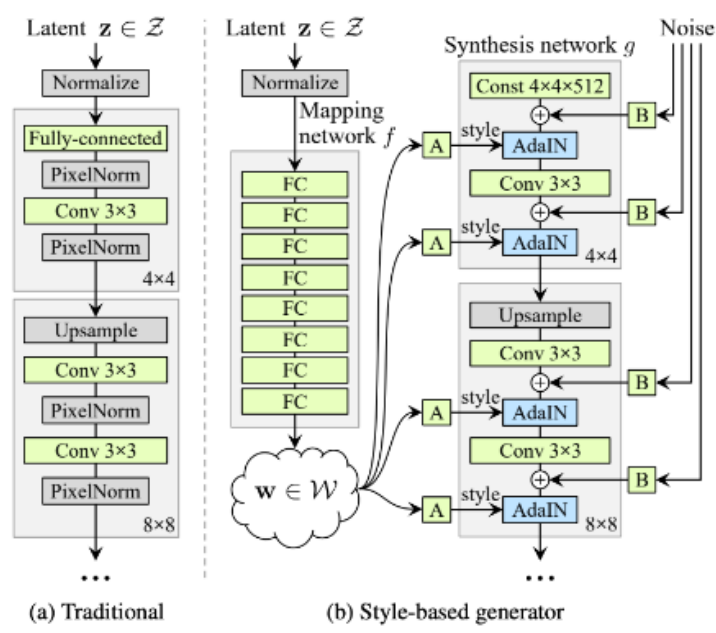


FIGURE 1.17 – L'architecture du Stylegan.[24]

Stylegan2 : Il s'agit d'une version améliorée de StyleGAN, qui a été publiée en 2018. L'utilisation de la normalisation adaptative des instances (AdaIN) pour augmenter la qualité et la variété des images produites est l'innovation fondamentale de StyleGAN2. L'inclusion d'un discriminateur adaptatif, qui permet un entraînement plus efficace et une meilleure convergence du générateur, est l'une des avancées significatives introduites dans StyleGAN2. De plus, StyleGAN2 comprend une nouvelle architecture de générateur qui permet un contrôle plus étendu des graphiques produits. Sa capacité à gérer la quantité de détails, la position et l'expression des visages générés en fait un outil populaire dans le domaine de la vision par ordinateur et du graphisme. Une autre caractéristique importante de StyleGAN2 est sa capacité à créer de grandes images avec une grande qualité et des détails allant jusqu'à 1024 par 1024 pixels. Il s'agit donc d'un excellent outil pour générer des photographies et des visages en haute résolution, ainsi que pour des applications telles que la réalité virtuelle et la réalité augmentée. StyleGAN2 est l'utilisation d'un nouveau type de réseau générateur appelé mapping network.. Le réseau de mapping prend un vecteur de bruit aléatoire en entrée et le mappe dans un espace latent qui est plus adapté à la génération d'images. Cela permet un contrôle plus fin du processus de génération d'images.

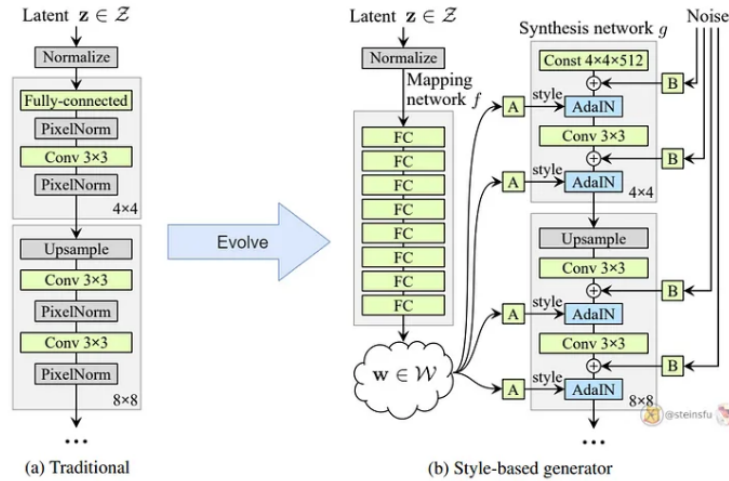


FIGURE 1.18 – L'architecture du StyleGAN2[24].

StyleGAN2-Ada StyleGAN2-ADA est une extension du modèle StyleGAN2, qui a été introduite pour répondre à certaines de ses limitations. Il a été créé par Tero Karras, Samuli Laine et Timo Aila de NVIDIA. Le terme " ADA " dans StyleGAN2-ADA signifie " adaptive discriminator augmentation " (augmentation adaptative du discriminateur), qui est une caractéristique clé du modèle.

L'un des principaux défis de l'entraînement des GAN est que le discriminateur peut rapidement devenir trop puissant, ce qui rend difficile l'apprentissage du générateur. StyleGAN2-ADA résout ce problème en utilisant une technique appelée augmentation adaptative du discriminateur, où le discriminateur est dynamiquement augmenté avec de nouvelles données pendant la formation. Cela permet d'éviter qu'il ne devienne trop puissant et permet au générateur d'apprendre plus efficacement.

StyleGAN2-ADA introduit également plusieurs autres améliorations par rapport au StyleGAN2 original, notamment une meilleure gestion des petits détails, une meilleure stabilité de l'apprentissage et la possibilité de générer des images haute résolution jusqu'à 1024x1024 pixels.[25]

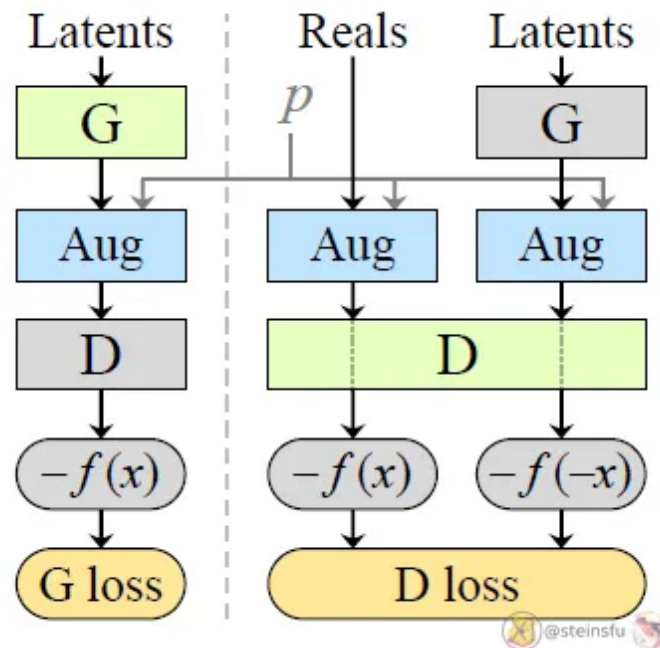


FIGURE 1.19 – L'architecture du Stylegan2-Ada.[16]

StyleGAN3 Le but de StyleGAN3 est de résoudre le problème du " collage de texture " qui se produisait lors de la transition de morphing (par exemple, le morphing d'un visage à un autre) dans StyleGAN2. StyleGAN3 est un modèle génératif de pointe pour la synthèse d'images et de vidéos de haute qualité. Il a été développé par des chercheurs de NVIDIA et constitue une extension des modèles StyleGAN précédents.

StyleGAN3 comprend plusieurs améliorations par rapport à ses prédécesseurs, notamment un nouvel algorithme d'apprentissage appelé Lazy Regularization, une architecture plus efficace appelée Convolutions Over Pixels et un nouveau jeu de données d'apprentissage appelé BaseGAN dataset.

Ces améliorations ont conduit à des progrès significatifs dans la qualité et la diversité des images générées par StyleGAN3. Le modèle peut générer des images très réalistes et complexes avec des détails et des textures fins, ce qui le rend utile pour une large gamme d'applications telles que l'art, la mode et le développement de jeux.

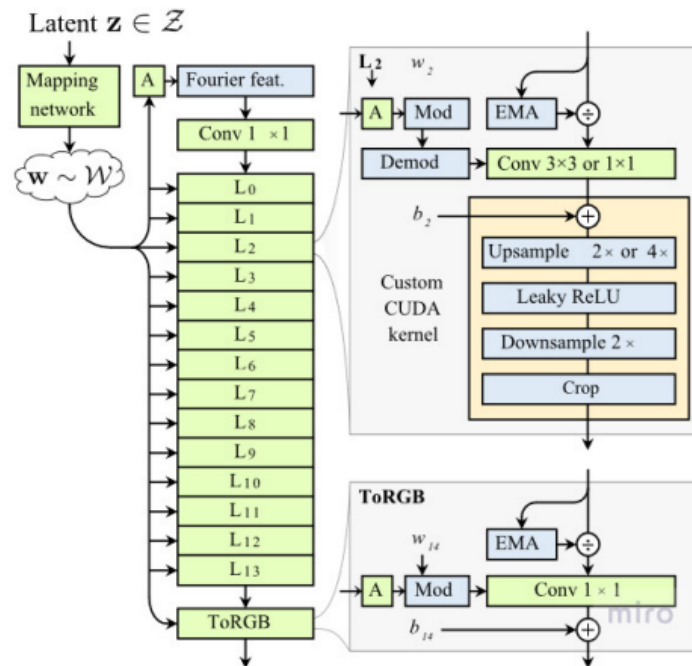


FIGURE 1.20 – L'architecture du StyleGAN3[25].

