



RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ IBN KHALDOUN - TIARET

MÉMOIRE

Présenté à :
FACULTÉ DES MATHÉMATIQUES ET DE L'INFORMATIQUE
DÉPARTEMENT DE L'INFORMATIQUE

Pour l'obtention du diplôme de :
MASTER

Spécialité : Génie Informatique

Par :
Belkhit Souhila
Cheriet Amira Nihad Intissar

Sur le thème :

**Détection d'anomalies sur les images médicales en
utilisant les Réseaux Antagonistes Génératifs (GAN)**

Soutenu publiquement le 12/06/2024 à Tiaret devant le jury composé de :

M. Belarbi Mostefa	Prof. Université de Tiaret	Président
M. Baghdadi Mohamed	MCA Université de Tiaret	Encadrant
M. Gafour Yacine	MCA Université de Tiaret	Examinateur

2023-2024

Remerciements

Nous remercions Dieu le tout puissant de nous avoir donné la santé et la volonté d'entamer et de terminer ce mémoire.

Tout d'abord, ce travail ne serait pas aussi riche et n'aurait pas pu avoir le jour sans l'aide et l'encadrement de Mr. Baghdadi Mohamed nous le remercions pour la qualité de son encadrement exceptionnel, pour sa patience, sa rigueur et sa disponibilité durant notre préparation de ce mémoire

Nous tenons également à remercier les membres du jury Mr.Belarbi Mostefa, Mr.Gafour Yacine qui ont bien voulu lire et examiner notre travail.

Un merci spécial à tous ceux qui nous ont soutenus pour terminer ce travail.

Dédicace

*À nos parents,
À nos familles,
À nos amies.*

Résumé

Le cancer du foie, un type de cancer courant qui se développe à partir des cellules hépatiques, représente un défi majeur en matière de diagnostic en raison de ses symptômes souvent asymptomatiques aux premiers stades. L'imagerie médicale, notamment la tomодensitométrie (CT scan), joue un rôle crucial dans le diagnostic de cette maladie, mais l'analyse manuelle des images est une tâche complexe et chronophage. Dans notre travail, nous avons exploité les réseaux antagonistes génératifs (GAN), en particulier les modèles DCGAN, pour générer des images synthétiques de foie et améliorer la détection des anomalies associées au cancer du foie. En utilisant une architecture spécifique de GAN développée pour les images CT de foie, nous avons pu produire des images visuellement plausibles qui ressemblent aux images réelles issues des CT scans, et notre modèle de détection a montré une capacité efficace à distinguer les images de foie cancéreux des images normales. Les résultats de notre travail montrent que les GAN, et en particulier les DCGAN, peuvent produire des images synthétiques de haute qualité et améliorer la détection précoce des anomalies hépatiques, ce qui est crucial pour une prise en charge efficace du cancer du foie.

Mots-clés : Imagerie médicale, Tomодensitométrie (CT scan), Cancer du foie, Deep Learning, Réseaux antagonistes génératifs, Réseaux de neurons convolutifs, DCGAN, TensorFlow, Keras.

Abstract

Liver cancer, a common type of cancer that develops from liver cells, represents a major diagnostic challenge due to its often-asymptomatic nature in the early stages. Medical imaging, particularly computed tomography (CT scan), plays a crucial role in diagnosing this disease, but manual image analysis is a complex and time-consuming task. In our work, we leveraged generative adversarial networks (GAN), particularly DCGAN models, to generate synthetic liver images and improve the detection of anomalies associated with liver cancer. By using a specific GAN architecture developed for liver CT images, we were able to produce visually plausible images that resemble real CT scans, and our detection model demonstrated an effective ability to distinguish between cancerous and normal liver images. The results of our work show that GANs, particularly DCGANs, can produce high-quality synthetic images and enhance the early detection of liver anomalies, which is crucial for the effective management of liver cancer.

Keywords : Medical imaging, Computed tomography (CT scan), Liver cancer, Deep Learning, Generative adversarial networks, Convolutional neural networks, DCGAN, TensorFlow, Keras.

Table des matières

Remerciements	i
Dédicace	ii
Résumé	iv
Table des matières	vii
Table des figures	ix
Acronymes	0
Introduction générale	1
1 Généralités	2
1.1 Introduction	2
1.2 Imagerie médicale	2
1.2.1 Historique de l'imagerie médicale	2
1.2.2 Les différentes technologies d'imagerie médicale	3
1.2.2.1 Les rayons X	4
1.2.2.2 Le scanner (tomodensitométrie)	4
1.2.2.3 Radiographie	5
1.2.2.4 L'imagerie par résonance magnétique (IRM)	5
1.2.3 L'importance de l'imagerie médicale	6
1.2.4 Le potentiel de l'IA pour l'imagerie médicale	6
1.3 Traitement d'images	6
1.3.1 Définition	7
1.3.1.1 L'image	7
1.3.1.2 L'image numérique	7
1.3.2 Les types d'image	7
1.3.2.1 Image binaire	7
1.3.2.2 Image en niveau de gris	7
1.3.2.3 Image couleur (RGB)	8
1.3.2.4 Image indexée	8
1.3.3 Les caractéristiques d'image	9
1.3.4 Segmentation des images	11
1.3.5 Classification des images	11
1.4 Détail sur le cancer du foie	11
1.4.1 Les facteurs de risque	13
1.4.2 Symptômes et diagnostic	13
1.4.2.1 Symptômes	13
1.4.2.2 Le diagnostic	14
1.4.3 Epidémiologie du cancer du foie	14
1.4.4 Traitements du cancer du foie	15
1.4.5 Conclusion	16

2	Apprentissage profond	17
2.1	Introduction	17
2.2	Intelligence artificielle (IA)	17
2.3	Apprentissage automatique (ML)	18
2.3.1	Différents types de l'apprentissage automatique	19
2.3.2	Tâches de l'apprentissage automatique	20
2.3.3	Limitations de l'apprentissage automatique	21
2.4	Réseaux de neurones artificiels	21
2.5	Fonctionnement d'un réseau de neurones artificiel	22
2.5.1	Contexte et historique des réseaux de neurones artificiels	23
2.5.2	Apprentissage et optimisation d'un réseau de neurones	23
2.5.2.1	Fonction de perte (<i>loss function</i>)	23
2.5.2.2	Optimisation avec l'algorithme de la descente de gradient	24
2.6	Apprentissage profond	24
2.6.1	Incorporation de couches cachées	24
2.6.2	Formalisation mathématique d'un réseau de neurones profond	25
2.6.2.1	Fonctions d'activation	26
2.6.2.2	Fonctions de perte pour l'apprentissage profond	27
2.6.2.3	Algorithmes d'optimisation	27
2.7	Réseaux neuronaux convolutifs (CNN)	28
2.7.1	Motivation des CNN	28
2.7.2	Concepts de base des CNN	30
2.7.2.1	Concept de convolution	30
2.7.2.2	Convolution des images	30
2.7.2.3	Convolution transposée	31
2.7.2.4	Concepts <i>padding</i> et <i>stride</i>	32
2.7.3	Applications des CNN	32
2.8	Conclusion	33
3	Réseaux antagonistes génératifs	34
3.1	Introduction	34
3.2	Réseaux antagonistes génératifs (GANs)	34
3.2.1	Introduction aux GAN	34
3.2.2	Architecture et fonctionnement des GANs	35
3.2.2.1	Modèle de générateur	36
3.2.2.2	Modèle de discriminateur	37
3.2.3	Avantages et inconvénients	38
3.2.3.1	Avantages	38
3.2.3.2	Inconvénients	38
3.3	Réseaux antagonistes génératifs à convolution profonde (DCGANs)	38
3.3.1	Introduction aux DCGANs	38
3.3.2	Architecture et fonctionnement des DCGANs	39
3.4	Défis et perspectives futures	41
3.5	État de l'art de l'utilisation des GAN pour le traitement des images médicales	43
3.5.1	Segmentation du Foie avec des GAN Basés sur les Transformers	43
3.5.2	Apprentissage Non Supervisé avec les DC-GAN	45
3.5.3	Augmentation des Images Médicales avec des GAN	45

3.5.4	Détection des Anomalies dans les Images Médicales	45
3.5.5	Synthèse et Perspectives	46
3.6	Conclusion	46
4	Implémentation, résultats et évaluation	47
4.1	Introduction	47
4.2	Description de la dataset utilisée : LiTS	47
4.3	Outils et bibliothèques	48
4.3.1	Python	48
4.3.2	TensorFlow	49
4.3.3	Keras	49
4.3.4	Matplotlib	49
4.3.5	Pandas	49
4.3.6	NumPy	49
4.3.7	Scikit-image (skimage)	49
4.3.8	Google Colab	50
4.3.8.1	Avantages de Google Colab	50
4.3.8.2	Utilisation de Colab	50
4.3.9	Le modèle DC-GAN	53
4.3.9.1	Générateur	53
4.3.9.2	Discriminateur	55
4.3.9.3	Entraînement du modèle	57
4.3.9.4	Génération d'images	58
4.3.10	Résultats de la génération d'images	59
4.4	Résultats et discussion du modèle DCGAN	59
4.5	Modèle de détection	60
4.5.1	Résultats du modèle de détection	62
4.6	Conclusion	63
	Conclusion générale	64
	Bibliographie	65

Table des figures

1.1	Historique de l'imagerie médicale	3
1.2	Une des premières radiographies prise par Wilhelm Rontgen	4
1.3	Image médicale par scanner	4
1.4	Radiologie rayons-X	5
1.5	IRM Cérébrale	6
1.6	Image gris	8
1.7	Image couleur	8
1.8	Image indexé	9
1.9	Présentation de pixel	9
1.10	Histogramme d'une image	10
1.11	Représentation de cancer de foie	12
1.12	Cancer du foie métastatique	12
1.13	incidence et mortalité selon le type de cancer, par Vivien Delmon	15
2.1	Exploration des techniques d'apprentissage automatique	18
2.2	Quelques applications et types de l'apprentissage automatique	19
2.3	Les tâches d'apprentissage automatique les plus courantes	20
2.4	Parmi les manières d'illustrer un réseau de neurones	22
2.5	Image caricaturale d'un neurone biologique	23
2.6	Architecture du perceptron multicouche (MLP)	25
2.7	Courbes des fonctions d'activation mentionnées	27
2.8	Illustration de l'invariance de translation	29
2.9	Illustration du principe de localité	29
2.10	Convolution d'un patch d'image avec un filtre CNN	31
2.11	Convolution transposée utilisant un noyau de dimensions 2×2	31
2.12	Corrélation croisée bidimensionnelle avec padding	32
2.13	Corrélation croisée avec un <i>stride</i> de 3 et 2 pour la hauteur et la largeur	32
3.1	Réseaux Antagonistes Génératifs	35
3.2	Exemple d'architecture de modèle de réseau antagoniste génératif [1]	36
3.3	Exemple de modèle générateur	37
3.4	Exemple de modèle discriminateur	38
3.5	Architecture du discriminateur d'un DCGAN	40
3.6	Architecture de générateur de DCGAN	41
3.7	Architecture générale de DCGAN	41
3.8	Schéma bloc du Transformer GAN	44
3.9	Résultats de segmentation du foie basés sur le GAN Transformer	44
3.10	DCGAN generator used for LSUN scene modeling	45
4.1	Échantillon de la dataset	48
4.2	Google colab	50
4.3	choisir l'accélération matériel	51
4.4	Importation des bibliothèques requises	51
4.5	Chargement de la dataset	52
4.6	Préparation de données	52
4.7	Les composants des blocs transposés composant le générateur	53
4.8	Les paramètres des 6 blocs transposés composant notre générateur DCGAN	54

4.9	Schéma général de notre générateur DCGAN	54
4.10	Portion de code montrant une partie de notre générateur DCGAN	55
4.11	Les composants des blocs de convolution composant le discriminateur	56
4.12	Les paramètres des 4 blocs de convolution composant notre discriminateur	56
4.14	Portion de code montrant une partie de notre discriminateur DCGAN	57
4.13	Schéma général de notre discriminateur DCGAN	57
4.15	Entraînement du modèle DCGAN	58
4.16	Génération d'images	58
4.17	Exemples d'images générées : (a) après 200 époques, et (b) après 300 époques . . .	59
4.18	La courbe de perte (loss)	60
4.19	Architecture du modèle de détection	61
4.20	Architecture de discriminateur 2	61
4.21	Valeurs de perte et d'accuracy pour le modèle de détection	62

Acronymes

- **CT scan** : tomodensitométrie.
- **IRM** : imagerie par résonance magnétique.
- **RGB** : image couleur composée des composantes rouge, verte et bleue.
- **IA** : intelligence artificielle.
- **ML** : apprentissage automatique (machine learning).
- **GAN** : Réseaux Antagonistes Génératifs.
- **DCGAN** : Deep Convolutional GAN.
- **AFP** : alpha-fœtoprotéine, un marqueur tumoral.
- **Adam** : Adaptive moment estimation.
- **RMSProp** : Root Mean Square Propagation.

INTRODUCTION GÉNÉRALE

Introduction générale

L'imagerie médicale est un domaine de recherche extrêmement dynamique dans le traitement des images. Grâce aux avancées récentes réalisées dans ce secteur, le diagnostic des pathologies peut être effectué avec une plus grande précision. De plus, ces progrès ouvrent de nouvelles perspectives prometteuses en matière de traitement pour de nombreuses maladies. Le cancer du foie est un type de cancer courant qui se développe à partir des cellules hépatiques. Les principaux facteurs de risque sont les hépatites virales B et C, la consommation excessive d'alcool, certaines maladies héréditaires et l'exposition à des toxines. Les symptômes peuvent inclure la perte de poids, la fatigue, les nausées, les vomissements et les douleurs abdominales, mais il est souvent asymptomatique aux premiers stades. Le diagnostic repose sur l'imagerie médicale, ainsi que sur la biopsie. Le traitement dépend du stade et comprend la chirurgie, la chimiothérapie, la radiothérapie ou la greffe de foie. Les techniques d'imagerie médicale, telles que la tomodensitométrie (CT scan), jouent un rôle crucial dans la détection et le diagnostic précoces de cette maladie. Cependant, l'analyse manuelle de ces images est une tâche complexe et chronophage qui dépend largement de l'expertise des radiologues. La prévention passe par la vaccination contre l'hépatite B, l'évitement des facteurs de risque comme l'alcool, et le dépistage précoce pour augmenter les chances de guérison. Dans notre projet, nous proposons l'utilisation des réseaux GAN pour la détection du cancer du foie. Les GAN sont une classe de modèles d'apprentissage profond capables de générer des données synthétiques qui imitent fidèlement les données réelles. En particulier, nous nous concentrons sur les modèles DCGAN (Deep Convolutional GAN) pour générer des images de foie et les utiliser pour améliorer la détection du cancer à partir de CT scans. L'objectif principal de ce travail est de développer un modèle capable non seulement de générer des images de foie réalistes mais aussi de détecter efficacement les anomalies indicatives de la présence de tumeurs. Ce mémoire comprend quatre chapitres :

- Dans le premier chapitre nous parlerons de traitement d'image et l'imagerie médicale et le cancer du foie.
- Dans le deuxième chapitre nous présenterons l'apprentissage en profondeur et nous expliquerons l'architecture des réseaux neuronaux convolutifs (Convolutional Neural network ou CNNs). Dans ce chapitre, nous apprendrons plus sur l'architecture d'un réseau neuronal convolutif et sur toutes ses propriétés telles que les couches, les fonctions.
- Dans le troisième chapitre nous présenterons GAN particulièrement DCGAN et un état de l'art DE l'utilisation desGAN pour le traitement des images médicales.
- le dernier chapitre concerne notre application de la classification et l'augmentation des images médicales.

CHAPITRE 1

Généralités

1 Généralités

1.1 Introduction

L'imagerie médicale est un outil indispensable pour le diagnostic et le suivi des maladies, permettant une visualisation non invasive des structures et des processus physiologiques du corps humain. Ce chapitre examine les principes fondamentaux de l'imagerie médicale, les diverses technologies utilisées et leur importance dans le diagnostic et le traitement. Nous discuterons également des bases du traitement d'images, des types d'images disponibles et de leurs caractéristiques distinctives. Enfin, nous aborderons le cancer du foie en détaillant ses types, ses symptômes et les méthodes de traitement, tout en mettant en lumière le rôle crucial de l'imagerie médicale.

1.2 Imagerie médicale

L'imagerie médicale est une discipline essentielle en médecine qui permet la visualisation des structures internes du corps humain de manière non invasive. Utilisant diverses technologies telles que les rayons X, l'IRM et l'échographie, elle joue un rôle crucial dans le diagnostic, le suivi et le traitement des maladies. L'imagerie médicale permet aux médecins d'examiner l'intérieur du corps d'un patient sans avoir recours à la chirurgie [2], Elle englobe un ensemble de techniques qui permettent de visualiser des parties ou des organes du corps humain et d'en conserver des images. Ces techniques sont utilisées pour établir un diagnostic, guider des interventions thérapeutiques, ou évaluer les résultats des traitements [3].

1.2.1 Historique de l'imagerie médicale

L'imagerie médicale a une longue histoire de développement et voici les innovations les plus importantes qui ont marqué l'histoire de la radiologie et de l'imagerie médicale (voir aussi la figure 1.1) :

- **1895** : Wilhelm Conrad Roentgen découvre les rayons X en Allemagne, capturant la première image de la main de sa femme.
- **1896** : Les rayons X sont rapidement utilisés médicalement, tandis qu'Antoine-Henri Becquerel découvre la radioactivité.
- **1914-1918** : Les équipements radiologiques sont utilisés dans les hôpitaux de campagne pendant la Première Guerre mondiale.
- **1918** : George Eastman introduit le film, remplaçant les radiographies sur plaques de verre.
- **1946** : Edward Purcell et Felix Bloch découvrent indépendamment la résonance magnétique nucléaire (RMN).
- **1955** : Ian Donald et Tom Brown développent un échographe portable pour la gynécologie.

- **1961** : James Robertson conçoit le premier scanner de Tomographie à Emission de Positons (TEP) à plan unique.
- **1975** : Les échographies en temps réel utilisant les ultra-sons sont enregistrées pour la première fois, élargissant les applications de l'imagerie médicale.
- **1977** : Raymond Damadian réalise la première IRM, ouvrant de nouvelles applications en imagerie médicale.
- **Années 1990** : L'échographie devient courante pendant la grossesse pour surveiller le développement fœtal.
- **1992** : Seiji Ogawa, John Belliveau et Pierre Bandettini réalisent la première IRM fonctionnelle du cerveau.
- **2000** : Le scanner PET-CT de David Townsend et Ronald Nutt est nommé invention médicale de l'année par TIME Magazine [4].

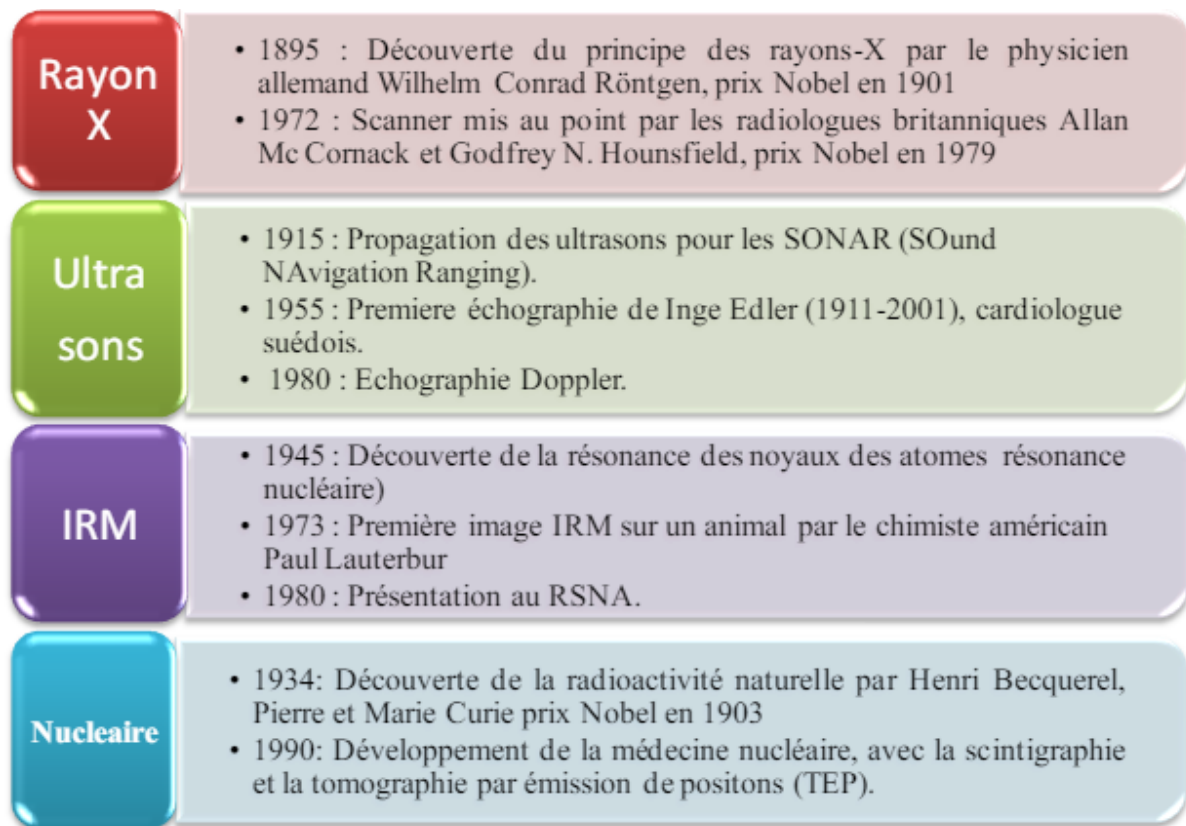


FIGURE 1.1 – Historique de l'imagerie médicale

1.2.2 Les différentes technologies d'imagerie médicale

Il existe une variété de technologies d'imagerie médicale, chacune utilisant des méthodes physiques distinctes pour observer les tissus du corps humain et fournir des informations spécifiques à leur sujet.

1.2.2.1 Les rayons X

Les rayons X sont une forme de rayonnement électromagnétique similaire à la lumière, mais invisible et hautement énergétique. Leur énergie leur permet de pénétrer profondément dans la matière vivante (voir la figure 1.2).

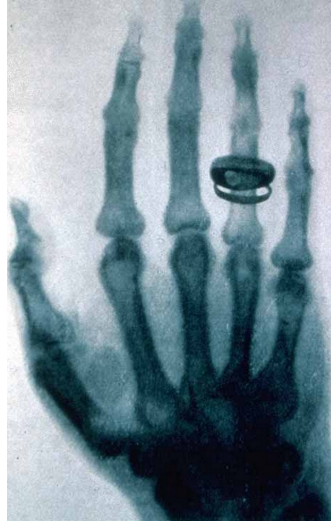


FIGURE 1.2 – Une des premières radiographies prise par Wilhelm Rontgen

1.2.2.2 Le scanner (tomodensitométrie)

Le scanner, également connu sous le nom de tomodensitométrie dans le langage scientifique, repose sur le même principe que la radiologie, utilisant une source de rayons X. Il permet d'étudier diverses parties du corps humain telles que le cerveau, la cage thoracique, l'abdomen ou les os, permettant ainsi de détecter des anomalies qui ne sont pas visibles sur des radiographies conventionnelles [5] (voir la figure 1.3).

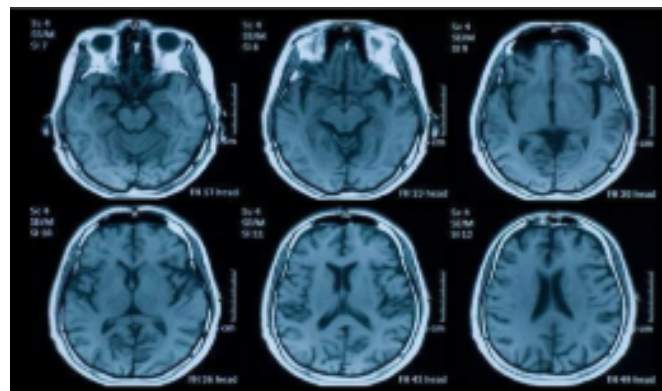


FIGURE 1.3 – Image médicale par scanner [5]

1.2.2.3 Radiographie

La radiographie, une technique d'imagerie médicale basée sur les rayons X découverts par Wilhelm Conrad Röntgen en 1895, fournit des images du corps humain en utilisant de faibles doses de rayons X. Elle enregistre sur un film photographique l'image projetée de transparence aux rayons X d'une région anatomique. Les radiographies simples ou standards permettent principalement d'obtenir des clichés en deux dimensions des structures osseuses et articulaires. Elles sont largement utilisées en orthopédie et en rhumatologie. La radiographie de contraste permet de visualiser certains organes ou parties creuses en les remplissant d'un produit opaque aux rayons X. Découverte il y a plus de 100 ans, la radiographie reste un outil diagnostique essentiel dans de nombreux domaines médicaux [6] (voir la figure 1.4).

