

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE  
LA RECHERCHE SCIENTIFIQUE

UNIVERSITE 20 Août 1955- SKIKDA

Faculté des Sciences. Département d'Informatique



Mémoire Présenté en vue de l'obtention du diplôme de  
Master en INFORMATIQUE  
Option Systèmes Informatiques.

Utilisation d'un CNN pour la reconnaissance des images :  
Application aux maladies des plantes.

Réalisé par :

Bouffénache Chahinaz  
Telilani Ahlam

Encadreur :

Mr Mansoul A.

Année universitaire 2023/2024

# REMERCIEMENTS

---

Tout d'abord on tient à remercier ALLAH le tout puissant de nous avoir donné la foi et de nous avoir permis d'en arriver là et pouvoir réaliser ce travail. En préambule à ce mémoire on souhaitait adresser nos remerciements les plus sincères aux tous les enseignants que nous avons connus pendant notre cursus. On tient à remercier sincèrement notre encadreur Mr. Mansoul Abdelhak , d'avoir accepté de nous encadrer et nous avoir guidé tout au long de notre projet de Master par ses précieuses directrices pour sa disponibilité et sa gentillesse , ainsi pour l'aide et le temps qu'elle a bien voulu nous consacrer On exprime également nos gratitude à nos examinateurs, qui vont nous honorer en évaluant ce travail.

Aussi nous devons remercier nos familles pour leur patience et leurs aides, ainsi que nos amis et nos proches sans oublier tous ceux qui ont contribué de près ou de loin, à ce projet.

---

## DÉDICACES

---

*Avec tous mes sentiments de respect, avec D'expériences de ma reconnaissance, je dédié ma remise de diplôme et ma joie.*

*Tout D'abord, à l'âme de mon chère père Nassim, j'espère qu'il apprécie cet humble geste comme preuve de reconnaissance de la part de sa fille qui a toujours prie Dieu ait son âme, tu resteras a jamais dans mon cœur.*

*À mon paradis, ma chère maman Chahra, ma raison de vivre en témoignage de ma reconnaissance, pour sa patience, son amour et ses sacrifices, que Dieu prolonge ta vie et accorde à ton cœur tout ce qu'il désire.*

*À mon frère Souhîl, mon soutien au meilleur de mes jours, et à celui qui m'a tendu la main dans ma faiblesse, je souhaite prospérité et réussite dans la vie.*

*À ma chère sœur Besma, et mes frères Aymen et Wassim qui je le sais ma réussite est important à leur yeux, que Dieu vous garde pour moi.*

*À mes meilleurs amies Maya et Rimah, votre amitié a été ma lumière et ma force, merci pour votre soutien et votre amour indéfectibles.*

*À mon binôme Chahinaz pour son soutien moral, sa patience et sa compréhension tout au long de ce projet.*

*À mes chères copines Hadjer, Ahlem, Imen, Lamis et Marwa, puissiez-vous toujours être des amies pour la vie et que Dieu vous bénisse de toute bonté.*

*À ma cousine Rawnak, et toute ma famille qui remplit ma vie depuis ma tendre enfance.*

*Pour fini à tous ceux que j'aime et qui m'aiment je dédié ce travail.*

*Ahlem*

# DÉDICACES

---

*Je me dédie cette réussite et je tiens à me remercier pour ce qu'elle est aujourd'hui.*

*À la lumière qui a illuminé mon chemin, à celui dont je porte le nom avec fierté, à mon premier maître, l'homme qui a consacré sa vie à notre réussite : mon cher père, «**Abd ElRahman**»*

*À celle qui a été mon premier soutien pour réaliser mes ambitions, à celle qui a été mon refuge et mon bras droit durant cette étape, à celle qui m'a guidé dans la vie et m'a permis de m'épanouir avec fierté, à ce cœur tendre dont les prières m'entourent toujours : ma chère mère «**Fatima** »*

*À ceux avec qui j'ai vécu les plus beaux moments de ma vie, aux lumières de mon chemin, à ceux qui ont partagé avec moi les défis des études et les nuits blanches... Mes frères, **Housseem Eddine** et **Ammar**, Mes sœurs, **Chaïma** et **Yousra**.*

*À ma belle-soeur **Amina**, merci pour votre soutien, votre amour et votre bienveillance.*

*À ma binôme **Ahlem** , pour son soutien moral, sa patience et sa compréhension tout au long de ce projet.*

*À mon cousin **DJihad**, notre rayon de soleil, et à tous les membres de ma famille qui illuminent ma vie depuis mon plus jeune âge.*

*À mes compagnons d'âme qui ont partagé chaque étape de ce chemin, à ceux qui m'ont encouragé à persévérer et à poursuivre ce parcours, à mes amis de toujours. **khaoula**, **Abir**, **Amel**, **Ranaï** , et **Aya** .*

*À ma compagne de route qui affronte la vie avec le sourire, à celle avec qui j'ai vécu un parcours universitaire formidable, à mon amie et chère **Imen**.*

*À mes amies et camarades de mon parcours académique **Lamis**, **Ahlem** et **Marwa**, merci pour ces années merveilleuses.*

*À tous ceux-là, je dédie le fruit de mon travail et ma joie.*

**Chahínaz**

---

# Résumé

---

La reconnaissance d'images est l'une des tâches les plus complexes en intelligence artificielle et en traitement d'images. Elle repose sur l'utilisation d'algorithmes et de modèles pour analyser et interpréter le contenu visuel des images. Ses applications pratiques sont nombreuses, allant de la classification d'images à la détection d'objets, en passant par la reconnaissance faciale, la reconnaissance de caractères et la vision par ordinateur.

Cela a conduit de nombreux chercheurs à explorer diverses techniques basées sur des données brutes pour la reconnaissance d'objets, en se concentrant sur les caractéristiques visuelles.

Ces applications sont très variées allant de la simple classification d'objets jusqu'à la reconnaissance d'objets, particularités ou traces qui peuvent détecter des anomalies telles que sur les radiographies en médecine qui permettent de détecter des maladies.

Dans ce contexte, nous avons adopté une approche basée sur les réseaux de neurones de convolution (CNN) pour reconnaître le contenu des images. Nous avons testé cette méthode sur un ensemble de données d'images représentant différentes catégories de maladies d'une plante à savoir la vigne qui donne le raisin. En utilisant un modèle CNN, nous avons pu identifier la maladie présente dans chaque image. Grâce à cette approche, nous avons atteint un taux de reconnaissance de 91,63 %.

**Mots clés :** plante, vigne, raisin, reconnaissance des images, CNN, réseau de neurones, convolution.

## Abstract

---

Image recognition is one of the most complex tasks in artificial intelligence and image processing. It relies on the use of algorithms and models to analyze and interpret the visual content of images. Its practical applications are numerous, ranging from image classification to object detection, facial recognition, character recognition, and computer vision.

This has led many researchers to explore various techniques based on raw data for object recognition, focusing on visual characteristics.

These applications are highly diverse, ranging from simple object classification to the recognition of objects, particularities, or traces that can detect anomalies, such as in medical radiographs that allow for disease detection.

In this context, we have adopted an approach based on convolutional neural networks (CNN) to recognize the content of images. We tested this method on a dataset of images representing different categories of diseases in a plant, namely the grapevine. Using a CNN model, we were able to identify the disease present in each image. Through this approach, we achieved a recognition rate of 91.63%.

**Keywords:** plant, grapevine, grape, image recognition, CNN, neural network, convolution.

UTILISATION D'UN CNN POUR LA RECONNAISSANCE DES IMAGES :  
APPLICATION AUX MALADIES DES PLANTES.

TABLE DES MATIÈRES

---

Remerciements et Dédicaces	
Résumé	
Table des matières	
Liste des figures	
Introduction générale	10
Chapitre 1 :Les images	12
1.1.Introduction	12
1.2.Les types des images	12
1.3. Les caractéristiques d'une image	14
1.4. Conclusion	16
Chapitre 2 :les méthodes de classification	17
2.1. Introduction	17
2.2. Les méthodes de classification	17
2.2.1. Les règles d'association	17
2.2.2. Les arbres de décision	19
2.2.3. Les réseaux de neurones	20
2.3. Conclusion	21
Chapitre 3 :Les réseaux de neurones_Convolutifs(CNN)	22
3.1. Introduction	22
3.2. Réseau de neurones convolutif (CNN)	22
3.2.1. Les différentes architectures d'un CNN	23
3.2.2. Fonctionnement d'un CNN	25
3.2.3. Architecture générale du CNN	26
3.3. Etat de l'art de la reconnaissance des images par CNN	27
3.4. Conclusion	28
Chapitre4 : Conception	29
4.1. Introduction	29
4.2. Le système de reconnaissance des images par CNN	29
4.3.Le prétraitement des images	31
4.4. L'apprentissage du modèle par CNN	31
4.5. La reconnaissance des images	32
4.6.Conclusion	32
Chapitre 5 : Implémentation	34

5.1 Introduction	34
5.2 Environnement de développement	34
5.3. Les interface du système	39
5.3.1. Le data set images	39
5.3.2. Prétraitement : L'annotation	40
5.3.3. La configuration	41
5.3.4. Le modèle	42
5.3.5. La reconnaissance des images	43
5.4. Résultats et discussions	43
5.4.1. Les métriques de performance utilisées	43
5.4.2. Résultats et Discussion	43
5.5. Conclusion	46
Conclusion générale	47
Bibliographie	48

---

# Liste des figures

---

Figure 1: image binaire [3].	13
Figure 2 : Les niveaux de gris [5].	13
Figure 3: Représentation des pixels [7].	14
Figure 4 : Schéma explicatif de la résolution de l'image [8].	15
Figure 5 : Exemple d'arborescence pour un problème à 5 classe [13].	20
Figure 6 : Les réseaux de neurones [12].	21
Figure 7 : L'architecture Le Net [14].	23
Figure 8 : L'architecture AlexNet [14].	23
Figure 9 : L'architecture ZFNet [14].	24
Figure 10 : L'architecture GoogLeNet/ Inception [14].	24
Figure 11 : L'architecture VGGNet [14].	25
Figure 12 : L'architecture ResNet [14].	25
Figure 13 : Architecture générale d'un CNN [15].	27
Figure 14 : Conception général de système.	30
Figure 15 : Représentation de data set sur le Platform kaggle	31
Figure 16 : L'environnement Google Colaboratory	35
Figure 17 : Python logo	36
Figure 18 :Tensorflow logo	36
Figure 19 : OpenCV logo	37
Figure 20 :Keras logo	38
Figure 21 :Kaggle logo	38
Figure 22 : Quelques bibliothèques dans le modèle.	38
Figure 23 : Les 4 classes dans le data set	39
Figure 24 : Création de classes (classe 'Grape Black rot')	40
Figure 25 : Représentation de data set annoté sur le Platform Roboflow	40
Figure 26 : Construction du modèle de réseau de neurones convolutif.	41
Figure 27 : Compilation do modèle	42
Figure 28 : la fenêtre principale de l'application.	42
Figure 29 : Résultats de la reconnaissance des images.	43
Figure 30 : Les résultats d'évaluation du modèle.	44
Figure 31 : La courbe d'erreur du modèle.	44
Figure 32 : La courbe de précision du modèle.	45
Figure 33 : La matrice de confusion du modèle	46

# INTRODUCTION GÉNÉRALE

---

Les réseaux de neurones convolutifs (CNN) sont devenus un outil incontournable pour la reconnaissance d'images, y compris dans le domaine de la botanique pour l'identification et la classification des plantes. Les CNN sont particulièrement bien adaptés à cette tâche grâce à leur capacité à extraire automatiquement des caractéristiques complexes des images, sans nécessiter de prétraitement manuel exhaustif.

Les CNN sont des algorithmes d'apprentissage automatique qui peuvent être entraînés pour extraire des caractéristiques discriminantes dans les images et les utiliser pour classer les plantes en fonction de certaines particularités ou pour détecter les maladies des plantes.

Pour implémenter ce système, il est important de choisir des architectures CNN appropriées.

Les modèles CNN peuvent être entraînés pour détecter des maladies spécifiques, Les résultats expérimentaux montrent que les modèles peuvent atteindre des précisions élevées, comme 96% à 99%, sur les données test, ce qui est satisfaisant pour une application de diagnostic des maladies des plantes.

Dans notre mémoire de master, nous nous sommes intéressées à la reconnaissance d'images portant d'indices de maladie. Pour ce faire, nous avons fait recours à un data set d'images de feuilles de vignes qui portent les traces de 4 maladies différentes et nous avons conçu un réseau de neurones convolutifs, pour l'apprentissage du modèle pour reconnaître le contenu de ces images. Le data set images considéré est de taille plus ou moins satisfaisante et permettant de reconnaître leur contenu. Notre mémoire est organisé comme suit :

**Le chapitre 1** : nous introduisons des notions générales de l'image en donnant leurs définitions et leurs caractéristiques.

**Le chapitre 2** : donne les différentes méthodes de classification, et un aperçu général sur les réseaux de neurones.

**Le chapitre 3** : sera dédié à la présentation des réseaux de neurones, nous nous penchons plus sur les CNN (les réseaux de neurones convolutifs). C'est ce dernier type de réseaux qui va nous servir pour la reconnaissance des images.