1.7 Conclusion

Apprentissage profonds est une branche de l'intelligence artificielle (IA) basée sur l'apprentissage automatique de représentations de données. Les réseaux de neurones, en particulier les CNN (Convolutional Neural Networks), sont devenus la pierre angulaire du Deep Learning. Les GAN (Generative Adversarial Networks) aussi sont une architecture de réseau de neurones utilisée pour la génération de données synthétiques. Le StyleGAN est une implémentation particulière de GAN permettant la génération de visages et d'images réalistes avec un haut niveau de contrôle sur les caractéristiques de l'image. il utilise une approche hiérarchique pour apprendre les caractéristiques d'images à différentes échelles, permettant la génération de détails fins et la séparation de l'apprentissage des styles et du contenu de l'image.

Chapitre 2

AvatarGénération de visages de dessins animés : Avatar ou cartoon.

2.1 Introduction

L'avatar est un modèle numérique représentant votre identité sur différents réseaux. Il vous permet d'interagir en ligne, de diffuser sur Internet, de tenir des réunions virtuelles, d'enseigner à un public en ligne et bien plus encore. Avoir un avatar vous ouvre les portes d'un monde virtuel passionnant et vous procure une joie infinie. Créons tous nos propres avatars 2D/3D pour commencer. Avec le développement rapide de la technologie de modélisation 3D, stimulé par l'essor du marché des jeux, nous pouvons désormais personnaliser nos avatars de manière plus poussée, les rendant plus proches de nous-mêmes ou de nos aspirations.

2.2 Définition

Les avatars numériques sont des représentations virtuelles d'eux-mêmes. Ils permettent aux personnes du monde entier de communiquer dans le monde numérique d'une manière naturelle et humaine. Les avatars sont largement utilisés dans les jeux en ligne et les sites de réseaux sociaux . Si la préservation de l'anonymat des utilisateurs est un droit fondamental des internautes, il est également souhaitable de disposer d'avatars virtuels capables de transmettre des émotions authentiques. Il est plus facile de communiquer et de distinguer les humains des faux personnages lorsqu'ils ont de véritables émotions. Les expressions faciales sont des caractéristiques humaines essentielles qui ne révèlent rien de l'identité d'une personne.[30]

2.3 Type d'avatar

2.3.1 Avatar tridimensionnelle

Un avatar 3D est une représentation virtuelle numérique d'un personnage qui peut vous représenter sur d'autres plateformes en ligne. Vous pouvez le programmer pour diffuser en direct sur Internet, interagir avec votre famille et vos amis en ligne, organiser des réunions avec vos collègues, ou enseigner à votre public en ligne, parmi d'autres possibilités. Posséder un avatar 3D vous permet de plonger dans un monde virtuel fascinant et peut vous offrir un plaisir illimité. Commençons tous par créer notre propre avatar 3D personnalisé.



FIGURE 2.1 – 3D avatar.

2.3.2 Animal Avatars :

Les individus choisissent un avatar animal qui, selon eux, incarne leur identité ou représente une signification ou un symbolisme qu'ils trouvent attrayant. Les animaux représentent souvent des attributs et des personnalités spécifiques. Dans la société moderne, de nombreux individus ont des animaux de compagnie tels que des chiens, des chats et des oiseaux, et ces animaux sont souvent considérés comme un membre de la famille, un compagnon ou une autre personnalité.[28]



FIGURE 2.2 – Animal Avatar.

2.3.3 Avatar de jeu :

C'est une Représentation graphique du joueur sous la forme d'un personnage humain ou non humain. Plusieurs degrés de personnalisation (corps, visage et vêtements) sont utilisés selon l'outil d'animation, qui sont contrôlé par l'utilisateur par des mouvements (flèches ou clics), des gestes et des émotions.[10]

2.3.4 Avatar humain :

Les avatars humains articulés et d'apparence réaliste a de nombreuses applications dans les domaines de la télé conférence, des jeux, de la réalité augmentée et de la réalité virtuelle. Créé des personnes qui représentent le contour du corps, y compris les déformations du visage et des mains sont crée les dernières années. Ces modèles sont basés sur une géométrie maillée et sont appris à partir d'ensembles de données de scanners corporels existants. Les vêtements et les cheveux ajoutent physiquement une couche supplémentaire de difficulté à la modélisation du corps ; ils sont considérablement plus difficiles à modéliser en apparence, et les données 3D correctes pour ces éléments sont rares et difficiles à obtenir. La création d'avatars réalistes est toutefois impossible sans la modélisation de ces caractéristiques.[14]



FIGURE 2.3 – Avatar Human[49].

2.3.5 Cartoon Avatar :

Les dessins animés sont des illustrations non réalistes ou semi-réalistes, qui peuvent être dessinées à la main ou générées par ordinateur. Lorsque ces dessins présentent des mouvements, ils deviennent des animations. Ils sont utilisés dans les médias imprimés, la publicité, les médias numériques et l'industrie du divertissement. Dans les dessins animés et les animations, un personnage vivant peut être créé sans l'intervention d'acteurs humains en tant qu'acteurs. Ces productions sont susceptibles de coûter moins cher qu'elles le seraient si des humains étaient impliqués en tant qu'acteurs. Dans les films d'animation, l'objectif est de créer des personnages réalistes, semblables à des humains, en utilisant des dessins animés. La proximité du réalisme est directement proportionnelle à la complexité de la génération d'un dessin animé. [42]



FIGURE 2.4 – Avatar Cartoon[42].

2.4 Domaines d'utilisation des avatars :

2.4.1 Dans la réalité virtuelle :

Un système de réalité virtuelle (RV) immerge l'utilisateur dans un espace ou un autre monde simulé. L'interactivité à l'intérieur de ladite zone, les affichages (c'est-à-dire les dimensions sensorielles de la vue, de l'ouïe, du toucher, etc.) fournissant un retour d'information pour les interactions à l'intérieur dudit espace, et un environnement fabriqué qui permet l'immersion, ou le sentiment de "se perdre" dans une expérience, sont des caractéristiques clés de la RV [40]. L'objectif de la technologie de RV pure est de fournir un substitut aux éléments authentiques ; une illusion complète d'une notion et/ou d'un cadre réel ou imaginaire. Ce type de virtualité diffère des formes de réalité mixte telles que la réalité augmentée, dans laquelle des éléments virtuels sont injectés dans des situations du monde réel, et la virtualité améliorée, dans laquelle des composants du monde réel sont insérés dans un environnement essentiellement virtuel[27]. Les simulations virtuelles ont été utilisées par les chercheurs.

2.4.2 Les jeux Video :

Les avatars jouent un rôle tout à fait distinct dans les jeux vidéo : ils permettent aux utilisateurs de construire un personnage 3D à part entière qui les représentera

dans certains jeux, notamment les jeux en réseau. Les utilisateurs qui jouaient sous la forme d'un avatar qui leur ressemblait dans un jeu avaient une pulsion intrinsèque plus élevée que les utilisateurs qui jouaient sous la forme d'un avatar aléatoire pour leur personnage [1].

2.4.3 Réseaux sociaux :

L'utilisation d'avatars dans les réseaux sociaux est très courante, ils sont là pour augmenter la valeur de reconnaissance d'une personne ou d'un profil au sein du réseau. Ils sont considérés comme un élément de confiance. De nombreuses personnes utilisent une photo authentique d'elles-mêmes comme avatar sur les réseaux sociaux. Cette pratique n'est pas courante sur les forums, où les utilisateurs préservent souvent leur anonymat. On utilise plutôt des dessins animés, des graphiques en 3D, des images de lieux, de créatures ou d'étoiles, des croquis, des slogans et d'autres éléments de ce type [35].

2.4.4 Les systèmes d'apprentissage en ligne

Les systèmes d'apprentissage en ligne basés sur la stratégie des avatars constituent un environnement particulièrement efficace pour dispenser une meilleure formation. Ces plateformes virtuelles ont été développées afin que les individus puissent personnaliser leur avatar : qu'il s'agisse d'une représentation humaine ou non humaine répondant ainsi parfaitement à leurs besoins spécifiques pour accomplir leurs objectifs éducatifs. Les utilisateurs interagissent via leur avatar avec diverses ressources d'apprentissage (instructeurs ou autres apprenants). Participent à plusieurs activités visant l'interactivité autour du processus éducatif : simulation, jeux sérieux, chatbots etc [32].

2.5 Les bases de données d'avatars :

2.5.1 FERG (Facial Expression Research Group Database) :

FERG est une base de données de personnages de dessins animés avec des expressions faciales annotées contenant 55 769 images de visages annotées de six personnages. Les images de chaque personnage sont regroupées en 7 types d'expressions de base, à savoir la colère, le dégoût, la peur, la joie, la neutralité, tristesse et la surprise, la joie, la neutralité, la tristesse, ils ont créé les poses clés pour chaque expression, et elles ont été étiquetées par Turc mécanique (MT) [7] pour alimenter la

base de données au départ. Le nombre de poses clés créées dépend de la complexité de l'expression pour chaque personnage.[5]. Quelques exemples d'images de cette base de données sont illustrés dans la figure 2.5.



FIGURE 2.5 – Quatre échantillons de sujets présentant sept expressions (neutre, colère, dégoût, peur, bonheur, tristesse et surprise)[5].