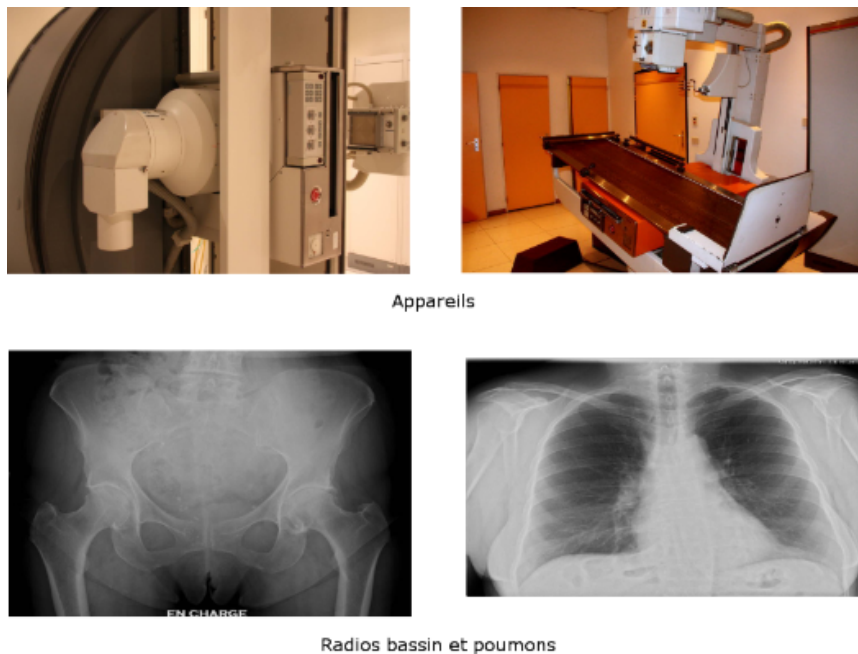


FIGURE 1.4 – Radiologie rayons-X [6]

1.2.2.4 L'imagerie par résonance magnétique (IRM)

est une technique développée depuis environ cinquante ans. elle permet de visualiser la structure anatomique de n'importe quelle partie du corps, en particulier les tissus mous tels que le cerveau, la moelle épinière, les viscères, les muscles et les tendons, grâce à la résonance magnétique nucléaire (RMN). L'IRM peut également suivre l'activité d'organes comme le cerveau en observant l'afflux de sang oxygéné dans certaines zones [7] (voir la figure 1.5).

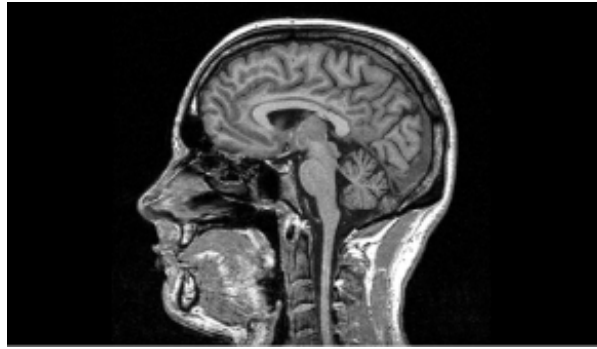


FIGURE 1.5 – IRM Cérébrale [7]

1.2.3 L'importance de l'imagerie médicale

L'imagerie médicale revêt une importance cruciale dans le domaine de la santé. Elle permet un diagnostic précis des pathologies et des lésions, facilitant ainsi une prise en charge médicale adéquate, tout en étant essentielle pour le suivi des traitements. En élargissant l'accès à l'imagerie médicale, il est possible de sauver des vies en diagnostiquant précocement des affections guérissables, réduisant ainsi le fardeau du cancer et d'autres maladies dans le monde. De plus, les avancées technologiques dans ce domaine ont révolutionné la médecine en permettant une visualisation détaillée de l'anatomie et des processus physiologiques du corps humain, ouvrant de nouvelles perspectives en recherche biomédicale et améliorant la compréhension des mécanismes pathologiques et des réponses aux traitements.

1.2.4 Le potentiel de l'IA pour l'imagerie médicale

L'intelligence artificielle (IA), notamment l'apprentissage automatique (ML), révolutionne le secteur de la santé, en particulier l'imagerie médicale, avec un impact significatif. Le marché mondial de l'IA devrait atteindre 914.8 milliards de dollars d'ici 2028, soulignant son potentiel croissant. Grâce à sa capacité à analyser les analyses médicales avec une précision et une rapidité supérieures, l'IA transforme l'imagerie médicale en améliorant les dépistages et en permettant une détection précoce des maladies. En radiologie, elle se révèle efficace pour identifier des affections telles que la tuberculose, la pneumonie et le cancer du poumon avec une précision surpassant souvent celle des radiologues expérimentés. Dans le domaine de la médecine de précision, l'IA joue un rôle significatif en décodant les caractéristiques des tumeurs, facilitant ainsi la personnalisation des traitements. De plus, elle est essentielle pour prédire et évaluer les risques, permettant une détection précoce des problèmes de santé avant l'apparition des symptômes [8].

1.3 Traitement d'images

Le traitement d'images désigne l'ensemble des techniques et des méthodes utilisées pour modifier ou améliorer une image numérique. Cela inclut un large éventail d'opérations, telles que l'ajustement de la luminosité et du contraste, la correction des couleurs, la suppression du bruit, la détection

d'objets, la segmentation, la classification, etc. L'objectif principal du traitement d'images est d'obtenir une image de meilleure qualité ou de rendre l'information visuelle plus compréhensible pour l'analyse ou la présentation. L'importance du traitement d'images réside dans sa capacité à améliorer la qualité visuelle des images, à extraire des informations pertinentes à partir de celles-ci et à automatiser des tâches complexes.

1.3.1 Définition

1.3.1.1 L'image

Une image est une représentation visuelle générée par différents moyens comme la peinture ou la photographie. Pour l'humain, c'est un ensemble d'informations structurées significatives à l'affichage. Pour une machine, une image analogique est décrite comme une fonction continue $f(x, y)$ d'intensité lumineuse et couleur en chaque point (x, y) . Sous cette forme, l'image est inexploitable et nécessite une numérisation préalable.

1.3.1.2 L'image numérique

Une image numérique est une représentation discrète où la surface est divisée en une grille de pixels. Chaque pixel encode un niveau de gris ou une valeur de couleur. La numérisation convertit une image analogique continue en une matrice bidimensionnelle de valeurs numériques $f(x, y)$. Les coordonnées (x, y) localisent chaque point, tandis que $f(x, y)$ représente l'intensité lumineuse mesurée au point par le capteur. Ainsi, chaque pixel stocke une valeur quantifiant la lumière reçue localement [9].

1.3.2 Les types d'image

1.3.2.1 Image binaire

Une image binaire est une image $M \times N$ où chaque pixel ne peut avoir que deux valeurs : 0 pour le noir ou 1 pour le blanc. Le niveau de gris est codé sur 1 bit, avec 0 représentant le noir et 1 le blanc. Ainsi, pour un pixel $p(i, j)$, seules les valeurs $p(i, j) = 0$ (noir) ou $p(i, j) = 1$ (blanc) sont possibles.

1.3.2.2 Image en niveau de gris

Le niveau de gris représente l'intensité lumineuse en un point de l'image. Un pixel peut prendre des valeurs allant du noir (0) au blanc (255), en passant par des niveaux intermédiaires. Pour coder les niveaux de gris, on attribue à chaque pixel une valeur numérique entre 0 et 255 correspondant à la quantité de lumière reçue. Ainsi, un pixel n'est plus codé sur 1 bit mais sur 1 octet (8 bits) permettant 256 niveaux [2] (voir la figure 1.6).



FIGURE 1.6 – Image gris [2]

1.3.2.3 Image couleur (RGB)

Une image RGB est une image numérique dans laquelle chaque pixel est représenté par trois valeurs codées sur 8 bits, correspondant aux intensités des composantes rouge, verte et bleue. La combinaison de ces trois couleurs primaires permet de reproduire une large palette de couleurs [10] (voir la figure 1.7).



FIGURE 1.7 – Image couleur [10]

1.3.2.4 Image indexée

La plupart des images n'exploitent pas les 16 millions de couleurs disponibles en raison des contraintes de stockage et de gestion de fichiers. Elles utilisent plutôt une palette de couleurs qui représente la liste des couleurs qu'elles contiennent. Ainsi, la valeur d'un pixel correspond à un

indice dans cette palette. Par exemple, un indice de 6 indique la référence dans la palette de couleur fournissant les proportions de chaque couleur de base (Rouge, Vert, Bleu) (voir la figure 1.8).

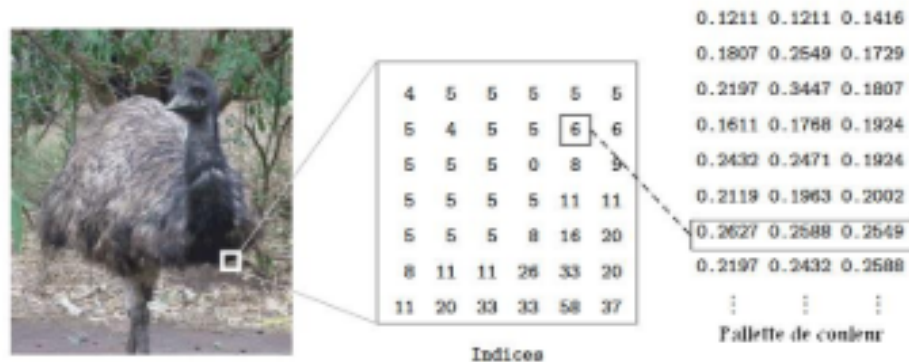


FIGURE 1.8 – Image indexé

1.3.3 Les caractéristiques d'image

L'image est caractérisé par les paramètres suivants :

- **Pixel** : Une image numérique est composée d'un ensemble de petits points appelés pixels (abréviation de "picture element"). Le pixel représente ainsi la plus petite unité constitutive d'une image numérique. C'est une valeur numérique représentative des intensités lumineuses. Tout comme le bit est la plus petite unité d'information qu'un ordinateur peut traiter, le pixel est le plus petit élément que les matériels et logiciels peuvent manipuler sur une image [11] (voir la figure 1.9).

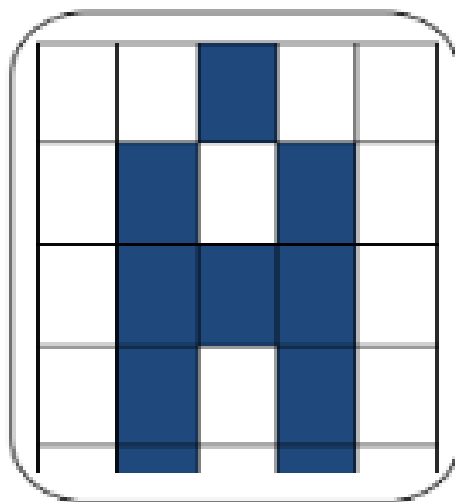


FIGURE 1.9 – Présentation de pixel [11]

- **Histogramme** : Il s'agit de la représentation visuelle d'une variable statistique, réalisée à

travers un histogramme ou un diagramme en bâtons, où chaque classe de la variable est associée à un rectangle dont la longueur est proportionnelle à l'amplitude de la classe et la hauteur à l'effectif de cette classe.

Dans le domaine de la photographie numérique, l'histogramme d'une image présente sur l'axe horizontal (abscisses) le nombre de valeurs de luminosité différentes qu'un pixel peut prendre, allant de 0 à la valeur maximale (voir la figure 1.10).

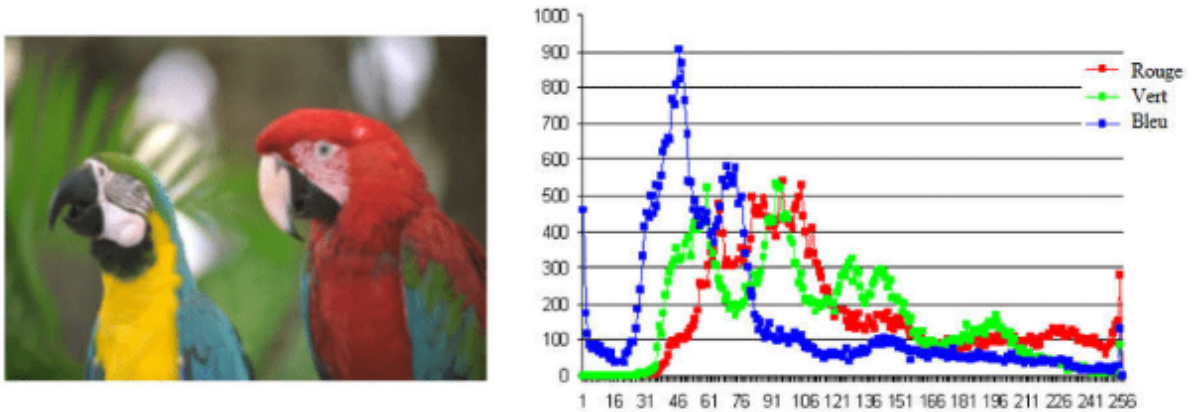


FIGURE 1.10 – Histogramme d'une image

- **Dimension** : Il s'agit de la dimension de l'image, qui est représentée sous forme d'une matrice où chaque élément représente une valeur numérique correspondant à l'intensité lumineuse d'un pixel. Le produit du nombre de lignes de cette matrice par le nombre de colonnes nous donne le nombre total de pixels dans l'image [12].
- **Contraste** : Il s'agit de la nette différence entre deux zones distinctes d'une image, spécifiquement entre les zones sombres et claires. Le contraste est établi en comparant les luminances de ces deux zones de l'image [13].
- **Luminance** : La luminance représente le niveau de luminosité des points présents dans une image. Elle est également définie comme le rapport entre l'intensité lumineuse d'une surface et sa surface apparente, pour un observateur distant. Le terme "luminance" est souvent utilisé à la place de "brillance", qui fait référence à l'éclat d'un objet. Une bonne luminance est caractérisée par :
 - Des images lumineuses et éclatantes.
 - Un bon contraste : il est important d'éviter les images où la gamme de contraste tend vers le blanc ou le noir, car cela entraîne une perte de détails dans les zones sombres ou lumineuses.
 - L'absence de parasites [14].
- **Bruit** : Le bruit, également appelé parasite, dans une image résulte de variations soudaines dans l'intensité d'un pixel par rapport à ses pixels voisins. Il est généré par l'éclairage des dispositifs optiques et électroniques du capteur. Les sources de bruit incluent :
 - La qualité de l'appareil de capture.

- Les conditions de prise de vue telles que la luminosité, les mouvements de la scène [15], etc.

1.3.4 Segmentation des images

La segmentation constitue l'une des étapes critiques de l'analyse d'images, influençant directement la qualité des mesures effectuées par la suite. Elle représente généralement la première phase d'un traitement plus complexe, tel que la reconnaissance de formes, et vise à délimiter les contours des objets soumis à l'analyse. Son objectif est de définir des régions d'intérêt afin de les extraire du fond. Une méthode de segmentation efficace sera celle permettant une interprétation précise de l'image, simplifiant son contenu sans le réduire excessivement. Actuellement, il existe de nombreuses techniques de segmentation, toutes sensibles au bruit. Il est donc crucial de commencer par nettoyer l'image en appliquant les filtres habituels d'atténuation du bruit. Parmi les approches de segmentation les plus connues, on peut citer : le seuillage, la segmentation basée sur les régions et la segmentation basée sur les contours [16].

Les objectifs de segmentation sont :

- Fournir des régions homogènes (selon un critère donné).
- Localiser de manière précise les contours des régions.
- L'étude et l'interprétation des structures anatomiques.
- Réduction de bruit [17].

1.3.5 Classification des images

La classification consiste à attribuer une forme à une ou plusieurs classes, qu'elles soient prédéfinies ou non, en se basant sur des caractéristiques similaires. Cette attribution est déterminée par une règle de décision [18].

Dans le domaine de la détection du cancer, la classification est le processus visant à identifier des zones potentiellement cancéreuses en se fondant sur certaines caractéristiques spécifiques des tissus ou des cellules. Par exemple, lorsqu'il est nécessaire d'analyser des images médicales pour détecter des signes précoces de cancer, la classification permet de traduire les informations visuelles en informations diagnostiques pertinentes. Habituellement, deux approches sont envisageables : la classification supervisée, où le modèle est entraîné sur des données annotées de manière à distinguer les caractéristiques associées au cancer, et la classification non supervisée, qui cherche à identifier des modèles ou des structures dans les données sans supervision préalable [19].

1.4 Détail sur le cancer du foie

Le cancer du foie est un type de cancer qui commence dans les cellules du foie. Les deux types principaux sont le carcinome hépatocellulaire, qui commence dans les principales cellules du foie appelées les hépatocytes, et le cholangiocarcinome, qui commence dans les cellules tapissant les conduits biliaires (voir la figure 1.11).



FIGURE 1.11 – Représentation de cancer de foie

Il existe deux grands types de cancers du foie : En fonction de leur lieu d'origine, les cancers se divisent en deux catégories : les cancers primitifs et les cancers métastatiques. Les cancers primaires du foie se développent à partir de cellules cancéreuses présentes dans le foie lui-même, tandis que les cancers métastatiques sont des tumeurs qui se forment dans d'autres parties du corps avant de migrer vers le foie.

- **Cancers primitifs du foie** : Les cancers primaires du foie, principalement des carcinomes hépatocellulaires, se développent souvent dans un foie déjà affecté par des maladies comme la cirrhose. Ils sont moins fréquents chez les personnes ayant un foie sain. D'autres types de tumeurs hépatiques, tels que le cholangiocarcinome et l'angiosarcome, sont moins courants.
- **Cancers métastatiques du foie** : Les cancers métastatiques sont bien plus fréquents que les cancers primaires (20 à 50 fois). Cela s'explique par la fonction filtrante du foie qui draine tout le sang de l'organisme. Ainsi, les cellules cancéreuses libérées dans le sang, par exemple par une tumeur pulmonaire, ont tendance à coloniser et à se développer dans le foie [20].

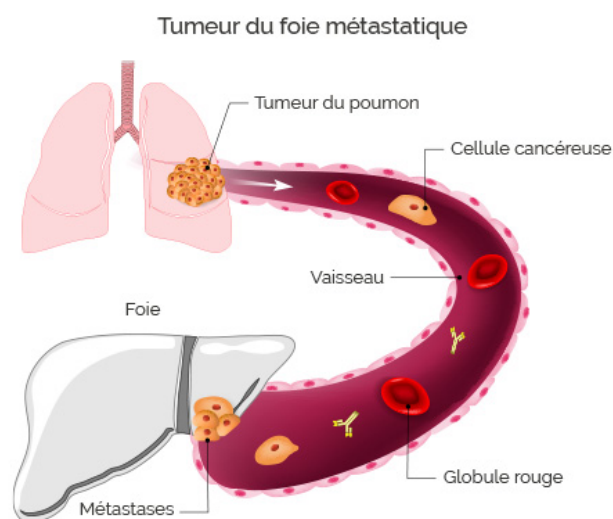


FIGURE 1.12 – Cancer du foie métastatique

1.4.1 Les facteurs de risque

- **Cirrhose et cancer du foie** :La majorité des cancers du foie (environ 80 %) sont liés à la cirrhose, une maladie hépatique chronique qui altère le fonctionnement du foie et favorise le développement de tumeurs malignes.
- **Causes de la cirrhose** : La cirrhose résulte d'une inflammation chronique du foie due principalement à une consommation excessive d'alcool, aux infections par les virus de l'hépatite B et C, ou plus rarement à certaines maladies métaboliques.
- **Processus de fibrose** : L'inflammation chronique entraîne une fibrose, la formation de tissu cicatriciel fibreux. Si cette fibrose persiste, elle peut évoluer vers une cirrhose où le foie est gravement endommagé et dysfonctionnel.
- **Risque de cancer et facteurs individuels** :Le cancer peut se développer dès les premiers stades de la fibrose, mais tous les patients cirrhotiques ne développent pas un cancer du foie. Le risque dépend de la cause initiale de la cirrhose et de facteurs individuels tels que l'âge, le sexe, et le mode de vie.
- **Virus de l'hépatite B et C** :Les hépatites B et C sont la principale cause mondiale de carcinome hépatocellulaire, causant une infection chronique du foie qui évolue vers la fibrose, la cirrhose, et le cancer.
- **Consommation d'alcool** :La consommation excessive d'alcool peut entraîner une cirrhose hépatique et augmenter le risque de cancer du foie, ce risque augmentant avec la quantité et la durée de consommation.
- **Stéatose hépatique non alcoolique** :Courante chez les personnes obèses, diabétiques ou avec un taux élevé de triglycérides, cette condition implique une accumulation de graisse dans le foie et une inflammation chronique. Traiter les pathologies sous-jacentes réduit les risques de cirrhose.
- **Hémochromatose héréditaire** :Cette condition entraîne un stockage excessif de fer dans le foie, conduisant à des lésions graves comme la cirrhose et augmentant le risque de cancer. Le dépistage et un traitement précoce réduisent les risques de complications hépatiques.
- **Exposition à des substances toxiques** :L'exposition répétée à certaines substances comme l'aflatoxine B1 (produite par des moisissures), le tabagisme, ou la consommation régulière de stéroïdes anabolisants, augmente le risque de cancer du foie [20].

1.4.2 Symptômes et diagnostic

1.4.2.1 Symptômes

Le cancer du foie peut se manifester par des symptômes peu spécifiques comme [21] :

- Perte de poids inexpliqué.
- Fatigue constante.
- Perte d'appétit ou sensation de satiété rapide.
- Nausées, vomissements.
- Fièvre.

- Masse palpable sous les côtes droites (foie augmenté).
- Masse palpable sous les côtes gauches (rate augmentée).
- Douleurs abdominales ou au niveau de l'omoplate droite.
- Démangeaisons.

1.4.2.2 Le diagnostic

Le diagnostic du cancer du foie est basé sur les examens suivants :

- **Un examen clinique** : Un examen clinique peut révéler une augmentation du volume du foie ou de la rate, une accumulation de liquide dans l'abdomen, un jaunissement cutané et oculaire (jaunisse), ou d'autres signes de dysfonctionnement hépatique.
- **Une prise de sang** : L'analyse sanguine peut indiquer des taux élevés d'alpha-fœtoprotéine (AFP), un marqueur tumoral, chez 50 à 75 % des patients atteints d'un cancer du foie. Cependant, un taux normal d'AFP n'exclut pas la présence d'une tumeur, et un taux élevé ne confirme pas nécessairement un cancer. Le dosage de l'AFP n'est donc pas fiable à lui seul pour détecter un cancer du foie.
- **L'imagerie** : L'échographie est le premier examen d'imagerie effectué. En cas d'anomalie, un scanner et/ou une IRM avec produit de contraste sont nécessaires pour mieux caractériser la lésion et différencier un cancer primaire du foie d'autres pathologies. Pour les petits nodules (<1 cm), un contrôle après 3 mois est recommandé. Pour les gros nodules (>3 cm), le diagnostic de cancer est généralement clair. Pour les nodules intermédiaires, surtout si le taux d'AFP est normal, une biopsie hépatique est nécessaire pour confirmer le diagnostic. L'imagerie permet donc de détecter et caractériser les anomalies hépatiques, complétée par la biopsie en cas de doute.
- **La biopsie** : Contrairement à d'autres cancers, la biopsie n'est pas systématiquement réalisée en premier pour le cancer du foie. Les examens d'imagerie modernes (échographie, scanner, IRM) combinés au dosage de l'AFP permettent souvent de diagnostiquer un cancer du foie sans biopsie. La biopsie est requise seulement si un doute persiste après les examens non invasifs.
- **Les examens du bilan d'extension** : Après confirmation du diagnostic de cancer du foie, des examens complémentaires sont nécessaires pour évaluer l'étendue de la maladie et vérifier la présence de métastases. Ils déterminent le stade d'évolution de la tumeur [21].

1.4.3 Epidémiologie du cancer du foie

L'épidémiologie du cancer du foie est marquée par une incidence croissante de la maladie ces dernières décennies, avec des facteurs de risque variés incluant des expositions environnementales et professionnelles, les hépatites virales, l'alcoolisme, l'obésité, le diabète de type 2, l'hépatite B/D ou C, et le tabagisme (voir la figure 1.13).

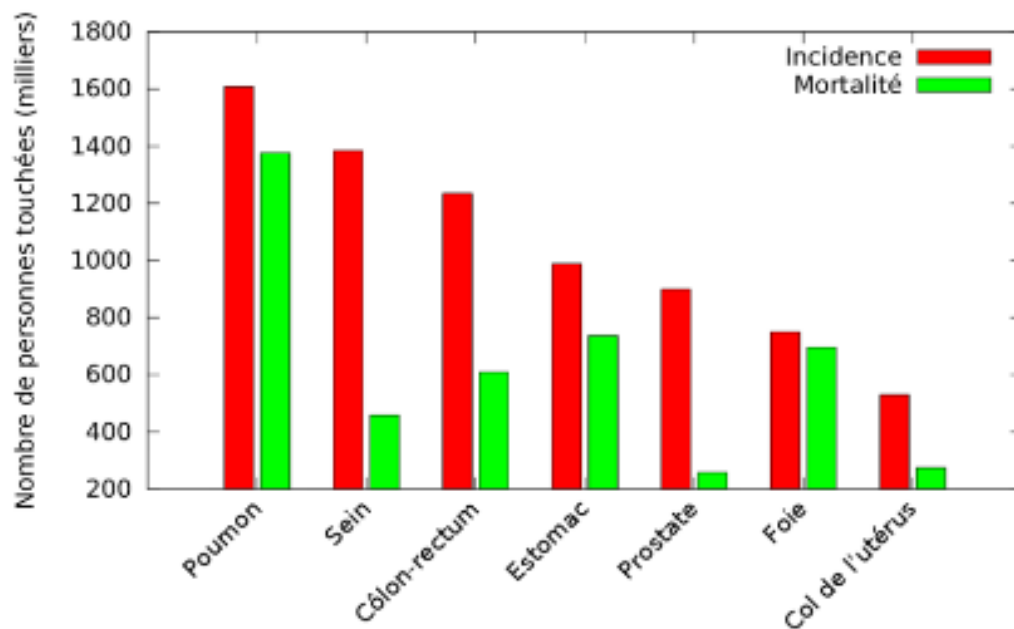


FIGURE 1.13 – incidence et mortalité selon le type de cancer, par Vivien Delmon

Le cancer du foie est le 4e cancer le plus mortel chez les hommes et le 7e chez les femmes, avec une survie à 5 ans des patients atteints de seulement 18%.

1.4.4 Traitements du cancer du foie

Le traitement du cancer du foie dépend de :

- La fonction hépatique résiduelle et l'état général du foie.
- Les caractéristiques de la ou des tumeurs, à savoir leur taille, leur localisation précise au sein du foie, ainsi que leur nombre.
- L'extension de la maladie cancéreuse, c'est-à-dire si elle s'est propagée ou non au-delà du foie vers d'autres organes.
- L'âge du patient et son état de santé global, prenant en compte les éventuelles.