**Le chapitre 4** : ce chapitre est consacré à la conception proposée. Nous présentons ainsi les détails de l'architecture proposée et les techniques utilisées.

**Le chapitre 5** : consacré à l'implémentation et la réalisation de la proposition de notre système, et aussi aux expérimentations et aux résultats obtenus.

Notre mémoire se termine par une conclusion générale

# CHAPITRE 1 :

## LES IMAGES

---

### 1.1. Introduction

L'image constitue l'un des moyens les plus importants qu'utilise l'homme pour communiquer avec autrui. C'est un moyen de communication universel dont la richesse du contenu permet aux êtres humains de tout âge et de toute culture de se comprendre. C'est aussi le moyen le plus efficace pour communiquer, chacun peut analyser l'image à sa manière, pour en dégager une impression et d'en extraire des informations précises.

Dans ce chapitre, on présente quelques principaux concepts qui sont en relation avec notre sujet d'étude.

**Définition d'une image :** Une image est une représentation visuelle, voire mentale, de quelque chose (objet, être vivante concept). Elle peut être naturelle (ombre, reflet) ou artificielle (sculpture, peinture, photographie), visuelle ou non, tangible ou conceptuelle (métaphore), elle peut entretenir un rapport de ressemblance directe avec son modèle ou au contraire y être liée par un rapport plus symbolique.

Platon donne une des plus anciennes définitions de l'image : « J'appelle image d'abord les ombres ensuite les reflets qu'on voit dans les eaux, ou à la surface des corps opaques, polis et brillants et toutes les représentations de ce genre [1].

### 1.2. Les types des images

#### 1.2.1. Les images à deux niveaux (image binaire)

Une image à deux niveaux peut faire référence à une image binaire où chaque pixel ne peut prendre que deux valeurs distinctes, généralement noir et blanc. Ce type d'image est souvent utilisé pour représenter des informations de manière simple et binaire. Les images à deux niveaux sont couramment utilisées dans divers domaines tels que la reconnaissance de formes, la vision par ordinateur et le traitement d'images [2].

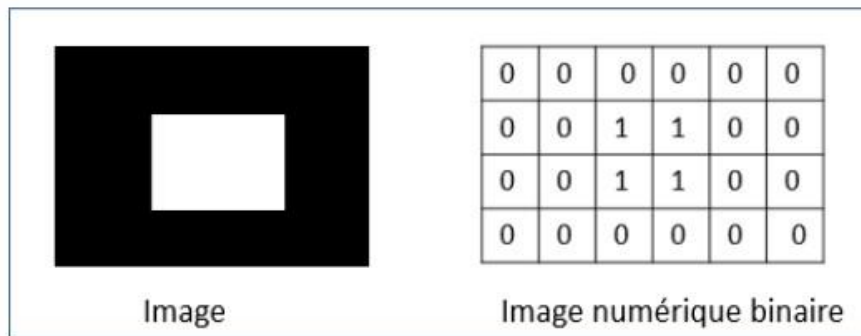


Figure 1: image binaire [3].

### 1.2.2. Les images à plusieurs niveaux de gris

En général, les images en niveaux de gris sont des images de profondeur  $k=8$  bits, donc chaque pixel peut prendre l'une des valeurs de l'intervalle  $[0..255]$ , où la valeur 0 représente la brillance minimale (le noir) et 255 la brillance maximale (le blanc). Ce type d'image est fréquemment utilisé pour reproduire des photos en noir et blanc ou du texte.

Dans plusieurs applications professionnelles de photographie et d'impression ainsi qu'en médecine et astronomie, 8 bits par pixel n'est pas suffisant, pour cela il existe d'autres types d'images en niveaux de gris de profondeur  $K=12$ ,  $K=14$  ou  $K=16$  bits.

Une image où chaque pixel est représenté par une seule valeur correspondant à l'intensité de gris. La valeur d'intensité varie de 0 (noir) au niveau maximum (blanc), avec des valeurs intermédiaires représentant différentes nuances de gris. Les images en niveaux de gris sont couramment utilisées dans diverses tâches de traitement d'images, telles que l'amélioration d'image, la segmentation et l'extraction de caractéristiques [4].

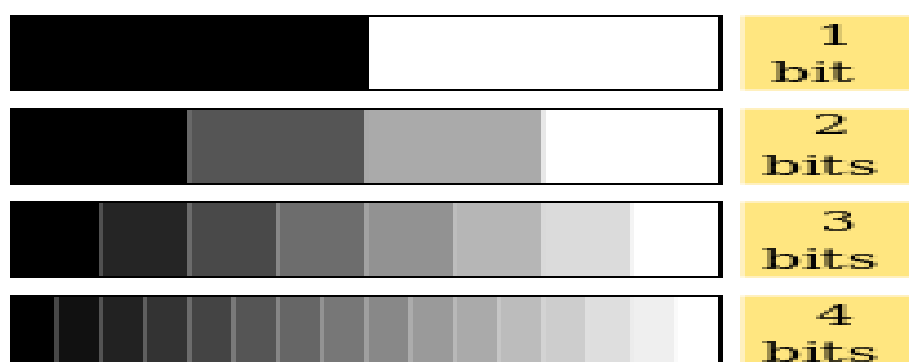


Figure 2 : Les niveaux de gris [5].

### 1.2.3. Les images couleurs

Dans image en couleurs chaque pixel possède une couleur décrite par la quantité de ces 3 composants : rouge(R), vert(V) et bleu (B), chacune de ces couleurs est codée sur l'intervalle [0, 255].) La combinaison de ces trois couleurs donne un point lumineux (un pixel) d'une certaine couleur.

Donc Le système RVB est une des façons de décrire une couleur en informatique. Par exemple: Le triplet {255, 255, 255} donnera du blanc, {255, 0, 0} un rouge pur, {100, 100, 100} un gris, etc. Le premier nombre donne la composante rouge, le deuxième la composante verte et le dernier la composante bleu [3].

### 1.3. Les caractéristiques d'une image

#### Le pixel

Un pixel, abréviation de l'anglais "picturelement" (élément d'image), est le plus petit élément d'une image numérique. Il s'agit d'un point individuel dans une grille régulière composant une image. Chaque pixel peut avoir une couleur spécifique, et la combinaison de millions de pixels différents crée l'image que vous visualisez sur un écran d'ordinateur, un smartphone, une télévision ou tout autre dispositif d'affichage numérique [6].

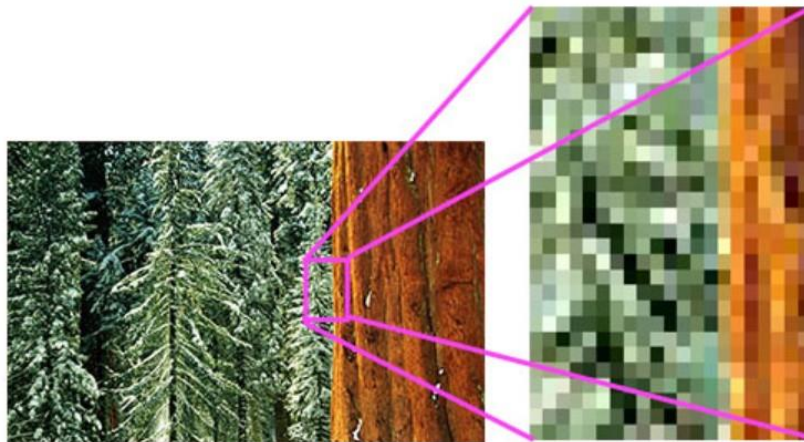


Figure 3: Représentation des pixels [7].

#### La définition d'une image

La définition est le nombre de points (ou pixels) que comporte une image numérique en largeur et en hauteur (le nombre de colonnes et nombre de lignes). On parle aussi de Taille en pixels.

Exemple : une image dont la définition est 1600x1200 correspond à une image de 1600 pixels en largeur et 1200 pixels en hauteur [7].

## La résolution

La résolution d'une image est le nombre de pixels par pouce qu'elle contient (1 pouce = 2.54 centimètres). Elle est exprimée en "PPP" (points par pouce) ou DPI (dots per inch). Plus il y a de pixels (ou points) par pouce et plus il y aura d'information dans l'image (plus précise). Par exemple, une résolution de 300dpi signifie que l'image comporte 300 pixels dans sa largeur et 300 pixels dans sa hauteur, elle est donc composée de 90 000 pixels (300x300 ppp). Grâce à cette formule, il est facile de connaître la dimension maximale d'un tirage.

Il est généralement admis qu'une résolution de 300 ppp pour une image est largement suffisante avant impression. Cette résolution peut être revue à la baisse dans le cas d'impressions devant être visualisées à une distance plus ou moins éloignée de l'observateur (par conséquent liée au pouvoir séparateur de l'œil humain [7]).

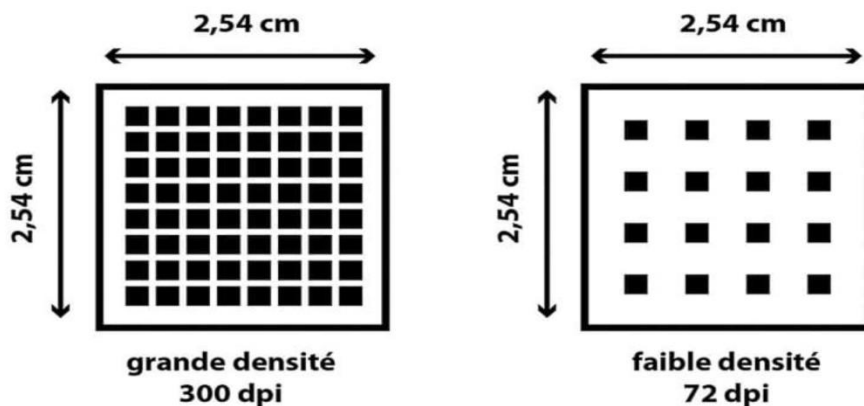


Figure 4 : Schéma explicatif de la résolution de l'image [8].

## La profondeur d'une image

La profondeur définit la quantité d'informations chromatiques disponibles pour chaque pixel dans une image. Le nombre de couleurs disponibles et la précision de la représentation des couleurs dans une image sont proportionnels au nombre de bits d'informations par pixel. Ainsi, dans une image d'une profondeur de 1 bit par pixel, les pixels peuvent prendre deux valeurs possibles : noir et blanc. Une image avec une profondeur de 8 bits par pixel compte 256 valeurs possibles (2<sup>8</sup>). Une image en mode Niveaux de gris avec une profondeur de 8 bits par pixel compte 256 valeurs possibles de gris.

Les images RVB sont constituées de 3 couches de couleur. Une image RVB de 8 bits par pixel a 256 valeurs possibles pour chaque couche, soit plus de 16 millions de valeurs chromatiques possibles. Les images RVB 8 bits par couche sont parfois appelées images 24 bits (8 bits x 3 couches = 24 bits de données pour chaque pixel [9]).

## **La taille d'une image**

La taille d'une image est le nombre de pixels de cette image, la taille d'une image qui est représenté par (328×456) dont 328 est le nombre de lignes, et 456 est le nombre de colonnes, est égale à :  $328 \times 456$ .

## **La qualité d'image**

La qualité d'image fait référence au niveau de précision avec lequel différents systèmes d'imagerie capturent, traitent, stockent, compressent, transmettent et affichent les signaux qui forment une image. Il existe deux définitions de la qualité d'image : l'une se concentre sur les caractéristiques du traitement du signal dans différents systèmes d'imagerie, tandis que l'autre se réfère à la qualité visuelle globale d'une image. La qualité d'image ne doit pas être confondue avec la fidélité d'image, qui se réfère à la capacité d'un processus à reproduire une copie de manière perpétuellement similaire à l'original. L'évaluation de la qualité d'image peut être effectuée de manière subjective par des évaluations perceptuelles humaines ou de manière objective par des modèles informatiques prédisant la qualité perceptive de l'image [9].

## **1.4. Conclusion**

Dans ce chapitre, nous avons présenté les différents concepts liés à l'image, on a parlé sur les types d'une image et les caractéristiques d'une image. Dans le chapitre suivant nous allons aborder, les différentes méthodes de prédiction, et un aperçu général sur les réseaux de neurones.

# CHAPITRE 2 :

## LES MÉTHODE DE

## CLASSIFICATION

---

### 2.1. Introduction

La classification d'images est un processus essentiel qui permet d'identifier et de catégoriser des images en utilisant différentes méthodes d'analyse et de modélisation. Utilisée dans des domaines variés tels que la sécurité, la médecine et les véhicules autonomes, cette technologie joue un rôle crucial en facilitant des décisions précises et en optimisant les opérations. Les techniques de classification varient, allant des analyses simples des caractéristiques visuelles aux modèles avancés basés sur des approches statistiques et mathématiques. En particulier, le Deep Learning, grâce à ses réseaux de neurones artificiels à plusieurs couches, révolutionne la reconnaissance d'image en automatisant l'extraction des caractéristiques et en permettant de traiter efficacement de grandes quantités de données non structurées, comme des images et des textes. Cette approche avancée capture des relations complexes dans les données, rendant les identifications plus précises et robustes pour des applications telles que la reconnaissance faciale et le traitement des images médicales.