2.5.2 MetFaces dataset :

L'ensemble de données MetFaces est une collection de plus de 140 000 images d'œuvres d'art provenant du Metropolitan Museum of Art de New York. L'ensemble de données comprend une variété de styles et de genres provenant de différentes périodes et cultures. Chaque image est accompagnée de métadonnées comprenant l'artiste, le titre, la date et le support de l'œuvre. L'ensemble de données peut être utilisé pour diverses tâches de vision artificielle et d'apprentissage automatique, telles que la classification d'images, la détection d'objets et l'art génératif[16]. La figure 2.6 montre deux exemples de cette base de données .

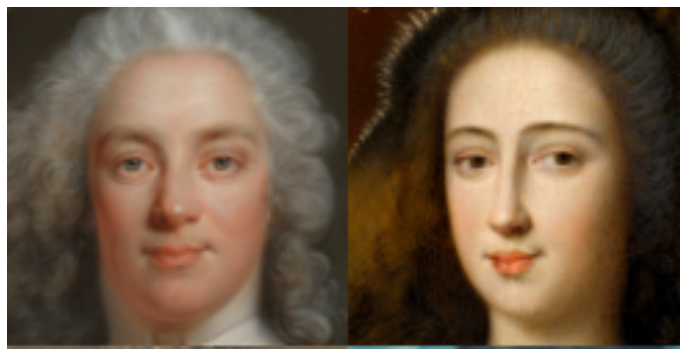


FIGURE 2.6 – Exemple de la base MetFaces.

2.5.3 Disney Dataset

est une collection d'images de visages de personnages animés populaires de Disney, notamment Mickey Mouse, Minnie Mouse, Donald Duck, Goofy et d'autres comme le montre la figure 2.7. L'ensemble de données comprend plus de 100 000 images, chacune d'une taille de 256x256 pixels, et couvre une large gamme d'émotions, d'expressions et de poses. Les images sont de haute qualité et mettent l'accent sur les détails du visage, ce qui les rend adaptées à la reconnaissance faciale, à l'analyse des expressions et à d'autres tâches connexes. L'ensemble de données est souvent utilisé dans la recherche sur l'apprentissage automatique et la vision par ordinateur, en particulier dans les modèles génératifs tels que les GAN pour créer de nouveaux visages de personnages de Disney.[31]

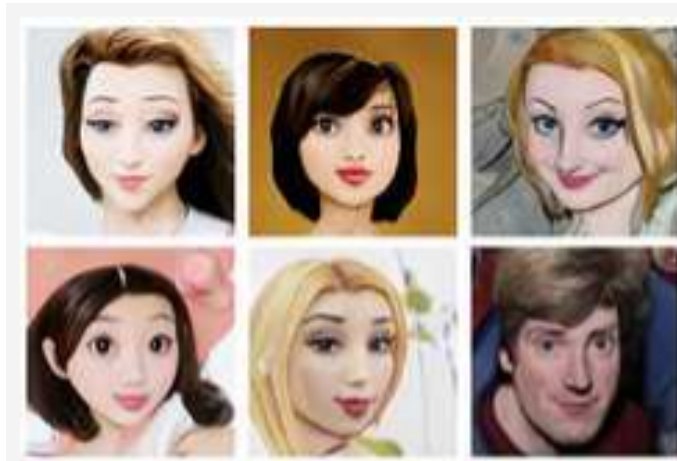


FIGURE 2.7 – Exemples de Disney Dataset [31].

2.5.4 CartoonSet Dataset

Cartoon Set est une collection d'images aléatoires d'avatars en 2D. Les dessins animés varient en 10 catégories de dessins, 4 catégories de couleurs et 4 catégories de proportions, avec un total de 1013 combinaisons possibles. la bdd fournissons des ensembles de 10 000 et 100 000 dessins animés choisis au hasard et des attributs étiquetés. Chaque visage de dessin animé est composé de 16 éléments, dont 12 attributs faciaux (par exemple, la pilosité faciale, la forme des yeux, etc.) et 4 attributs de couleur (comme la couleur de la peau ou des cheveux) qui sont choisis parmi un ensemble discret de valeurs RGB.[34]



FIGURE 2.8 – Exemple de la base CartoonSet .

2.6 Travaux connexes

2.6.1 Article 1 : "3D Cartoon Face Generation with Controllable Expressions from a Single GAN Image[46] "

Dans cet article les auteurs proposent une nouvelle approche pour générer des visages de dessins animés en 3D avec des expressions contrôlables à l'aide d'une seule image GAN. La méthode proposée utilise un réseau de génération d'images 2D vers des formes 3D pour générer des modèles de visage 3D, puis cartographie les modèles générés sur le domaine d'image GAN pour synthétiser des visages de dessins animés 3D avec les expressions souhaitées. L'approche proposée introduit deux éléments clés dans les méthodes existantes : 1) un réseau de génération de formes 3D qui peut générer des modèles 3D avec différentes expressions faciales, et 2) un réseau de mappage d'expressions faciales qui peut transférer les expressions faciales du domaine de l'image GAN au domaine de la forme 3D. Les auteurs évaluent l'approche proposée sur un ensemble de données de visages de dessins animés en 3D et démontrent qu'elle peut générer des visages de dessins animés en 3D de haute qualité avec des expressions contrôlables à partir d'une seule image GAN. La méthode proposée surpasse les méthodes de pointe existantes en termes de qualité visuelle et de contrôlabilité des expressions. Les auteurs fournissent également des études et des analyses d'ablation pour valider l'efficacité de la méthode proposée.

2.6.2 Article 2 : "XGAN : Unsupervised Image-to-Image Translation for Many-to-Many Mappings[34] "

Dans ce travail les auteurs décrivent une méthode pour la traduction d'images dans un cadre non supervisé pour les cartes de correspondance many-to-many. La méthode proposée utilise un modèle génératif adversaire conditionnel (CGAN) pour apprendre des fonctions de mappage entre les distributions de données d'entrée et

de sortie. Le modèle XGAN utilise une architecture de réseau de neurones profonds pour apprendre les cartes de correspondance entre les domaines d'entrée et de sortie, qui peuvent être différents en termes de style, d'orientation, de texture et de couleur. Le modèle est formé sur des paires d'images, l'une provenant du domaine d'entrée et l'autre du domaine de sortie, sans étiquettes de correspondance. Les résultats expérimentaux montrent que la méthode XGAN est compétitive avec les méthodes d'apprentissage supervisé existantes pour la traduction d'images "many-to-many". Les auteurs ont également montré que leur méthode est capable de produire des traductions de haute qualité pour une variété de tâches de traduction d'images.

2.6.3 Article 3 : "Cartoon Face to Human Face Translation using Contour Loss based CycleGANs[42]"

Les auteurs propose une méthode de traduction des visages de dessins animés en visages humains à l'aide d'un CycleGAN basé sur la perte de contour. Les auteurs affirment que les méthodes existantes ne préservent pas la structure faciale des visages de dessins animés, ce qui conduit à des résultats irréalistes. Ils proposent d'utiliser un terme de perte de contour pour s'assurer que les visages humains générés conservent la structure faciale du dessin animé tout en étant transformés en visages humains réalistes. La méthode proposée se compose de deux générateurs et de deux discriminateurs. Les générateurs apprennent à traduire les images entre les deux domaines, tandis que les discriminateurs font la différence entre les images réelles et les images générées. La perte de contour est ajoutée à la fonction objective du générateur pour encourager l'image générée à conserver la structure faciale originale. Les auteurs utilisent également un terme de perte d'identité pour préserver l'identité de l'image d'entrée pendant la traduction. Les auteurs concluent que leur méthode peut être utilisée pour diverses applications, telles que la création d'avatars réalistes pour les jeux ou l'animation.

2.6.4 Article 4 : "Resolution Dependent GAN Interpolation for Controllable Image Synthesis Between Domains [31]"

Cet article propose une nouvelle méthode de synthèse d'images contrôlable entre domaines à l'aide d'une technique d'interpolation GAN dépendante de la résolution. L'approche consiste à entraîner deux GAN sur des domaines différents, l'un avec des images à plus haute résolution et l'autre avec des images à plus faible résolution, et à interpoler l'espace d'image intermédiaire pour obtenir une synthèse d'image contrôlée entre les domaines. Les auteurs présentent également une nouvelle mesure

de qualité et de diversité des images, appelée indice de séparation (SI), et montre que leur méthode est plus performante que les méthodes existantes basées sur les GAN en termes de qualité visuelle et de diversité des images synthétisées. La méthode proposée a des applications potentielles dans des domaines tels que le transfert de style, l'édition d'images et la réalité augmentée.

2.6.5 Article 5 : "Landmark-guided deformation transfer of template facial expressions for automatic generation of avatar blendshapes [30]"

Cet article présente une méthode pour la génération automatique de modèles de déformation faciale à l'aide de repères faciaux et de transfert de déformation. Les auteurs proposent un pipeline de traitement qui utilise des repères faciaux pour déformer un modèle de référence en une série d'expressions faciales et pour transférer ces déformations à un modèle cible. Les résultats montrent que cette méthode peut produire des modèles de déformation précis et efficaces pour l'animation faciale, en particulier pour les expressions complexes et subtiles. Les auteurs concluent que cette méthode peut être utilisée dans la production de films d'animation et de jeux vidéo, ainsi que dans la recherche en psychologie sociale et cognitive.