Les cancers du foie qui peuvent être réséqués ou transplantés (stade précoce et certaines tumeurs au stade intermédiaire) :

- La résection chirurgicale partielle du foie est envisagée pour les cancers détectés précocement, avec un foie sain. Le succès dépend de la taille et de la proximité des tumeurs avec les vaisseaux sanguins. Une évaluation minutieuse est nécessaire pour les tumeurs volumineuses ou proches des vaisseaux, car les risques de complications et de récurrence sont accrus.
- Pour les tumeurs de grande taille ou ayant atteint les vaisseaux, le risque de récurrence locale est élevé. La transplantation hépatique peut être envisagée pour les patients avec un stade très précoce du cancer, si leur état le permet. L'évaluation de l'extension tumorale, de la fonction hépatique, et de l'état général est cruciale pour choisir la meilleure option de traitement.

- La chimiothérapie associée à la chirurgie pourrait prolonger la survie de certains patients, mais ces résultats ne sont pas universels. Des études supplémentaires sont nécessaires pour évaluer l'efficacité d'autres traitements combinés à la chirurgie.
- La greffe de foie est une option si le cancer est précoce mais que le foie est endommagé, ou si la tumeur est difficilement accessible chirurgicalement. Cependant, l'attente peut être longue. En attendant, d'autres options comme l'embolisation ou l'ablation peuvent être envisagées pour contrôler la progression tumorale [22].

1.4.5 Conclusion

L'imagerie médicale joue un rôle essentiel dans le diagnostic et le suivi des maladies. Dans ce chapitre, nous avons exploré les différentes technologies d'imagerie médicale ainsi que leur importance respective. Nous avons également abordé les principes de base du traitement d'images et leur application en médecine. Plus spécifiquement, nous nous sommes penchés sur le cancer du foie, en examinant les différents types de cette maladie, leurs symptômes, et les options thérapeutiques disponibles.

CHAPITRE 2

Apprentissage profond

2 Apprentissage profond

Ce chapitre est consacré à l'étude du domaine de l'apprentissage profond (deep learning), en mettant particulièrement l'accent sur les réseaux neuronaux convolutifs (CNN). Les CNN sont une technologie de pointe dans le domaine de l'apprentissage profond, révolutionnant la manière dont les machines peuvent analyser et interpréter des données visuelles, comme les images et les vidéos. Ils sont particulièrement efficaces pour la détection de motifs complexes dans les données visuelles et jouent un rôle crucial dans des applications telles que la détection du cancer du foie.

2.1 Introduction

Ce chapitre a pour objectif de fournir une compréhension approfondie des concepts fondamentaux de l'apprentissage profond et des réseaux neuronaux convolutifs (CNN), en préparation à l'étude des Réseaux antagonistes génératifs à convolution profonde (DCGAN, de l'anglais "Deep Convolutional Generative Adversarial Networks") qui seront utilisés dans notre application pour la détection du cancer du foie :

- Nous commencerons par une exploration approfondie des bases de l'apprentissage profond, incluant les principes de l'apprentissage automatique, de l'intelligence artificielle (IA), de l'apprentissage machine (ML), et des concepts clés de l'apprentissage profond.
- Par la suite, nous examinerons les réseaux neuronaux convolutifs (CNN), en abordant leur motivation, leur architecture et leur fonctionnement, ainsi que leurs applications.

2.2 Intelligence artificielle (IA)

L'intelligence artificielle (IA) est la simulation de l'intelligence humaine dans les machines, leur permettant d'accomplir des tâches qui nécessitaient auparavant l'intervention humaine. Elle permet aux machines d'apprendre de l'expérience, reconnaître des schémas et s'adapter à de nouvelles situations [23, 24], ce qui les rend précieuses pour diverses applications, de la recommandation [25], au génie civil [26], à la santé [27] en passant par la finance [28].

Les technologies de l'IA englobent plusieurs domaines, notamment l'apprentissage automatique, le deep learning (voir la figure 2.1), le traitement du langage naturel, la robotique et les systèmes experts. En ce qu'il s'agit de la santé, elle a notamment un impact significatif sur la gestion des patients atteints de cancer et la médecine en général :

- **Diagnostic et détection du cancer** : en aidant à la détection des signes précoces du cancer et en analysant des images médicales telles que les radiographies et les scanners. L'IA peut également contribuer à la classification des tumeurs malignes [29, 30].
- **Prédiction de la réponse au traitement** : en utilisant des algorithmes prédictifs, on arrive à estimer la probabilité de réponse d'un patient à un traitement spécifique [31]. Cela permet une personnalisation des thérapies.

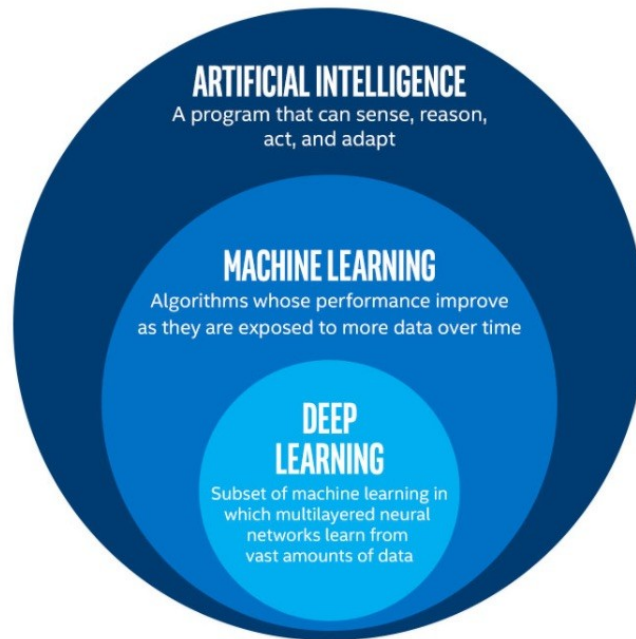


FIGURE 2.1 – Exploration des techniques d'apprentissage automatique

- **Génomique du cancer** : l'IA peut analyser les données génomiques pour identifier des mutations spécifiques associées au cancer pour aider à orienter les traitements ciblés [32].

2.3 Apprentissage automatique (ML)

L'Apprentissage Automatique (ML) est une sous-discipline de l'IA (voir la figure 2.1) qui permet aux machines d'apprendre à partir de données sans être explicitement programmées. En d'autres termes, au lieu d'écrire un programme spécifique pour effectuer une tâche donnée, un algorithme d'apprentissage automatique peut être entraîné sur un ensemble de données pour identifier des motifs et des relations dans ces données, puis utiliser ces connaissances pour faire des prédictions ou prendre des décisions sur de nouvelles données non explorées auparavant par l'algorithme [24].

L'expérience dont un algorithme d'apprentissage automatique apprend peut prendre différentes formes : des données étiquetées ou non étiquetées, des retours d'utilisateurs, des archives de journalisation ou de recensement quelconques, ou des interactions avec l'environnement.

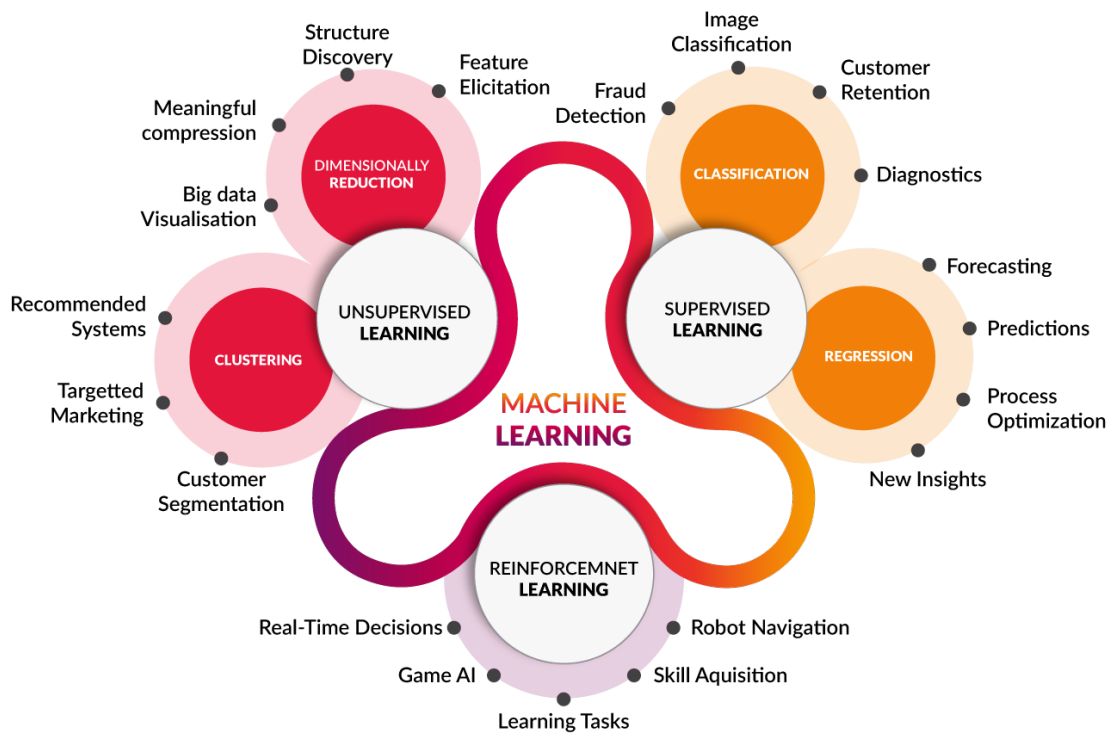


FIGURE 2.2 – Quelques applications et types de l’apprentissage automatique / Christelle Julias, smartpredict.ai

2.3.1 Différents types de l’apprentissage automatique

Les méthodes d’apprentissage automatique sont généralement divisées en quatre catégories, en fonction de la tâche et du type de rétroaction disponible pour le système d’apprentissage : [24] :

- **Apprentissage supervisé** : l’algorithme est présenté avec un ensemble de données d’entraînement composé de paires entrée/sortie, où les caractéristiques d’entrée sont généralement représentées sous forme de vecteur, et les étiquettes de sortie sont des catégories ou des valeurs prédéfinies [24, 33]. L’objectif de l’algorithme est d’apprendre une fonction qui peut prédire avec précision l’étiquette de sortie pour une nouvelle entrée¹.
- **Apprentissage non supervisé** : contrairement à l’apprentissage supervisé, l’apprentissage non supervisé apprend des motifs dans les données sans aucune supervision explicite ou orientation à partir d’exemples étiquetés. L’algorithme est présenté avec un ensemble de données d’entrée mais sans étiquettes de sortie correspondantes. L’objectif de l’algorithme est de découvrir des motifs ou des structures significatifs dans les données, comme des clusters, des composantes principales ou des motifs fréquents² [33].
- **Apprentissage semi-supervisé** : l’algorithme reçoit à la fois des exemples étiquetés et non étiquetés, en supposant que les exemples étiquetés sont plus coûteux ou difficiles à obtenir

1. Par exemple, la classification d’images, la reconnaissance vocale, le traitement du langage naturel et les systèmes de recommandation, etc.

2. La détection d’anomalies, la compression de données et la visualisation de données, etc.

que les non étiquetés. L'objectif est d'utiliser les exemples étiquetés pour guider le processus d'apprentissage et améliorer la précision du modèle sur les données non étiquetées [33].

- **Apprentissage par renforcement** : l'agent³ apprend à prendre des actions dans un environnement afin de maximiser un signal de récompense cumulatif. L'agent reçoit des retours sous forme de récompense ou de pénalité en fonction de ses actions, et l'objectif est d'apprendre une politique qui maximise la récompense cumulée attendue au fil du temps [24, 33].

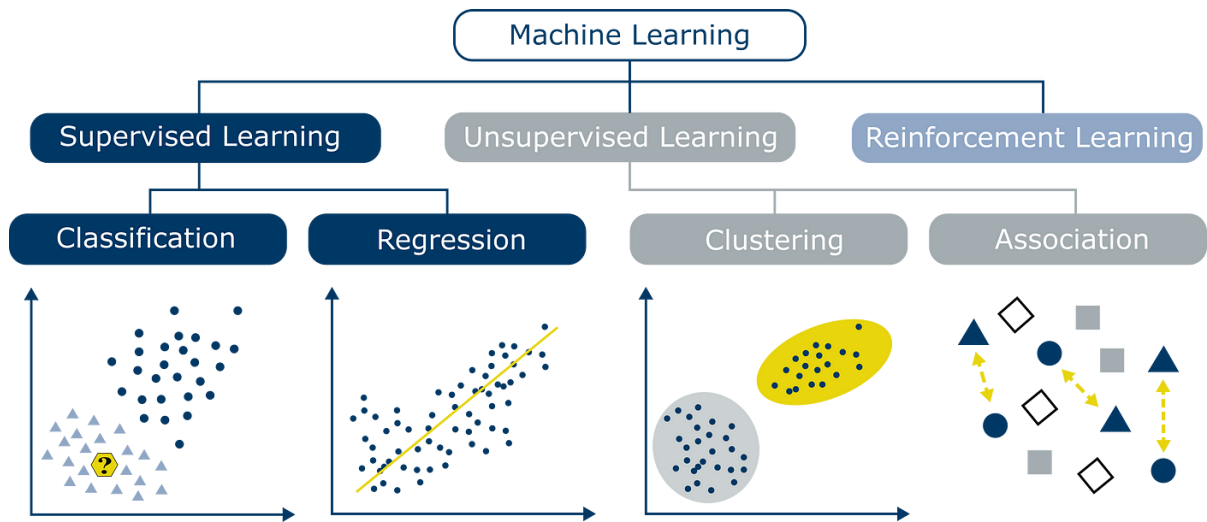


FIGURE 2.3 – Les tâches d'apprentissage automatique les plus courantes / Dominik Polzer, TowardsDataScience

2.3.2 Tâches de l'apprentissage automatique

Les types de tâches auxquelles les algorithmes d'apprentissage automatique peuvent être appliqués est assez large, les plus courantes incluent des tâches telles que la **régression**, la **classification** et le **groupement** (voir la figure 2.3) [24, 33].

- **Régression** : l'ordinateur reçoit un ensemble de données d'entrée et a pour tâche de prédire une valeur numérique, c'est-à-dire une fonction f qui associe les données d'entrée à une sortie numérique⁴ $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Une tâche de régression pourrait consister à prédire le prix d'une maison en fonction de ses caractéristiques telles que la taille, le nombre de chambres ou la position géographique par exemple [24].
- **Classification** : l'objectif est de prédire la classe ou la catégorie à laquelle appartient une nouvelle donnée, en fonction de ses caractéristiques ou attributs. L'algorithme d'apprentissage est entraîné sur un ensemble d'exemples étiquetés, et la sortie est une fonction qui mappe les données d'entrée vers une sortie discrète⁵ [24].

3. Un agent est une entité qui perçoit son environnement par le biais de capteurs et agit sur l'environnement par le biais d'actionneurs pour atteindre des objectifs spécifiques.

4. Ici, n représente la dimensionnalité du vecteur de caractéristiques (*features*).

5. Une valeur discrète, ou en d'autres termes *catégorique*, est une étiquette qui représente une classe.

Par exemple, identifier si un e-mail est un spam ou non en se basant sur son contenu et ses métadonnées. Il est utile de mentionner le fait que cet exemple possède deux classes *spam* et *non spam*, cela signifie qu'il s'agit d'une classification binaire.

- **Groupement (*clustering*)** : il consiste à regrouper des points de données similaires en fonction de leurs caractéristiques ou attributs. L'objectif du clustering est de partitionner un ensemble de points de données en clusters distincts ou groupes, où les points au sein de chaque groupe sont plus similaires les uns aux autres qu'à ceux des autres groupes. L'algorithme d'apprentissage ne reçoit pas d'exemples étiquetés, mais doit plutôt découvrir la structure sous-jacente des données par lui-même [33].

2.3.3 Limitations de l'apprentissage automatique

Les limitations les plus courantes de l'apprentissage automatique sont les suivantes :

- **Qualité et disponibilité des données** : dans de nombreux cas, les données peuvent être manquantes ou incomplètes, ou contenir des valeurs aberrantes qui peuvent affecter la qualité des prédictions. De plus, elles peuvent également être biaisées ou non représentatives, ce qui entraîne des modèles peu performants sur de nouvelles données ou des données non vues précédemment [34]. Pour relever ces défis, il est nécessaire de nettoyer et de prétraiter soigneusement les données, ainsi que de mettre en place des stratégies pour gérer les données manquantes ou incomplètes.
- **Interprétabilité et explicabilité** : l'interprétabilité fait référence à la capacité de comprendre et d'expliquer comment un modèle parvient à ses prédictions ou décisions, tandis que l'explicabilité concerne la capacité à fournir une explication claire et intuitive pour ces prédictions ou décisions [35]. Certains modèles d'apprentissage automatique sont très complexes et opaques, rendant difficile pour les humains de comprendre le raisonnement derrière leurs résultats. Cela peut soulever des préoccupations sérieuses en matière de sécurité dans des domaines où l'apprentissage automatique est sollicité pour assister des opérations critiques telles que la santé.
- **Biais et équité** : est une limitation courante en apprentissage automatique, découlant du fait que les algorithmes sont aussi objectifs que les données sur lesquelles ils sont entraînés. Il peut survenir lorsque les données d'entraînement contiennent des erreurs systématiques ou ne sont pas représentatives de la population réelle, conduisant à des modèles qui font des prédictions incorrectes ou injustes pour certains groupes [36].

2.4 Réseaux de neurones artificiels

Les réseaux de neurones artificiels sont des modèles computationnels d'apprentissage automatique inspirés du fonctionnement du cerveau humain. Ils sont utilisés pour résoudre des problèmes complexes où il n'est pas évident d'expliquer le rapport entre les caractéristiques et leur effet.

2.5 Fonctionnement d'un réseau de neurones artificiel

Un réseau de neurones est composé de plusieurs couches de neurones interconnectés. Chaque neurone reçoit des entrées pondérées, les combine à l'aide d'une fonction d'activation et produit une sortie [24, 37].

Il est courant d'illustrer un réseau neuronal de deux manières : la méthode de la somme des produits ou la méthode de la transformation linéaire (voir la figure 2.4). Dans la première approche, on décompose les opérations linéaires en une série de produits et de sommes plus simples. Dans la seconde approche, on considère les transformations linéaires directement. Ces deux méthodes permettent de représenter les opérations effectuées par un réseau neuronal mais elles sont équivalentes en termes de fonctionnalité.

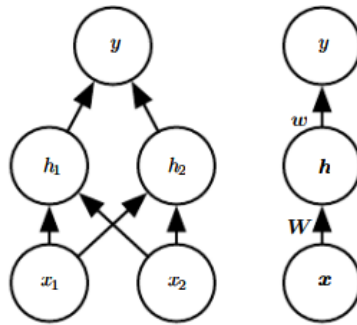


FIGURE 2.4 – Parmi les manières d'illustrer un réseau de neurones [38]

Formellement, soit X l'ensemble des entrées (vecteur d'entrée), W la matrice des poids qui combine les entrées pondérées, b le vecteur de biais, et σ la fonction d'activation. La sortie Y d'une couche de neurone est donnée par

$$Y = \sigma(WX + b), \quad (2.1)$$

$WX + b$ représente la transformation linéaire des entrées, suivie de l'application de la fonction d'activation σ . Les paramètres W et b sont optimisés grâce à des algorithmes d'optimisation telle que la descente du gradient lors de l'apprentissage. Par exemple, pour les matrices illustrées dans l'équation (2.2), la sortie Y sera de dimension \mathbb{R}^2 ,

$$X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \quad W = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}. \quad (2.2)$$

la puissance des réseaux de neurones artificiels réside dans leur capacité à apprendre les relations non linéaires et à étendre la dimensionalité afin d'accomoder plus d'informations et proposer une certaine flexibilité [38].

2.5.1 Contexte et historique des réseaux de neurones artificiels

Les scientifiques Warren McCulloch et Walter Pitts sont les premiers à avoir développé des modèles de neurones artificiels [39].

Considérez l'analogie avec un neurone biologique, présentée dans la figure 2.5, qui se compose de dendrites (terminaux d'entrée), du noyau (CPU), de l'axone (fil de sortie) et des terminaux d'axone (terminaux de sortie), permettant des connexions avec d'autres neurones via des synapses [40].

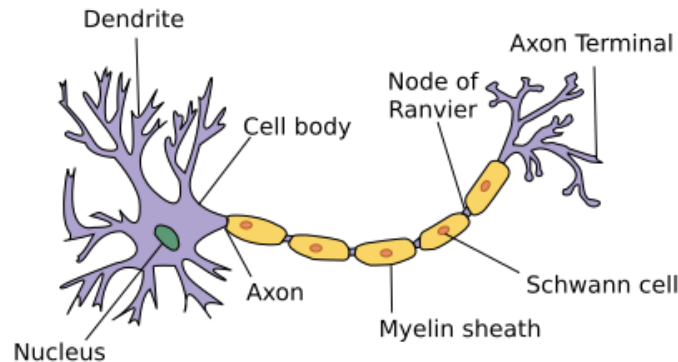


FIGURE 2.5 – Image caricaturale d'un neurone biologique [40]

L'information x_i provenant d'autres neurones (ou de capteurs environnementaux) est reçue dans les dendrites. En particulier, cette information est pondérée par des poids synaptiques w_i , déterminant l'effet des entrées, par exemple, l'activation ou l'inhibition via le produit $x_i w_i$. Les entrées pondérées provenant de sources multiples sont agrégées dans le noyau sous forme de somme pondérée $y = \sum_i x_i w_i + b$, éventuellement soumise à un post-traitement non linéaire via une fonction $\sigma(y)$. Cette information est ensuite envoyée via l'axone vers les terminaux d'axone, où elle atteint sa destination⁶ ou est transmise à un autre neurone via ses dendrites [39].

2.5.2 Apprentissage et optimisation d'un réseau de neurones

Pour ajuster un réseau de neurones aux données, il faut calculer une mesure de l'adéquation ou d'erreur (aussi appelé fonction de perte). Les fonctions de perte quantifient la distance entre les valeurs réelles et prédites de la cible [24, 37].

2.5.2.1 Fonction de perte (*loss function*)

La perte est généralement un nombre non négatif, où les valeurs plus petites sont meilleures et les prédictions parfaites entraînent une perte de 0. La fonction de perte la plus courante est l'erreur quadratique (*squared error*) (équation 2.3). Lorsque notre prédiction pour un exemple i est \hat{y}_i et l'étiquette réelle correspondante est y_i , l'erreur quadratique est donnée par

$$l_i(W, b) = \frac{1}{2}(\hat{y}_i - y_i)^2, \quad (2.3)$$

6. Par exemple, un actionneur tel qu'un muscle.

Pour mesurer la qualité d'un modèle sur l'ensemble de données de n exemples, nous prenons simplement la moyenne⁷ (ou, de manière équivalente, la somme) des pertes sur l'ensemble d'entraînement (*mean squared error*) [24, 33, 38] :

$$L(W, b) = \frac{1}{n} \sum_{i=1}^n l_i(W, b) = \frac{1}{n} \sum_{i=1}^n \frac{1}{2} (WX_i + b - y_i)^2. \quad (2.4)$$

2.5.2.2 Optimisation avec l'algorithme de la descente de gradient

La technique clé pour optimiser presque tous les modèles d'apprentissage automatique consiste à réduire itérativement l'erreur en mettant à jour les paramètres dans la direction qui diminue graduellement la fonction de perte. Cet algorithme, appelé la descente de gradient, ajuste les paramètres du modèle de manière à minimiser l'erreur prédite par rapport aux valeurs réelles. Cet algorithme peut être simplifié comme suite : à chaque itération t , nous échantillons d'abord de manière aléatoire un minibatch \mathcal{B}_t composé d'un nombre fixe $|\mathcal{B}|$ d'exemples d'entraînement. Ensuite, nous calculons la dérivée (gradient) de la perte moyenne sur le minibatch par rapport aux paramètres du modèle. Enfin, nous multiplions le gradient par une petite valeur positive prédéterminée η , appelée taux d'apprentissage, et soustrayons le terme résultant des valeurs actuelles des paramètres.

Nous pouvons donc exprimer la mise à jour par la formule suivante :

$$(W, b) \leftarrow (W, b) - \frac{\eta}{|\mathcal{B}|} \sum_{i \in \mathcal{B}_t} \partial_{(W, b)} l_i(W, b). \quad (2.5)$$

2.6 Apprentissage profond

L'apprentissage profond repose sur les principes fondamentaux de l'apprentissage automatique, où les algorithmes sont conçus pour apprendre des modèles à partir des données sans être explicitement programmés. Dans ce contexte, les réseaux de neurones artificiels jouent un rôle central en imitant le fonctionnement du cerveau humain pour effectuer diverses tâches. Nous avons décrit les réseaux de neurones dans la sous-section 2.5 comme des transformations linéaires avec un biais ajouté. Un réseau de neurones ne fait que mapper les entrées directement vers les sorties via une seule transformation linéaire, suivie d'une fonction non linéaire facultative. Si nos étiquettes étaient vraiment liées aux données d'entrée par une simple transformation affine, cette approche serait suffisante. Cependant, ce n'est pas toujours le cas [38].

2.6.1 Incorporation de couches cachées

Les limitations des modèles monocouches peuvent être surmontées en incorporant une ou plusieurs couches cachées. La manière la plus simple de le faire est de superposer L couches entièrement connectées les unes sur les autres. Chaque couche alimente la couche au-dessus d'elle, jusqu'à la

7. Dans ce cas, il s'agit de l'erreur quadratique moyenne (MSE) définie dans l'équation 2.13.

sortie. Les $L - 1$ premières couches sont considérées comme la représentation du modèle et la dernière couche comme le prédicteur linéaire. Cette architecture est appelée perceptron multicouche (MLP) [37, 38].