### 2.2. Les méthodes de classification

#### 2.2.1. Les règles d'association

Une règle d'association est une application de la forme  $X \rightarrow Y$  dans laquelle  $X$  et  $Y$  sont des ensembles d'items ou Motif [10].

Une règle d'association est de la forme :

$$X \rightarrow Y,$$

**Item** est un objet, élément ou un article d'une base de données.

**Un Itemset** est un ensemble d'items, d'objets ou d'articles d'une base de données.

#### **item ou motif fréquent**

Si un motif 't' apparaît un nombre de fois et ce nombre est supérieur à un support minimal dans l'ensemble des données  $D$  alors on dit qu'il est fréquent, i.e.  $\text{support}(t) \geq \text{minsup}$  où  $\text{minsup}$  est le support minimal qui est donné par l'utilisateur [10].

Il existe deux mesures importantes, le support et la confiance, la robustesse d'une règle d'association est déterminée grâce à ces deux métriques. Une règle d'association qui a un support faible

va être observée rarement. La confiance mesure la pertinence de l'inférence dans une règle, par exemple plus grande est la mesure de confiance de la règle  $X \rightarrow Y$ , plus cette règle sera pertinente [10].

### **Item et Itemset**

**Itemset** est un ensemble d'items, d'objets ou d'articles d'une base de données.

**K-Itemset** est un ensemble de k éléments, ou k-Items, il est aussi un Itemset.

### **Support d'une règle d'association**

Le support d'une règle d'association  $A \rightarrow C$  définit le pourcentage de documents qui contiennent A et C. Il représente le nombre de documents qui contiennent A et C support  $(A \cup C)$  divisé par le nombre total des documents [11] :

$$\text{Support}(A \rightarrow C) = \frac{\text{support}(A \cup C)}{|D|} * \text{support}(A \rightarrow C) \in [0,1]$$

### **Confiance d'une règle d'association**

La confiance est une mesure permettant d'évaluer la validité d'une règle d'association. La confiance d'une règle d'association  $A \rightarrow C$  notée  $\text{conf}(A \rightarrow C)$  représente la proportion de documents qui contient A et qui contient aussi C. Elle est comme suit [10].

$$\text{conf}(A \rightarrow C) = \frac{\text{support}(A \cup C)}{\text{support}(A)} * \text{conf}(A \rightarrow C) \in [0,1]$$

#### **2.2.1.1. Extraction des règles d'association**

Avant tous on doit générer tous les motifs possibles à l'aide des termes d'une collection de documents textuels. Puis, calculer le support de chaque motif en parcourant la collection pour trouver le nombre de documents qui le contient, comparer la valeur trouvée enfin ne garder que les motifs dont le support est supérieur au support minimal. Elle permet d'éliminer les motifs non fréquents qui sont sans intérêt pour générer les règles d'association. Les résultats de cette étape sont utilisés dans l'étape suivante, donc elle est primordiale.

A partir de l'ensemble des itemsets fréquents pour un seuil minimal de support  $\text{minsup}$ , la génération des règles d'association est un problème qui dépend exponentiellement de la taille de l'ensemble des itemsets fréquents [10].

### **Algorithmes d'extraction des règles d'association**

Il existe plusieurs façons d'explorer les règles d'association, l'une de ces méthodes est la méthode naïve, on utilise alors toutes les combinaisons possibles des attributs et de leurs valeurs pour créer toutes les règles d'association possibles. Ce qui pose problème sur le plan complexité computationnelle du fait de l'explosion combinatoire. En effet, le nombre de règles générées est

énorme. On peut optimiser cette méthode en gardant juste les règles avec un support et une confiance minimum. Cela reste insuffisant et les résultats sont insatisfaisants.

L'algorithme Apriori représente une approche révolutionnaire dans l'apprentissage et l'exploration des règles d'association [10].

### **Algorithme Apriori**

L'algorithme Apriori créé par Agrawal et Srikant en 1994, procède en deux temps. Il est basé sur le principe lié à l'approche de support et de confiance. L'algorithme parcourt le treillis des itemsets pour rechercher les itemsets fréquents et en déduire les règles d'association dont la confiance dépasse le seuil de confiance minconf. Le treillis des itemsets permet d'utiliser plus efficacement cet algorithme d'extraction en admettant les propriétés suivantes :

- Tout sous-ensemble d'un itemset fréquent est fréquent.
- Tout sur-ensemble d'un itemset non fréquent est non fréquent.

Le nombre d'itemsets fréquents qui peuvent être générés de  $n$  items est de  $2^n - 1$ . La génération des itemsets fréquents est de complexité exponentielle, il est alors essentiel de trouver la méthode de recherche la plus optimale [11].

L'algorithme Apriori se fait comme suit :

- Générer les Règles candidates.
- Calculer le support pour chaque règle candidate.
- Apparier les règles dont on a calculé le support avec le support choisi.
- On rejette les candidats dont le support est inférieur au support minimale.
- On termine en sortie avec toutes les règles dont le support est supérieur au support minimal [11].

### **2.2.2. Les arbres de décision**

Un arbre de décision est un schéma représentant les résultats possibles d'une série de choix interconnectés. Il permet à une personne ou une organisation d'évaluer différentes actions possibles en fonction de leur coût, leur probabilité et leurs bénéfices. Il peut être utilisé pour alimenter une discussion informelle ou pour générer un algorithme qui détermine le meilleur choix de façon mathématique.

Un arbre de décision commence généralement par un nœud d'où découlent plusieurs résultats possibles. Chacun de ces résultats mène à d'autres nœuds, d'où émanent d'autres possibilités. Le schéma ainsi obtenu rappelle la forme d'un arbre.

Il existe trois types de nœuds différents : des nœuds de hasard, des nœuds de décision et des nœuds terminaux. Un nœud de hasard, représenté par un cercle, montre les probabilités de certains

résultats. Un nœud de décision, représenté par un carré, illustre une décision à prendre, et un nœud terminal le résultat final d'un chemin de décision [13].

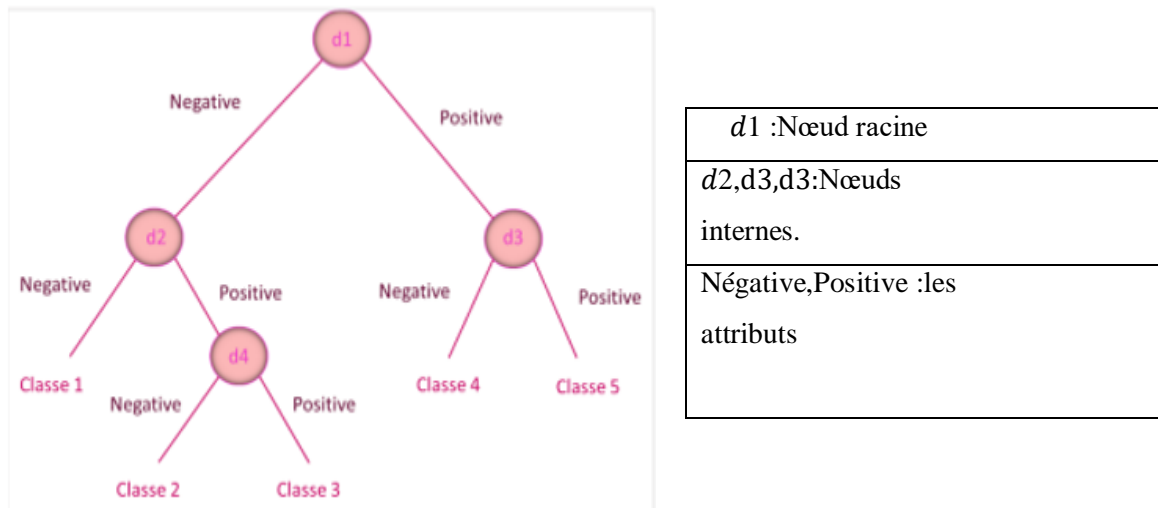


Figure 5 : Exemple d'arborescence pour un problème à 5 classe [13].

### 2.2.3. Les réseaux de neurones

Les réseaux neuronaux, également appelés réseaux de neurones artificiels, sont des modèles d'apprentissage automatique inspirés par le fonctionnement du cerveau humain. Ils sont largement utilisés dans divers domaines, y compris la vision par ordinateur, la reconnaissance vocale, la traduction automatique et bien d'autres applications.

Un réseau neuronal est composé de plusieurs couches de neurones artificiels interconnectés. Chaque neurone reçoit des entrées pondérées, les combine à l'aide d'une fonction d'activation et produit une sortie. Les couches intermédiaires entre l'entrée et la sortie sont appelées couches cachées, et elles permettent au réseau d'apprendre des représentations hiérarchiques et complexes des données

L'apprentissage dans un réseau neuronal se fait généralement en deux phases principales : la phase d'apprentissage et la phase de prédiction. Pendant la phase d'apprentissage, le réseau est entraîné sur un ensemble de données d'entraînement étiquetées, où les poids des connexions entre les neurones sont ajustés afin de minimiser une fonction de perte ou d'erreur. Ce processus d'ajustement des poids est généralement réalisé à l'aide d'algorithmes d'optimisation tels que la rétro propagation du gradient.

Une fois que le réseau est entraîné, il peut être utilisé pour faire des prédictions sur de nouvelles données non étiquetées pendant la phase de prédiction. Les entrées sont propagées à travers le réseau, et les sorties produites par le réseau fournissent les prédictions pour les différentes classes ou catégories [12].

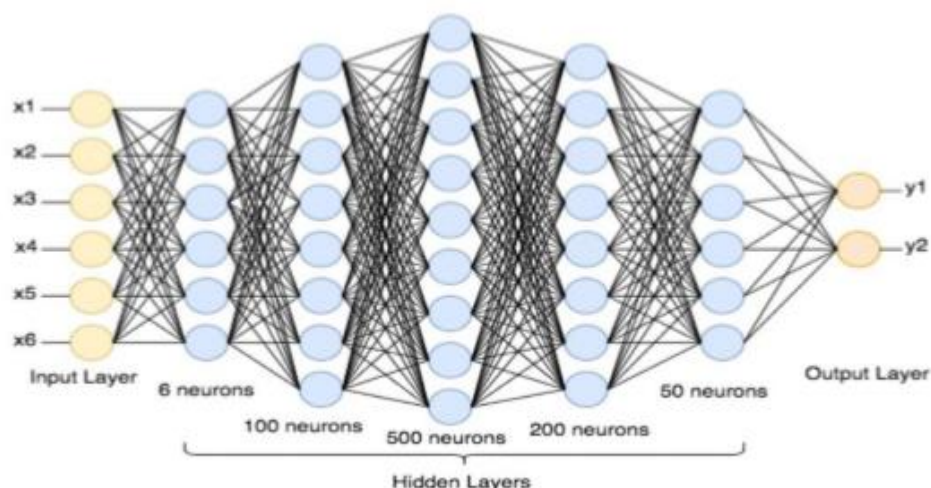


Figure 6 : Les réseaux de neurones [12].

### 2.3. Conclusion

En conclusion de ce chapitre sur les méthodes de prédiction en informatique, nous avons exploré divers outils et techniques essentiels pour l'analyse et l'anticipation des données. Les règles d'association permettent de découvrir des relations cachées entre les variables, facilitant ainsi la compréhension des motifs récurrents dans les bases de données. Les arbres de décision offrent une approche intuitive et visuelle pour la prise de décision, en structurant les choix sous forme de branches hiérarchiques basées sur les caractéristiques des données.

Chacune de ces méthodes possède ses propres avantages et limitations, et le choix de la méthode appropriée dépend souvent du contexte spécifique et des caractéristiques des données. En combinant ces techniques, il est possible de créer des modèles prédictifs robustes et efficaces, adaptés à une grande variété de problèmes pratiques. Ainsi, ce chapitre a mis en lumière l'importance de maîtriser ces outils pour améliorer la précision et la fiabilité des prédictions informatiques, ouvrant la voie à des applications toujours plus innovantes et pertinentes.

# CHAPITRE 3 :

## LES RÉSEAUX DE NEURONES

### CONVOLUTIFS( (CNN)

---

#### 3.1. Introduction

Le Deep Learning, ou apprentissage profond en français, est une branche de l'intelligence artificielle qui vise à créer des modèles d'apprentissage automatique capables d'apprendre des représentations hiérarchiques et complexes des données. Cette approche repose sur l'utilisation de réseaux neuronaux profonds, qui sont des modèles constitués de multiples couches de neurones interconnectés.

Dans ce chapitre, nous présentons les réseaux neuronaux, également appelés réseaux de neurones artificiels, qui sont des modèles d'apprentissage automatique inspirés du fonctionnement du cerveau humain. Ces modèles ont révolutionné divers domaines de l'intelligence artificielle, permettant des progrès significatifs dans des tâches comme la reconnaissance d'images, la traduction automatique, le traitement du langage naturel, et bien d'autres encore.

Le principe fondamental des réseaux neuronaux est de simuler le comportement des neurones biologiques et de leurs connexions synaptiques. Un réseau neuronal est composé de plusieurs couches de neurones interconnectés, chaque neurone agissant comme un élément de traitement de l'information.