2.6.6 Article 6 : " normalize STYLEGAN2 for cartoon face generation [6]"

Les auteurs proposent une méthode pour générer des visages de dessins animés de haute qualité en utilisant StyleGAN2, un modèle génératif de pointe. Les auteurs affinent le modèle StyleGAN2 sur un ensemble de données de visages de dessins animés personnalisé, qu'ils créent en sélectionnant et en recadrant manuellement des images de dessins animés existants.

Ils utilisent ensuite le modèle affiné pour générer un ensemble varié de visages de dessins animés avec des expressions et des styles différents. Pour évaluer la qualité des images générées, ils mènent une étude auprès des utilisateurs et comparent les résultats avec les méthodes existantes de génération de visages de dessins animés. L'étude montre que la méthode proposée est plus performante que les méthodes existantes en termes de qualité visuelle et de diversité des images générées. Les auteurs fournissent également une analyse détaillée du processus de réglage fin.

2.6.7 Article 7 : "Automatic Unpaired Human Character Facial Retargeting Based on Deep learning Approach [19]"

Les auteurs proposent une nouvelle approche pour recibler automatiquement les expressions faciales d'un personnage source sur un personnage cible sans avoir besoin de données d'entraînement appariées. L'approche proposée utilise des méthodes d'apprentissage profond et un pipeline en deux étapes pour le reciblage des expressions faciales. La méthode proposée comprend deux étapes : (1) la génération d'une image grossière avec le modèle StyleGAN, et (2) l'affinage de l'image à l'aide d'un réseau d'affinage perceptuel qui supprime les artefacts et améliore la qualité de l'image. Dans la première étape, la détection des repères faciaux est effectuée sur les personnages Source et cible à l'aide de la bibliothèque dlib. Ensuite, un processus de normalisation des visages est appliqué aux points de repère détectés pour supprimer les différences de pose et d'échelle entre les visages source et cible.

Dans un deuxième temps, un réseau adversarial génératif (GAN) est entraîné pour générer les expressions faciales reciblées. Le GAN se compose d'un générateur et d'un discriminateur, le générateur prenant en entrée les repères faciaux normalisés du personnage source et générant un nouvel ensemble de repères correspondant à l'expression faciale du personnage cible. Le discriminateur est entraîné à faire la distinction entre les points de repère générés et les points de repère cibles de la vérité de terrain.

L'approche proposée a été évaluée sur un ensemble de données de personnages source et cible non appariés, et les résultats ont montré que l'approche est capable de recibler avec précision les expressions faciales du personnage source sur le personnage cible. Les auteurs proposent dans leur approche l'industrie de l'animation pour le reciblage facial automatique

2.6.8 Article 8 : "Normalized Avatar Synthesis Using StyleGAN and Perceptual Refinement [22]"

Les auteurs propose une méthode pour générer des avatars de haute qualité en utilisant StyleGAN et le raffinement perceptuel. Les auteurs notent que les méthodes existantes de génération d'avatars produisent souvent des résultats irréalistes ou déformés. Pour y remédier, ils proposent une approche en deux étapes : d'abord, ils utilisent StyleGAN pour générer un avatar approximatif, puis ils utilisent un réseau de raffinement perceptuel pour ajuster les détails et améliorer le réalisme de l'avatar. Les auteurs évaluent leur approche sur plusieurs ensembles de données et comparent leurs résultats avec plusieurs méthodes existantes. Ils montrent que leur méthode

est capable de générer des avatars plus réalistes et visuellement plus agréables que les méthodes existantes, et que l'étape de raffinement perceptuel est cruciale pour y parvenir.

2.7 Conclusion

le visage joue un rôle prépondérant dans plusieurs applications puisqu'il est l'un des principaux vecteurs de l'émotion et de la communication chez l'humain. De plus en plus présent sur internet, l'avatar peut être source de valeur utilitaire et exponentielle dans plusieurs domaines tel que dans la réalité virtuelle et les jeux vidéo. La capacité de ces techniques de créer des Cartoons et des avatars deviendra dans un futur proche nécessaire, avec les capacités technologiques croissantes.

Chapitre 3

Conception

3.1 Introduction

La génération d'images, qui est le sujet principal de ce mémoire, est une forme de modèle génératif qui, en même temps, appartient au domaine non supervisé de l'apprentissage profond. Il existe différentes approches de la génération d'images. Les plus populaires sont les suivantes : les autoencodeurs variationnels et les réseaux adversariels génératifs.

Dans ce travail, nous nous concentrerons sur les techniques de génération d'images faciales synthétiques à partir des points caractéristiques en générant un dessin animé qui conserve les mêmes traits faciaux que l'image d'entrée. Ce dessin animé peut être utilisé comme avatar dans diverses applications ou comme image clé dans des animations.

3.2 Objectifs

L'objectif de notre travail est de concevoir et de mettre en œuvre une application qui permet de générer un cartoon qui conserve les mêmes traits faciaux de l'image d'entrée comme le montre dans la figure 3.1. Pour cela nous proposons un système qui se base sur les modèles de l'apprentissage profond. Après l'extraction des points caractéristique de visage de l'image d'entrée, Nous utilisons encodeur pour extraire un vecteur latent(512 valeur réel) , qui sera utiliser comme entrée du générateur du modèle Stylegan3.



FIGURE 3.1 – L'objectif de l'application

3.3 Architecture du système proposé

StyleGAN3 est un modèle de génération d'images basé sur l'apprentissage automatique. Il permet de créer des photographies réalistes et de haute qualité en contrôlant plusieurs variables telles que l'édition d'images et de vidéos[3], le style, la posture et les attributs de l'objet[17]. Pour cette raison que nous l'avons choisi comme modèle de base. Notre système est composé de trois modules :

1. Module de détection de visage et d'extraction des points caractéristiques.
2. Module de mappage, qui est un encodeur permettant de générer un vecteur latent.
3. La génération de cartoon, qui utilise le générateur pré-entraîné de StyleGan3, auquel est appliqué un apprentissage par transfert.

La figure(3.2) illustre cette architecture

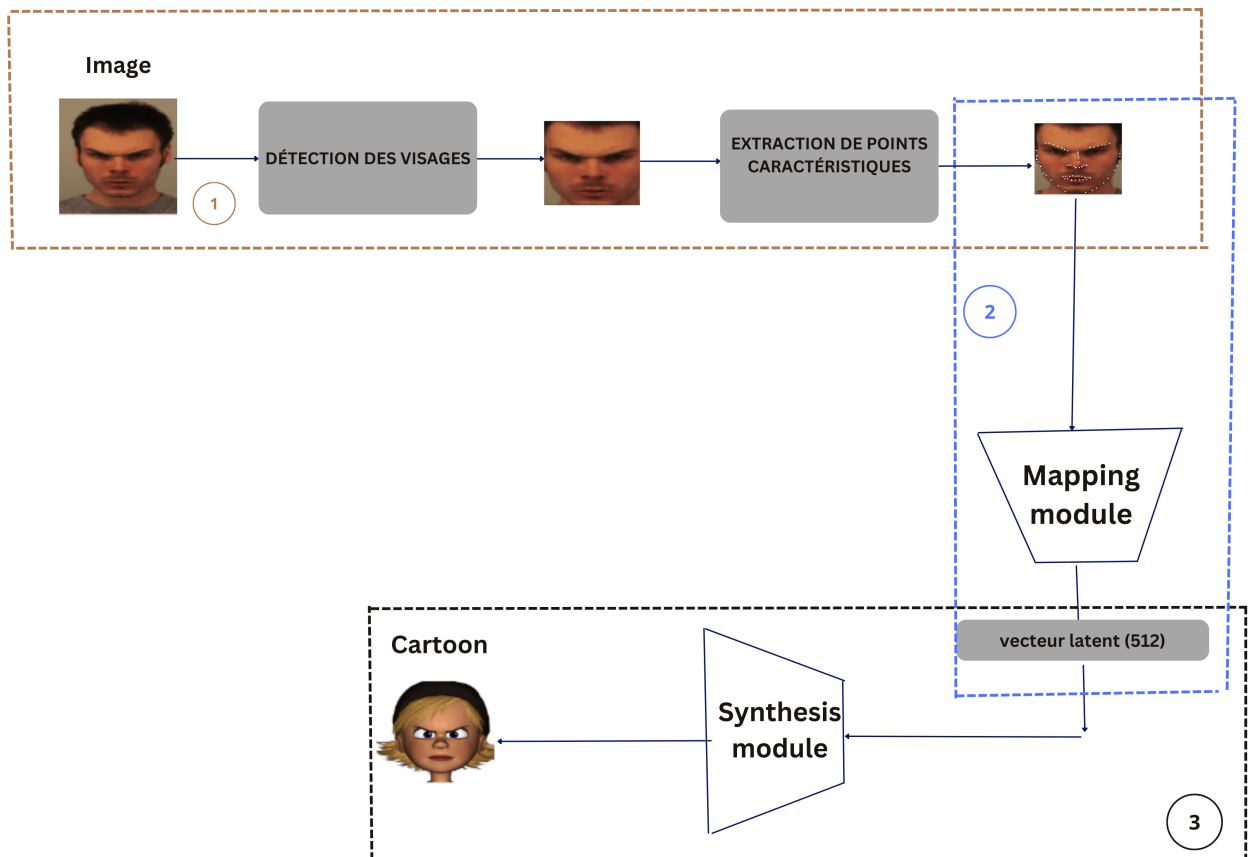


FIGURE 3.2 – Architecture générale de l'application

3.4 Détection de visage et Extraction les points caractéristiques

La détection de visage consiste à localiser l'emplacement du visage dans l'image. Le détecteur de visage est un outil qui permet de repérer le visage dans une image et de le représenter sous la forme d'une boîte de délimitation ou d'une valeur rectangulaire.