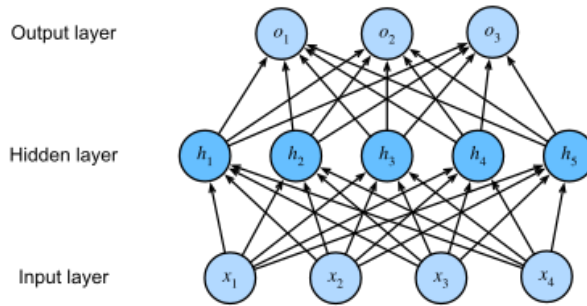


FIGURE 2.6 – Architecture du perceptron multicouche (MLP)

Le MLP illustré dans la figure 2.6 a quatre entrées, trois sorties et sa couche cachée contient cinq unités (neurones) cachées. Comme la couche d'entrée n'implique aucun calcul, la production de sorties avec ce réseau nécessite la mise en œuvre des calculs pour les couches cachées et de sortie ; ainsi, le nombre de couches dans ce MLP est de deux. Les couches cachées d'un MLP sont entièrement connectées. Chaque entrée influence chaque neurone de la couche cachée, et chacun de ceux-ci influence à son tour chaque neurone de la couche de sortie [37, 38].

2.6.2 Formalisation mathématique d'un réseau de neurones profond

Nous désignons par la matrice $X \in \mathbb{R}^{n \times d}$ un minibatch de n exemples où chaque exemple a d entrées (caractéristiques). Pour un MLP à une seule couche cachée dont la couche cachée a h unités cachées, nous désignons par $\mathbf{H} \in \mathbb{R}^{n \times h}$ les sorties de la couche cachée, qui sont des représentations cachées. Étant donné que les couches cachées et de sortie sont toutes deux entièrement connectées, nous avons des poids de couche cachée $W^{(1)} \in \mathbb{R}^{h \times d}$ et des biais $b^{(1)} \in \mathbb{R}^h$, ainsi que des poids de couche de sortie $W^{(2)} \in \mathbb{R}^{q \times h}$ et des biais $b^{(2)} \in \mathbb{R}^q$. Cela nous permet de calculer les sorties $\mathbf{O} \in \mathbb{R}^{n \times q}$ du MLP à une seule couche cachée comme suit :

$$\mathbf{H} = \sigma(XW^{(1)\top} + b^{(1)}), \quad (2.6)$$

$$\mathbf{O} = \mathbf{H}W^{(2)\top} + b^{(2)}. \quad (2.7)$$

Il est possible de superposer plusieurs couches cachées les unes au-dessus les autres, par exemple $\mathbf{H}^{(1)} = \sigma^{(1)}(XW^{(1)\top} + b^{(1)})$ et $\mathbf{H}^{(2)} = \sigma^{(2)}(\mathbf{H}^{(1)}W^{(2)\top} + b^{(2)})$, pour construire des MLP plus généraux, ce qui donne des modèles de plus en plus expressifs.

L'optimisation se déroule En appliquant la règle de la chaîne pour mettre à jour les poids et les biais à chaque couche du réseau :

$$\frac{\partial z}{\partial x} = \frac{\partial z}{\partial y} \cdot \frac{\partial y}{\partial x}, \quad (2.8)$$

Le gradient de la fonction de perte par rapport aux poids et aux biais est calculé à chaque couche,

en commençant par la couche de sortie et en travaillant en arrière vers la couche d'entrée. Cette opération est appelée rétropropagation (*backpropagation*) [37].

2.6.2.1 Fonctions d'activation

Les sorties des fonctions σ sont appelées *activations*. Les fonctions d'activation décident si un neurone doit être activé ou non, elles sont des opérateurs différentiables qui transforment les signaux d'entrée en sorties [38].

- **Unité linéaire redressée (ReLU)** : la fonction la plus utilisée, en raison de sa simplicité de mise en œuvre et de ses bonnes performances sur une variété de tâches prédictives. ReLU [41] fournit une transformation non linéaire très simple. Étant donné un élément x , la fonction est définie comme le maximum de cet élément et 0 :

$$\text{ReLU}(x) = \max(x, 0). \quad (2.9)$$

- **Fonction sigmoïde (sigmoid)** : la fonction sigmoïde transforme les entrées dont les valeurs se situent dans le domaine \mathbb{R} en sorties qui se situent dans l'intervalle $]0, 1[$. fonction est donnée par la formule suivante :

$$\text{sigmoid}(x) = \frac{1}{1 + \exp(-x)}. \quad (2.10)$$

La fonction sigmoïde est très utilisée comme fonction d'activation sur les neurones de sortie lorsque nous voulons interpréter les sorties comme des probabilités pour des problèmes de classification binaire⁸ [37]. Cependant, elle a largement été remplacée par la fonction ReLU plus simple et plus facilement entraînable pour la plupart des utilisations dans les couches cachées.

- **Tangente hyperbolique (tanh)** : la fonction \tanh transforme ses entrées en éléments dans l'intervalle $] -1, 1[$:

$$\tanh(x) = \frac{1 - \exp(-2x)}{1 + \exp(-2x)}. \quad (2.11)$$

Bien que la forme de la fonction soit similaire à celle de la fonction sigmoïde (voir la figure 2.7), la fonction \tanh présente une symétrie ponctuelle autour de l'origine du système de coordonnées [42].

- **Fonction softmax** : la fonction softmax est couramment utilisée pour transformer les scores en probabilités normalisées. Elle est particulièrement adaptée aux tâches de classification. Si nous avons un vecteur de scores $\mathbf{z} = (z_1, z_2, \dots, z_k)$, où chaque z_i est le score associé à la classe i . La fonction softmax transforme ces scores en probabilités normalisées

$$\hat{y}_i = \frac{\exp(z_i)}{\sum_j \exp(z_j)}, \quad (2.12)$$

8. Nous pouvons considérer cette fonction comme un cas particulier de la fonction softmax (voir l'élément 2.6.2.1).

où \hat{y}_i est la probabilité que l'entrée appartienne à la classe i . La somme des \hat{y}_i pour toutes les classes est garantie d'être égale à 1 [38]. La fonction softmax est souvent utilisée conjointement à la fonction de perte de l'entropie croisée (voir l'élément 2.6.2.2).

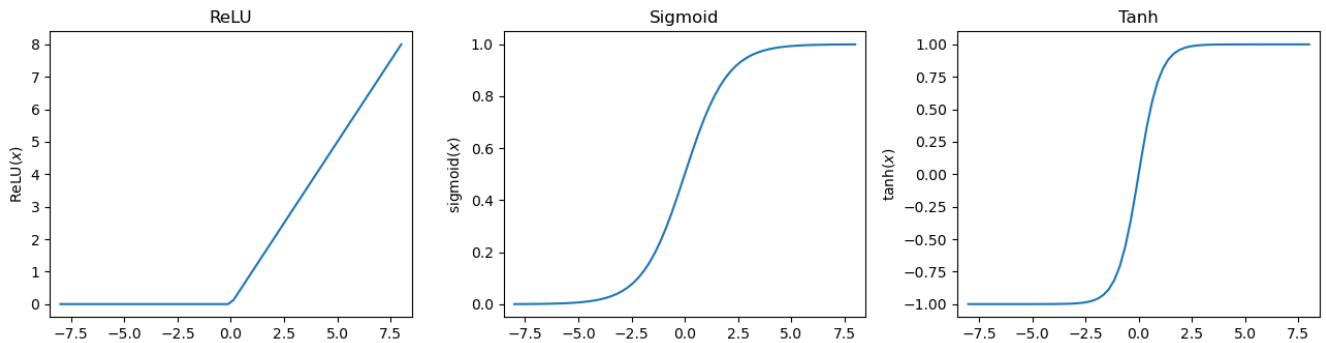


FIGURE 2.7 – Courbes des fonctions d'activation mentionnées

2.6.2.2 Fonctions de perte pour l'apprentissage profond

- **L'erreur quadratique moyenne (MSE de l'anglais Mean Square Error)** : l'erreur quadratique moyenne est utilisée pour évaluer la qualité d'un modèle de régression. Elle mesure la moyenne des carrés des différences entre les valeurs prédites et les valeurs réelles [37, 38] :

$$\text{MSE}(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (2.13)$$

- **L'erreur absolue moyenne (MAE de l'anglais Mean Absolute Error)** : est une autre mesure d'erreur similaire pour les modèles de régression. Elle calcule la moyenne des valeurs absolues des différences entre les valeurs prédites et les valeurs réelles [37, 38] :

$$\text{MAE}(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|. \quad (2.14)$$

- **Variation perte de l'entropie croisée (cross-entropy)** : cette mesure est utilisée pour évaluer la qualité des modèles de classification. Elle mesure la divergence entre les distributions de probabilité des classes réelles et prédites [38] :

$$\text{CrossEntropyLoss}(y, \hat{y}) = - \sum_{i=1}^n y_i \log(\hat{y}_i) - (1 - y_i) \log(1 - \hat{y}_i). \quad (2.15)$$

2.6.2.3 Algorithmes d'optimisation

Il existe plusieurs algorithmes d'optimisation, les deux algorithmes les plus utilisés sont :

- **Adaptive moment estimation (Adam de l'anglais Adaptive moment estimation)** : Adam intègre plusieurs techniques efficaces pour l'optimisation en apprentissage profond,

offrant robustesse et efficacité. Introduit par Kingma et Ba en 2014 [43], Adam aborde des problèmes tels que la gestion des données redondantes et l'amélioration de la vitesse de convergence. Malgré sa popularité, il peut rencontrer des problèmes de divergence, qui ont été traités par des algorithmes ultérieurs comme Yogi proposé par Zaheer et al. en 2018 [44].

- **Propagation de la racine carrée de la moyenne (RMSProp de l'anglais Root Mean Square Propagation)** : est un algorithme d'optimisation qui utilise une moyenne mobile des gradients au carré pour ajuster le taux d'apprentissage [45].

Il est utile de mentionner que changer le taux d'apprentissage pendant l'entraînement peut entraîner une amélioration de la précision et une réduction du surajustement du modèle [37]. De plus, une diminution par paliers du taux d'apprentissage chaque fois que le progrès stagne est efficace en pratique. Cela garantit une convergence efficace vers une solution appropriée, suivie d'une réduction uniquement de la variance inhérente des paramètres en diminuant le taux d'apprentissage [43, 46].

2.7 Réseaux neuronaux convolutifs (CNN)

Dans cette section, nous abordons les réseaux neuronaux convolutionnels (CNN, pour "Convolutional Neural Networks" en anglais)[47]. Les architectures basées sur les CNN sont désormais omniprésentes dans le domaine de la vision par ordinateur. Par exemple, sur la collection Imagenet [48], ce sont uniquement les réseaux neuronaux convolutionnels, ou Convnets, qui ont permis des améliorations significatives de performance[49].

Les CNN sont le fruit d'une conception basée sur des principes biologiques, des concepts issus de la théorie des groupes, et de recherches expérimentales approfondies. En plus de leur efficacité d'échantillonnage pour obtenir des modèles précis, les CNN ont tendance à être efficaces en termes de calcul, à la fois parce qu'ils nécessitent moins de paramètres que les architectures de réseaux de neurones classiques, et parce que les convolutions sont faciles à paralléliser sur les cœurs GPU [50]. Par conséquent, les praticiens utilisent fréquemment les CNN dès que possible. De plus en plus, ils émergent comme des concurrents crédibles même pour des tâches impliquant une structure de séquence unidimensionnelle, telles que l'analyse audio [51], le traitement de texte [52], et l'analyse de séries temporelles [47], où les réseaux neuronaux récurrents sont traditionnellement utilisés. Certaines adaptations astucieuses des CNN les ont également amenés à porter sur des données structurées en graphes [53] et dans les systèmes de recommandation.

2.7.1 Motivation des CNN

Les CNN sont motivés par des principes clés qui reflètent la manière dont nous percevons visuellement le monde.

1. Le premier principe est **l'invariance de translation**, qui stipule que notre modèle⁹ devrait

9. On parle alors d'un estimateur, approximateur : n'importe quelle fonction dont l'intérêt est de faire une prédiction.

répondre de manière similaire à un même patch¹⁰ peu importe où il apparaît dans l'image. Les caractéristiques importantes d'une image doivent être reconnues indépendamment de leur position [54]. Cette idée prend son évidence du fait que les objets d'intérêt peuvent apparaître n'importe où dans l'image (voir la figure 2.8) [47].

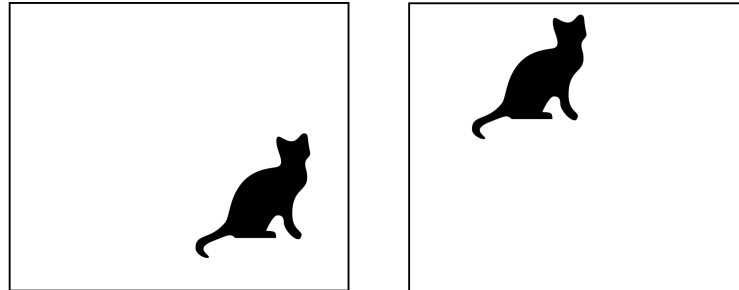


FIGURE 2.8 – Illustration de l'invariance de translation : deux images présentant un chat, avec des positions différentes / Bogdan Penkovsky, penkovsky.com

2. Le deuxième principe est celui de **la localité**. En restant dans le contexte de l'apprentissage profond, les premières couches du réseau devraient se concentrer sur des régions locales, sans tenir compte du contenu de l'image dans des régions éloignées (voir la figure 2.9). Les premières couches détectent des caractéristiques de bas niveau dans de petites régions de l'image, tandis que les couches ultérieures combinent ces caractéristiques locales pour former une représentation globale de l'image [47, 49].

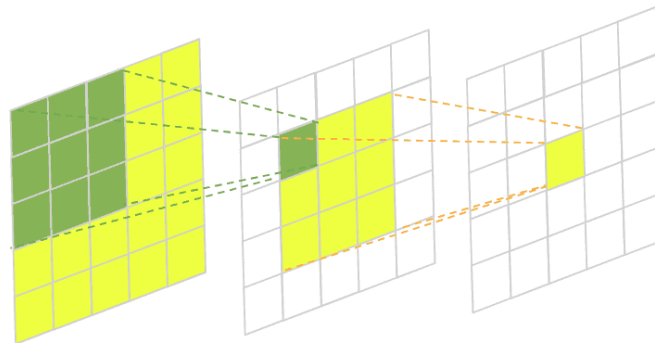


FIGURE 2.9 – Illustration du principe de la localité : les couches ultérieures combinent les caractéristiques locales pour former une représentation globale de l'image / baeldung.com

Enfin, à mesure que nous avançons dans le réseau, les couches plus profondes devraient être capables de capturer des caractéristiques à plus longue portée de l'image, de manière similaire à la vision de haut niveau dans la nature. Les couches plus profondes du réseau peuvent reconnaître des caractéristiques plus complexes et plus abstraites en combinant les caractéristiques de bas niveau détectées par les couches précédentes.

10. Un *patch* est une région d'une image.

2.7.2 Concepts de base des CNN

Les CNN respectent les deux principes de l'invariance de translation et de la localité. Elles sont formalisées à partir du concept mathématique de convolution, ou plus précisément, du concept de *corrélacion croisée* [38].

2.7.2.1 Concept de convolution

En mathématiques, la convolution ($*$) entre deux fonctions, disons $f, g : \mathbb{R}^d \rightarrow \mathbb{R}$ est définie comme

$$(f * g)(\mathbf{x}) = \int f(\mathbf{z})g(\mathbf{x} - \mathbf{z})d\mathbf{z}. \quad (2.16)$$

C'est-à-dire, nous mesurons le chevauchement entre f et g quand une fonction est *inversée* et décalée par \mathbf{x} [38, 55]. Pour les matrices bidimensionnelles, nous avons une somme correspondante¹¹ avec les indices (a, b) pour f et $(i - a, j - b)$ pour g :

$$(f * g)(i, j) = \sum_a \sum_b f(a, b)g(i - a, j - b). \quad (2.17)$$

2.7.2.2 Convolution des images

En ce qui concerne les images, elles se composent de trois *canaux* : rouge, vert et bleu. En somme, les images ne sont pas des objets bidimensionnels mais plutôt des tenseurs d'ordre trois, caractérisés par une hauteur, une largeur et un canal. Alors que les deux premiers de ces axes concernent les relations spatiales, le troisième peut être considéré comme attribuant une représentation multidimensionnelle à chaque emplacement de pixel.

Dans l'apprentissage profond, V est désigné comme un noyau (*kernel*) de convolution, un filtre, ou simplement les poids de la couche qui sont des paramètres à apprendre [38]. Nous indexons donc X comme $[X]_{i,j,k}$. Le filtre de convolution doit s'adapter en conséquence, $[V]_{a,b,c}$. Pour prendre en charge plusieurs canaux à la fois dans les entrées X et les représentations cachées \mathbf{H} , nous pouvons ajouter une quatrième coordonnée à V : $[V]_{a,b,c,d}$. En rassemblant tout, nous avons

$$[\mathbf{H}]_{i,j,d} = \sum_{a=-\Delta}^{\Delta} \sum_{b=-\Delta}^{\Delta} \sum_c [V]_{a,b,c,d} [X]_{i+a,j+b,c}, \quad (2.18)$$

où d indexe les canaux de sortie dans les représentations cachées \mathbf{H} . La couche de convolution suivante continuera à prendre un tenseur d'ordre trois, \mathbf{H} , en entrée.

11. L'intégrale se transforme en une somme pour les grilles, comme est le cas des images.

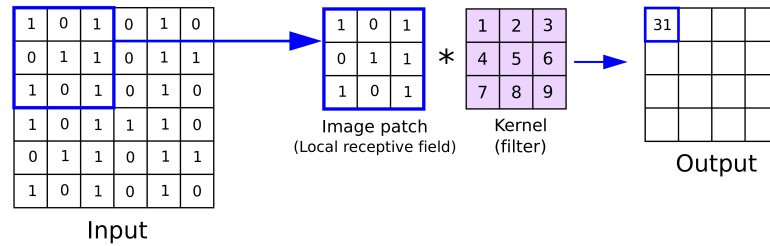


FIGURE 2.10 – Convolution d’un patch d’image avec un filtre CNN / Anh H. Reynolds, anhreynolds.com

2.7.2.3 Convolution transposée

Les couches de convolution réduisent généralement (sous-échantillonnent) les dimensions spatiales de l’entrée, ou les maintiennent inchangées. Dans la segmentation sémantique qui classe au niveau des pixels, il serait pratique que les dimensions spatiales de l’entrée et de la sortie soient les mêmes [56]. Par exemple, la dimension de canal à un pixel de sortie peut contenir les résultats de classification pour le pixel d’entrée à la même position spatiale.

Pour y parvenir, surtout après que les dimensions spatiales ont été réduites par les couches de CNN, nous pouvons utiliser un autre type de couches de CNN qui peuvent augmenter les dimensions spatiales des cartes de caractéristiques intermédiaires : la convolution transposée, qui est également appelée convolution à pas fractionnaire [57], pour inverser les opérations de sous-échantillonnage par la convolution.

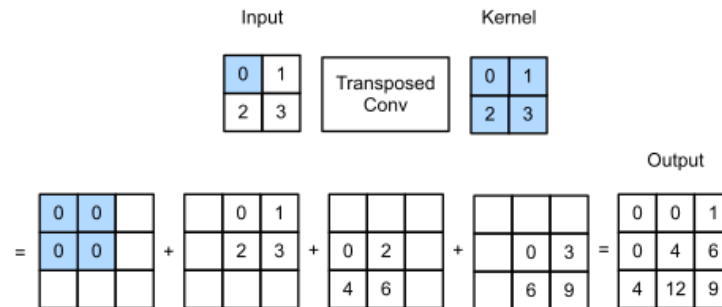


FIGURE 2.11 – Convolution transposée utilisant un noyau de dimensions 2×2 . Les zones en bleu indiquent une partie de la matrice intermédiaire, ainsi que les éléments des matrices d’entrée et du noyau employés pour le calcul / d2l.ai

Soit M une matrice d’entrée de dimensions $n_h \times n_w$ et K un noyau de dimensions $k_h \times k_w$. En faisant glisser la fenêtre du noyau avec un pas de 1, n_w fois dans chaque ligne et n_h fois dans chaque colonne, nous obtenons un total de $n_h n_w$ résultats intermédiaires. Chaque résultat intermédiaire est une matrice de dimensions $(n_h + k_h - 1) \times (n_w + k_w - 1)$ initialisée à zéro. Pour calculer chaque matrice intermédiaire, chaque élément de la matrice d’entrée est multipliée par le noyau, de sorte que la matrice résultante de dimensions $k_h \times k_w$ remplace une partie de chaque matrice intermédiaire.

Notez que la position de la partie remplacée dans chaque matrice intermédiaire correspond à la position de l'élément dans la matrice d'entrée utilisée pour le calcul. En fin de compte, tous les résultats intermédiaires sont sommés pour produire la sortie.

2.7.2.4 Concepts *padding* et *stride*

Dans les CNN, le *padding* et le *stride* sont deux concepts clés qui influencent la dimensionnalité de la sortie :

- ***Padding*** : fait référence à l'ajout d'une ou plusieurs couches de pixels fictifs autour de l'image d'entrée. Cette technique permet de conserver la taille spatiale de l'output, ce qui peut être utile pour contrôler le volume de sortie [38, 58].

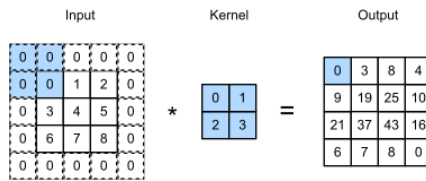


FIGURE 2.12 – Corrélation croisée bidimensionnelle avec *padding* / d2l.ai

- ***Stride*** : détermine le pas de déplacement du filtre de convolution sur l'image d'entrée. Un *stride* de 1 signifie que le filtre se déplace pixel par pixel, tandis qu'un *stride* de 2 signifie qu'il saute un pixel à chaque déplacement [38]. Un *stride* plus grand réduit la dimensionnalité de la sortie, ce qui peut aider à réduire la complexité computationnelle du modèle.

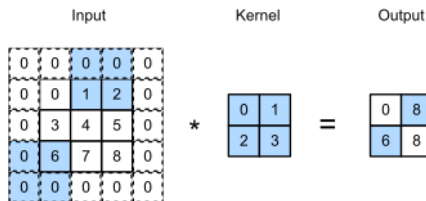


FIGURE 2.13 – Corrélation croisée avec un *stride* de 3 et 2 pour la hauteur et la largeur / d2l.ai

Pour prévoir les dimensions de sortie, on emploie le calcul

$$\text{dim. sortie} = \left\lfloor \frac{\text{dim. entrée} - \text{dim. noyau} + 2 \times \text{padding}}{\text{stride}} \right\rfloor + 1, \quad (2.19)$$

pour calculer la dimension de chaque dimension. $\lfloor \cdot \rfloor$ est la fonction partie entière (*floor*).

2.7.3 Applications des CNN

Les CNN sont une idée très importante dans le domaine de l'apprentissage profond. Contrairement aux méthodes traditionnelles, elles peuvent tirer parti d'une grande quantité de données pour obtenir des résultats prometteurs, ce qui est particulièrement pertinent à l'ère du big data. En

conséquence, de nombreuses applications sont développées. Les CNN ne sont pas seulement utilisées pour traiter des images 2D, mais aussi dans des situations à une, voire plusieurs dimensions [56].

- **Classification d’images** : consiste à attribuer une catégorie à une image. Les CNN représentent une avancée majeure dans ce domaine. LeNet-5 [59] est considéré comme la première application utilisée pour la classification de chiffres manuscrits. AlexNet [49] a permis de lancer les approches de classification basées sur les CNN. Ensuite, Simonyan et al. [60] ont souligné l’importance de la profondeur, mais ces CNN primitifs ne comportent pas plus de dix couches. Par la suite, des structures de réseaux plus profondes ont émergé, telles que GoogLeNet [61] et VGGNets [60], qui ont considérablement amélioré la précision dans les tâches de classification.
- **Segmentation d’images** : consiste à diviser une image en différentes zones. En 2014, Long et al. [62] ont proposé le concept de réseaux convolutifs entièrement convolutifs (FCN) et ont appliqué des structures CNN à la segmentation sémantique d’images pour la première fois. En 2015, Ronneberger et al. [63] ont proposé U-Net, qui présente des caractéristiques multi-échelles plus importantes et a été appliqué à la segmentation d’images médicales. En ce qui concerne les tâches de segmentation d’instances, He et al. [64] ont proposé Mask-RCNN qui partage les caractéristiques de convolution entre deux tâches grâce à une structure en cascade.
- **Reconnaissance faciale** : est une technique d’identification biométrique basée sur les caractéristiques du visage humain. DeepFace [65] et DeepID [66] ont obtenu d’excellents résultats sur l’ensemble de données LFW [67], dépassant les performances humaines pour la première fois dans des scénarios non contraints. Par la suite, les approches basées sur l’apprentissage profond ont suscité beaucoup plus d’attention. Le processus de DeepFace proposé par Taigman et al. [65] comprend la détection, l’alignement, l’extraction et la classification.

2.8 Conclusion

Ce chapitre a exploré en profondeur les principes fondamentaux de l’apprentissage profond, en mettant particulièrement l’accent sur les réseaux neuronaux convolutifs (CNN). Nous avons vu comment les CNN, inspirés de l’organisation du cortex visuel humain, permettent d’analyser et de traiter efficacement des données visuelles complexes. Les CNN se sont révélés particulièrement adaptés à des tâches de reconnaissance d’images, de classification, de détection d’objets et de segmentation d’images, offrant des performances inégalées dans ces domaines. Leur capacité à extraire automatiquement des caractéristiques pertinentes à partir de données brutes et à les combiner pour former des représentations hiérarchiques a permis des avancées significatives dans de nombreuses applications pratiques, notamment la détection de plusieurs types de cancers.

CHAPITRE 3

Réseaux antagonistes génératifs

3 Réseaux antagonistes génératifs

Ce chapitre est consacré à l'étude des réseaux antagonistes génératifs, en mettant particulièrement l'accent sur les réseaux antagonistes génératifs à convolution profonde. Il s'agit d'une technologie de pointe dans le domaine de l'apprentissage profond, révolutionnant la manière dont les machines peuvent générer de nouvelles données similaires à celles utilisées pour l'entraînement. Cela permet de construire des modèles hybrides GAN-CNN puissants et utiles dans plusieurs domaines, notamment pour la détection de cancers.