#### 3.2. Réseau de neurones convolutif (CNN)

Le nom '*Réseau de neurones à convolution*' indique que le réseau emploie une opération mathématique appelée la convolution. Les réseaux de convolution sont un type spécialisé de réseaux neuronaux qui utilisent la convolution à la place de la multiplication matricielle générale dans au moins une de leurs couches. Les CNN sont l'un des meilleurs algorithmes d'apprentissage pour faire l'opération de convolution qui aide à l'extraction de fonctionnalités utiles à partir de points de données corrélés localement.

La sortie des noyaux convolutifs est ensuite affectée à l'unité de traitement non linéaire (fonction d'activation), qui non seulement aide à apprendre les abstractions, mais intègre également la non-linéarité dans l'espace des fonctionnalités. Cette non-linéarité génère différents modèles

d'activations pour différentes réponses et facilite ainsi l'apprentissage des différences sémantiques dans les images [12].

### 3.2.1. Les différentes architectures d'un CNN

Il existe différentes architectures de réseaux neuronaux convolutifs (CNN) qui ont été développées pour répondre à des problèmes spécifiques de vision par ordinateur. Voici quelques-unes des architectures les plus couramment utilisées :

- **LeNet (1990)**

Les premières applications réussies des réseaux convolutifs ont été développées par Yann Le Cun dans les années 1990. Parmi ceux-ci, le plus connu est l'architecture Le Net utilisée pour lire les codes postaux, les chiffres, etc... [14].

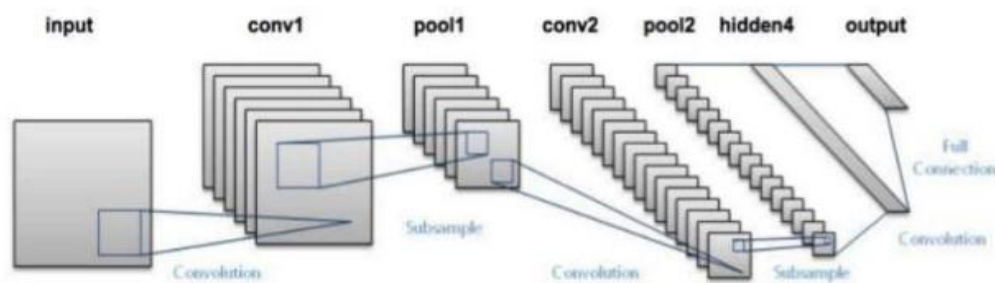


Figure 7 : L'architecture Le Net [14].

- **AlexNet(2012)**

Le premier travail qui a popularisé les réseaux convolutifs dans la vision par ordinateur était Alex Net, développé par Alex Krizhevsky, Ilya Sutskever et Geoff Hinton. Ce CNN a été soumis au défi de la base Image Net en 2012 et a nettement surpassé ses concurrents. Le réseau avait une architecture très similaire à Le Net, mais était plus profond, plus grand et comportait des couches convolutives empilées les unes sur les autres (auparavant, il était commun de ne disposer que d'une seule couche convolutifs toujours immédiatement suivie d'une couche de pooling) [14].

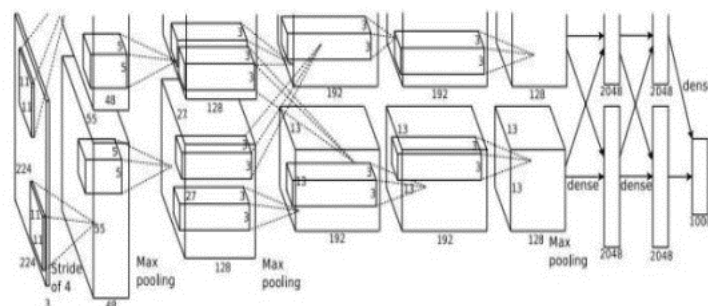


Figure 8 : L'architecture AlexNet [14].

- **ZFNet (2013)**

C'était une amélioration d'AlexNet en ajustant les hyper-paramètres de l'architecture, en particulier en élargissant la taille des couches convolutifs et en réduisant la taille du noyau sur la première couche [15].

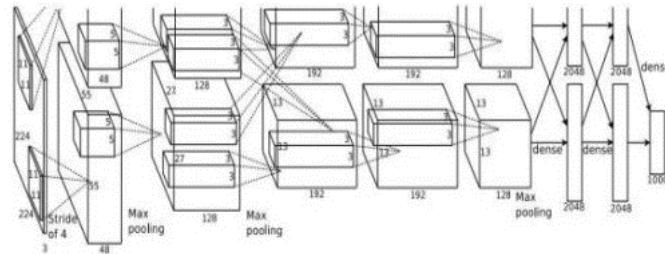


Figure 9 : L'architecture ZFNet [14].

- **GoogLeNet (2014)**

C'est un modèle de Google. Sa principale contribution a été le développement d'un module inception qui a considérablement réduit le nombre de paramètres dans le réseau (4M, par rapport à Alex Net avec 60M). En outre, ce module utilise le global Average pooling ce qui élimine une grande quantité de paramètres. Il existe également plusieurs versions de GoogLeNet, parmi elles, Inception-v4 et X Inception ce dernier est l'un des modèles lesquels notre système s'inspire, plus de détails dans le chapitre de conception [14].

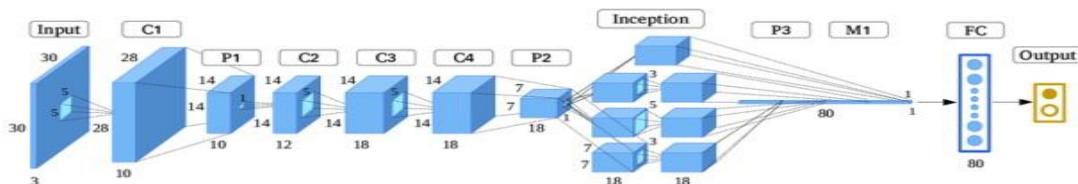


Figure 10 : L'architecture GoogLeNet/ Inception [14].

- **VGGNet (2014)**

Il s'agit d'une structure du Visual Geometry Group d'Oxford réalisée par Andrea Vedaldi et Andrew Zisserman, Sa principale contribution a été de montrer que la profondeur du réseau est un composant essentiel pour de bonnes performances. Son inconvénient est qu'il est plus coûteux à évaluer et utilise beaucoup plus de mémoire et de paramètres (140M) [14].



### 3.2.3. Architecture générale du CNN

L'architecture générale d'un réseau neuronal convolutif (CNN) comprend plusieurs couches qui sont conçues pour extraire et représenter les caractéristiques des images de manière progressive. Voici une description des principales couches qui composent l'architecture d'un CNN :

- **Couche d'entrée (Input Layer)** : Cette couche reçoit les images en tant que données d'entrée. Les images sont généralement représentées sous forme de tenseurs multidimensionnels, où chaque dimension correspond à une dimension spatiale de l'image (largeur, hauteur) et au nombre de canaux (par exemple, les canaux RVB pour les images en couleur) [15].

- **Couches de convolution (Convolutional Layers)**: Les couches de convolution sont responsables de l'extraction des caractéristiques à partir des images. Chaque couche de convolution applique un ensemble de filtres (noyaux de convolution) à l'image d'entrée, produisant des cartes d'activation qui mettent en évidence les motifs et les structures présents dans l'image. Ces filtres sont appris par le réseau pendant la phase d'entraînement [15].

- **Couches de pooling (Pooling Layers)** : Les couches de pooling réduisent la dimension spatiale des cartes d'activation produites par les couches de convolution. Elles regroupent les activations voisines en calculant, par exemple, la valeur maximale (max pooling) ou la valeur moyenne (average pooling) dans chaque région. Cela permet de réduire la complexité du modèle et d'obtenir une représentation plus compacte des caractéristiques [15].

- **Couches de non-linéarité (Activation Layers)** : Après chaque couche de convolution ou de pooling, une fonction d'activation non linéaire est appliquée aux cartes d'activation. La fonction d'activation la plus couramment utilisée est la fonction ReLU (Rectified Linear Unit), qui remplace les valeurs négatives par zéro et garde les valeurs positives inchangées. Cela introduit de la non-linéarité dans le modèle, ce qui lui permet de capturer des relations complexes entre les caractéristiques [16].

- **Couches entièrement connectées (Fully Connected Layers)**: Les couches entièrement connectées sont situées à la fin du CNN. Elles prennent les caractéristiques extraites par les couches précédentes et les transforment en une représentation finale pour effectuer la classification ou d'autres tâches. Les neurones de ces couches sont connectés à tous les neurones de la couche précédente, ce qui permet au réseau d'apprendre des combinaisons complexes de caractéristiques. [15]

- **Couche de sortie (Output Layer)** : La couche de sortie est la dernière couche du CNN. Elle produit les prédictions finales du modèle, généralement sous la forme de probabilités pour chaque classe ou d'un vecteur de sortie [15].

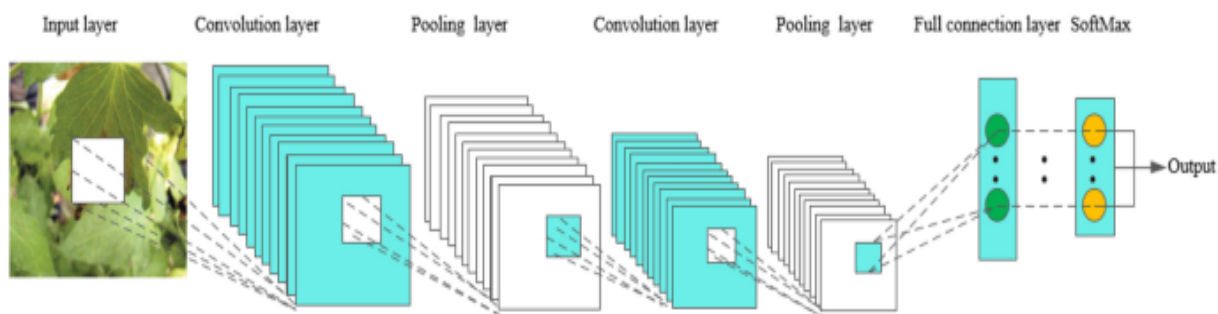


Figure 13 : Architecture générale d'un CNN [15].

### 3.3. Etat de l'art de la reconnaissance des images par CNN

- Un certain nombre d'industries utilisent l'inspection humaine pour classer visuellement la qualité de leurs produits. Cette étude compare l'utilisation de l'apprentissage profond, en particulier des réseaux de neurones convolutifs (CNN), avec des techniques traditionnelles pour la classification de la qualité des planches de bois basée sur leurs images. Bien que les techniques de Deep Learning aient généralement montré des performances supérieures dans des scénarios complexes, les résultats empiriques ont révélé que la méthode de descripteur de texture proposée était très compétitive par rapport au CNN pour toutes les expériences réalisées. L'étude souligne l'importance de l'équilibre entre précision et coût de calcul, et suggère des perspectives futures impliquant l'apprentissage actif et semi-supervisé [17].
- Une nouvelle méthode utilisant des réseaux de neurones convolutifs profonds pour la reconnaissance des maladies des plantes à partir d'images de feuilles a été développée, atteignant une précision moyenne de 96,3 %. Caffe a été utilisé pour la formation, et le modèle distingue 13 types de maladies et les feuilles saines de leur environnement [18].
- La reconnaissance des espèces végétales en utilisant deux approches distinctes : la méthode traditionnelle et l'apprentissage en profondeur. La méthode traditionnelle se concentre sur l'extraction de caractéristiques telles que les moments Hu, la texture Haralick, le motif binaire local (LBP) et les statistiques des canaux de couleur, combinées avec différents classificateurs. Pour l'ensemble de données Leaf12, cette méthode a atteint une précision de 82,38 % avec Random Forest. En comparaison, l'utilisation des architectures de réseaux neuronaux convolutifs (CNN) comme VGG 16 et VGG 19 a produit des précisions plus élevées, allant jusqu'à 99,41 % pour l'ensemble de données Swedish leaf, démontrant ainsi la supériorité des CNN dans la reconnaissance des espèces végétales face aux méthodes traditionnelles [19].

### 3.4. Conclusion

En conclusion, l'utilisation du CNN en traitement d'images représente une approche puissante et innovante pour prédire les maladies des plantes. Grâce aux CNN et à d'autres techniques de deep learning, nous sommes en mesure d'extraire des informations précieuses à partir d'images botaniques ; Dans le chapitre suivant, nous allons tenter de mettre en œuvre la solution en utilisant plusieurs méthodes et discuterons des différents avantages et inconvénients de chacune d'entre elles.

# CHAPITRE 4 :

## CONCEPTION

---

### 4.1. Introduction

Dans ce chapitre, nous plongerons dans la conception d'un système de prédiction des maladies des plants de raisin en utilisant des réseaux de neurones convolutifs (CNN). Nous allons détailler comment les CNN peuvent être formés pour reconnaître et classer les maladies des plants de raisin à partir d'images. Nous explorerons également les différentes étapes, de la formation du modèle à l'étape de prédiction, démontrant comment ce système peut détecter efficacement les maladies et améliorer la gestion des cultures.