La bibliothèque Dlib est une technique de vision par ordinateur qui permet de détecter 68 points de repère faciaux, tels que les yeux, le nez et la bouche. Après avoir localisé le visage dans l'image, La détection des points de repère faciaux nous fournit des informations précieuses sur les différentes caractéristiques d'un visage.

Les points de repère faciaux numérotés de 1 à 68 sont visibles dans la figure 3.4.

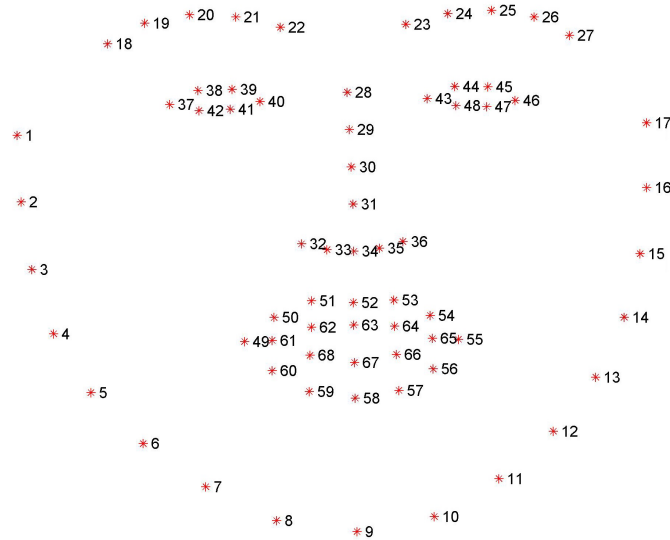


FIGURE 3.3 – Les 68 points caractéristiques détecter par Dlib[39].

La détection des points caractéristiques passe par deux étapes essentielles :

.Détection de visage : La première technique trouve un visage humain et renvoie une valeur , qui est un rectangle.

.Repère du visage : Après avoir déterminé la position d'un visage dans une image, nous devons parcourir des points à l'intérieur de ce rectangle.

Il existe plusieurs méthodes de détection des visages, mais nous nous concentrons sur une seule dans ce billet : L'approche de Dlib. Par exemple, Opencv utilise des cascades de LBP, alors que HAAR et Dlib utilisent les algorithmes HOG (Histogramme de Gradients Orientés) et SVM (Support Vector Machine)[39].

La Figure 3.5 illustre la première étape du processus.



FIGURE 3.4 – la première étape du notre processus.

3.5 Module de mappage(Mapping module)

Après l'extraction des points caractéristique du l'image , Nous utilisons l'enco-
deur pour extraire un vecteur latent avec d'une taille de 512.

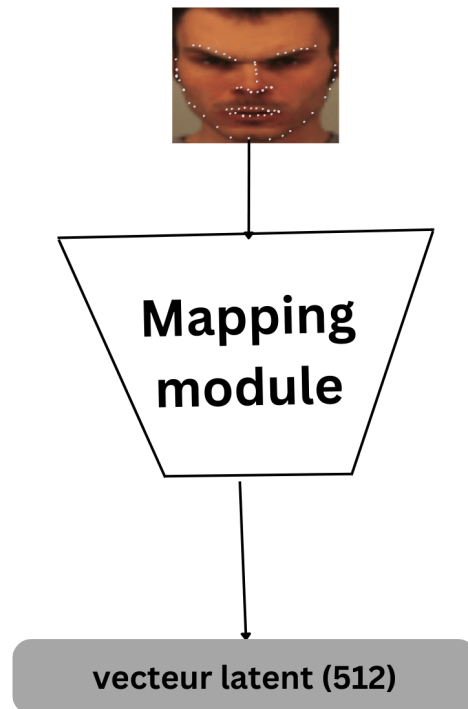


FIGURE 3.5 – Le module de mappage

La figure 3.5 illustre le module de mappage. L'autoencodeur (AE) utilise donc, ces points caractéristiques comme entrée avec une résolution de 64x64. Le modèle autoencodeur, que nous proposons, est composé de trois couches de convolution, chacune suivi d'une couche de Max-pooling, le résultat de ces couches est ensuite aplati, et en tant que dernière couche de l'encodeur, une couche entièrement connectée, doté de la fonction d'activation Softmax. Pour le décodeur, les couches sont inversées. Cette architecture est résumée dans la figure 3.6.

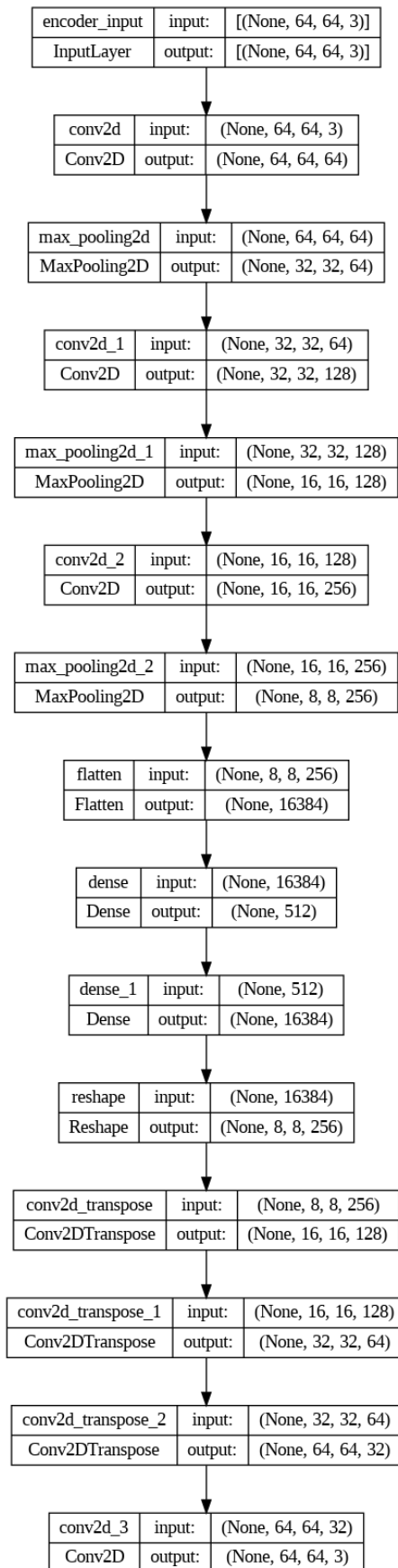


FIGURE 3.6 – Architecture détaillée de l'autoencoder

3.6 Le module de génération d'image(Synthesis module)

La génération des images se fait à l'aide du puissant générateur du modèle StyleGAN3. Ce générateur nous permet de créer un cartoon en utilisant en entrée un vecteur latent généré par le module de mappage. Il est important de noter que nous utilisons une étape d'apprentissage par transfert pour le générateur pré-entraîné de StyleGAN3.

La Figure 3.7 illustre la deuxième étape du processus.

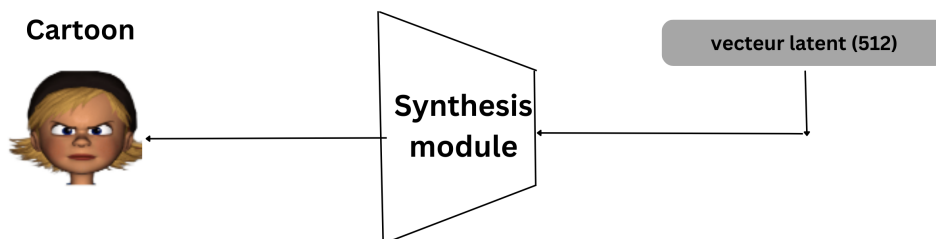


FIGURE 3.7 – la troisième étape du notre processus

Nous avons initialement prévu d'utiliser StyleGAN2, cependant, nous avons rencontré un obstacle car la version TensorFlow 1 n'était plus disponible après la mise à jour de Colab en Mai 2023.

Comme nous l'avons déjà mentionné, StyleGAN3, développé par NVIDIA, est un modèle génératif avancé pour la synthèse d'images et de vidéos de haute qualité. Il résout le problème du "collage de texture" lors de la transition de morphing dans StyleGAN2. Les améliorations incluent un nouvel algorithme d'apprentissage, une architecture plus efficace et un nouveau jeu de données d'apprentissage. StyleGAN3 génère des images réalistes et détaillées, avec des applications potentielles dans l'art, la mode et les jeux.