3.1 Introduction

Ce chapitre a pour objectif de fournir une compréhension approfondie des concepts fondamentaux des réseaux antagonistes génératifs (GAN, de l'anglais "Generative Adversarial Networks"), en préparation à l'étude des réseaux antagonistes génératifs à convolution profonde (DCGAN, de l'anglais "Deep Convolutional Generative Adversarial Networks") qui seront utilisés dans notre application pour la détection du cancer du foie. Ce chapitre est divisé en trois parties :

- Nous commencerons par une étude des réseaux antagonistes génératifs (GAN), en détaillant leur architecture, leur fonctionnement, et leur processus d'entraînement et d'optimisation.
- La deuxième partie sera consacrée aux réseaux antagonistes génératifs à convolution profonde (DCGAN). Nous introduirons le concept, discuterons de leur architecture et de leur fonctionnement, et explorerons des techniques spécifiques d'entraînement et d'optimisation pour les DCGAN.
- Ensuite, nous présenterons un état de l'art sur l'utilisation des réseaux GAN pour la détection du cancer du foie.

3.2 Réseaux antagonistes génératifs (GANs)

Les réseaux antagonistes génératifs (GAN) sont une architecture révolutionnaire dans le domaine de l'intelligence artificielle. Introduits par Goodfellow et al. [68]. en 2014, ils représentent une technologie très prometteuse en apprentissage automatique. Les GAN utilisent des méthodes d'apprentissage profond, comme les réseaux de neurones convolutifs, pour la modélisation générative, permettant de découvrir et d'apprendre automatiquement les modèles présents dans les données d'entrée. Ils reposent sur un concept ingénieux de deux sous-modèles : le générateur, formé pour produire de nouveaux exemples, et le discriminateur, qui cherche à distinguer les exemples réels des exemples générés [69].

3.2.1 Introduction aux GAN

Les GAN se composent principalement de deux modèles de réseaux neuronaux en compétition, capables d'analyser, de saisir et de reproduire les variations présentes dans un ensemble de données.

Ces deux réseaux sont engagés dans un jeu à somme nulle, où chacun cherche à surpasser l'autre. Les deux modèles de réseaux neuronaux dans un GAN sont connus sous le nom de générateur et de discriminateur. Le générateur est un type de réseau neuronal convolutif chargé de créer de nouvelles instances d'un objet, tandis que le discriminateur est un type de réseau neuronal convolutif chargé de déterminer l'authenticité de cet objet ou son appartenance à un ensemble de données [70].

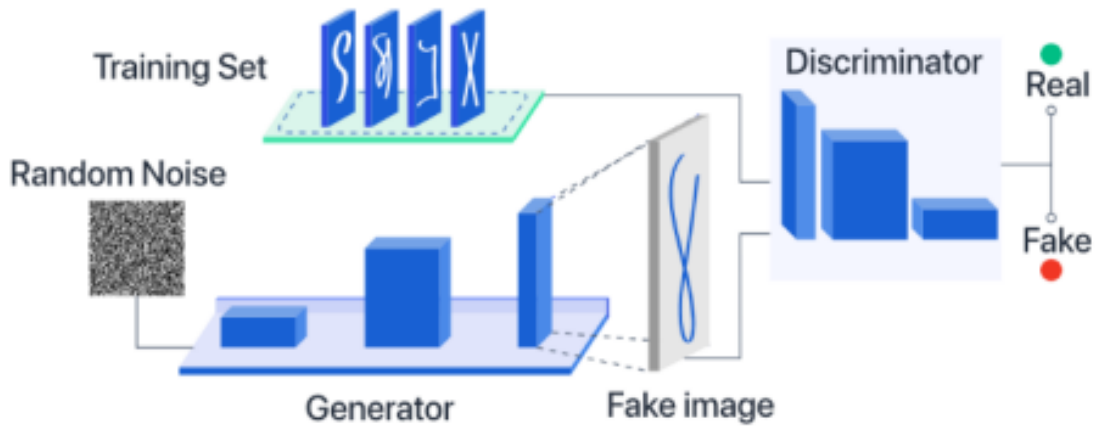


FIGURE 3.1 – Réseaux Antagonistes Génératifs

3.2.2 Architecture et fonctionnement des GANs

L'architecture du modèle GAN implique deux composantes principales : un générateur et un discriminateur [71]. Ces deux composantes sont des réseaux de neurones et fonctionnent en compétition lors de la phase d'entraînement. Les étapes d'entraînement sont répétées plusieurs fois, ce qui permet au générateur et au discriminateur de s'améliorer progressivement dans leurs tâches respectives à chaque itération.

- Le générateur a pour objectif de capturer la distribution des données et est entraîné pour maximiser la probabilité que le discriminateur commette une erreur.
- le discriminateur est basé sur un modèle qui estime la probabilité que l'échantillon qu'il reçoit provienne des données d'apprentissage plutôt que du générateur.

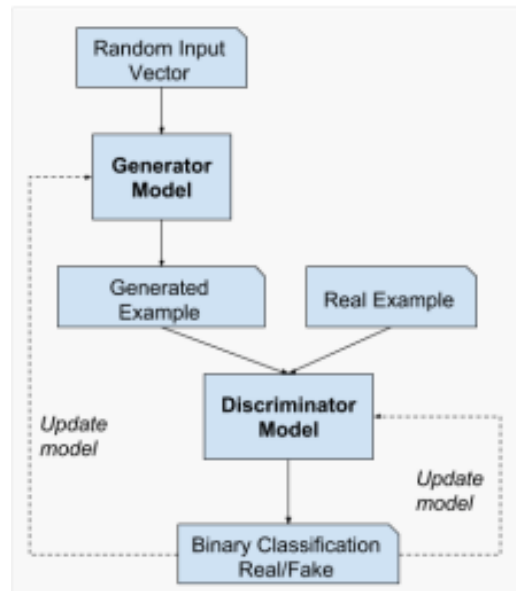


FIGURE 3.2 – Exemple d’architecture de modèle de réseau antagoniste génératif [1]

Les GANs sont formulés comme un jeu minimax, où le Discriminateur essaie de minimiser sa récompense $V(D, G)$ et le Générateur essaie de minimiser la récompense du Discriminateur ou en d’autres termes, de maximiser sa perte. Il peut être décrit mathématiquement par la formule ci-dessous :

$$\min_G \max_D V(D, G) \quad (3.1)$$

$$V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})} [\log(1 - D(G(\mathbf{z})))] \quad (3.2)$$

où : G = générateur ; D = discriminateur

$P_{\text{data}}(x)$ = distribution des données réelles ;

$P(z)$ = distribution du générateur ;

x = échantillon de $P_{\text{data}}(x)$; z = échantillon de $P(z)$

$D(x)$ = réseau discriminateur ; $G(z)$ = réseau de générateur

3.2.2.1 Modèle de générateur

Le modèle générateur est utilisé pour créer de nouveaux exemples plausibles à partir du domaine du problème. Il prend en entrée un vecteur aléatoire de longueur fixe et génère un échantillon dans ce domaine. Ce vecteur est tiré aléatoirement d’une distribution gaussienne et sert à initier le processus de génération.

Après l'entraînement, les points de cet espace vectoriel multidimensionnel correspondent aux points du domaine du problème, formant ainsi une représentation compressée de la distribution des données. Cet espace vectoriel est appelé espace latent ou espace vectoriel composé de variables latentes. Les variables latentes, ou variables cachées, sont des variables importantes pour un domaine mais qui ne sont pas directement observables.

Une fois l'apprentissage terminé, le modèle générateur est conservé et utilisé pour produire de nouveaux échantillons [72].

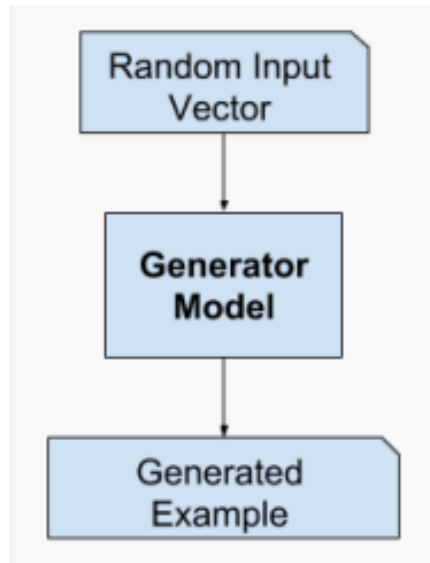


FIGURE 3.3 – Exemple de modèle générateur [1]

3.2.2.2 Modèle de discriminateur

Le modèle utilisé pour classer les exemples générés par le générateur comme vrais (provenant du domaine) ou faux (générés).

Le modèle discriminateur prend un exemple en entrée (réel ou généré) et prédit une étiquette de classe binaire indiquant s'il est réel ou faux (généré). Les exemples réels proviennent de l'ensemble de données d'apprentissage, tandis que les exemples générés sont produits par le modèle générateur.

Le discriminateur est un modèle de classification classique. Après l'apprentissage, le discriminateur est mis de côté et l'entraînement du générateur commence [73].

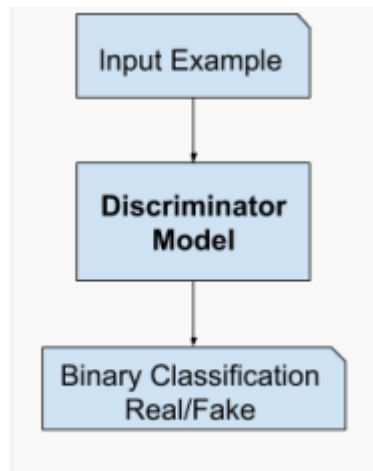


FIGURE 3.4 – Exemple de modèle discriminateur [1]

3.2.3 Avantages et inconvénients

Les GANs présentent plusieurs avantages et inconvénients dont nous citons les plus importants ci-dessous :

3.2.3.1 Avantages

- Apprentissage par rétropropagation,
- Capacité à synthétiser des images à des niveaux de réalisme sans précédent,
- Permet l'élaboration de modèles plus performants, avec un taux d'erreur fortement réduit.

3.2.3.2 Inconvénients

- Non-convergence : les paramètres oscillent fortement sans se stabiliser.
- Mode collapse : le générateur produit des observations avec très peu de variété.
- Difficulté d'entraînement.

3.3 Réseaux antagonistes génératifs à convolution profonde (DCGANs)

3.3.1 Introduction aux DCGANs

Les réseaux antagonistes génératifs à convolution profonde (DCGAN, de l'anglais "Deep Convolutional Generative Adversarial Networks") sont une extension des réseaux GAN conçue pour améliorer la qualité des images générées en utilisant des réseaux de neurones convolutifs profonds [74]. Cette architecture a été introduite en 2015 par Alec Radford, Luke Metz et Soumith Chintala. Les DCGAN se distinguent par plusieurs innovations clés qui les rendent particulièrement efficaces pour la génération d'images réalistes :

- **Convolution sans Pooling** : Les couches de pooling sont remplacées par des convolutions striées (pour le discriminateur) et des convolutions fractionnelles-striées (pour le générateur).
- **Normalisation par Lots** : Utilisation de la normalisation par lots (batch normalization) dans les deux sous-modèles pour stabiliser et accélérer l'entraînement
- **Suppression des Couches Complètement Connectées** : Pour des architectures plus profondes, les couches entièrement connectées sont retirées.
- **Fonctions d'Activation** : Utilisation de ReLU pour toutes les couches du générateur sauf la sortie, qui utilise tanh, et LeakyReLU pour toutes les couches du discriminateur.

Ces caractéristiques permettent aux DC-GANs de générer des images avec des détails plus fins et une meilleure qualité globale par rapport aux GANs traditionnels [75].

Remarques :

- **Convolution striée (convolution avec un pas (stride))** : est une opération de convolution dans laquelle le noyau de convolution (filtre) se déplace sur l'image d'entrée avec un certain pas ou un certain décalage entre chaque déplacement. Contrairement à la convolution classique où le noyau se déplace pixel par pixel, la convolution striée permet au noyau de se déplacer sur l'image avec un pas donné, ce qui peut réduire la dimension de la sortie.
- **Convolution fractionnée-striée** : est une variation de la convolution striée dans laquelle le noyau de convolution est appliqué avec un certain pas, mais les sorties de différentes parties du noyau sont combinées de manière fractionnée. Cela permet d'augmenter la dimension de la sortie par rapport à une convolution striée traditionnelle.
- **Leaky ReLU (Rectified Linear Unit avec fuite)** : Contrairement à la fonction ReLU standard qui mappe toutes les valeurs négatives à zéro, la fonction Leaky ReLU mappe les valeurs négatives à une petite pente plutôt que à zéro. Cela permet de résoudre le problème de la disparition du gradient pour les valeurs négatives rencontré avec la fonction ReLU standard, ce qui peut améliorer la convergence et la performance du réseau de neurones. La formule de Leaky ReLU est donnée comme suit :

$$\text{LeakyReLU}(x) = \begin{cases} x & \text{si } x \geq 0 \\ \alpha \cdot x & \text{si } x < 0 \end{cases}$$

Où α est une petite pente (généralement proche de zéro), permettant un léger passage des valeurs négatives.

3.3.2 Architecture et fonctionnement des DCGANs

Le réseau DCGAN est une variante des réseaux GAN qui utilise des réseaux de neurones convolutifs (CNN) pour son discriminateur et un réseau basé sur des convolutions pour son générateur. Pour mettre en œuvre un DCGAN, il est nécessaire de spécifier les trois éléments clés suivants :

- **Le Discriminateur** : Un réseau convolutif qui évalue et différencie les images réelles des

images générées.

- **Le Générateur** : Un réseau convolutif qui crée de nouvelles images à partir d'un vecteur latent.
- **La Procédure d'Apprentissage (Training Procedure)** : Le processus par lequel le générateur et le discriminateur sont entraînés de manière adversariale.

Cette architecture permet d'améliorer la qualité des images générées par rapport aux GANs traditionnels en utilisant les propriétés des réseaux convolutifs pour capturer les caractéristiques spatiales des données d'image. Le discriminateur dans l'architecture du DCGAN est un CNN qui suit le schéma bloc montré sur la Figure 3.5

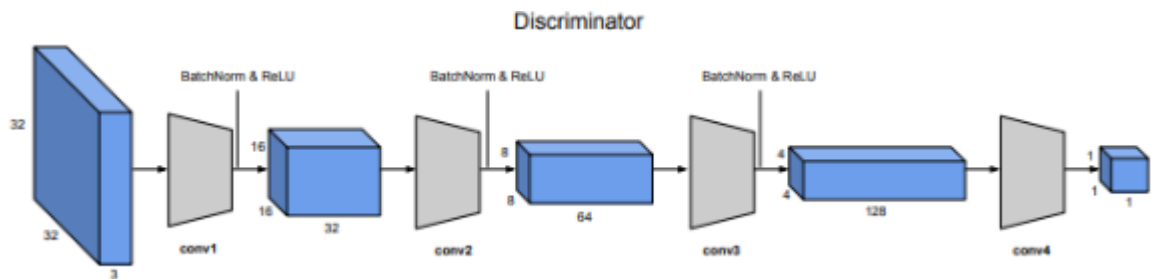


FIGURE 3.5 – Architecture du discriminateur d'un DCGAN

À chaque strate de convolution, un sous-échantillonnage spatial est réalisé avec un facteur de réduction de 2. Dans chaque strate, des opérations de convolution sont effectuées entre les données d'entrée et les noyaux de convolution.

Supposons que les données d'entrée soient de dimension $n_h \times n_w$ et que le noyau de convolution soit de dimension $k_h \times k_w$. Dans ce cas, la dimension de la sortie du modèle sera $(n_h - k_h + 1) \times (n_w - k_w + 1)$. Le rembourrage (padding) est utilisé pour traiter les problèmes de bord. Dans cette étude, nous avons choisi d'ajouter des zéros pour gérer ces problèmes lors de l'étape de convolution. La convolution est réalisée avec des noyaux spécifiques à chaque strate. En général, en ajoutant p_h lignes (en haut et en bas de l'entrée) et p_w colonnes de rembourrage (de zéros à gauche et à droite de l'entrée), la taille de la sortie du modèle sera $(n_h - k_h + p_h + 1) \times (n_w - k_w + p_w + 1)$. Dans de nombreux cas, $p_h = k_h - 1$ et $p_w = k_w - 1$.

Après la convolution, vient la normalisation par lot (batch normalization) suivie de l'activation ReLU. La normalisation par lot est une technique avancée de formation de réseaux neuronaux qui normalise les entrées de chaque couche pour chaque mini-lot, stabilisant ainsi le processus d'apprentissage et réduisant le nombre d'époques nécessaires pour former des réseaux profonds, ce qui accélère l'entraînement.

Le générateur du DCGAN consiste en une séquence de couche de convolution qui sur échantillonnent progressivement l'input bruit pour générer l'image fausse (fake). La figure 3.6 illustre l'architecture du générateur.

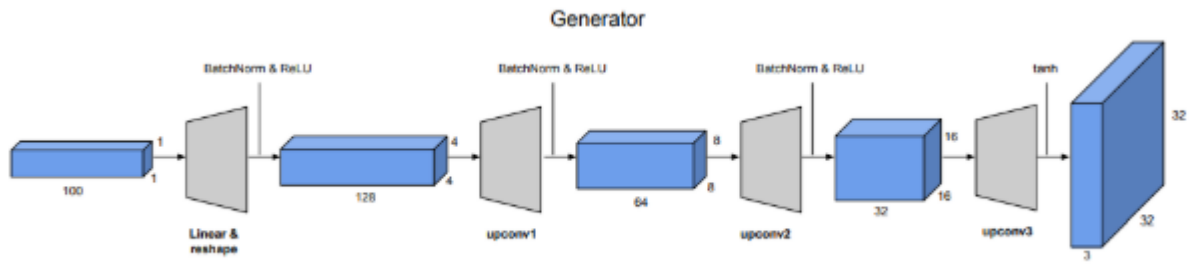


FIGURE 3.6 – Architecture de générateur de DCGAN

Le bloc upconv consiste en une couche de sur-échantillonnage (par simple interpolation bilinéaire) suivie d'une convolution 2D. L'architecture globale du DCGAN est donc représentée en regroupant le générateur et le discriminateur dans une seule structure comme c'est montré sur la Figure 3.7

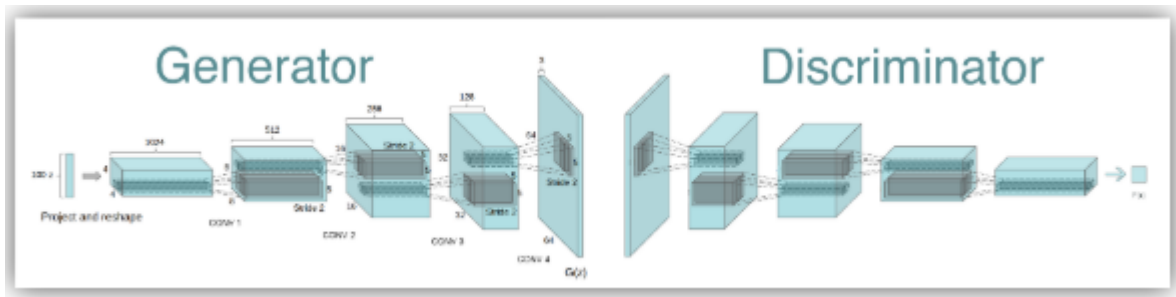


FIGURE 3.7 – Architecture générale de DCGAN

3.4 Défis et perspectives futures

Les DCGANs ont été une avancée majeure dans le domaine de la vision par ordinateur et de l'apprentissage profond. Cependant, malgré leurs succès, plusieurs défis et perspectives futures persistent :

- **Stabilité de l'entraînement** : L'entraînement des DCGANs peut être instable, avec des problèmes tels que le mode collapse où le générateur produit des échantillons similaires et répétitifs [76]. Améliorer la stabilité de l'entraînement reste un défi, mais des techniques comme le recours à des architectures plus complexes [77] ou l'utilisation de techniques d'optimisation plus avancées [78] peuvent aider à atténuer ce problème.
- **Contrôle du résultat de la génération** : Bien que les DCGANs soient capables de générer des images de qualité, contrôler précisément les caractéristiques des images générées reste un défi [79]. Des méthodes telles que le conditionnement du générateur avec des vecteurs de classe [80] ou l'utilisation de mécanismes d'attention [79] peuvent permettre un contrôle plus précis sur le processus de génération.
- **Évaluation de la qualité des échantillons** : Développer des métriques objectives pour évaluer la qualité des images générées par les DCGANs est crucial [81]. Des approches telles

que l'utilisation de réseaux de neurones discriminatoires supplémentaires [78] ou l'analyse de la similarité des embeddings [79] peuvent fournir des mesures plus fiables de la qualité des échantillons générés.

- **Génération de données diverses et réalistes** : Les DCGANs ont montré des performances impressionnantes dans la génération d'images réalistes [76], mais étendre cette capacité à d'autres types de données reste un défi [77]. Des recherches continues sont nécessaires pour adapter les architectures et les méthodes d'entraînement des DCGANs à ces différents types de données.
- **Interprétabilité et compréhension des modèles** : Comprendre comment et pourquoi les DCGANs génèrent des échantillons spécifiques est crucial pour une utilisation sûre et éthique de ces modèles . Améliorer l'interprétabilité des modèles reste un défi majeur , mais des approches telles que la visualisation des activations des couches internes ou l'utilisation de techniques d'interprétabilité spécifiques aux réseaux de neurones peuvent contribuer à éclairer le fonctionnement des DCGANs.

Les GAN ont révolutionné le domaine de l'apprentissage profond en permettant la génération de données réalistes. Malgré les défis tels que l'instabilité de l'entraînement et l'évaluation de la qualité des échantillons générés, les GAN offrent un potentiel immense pour la génération de contenus multimédias, la synthèse de données et bien d'autres applications. Ces défis nécessitent des recherches continues pour exploiter pleinement leur potentiel.

3.5 État de l'art de l'utilisation des GAN pour le traitement des images médicales

Les GAN ont été largement utilisés ces dernières années pour le traitement des images médicales et, de manière plus générale, dans le domaine de la santé. Les GAN ont suscité un intérêt croissant en raison de leur capacité à générer des données réalistes à partir de distributions de données existantes. Dans le domaine médical, les GAN ont été explorés selon plusieurs axes de recherche et d'application, allant de la segmentation des organes à la détection des anomalies et à l'augmentation des données médicales. Ces techniques offrent des perspectives prometteuses pour améliorer le diagnostic, la planification des traitements et la compréhension des maladies. Les sections suivantes résument quelques travaux et contributions significatifs dans ce domaine.

3.5.1 Segmentation du Foie avec des GAN Basés sur les Transformers

Le travail de Wang et al. publié en 2022, intitulé "Transformer based Generative Adversarial Network for Liver Segmentation", se concentre sur l'application des GAN en combinaison avec des architectures de transformers pour la segmentation du foie. Cette méthode novatrice utilise un générateur basé sur un transformer pour prédire les masques de segmentation du foie à partir des images médicales en entrée. En parallèle, des discriminateurs sont utilisés pour évaluer la qualité et la vraisemblance des masques générés par rapport aux annotations réelles fournies par des experts médicaux. L'approche proposée par Wang et al. repose sur l'utilisation d'un générateur transformer, une architecture de réseau neuronal bien adaptée à la compréhension des relations spatiales complexes dans les données d'image (voir la figure 3.8). En combinant cette architecture avec des GAN, l'équipe de recherche a pu exploiter les capacités de génération des GAN tout en bénéficiant des avantages des transformers pour la segmentation précise du foie. Les résultats de leur étude ont démontré une performance remarquable, avec un coefficient Dice de 0,9433, indiquant une segmentation précise et robuste du foie. Ces résultats ont été validés qualitativement par des experts médicaux, soulignant ainsi la pertinence et l'efficacité de l'approche proposée par Wang et al. pour la segmentation du foie dans les images médicales. La figure 3.8 illustre visuellement la qualité des segments de foie générés par leur méthode, confirmant la précision et la fidélité des résultats obtenus.