Cette approche innovante capitalise sur les avancées de l'apprentissage profond pour offrir une solution précise et automatisée à un problème crucial dans l'agriculture moderne : la détection précoce des maladies des plants de raisin. En intégrant les capacités des CNN, notre système vise à révolutionner la surveillance des cultures, permettant aux agriculteurs de prendre des mesures préventives rapides et précises pour protéger leurs récoltes.

### 4.2. Le système de reconnaissance des images par CNN

Notre modèle est basé sur l'architecture séquentielle. Il convient de noter que, bien que les modèles séquentiels de CNN soient linéaires dans leur structure, ils peuvent être suffisamment profonds et complexes pour capturer des motifs visuels et des hiérarchies de caractéristiques de plus en plus abstraites. En ajustant l'architecture, la taille des filtres, le nombre de couches et d'autres hyperparamètres, il est possible de construire des modèles séquentiels de CNN puissants pour la reconnaissance.

- Accédez à l'ensemble de données d'images sur les nouvelles maladies végétales de Kaggle.
- Annotez manuellement les images en les classant en différentes classes (Grape-Esca, Grape\_Leaf\_blight, Grape\_healthy, blackrot) et en divisant l'ensemble de données en 70 % pour la formation, 20 % pour la validation et 10 % pour les tests.
- Un modèle de réseau neuronal convolutif (CNN) est construit à partir des données d'entraînement, puis nous assemblons le modèle en définissant la fonction de perte, l'optimiseur et les métriques d'évaluation.
- Entraînez un modèle CNN sur les données d'entraînement, puis évaluez ses performances sur les données de validation et de test.

- Si les performances sont satisfaisantes, le modèle entraîné est sauvegardé pour une utilisation future dans la prédiction des classes d'images (reconnaissance des maladies des plantes, des saisons, etc.).

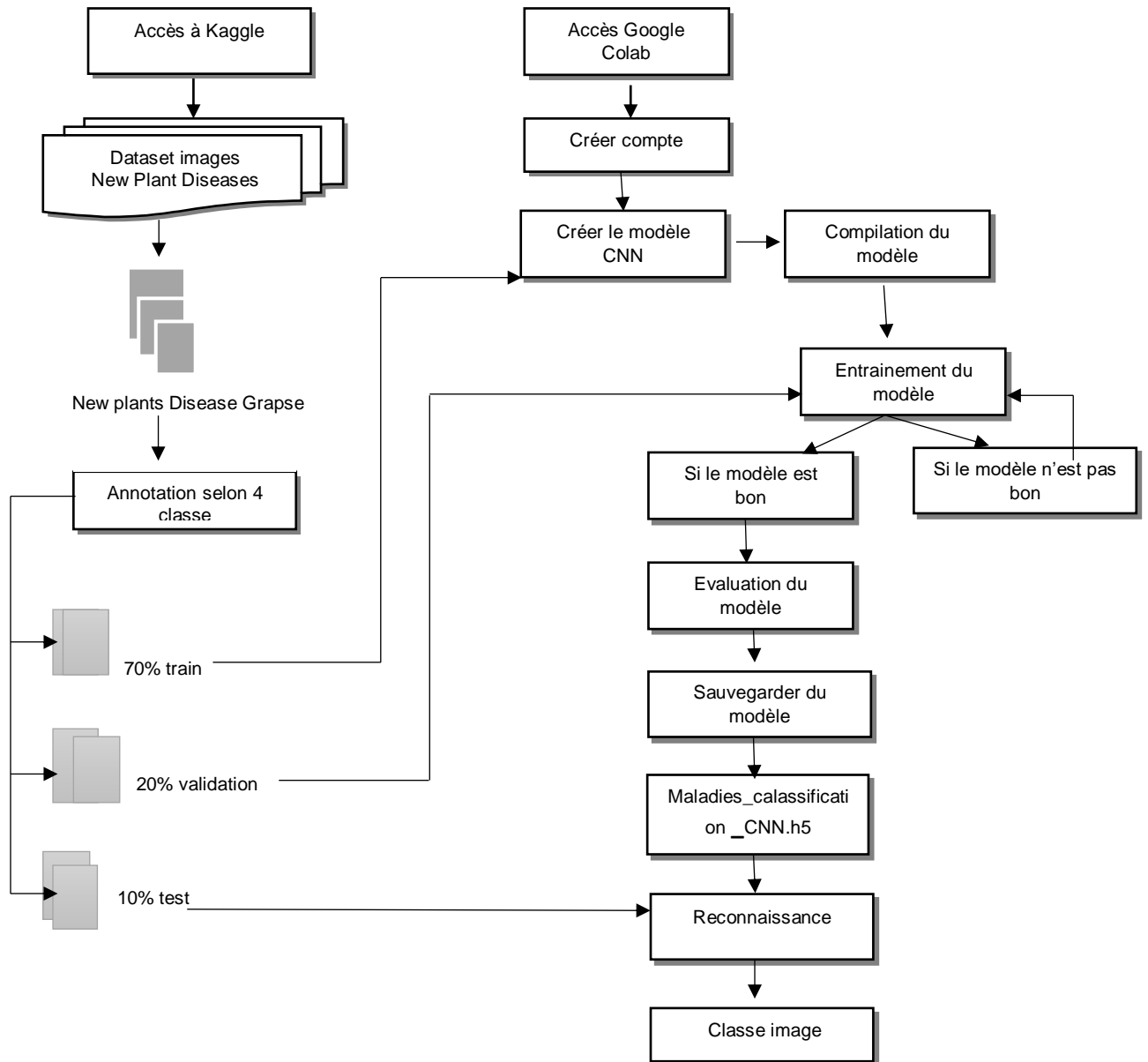


Figure 14 : Conception général de système.

### 4.3. Le prétraitement des images

#### ➤ Chargements de data set

Pour détecter les maladies affectant la vigne, nous nous sommes appuyés sur la source de données disponible sur la plateforme Kaggle. Plus précisément, nous avons mis en ligne une collection appelée « Nouvelles maladies des plantes »[28] qui rassemble des images de différentes plantes affectées par différentes maladies. Nous avons ensuite extrait un sous-ensemble de 2 500 images de feuilles de vigne exprimant différents types de maladies.

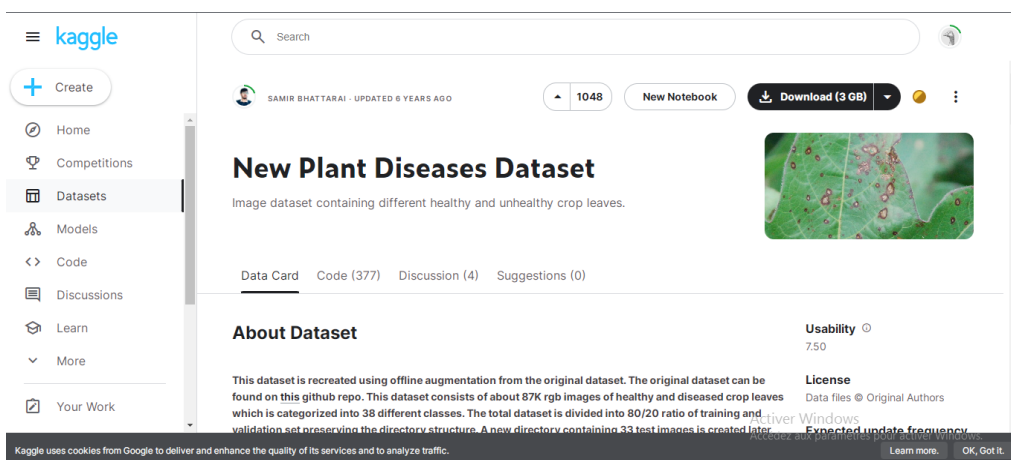


Figure 15 : Représentation de data set sur le Platform kaggle

### 4.4. L'apprentissage du modelé par CNN

Après le prétraitement de data set sont passées au modèle proposé pour apprentissage. j'ai développé un modèle de réseau de neurones convolutif (CNN) spécifiquement conçu pour la classification d'images de feuilles de vigne atteintes par différentes maladies en quatre catégories distinctes. Son architecture se compose de multiples couches organisées de manière séquentielle.

En entrée, une couche Conv2D permet d'injecter les images RGB aux dimensions spécifiées en hauteur et largeur. Cette couche applique des filtres sur les images pour détecter les premiers motifs et caractéristiques.

Ensuite, une série de couches cachées interviennent pour extraire des caractéristiques visuelles de plus en plus complexes. Celles-ci comprennent d'autres couches Conv2D qu'appliquent des filtres supplémentaires, ainsi que des couches MaxPooling2D qui réduisent la dimensionnalité spatiale tout en conservant l'essentiel. La normalisation BatchNormalization stabilise l'entraînement, tandis que le Dropout prévient le sur-apprentissage.

Les dernières couches convolutives Conv2D, plus profondes, permettent d'apprendre des traits caractéristiques de haut niveau. Finalement, une couche Dense entièrement connectée avec une activation Softmax produit les probabilités pour chacune des quatre classes de maladies de vigne, garantissant que leur somme soit égale à 1.

Cette architecture stratifiée combinant ces différentes couches offre une extraction optimale des caractéristiques discriminantes à partir des images, avec une réduction de dimensions et un contrôle du sur-apprentissage. Cela permet au modèle final de classifier avec précision les différentes pathologies affectant les plants de vigne dans les quatre catégories cibles de mon projet.

## **4.5. La reconnaissance des images**

Le processus de reconnaissance d'images par les CNN se déroule en plusieurs étapes interconnectées. Il débute par l'introduction d'une image dans le réseau, qui traverse ensuite une série de couches convolutives. Chaque couche est équipée de filtres spécialisés qui extraient des caractéristiques visuelles de plus en plus sophistiquées, allant des simples contours aux motifs complexes.

Après le traitement de l'image par les couches convolutives, les caractéristiques extraites sont agrégées et converties en un vecteur. Ce vecteur est alors acheminé vers la couche dense finale, également connue sous le nom de couche entièrement connectée. Cette couche calcule un score de probabilité pour chaque catégorie d'objets ou de maladies, basé sur les caractéristiques visuelles identifiées.

Enfin, la catégorie ayant obtenu le score le plus élevé est sélectionnée comme prédiction finale. Cette prédiction représente l'objet ou la maladie le plus probable présent dans l'image analysée. Ainsi, la reconnaissance d'images tire parti des capacités d'extraction de caractéristiques du CNN, associées aux calculs de probabilités de la couche de sortie dense, pour identifier avec précision le contenu visuel de l'image examinée.

## **4.6. Conclusion**

En définitive, ce chapitre a détaillé la conception et la mise en œuvre d'un système de prédiction des maladies affectant les vignes, en exploitant la puissance des réseaux de neurones convolutifs (CNN). Nous avons abordé l'architecture de ce modèle d'apprentissage profond, les techniques d'entraînement employées ainsi que la phase cruciale de prédiction.

Ce système novateur tire parti des capacités de l'apprentissage profond pour offrir une méthode à la fois efficace et précise de détection et de classification des maladies impactant les vignobles. Il contribue ainsi à une gestion plus proactive et mieux informée des cultures viticoles.

Cette technologie de pointe promet d'améliorer significativement la productivité agricole en réduisant les pertes liées aux maladies végétales. Elle représente une solution durable pour répondre aux défis de l'agriculture moderne, en fournissant aux viticulteurs un outil puissant pour préserver la santé de leurs vignes et optimiser leurs rendements.

En tirant parti de l'intelligence artificielle et plus particulièrement des réseaux neuronaux convolutifs , ce système ouvre la voie vers une viticulture plus résiliente, productive et respectueuse de l'environnement, mieux armée pour relever les défis phytosanitaires à venir.

# CHAPITRE 5 :

## IMPLÉMENTATION

---

### 5.1 Introduction

La reconnaissance d'image est l'une des tâches les plus importantes et courantes dans le domaine de la vision par ordinateur. Elle consiste à entraîner un modèle informatique pour identifier et classer automatiquement les objets, les motifs ou les caractéristiques présents dans une image.

Les réseaux neuronaux convolutifs (CNN) ont révolutionné la reconnaissance d'image en fournissant des performances exceptionnelles. Les CNN sont spécialement conçus pour extraire automatiquement des caractéristiques pertinentes à partir des images brutes, sans nécessiter une extraction manuelle de ces caractéristiques.

Dans ce chapitre nous allons définir notre architecture de CNN mise en point pour la reconnaissance des images des plantes, via le langage de programmation python. Nous commençons tout d'abord par la présentation des ressources, du langage et de l'environnement de développement que nous avons utilisé. Puis les étapes d'implémentation du système modèle et on termine par les tests et les résultats effectués.

### 5.2 Environnement de développement

#### Le matériel

Le matériel réalisé est PC personnel Dell avec un 8GB capacité mémoire, et un processeur Intel(R) Core(TM) i5-8365U CPU @ 1.60GHz 1.90 GHz , avec Windows 11 édition intégral, service pack 1 64 bits type système.