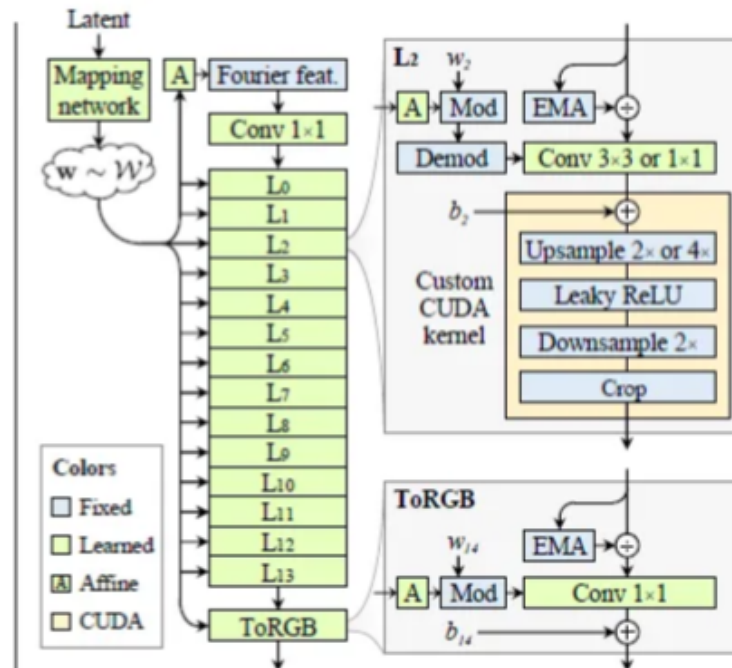


FIGURE 3.8 – L'architecture du Stylegan3[25].

3.7 Conclusion

Pour réaliser la tâche de génération d'un visage de dessin animé (cartoon) à partir d'un visage réel, nous avons conçu un système basé sur trois étapes essentielles : la détection et l'extraction des points caractéristiques du visage source, la génération d'un vecteur latent spécifique au visage d'entrée, et enfin la génération d'un cartoon avec les mêmes caractéristiques que le visage source. Deux phases d'apprentissage sont nécessaires : l'apprentissage à partir de zéro pour l'autoencodeur, et l'apprentissage par transfert pour le générateur pré-entraîné de StyleGan3.

Chapitre 4

Implémentation

4.1 Introduction

Dans ce chapitre, nous présentons les étapes de mise en œuvre du système de génération d'images basée sur les points de caractéristique. Notre objectif est de concevoir un système qui commence par la détection d'un visage dans une image et l'extraction de ses points caractéristiques. Pour cela, le système proposé est décrit dans le chapitre précédent. Ensuite, nous décrivons les différents paramètres utilisés dans la mise en œuvre et l'apprentissage du système proposé. Enfin, nous présentons un aperçu des résultats obtenus.

4.2 Environnement de développement

Notre application a été créée à l'aide de Google Colab qui offre une grande capacité de traitement pour l'apprentissage automatique.

4.2.1 Google Colaboratory

Google Colaboratory, parfois connu sous le nom de Colab, est un service cloud gratuit fourni par Google, basé sur Jupyter Notebook et destiné à la formation et à la recherche en matière d'apprentissage automatique. Les modèles d'apprentissage automatique peuvent être formés directement dans le nuage à l'aide de cette plateforme. Ainsi, à part un navigateur, nous n'avons pas besoin d'installer quoi que ce soit sur notre machine[26].



FIGURE 4.1 – Interface de Colab

Colab vous permet d'utiliser des accélérateurs matériels tels que les GPU et les TPU pour accélérer l'exécution d'opérations particulières telles que l'apprentissage en profondeur. Cela permet d'accélérer le traitement des tâches complexes sur le plan informatique.

4.3 Langage de programmation et bibliothèques utilisées

4.3.1 Python

Python est un langage de programmation général de haut niveau qui est largement utilisé. Le langage de programmation Python (dernière version Python 3) est utilisé dans le développement web, les applications d'apprentissage automatique et toutes les technologies logicielles de pointe. Python est utilisé par pratiquement tous les géants de l'internet, notamment Google, Amazon, Facebook, Instagram, Dropbox, Uber, etc. Python est aujourd'hui le langage de programmation de haut niveau polyvalent le plus utilisé, permettant la programmation dans les paradigmes orienté objet et procédural. Les applications Python sont généralement plus petites que celles écrites dans d'autres langages de programmation tels que Java. Les programmeurs doivent taper très peu, et l'exigence d'indentation du langage garantit que leur code est toujours compréhensible.[11]

4.4 Les bibliothèques utilisées

4.4.1 TensorFlow

TensorFlow est une plateforme d'apprentissage automatique open source complète. Elle propose un écosystème riche et flexible d'outils, de bibliothèques et de

ressources communautaires qui permettent aux universitaires de repousser les limites de l'apprentissage automatique et aux développeurs de créer et de déployer simplement des applications basées sur l'apprentissage automatique. TensorFlow a été créé par des chercheurs et des ingénieurs de l'équipe Google Brain de l'organisation Machine Intelligence Research de Google pour entreprendre des recherches sur l'apprentissage automatique et les réseaux neuronaux profonds. Le système est suffisamment large pour être utilisé dans un large éventail de domaines. TensorFlow offre des API Python et C++ robustes, ainsi qu'une compatibilité ascendante non garantie avec d'autres langages [12].

4.4.2 Keras

Keras est une API de réseau neuronal de haut niveau basée sur Python qui s'interface avec TensorFlow, CNTK et Theano. Il a été créé dans le but de permettre une expérimentation rapide. La capacité de passer d'un concept à un résultat en un minimum de temps est essentielle pour une recherche efficace. Keras est un bon choix si vous voulez une bibliothèque qui : permet un prototypage rapide et facile (grâce à sa convivialité, sa modularité et son extensibilité). Les réseaux convolutifs et récurrents, ainsi que les mélanges des deux, sont pris en charge. fonctionne de manière transparente sur le CPU et le GPU[2].

4.4.3 Numpy

NumPy est la bibliothèque Python fondamentale pour le calcul scientifique. Il s'agit d'une bibliothèque Python qui comprend un objet tableau multidimensionnel, divers objets dérivés (tels que les tableaux masqués et les matrices) et une variété de routines permettant d'effectuer des opérations rapides sur les tableaux, telles que des opérations mathématiques, logiques, de manipulation de formes, de tri, de sélection, d'E/S, de transformée de Fourier discrète, d'algèbre linéaire de base, d'opérations statistiques de base, de simulation aléatoire, et bien plus encore[47].

4.4.4 Matplotlib

Matplotlib est une bibliothèque de dessin open-source qui prend en charge un large éventail de formats de dessin. Quelques lignes de code suffisent pour créer des graphiques, des histogrammes, des diagrammes à barres et d'autres types de diagrammes. Elle est couramment utilisée dans les serveurs d'applications web, les scripts shell et les scripts Python [41].

4.4.5 Gradio

Gradio est une bibliothèque Python open-source qui offre la possibilité de créer facilement des composants d'interface utilisateur personnalisables et conviviaux pour vos modèles d'apprentissage automatique, des API ou même des fonctions Python, en utilisant seulement quelques lignes de code. Avec Gradio, vous pouvez intégrer l'interface graphique directement dans votre environnement de développement, tel que le carnet Jupyter, ou la partager sous forme de lien avec d'autres utilisateurs. Cela facilite la visualisation et l'interaction avec vos modèles et fonctionnalités, rendant leur utilisation plus accessible et conviviale[23].

4.5 Implémentation et apprentissage

4.5.1 Bases de données utilisées

Nous avons utilisé l'ensemble de données FER-G-DB avec expressions faciales annotées contenant 55 769 images de visages annotées de six personnages de 256*256 pixels en format PNG, étiquetée en 7 classes : 0=colère,1=dégoût, 2=peur, 3=joie,4=triste, 5=surprise, 6=neutre. La figure4.2 montre la distribution des images d'une caractere par classe dans la base de données FER-G-DB.

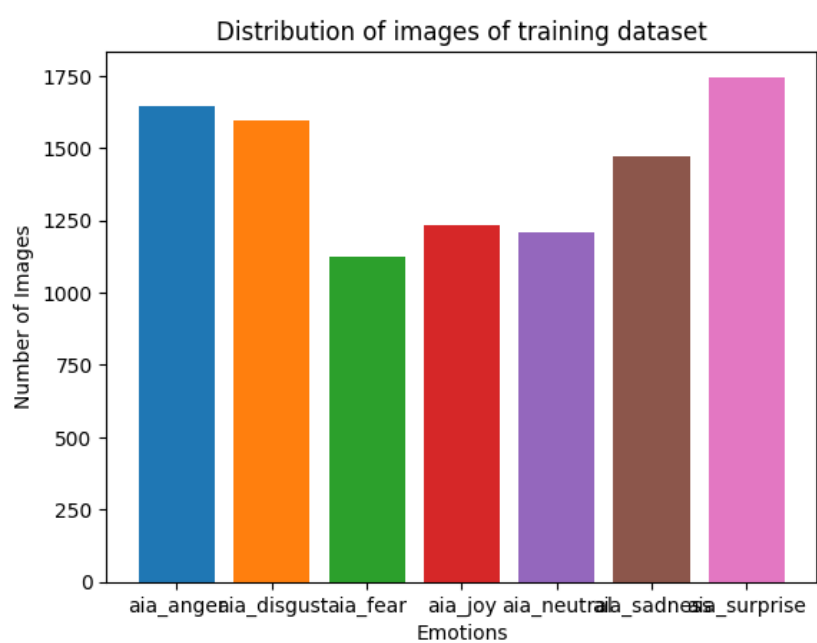


FIGURE 4.2 – Nombre d'images par émotions de caractère aia de la base de données FER-G

Les tests ont été réalisés en utilisant les bases de données CK+[21] et CelebA[20] pour évaluer les performances.

4.5.2 Implémentation de l’autoencodeur

Pendant la phase d’apprentissage, nous avons testé plusieurs configurations basées sur la modification de certains paramètres du réseau tels que :

La taille du lot
 nombre d’épochs
 le taux d’apprentissage
 le nombre d’itération.