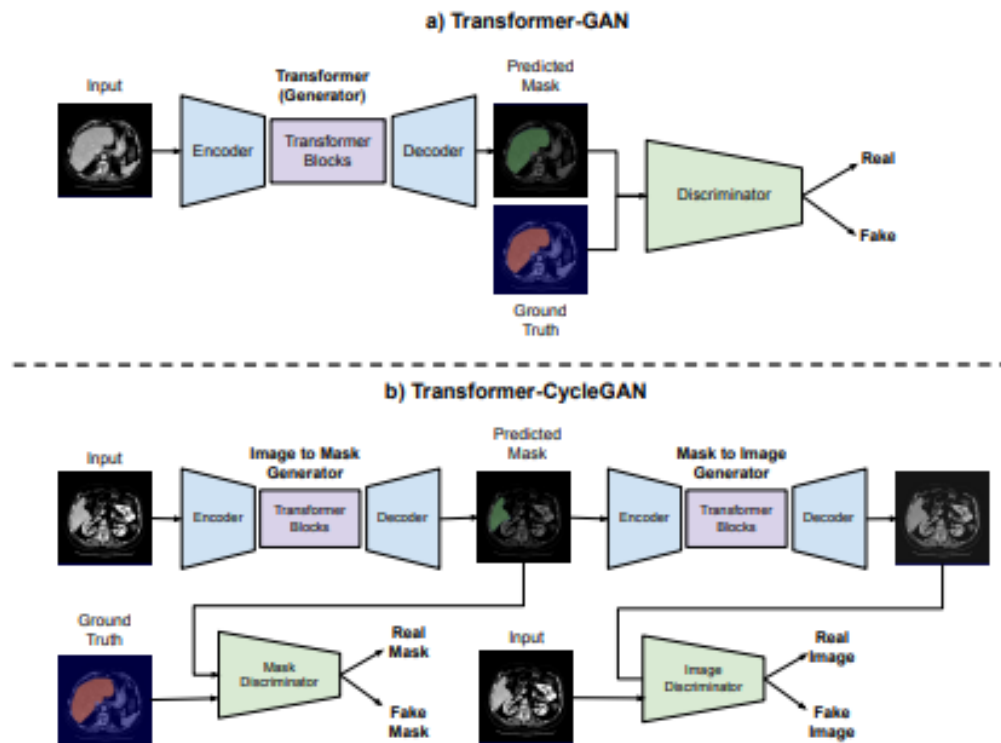


FIGURE 3.8 – Schéma bloc du Transformer GAN. (a) GAN vanille et (b) CycleGAN avec architectures de générateur Transformer [82]

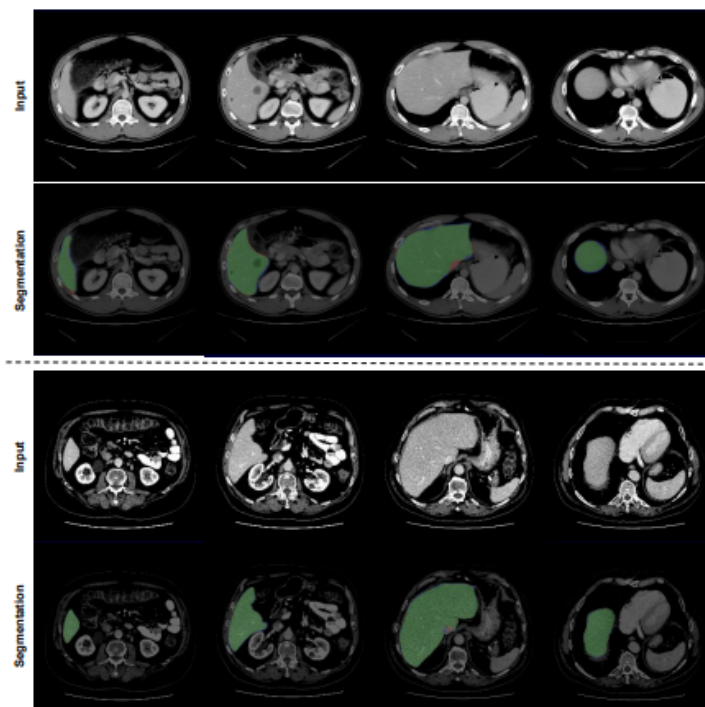


FIGURE 3.9 – Résultats de segmentation du foie basés sur le GAN Transformer. Vert : Vrai positif, Rouge : Faux positif, Bleu : Faux négatif [82]

3.5.2 Apprentissage Non Supervisé avec les DC-GAN

Le travail de Radford et al. en 2015, intitulé "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (DCGAN)", met en lumière les architectures DCGAN pour l'apprentissage non supervisé à partir de données d'image. Les DCGAN améliorent la stabilité de l'entraînement des GAN en utilisant des convolutions striées, la normalisation par lots et des fonctions d'activation spécifiques. Cette approche permet non seulement de générer des images synthétiques de haute qualité, mais aussi d'extraire des caractéristiques efficaces pour des tâches supervisées telles que la classification d'images [83]. La figure 3.10 présente l'architecture du générateur DCGAN utilisée pour la modélisation des scènes LSUN.

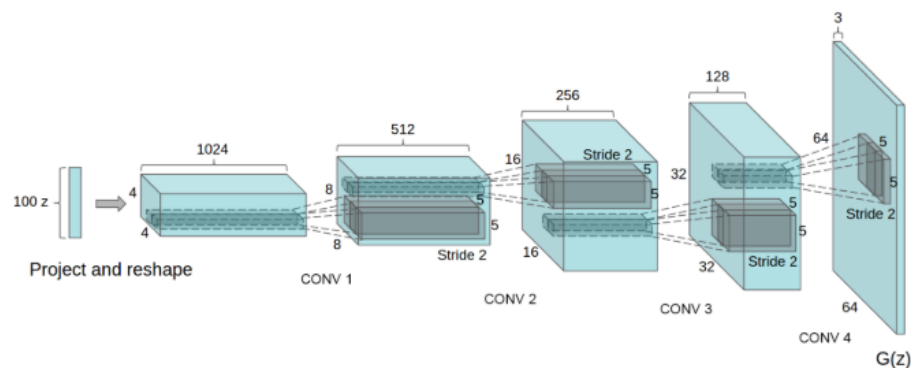


FIGURE 3.10 – DCGAN generator used for LSUN scene modeling [83]

3.5.3 Augmentation des Images Médicales avec des GAN

Le travail de Makhoul et al. en 2023, intitulé "The use of generative adversarial networks in medical image augmentation" aborde l'utilisation des GAN pour augmenter les images médicales. Dans cet article, une revue systématique de la littérature met en évidence que l'augmentation basée sur les GAN améliore la performance des modèles de traitement des images médicales. Les auteurs ont évalué cette méthode à la fois qualitativement et quantitativement, tout en fournissant une description détaillée des modèles et des hyperparamètres. Cette étude met en lumière le potentiel des GAN dans le domaine médical, en particulier en ce qui concerne l'amélioration de la performance des modèles grâce à l'augmentation des données [84].

3.5.4 Détection des Anomalies dans les Images Médicales

Le travail de Dylan et al., intitulé "Generative Adversarial Networks for Anomaly Detection in Medical Images : A Comprehensive Study" publié en 2024, met en lumière l'importance de la détection précoce des anomalies dans les images médicales pour le diagnostic et le traitement des maladies. Les GAN, initialement conçus pour la génération d'images, sont adaptés pour apprendre les motifs normaux et détecter les anomalies. Diverses architectures GAN, telles que AnoGAN,

DAGAN et AD-GAN, sont passées en revue, démontrant leur efficacité dans différents scénarios de détection d'anomalies. Les défis tels que la rareté des données et le déséquilibre des classes sont également abordés [85].

3.5.5 Synthèse et Perspectives

L'utilisation des GAN dans le domaine médical présente plusieurs avantages :

- **Segmentation Précise** : Les GAN combinés avec des transformers améliorent la segmentation d'organes tels que le foie, offrant des performances élevées validées par des experts médicaux.
- **Apprentissage Non Supervisé** : Les DCGAN permettent l'apprentissage de représentations hiérarchiques à partir de données non étiquetées, utiles pour des tâches supervisées.
- **Augmentation des Données** : Les GAN augmentent les jeux de données médicaux, améliorant ainsi la performance des modèles de traitement des images.
- **Détection des Anomalies** : Les GAN détectent efficacement les anomalies dans les images médicales, surpassant les méthodes traditionnelles.

Les futures recherches devraient se concentrer sur l'intégration des données multi-modales, l'amélioration des architectures GAN et l'évaluation des modèles dans des contextes cliniques réels pour garantir leur applicabilité et leur robustesse dans des environnements médicaux variés.

3.6 Conclusion

Les GANs (Generative Adversarial Networks) et les DCGANs (Deep Convolutional Generative Adversarial Networks) ont révolutionné le domaine de l'apprentissage profond, offrant des méthodes novatrices pour générer des données réalistes dans divers domaines. Ces réseaux ont prouvé leur efficacité dans des applications telles que la vision par ordinateur, le traitement du langage naturel et la synthèse audio, en générant des données à partir de distributions latentes et en exploitant la puissance des réseaux convolutifs pour produire des images de haute qualité.

Dans le domaine médical, les GANs et les DCGANs ont montré un potentiel immense. Ils sont utilisés pour des tâches cruciales comme la segmentation précise des organes, l'augmentation des jeux de données médicaux et la détection des anomalies dans les images médicales. Cependant, des défis subsistent, notamment en ce qui concerne la stabilité de l'entraînement des modèles, l'évaluation de la qualité des échantillons générés, et le contrôle précis des caractéristiques des données générées. Les recherches futures devraient se concentrer sur l'intégration des données multi-modales, l'amélioration des architectures existantes, et l'évaluation des modèles dans des contextes cliniques réels pour assurer leur robustesse et leur applicabilité.

Notre travail, qui sera détaillé dans le prochain chapitre, s'inscrit dans cette lignée de recherche. Nous explorons l'utilisation des GANs pour la détection du cancer du foie, en visant à améliorer la précision et l'efficacité des diagnostics à partir des images médicales.

CHAPITRE 4

Implémentation, résultats et évaluation

4 Implémentation, résultats et évaluation

Après avoir introduit les concepts de base des réseaux CNN, GAN et DCGAN, ce chapitre se concentre sur l'introduction du modèle développé pour la génération et la détection du cancer du foie. Les résultats du modèle proposé sont aussi présentés et évalués.

4.1 Introduction

Dans ce chapitre, nous nous concentrerons sur l'implémentation, les résultats et l'évaluation de nos modèles de génération et de détection d'images de foie. Nous proposons deux modèles distincts pour cette étude : un modèle basé sur les DCGAN pour la génération d'images de foie, et un modèle hybride DCGAN-CNN pour la détection du cancer du foie. Tout d'abord, nous détaillerons le modèle de génération d'images. Ce modèle utilise une architecture spécifique DCGAN pour créer des images de foie avec ou sans tumeur. L'objectif principal de ce modèle est de produire des images synthétiques réalistes qui peuvent être utilisées pour enrichir les jeux de données, facilitant ainsi l'entraînement et l'évaluation des algorithmes de détection. Ensuite, nous aborderons le modèle de détection du cancer du foie. Ce modèle hybride combine les capacités des DCGAN pour l'extraction de caractéristiques avec la puissance des réseaux CNN pour la classification. Cette approche permet d'identifier efficacement les images de foie présentant des tumeurs, améliorant ainsi le diagnostic précoce du cancer du foie. Nous présenterons les résultats obtenus pour chaque modèle, en analysant leur performance, leur précision et les défis rencontrés lors de leur implémentation. Enfin, une évaluation des deux modèles sera fournie.

4.2 Description de la dataset utilisée : LiTS

Le cancer du foie est le cinquième cancer le plus fréquent chez les hommes et le neuvième chez les femmes, avec plus de 840 000 nouveaux cas en 2018. Le foie est un site courant de développement de tumeurs primaires ou secondaires. En raison de leurs formes hétérogènes et diffuses, les opérations de détection, de classification et de segmentation automatique des lésions tumorales sont très difficiles. Pour cette raison, une dataset a été créée pour étudier ces cas plus en détail. La dataset LiTS (Liver and Liver Tumor Segmentation) contient plus de 50 000 images PNG du foie et des tumeurs hépatiques. Elle est principalement utilisée pour des tâches de recherche en apprentissage automatique et en imagerie médicale. La dataset inclut des images de tomodensitométrie (CT) annotées, permettant aux utilisateurs de développer et d'évaluer des modèles de détection, de classification et de segmentation pour identifier les tumeurs dans le foie. Cette dataset est diversifiée et contient des tumeurs primaires et secondaires de tailles et d'apparences variées, avec différents niveaux de lésion par rapport à l'arrière-plan. Elle a été créée en collaboration avec sept hôpitaux et instituts de recherche, et contient plus de 50 000 images de CT de l'abdomen. La dataset ainsi que les données annotées peuvent être téléchargées depuis le site web de LiTS (www.lits-challenge.com). La cohorte étudiée dans cette dataset couvre divers types de maladies tumorales hépatiques, y

compris les tumeurs primaires (comme le carcinome hépatocellulaire et le cholangiocarcinome) et les tumeurs secondaires du foie (comme les métastases des cancers colorectal, mammaire et pulmonaire). Les tumeurs présentaient des rapports lésion/fond variés (hyper- ou hypo-denses). Les images représentaient un mélange de scans abdominaux pré- et post-thérapie et ont été acquises avec différents scanners CT et protocoles d'acquisition, y compris des artefacts d'imagerie (par exemple, des artefacts métalliques) couramment trouvés dans les données cliniques réelles. Par conséquent, la dataset est considérée comme très diversifiée en ce qui concerne la résolution et la qualité des images. La résolution des images en plan varie de 0,56 mm à 1,0 mm, et l'épaisseur de coupe de 0,45 mm à 6,0 mm. Le nombre de coupes axiales varie entre 42 et 1026. Le nombre de tumeurs varie entre 0 et 12, et la taille des tumeurs varie entre 38 mm³ et 1231 mm³.

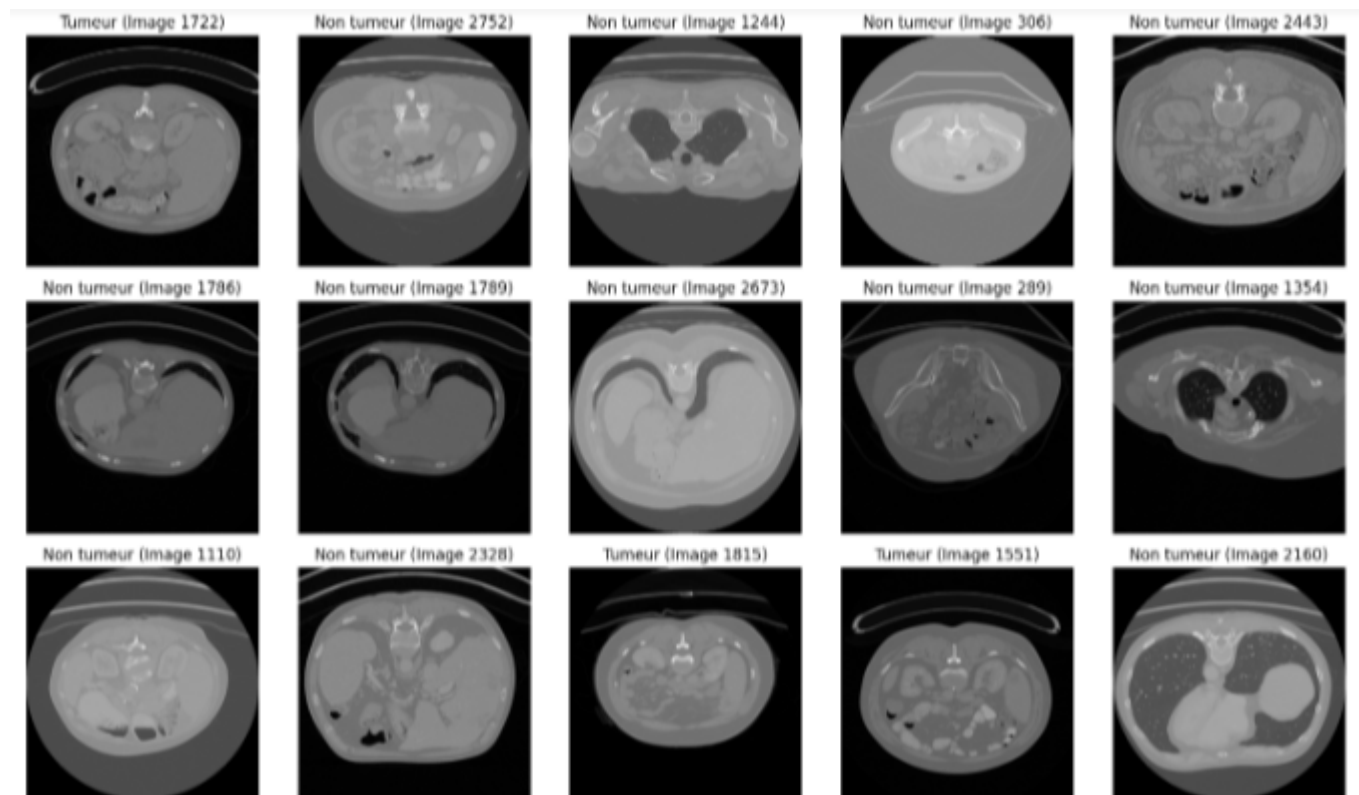


FIGURE 4.1 – Échantillon de la dataset

4.3 Outils et bibliothèques

4.3.1 Python

Python est le langage de programmation open source le plus utilisé dans le domaine de l'intelligence artificielle, particulièrement dans le Deep Learning, en raison de ses nombreuses bibliothèques performantes et utiles, notamment pour la vision artificielle et les réseaux de neurones. Conçu pour optimiser la productivité des programmeurs grâce à des outils de haut niveau et une syntaxe simple, Python s'est imposé ces dernières années comme un langage incontournable. Notre

projet, développé en Python et hébergé dans le cloud, ne nécessite pas l'installation de logiciels ou d'IDE sur notre ordinateur [86].

4.3.2 TensorFlow

TensorFlow est une plateforme open source complète pour l'apprentissage automatique, conçue pour faciliter la construction et le déploiement de modèles de machine learning de manière efficace et scalable. Développée par Google, elle offre une large gamme d'outils et de bibliothèques permettant aux chercheurs et ingénieurs de créer des modèles d'intelligence artificielle avancés, allant de simples algorithmes de classification à des réseaux de neurones profonds complexes [87].

4.3.3 Keras

Keras est une API d'apprentissage profond écrite en Python, fonctionnant sur la plateforme d'apprentissage automatique TensorFlow. Elle a été développée avec un accent particulier sur l'expérimentation. La capacité de passer d'une idée à un résultat aussi rapidement que possible est essentielle pour mener à bien des recherches efficaces [88].

4.3.4 Matplotlib

Matplotlib est une bibliothèque complète pour créer des visualisations statiques, animées et interactives en Python.

4.3.5 Pandas

Pandas est une bibliothèque Python puissante pour la manipulation et l'analyse des données. Elle offre des fonctionnalités essentielles pour les projets d'apprentissage automatique, permettant de traiter efficacement les ensembles de données qui contiennent des valeurs manquantes ou des champs nuls. Ces capacités de manipulation des données aident à améliorer les performances des modèles en garantissant des données de qualité[89].

4.3.6 NumPy

NumPy est une bibliothèque essentielle pour le langage de programmation Python, conçue pour la manipulation efficace des tableaux multidimensionnels. Elle fournit des outils puissants pour effectuer des opérations mathématiques et statistiques complexes sur des données en grande quantité, rendant le calcul scientifique plus rapide et plus intuitif.

4.3.7 Scikit-image (skimage)

Scikit-image est une collection d'algorithmes pour le traitement d'images en Python. Elle fournit une large gamme de fonctions pour des tâches telles que le filtrage, la segmentation, l'extraction de caractéristiques, et bien d'autres encore [90].

4.3.8 Google Colab

Colaboratory, ou *Colab*, est un produit de Google Research. Colab permet à tout le monde d'écrire et d'exécuter du code Python directement dans le navigateur, et est particulièrement utilisé pour l'apprentissage automatique, l'analyse de données et l'éducation. Plus précisément, Colab est un service de notebooks Jupyter hébergé [91], qui ne nécessite aucune configuration pour être utilisé, tout en offrant un accès gratuit à des ressources informatiques, y compris les GPU [92].

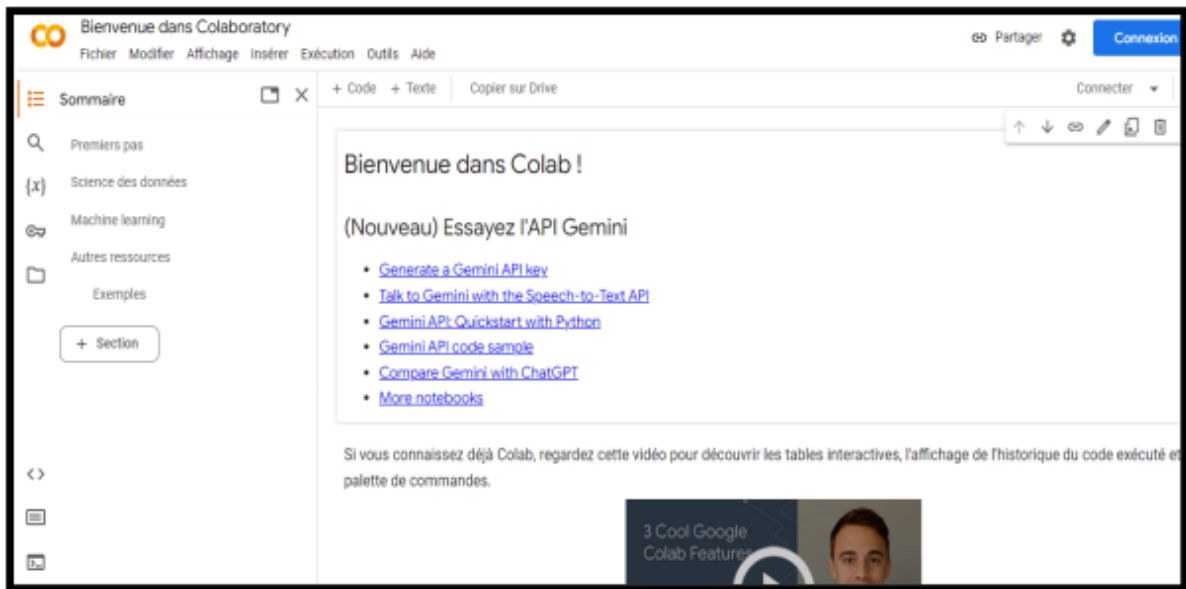


FIGURE 4.2 – Google colab

4.3.8.1 Avantages de Google Colab

En plus d'être simple à utiliser, Colab offre une grande flexibilité dans sa configuration et facilite de nombreuses tâches pour nous [92] :

- Prise en charge de Python 2.7 et Python 3.6.
- Accélération GPU gratuite.
- Bibliothèques préinstallées.
- Support des commandes Bash (langage de script).

4.3.8.2 Utilisation de Colab

- **Sélectionner l'accélérateur matériel** : Dans la barre de menu, sélectionnez "Exécution" (Runtime), puis "Modifier le type d'exécution" (Change Runtime Type) et choisissez l'option "Accélérateur matériel" (Hardware Accelerator) en mode GPU ou TPU (Tensor Processing Unit) (voir la figure 4.3).

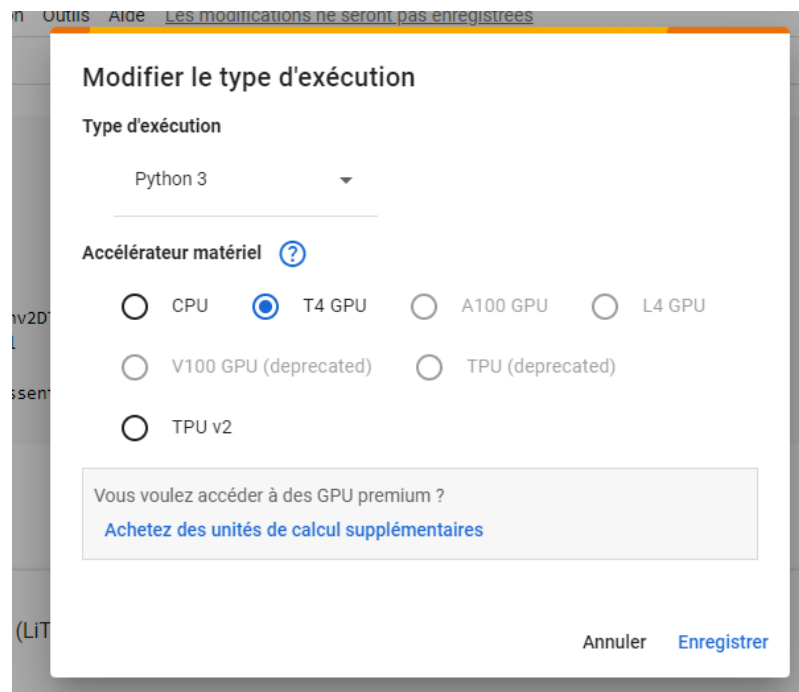


FIGURE 4.3 – choisir l'accélération matériel

- **Importer les bibliothèques Python** : les bibliothèques et les modules nécessaires pour le traitement des données et la génération des images sont :
 - `pandas` : pour le traitement des données.
 - `skimage` : pour les opérations d'image.
 - `matplotlib` : pour la visualisation des graphiques.
 - `numpy` : pour les opérations numériques.
 - `tensorflow` et `keras` : pour la création du modèle de GAN.
 - `tqdm` : pour la gestion des progressions.

```
import pandas as pd
import skimage as ski
from random import shuffle
from math import floor
import matplotlib.pyplot as plt
import numpy as np
import tensorflow as tf
from keras.layers import Layer, Conv2DTranspose, BatchNormalization, ReLU, Conv2D, LeakyReLU, Flatten, Dropout, Dense
from keras.models import Sequential
from keras.optimizers import Adam
from keras.losses import BinaryCrossentropy
from tqdm import tqdm
```

FIGURE 4.4 – Importation des bibliothèques requises

- **Importer la dataset** : importer un dataset depuis Google Drive ou le télécharger via `wget` présentent chacun des avantages distincts. Google Drive offre une simplicité d'utilisation, facilitant le partage et la gestion des permissions, tout en permettant de bénéficier des fonctionnalités de versioning et de sécurité de l'écosystème Google.

En revanche, `wget` est idéal pour des besoins d'automatisation et d'accès rapide à des fichiers, offrant une indépendance vis-à-vis des services Google et une potentielle vitesse de téléchargement accrue, bien qu'il nécessite de gérer l'hébergement et la sécurité des fichiers de manière autonome. Ainsi, le choix entre ces deux méthodes dépend principalement des besoins spécifiques en termes de collaboration, de sécurité, et de flexibilité d'accès.

```
!wget -O dataset.zip 'https://storage.googleapis.com/kaggle-data-sets/784753/1442363/bundle/archive.'
!unzip -q dataset.zip -d /content/dataset/

--2024-05-17 11:43:30-- https://storage.googleapis.com/kaggle-data-sets/784753/1442363/bundle/arch
Resolving storage.googleapis.com (storage.googleapis.com)... 173.194.196.207, 173.194.197.207, 209.
Connecting to storage.googleapis.com (storage.googleapis.com)|173.194.196.207|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3646662544 (3.4G) [application/zip]
Saving to: 'dataset.zip'

dataset.zip      100%[----->]  3.40G  180MB/s   in 23s

2024-05-17 11:43:53 (152 MB/s) - 'dataset.zip' saved [3646662544/3646662544]
```

FIGURE 4.5 – Chargement de la dataset

- **Préparation des données** : Après le téléchargement, on prépare les données en créant un dataframe `lits_df` à partir du fichier CSV `lits_df.csv`. Ce dataframe contient des informations sur les chemins des images, des masques de foie et de tumeurs, ainsi que des numéros d'étude et d'instance. puis on définit définit une fonction `trim_dir` qui retire les chemins parent pour les chemins des fichiers. Cette fonction est appliquée aux colonnes `filepath`, `liver_maskpath`, et `tumor_maskpath` du dataframe `lits_df`.

```
DATASET_PATH = '/content/dataset'
IMAGES_PATH = f'{DATASET_PATH}/dataset_6/dataset_6'
lits_df = pd.read_csv(f'{DATASET_PATH}/lits_df.csv')

def trim_dir(path):
    return path[len('../input/lits-png/dataset_6/'):]

lits_df['filepath'] = lits_df['filepath'].map(trim_dir)
lits_df['liver_maskpath'] = lits_df['liver_maskpath'].map(trim_dir)
lits_df['tumor_maskpath'] = lits_df['tumor_maskpath'].map(trim_dir)

lits_df
```

FIGURE 4.6 – Préparation de données

4.3.9 Le modèle DC-GAN

Dans cette section, nous aborderons en détail le modèle DC-GAN que nous avons développé pour la génération d'images de foie, y compris celles avec ou sans tumeurs. Le DC-GAN, est une architecture de réseau de neurones profonds spécifiquement conçue pour la génération d'images réalistes à partir de données aléatoires de bruit. Notre objectif est de fournir une compréhension approfondie de l'architecture du générateur et du discriminateur qui composent ce modèle, ainsi que des techniques d'apprentissage et des hyperparamètres utilisés pour son entraînement.