#### Les logiciels

- **Google Colab, Python, Bibliothèque**

Pendant de nombreuses années, Google a développé un outil de développement appelé Colaboratory (Google colab). Aujourd'hui, Google a rendu Colaboratory gratuit pour une utilisation publique. Google colab est un environnement de bloc-notes gratuit qui fonctionne entièrement dans le cloud. Il permet, ainsi qu'aux membres de l'équipe, de modifier des documents, de la même manière que travaillez avec Google Docs.

Colab prend en charge de nombreuses bibliothèques d'apprentissage automatique populaires qui peuvent être facilement chargées dans le carnet de notes. Avec Google colab, il suffit de quelques lignes de code pour importer un ensemble de données d'images, entraîner un classificateur d'images et évaluer le modèle.

Les notebooks Colab exécutent le code sur les serveurs en nuage de Google, ce qui signifie que les utilisateurs peuvent exploiter la puissance du matériel Google, notamment les GPU et les TPU, quelle que soit la puissance de leur machine. En utilisant un langage de programmation soit : C++ /Python [20].

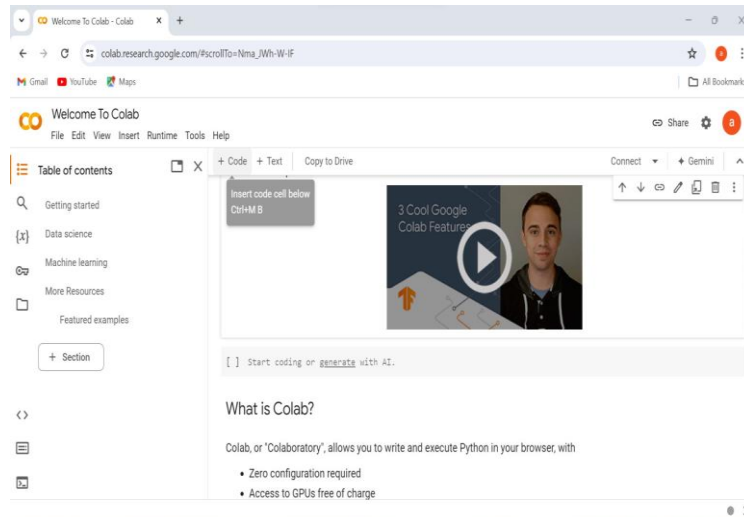


Figure 16 : L`environnement Google Colab

### **Pourquoi utiliser Google Colab ?**

Il y a plusieurs raisons pour l'adopter Google Colab au lieu d'une simple instance de Jupyter Notebook :

#### ▪ **Bibliothèques préinstallées**

La distribution Anaconda de Jupyter Notebook est livrée avec plusieurs bibliothèques de données préinstallées, telles que Pandas, NumPy, Matplotlib, ce qui est génial. Google Colab, quant à lui, fournit encore plus de bibliothèques d'apprentissage automatique préinstallées, telles que Keras, TensorFlow et PyTorch. Sauvegardé dans le Cloud.

Lorsque vous choisissez d'utiliser un simple carnet Jupyter comme environnement de développement, tout est enregistré sur votre machine locale. Si vous êtes prudent en matière de confidentialité, il s'agit peut-être d'une fonctionnalité que vous préférez. Toutefois, si vous souhaitez que vos carnets soient accessibles à partir de n'importe quel appareil avec une simple connexion Google, alors Google Colab est la solution.

Tous vos carnets Google Colab sont enregistrés sous votre compte Google Drive, tout comme vos fichiers Google Docs et Google Sheets.

#### ▪ **La fonction Collaboration**

La fonction de collaboration est une autre fonctionnalité intéressante de Google Colab. Si vous travaillez avec plusieurs développeurs sur un projet, l'utilisation de Google Colab note book est idéale. Tout comme vous collaborez sur un document Google Docs, vous pouvez coder avec plusieurs

développeurs à l'aide d'un bloc-notes Google Colab. En outre, vous pouvez également partager votre travail terminé avec d'autres développeurs.

- **Utilisation gratuite des GPU et TPU**

Google Research vous permet d'utiliser ses GPU et TPU dédiés pour vos projets personnels d'apprentissage automatique. D'après mon expérience, pour certains projets, l'accélération des GPU et TPU fait une énorme différence, même pour de petits projets.

En outre, comme il utilise les ressources de Google, les opérations d'optimisation du réseau neuronal ne perturbent pas mes processeurs [21].

- **Python3**

Python est un langage de programmation interprété, orienté objet, de haut niveau et doté d'une sémantique dynamique. Ses structures de données intégrées de haut niveau, combinées au typage dynamique et à la liaison dynamique, le rendent très attrayant pour le développement rapide d'applications, ainsi que pour une utilisation en tant que langage de script.

Python supporte les modules et les packages, ce qui encourage la modularité des programmes et la réutilisation du code. L'interpréteur Python et la bibliothèque standard étendue sont disponibles gratuitement sous forme de source ou de binaire pour toutes les principales plates-formes et peuvent être distribués librement [22].



Figure 17 : Python logo

- **TensorFlow**

TensorFlow est une plateforme open source de bout en bout pour l'apprentissage automatique. Elle dispose d'un système complet et flexible d'outils, de bibliothèques et de ressources communautaires qui permet aux chercheurs de faire avancer l'apprentissage automatique et aux développeurs de créer et de déployer facilement des applications basées sur l'apprentissage automatique [23].



Figure 18 :Tensorflow logo

### **Pourquoi TensorFlow ?**

TensorFlow a facilité l'apprentissage automatique : grâce aux modèles pré-entraînés, aux données et aux API de haut niveau, il est devenu facile pour tout le monde de construire des modèles ML. Principalement utilisé par les chercheurs : La plupart des chercheurs et des étudiants utilisent TensorFlow dans leurs recherches et la construction de modèles. Modèles prêts à l'emploi à des fins de production : TensorFlow prend en charge des modèles pré-entraînés qui peuvent être utilisés instantanément pour la production et l'expérimentation [23].

- **OpenCV**

OpenCV (Open Source Computer Vision Library) est une bibliothèque logicielle open source de vision par ordinateur et d'apprentissage automatique. OpenCV a été construit pour fournir une infrastructure commune pour les applications de vision par ordinateur et pour accélérer l'utilisation de la perception artificielle dans les produits commerciaux. Étant un produit sous licence BSD, OpenCV permet aux entreprises d'utiliser et de modifier facilement le code [24].



Figure 19 : OpenCV logo

### **Pourquoi OpenCV ?**

La bibliothèque compte plus de 2500 algorithmes optimisés, ce qui inclut un ensemble complet d'algorithmes de vision par ordinateur et d'apprentissage automatique classiques et de pointe.

Pour produire une image haute résolution d'une scène entière, trouver des images similaires dans une base de données d'images, Ces algorithmes peuvent être utilisés pour détecter et reconnaître des visages, identifier des objets, classer des actions humaines dans des vidéos, suivre les mouvements de la caméra, suivre des objets en mouvement, assembler des images supprimer les yeux rouges des images prises au flash, suivre les mouvements des yeux, reconnaître des paysages et établir des marqueurs pour les superposer à la réalité augmentée, etc[25].

- **Keras**

Keras est une API d'apprentissage profond écrite en Python, fonctionnant au-dessus de la plateforme d'apprentissage automatique TensorFlow. Elle a été développée dans le but de permettre une expérimentation rapide. La capacité de passer de l'idée au résultat aussi rapidement que possible est essentielle pour mener de bonnes recherches [26].



Figure 20 :Keras logo

#### ▪ **Kaggle**

Kaggle, une filiale de Google LLC, est une communauté en ligne de spécialistes des données et de l'apprentissage automatique. Kaggle permet aux utilisateurs de trouver et de publier des ensembles de données, d'explorer et de construire des modèles dans un environnement de science des données basé sur le Web, de travailler avec d'autres scientifiques des données et ingénieurs d'apprentissage automatique, et de participer à des concours pour résoudre des défis de science des données [26].



Figure 21 :Kaggle logo

#### • **Bibliothèque**

```
[ ] # Import necessary libraries
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.utils import to_categorical, Sequence
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
from sklearn.preprocessing import LabelEncoder
from tensorflow.keras.preprocessing.image import load_img, img_to_array
import seaborn as sns
from sklearn.metrics import confusion_matrix
```

Figure 22 : Quelques bibliothèques dans le modèle.

## 5.3. Les interface du système

### 5.3.1. Le data set images

Afin de développer une application à base de réseaux de neurones, il est nécessaire de disposer de deux data sets : un data set pour effectuer l'apprentissage et une autre data set pour tester le réseau obtenu et déterminer ses performances.

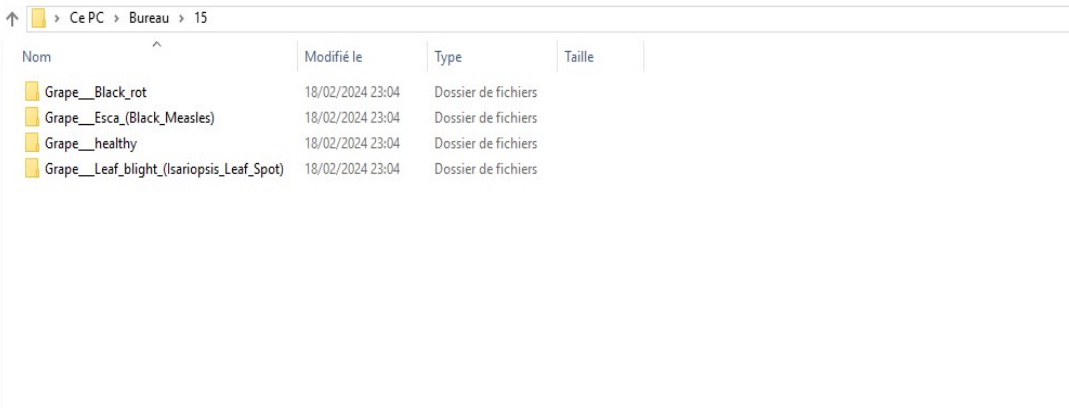
Dans notre cas, le data set est acquis à partir d'un site Web bien connu (Kaggle), il contient un total de 2500 images elles sont ordonnées dans 4 classes.

Nous allons maintenant présenter les résultats obtenus grâce à des expériences réalisées par le modèle de notre approche de classification des images des maladies en quatre classes (class 'Grape\_Black\_rot', class 'Grape\_Esca\_(Black\_Measles)', class 'Grape\_healthy' et class 'Grape\_Leaf\_blight\_(Isariopsis\_Leaf\_Spot)'). On utilise deep learning et la méthode de classification CNN.

Le schéma de notre système est le suivant :

1. Création des classes.
2. La classification et l'apprentissage.
3. Le test.

Voici les catégories utilisées dans notre base de données :



The screenshot shows a Windows File Explorer window with the address bar displaying 'Ce PC > Bureau > 15'. The main area contains a table with four rows, each representing a folder. The columns are 'Nom', 'Modifié le', 'Type', and 'Taille'.

Nom	Modifié le	Type	Taille
Grape__Black_rot	18/02/2024 23:04	Dossier de fichiers	
Grape__Esca_(Black_Measles)	18/02/2024 23:04	Dossier de fichiers	
Grape__healthy	18/02/2024 23:04	Dossier de fichiers	
Grape__Leaf_blight_(Isariopsis_Leaf_Spot)	18/02/2024 23:04	Dossier de fichiers	

Figure 23 : Les 4 classes dans le data set

#### Création des classes

Nous allons rechercher les images des mêmes classes des maladies, à savoir Grape\_\_Black\_rot, Grape\_\_Esca\_(Black\_Measles), Grape\_\_Leaf\_blight\_(Isariopsis\_Leaf\_Spot) et Grape\_\_healthy. Nous allons ensuite classer les fichiers en vue de la préparation de l'apprentissage.

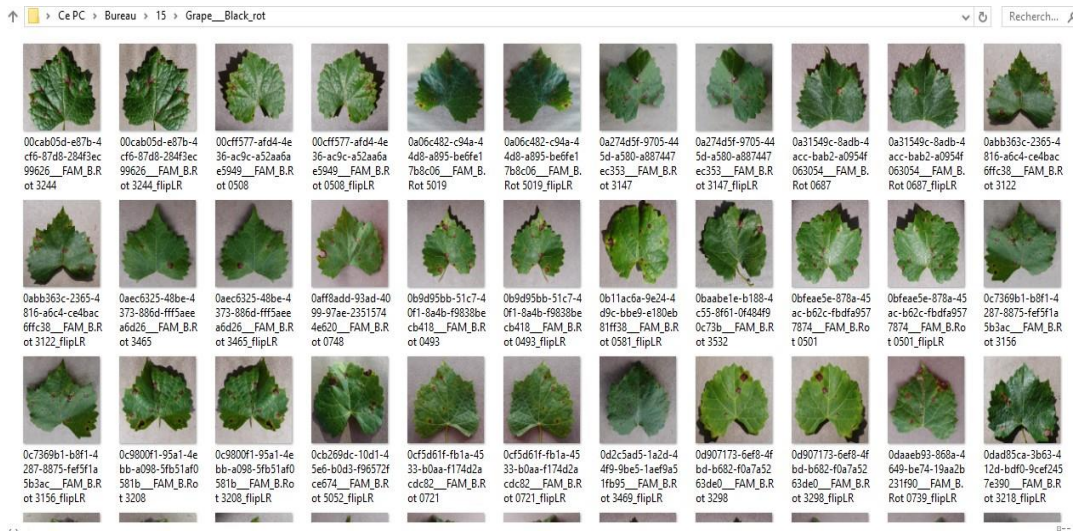


Figure 24 : Création de classes (classe 'Grape Black rot')

### 5.3.2. Prétraitement : L'annotation

Afin de préparer ces données pour l'entraînement de notre modèle de réseau de neurones convolutifs (CNN), nous avons entrepris une étape cruciale d'annotation sur la plateforme Roboflow. Cela nous a permis de classer les images de feuilles de vigne affectées en quatre catégories distinctes.