Différentes configurations ont été expérimentées, elles sont résumées dans le tableau 4.1 suivant :

Test	Nombre d’époch	Batch size
Test 1	100	64
Test 2	100	32
Test 3	50	32
Test 4	50	64
Test 5	200	32

TABLE 4.1 – Les paramètres d’apprentissage de l’autoencodeur

4.5.3 Implémentation du générateur

Avant d’utiliser le générateur StyleGan3, il est nécessaire de le former sur notre ensemble de données. Voici les étapes suivies pour accomplir cet apprentissage :

1.Préparation de la base de données :

- Réduction de la taille de la base de données FER-G-DB à 2067 images.
- Redimensionnement des images de la base de données en 64*64, Conversion des images en format RGB.

2- Le lancement de l’apprentissage du modèle :

```
-outdir=/content/out/results
-cfg=stylegan3-r
-data=/content/mydataset.zip
-gpus=1
```

```

-batch=32
-batch-gpu=8
-gamma=6.6
-mirror=1
-kimg=1000
-snap=50

```

4.6 Résultats et discussions

Après avoir effectué plusieurs essais, nous avons obtenu une précision de 77% pour l'expérience 1. Il est à noter que l'expérience 2 a atteint une précision supérieure de 82%, tandis que l'expérience 5 a obtenu une précision de 78%. Les courbes de précision et de pertes de ces expérimentations sont présentées dans les figures (4.4, 4.5). ces expérimentations présentent un sur apprentissage. Pour cette raison nous avons choisi l'expérimentation 5 (figure (4.3)).

Les résultats sont synthétisés dans le tableau 4.1 ci-dessous :

Test	Résultat
Test 1	Evaluation loss : 0.10 Evaluation accuracy : 0.78
Test 2	Evaluation loss : 0.05 Evaluation accuracy : 0.87
Test 3	Evaluation loss : 0.020 Evaluation accuracy : 0.80
Test 4	Evaluation loss : 0.032 Evaluation accuracy : 0.81
Test 5	Evaluation loss : 0.030 Evaluation accuracy : 0.77

TABLE 4.2 – Les Résultats d'évaluation réalisées

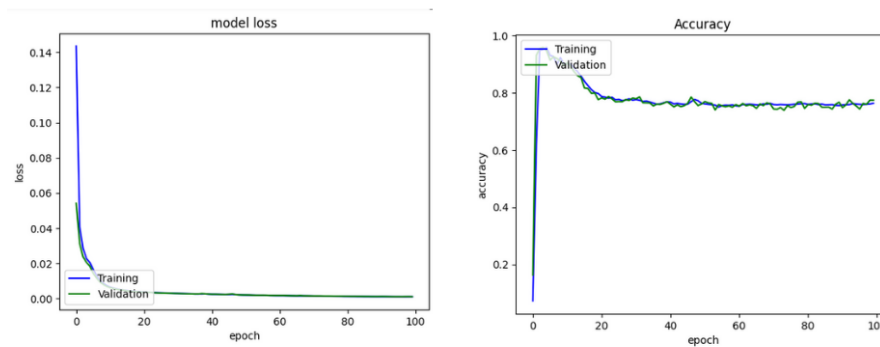


FIGURE 4.3 – Les courbes de précision et de perte de l'expérimentation 1

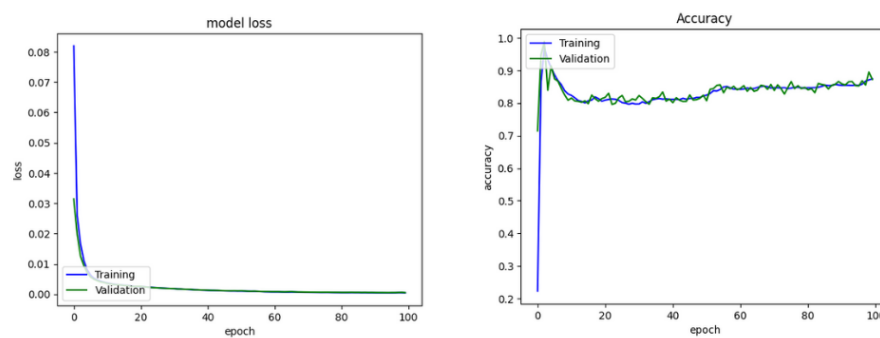


FIGURE 4.4 – Les courbes de précision et de perte de l'expérimentation 2

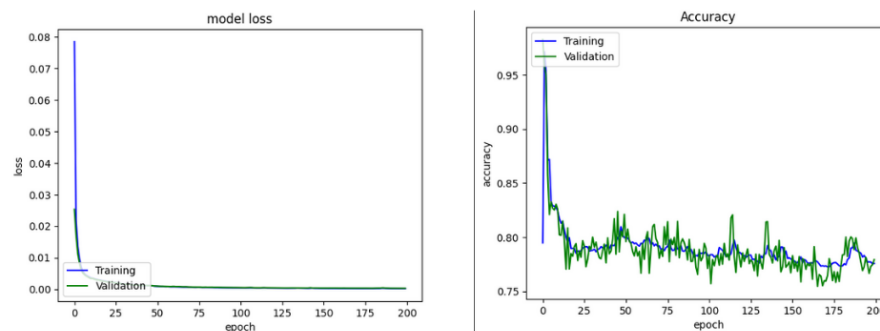


FIGURE 4.5 – Les courbes de précision et de perte de l'expérimentation 5

4.7 Quelques résultats

Après l'étape de l'évaluation du modèle, nous l'avons testé le générateur sur des images de l'ensemble de données CK+ et celebA (figure 4.6).


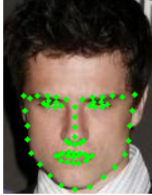




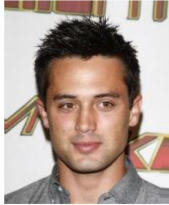
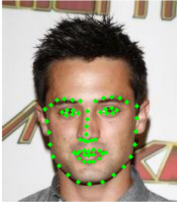


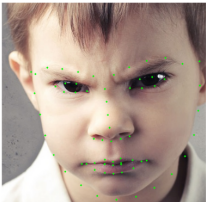

Visage d'entree	point de caractéristique	Cartoon généré
		
		
		
		

FIGURE 4.6 – Quelques exemples de génération de cartoon.

Après avoir effectué plusieurs entraînements de notre modèle sur différents comptes Google Colab, nous avons obtenu les résultats présentés dans le figure 4.6. Nous observons que les images générées ont une qualité inférieure, ce qui est dû à la nécessité d'un apprentissage avec plus de cartes graphiques (GPU) puissantes et plusieurs jours d'entraînement. Nous avons obtenu à la fois de mauvais résultats et de bons

résultats, où les expressions faciales des visages générés semblent se rapprocher des expressions des visages d'entrée.

4.8 Conclusion

Dans ce chapitre, nous avons présenté les résultats des différentes expériences effectuées sur l'extraction des points caractéristique et sur la génération des images de visages de cartoon. Les résultats obtenus sont prometteurs et les performances de notre système peuvent être améliorées en utilisant un ensemble de données plus vaste pour l'apprentissage du générateur StyleGAN3. De plus, il serait bénéfique d'avoir accès à des machines et des outils plus performants, tels que Colab Pro, qui ne sont pas encore disponibles en Algérie, afin de générer des images de haute qualité avec des capacités exceptionnelles.

Conclusion générale

Dans le domaine de translation d'image à image, les modèles existants peuvent générer des images de visages cibles réalistes, mais il est difficile de conserver les caractéristiques faciales de l'image source. Pour y remédier, nous proposons dans ce mémoire une méthode de génération d'image qui consiste à générer des images de visages de dessins animés à partir d'images de visages réels, en affinant le générateur de stylegan3 pré-entraîné. L'objectif est donc de créer des dessins animés uniques et attrayants qui ressemblent d'avantage à la personne réelle et qui peuvent enrichir l'expérience de l'utilisateur.

Notre système se compose de trois modules : tout d'abord, le module de détection et d'extraction des points caractéristiques, qui sert d'entrée au deuxième module, lequel extrait le vecteur latent à l'aide de l'encodeur. Le dernier module génère des images de dessins animés à l'aide du générateur affiné StyleGAN3.

Lors de la mise en œuvre de ce travail, une phase d'apprentissage est nécessaire à la fois pour l'autoencodeur et pour le générateur. L'autoencodeur proposé a été entraîné sur le jeu de données FER-G-DB et après plusieurs expériences, nous avons obtenu une précision de 78%. L'apprentissage par transfert a été appliqué à StyleGAN3. Ce dernier a été entraîné sur un sous-ensemble de la base de données FER-G-DB.

Tout au long de la conception et de la mise en œuvre de ce travail, nous avons été confrontés à un certain nombre de problèmes. En effet, deux problèmes majeurs ont ralenti notre progression dans la phase de génération des dessins animés. Le premier problème était lié aux versions des bibliothèques utilisées, notamment en ce qui concerne le générateur pré-entraîné. Le second problème était le manque de ressources GPU, qui a affecté la vitesse d'exécution du système.

La performance de notre système pourrait être encore améliorée dans des projets futurs. Dans cette optique, nous proposons les points suivants à étudier :

- Proposer une approche d'apprentissage innovante utilisant une nouvelle fonction de perte pour le générateur StyleGAN3.
- Optimiser l'apprentissage du système de génération en tirant parti de plusieurs GPU.
- Augmenter la qualité de l'apprentissage en utilisant plusieurs bases de données pour assurer une généralisation optimale.