4.3.9.1 Générateur

Le générateur est responsable de produire des images de foie fictives qui peuvent être indistinguables des images réelles. Il prend un bruit aléatoire comme entrée et produit une image de foie. Le générateur est composé de plusieurs couches :

- **Conv2DTranspose** : cette couche est utilisée pour augmenter la taille de l'image en utilisant des filtres de convolution.
- **BatchNormalization** : cette couche normalise les entrées pour améliorer la stabilité du modèle.
- **ReLU** : cette couche applique une fonction d'activation de type ReLU (Rectified Linear Unit) pour introduire des non-linéarités dans le modèle.
- **Conv2D** : cette couche est utilisée pour extraire des caractéristiques de l'image en utilisant des filtres de convolution.

La figure 4.7 présente l'architecture des 6 blocs composant le générateur, les paramètres de chaque bloc sont détaillés dans la figure 4.8, tandis que la figure 4.9 offre une vue d'ensemble de l'architecture globale de notre générateur DCGAN.

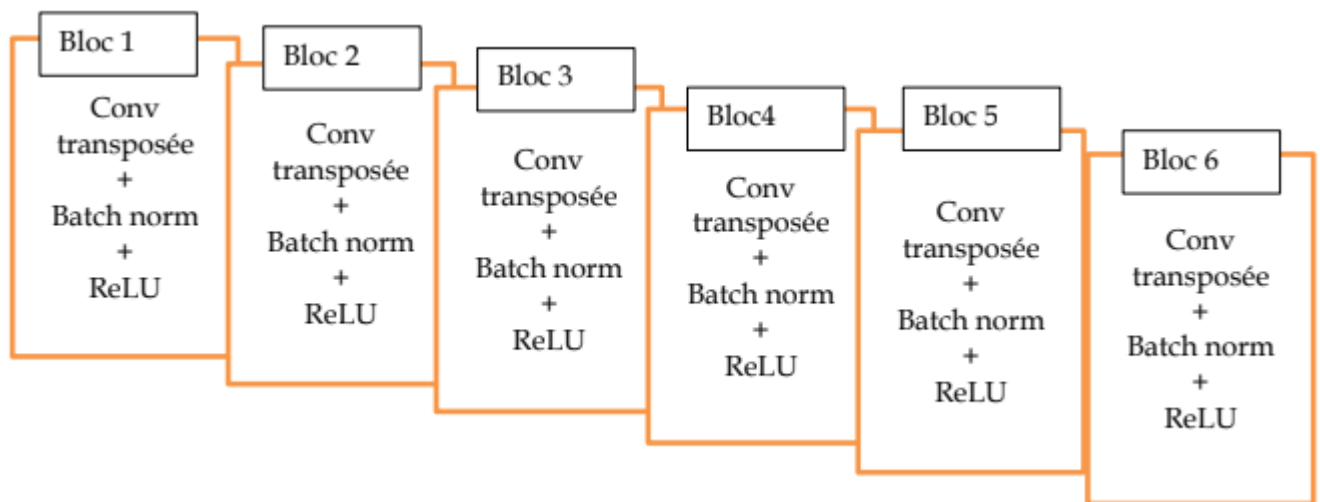


FIGURE 4.7 – Les composants des blocs transposés composant le générateur

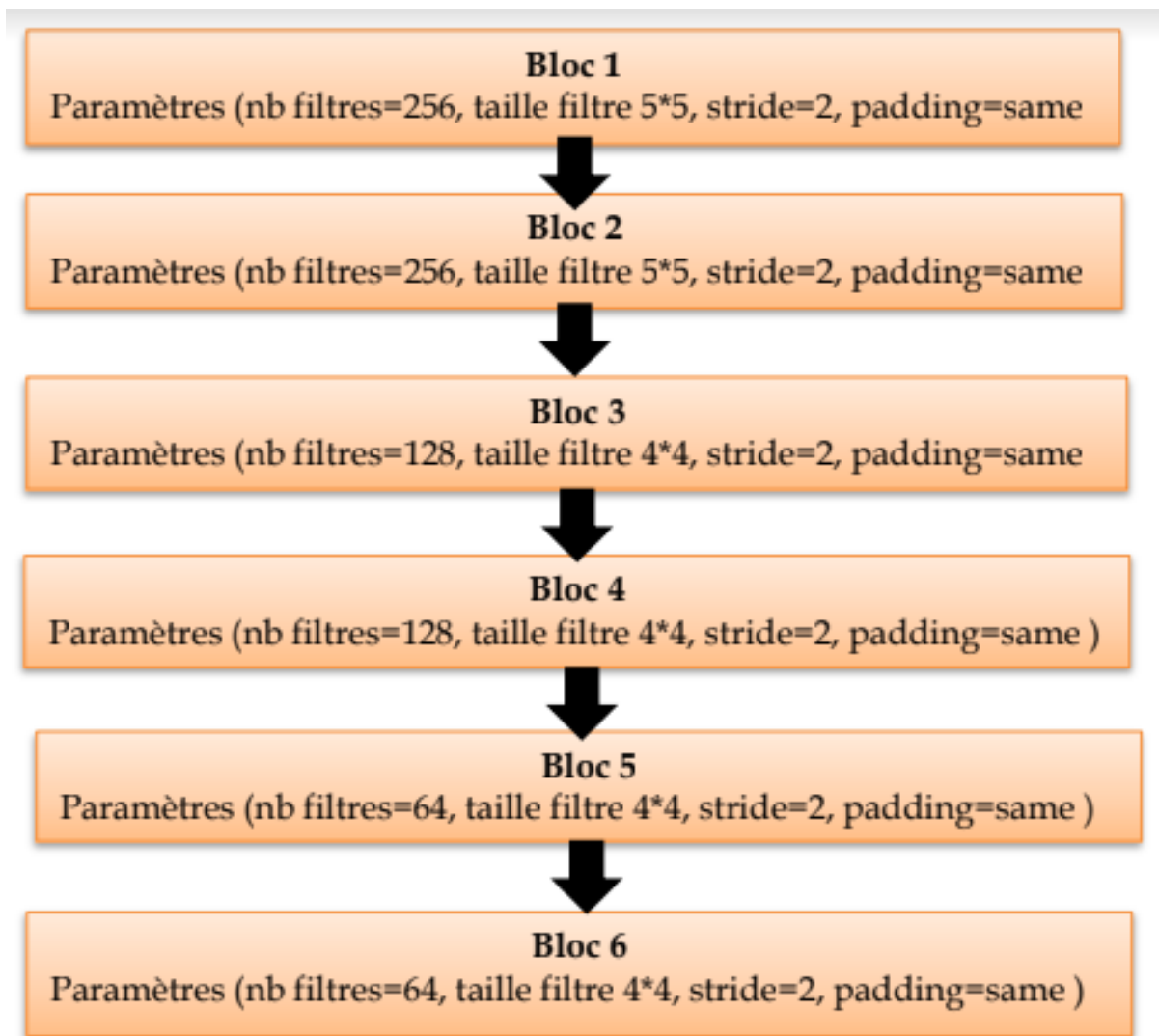


FIGURE 4.8 – Les paramètres des 6 blocs transposés composant notre générateur DCGAN

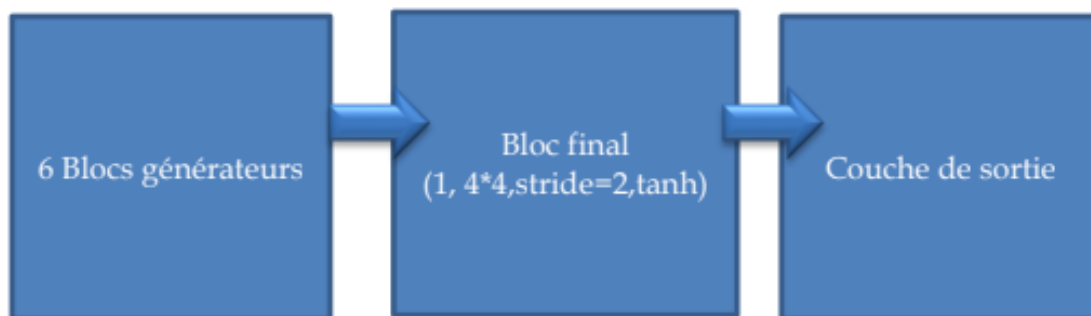


FIGURE 4.9 – Schéma général de notre générateur DCGAN

```
] class Deconv2DBlock(Layer):
    def __init__(self, filters=1, kernel_size=(4, 4), strides=(2, 2), padding='same', **kwargs):
        super().__init__(**kwargs)
        self.deconv2d = Conv2DTranspose(filters, kernel_size, strides, padding, use_bias=False)
        self.batch_norm = BatchNormalization()
        self.activation = ReLU()

    def call(self, x, training=False):
        x = self.deconv2d(x)
        x = self.batch_norm(x, training)
        x = self.activation(x)
        return x

) generator = Sequential([
    Deconv2DBlock(filters=256, kernel_size=(5, 5), strides=2, padding='same'),
    Deconv2DBlock(filters=256, kernel_size=(5, 5), strides=2, padding='same'),
    Deconv2DBlock(filters=128, kernel_size=(4, 4), strides=2, padding='same'),
    Deconv2DBlock(filters=128, kernel_size=(4, 4), strides=2, padding='same'),
    Deconv2DBlock(filters=64, kernel_size=(4, 4), strides=2, padding='same'),
    Deconv2DBlock(filters=64, kernel_size=(4, 4), strides=2, padding='same'),
    Conv2DTranspose(filters=1, kernel_size=(4, 4), strides=2, padding='same', use_bias=False, activation='tanh')
])
```

FIGURE 4.10 – Portion de code montrant une partie de notre générateur DCGAN

4.3.9.2 Discriminateur

Le discriminateur est responsable de juger si une image est réelle ou générée par le modèle. Il prend une image comme entrée et produit une probabilité que l'image soit réelle. Le discriminateur est composé de plusieurs couches :

- **Conv2D** : cette couche est utilisée pour extraire des caractéristiques de l'image en utilisant des filtres de convolution.
- **BatchNormalization** : cette couche normalise les entrées pour améliorer la stabilité du modèle.
- **LeakyReLU** : cette couche applique une fonction d'activation de type LeakyReLU (Leaky Rectified Linear Unit) pour introduire des non-linéarités dans le modèle.
- **Flatten** : cette couche flatten les données pour les préparer à l'entrée du modèle.
- **Dense** : cette couche est utilisée pour produire une sortie finale.

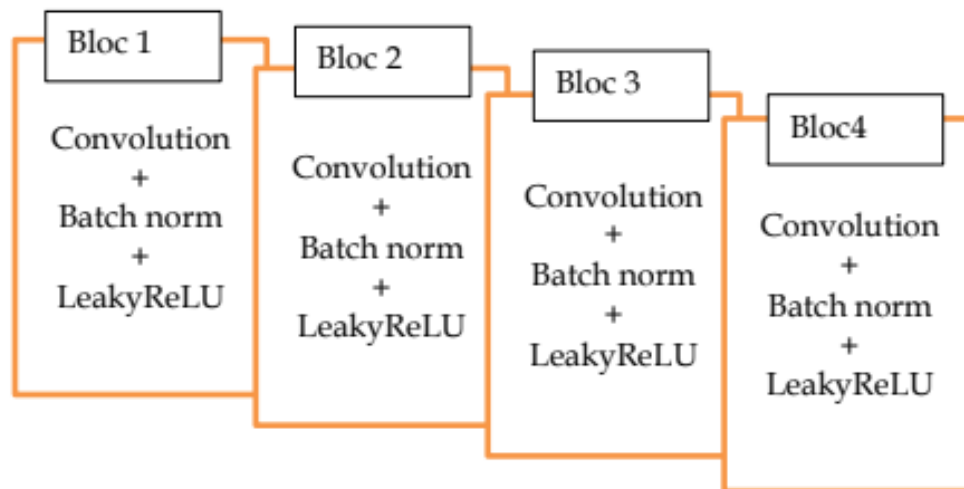


FIGURE 4.11 – Les composants des blocs de convolution composant le discriminateur

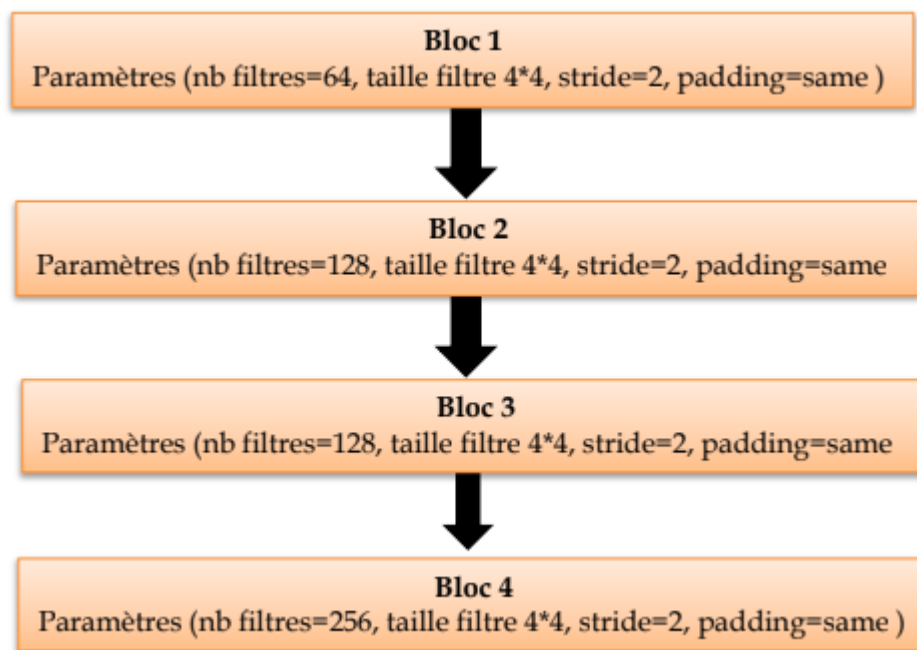


FIGURE 4.12 – Les paramètres des 4 blocs de convolution composant notre discriminateur

```

class Conv2DBlock(Layer):
| def __init__(self, filters=1, kernel_size=(4, 4), strides=(2, 2), padding='same', alpha=0.2, **kwargs):
|     super().__init__(**kwargs)
|     self.conv2d = Conv2D(filters, kernel_size, strides, padding, use_bias=False)
|     self.batch_norm = BatchNormalization()
|     self.activation = LeakyReLU(alpha)
|
| def call(self, x, training=False):
|     x = self.conv2d(x)
|     x = self.batch_norm(x, training)
|     x = self.activation(x)
|     return x

[ ] discriminator = Sequential([
|     Conv2DBlock(filters=64, kernel_size=(4, 4), strides=2, padding='same'),
|     Conv2DBlock(filters=128, kernel_size=(4, 4), strides=2, padding='same'),
|     Conv2DBlock(filters=128, kernel_size=(4, 4), strides=2, padding='same'),
|     Conv2DBlock(filters=256, kernel_size=(4, 4), strides=2, padding='same'),
|     Conv2D(filters=1, kernel_size=(4, 4), strides=1, padding='valid', use_bias=False),
|     Flatten(),
|     Dense(units=1, use_bias=False)
| ])

```

FIGURE 4.14 – Portion de code montrant une partie de notre discriminateur DCGAN

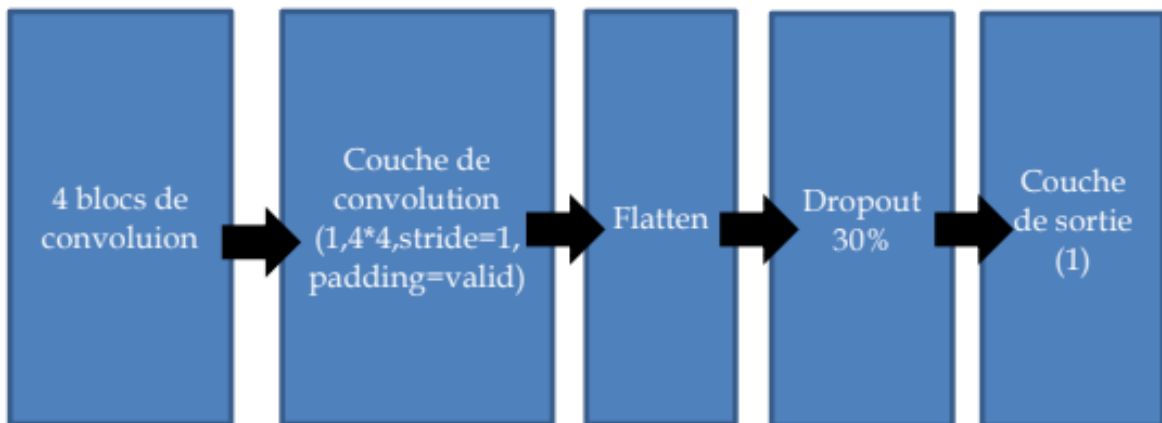


FIGURE 4.13 – Schéma général de notre discriminateur DCGAN

4.3.9.3 Entraînement du modèle

Pendant l’entraînement, le modèle utilise un algorithme de minimisation de la perte. Cette perte est calculée à l’aide de la fonction de perte Binary Cross-Entropy entre la sortie du discriminateur et l’étiquette de l’image (réelle ou générée). L’optimisation du modèle est réalisée avec l’algorithme Adam (voir la figure 4.15).

```

discriminateur_loss,] generateur_loss = [], []

for epoque in tqdm(range(EPOQUES)):
    discriminateur_batch_loss, generateur_batch_loss = [], []

    for x in dataset:
        z = tf.random.normal(mean=0, stddev=1, shape=(BATCH_TAILLE, z_h, z_w, z_d))

        discriminateur_loss = update_discriminateur(discriminateur, generateur, loss, discriminateur_optimiseur, z, x)
        generateur_loss = update_generateur(discriminateur, generateur, loss, generateur_optimiseur, z)
        discriminateur_batch_loss.append(discriminateur_loss)
        generateur_batch_loss.append(generateur_loss)

    discriminateur_epoque_loss = np.mean(discriminateur_batch_loss)
    generateur_epoque_loss = np.mean(generateur_batch_loss)
    discriminateur_loss.append(discriminateur_epoque_loss)
    generateur_loss.append(generateur_epoque_loss)

    print(f'\n----- Epoque {epoque + 1} -----')
    print(f'(discriminateur_epoque_loss=)')
    print(f'(generateur_epoque_loss=)')

generateur.save_poids('model_poids/gan_generateur.keras')
discriminateur.save_poids('model_poids/gan_discriminateur.keras')
# sauvegarder le generateur
generateur.save('/content/drive/MyDrive/CGAN_generateur.h5')
discriminateur.save('/content/drive/MyDrive/CGAN_discriminateur.h5')

plt.plot(discriminateur_loss, label='Discriminateur loss')
plt.plot(generateur_loss, label='Generateur loss')
plt.xlabel('Epoques')
plt.ylabel('Loss')
plt.title('perte entrainement GAN')
plt.legend()

```

FIGURE 4.15 – Entraînement du modèle DCGAN

4.3.9.4 Génération d'images

Une fois le modèle entraîné, il peut être utilisé pour générer des images de foie fictives. Pour cela, le modèle prend un bruit aléatoire comme entrée et produit une image de foie.

```

generated_images = (generator(tf.random.normal(mean=0, stddev=1, shape=(25, z_h, z_w, z_d), dtype=tf.float32), training=False) + 1.0) / 2.0

fig, axes = plt.subplots(nrows=5, ncols=5, figsize=(8, 8))

for i, ax in enumerate(axes.flatten()):
    ax.set_axis_off()
    ax.imshow(tf.squeeze(generated_images[i]), cmap='gray')

fig.suptitle('Generated images')
fig.tight_layout()

```

FIGURE 4.16 – Génération d'images

4.3.10 Résultats de la génération d'images

Nous présentons les résultats obtenus par notre modèle DCGAN pour générer des images de tomodensitométrie (CT scan) du cancer du foie. Le modèle a été entraîné sur un ensemble de données composé de CT scans de patients atteints de cette maladie. Après l'entraînement, le DCGAN a réussi à produire des images qui imitent de manière convaincante les radiographies réelles du cancer du foie.

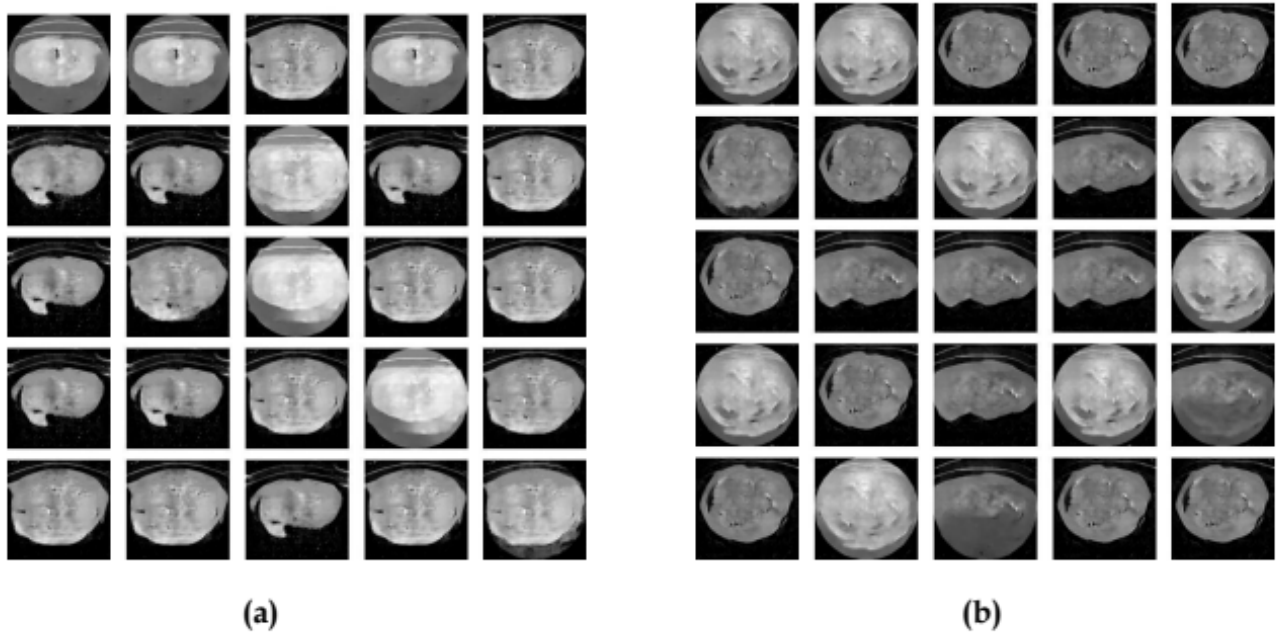


FIGURE 4.17 – Exemples d'images générées : (a) après 200 époques, et (b) après 300 époques

Les images générées présentent une variété de motifs et de caractéristiques visuelles similaires à celles des images originales. Nous remarquons que des détails tels que la texture, les contours des tumeurs, et les variations d'intensité ont été reproduits avec une grande fidélité. Cependant, En l'absence d'une évaluation quantitative, nous nous appuyons principalement sur une évaluation visuelle pour juger de la qualité des images produites.

4.4 Résultats et discussion du modèle DCGAN

La figure ci-dessous représente l'évolution de la perte (loss) lors de l'entraînement de notre DCGAN sur 200 époques. La courbe bleue représente la perte du discriminateur, tandis que la courbe orange représente la perte du générateur (voir la figure 4.18).