Par la suite, nous avons scindé cet ensemble annoté en trois sous-groupes pour assurer un entraînement optimal et une évaluation rigoureuse :

- Un échantillon d'entraînement de 1744 images (70%) dédié à l'apprentissage du modèle.
- Un échantillon de validation de 470 images (20%) pour ajuster les hyperparamètres pendant l'entraînement.
- Un échantillon test de 246 images (10%) réservé pour tester les performances finales du modèle entraîné.

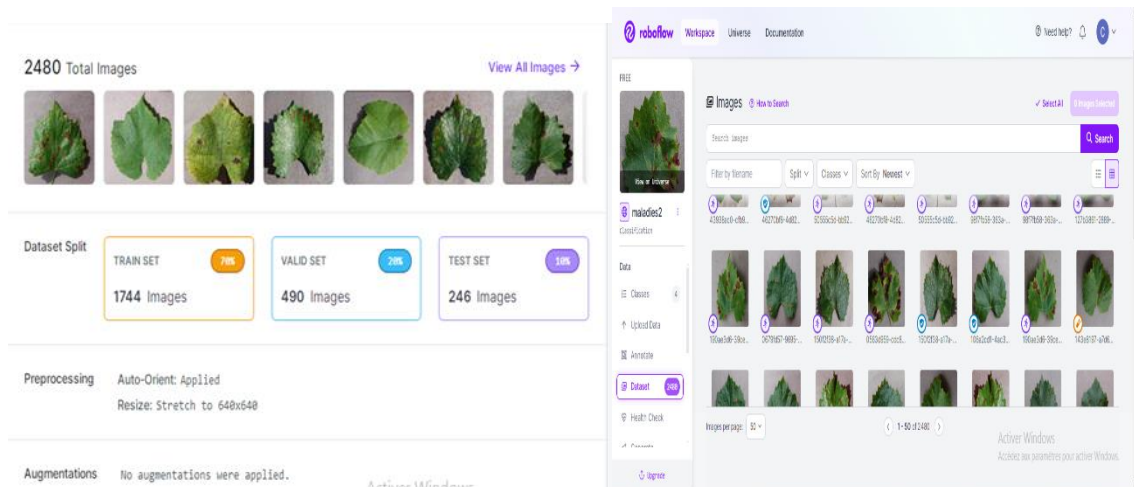


Figure 25 : Représentation de data set annoté sur le Platform Roboflow

### 5.3.3. La configuration

Ce réseau neuronal convolutif est conçu pour traiter des images en couleur. Il commence par une couche de convolution avec 32 filtres de taille 3x3, utilisant la fonction d'activation ReLU. La forme d'entrée est adaptée aux dimensions spécifiques de l'image et ses trois canaux de couleur. Cette couche est suivie d'une couche de max-pooling avec une fenêtre 2x2 pour réduire les dimensions spatiales.

Ensuite, le réseau enchaîne avec une deuxième couche de convolution de 64 filtres 3x3 et activation ReLU, suivie d'une autre couche de max-pooling 2x2. Ce motif se répète deux fois de plus, avec des couches de convolution de 128 filtres chacune, toujours suivies de max-pooling 2x2.

Après ces opérations de convolution et de pooling, les données sont aplaties par une couche Flatten. Elles passent ensuite par une couche dense de 512 unités avec activation ReLU, suivie d'une couche de dropout avec un taux de 0,5.

Enfin, le réseau se termine par une couche dense d'une seule unité avec une fonction d'activation sigmoïde, ce qui le rend adapté à une tâche de classification binaire.

Cette architecture permet d'extraire progressivement des caractéristiques de plus en plus complexes de l'image d'entrée, puis de les utiliser pour effectuer une classification finale.

```
# Define CNN architecture
model = Sequential()

model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(img_width, img_height, 3)))
model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D((2, 2)))

model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D((2, 2)))

model.add(Flatten())

model.add(Dense(512, activation='relu'))
model.add(Dropout(0.5))

model.add(Dense(1, activation='sigmoid'))
```

Figure 26 : Construction du modèle de réseau de neurones convolutif.

- **Compilation du Modèle**

La compilation du modèle utilise **Adam** comme optimiseur, **categorical\_crossentropy** comme fonction de perte pour la classification multi-classes, et **accuracy** comme métrique à suivre. Cette configuration prépare le réseau neuronal pour l'entraînement

```
# Compilation du modèle
model.compile(optimizer='adam',
              loss='categorical_crossentropy', # Utilisation de la perte 'categorical_crossentropy' pour plusieurs classes
              metrics=['accuracy'])
```

Figure 27 : Compilation du modèle

#### 5.3.4. Le modèle

Le modèle proposé a été complété par une interface graphique simple permettant son exécution facile.

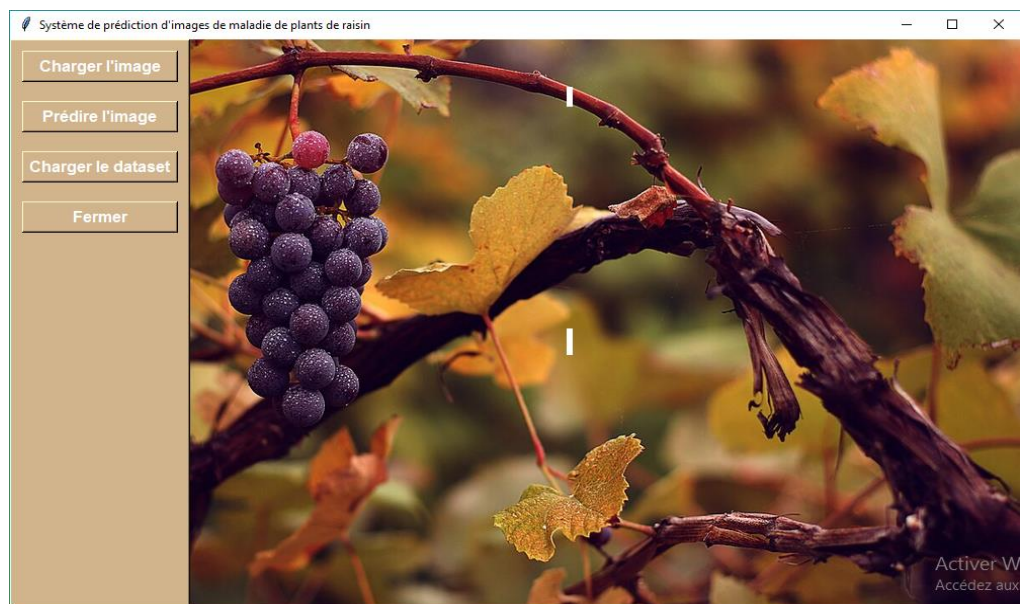


Figure 28 : la fenêtre principale de l'application.

- **Chargement d'image** : L'utilisateur sélectionne une image qui est redimensionnée et affichée dans l'interface.
- **Prédiction** : Après le chargement, l'utilisateur active la prédiction. Le modèle analyse l'image et affiche le résultat.

- **Chargement du dataset** : Cette action ouvre une fenêtre présentant un tableau des données de test.
- **Fermer**

### 5.3.5. La reconnaissance des images

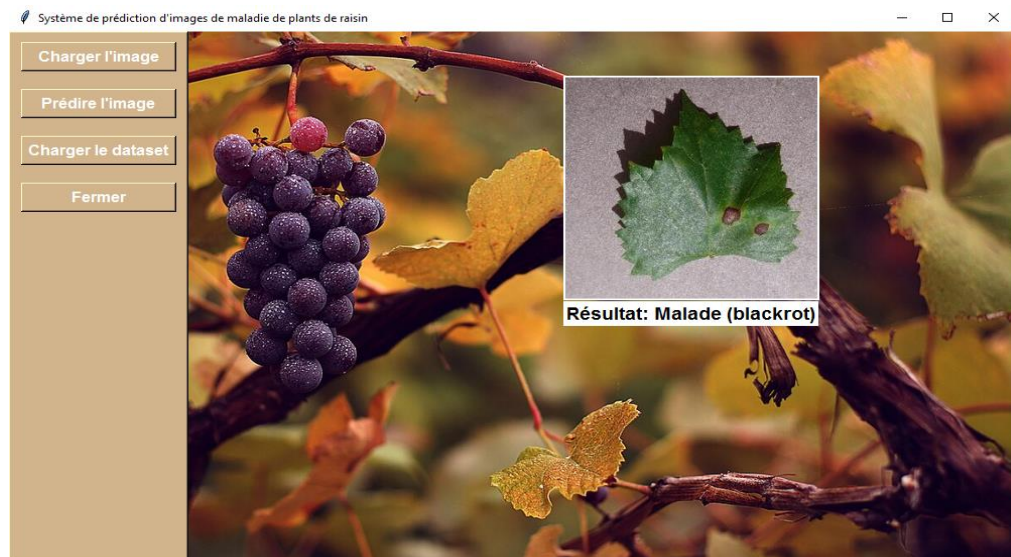


Figure 29 : Résultats de la reconnaissance des images.

## 5.4. Résultats et Discussion

### 5.4.1. Les métriques de performance utilisées

- **Exactitude( Accuracy)**

La précision (Accuracy) est une méthode permettant de mesurer les performances d'un modèle de classification. Il est généralement exprimé en pourcentage. La précision est le nombre de prédictions où la valeur prédite est égale à la valeur vraie. C'est binaire (vrai/faux) pour un échantillon particulier. La précision est souvent représentée graphiquement et surveillée pendant la phase de formation, bien que la valeur soit souvent associée à la précision globale ou finale du modèle. La précision est plus facile à interpréter que la perte [27].

- **Perte(LOSS)**

Une fonction de perte(LOSS), également appelée fonction de coût, prend en compte les probabilités ou l'incertitude d'une prédiction en fonction de l'écart entre la prédiction et la valeur réelle. Cela nous donne une vision plus nuancée des performances du modèle [27].

### 5.4.2. Résultats et discussion

Une fois le modèle entraîné, il est crucial de procéder à une évaluation rigoureuse de ses performances. Cette étape est indispensable pour garantir son bon fonctionnement et la fiabilité de ses

résultats. Sans ces vérifications approfondies, il serait imprudent de déployer le modèle dans des situations réelles.

- **Résultat de l'entraînement**

Un système de classification fondé sur l'apprentissage profond a été élaboré. Ce modèle a fait l'objet de multiples cycles d'entraînement dans le but d'optimiser ses capacités. L'étape finale, d'une importance capitale, consiste à effectuer une analyse approfondie des résultats de cet entraînement intensif. Cette démarche est indispensable pour s'assurer que le système atteint les niveaux requis de fiabilité et d'efficacité avant son déploiement dans des conditions réelles. La vérification minutieuse de la qualité des résultats constitue ainsi le socle garantissant la réussite du modèle dans ses applications pratiques.

D'après l'entraînement nous avons obtenu les résultats illustrés dans la figure 30.

```
Epoch 40/40  
54/54 [=====] - 168s 3s/step - loss: 0.1439 - accuracy: 0.9573 - val_loss: 5.6740 - val_accuracy: 0.4250
```

Figure 30 : Les résultats d'évaluation du modèle.

- **Évaluation des résultats de la perte (Loss)**

Les résultats affichés dans le graphique démontrent l'évolution de la fonction de perte (Loss) par rapport au nombre d'itérations effectuées, et ce pour les deux phases cruciales : la validation et l'entraînement. Cette représentation visuelle permet de suivre la progression de la performance du modèle tout au long de son apprentissage, offrant ainsi un aperçu détaillé de l'efficacité du processus à chaque étape.

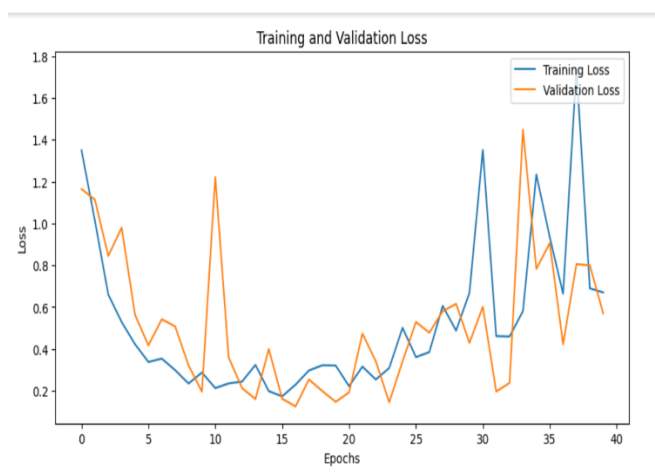


Figure 31 : La courbe d'erreur du modèle.