Bibliographie

- [1] 1min30: *1min30*, 2023. <https://www.1min30.com/dictionnaire-du-web/avatar-web>.
- [2] actuaia, 2023. <https://www.actuaia.com/keras/>.
- [3] Alaluf, Yuval, Or Patashnik, Zongze Wu, Asif Zamir, Eli Shechtman, Dani Lischinski et Daniel Cohen-Or: *Third time's the charm? image and video editing with stylegan3*. Dans *Computer Vision—ECCV 2022 Workshops : Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 204–220. Springer, 2023.
- [4] Alzubaidi, Laith, Jinglan Zhang, Amjad J Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, José Santamaría, Mohammed A Fadhel, Muthana Al-Amidie et Laith Farhan: *Review of deep learning : Concepts, CNN architectures, challenges, applications, future directions*. Journal of big Data, 8(1) :1–74, 2021.
- [5] Aneja, Deepali, Alex Colburn, Gary Faigin, Linda Shapiro et Barbara Mones: *Modeling stylized character expressions via deep learning*. pages 136–153, 2017.
- [6] Back, Jihye: *Fine-tuning stylegan2 for cartoon face generation*. arXiv preprint arXiv :2106.12445, 2021.
- [7] Buhrmester, Michael, Tracy Kwang et Samuel D Gosling: *Amazon's Mechanical Turk : A new source of inexpensive, yet high-quality, data?* Perspectives on psychological science, 6(1) :3–5, 2011.
- [8] DIALLO, Nene Adama Dian: *La reconnaissance des expressions faciales*. 2019.
- [9] Dif, Nassima: *L'apprentissage profond pour le traitement des images*, 2020.
- [10] Garnier, Marion et Ingrid Poncin: *L'avatar en marketing : synthèse, cadre intégrateur et perspectives*. Recherche et Applications en Marketing (French Edition), 28(1) :92–123, 2013.

- [11] geeksforgeeks: *Python Programming Language*, 2019. <https://www.geeksforgeeks.org/python-programming-language>.
- [12] github: *tensorflow*, 2023. <https://github.com/tensorflow/tensorflow>.
- [13] Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville et Yoshua Bengio: *Generative adversarial networks*. *Communications of the ACM*, 63(11) :139–144, 2020.
- [14] Grigorev, Artur, Karim Isakov, Anastasia Ianina, Renat Bashirov, Ilya Zakharkin, Alexander Vakhitov et Victor Lempitsky: *Stylepeople : A generative model of fullbody human avatars*. pages 5151–5160, 2021.
- [15] Hardy, Corentin: *Contribution au développement de l'apprentissage profond dans les systèmes distribués*, 2019.
- [16] Karras, Tero, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen et Timo Aila: *Training generative adversarial networks with limited data*. *Advances in neural information processing systems*, 33 :12104–12114, 2020.
- [17] Karras, Tero, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen et Timo Aila: *Alias-free generative adversarial networks*. *Advances in Neural Information Processing Systems*, 34 :852–863, 2021.
- [18] Karras, Tero, Samuli Laine et Timo Aila: *A style-based generator architecture for generative adversarial networks*. pages 4401–4410, 2019.
- [19] Kim, Seonghyeon: *Automatic unpaired human character facial retargeting based on deep learning approach*. 2021.
- [20] Liu, Ziwei, Ping Luo, Xiaogang Wang et Xiaoou Tang: *Deep Learning Face Attributes in the Wild*. Dans *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.
- [21] Lucey, Patrick, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar et Iain Matthews: *The Extended Cohn-Kanade Dataset (CK+) : A complete dataset for action unit and emotion-specified expression*. Dans *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 94–101, 2010.

- [22] Luo, Huiwen, Koki Nagano, Han Wei Kung, Qingguo Xu, Zejian Wang, Lingyu Wei, Liwen Hu et Hao Li: *Normalized avatar synthesis using stylegan and perceptual refinement*. Dans *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11662–11672, 2021.
- [23] machinelearningnuggets, 2023. <https://www.machinelearningnuggets.com/gradio-tutorial/>.
- [24] medium: *medium*, 2023. <https://medium.com/analytics-vidhya/understanding-the-stylegan-and-stylegan2-architecture-add9e992747d>.
- [25] Melnik, Andrew, Maksim Miasayedzenkau, Dzianis Makarovets, Dzianis Pirsh-tuk, Eren Akbulut, Dennis Holzmann, Tarek Renusch, Gustav Reichert et Helge Ritter: *Face Generation and Editing with StyleGAN : A Survey*. arXiv preprint arXiv :2212.09102, 2022.
- [26] Michel, Henri et Olivier: *Le guide ultime*, Nov 2019. <https://ledatascientist.com/google-colab-le-guide-ultime/>.
- [27] Milgram, Paul, Haruo Takemura, Akira Utsumi et Fumio Kishino: *Augmented reality : A class of displays on the reality-virtuality continuum*. 2351 :282–292, 1995.
- [28] Nam, Yong Hyun et Ji Yong Park: *The understanding of avatar design : Various types and styles of avatars and the design*. *Journal of The Korea Society of Illustration*, 18(2009) :91–102, 2009.
- [29] Noor, Noor Adibah Najihah Mat et Norhaida Mohd Suaib: *Facial Expression Transfer using Generative Adversarial Network : A Review*. 864(1) :012077, 2020.
- [30] Onizuka, Hayato, Diego Thomas, Hideaki Uchiyama et Rin ichiro Taniguchi: *Landmark-guided deformation transfer of template facial expressions for automatic generation of avatar blendshapes*. pages 0–0, 2019.
- [31] Pinkney, Justin NM et Doron Adler: *Resolution dependent gan interpolation for controllable image synthesis between domains*. arXiv preprint arXiv :2010.05334, 2020.
- [32] Puustinen, Minna: *Aider et être aidé : l'importance de la notion d'aide dans les dispositifs d'apprentissage en ligne*. *Revue internationale des technologies en pédagogie universitaire*, 9(3) :6–9, 2012.

- [33] Ranganathan, Varun et S Natarajan: *A new backpropagation algorithm without gradient descent*. arXiv preprint arXiv :1802.00027, 2018.
- [34] Royer, Amélie, Konstantinos Bousmalis, Stephan Gouws, Fred Bertsch, Inbar Mosseri, Forrester Cole et Kevin Murphy: *Xgan : Unsupervised image-to-image translation for many-to-many mappings*. Dans *Domain Adaptation for Visual Understanding*, pages 33–49. Springer, 2020.
- [35] ryte: *ryte*, 2023. <https://fr.ryte.com/wiki/Avatar>.
- [36] Sarker, Iqbal H: *Deep learning : a comprehensive overview on techniques, taxonomy, applications and research directions*. SN Computer Science, 2(6) :1–20, 2021.
- [37] SEKKIL, Hicham Mohamed et Mahmoud MEBROUKI: *Etude comparative entre les différentes architectures des réseaux de neurones convolutifs (CNNs) pour la détection de la fatigue du conducteur.*, 2021.
- [38] Sharma, Sagar, Simone Sharma et Anidhya Athaiya: *Activation functions in neural networks*. towards data science, 6(12) :310–316, 2017.
- [39] Shekharpandey: *Dlib 68 points face landmark detection with opencv and python*, May 2023. <https://www.studytonight.com/post/dlib-68-points-face-landmark-detection-with-opencv-and-python>.
- [40] Sherman, William R et Alan B Craig: *Understanding virtual reality : Interface, application, and design*, 2018.
- [41] simplilearn: *matplotlib*, 2023. <https://www.simplilearn.com/tutorials/python-tutorial/matplotlib>.
- [42] Singhal, Mayank et Ritu Agarwal: *Cartoon Face to Human Face Translation using Contour Loss based CycleGAN*. pages 1–6, 2021.
- [43] Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever et Ruslan Salakhutdinov: *Dropout : a simple way to prevent neural networks from overfitting*. The journal of machine learning research, 15(1) :1929–1958, 2014.
- [44] Tanielian, Ugo: *Generative Adversarial Networks : theory and practice*, 2021.
- [45] Thomas Neff, BSc.

- [46] Wang, Hao, Guosheng Lin, Steven CH Hoi et Chunyan Miao: *3D Cartoon Face Generation with Controllable Expressions from a Single GAN Image*. arXiv preprint arXiv :2207.14425, 2022.
- [47] whatisnumpy: *numpy*, 2023. <https://numpy.org/doc/stable/user/whatisnumpy.html>.
- [48] Zhang, Haining et Seung Ki Moon: *Reviews on machine learning approaches for process optimization in noncontact direct ink writing*. ACS Applied Materials & Interfaces, 13(45) :53323–53345, 2021.
- [49] Zhang, Jianfeng, Zihang Jiang, Dingdong Yang, Hongyi Xu, Yichun Shi, Guoxian Song, Zhongcong Xu, Xinchao Wang et Jiashi Feng: *Avatargen : a 3d generative model for animatable human avatars*. pages 668–685, 2023.
- [50] Zhang, Jiawei: *Gradient descent based optimization algorithms for deep learning models training*. arXiv preprint arXiv :1903.03614, 2019.
- [51] Zouinar, Moustafa: *Évolutions de l'Intelligence Artificielle : quels enjeux pour l'activité humaine et la relation Humain-Machine au travail ?* Activités, (17-1), 2020.