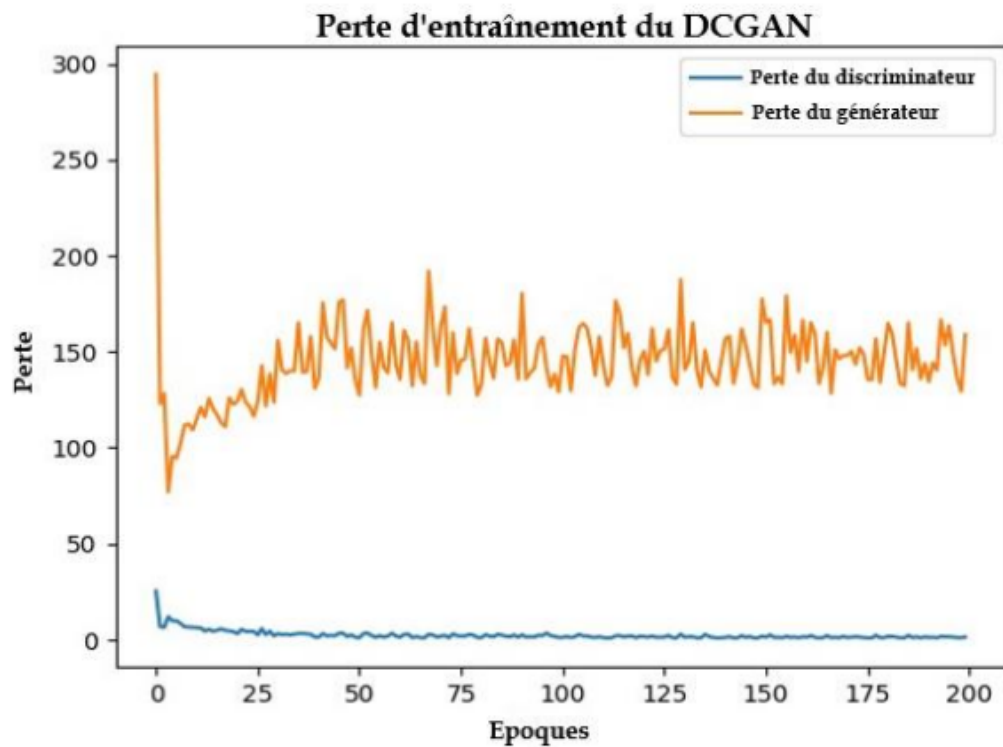


FIGURE 4.18 – La courbe de perte (loss)

On observe que la perte du discriminateur diminue rapidement au début des premières époques, pour se stabiliser ensuite autour d'une valeur proche de zéro. Cela suggère que le discriminateur devient rapidement efficace pour distinguer entre les exemples réels et générés. En revanche, la perte du générateur démarre très élevée, chute rapidement, puis fluctue de manière significative autour d'une valeur plus élevée que celle du discriminateur. Ces fluctuations indiquent que le générateur continue d'apprendre et de s'adapter en réponse aux changements dans le discriminateur, mais sans atteindre une convergence claire. L'ensemble des données suggère un entraînement dynamique avec une compétition continue entre les deux réseaux du DCGAN, ce qui est typique pour ce type de modèles. Les résultats obtenus démontrent le potentiel du modèle DCGAN pour la génération d'images CT scan du cancer du foie.

4.5 Modèle de détection

Notre modèle de détection du cancer du foie est constitué d'un générateur et de deux discriminateurs, comme montré dans la figure 4.19. L'architecture du générateur est la même que celle du générateur DCGAN précédemment décrite. Le discriminateur 1 a également la même architecture que celle du discriminateur DCGAN. L'architecture du discriminateur 2 est donnée par la figure 4.20. Le discriminateur 2 permet de distinguer entre les images qui présentent un cancer du foie et celles qui sont normales.

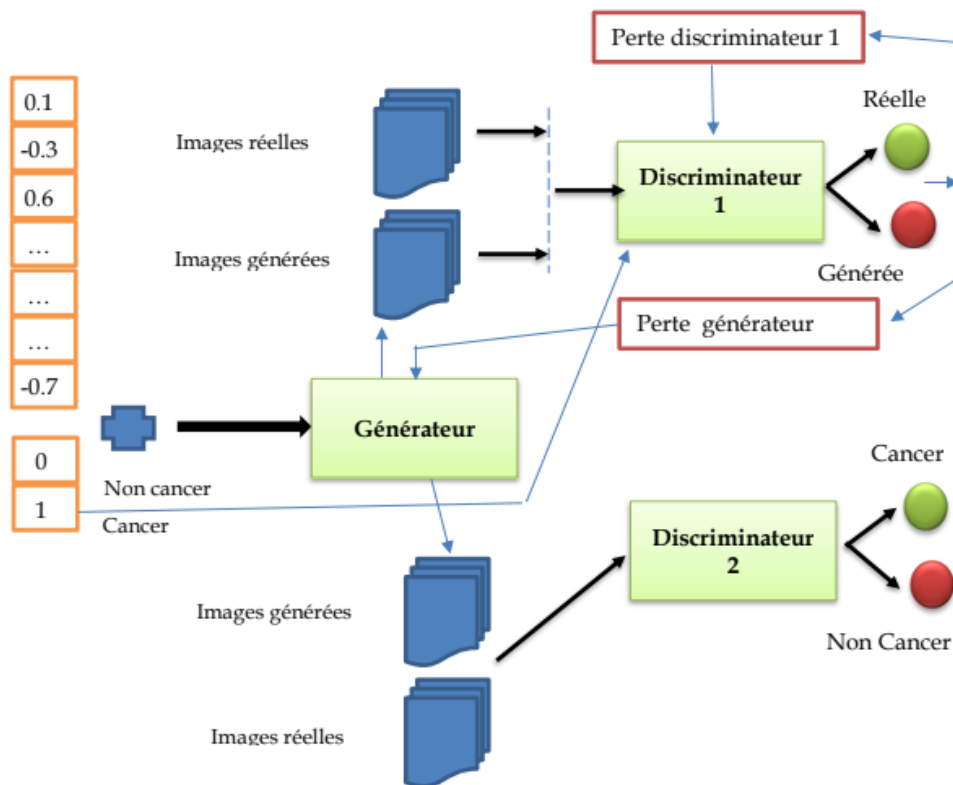


FIGURE 4.19 – Architecture du modèle de détection

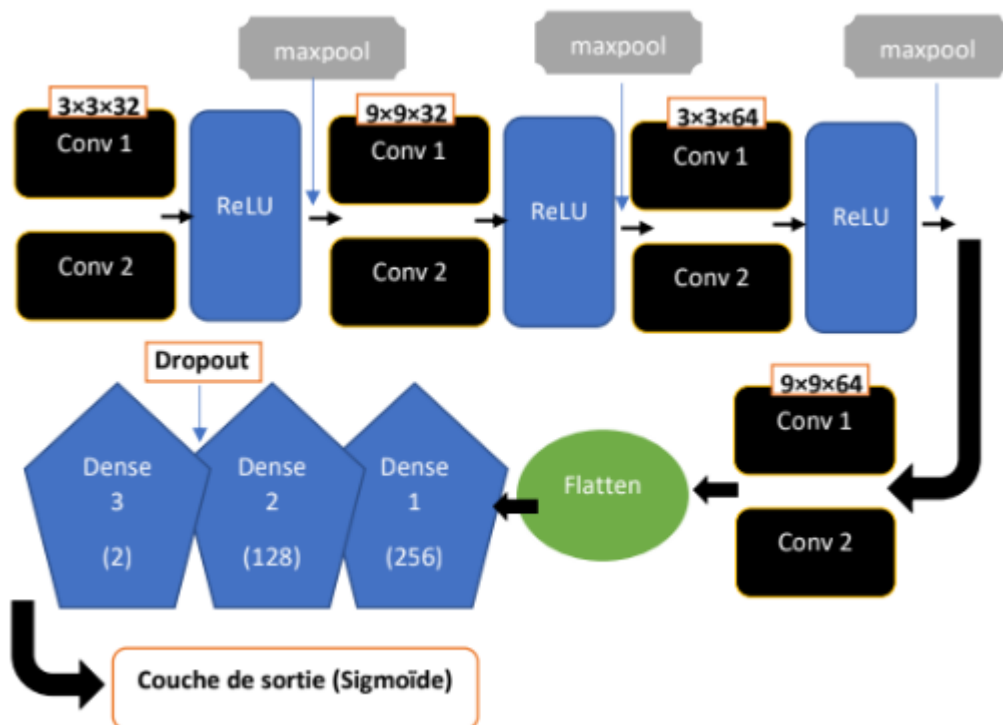


FIGURE 4.20 – Architecture de discriminateur 2

Le modèle de discriminateur 2 est composé de 4 blocs de couches de convolution où la fonction d'activation utilisée est ReLU, suivie de MaxPooling. Nous avons utilisé le padding avec les couches de convolution et gardé le stride à 1. Enfin, nous avons ajouté un réseau de neurones composé de trois couches entièrement connectées. La couche de dropout est ajoutée avec une probabilité de 0,5 et finalement une couche sigmoïde est ajoutée pour la classification finale.

4.5.1 Résultats du modèle de détection

D'après les résultats obtenus, on peut observer que la précision du modèle de détection sur les ensembles d'entraînement et de validation augmente à chaque époque, ce qui indique que le modèle apprend davantage d'informations et améliore ses performances. L'ensemble d'entraînement a atteint un taux d'accuracy de 91 % avec un taux d'erreur d'entraînement de 25 %, tandis que l'ensemble de validation a atteint un taux d'accuracy de 92 % avec un taux d'erreur de validation de 21 %. (voir la figure 4.21).



FIGURE 4.21 – Valeurs de perte et d'accuracy pour le modèle de détection

4.6 Conclusion

Les expériences et les résultats obtenus ont démontré le potentiel du modèle DCGAN pour générer des images de tomodensitométrie (CT scan) du cancer du foie, produisant des résultats visuellement convaincants. La capacité du modèle à générer des images plausibles suggère qu'il pourrait être utilisé pour des applications futures, telles que l'augmentation de données pour l'entraînement d'autres modèles d'apprentissage automatique ou pour des simulations médicales. Cependant, plusieurs limitations doivent être soulignées. Premièrement, aucune métrique de performance n'a été calculée pour évaluer objectivement la qualité des images générées. Deuxièmement, nous n'avons pas comparé notre modèle à d'autres modèles de génération d'images, ce qui limite notre capacité à situer notre travail dans le contexte des recherches existantes. Les résultats obtenus sont prometteurs, mais des recherches supplémentaires sont nécessaires pour quantifier et comparer l'efficacité de notre approche. Il serait bénéfique de développer des métriques quantitatives pour évaluer la qualité des images générées et de comparer notre modèle à d'autres approches existantes dans la littérature. Ces futures recherches permettront de mieux comprendre l'utilité et les limites du DCGAN dans le domaine de la génération d'images médicales. En ce qui concerne le modèle de détection, les résultats obtenus sont également encourageants. Le modèle a montré une augmentation constante de la précision sur les ensembles d'entraînement et de validation. Ces performances indiquent que le modèle apprend efficacement à détecter les images de cancer du foie. Bien que des améliorations et des comparaisons supplémentaires soient nécessaires, les modèles développés dans ce chapitre montrent un potentiel significatif pour des applications futures en médecine, tant pour la génération d'images que pour la détection des anomalies. Ces travaux ouvrent la voie à des avancées prometteuses dans le domaine de l'intelligence artificielle et de la médecine.

CONCLUSION GÉNÉRALE

Conclusion générale

Notre projet s'inscrit dans un contexte de recherche dynamique et en constante évolution dans le domaine de la santé, axé sur l'utilisation des réseaux antagonistes génératifs (GAN) pour la détection précoce du cancer du foie à partir d'imagerie médicale. L'imagerie médicale a considérablement amélioré la précision des diagnostics et ouvert de nouvelles perspectives pour le traitement des maladies, notamment le cancer du foie.

Le cancer du foie, souvent asymptomatique à ses débuts, constitue un défi diagnostique majeur. Notre projet vise à pallier cette lacune en exploitant les avancées technologiques dans le domaine de l'apprentissage profond, notamment les DCGAN, pour détecter des anomalies sur des images médicales. En formant notre modèle à générer des images réalistes de tissus sains et malades, nous avons pu identifier des anomalies indiquant la présence de tumeurs hépatiques, ce qui pourrait permettre une détection précoce et une prise en charge efficace de cette maladie. En combinant l'imagerie médicale de pointe avec les techniques d'apprentissage profond, notre projet ouvre de nouvelles perspectives dans la lutte contre le cancer du foie. Il illustre comment les avancées technologiques peuvent être mises au service de la santé publique, en permettant une détection plus précoce des maladies et en améliorant les résultats cliniques pour les patients.

Notre étude met en lumière le potentiel prometteur des réseaux antagonistes génératifs (GAN), en particulier des DCGAN, pour la génération et la détection d'images de cancer du foie. Elle démontre que les GAN peuvent produire des images synthétiques de haute qualité, reproduisant fidèlement les caractéristiques visuelles des CT scans réels du foie. Ces images offrent de multiples applications, notamment l'enrichissement des données pour l'entraînement des modèles d'apprentissage automatique. Bien que notre modèle de détection ait montré des résultats prometteurs dans la distinction entre les images de foie cancéreux et les images de foie sain, des recherches supplémentaires sont nécessaires pour renforcer la validité de nos conclusions. Des évaluations quantitatives et des comparaisons avec d'autres méthodes de génération et de détection d'images seraient nécessaires pour une validation plus approfondie de notre approche.

Bibliographie

- [1] What are generative adversarial networks (gans)? Retrieved from <https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/>, . Consulté le : 28-04-2024.
- [2] Rachida Bouchouareb. Réduction de bruit des taches dans l'imagerie médicale en utilisant le réseaux de neurones cellulaires (cnn). Mémoire de magister, Université de Batna, Faculté des Sciences de l'Ingénieur, 2008/2009.
- [3] Ali Khaled. Compression d'image médicale. Rapport de stage, Laboratoire L.E.R.S.S, Université Paris XII-VAL De MARNE, 2004.
- [4] L'histoire de l'imagerie médicale : 130 ans d'innovations. <https://www.radiologie-anim.fr/post/l-histoire-de-l-imagerie-medicale-130-ans-d-innovations-radiologie-nimes>, . Accessed on : 2024-05-28.
- [5] S.Chabane and N. Haddache. Segmentation d'images mammographies. Master's thesis, Mémoire de Master en vue d'obtenir le diplôme de master professionnel en administration et sécurité des réseaux, 2017.
- [6] Réda BENTATA. *Segmentation d'images tomographiques par émission de positons*. PhD thesis, Thèse de doctorat, 2012.
- [7] Mohamed Amine Belarbi and Fathi Kessair. Accès sécurisé aux images médicales, 2008-2009. Encadré par Mme Chahineze Merieme Bentaouza.
- [8] Artificial intelligence in medical imaging : Transforming healthcare diagnostics. <https://fr.shaip.com/in-the-media/artificial-intelligence-in-medical-imaging-transforming-healthcare-diagnostics/>, .
- [9] Mohamed SANDELI. *Traitement d'images par des approches bio-inspirées Application à la segmentation d'images*. PhD thesis, Université Constantine, 2014.
- [10] P. Bolon, J. M. Chassery, J. P. Cocquerez, D. Demigny, C. Graffigne, A. Montanvert, and H. Maître. *Analyse d'images : filtrage et segmentation*. InterEditions, 1995.
- [11] AMINA MADJAOUI and FADIA FARES. *Segmentation des Images par Contours Actifs : Application sur les Images Satellitaires à Haute Résolutions*. PhD thesis, Thèse de doctorat, 2012.
- [12] A. Boucher and C. Garbay. Des agents specialises pour la comprehension de sequences d'images. In *RFIA. 98*, volume 11, pages 275–284, 1998.
- [13] F. Barki, R. Bouhali, and K. Laidi. Segmentation d'image couleure par les méthodes de seuillge. 2013.
- [14] Mohammed Tahar CHIKH. *Amélioration des images par un modèle de réseau de neurones (Comparaison avec les Filtres de base)*. PhD thesis, Thèse de doctorat, 2011.

- [15] Charif HOUASSINE. *Segmentation d'images par une approche biomimétique*. PhD thesis, Thèse de doctorat, 2012.
- [16] K. Aounallah. les approches de segmentation d'image par coopération régions, contours. 2010.
- [17] L. Lechlek and M. Ghougal. la segmentation d'image médicale par la méthode d'ensemble de niveaux (level sets). Master's thesis, Thèse de Master, 2012.
- [18] A. Meziane and A. Merabet. Détection de zones d'intérêt dans des images par les réseaux de neurones convolutifs. Mémoire de master en système de télécommunication, Université des Sciences et de la Technologie Houari Boumediene, Alger, 2024. Sous la direction de Serir Amina.
- [19] J. Krapac. *Représentations d'images pour la recherche et la classification d'images*. PhD thesis, Doctoral dissertation, Caen, 2011.
- [20] Cancer du foie. <https://www.sante-sur-le-net.com/maladies/cancer/cancer-foie/>, . Accessed on : 2024-05-28.
- [21] European Society for Medical Oncology (ESMO). *Cancer du Foie : Guide pour les Patients*. Year of publication. URL <https://www.esmo.org/content/download/6613/115095/file/FR-Cancer-du-Foie-Guide-pour-les-Patients.pdf>. Accessed on : 2024-05-28.
- [22] Ministry of Health, Saudi Arabia. *Saudi Arabia National Cancer Registry : Cancer Incidence Report Saudi Arabia 2015*. 2018. URL <https://www.moh.gov.sa/en/Ministry/About/Health%20Policies/017.pdf>. Accessed on : 2024-05-28.
- [23] N. Nilsson. Principles of artificial intelligence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3 :112–112, 1980. doi : 10.1007/978-3-662-09438-9.
- [24] Tom M Mitchell. *Machine learning*, volume 1. McGraw-hill New York, 1997.
- [25] Qian Zhang, Jie Lu, and Yaochu Jin. Artificial intelligence in recommender systems. *Complex & Intelligent Systems*, 7 :439–457, 2020. doi : 10.1007/s40747-020-00212-w.
- [26] H. Salehi and R. Burgueño. Emerging artificial intelligence methods in structural engineering. *Engineering Structures*, 2018. doi : 10.1016/J.ENGSTRUCT.2018.05.084.
- [27] Fei Wang and A. Preininger. Ai in health : State of the art, challenges, and future directions. *Yearbook of Medical Informatics*, 28 :16 – 26, 2019. doi : 10.1055/s-0039-1677908.
- [28] Longbing Cao. Ai in finance : Challenges, techniques, and opportunities. *ACM Computing Surveys (CSUR)*, 55 :1 – 38, 2021. doi : 10.1145/3502289.
- [29] Konstantina D. Kourou, T. Exarchos, K. Exarchos, M. Karamouzis, and D. Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13 :8 – 17, 2014. doi : 10.1016/j.csbj.2014.11.005.
- [30] A. Osareh and B. Shadgar. Machine learning techniques to diagnose breast cancer. *2010 5th International Symposium on Health Informatics and Bioinformatics*, pages 114–120, 2010. doi : 10.1109/HIBIT.2010.5478895.

- [31] N. Ghaffari Laleh, M. Ligeró, R. Pérez-Lopez, and Jakob Nikolas Kather. Facts and hopes on the use of artificial intelligence for predictive immunotherapy biomarkers in cancer. *Clinical cancer research : an official journal of the American Association for Cancer Research*, 2022. doi : 10.1158/1078-0432.CCR-22-0390.
- [32] M. R. Alam, Kyung Jin Seo, J. Abdul-Ghafar, Kwangil Yim, Sung Hak Lee, Hyun-Jong Jang, C. Jung, and Y. Chong. Recent application of artificial intelligence on histopathologic image-based prediction of gene mutation in solid cancers. *Briefings in bioinformatics*, 2023. doi : 10.1093/bib/bbad151.
- [33] M. Kubát. An introduction to machine learning. pages 1–348, 2017. doi : 10.1007/978-3-319-63913-0.
- [34] P. E. Miller. Predictive abilities of machine learning techniques may be limited by dataset characteristics : Insights from the unos database. *Journal of cardiac failure*, 2019.
- [35] P. Jonathon Phillips, Carina Hahn, Peter Fontana, Amy Yates, Kristen K. Greene, David Broniatowski, and Mark A. Przybocki. Four principles of explainable artificial intelligence, 2021-09-29 04 :09 :00 2021. URL https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=933399.
- [36] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Comput. Surv.*, 54(6), jul 2021. ISSN 0360-0300. doi : 10.1145/3457607. URL <https://doi.org/10.1145/3457607>.
- [37] Stuart J Russell and Peter Norvig. *Artificial intelligence : a modern approach*. Pearson, 2016.
- [38] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [39] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4) :115–133, Dec 1943. ISSN 1522-9602. doi : 10.1007/BF02478259. URL <https://doi.org/10.1007/BF02478259>.
- [40] J.G. Betts. *Anatomy and Physiology*. Open Textbook Library. OpenStax College, Rice University, 2013. ISBN 9781938168130. URL <https://books.google.dz/books?id=dvVgngEACAAJ>.
- [41] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML 2010*, pages 807–814, 2010.
- [42] B.L. Kalman and S.C. Kwasny. Why tanh : choosing a sigmoidal function. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 4, pages 578–581 vol.4, 1992. doi : 10.1109/IJCNN.1992.227257.
- [43] Diederik P. Kingma and Jimmy Ba. Adam : A method for stochastic optimization. 2017.
- [44] Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/90365351ccc7437a1309dc64e4db32a3-Paper.pdf.

- [45] T. Tieleman. Lecture 6.5-rmsprop : Divide the gradient by a running average of its recent magnitude. *COURSERA : Neural Networks for Machine Learning*, 4(2) :26, 2012. URL <https://cir.nii.ac.jp/crid/1370017282431050757>.
- [46] B.T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5) :1–17, 1964. ISSN 0041-5553. doi : [https://doi.org/10.1016/0041-5553\(64\)90137-5](https://doi.org/10.1016/0041-5553(64)90137-5). URL <https://www.sciencedirect.com/science/article/pii/0041555364901375>.
- [47] Yann LeCun, Lawrence D. Jackel, Léon Bottou, A. Brunot, Corinna Cortes, John S. Denker, Harris Drucker, Isabelle M Guyon, Urs Muller, E. Sackinger, Patrice Y. Simard, and Vladimir Naumovich Vapnik. Comparison of learning algorithms for handwritten digit recognition. 1995. URL <https://api.semanticscholar.org/CorpusID:11259076>.
- [48] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet : A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi : 10.1109/CVPR.2009.5206848.
- [49] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
- [50] Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cudnn : Efficient primitives for deep learning. 2014.
- [51] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(10) :1533–1545, 2014. doi : 10.1109/TASLP.2014.2339736.
- [52] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. 2014.
- [53] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. 2017.
- [54] Wei Zhang, Jun Tanida, Kazuyoshi Itoh, and Yoshiki Ichioka. Shift-invariant pattern recognition neural network and its optical architecture. In *Proceedings of annual conference of the Japan Society of Applied Physics*, volume 564. Montreal, CA, 1988.
- [55] W Rudin. *Functional analysis* tata mcgraw, 1973.
- [56] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. A survey of convolutional neural networks : Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12) :6999–7019, 2022. doi : 10.1109/TNNLS.2021.3084827.
- [57] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv :1603.07285*, 2016.
- [58] Bilal Alsallakh, Narine Kokhlikyan, Vivek Miglani, Jun Yuan, and Orion Reblitz-Richardson. Mind the pad–cnns can develop blind spots. *arXiv preprint arXiv :2010.02178*, 2020.

- [59] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324, 1998.
- [60] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.
- [61] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [62] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [63] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net : Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015 : 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.
- [64] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. 2018.
- [65] Yaniv Taigman, Ming Yang, Marc’Aurelio Ranzato, and Lior Wolf. Deepface : Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.
- [66] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1891–1898, 2014.
- [67] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin softmax loss for convolutional neural networks. *arXiv preprint arXiv :1612.02295*, 2016.
- [68] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, 2014.
- [69] Jason Brownlee. A gentle introduction to generative adversarial networks (gans). Retrieved June 17, 2019, 2019.
- [70] Sinh-Ngoc Nguyen, Van-Quyet Nguyen, Jintae Choi, and Kyungbaek Kim. Design and implementation of intrusion detection system using convolutional neural network for dos detection. In *Proceedings of the 2nd International Conference on Machine Learning and Soft Computing*, pages 34–38, 2018.
- [71] Kwang Gi Kim. Book review : Deep learning. *Healthcare Informatics Research*, 22(4) :351–354, 2016.
- [72] Ian Goodfellow. Nips 2016 tutorial : Generative adversarial networks. *arXiv preprint arXiv :1701.00160*, 2016.

- [73] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv :1511.06434*, 2015.
- [74] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv :1511.06434*, 2015.
- [75] WandB. How to implement deep convolutional generative adversarial networks (dcgan) in tensorflow, 2023. URL <https://wandb.ai/generative-adversarial-networks/dcgan-tensorflow/reports/How-to-Implement-Deep-Convolutional-Generative-Adversarial-Networks-DCGAN-in-Tensorflow--Vml>. Consulté le : 29-05-2024.
- [76] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv :1511.06434*, 2015.
- [77] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv :1809.11096*, 2018.
- [78] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.
- [79] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, and Xiaogang Wang. Self-attention generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 735–744, 2018.
- [80] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Cgans with projection discriminator. *arXiv preprint arXiv :1802.05637*, 2018.
- [81] Shan Barratt, Rishi Sharma, Jelena Popovic, and Raha Raziperchikolaei. A note on the inception score. *arXiv preprint arXiv :1801.01973*, 2018.
- [82] Bin Wang Matthew Antalek Elif Keles Debesh Jha Amir Borhani Daniela Ladner Ulas Bagci Ugur Demir, Zheyuan Zhang. Transformer based generative adversarial network for liver segmentation. *ResearchGate*, 2022. URL https://www.researchgate.net/publication/362467336_Transformer_Based_Generative_Adversarial_Network_for_Liver_Segmentation.
- [83] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv :1511.06434*, 2015. URL <https://arxiv.org/html/1511.06434>. indico Research, Boston, MA ; Facebook AI Research, New York, NY.
- [84] Ahmed Makhlof, Marina Maayah, Nada Abughanam, and Cagatay Catal. The use of generative adversarial networks in medical image augmentation. *SpringerProfessional*, 2023. URL <https://www.springerprofessional.de/en/the-use-of-generative-adversarial-networks-in-medical-image-augm/26183352>. Received : 22 November 2022 / Accepted : 20 September 2023 / Published online : 15 October 2023.
- [85] Generative adversarial networks for anomaly detection in medical images : A comprehensive study. 2024. URL <https://www.researchgate.net/publication/377781154>. Research Proposal · January 2024, DOI : 10.13140/RG.2.2.35453.33767.

-
- [86] Python Software Foundation. The python tutorial — python 3.8.6rc1 documentation, 2024. URL <https://docs.python.org/3/tutorial/>.
- [87] TensorFlow, . URL <https://www.tensorflow.org/>.
- [88] Keras Team. Keras documentation : About keras. URL <https://keras.io/about/>.
- [89] Pandas, . URL <https://pandas.pydata.org/>.
- [90] scikit-image, . URL <https://scikit-image.org/>.
- [91] Kaggle discussion : General discussion. <https://www.kaggle.com/discussions/general/328265>, . Consulté le : 27-05-2024.
- [92] Colaboratory, . URL <https://research.google.com/colaboratory/faq.html#resource-limits>.