D'après les illustrations de la figure, on observe que les pertes d'entraînement et de validation évoluent de manière dynamique au fil des époques. Les courbes présentent une tendance générale à la

baisse jusqu'à l'époque 15 environ, suivie d'une remontée progressive et irrégulière. Cette variabilité, particulièrement prononcée après l'époque 25, peut être attribuée à l'utilisation du mécanisme de Drop-Out, une technique visant à prévenir le sur apprentissage.

Bien que les valeurs ne se maintiennent pas à des niveaux constamment optimaux, les fluctuations observées indiquent que le modèle continue d'apprendre et de s'ajuster. Les courbes de perte d'entraînement et de validation suivent des trajectoires similaires, suggérant une cohérence dans l'apprentissage du modèle.

Il est à noter que la valeur moyenne de la perte sur l'ensemble des époques est d'environ 0.54, ce qui représente un équilibre entre la performance du modèle et sa capacité à généraliser sur de nouvelles données.

- **Évaluation des résultats de la précision (Accuracy)**

Le résultat de la précision du modèle a été testé à une fraction de 10 % de .Des données réservées à cet effet, il ressort : 91,63

Score de précision en fonction du nombre d'itérations .La vérification et la formation sont illustrées dans la figure

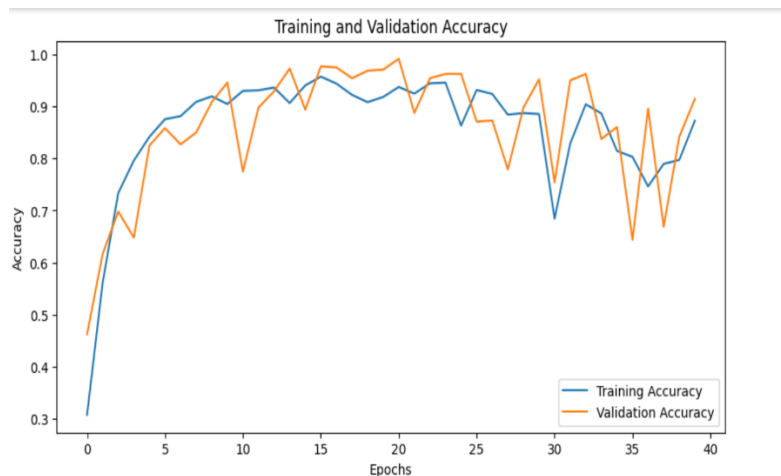


Figure 32 : La courbe de précision du modelé.

- **Evaluation du résultat de la matrice de confusion**

La matrice de confusion présentée dans la figure 33 offre une vue détaillée des performances du modèle de classification des maladies de la vigne. Chaque cellule de cette matrice représente une intersection spécifique entre une classe prédite et une classe réelle.

Les valeurs numériques dans chaque cellule indiquent le nombre d'échantillons correspondant à cette combinaison particulière de prédiction et de réalité. Cette représentation permet une analyse approfondie de la précision du modèle pour chaque catégorie.

Un élément clé de cette matrice est sa diagonale principale, qui s'étend du coin supérieur gauche au coin inférieur droit. Les cellules situées sur cette diagonale sont d'une importance cruciale car elles représentent les cas où la prédiction du modèle correspond exactement à la classe réelle. Ainsi, les nombres figurant dans ces cellules diagonales indiquent le compte des prédictions correctes pour chaque catégorie de maladie ou d'état de santé de la vigne.

Cette structure permet non seulement d'évaluer la précision globale du modèle, mais aussi d'identifier spécifiquement les forces et les faiblesses de la classification pour chaque type de maladie ou condition de la vigne.

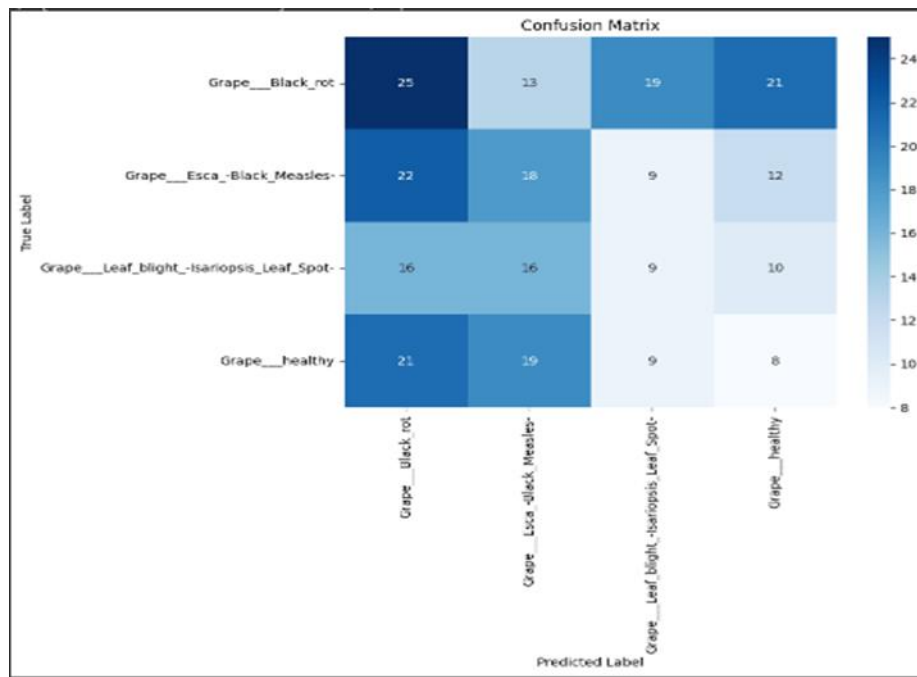


Figure 33 : La matrice de confusion du modèle.

## 5.5. Conclusion

Dans ce chapitre, nous avons fourni les outils indispensables à la réalisation de notre travail et présenté l'environnement de développement. Nous avons également décrit notre modèle en incluant quelques captures d'écran pour illustrer le déroulement et le fonctionnement de notre approche, ainsi que les résultats obtenus. Un exemple de reconnaissance d'images utilisant notre méthode a été donné. Enfin, nous avons démontré l'impact de la méthode choisie (basée sur les réseaux de neurones convolutifs, CNN) sur la qualité des résultats de la reconnaissance par classification.

# CONCLUSION GÉNÉRALE

---

La reconnaissance d'images à l'aide des réseaux de neurones convolutifs (CNN) est un domaine de recherche en plein essor, notamment pour ses applications variées et ses résultats prometteurs. Ce mémoire s'est concentré sur l'utilisation des CNN pour la reconnaissance des maladies des plantes, un domaine crucial pour l'agriculture et la sécurité alimentaire mondiale.

Les travaux réalisés ont démontré que les CNN sont particulièrement efficaces pour l'identification et la classification des maladies des plantes à partir d'images. Grâce à leur capacité à extraire et analyser automatiquement des caractéristiques complexes des images, les CNN surpassent les méthodes traditionnelles de reconnaissance d'image. Nos expérimentations ont permis d'atteindre des taux de précision élevés, validant ainsi l'approche proposée.

En conclusion, cette étude a prouvé l'efficacité des CNN pour la reconnaissance des maladies des plantes, offrant ainsi une solution potentielle pour les agriculteurs et les chercheurs dans le domaine de la phytopathologie. L'adoption de telles technologies peut contribuer à une détection précoce et précise des maladies, réduisant ainsi les pertes de récoltes et les coûts associés aux traitements phytosanitaires. De plus, cette approche peut être étendue à d'autres domaines de la reconnaissance d'image, ouvrant la voie à de nombreuses applications innovantes.

La recherche future pourrait se concentrer sur l'implémentation de ces modèles dans des dispositifs portables et des applications mobiles, facilitant ainsi leur utilisation sur le terrain. Ainsi, les réseaux de neurones convolutifs s'imposent comme un outil puissant et indispensable pour l'avenir de l'agriculture intelligente.

# Bibliographie

---

- [ 1 ] Définition des images :  
<https://fr.wikipedia.org/wiki/Image#:~:text=Une%20image%20est,genre1.%C2%A0%C2%BB>
- [ 2 ] Les types des images :  
[.https://www.perplexity.ai/search/les-images-acfn7qlw5SQykJ6PSV.WdYg#:~:text=une%20image%20%C3%A0%20deux%20niveaux%20peut,par%20ordinateur%20et%20le%20traitement%20d%27images](https://www.perplexity.ai/search/les-images-acfn7qlw5SQykJ6PSV.WdYg#:~:text=une%20image%20%C3%A0%20deux%20niveaux%20peut,par%20ordinateur%20et%20le%20traitement%20d%27images).
- [ 3 ] B. Guerfi.: représentation d'une image numérique, Université Ferhat Abbas — Sétif département des sciences de la terre, Géographie et Aménagement du Territoire ,Enseignante 2020 .
- [ 4 ] R.Gonzalez, R.Woods, , Traitement d'images numériques, De Boeck Supérieur ,université de Tennessee, 2008
- [ 5 ] Les niveaux de gris :[https://www.fr.m.wikibooks.org/wiki/Fichier:Bits\\_et\\_niveaux\\_de\\_gris.svg](https://www.fr.m.wikibooks.org/wiki/Fichier:Bits_et_niveaux_de_gris.svg)
- [ 6 ] Représentation des pixels. Définition d'une image :<https://www.ionos.fr/digitalguide/sites-internet/web-design/quest-ce-quun-pixel>
- [ 7 ] La résolution :<https://www.ultimate-photo-tips.com/what-is-a-pixel.html>
- [ 8 ] Résolution d'une image numérique : <https://www.nicolasroger.fr/2015/01/definition-et-resolution-dune-image-numerique/>
- [ 9 ] Le profondeur d'une image : [https://blog.dehesdin.com/workflow/formats-profil-couleur-et-profondeur-8-ou-16-bits-dune-image/#:~:text=2\)%20La%20profondeur%20d'une%20image%3A&text=La%20profondeur%20d%C3%A9finit%20la%20quantit%C3%A9,bits%20d'informations%20par%20pixel](https://blog.dehesdin.com/workflow/formats-profil-couleur-et-profondeur-8-ou-16-bits-dune-image/#:~:text=2)%20La%20profondeur%20d'une%20image%3A&text=La%20profondeur%20d%C3%A9finit%20la%20quantit%C3%A9,bits%20d'informations%20par%20pixel).
- [ 10 ] Fausett ,fundamentals of neural networks:architectures,algorithm and application. 1993
- [ 11 ] S.Patil, U.Kulkarni. Accuracy Prediction for Distributed Decision Tree using Machine Learning approach,3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019.
- [ 12 ] K.Gurney. introduction to neural networks . 1997
- [ 13 ] Les arbres de décision :<https://datasciencetoday.net/index.php/fr/machine-learning/109-ml-sup/178-les-arbres-de-decision-introduction>. (2021). Retrieved from datasciencetoday.net.
- [ 14 ] S.BEGGARI ,K.KHAMRA, Système de reconnaissance de visage par un réseau de neurone convolutionnel(CNN),2017
- [ 15 ] W.Mages .Architecture générale d'un CNN ,2019

- [ 16 ] A.Carlos ,Deep learning for biological image classification ,2017
- [ 17 ] S.Srdjan,Deep Neural Networks Based Recognition of plant diseases by leaf images clasification ,2016
- [ 18 ] S.Anubha Pearlina,A study on plant recognition using conventional images processing and deep learning approaches ,2019
- [ 19 ] <https://www.medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>.
- [ 20 ] F. Nacer «Reconnaissance d’expression faciale à partir d’un visage réel» Mémoire de master, Université de 8/Mai/19450, Guelma , 2019
- [ 21 ] Utilisation gratuite des GPU et TPU : Machin learning.<https://colab.research.google.com/notebooks/intro.ipynb#scrollTo=OwuxHmxllTwN>
- [ 22 ] Reasons Why You Should Use Google Colab for Your Next Project. Disponible à l’adresse <https://towardsdatascience.com/4-reasons-why-you-should-use-google-colab-for-your-nextprojectb0c4aaad39ed>
- [ 23 ] What is Python? Executive Summary <https://www.python.org/doc/essays/blurb/>
- [ 24 ] Why TensorFlow is So Popular – Tensorflow Features Disponible à l’adresse <https://www.geeksforgeeks.org/why-tensorflow-is-so-popular-tensorflow-features>
- [ 25 ] OpenCv. <https://opencv.org/about/>
- [ 26 ] N. Xavier et Y.Victor Compétition Kaggle : “Home Depot Product Search Relevance” , 2016
- [ 27 ] I.Goodfellow,Y.Bengio & A.Courville , Deep Learning. MIT Press.,2016
- [ 28 ] Nouvelles maladies des plantes : <https://www.kaggle.com/datasets/vipooooool/new-plant-diseases-dataset>